Modeling and Shaping Human-Machine Interactions in Closed-loop, Co-adaptive Neural Interfaces

Manchali Maneeshika Madduri

A document
submitted in partial fulfillment of the
General Exam requirements for the degree of

Doctor of Philosophy

University of Washington

2024

Reading Committee:

Amy L. Orsborn, Chair

Samuel A. Burden, Chair

Kimberly A. Ingraham

Program Authorized to Offer Degree:

Electrical & Computer Engineering

University of Washington

Abstract

Modeling and Shaping Human-Machine Interactions in Closed-loop, Co-adaptive Neural Interfaces

Manchali Maneeshika Madduri

Co-Chairs of the Supervisory Committee:

Amy L. Orsborn

Electrical & Computer Engineering

Samuel A. Burden

Electrical & Computer Engineering

Neural interfaces map biological signals measured from a user to control commands for external devices. The mapping from biosignals to device inputs is performed by the decoder. Adaptation of both the user and decoder—*co-adaptation*—provides opportunities to improve the accessibility and usability of interfaces across diverse users and applications. User learning leads to robust interface control that can generalize across environments and contexts. Decoder adaptation can individualize interfaces and account for signal variability. Co-adaptation therefore creates opportunities to shape the user and decoder system to achieve robust and generalizable personalized interfaces. However, co-adaptation creates a two-learner system with dynamic interactions between the user and decoder. Engineering co-adaptive interfaces requires new tools and frameworks to achieve stable user-decoder interactions.

This thesis aims to develop and experimentally validate methods for designing and measuring co-adaptive interfaces. I present new computational methods based on control theory and game theory to analyze and generate predictions for user-decoder co-adaptive outcomes in continuous interactions. I tested these computational methods using an experimental platform where human participants learn to control a cursor using an adaptive myoelectric interface to track a target on a computer display. Our framework predicted the outcome of co-adaptive interface interactions and revealed how interface properties can shape user behavior. These findings contribute new tools to design personalized, closed-loop, co-adaptive neural interfaces. The overarching aim of this thesis is to propose co-adaptive analysis and tools that can predictably influence co-adaptive interface performance and user-decoder dynamics. These findings from this thesis contribute new tools to design personalized, closed-loop, co-adaptive neural interfaces.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Figure Number | Page

LIST OF TABLES

ACKNOWLEDGMENTS

I entered my PhD in the Fall of 2019 generally knowing that I wanted to learn about brain-computer interfaces, but completely unsure about how to get there and whether I should have returned to school at all. These past few years have been challenging yet rewarding, and I certainly would not have been made it without the support and guidance of several individuals. I would like to take this opportunity to thank them here.

First and foremost, thank you to my advisors, Professor Amy Orsborn and Professor Sam Burden. I certainly would not have finished this without your invaluable research guidance, scientific enthusiasm, and continual encouragement. Thank you for always being willing to support me and discuss all the questions, doubts, and concerns I had along the way — whether they were about careers, research methodologies, or color schemes. I truly appreciate how much both of you have pushed me to grow and expanded my ability to think. The attention and tenacity with which both of you approach research is inspirational, and I continually aspire to reach your rigor, level of focus, and figure-making skills!

None of this work would have been possible without my collaborators and labmates. Firstly, I would like to thank Momona Yamagami for her incredible research input, her continual support and encouragement, and all the glorious Seattle hikes. Momona, you have been a mentor, collaborator, and friend, all in one. Thank you to my partners in EMG experimentation: Amber Chou and Si Jia Li. It's been such a joy collaborating, brainstorming, and generally hanging out with both of you. To Gus Millevolte and Sasha Burckhardt, thank you for your tireless efforts in developing and learning how to run human-subject experiments with me. Thank you to Pavi Rajeswaran and Leo Scholl for all the walks, pep talks, and time both of you have dedicated to helping not just me, but all members of our lab. I don't know if any of us would have made it without the two of you! To Andrew Pace, Ben Chasnov, and Joey Sullivan, thank you for all the discussions on game theory and dynamical systems and for all the words of encouragement along the way. It's been a privilege to work with and learn from all the members of Orsborn and Burden Labs, and I'm excited to see what the future holds for each one of you.

Thank you to those who have celebrated the wins with me and listened to me complain over coffee or runs. To the members of the UW community, Nina Vincent, Yana Sosnovskaya, Nivii Kalavakonda, Shruti Misra, Melody Mojib, and Andy Lewis, and many others along the way — thank you for making my experience at UW and in Seattle so joyful. Thank you, especially, to Dan Calderone for explaining game theory to me on many napkins in many restaurants, running stairs with me, and for supporting me throughout everything. To Shruthi, Katie, Laura, Kim, Liz – thank you for being there for all the ups and downs, for making sure I stayed sane during lockdowns, and for always only being a text or call away.

Lastly, thank you so much to my inspiration: my family. To my Mom, Dad, Achintya, Gayatri, Indu, and Tejas—thank you for continually encouraging me, always believing in me, and keeping me grounded. To Saketh and Maggie, thank you for making sure I ate three meals a day, reminding me to take regular walks, and for always being the highlight of my day. I absolutely wouldn't be where I am without all of you.

PUBLICATIONS RESULTING FROM THIS THESIS

Game Theory Models (Chapter 2)

Journal articles and peer-reviewed conference proceedings

Madduri, M. M., Burden, S. A. and Orsborn, A. L. A Game-Theoretic Model for Co-Adaptive Brain-Machine Interfaces. In 10th International IEEE/EMBS Conference on Neural Engineering (NER) 327–330 (2021). doi:10.1109/NER49283.2021.9441081.

Conference abstracts

Madduri, M. M., Burden, S. A. and Orsborn, A. L. (2020). Simulating Neural Dynamics in a Closed-Loop Adaptive Decoder Brain-Machine Interface. Neural Control of Movement, Remote, May 2020.

Madduri, M. M., Burden, S. A. and Orsborn, A. L. (2021). A Game-Theoretic Model for Co-Adaptive Brain-Machine Interfaces. IEEE Conference of Neural Engineering and Rehabilitation, Remote, May 2021.

Madduri, M. M., Burden, S. A. and Orsborn, A. L. (2022). A Game-Theoretic Model for Co-Adaptive Brain-Machine Interfaces. University of Washington ECE Research Showcase, Seattle, WA, March 2022.
Received Award for Best Theoretical Research Student Poster

Co-adaptive Myoelectric Interfaces (Chapter 3)

Journal articles and peer-reviewed conference proceedings

Madduri, M.M., Yamagami, M., Millevolte, A.X.T, Li, S.J., Burckhardt, S., Burden, S.A. and Orsborn, A.L. Co-Adaptive Myoelectric Interface for Continuous Control. In 4th IFAC Workshop on Cyber-Physical Human Systems (2022).
Selected as Finalist for Best Student Paper Award

Conference abstracts

Madduri, M.M, Yamagami, M., Millevolte, A.X.T, Li, S.J., Burckhardt, S., Burden, S.A. and Orsborn, A.L. (2022). Modeling User-Decoder Learning Dynamics in Co-adaptive Myoelectric Interfaces for Continuous Control. Neural Control of Movement, Dublin, Ireland, July 2022.

Madduri, M.M, Yamagami, M., Millevolte, A.X.T, Li, S. Burckhardt, S., Burden, S.A. and Orsborn, A.L. (2022). User-Decoder Learning Dynamics in Co-adaptive Myoelectric Interfaces. Whitaker Neuroengineering Workshop, Cambridge, UK, August 2022.

Madduri, M.M, Yamagami, M., Millevolte, A.X.T, Li, S.J. Burckhardt, S., Burden, S.A. and Orsborn, A.L. (2022). Co-Adaptive Myoelectric Interface for Continuous Control. Neural Computation and Engineering Connection, Seattle, WA, May 2022.

Pfister, A., Madduri, M.M., Chou, A.H., and Burden, S.A. (2023). Matching User and Machine Learning Rates in Co-adaptive Closed-Loop Myoelectric Interfaces. IEEE Conference on Neural Engineering and Rehabilitation, Baltimore, Maryland, December 2023.

Human-Machine Interactions in Co-adaptive Interfaces (Chapter 4 and 5)

Journal articles and peer-reviewed conference proceedings

Madduri, M. M., Yamagami, M., Li, S. Burckhardt, S., Burden, S.A. and Orsborn, A.L. (2024). Predicting and Shaping Human-Machine Interactions in Closed-loop, Co-adaptive Neural Interfaces. bioRxiv (2024). doi: 10.1101/2024.05.23.595598

Conference abstracts

Madduri, M. M., Yamagami, M., Li, S. Burckhardt, S., Burden, S.A. and Orsborn, A.L. (2023). User Learning Across Decoder Parameters in Co-Adaptive Myoelectric Interfaces. Neural Control of Movement, Victoria, Canada, July 2023.

Madduri, M. M., Yamagami, M., Li, S. Burckhardt, S., Burden, S.A. and Orsborn, A.L. (2023). User Learning Across Decoder Parameters in Co-Adaptive Myoelectric Interfaces. Neural Control of Movement, Victoria, Canada, July 2023.

Madduri, M. M., Yamagami, M., Li, S. Burckhardt, S., Burden, S.A. and Orsborn, A.L. (2024). Shaping Human-Machine Interactions in Closed-Loop Myoelectric Interfaces via Game-Theoretic Models. Gordon Conference on Neuroelectric Interfaces, Galveston, Texas, March 2024.

Other

Review articles

Madduri, M. M., Burden, S. A. and Orsborn, A. L. Biosignal-based co-adaptive user-machine interfaces for motor control. Current Opinion in Biomedical Engineering 27, 100462 (2023).

Journal articles and peer-reviewed conference proceedings

Yamagami, M., Madduri, M.M., Chasnov, B., Chou, A.H.Y., Peterson, L.N. and Burden, S.A. Co-Adaptation Improves Performance in a Dynamic Human-Machine Interface. bioRxiv (2023). doi:10.1101/2023.07.14.549053.

Madduri, M.M.*, Chou, A.H.Y, *, Li S.J., Burckhardt S., Christensen, A., Hutchison, F., Orsborn, A.L., and Burden, S.A. Design Principles for Co-adaptive, Multimodal Interfaces. In SIGCHI, PhysioCHI Workshop 2024.

Chou, A. H., Madduri, M.M.*, Li, S.J.*, Christensen, A., Burden, S.A., Orsborn, A.L. Using Eye Gaze to Train an Adaptive Myoelectric Interface. bioRxiv (2024).

S.J. Li, M.M. Madduri, A.H.Y. Chou, A.L. Orsborn, Influencing Task Performance in Novel Hybrid Myoelectric Interfaces Through Decoder Adaptation. In preparation.

Conference abstracts

Chou, A. H., Madduri, M.M.*, Li, S.J.*, Christensen, A., Burden, S.A., Orsborn, A.L. (2024) Using Eye Gaze to Train an Adaptive Myoelectric Interface. Neural Control of Movement, May 2024.

Li, S., Madduri, M., Millevolte, A.X.T., Burckhardt, S., Burden, S.A., Orsborn, A.L. (2023). Design of a Myoelectric Co-Adaptive Decoder for Rapid and Concurrent Acquisition of a Hybrid Task. Neural Control of Movement, May 2023.

Burckhardt, S., Madduri, M.M., Li S., Burden S.A., Orsborn, A.L. Examining the Role of Control Space Visualization Trainings in User Learning of Co-adaptive Myoelectric Interfaces. Neural Computation and Engineering Connection, Seattle, WA, May 2023.

---

*These authors contributed equally to this work

Chapter 1

INTRODUCTION

Interfacing machines with humans provides new opportunities to restore or enhance our interactions with the world (Serruya et al., 2002; Taylor et al., 2002; Carmena et al., 2003; Hochberg et al., 2006; Pandarinath and Bensmaia, 2022; Dadarlat et al., 2023). Speech prostheses can restore verbal communication (Moses et al., 2021), brain-to-text interfaces can translate imagined letters to computer text Willett et al. (2021), and neuroprosthetics can restore movements (Collinger et al., 2013). These biosignal-based user-machine interfaces translate biological signals like neural activity, muscle activity, or movements to control inputs for devices like computers or prostheses. By using rich, high-dimensional inputs, these interfaces have the potential to address neurological conditions, increase access to technologies, and provide high bandwidth control of our increasingly complex devices.

However, the full promise of user-machine interfaces has not yet been realized. Adoption of any technology entails trade-offs between costs and benefits that are multi-faceted and vary across individuals (e.g., (Ingraham et al., 2022). User-machine interfaces have the potential to improve the quality-of-life or abilities of users, but these possible benefits will be weighed against costs like invasiveness, price, time investment, and more. Some interfaces can use signals from wearable devices whereas others require surgical implants. Across different biosignal modalities, interfaces can be time-consuming to learn (Zhang et al., 2020) and offer variable performance across users (e.g., Acqualagna et al. 2016; Pandarinath et al. 2017) or tasks and contexts (Ganguly and Carmena, 2009). Easily learnable and generalizable interfaces that can be customized to user needs will drastically improve the landscape for widespread adoption. **The overarching motivation of my thesis is to develop user-machine interfaces that can be robust across a diverse set of users and can adapt across different tasks and environments.** This thesis focuses primarily on neural interfaces, but the concepts leverage and apply to other biosignal-based user-machine interfaces. Ensuring that neural interfaces work consistently for all users and across a variety of different functions is key to developing future, complex interfaces.

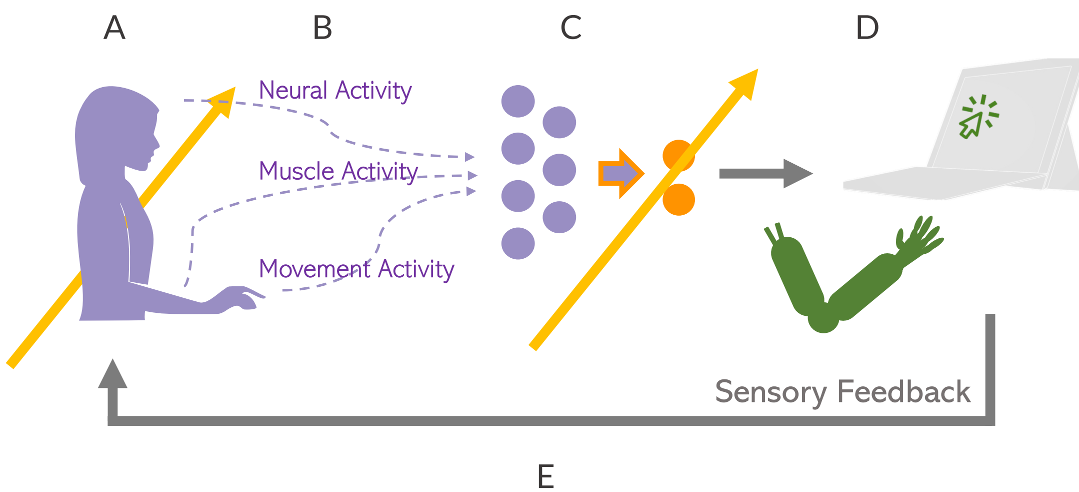

Figure 1.1: Diagram of User-Machine Interface. The **A. user** outputs **B. biosignals**, such as neural activity, muscle activity and/or movement activity, to the **C. decoder**, which, in real-time, converts biosignals into a **D. control command** for controlling a computer cursor or a robotic limb. The **E. sensory feedback** of the computer cursor or robotic limb causes the user to adapt, resulting in a closed-loop interface. This user adaptation then motivates decoder adaptation. Both the user and decoder adapting together is referred to as *co-adaptation*.

1.1 Closed-Loop, Co-adaptive Interfaces

Neural interfaces are inherently *closed-loop* (Fig. 1.1). Neural activity is transformed to control an external device via a *decoder*, and sensory feedback from the device is presented to the user. This closed-loop interaction naturally causes users to adapt to the interface.

The inevitability of user learning motivates use of decoders that can adapt in response. We refer to systems like these – with two agents adapting simultaneously in a closed-loop – as *co-adaptive*. Co-adaptation has promise to be a tool to enhance and guide user-machine interactions (Orsborn et al., 2014; Silversmith et al., 2020). Creating user-machine interfaces that facilitate user adaptation may be critical to restore complex behaviors and to enable interfaces with entirely novel technologies. Furthermore, as neural interfaces become more commonplace, the spectrum of users will expand. Decoders that adapt to users have the potential to accommodate user diversity and offer customization to individuals. These observations underscore the value of studying approaches that consider both user and decoder learning.

1.2 Importance of Decoder and User Adaptation

To fully understand the benefits of co-adaptation, it is first important to discuss the benefits of decoder or user adaptation, which each provide unique challenges and opportunities.

1.2.1 Decoder Adaptation

Adaptive decoding, firstly, is commonly used to train decoders for initial interface use and to retrain decoders to address day-to-day measurement variability, without explicitly considering user learning that may occur simultaneously. Closed-loop dynamics in neural interfaces influences decoder design and training. Motivated by machine learning approaches, many decoders are trained in open-loop: data is recorded as users make or imagine moving, and statistical relationships between biosignals and intended behaviors are identified with no feedback to the user (Chase et al., 2009; Zhang and Chase, 2015). Performance of a decoder on open-loop datasets does not necessarily predict performance when the decoder is used in closed-loop, where feedback allows the user to alter their behavior (Chase et al., 2009; Koyama et al., 2010).

An alternative to open-loop decoder training is to train the decoder in the same closed-loop context in which it will be used. Closed-loop decoder adaptation (CLDA) updates decoding parameters using real-time user activity (Orsborn et al., 2012, 2014). Updating the decoder's parameters in real-time can better capture how each user actually interfaces with the device during closed-loop operation. In this way, CLDA can provide rapid calibration and customization of an interface on initial use (Brandman et al., 2018; Dangi et al., 2013; Shanechi et al., 2016). Indeed, in BCIs, closed-loop decoder updates improve performance compared to open-loop calibration (Gilja et al., 2012; Jarosiewicz et al., 2013; Willsey et al., 2022).

Beyond calibrating interfaces to accommodate differences between users, CLDA can also be used to account for within-user variability over time. For example, once customized to a user, changes in the signals being measured can degrade performance (e.g., Degenhart et al. 2020; Perge et al. 2013). Adapting a decoder on a timescale comparable to the drift in biosignals can help prevent performance degradation. Day-to-day variability in the neurons recorded by an implant can be overcome with methods to adapt parameters across days (Degenhart et al., 2020). Continual decoder updates can maintain performance despite rapid drifts in neural measurements on the timescale of minutes (Jarosiewicz et al., 2015). Continual decoder updates can maintain performance despite rapid drifts in neural measurements on the timescale of minutes (Jarosiewicz et al., 2015). In BCIs, CLDA has been shown to provide consistent performance across months (Silversmith et al., 2020).

Decoder adaptation has undeniable advantages for accessibility, stability, and performance, but any decoder model is only as good as its inputs. Even with an adaptive decoder, an electroencephalography (EEG) BCI study of 168 naive users found that 22% could not achieve efficient control, possibly because users were unable to modulate the biosignal features used as decoder inputs (Acqualagna et al., 2016). Decoding algorithms inherently rely on the user and the user's adaptation. Encouraging user adaptation is key for robust, generalizable interfaces.

1.2.2 User Adaptation

Controlling a neural interface should be similar to riding a bike – once you learn, you can quickly ride with ease any time you grab a bike. The impressive capability underlying this colloquialism is natural motor skill learning, where training and practice yield robust "motor memories". Prior research suggests that similar skill learning mechanisms may help achieve robust, generalizable, and rich control in user-machine interfaces.

Feedback provided by a closed-loop device allows users to learn and adapt (Green and Kalaska, 2011; Jackson and Fetz, 2011). While neural interfaces involve artificial or altered sensorimotor pathways, this learning shows notable parallels to natural sensorimotor learning (Green and Kalaska, 2011; Jackson and Fetz, 2011; Orsborn and Carmena, 2013). For instance, extended practice with a stable BCI (neural measurements and decoder parameters fixed) results in performance improvements that are rapidly recalled each day (Ganguly and Carmena, 2009). This long-term learning in BCI also yields a stable neural encoding (Ganguly and Carmena, 2009). These learned neural encoders are heavily influenced by the decoder. In a kinematic interface study with fixed decoders, subjects formed an internal model that converged towards the decoder inverse (Pierella et al., 2019). Similarly, in BCI experiments with fixed decoders, neural activity patterns reorganize to align with the decoder (Athalye et al., 2017). These findings suggest users can form a motor memory specific to an interface that enables robust control.

A key property of motor memories is that, once consolidated, they remain stable as we learn additional tasks. Iteratively adding control dimensions has been used to achieve high-dimensional BCI control (Collinger et al., 2013; Silversmith et al., 2020). Interestingly, Silversmith and colleagues (2020) showed that, once a stable neural encoding of a two-dimensional BCI task had formed, the participant could learn to control an additional discrete behavior (a 'mouse click') without altering performance of the continuous behavior. This highlights the potential benefits of user learning for achieving robust performance across different contexts and the importance of user learning for stable neural interface control.

Beyond potential benefits of user learning, game theory formulations of the two-learner problem suggest that ignoring user learning could actually hinder performance. Indeed, adaptive decoding learning rates must be tuned appropriately relative to the rate of user learning to ensure the system will converge at all (Müller et al., 2017; Madduri et al., 2021; Chasnov et al., 2020), and the solution a co-adaptive system converges to could be one of any number of game-theoretic stationary points (e.g., Nash or Stackelberg equilibria) (Fudenberg and Levine, 1998; Chasnov et al., 2023).

While user learning offers potential benefits and should be considered in interface design, relying on user learning alone may be limiting. Learning to control a novel interface can be slow, taking anywhere from hours to days of practice depending on the biosignal modality (Jackson and Fetz, 2011). A static interface that requires significant practice for users to master is likely to produce frustrating devices that are quickly abandoned. For instance, even with the recent advancements in myoelectric prostheses, abandonment rates of upper-limb prostheses remain at over 40% due in part to poor ease of use (Salminger et al., 2022). More broadly, fixed interfaces that make assumptions about users may reduce the subset of people who can use these devices. An example from airplane design illustrates the pitfalls of fixed interfaces (Hendren, 2020): in the 1940s, airplane cockpit designs based on averages of pilot measurements fit no individual pilot well, leading to the development of cockpits that could adjust to individual specifications.

1.3 Combining Decoder and User Adaptation Towards Co-adaptation

Co-adaptive interfaces can leverage the interplay between user and decoder learning to generate more powerful interfaces. Experimental studies hint at the promise of co-adaptive strategies. BCI studies in animals and humans demonstrate that stable neural encoders still form when decoders adapt over time (Orsborn et al., 2014; Silversmith et al., 2020). Stable encoder formation in co-adaptive BCIs also correlates with high performance that is rapidly recalled each day (Orsborn et al., 2014). Towards generalizable interfaces, user learning in a co-adaptive BCI resulted in control that resisted interference from context changes (Orsborn et al., 2014). Similarly, co-adaptation enabled a human user to add additional control dimensions to a BCI (Silversmith et al., 2020). Gradual decoder changes have also been shown to help users control BCIs that are initially challenging to learn (Oby et al., 2019), which may be critical for increasing interface accessibility.

Experimental observations highlight the interdependence between decoders and encoders in co-adaptive systems. The degree of decoder adaptation performed in a BCI influences the amount of changes in a user's encoder (Orsborn et al., 2014). The number of neurons needed to successfully control a BCI has been shown to decrease in systems where decoders adapt alongside users (Taylor et al., 2002). In kinematic interfaces, altering the alignment between a decoder and the user's movement encoding space impacts the redundancy and efficiency of user movements (De Santis and Mussa-Ivaldi, 2020). Thus, adaptive decoding may influence the form of learned encoders. This outcome may be beneficial for yielding robust interface performance, and opens opportunities to use co-adaptation for rehabilitation (e.g., De Santis 2021).

However, co-adaptation introduces a two-learner problem: both the user and decoder are adapting and interacting in real-time, similar to two agents playing a game (Neumann and Morgenstern, 1944). Methods to analyze and synthesize these two-learner interactions are still active areas of research, both within the context of user-machine interfaces (Dangi et al., 2013; Hsieh and Shanechi, 2018) and dynamic game theory Mazumdar et al. (2020). We need principled and experimentally-validated frameworks to understand existing co-adaptive interfaces and create future interfaces. **The overarching goal of my thesis is to develop theoretical tools and algorithms to design co-adaptive neural interfaces that can be high-performing and robust across a diverse set of users.**

1.4 Thesis Organization

Overview

In *Chapter 2*, we model co-adaptive user-decoder interactions using game theory to guide adaptive decoder designs. *Chapter 3* tests these game-theoretic adaptive decoder schemes in myoelectric interfaces and evaluates their impact on performance. In *Chapter 4*, we use control theory to estimate the user's *encoders* and evaluate closed-loop user-decoder interactions. *Chapter 5* explores user-decoder learning dynamics, demonstrating how the game-theoretic model can predict user-decoder interactions and how decoder adaptation influences user learning. Lastly, the *Discussion* describes the potential of this research towards designing the next-generation of personalized, co-adaptive neural interfaces.

Chapter 2

Chapter 2 uses game theory techniques to model the user-decoder interactions in a co-adaptive interface. The user and decoder are considered as two agents adapting to their individual cost functions in a *dynamic game*. By analyzing the conditions for stable user-decoder dynamics, we identify key three parameters of decoder design: the rate of decoder adaptation, the emphasis on minimizing decoder "effort" versus task error, and the decoder initialization. Chapter 2 contributes a framework and analysis for co-adaptive dynamics that provides a new, multi-objective decoder adaptation scheme and ways to influence co-adaptive interface performance and interactions.

Chapter 3

Chapter 3 tests and verifies the proposed adaptive decoder scheme in continuous myoelectric interfaces. The decoder adaptation parameters are shown to influence the myoelectric interface performance in the ways that the game-theoretic model predicted, experimentally-validating this framework. This chapter highlights the value of a framework that can systematically analyze and predict the influence of decoder design choices on system performance.

Chapter 4

Chapter 4 develops a control-theoretic method for estimating users *encoders* in closed-loop contexts. Using these methods, this chapter shows that users' encoders are changing alongside the adaptive decoders in our myoelectric interface, furthering supporting evidence of co-adaptation. This chapter then investigates the interactions between the encoder and decoder using these estimations methods, providing a technique for quantifying user-decoder co-adaptation.

Chapter 5

Chapter 5 reveals how user-decoder learning dynamics are influenced by different decoder adaptation designs. Using the encoder estimation techniques from the previous chapter, this chapter shows the impact of decoder learning rate on user learning and the effect of decoder effort on user effort. Importantly, the decoder adaptation parameters influence user behavior as the game-theoretic model predicted, verifying that the framework predicted the outcome of co-adaptive interface interactions and revealed how interface properties can shape user behavior.

Chapter 2

MODELING CLOSED-LOOP USER-DECODER DYNAMICS USING GAME THEORY

2.1 Introduction

Uncovering the neural principles of learning in the brain is key to developing high-performing neural interfaces (Shenoy and Carmena, 2014; Moritz, 2018). Closed-loop neural interfaces define an input-output relationship for users and can provide insight into fundamental neural mechanisms underlying sensorimotor learning (Carmena, 2013). Motor-based neural interfaces use a *decoder* to convert neural data to a control signal, such as moving a computer cursor. Closed-loop neural interfaces form a closed control loop via visual feedback, creating novel sensorimotor systems that can be manipulated to study neural functions (Orsborn and Pesaran, 2017).

Learning in neural interfaces can lead to the formation of a *stable cortical map*, similar to natural motor learning (Ganguly and Carmena, 2009). Stable cortical maps are accompanied by stable neural interface task performance. This convergence to stable performance is a desirable trait of neural interfaces and the motivating design characteristic behind this model. Stable performance is evidenced in neural interfaces through consistent cursor trajectory to the target with low variability and other signatures of skillful motor control (Ganguly and Carmena, 2009). While these experiments showing stable cortical maps are specific to invasive neural interfaces with non-human primates, stable performance is still a necessary condition for neural interface usability and adoption for invasive and non-invasive user-machine interfaces.

Recent research has shown that *closed-loop decoder adaptation* (CLDA)—algorithms to modify decoder parameters during closed-loop neural interface operation—are potentially able to guide user learning to facilitate the emergence of stable cortical maps (Orsborn et al., 2012, 2014). CLDA algorithms have been designed to improve neural interface performance and rapidly calibrate interfaces to users. Designing CLDA algorithms to guide user learning is an open challenge. As an inherently real-time process, CLDA requires online neural interface validation. However, testing a wide range of CLDA algorithms through experimentation is impractical.

To constrain the space of testable CLDA algorithms that aim to shape user learning and reduce experimental load, this chapter proposes and analyzes a mathematical model for user-decoder dynamics in closed-loop co-adaptive neural interfaces. The purpose of this work is to model the interactive dynamics between user and decoder in a co-adaptive neural interface without assuming specific neural mechanisms in the brain. We adopt the perspective that the brain and decoder are two agents playing a *dynamic game* Hespanha, J. (2017), and apply techniques from game theory to study the asymptotic dynamics of co-adaptation. We use a *potential game* formulation to derive stationary points (Nash equilibria), which points are our model's surrogates for *stable cortical maps* observed in experiments. Assuming the brain and decoder adapt using (stochastic) gradient descent, we assess convergence to stationary points. Finally, we corroborate our mathematical analyses using numerical simulations.

2.1.1 Related Work

Principled Design of Co-adaptive Interfaces

Current empirical observations suggest decoder adaptation can be used to shape user adaptation, opening new ways to improve user interface performance. However, tuning current co-adaptive interfaces is challenging and largely done based on heuristics. Computational frameworks that predict decoder performance by modeling online user behavior (e.g., Chase et al. 2009; Willett et al. 2019) suggest models may be useful for optimizing decoder performance. However, while these frameworks account for the closed-loop nature of interfaces, they are not designed to capture user adaptation. Developing a toolkit for co-adaptation will enable rigorous design of user interfaces that fully leverages the benefits of co-adaptive systems.

A key consideration for co-adaptive systems is the relative timing of adaptation for the two learners. The user and decoder do not necessarily have to adapt in lock-step with one another. The user or decoder could lead, guiding the other to follow. Existing co-adaptive user-machine interface frameworks can be viewed through this leader-follower lens. Merel et al. (2015) propose a method to estimate the optimal decoder that “anticipates” user adaptation. They find that users, as followers, can adapt to match the fixed estimated optimal decoder mapping. So long as the users obey the statistical modeling assumptions of their model, this could produce desirable co-adaptation. On the other end of the spectrum, De Santis (2021) proposes a co-adaptive framework where users lead, and decoders adapt to them. Their method modifies decoders to optimize the interaction efficiency between the user and decoder. This approach assumes that users adapt to maximize rewards while the decoder adapts to approximate the covariance of the user’s activity. Many co-adaptive schemes in BCI are structured such that users lead while decoders follow (Orsborn et al., 2014; Silversmith et al., 2020; Taylor et al., 2002).

We believe the potential power of co-adaptation arises from dynamic interactions between the two learners. Frameworks that can flexibly capture the full spectrum of potential users and decoder interaction timescales are needed. For example, early interactions like interface calibration might benefit from decoders following users to maximize initial performance and encourage engagement. But later interactions might require decoders to lead users towards new strategies that improve overall performance or to encourage motor recovery in rehabilitation applications. A linear two-learner model, where the user and decoder both adapt concurrently according to a joint cost function, allowed Müller and colleagues (2017) to explore the influence of adaptation timescales on co-adaptive systems.

The studies described above represent existing theoretical approaches to co-adaptation in biosignal interfaces. To create an even more versatile framework to explore user-decoder co-adaptation, we have proposed directly modeling the system as a game. Game theory provides a suite of tools to analyze and design learning dynamics and provides ways to encourage co-adaptive systems to converge to stable user-decoder equilibria. Game-theoretic techniques have been used to develop a framework for defining dyadic interactions (Jarrassé et al., 2012) and to design a human-robot controller that estimates each agent's cost and adapts to complete a shared goal (Li et al., 2019).

Prior Co-adaptive Interface Mathematical Models

Existing neural models draw upon control theory to conceptualize the user and decoder as systems adapting according to specified learning processes and subject to cost functions (Héliot et al., 2010; Merel et al., 2015; Müller et al., 2017). The Merel and Muller models identify co-adaptation strategies using optimization principles to propose adaptive decoder paradigms. The user and the decoder in these models are treated as two distinct stochastic optimal control systems (Merel et al., 2015) or as two systems optimizing a joint cost function (Müller et al., 2017). The learning dynamics of co-adaptation in these models either propose a stochastic coordinate descent, where the decoder learns to anticipate the user's adaptation (Merel et al., 2015) or a stochastic gradient descent with an online least-squares estimator, where the brain and the decoder estimate each other's parameters to form internal approximations of the other learner (Müller et al., 2017). The proposed optimal brain-decoder solutions for each model are identified through either mathematical analysis (Merel et al., 2015) or computational simulations (Müller et al., 2017).

User Learning and Cost Functions

The connections and adaptation of the brain has inspired artificial neural networks, but applying computational deep learning models to understand how the brain operates is an active field of research (e.g., Pospisil et al. 2018). Research has proposed that the brain might be optimizing some cost function. Physiologically means that the brain can change its own properties to traverse a cost function. One theory is that the brain locally samples and measures its properties to minimize error at a global level, and stochastic exploration at a local level can be a way that the brain improves performance or decreases error signals (Marblestone et al., 2016). Adding noise or a perturbation to synaptic inputs have been proposed as neural learning mechanisms as the brain computes the cost of this noisy input (Lillicrap et al., 2020). The idea of the brain/body optimizing some cost function has been evidenced in the minimization of metabolically costly movements (Taylor and Faisal, 2011) and has been computationally suggested for the coordination of complex motor behaviors (Todorov and Jordan, 2002). Jarosiewicz et al. 2008 also find evidence that the brain's reorganization during a BCI task is decreasing some global error by adjusting local synaptic connections.

Experimental Connection to Model

This model proposed in this chapter is designed to be simplistic and broadly applicable towards closed-loop neural interfaces (e.g., Orsborn et al. 2012, 2014; Gilja et al. 2012; Hahne et al. 2015). In these sorts of interfaces, the user is trying to control a cursor on the screen towards a specified target. The interface task, therefore, is to move the cursor towards the target. We assume the task to be 2-dimensional (2D); the user has to control both the x- and y-direction to move their cursor towards the target. The user's biosignals (e.g., neural activity or myoelectric activity) is input to an adaptive decoder. The output of the adaptive decoder is cursor position or cursor velocity. For the model presented in this chapter, we are making the following assumptions: the user is trying to control a 2D cursor on the screen to reach a 2D target. The user's neural activity is output to the decoder, which outputs cursor position.

2.1.2 Contributions of This Chapter

The purpose of this chapter is to model the interactive dynamics between the user and decoder in a co-adaptive neural interface without assuming specific neural mechanisms of the user. Additionally, the goal of our model is to emphasize the convergence to stationary points and the conditions to keep the user-machine system at stationarity.

The chapter demonstrates that:

1. the two-learner problem for motor neural interface can be formulated using a game-theoretic framework;
2. we derived stationary points (Nash equilibria) of the game motivated by experimental observations of stable cortical map formation;
3. we determined analytical predictions for convergence of a model for co-adaptation to these stationary points and use these predictions to inform adaptive decoder design.

2.2 Methods

2.2.1 Game Formulation

Fig. 2.1 displays each component and corresponding symbolic representation. Signals are denoted with lower-case symbols (e.g., x) and parameterizations of the transformations (i.e., matrices) are capitalized (e.g., X). Since we are considering the usage of a closed-loop interface across time, we consider the interface changing each time a new target position is presented to the user. In a continuous context, this would be the rate of a new target position presented on the screen. Time is represented by t and indexed by the subscript $t = 1, 2, \ldots$. Task dimensionality is indexed by the subscript $d = 1, 2, \ldots$. There are N channels of biosignal data and each channel n is indexed by the superscript $n \in \{1, \ldots, N\}$.

We intend this biosignal data to either represent neural activity, in which case N could be the number of neurons and u could be neuronal firing rates, or to represent muscle activity, in which case N is the number of EMG channels and u is EMG/myoelectric activity. We will suppress the sub- and/or super-scripts when they are clear from context. For step-by-step details of these methods, see Sec. 7.1).

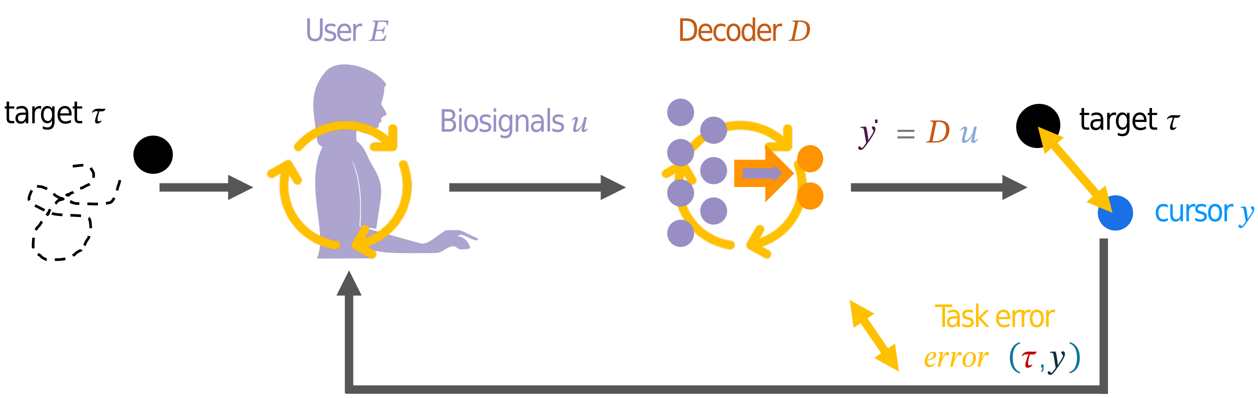

Figure 2.1: Closed-loop Co-adaptive Interface Model. Diagram showing the forward model of the user and the decoder as well as the visual feedback error of the cursor (task error feedback) that drives the co-adaptation of the brain and the decoder (Fetz, 2007). Based on prior neural interface models (Merel et al., 2015; Müller et al., 2017), the elements of this model are: the user (Encoder E) who generates real-time biosignals u to control a computer cursor, the decoder D that maps the biosignals to a control output: the cursor velocity v_y, and the error between the target τ and cursor y, that motivates the adaptation of the user and decoder.

Target Position

The task presented for this user-interface model is a center-out cursor control task (e.g., Orsborn et al. 2014, 2012). The target position is at τ.

User and Neural Activity

The user (Héliot et al., 2010; Müller et al., 2017) is represented simplistically by a linear transformation of the target τ to neural activity u.

$$u_t = f(E_t, \tau_t) = E_t \tau_t \tag{2.1}$$

where $u_t \in \mathbb{R}^{N\times 1}, E_t \in \mathbb{R}^{N\times d}, \tau_t \in \mathbb{R}^{d\times 1}$. At each new time sample t, the EMG signals u are updated and are inputs to the decoder (Eq. 2.2).

Decoder and Cursor Position

The decoder in this model is represented as a Wiener filter (i.e., a static linear function), a decoder model commonly employed in prior work (Héliot et al., 2010; Müller et al., 2017). [1] The decoder transforms neural activity, u, to a d-dimensional cursor position, y, by multiplying the neural activity by the decoder's parameters, D (Eq. 2.2).

$$y_t = f(D_t, u_t) = D_t u_t \tag{2.2}$$

where $u_t \in \mathbb{R}^{N\times 1}, D_t \in \mathbb{R}^{d\times N}, y_t \in \mathbb{R}^{d\times 1}$.

2.2.2 User and Decoder Cost Functions

For each time sample, the task error (Eq. 2.3) is calculated as the quadratic loss of the cursor and target positions (Héliot et al., 2010; Müller et al., 2017):

$$\varepsilon_t(\tau_t, y_t) = \|\tau_t - y_t\|_2^2. \tag{2.3}$$

The user and the decoder parameters, E and E, update through the trial using a gradient-based scheme detailed in Sec. 2.2.3. From a game-theoretic lens, this model can treat the user and the decoder as two agents in a 2-player game. Each agent, the user or the decoder,

[1] We note that the steady-state transformation between constant inputs and outputs produced by a Kalman Filter is a Wiener filter (Hutchinson, 1966).

is trying to minimize its own cost function through co-adaptation. The cost function of the user and decoder are:

$$\begin{aligned} c_E(E,D) &= \varepsilon(\tau,y) + \lambda_E\|E\|_2^2 \\ c_D(E,D) &= \varepsilon(\tau,y) + \lambda_D\|D\|_2^2. \end{aligned} \tag{2.4}$$

The user and the decoder cost functions both include regularization terms: $\lambda_E\|E\|_2^2$ for the user and $\lambda_D\|D\|_2^2$ for the decoder. The parameters λ_E and λ_D are scaling factors on the regularization terms and, as we will show later in this paper, play crucial roles in ensuring the convergence of the cost functions. These regularization terms serve to penalize the effort of the user and the decoder. User effort is a proxy for the magnitude or number of channels of muscle or neural activity exerted (Todorov and Jordan, 2002). Decoder effort is a way to constrain the decoder adaptation parameters. Fig. 2.2A shows a visualization of the user and decoder cost functions.

2.2.3 Solving for Stationary Points

Defining a Potential Function

With the mathematical model described above, we can reformulate the user-decoder interactions into a *potential game*. In a potential game, the action of all agents can be expressed with a single function called the potential function. If there exists a continuously differentiable function $\phi : \mathbb{R}^{n\times d} \mapsto \mathbb{R}$ such that $\frac{\partial\phi}{\partial E} = \frac{\partial C_E}{\partial E}$ and $\frac{\partial\phi}{\partial D} = \frac{\partial C_D}{\partial D}$, this function ϕ can be considered to be a potential function (Hespanha, J., 2017, Ch. 12). This highlights an important assumption: the cost functions of the user and the decoder are differentiable. The user and the decoder cost functions satisfy the conditions of a potential game and can be thus characterized with the following potential function:

$$\phi(E,D) = \varepsilon(\tau,y) + \lambda_E\|E\|_2^2 + \lambda_D\|D\|_2^2. \tag{2.5}$$

This potential function matches the two-learner system cost function proposed by a prior model (Müller et al., 2017). The potential function ϕ tracks the changes of both players, the

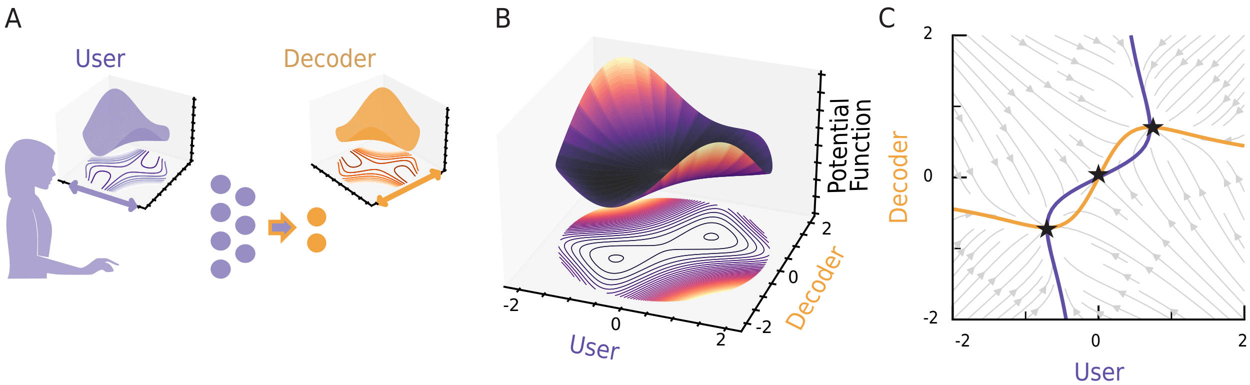

Figure 2.2: Game-theoretic Model of Co-adaptive User-Machine Systems.

A. Schematic of co-adaptive interfaces, where the user and decoder are modeled as adapting to minimize their own individual cost functions (inset plots for each agent).

B. Visualization of the potential function that describes dynamics within the user-decoder game model (simplified 1D user and decoder, see Methods). Two axes represent the scalar values of the user and decoder, and the vertical axis represents the value of the potential function. A 2D projection of the potential function is also shown.

C. Gradient field of the user and decoder cost functions (with equal penalty terms, $\lambda_D = \lambda_E = 0.5$). Purple (user) and orange (decoder) curves are nullclines (where the agent's gradient equals 0), which intersect at stationary points (black stars).

user and the decoder in this case, and treats co-adaptation as a single-objective optimization problem. Analyzing ϕ can lead to insight about the co-adaptive dynamics of the user and decoder. Therefore, the subsequent mathematical methods focus on understanding the dynamics of the user and decoder as characterized by ϕ. Importantly, the advantage of expressing the user-decoder game as a potential game is that the potential game provides structure and brings a suite of established techniques from game theory that we can use to analyze the user-decoder co-adaptive game. Fig. 2.2B shows a visualization of the potential function.

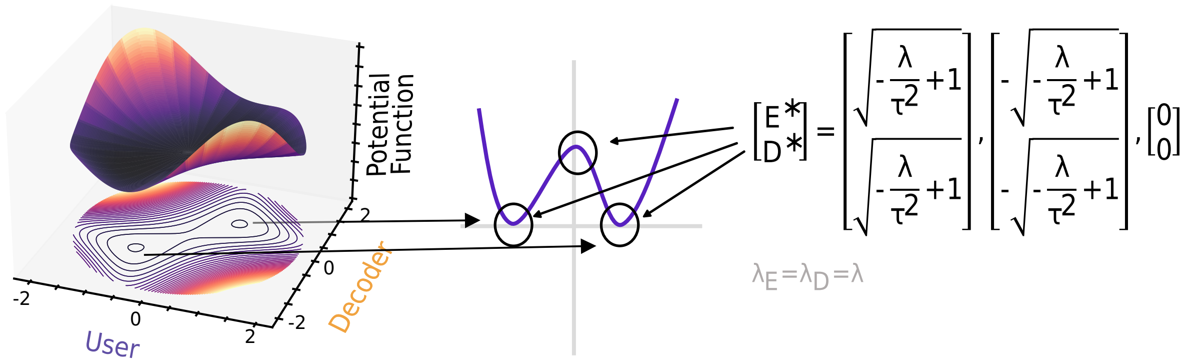

Figure 2.3: Potential Function Landscape and Stationary Points. (a) This is an example of the curvature of the potential function (d = 1, N = 1), ϕ (Eq. 2.5), with λ_E = λ_D = $\frac{1}{2}$. The two minimas and saddle point can be seen. (b) In the simplest case of $d = 1$ and $n = 1$, this cartoon illustrates the stationary points of the potential function when $\lambda_E = \lambda_D = \lambda$.

Stationary Points of Potential Game

The Nash equilibria of potential games, referred to as *stationary points* in this paper, are the directionally-local minima of the potential function (Hespanha, J., 2017, Prop. 12.2). The stationary points E^*, E^* are determined by solving $\frac{\partial \phi}{\partial E}\Big|_{E=E^*} = 0$ and $\frac{\partial \phi}{\partial D}\Big|_{D=D^*} = 0$. These are points where the user and the decoder reach steady-state, or are no longer adapting, in the user-machine interface model. Fig. 2.2C depicts the stationary points of the user and decoder game when the penalty parameters are equal. Fig. 2.3 depicts how we solved for stationary points by linearizing around the fixed points.

2.2.4 *Gradient-Based Co-Adaptation*

The user and the decoder parameters, E and D, are updated according to the gradient-based adaptation scheme:

$$\begin{aligned} E_{t+1} &= E_t - \alpha_E \Delta_{E_t} \\ D_{t+1} &= D_t - \alpha_D \Delta_{D_t} \end{aligned} \tag{2.6}$$

where $\alpha_{E_t}, \alpha_{D_t} > 0$ denote learning rates and $\Delta_{E_t}, \Delta_{D_t}$ denote (approximations of) gradients of the agents' cost functions with respect to their parameters. Note that Eq. 2.6 models adapt at the frequency of target position updates; no longer- or shorter-term learning processes are represented. The learning rates are assumed constant throughout the updates, signifying that the user and the decoder both adapt at a constant rate throughout the timeline that the model illustrates. We analyze convergence of Eq. 2.6 to stationary points in two different regimes: 1) *deterministic* gradients with constant learning rates; and 2) *stochastic* gradients with decreasing learning rates.

Deterministic gradient descent

If learning rates are constant and agents adapt using their exact gradients,

$$\Delta_{E_t} = \frac{\partial \phi}{\partial E}\Big|_{E=E_t}, \Delta_{D_t} = \frac{\partial \phi}{\partial D}\Big|_{D=D_t} \tag{2.7}$$

then Eq. 2.6 defines a deterministic dynamical system. To assess convergence, we linearize this system about the stationary points from the preceding section and evaluate eigenvalues of the Jacobian matrix (Fig. 2.3). Note, for a helpful reference on this process, see Brunton and Kutz 2017. The system with constant learning rates can be considered discrete-time, since the updates to time occur at discrete time intervals. Since the system is discrete-time, the magnitude of these eigenvalues determines how quickly the co-adaptation dynamics converge to stationarity.

Stochastic gradient descent

As noted in Müller et al. 2017, evaluating the exact gradients in Eq. 2.7 requires each agent to know the other's parameters. As an alternative, we consider the dynamics obtained when agents use a *stochastic approximation* of their gradient. One such approximation is Nesterov and Spokoiny (2017, Eq. 31).

$$\Delta_E = \frac{1}{\ell \nu_E} \sum_{i=1}^{\ell} (c_E(E + \nu_E \delta_E, D) - c_D(E, D)) \cdot M_E^{-1} \delta_E$$
$$\Delta_D = \frac{1}{\ell \nu_D} \sum_{i=1}^{\ell} (c_D(E, D + \nu_D \delta_D) - c_D(E, D)) \cdot M_D^{-1} \delta_D \tag{2.8}$$

where $\delta_E \sim \mathcal{N}(0, M_E), \delta_D \sim \mathcal{N}(0, M_D)$ are zero-mean normal random vectors with covariances M_E, M_D, respectively. δ represents the direction of the random search. M^{-1} represents a correlation operator to constrain δ to the unit sphere so that the stochastic gradient descent converges. ν_E and ν_D are the difference parameters that scale the contribution of the δ_E and δ_D, respectively. ν scales the contribution of that perturbation to the overall gradient update. We implement this approximation for our numerical simulations. Note that Eq. 2.8 defines unbiased estimates for the exact gradients in Eq. 2.7 that each agent can evaluate using its own cost and parameters without requiring it to know the cost or parameters of the other.

2.3 Results

The methods of the preceding section apply to task and biosignal spaces of arbitrary dimension. To illustrate an interpretation of this model, we analyzed the simplest case where the task is 1-dimensional ($d = 1$) and there is a single biosignal channel ($n = 1$) being input to the decoder. We also apply some assumptions in our results for simplicity of analysis.

2.3.1 Stationary Points of the Potential Game

When the same value $\lambda_E = \lambda_D = \lambda \in (0, \tau^2)$ is chosen for the regularization parameters in the user and the decoder cost functions (Eq. 2.4), the potential function (Eq. 2.5) has two stationary points (the Nash equilibria of the potential game),

$$E^* = D^* = +\sqrt{1 - \frac{\lambda}{\tau^2}} \text{ and } E^* = D^* = -\sqrt{1 - \frac{\lambda}{\tau^2}}, \tag{2.9}$$

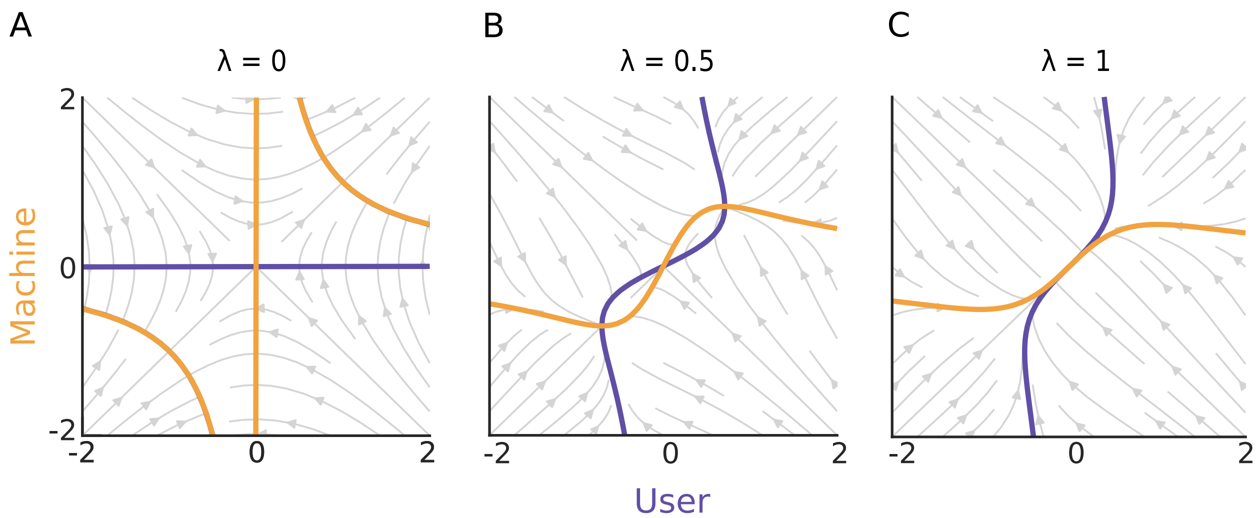

Figure 2.4: Single Dimensional Simulations. Gradient field of potential function (Eq. 2.5) with penalties $\lambda_E = \lambda_D = \lambda$ and varying amounts of λ.This diagram illustrates the presence of stationary points as the λ term changes: **A.** $\lambda = 0$, **B.** $\lambda = 0.5$, and **C.** $\lambda = 1$. Purple (User) and orange (decoder) curves are *nullclines* (where $\frac{\partial \phi}{\partial E} = 0$ or $\frac{\partial \phi}{\partial D} = 0$, respectively) that intersect at stationary points. The cost functions are from Eq. 2.4 and normalized task ($\tau^2 = 1$) as in Sec. 2.2.3.

obtained by applying the procedure described in Sec. 2.2.3. (When $\lambda > \tau^2$, the only stationary point is at the origin $E^* = D^* = 0$; when $\lambda < 0$ there are no stationary points.) Fig. 2.4 illustrates these stationary points as λ varies. Substituting the stationary point (Eq. 2.9) into the cost functions (Eq. 2.4) yields stationary costs

$$c_E(E^*, D^*) = c_D(E^*, D^*) = \lambda \tag{2.10}$$

and stationary task error

$$\varepsilon(E^*, D^*) = \frac{\lambda^2}{\tau^2}. \tag{2.11}$$

Note that, as the penalty parameter λ goes to zero, the stationary task error goes to zero at a quadratic rate.

2.3.2 Convergence of Deterministic Gradient Descent

When the same value $\lambda_E = \lambda_D = \frac{1}{2}$ is chosen for the L2-regularization parameters in the user and the decoder cost functions, Eq. 2.4, and the task space is normalized so that $\tau^2 = 1$, the eigenvalues of the linearization of Eq. 2.6 at either of the stationary points Eq. 2.9 are

$$\{1 - 2\alpha_E, 1 - 2\alpha_D\}\,. \tag{2.12}$$

The co-adaptation dynamics converge if the eigenvalues lie inside the unit circle, which occurs precisely when

$$0 < \alpha_E < 1,\ 0 < \alpha_D < 1. \tag{2.13}$$

Since the eigenvalue with the maximum magnitude determines the asymptotic convergence rate, the learning rates $\alpha_E = \alpha_D = \frac{1}{2}$ yield the fastest convergence rate.

2.3.3 Convergence of Stochastic Gradient Descent

To corroborate analytical results reported in the preceding sections, we ran batches of numerical simulations of the co-adaptation dynamics in Eq. 2.6 using the stochastic gradient estimate in Eq. 2.8. These simulations converged to the stationary points in Eq. 2.9 as predicted (Fig. 2.5).

2.3.4 Extension to Multi-channel Case

The dynamical systems methods still hold for the multichannel case where $D \in \mathbb{R}^{N\times 1}$, $E \in \mathbb{R}^{N\times 1}$ and $N > 1$. We keep the target to be 1-dimensional: $\tau \in \mathbb{R}^1$. Linearizing about the fixed point and setting $\lambda_E = \lambda_D = \lambda$, we find:

$$D^{*\top}E^* = 1 - \frac{\lambda}{\tau^2}. \tag{2.14}$$

Thus, since, $1 - \frac{\lambda}{\tau^2}$ must be a scalar value, D and E are not linearly independent at the stationary points. The implication is that at stable task performance in a 1D co-adaptive

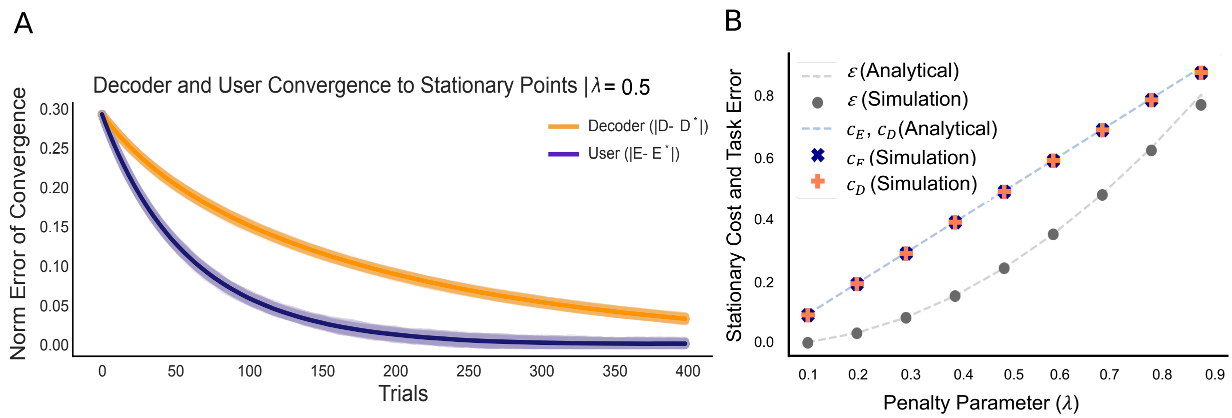

Figure 2.5: Simulations of single-dimensional case.

A. Numerical simulations of co-adaptation (Eq. 2.6) with stochastic gradients (Eq. 2.8) demonstrate exponential rate of convergence to stationarity (Decoder (orange): $\|D_t - D^*\|$, User (purple): $\|E_t - E^*\|$ converge to zero at exponential rate); the difference in convergence rate is caused by a difference in learning rates ($\alpha_E = 5 \times 10^{-3}$, $\alpha_D = 2.5 \times 10^{-3}$). Each trace represents a run ($n = 100$) and the solid trace represents the mean across runs.

B. Stationary cost (Eq. 2.10) and task error (Eq. 2.11) determined analytically (traces) and computed from numerical simulations (markers) for different values of penalty parameter $\lambda_E = \lambda_D = \lambda$.

interface system, the user and the decoder parameters would be co-linear. As the user and decoder approach convergence, the task error approaches

$$\varepsilon(\tau, y) = \|\tau - y\| = \|\tau - (1 - \frac{\lambda}{\tau^2})\tau\|. \tag{2.15}$$

Numerically simulating the multidimensional case with $N = 10$ channels, $\tau = 1$, and $\lambda = 0.25$, we find that the user and decoder co-adapt to approach the cursor position ($y = 0.75\tau$) and task error ($\varepsilon = \|0.25\tau\|$) that was computed based on these analytical methods (Fig. 2.6).

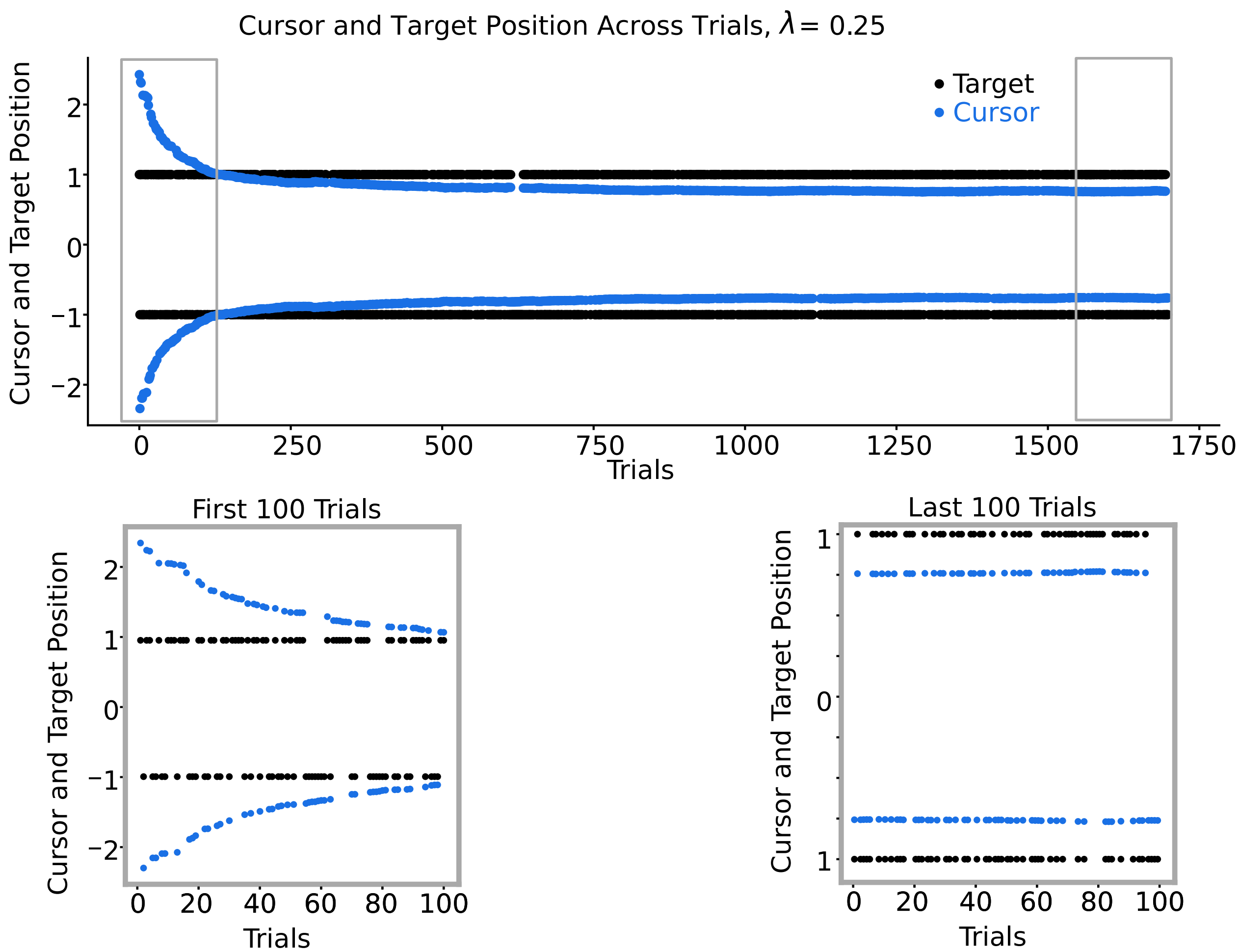

Figure 2.6: Multidimensional Simulation. Numerical simulations of co-adaptation (Eq. 2.6) with stochastic gradients (Eq. 2.8) demonstrate approach towards predicted target (black) and cursor (blue) position. The target position was alternating across $x = -1$ and $x = 1$, which represented the target at two different positions. The following parameters were used for this simulation: $\lambda_D = \lambda_E = 0.25$, $N = 10$.

2.4 Discussion

2.4.1 Key Assumptions and Considerations of Model

It is important to keep in mind key assumptions and considerations of this user-machine model formulation here, in no significant order.

- Learning in this model occurs in discrete steps from trial-to-trial or sample-to-sample, and no longer-term or shorter-term learning processes are integrated into this model.
- The learning rates associated with stochastic gradient descent (α_E, α_E) are constant throughout the trials, signifying that the brain and the decoder both adapt at a constant rate throughout the experiment.
- This model assumes that the user has knowledge of its own parameters and the ability to evaluate its own cost function. The biological plausibility of the user's encoder being able to compute the gradient of its own cost function is debatable.
- The user's biosignals are assumed to adapt independently in this model. The change in the activity of a single channel (i.e., change in firing rate of a single neuron) does not influence the activity of other channels during the same trial.
- The cost functions of the user and the decoder are differentiable. This assumption is important for mathematical and computational analyses.

2.4.2 Interpretation of Results

Stationary points

The formulation of user-decoder co-adaptation in a game-theoretic framework naturally leads to the consideration of stationary points (Nash equilibria). Our analysis focused on how the penalty parameters λ_E, λ_D (Eq. 2.4) affect these stationary points. Focusing on the case where $\lambda_E = \lambda_D = \lambda \in (0, \tau^2)$, we found that $\lambda \in (0, \tau^2)$ is necessary for the existence of non-trivial stationary points: no points are stationary when $\lambda < 0$ and only the origin is stationary when $\lambda > \tau^2$. Since these stationary points are our model's surrogates for the *stable cortical maps* that experiments suggest are highly desirable for skillful control, this finding indicates that penalty parameters can play an important role in ensuring such a stable outcome can be achieved.

Convergence of co-adaptation

Assuming both the user and decoder adapt their parameters using a gradient-based scheme, we analyzed convergence of co-adaptation to stationarity. Our analysis showed that the learning rates α_E, α_D play a critical role in convergence. Focusing on the case where $\lambda_E = \lambda_D = \frac{1}{2}$ and $\tau^2 = 1$, we found that convergence is guaranteed so long as

$$\rho = \max\{|1 - 2\alpha_E|, |1 - 2\alpha_D|\} < 1, \tag{2.16}$$

in which case ρ determines the rate of convergence since $E_t \to E^*$ and $D_t \to D^*$ exponentially in time/trial t with rate $\rho < 1$:

$$|E_t - E^*| \le C_E\rho^t,\ |D_t - D^*| \le C_D\rho^t, \tag{2.17}$$

for some fixed constants $C_E, C_D > 0$. Our findings demonstrate that the decoder's learning rate must be chosen carefully, as learning too fast may lead to an unstable outcome, whereas learning too slow may limit the rate of convergence to stationarity. These analytical results are corroborated by previous simulation and experimental findings (Müller et al., 2017; Orsborn et al., 2012).

2.4.3 Relevance of Game Theory to Co-adaptive Neural Interfaces

The field of game theory considers interactions between multiple decision-makers that act independently in their own self-interest (Başar and Olsder, 1999; Fudenberg and Levine, 1998; Neumann and Morgenstern, 1944). One common paradigm models these actors (also referred to as agents or players) as making decisions rationally by optimizing individual utility functions (also referred to as costs or rewards) whose values are determined by

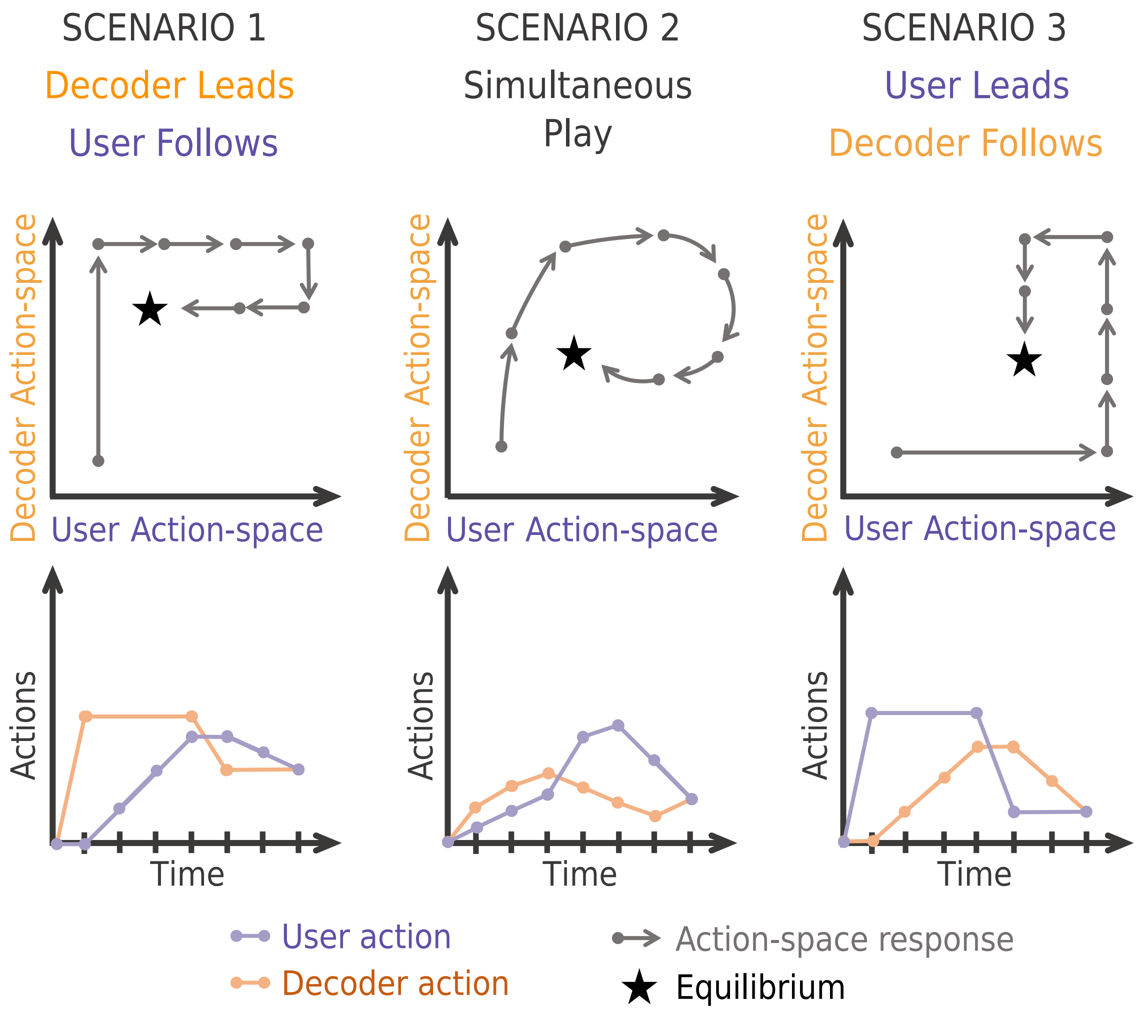

Figure 2.7: Conceptual illustration of leader-follower relationships in co-adaptive user interfaces. The top graphs show the combined user-decoder action space, which represents the action taken by either the decoder, the user, or both simultaneously. The bottom graphs represent the user and the decoder actions across time.

Scenario 1. (Left) shows the decoder leading. The decoder starts with an action, and the user follows in response with multiple actions. This repeats until convergence is met.

Scenario 2. (Middle) shows simultaneous play, both agents update at the same time.

Scenario 3. (Right) is also a leader-follower interaction, but with the user leading. Importantly, the relationship between player actions will lead the system to converge to different equilibria.

all player decision variables. In such games, individual player decisions influence those of their opponents and are, in turn, influenced by their opponents' decisions. Depending on the relationship between player utility functions, these games can be zero-sum (utilities exactly opposed), potential (utilities exactly aligned), or general-sum (utilities neither exactly opposed nor aligned). If players are not willing or able to collude, the game is termed non-cooperative.

In the context of co-adaptive user-machine interfaces, it is generally appropriate to regard the user and machine as independent decision-making agents playing a non-cooperative game, since neither of the "intelligent" agents can know exactly what is on the other's "mind". Human decision-making may be influenced by a variety of factors including task performance, physical exertion, or personal preferences. Games that arise in user-machine interfaces are unlikely to be zero-sum by design, assuming the machine's goal is to assist the user. However, it may not be possible to perfectly align player utilities if the factors the human attends to and their relative weights in the human's utility are unknown or uncertain, hence the general-sum setting may be appropriate in many cases.

Game theory is commonly used to predict the outcome of agent interactions using different notions of stationary or equilibrium play. Multiple equilibria can arise in the same game, depending on the details of how agents adapt. For instance, Fig. 2.7 illustrates a co-adaptive system that converges to either Nash or one of two Stackelberg equilibria depending on the ratio of human and machine adaptation rates, which determines an order of play for the game. Game-theoretic equilibria generally represent trade-offs between players' conflicting goals. In the design of human-machine interfaces, we may seek machine adaptation schemes yielding equilibrium outcomes that favor the human. This theoretical framework presents a basis for simulating co-adaption using dynamic game theory and can be extended to tasks with multiple dimensions and to different decoder models. This framework can ultimately be used to inform adaptive decoder design to shape user learning and optimize user-machine interface performance as will be shown in the subsequent chapters.

Chapter 3

EVALUATING PROPOSED DECODER ADAPTATION SCHEME IN MYOELECTRIC INTERFACES

3.1 Introduction

Direct control of devices using neural or myoelectric activity has many applications, including brain-computer interfaces to restore function (BCIs) (Carmena, 2013; Shanechi et al., 2017), human-machine interfaces (HMIs) (De Santis, 2021), rehabilitation (Reinkensmeyer et al., 2016; Li et al., 2016), neuroprosthetics (Hochberg et al., 2012), and augmenting human capabilities (Willett et al., 2021). However, variability in the efficacy of neural interfaces across users (Zhang et al., 2020) and variability in neural signals within a single user over time (Yamagami et al., 2018) present challenges to the safety and performance of these interfaces.

Designing neural interfaces that adapt to individual users while also guiding how the user learns to control the interface can potentially improve usability. Such *co-adaptive* interfaces are more robust to variability across and within users by jointly optimizing the closed-loop interaction between the user and device. Previous studies explored how co-adaptation can reduce task error (Müller et al., 2017; Orsborn et al., 2012, 2014) and maximize interaction efficiency (De Santis, 2021). For example, Orsborn et al. (2012) proposes and validates *SmoothBatch*, a closed-loop adaptation algorithm that iteratively updates the decoder to reduce task error. However, algorithms that only consider error and ignore user effort lack theoretical guarantees of convergence. Co-adaptive interfaces present a new challenge: both the user and the decoder are adapting in a closed-loop system, creating a *two-learner*

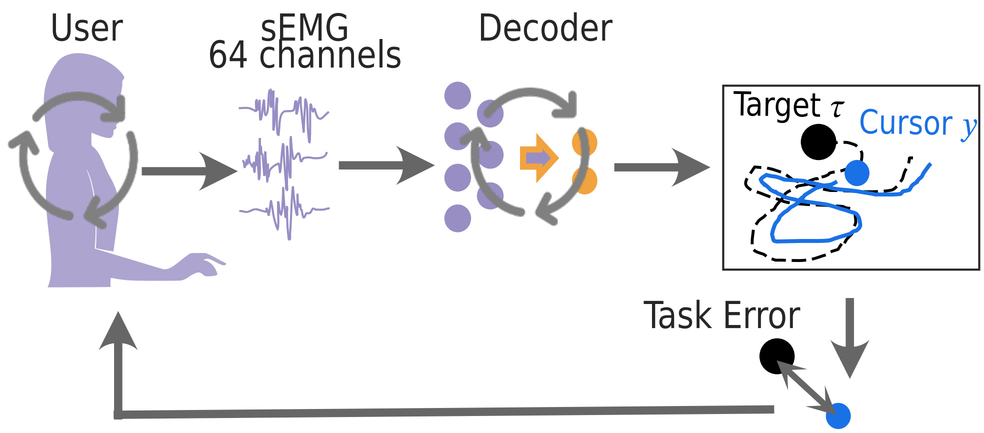

Figure 3.1: Myoelectric interface experiment schematic. Users track a target by controlling 2D cursor velocity with their muscle activity via an adaptive decoder. Feedback to the user is provided via task error between the target and cursor position.

problem. Thus, creating co-adaptive systems that converge to equilibria and are controllable for a diversity of users remains a challenge.

With the aim of developing and testing computational frameworks for measuring and shaping co-adaptive interactions, such as the one introduced in Chapter 2, we created an experimental platform to study co-adaptation using non-invasive myoelectric interfaces (Fig. 3.1). This platform created novel human-machine interactions for our human participants (N=14) which let us systematically manipulate decoder properties to investigate the effect of decoder adaptation on user learning. Our myoelectric interface had similar properties to many common motor neural interfaces, including brain-computer interfaces: it used high-dimensional biosignals to control a lower-dimensional movement (Seo et al., 2022; Portnova-Fahreeva et al., 2023) and was a closed-loop system that can facilitate user adaptation (Hahne et al., 2017). Past work highlights that co-adaptive myoelectric interfaces also face engineering challenges due to complex user-decoder dynamics (Hahne et al., 2015; Couraud et al., 2018). The non-invasive nature of myoelectric interfaces let us validate our approach on a larger subject cohort than is possible with implants.

Users were asked to control a 2D continuous cursor to follow a 2D target trajectory in a closed-loop myoelectric interface. We use a similar decoder update paradigm as SmoothBatch (Orsborn et al., 2014) and verify that our new adaptive decoder (proposed in Chapter 2) performs as expected; our decoder can be randomly initialized and adapt with the user-in-the-loop to improve control of a 2D continuous cursor. Human subjects learned to use our interface to perform a two-dimensional trajectory-tracking task. The focus of this chapter is the translation of our game-theoretic framework to experiment by testing a proposed adaptive decoder and decoder parameters in a myoelectric interface.

3.1.1 Related Work

Prior research modeling closed-loop interface dynamics has also combined both analytical model predictions with experimental results to reveal important adaptive decoder characteristics for future decoder design. For example, Müller et al. 2017 presented a model for co-adaptive human-machine interactions, treating the human and machine as two linear systems coupled by a joint loss function, and focusing on the learning rates of the two learners. Müller et al. 2017 implemented their model in a 1D mouse-controlled tracking task and empirically verified their model's prediction that decoder learning rates are constrained within a range for best performance results. Similarly, Dangi et al. 2013 modeled the convergence properties of closed-loop decoder adaptation algorithms, aiming to identify decoder parameters for rapid convergence or maintaining high performance in brain-computer interfaces. Dangi et al. 2013 validated their analytical predictions about closed-loop decoder convergence with empirical results from non-human primate brain-computer interface experiments. Leveraging their experimentally-tested mathematical analysis, Dangi et al. 2013 then determined ways to improve their adaptive decoding scheme and to inform future decoder design.

In recent years, game theory has been proposed as a framework to study and model two-learner dynamics in sensorimotor control (Braun et al., 2009; Li et al., 2019, 2016; Müller et al., 2017; Madduri et al., 2021). Game theory provides established techniques for predicting convergence to and stability of stationary points in two-learner systems (Ratliff et al., 2013; Başar and Olsder, 1999). Prior studies have leveraged potential games to explore how human-decoder interactions co-adapt with practice, both in simulation (Madduri et al., 2021) and in human-subject experiments (Li et al., 2016, 2019). For example, (Li et al., 2019) modeled a robot and human physically interacting using game theory, and demonstrated that each learner estimating the other's controller will converge to the Nash equilibrium. Such two-learner models could predict and shape the evolution of user-decoder dynamics. However, predictions from the game-theoretic framework have not yet been experimentally tested for neural interfaces.

3.1.2 Contributions of This Chapter

In this chapter, we propose and experimentally test a game-theoretic framework to optimize closed-loop performance of a myoelectric interface (using surface electromyography, sEMG) through co-adaptation of the user and decoder. In particular, we model the user and decoder as two players in a potential game and implement a natural (co-)adaptation strategy for the decoder: iteratively update to minimize the decoder's cost. We demonstrate that the adaptive algorithm can be used to optimize cursor control without prior initialization or calibration of the decoder.

In a human subjects experiment with 14 participants, we found that the user and decoder co-adapt to significantly improve tracking performance during a five-minute two-dimensional trajectory-tracking task. We explored three different decoder adaptation parameters: (1) decoder learning rate (i.e., rate of decoder adaptation), (2) decoder initialization, and (3) decoder penalty term (i.e., scaling of the decoder cost function). We found that decoder learning rate affected performance: the slower learning rate led to better tracking performance than the faster learning rate. We additionally found that neither decoder initialization nor decoder penalty term affected task performance. While closed-loop adaptive decoders are not new, this chapter contributes a systematic exploration of decoder adaptation parameters on closed-loop myoelectric interface performance. This chapter can be viewed as a stepping stone towards more complex co-adaptive interfaces that use closed-loop decoder adaptation (CLDA, Orsborn et al. 2012) as a tool for shaping and guiding user adaptation.

The chapter demonstrates:

1. experimentation and evaluation of a new decoder adaptation algorithm that performed similarly to previous decoder adaptation algorithms;
2. an exploration of decoder adaptation parameters on interface performance;
3. a general-purpose optimization tool to design decoders with multiple objectives.

3.2 Methods

3.2.1 Experimental Methods

Participants

Fourteen volunteers were recruited for this study and gave their written consent prior to experiments, according to study procedures approved by the University of Washington's Institutional Review Board (IRB #STUDY00014060). All participants were compensated monetarily for their time. All participants had no known motor disorders. Participant demographic information (including gender, weight, height, age, and handedness) was collected via a demographics survey that participants completed before the experiment (see Table 3.1). Only forearm circumference was measured by the experimenter, all other information was self-reported.

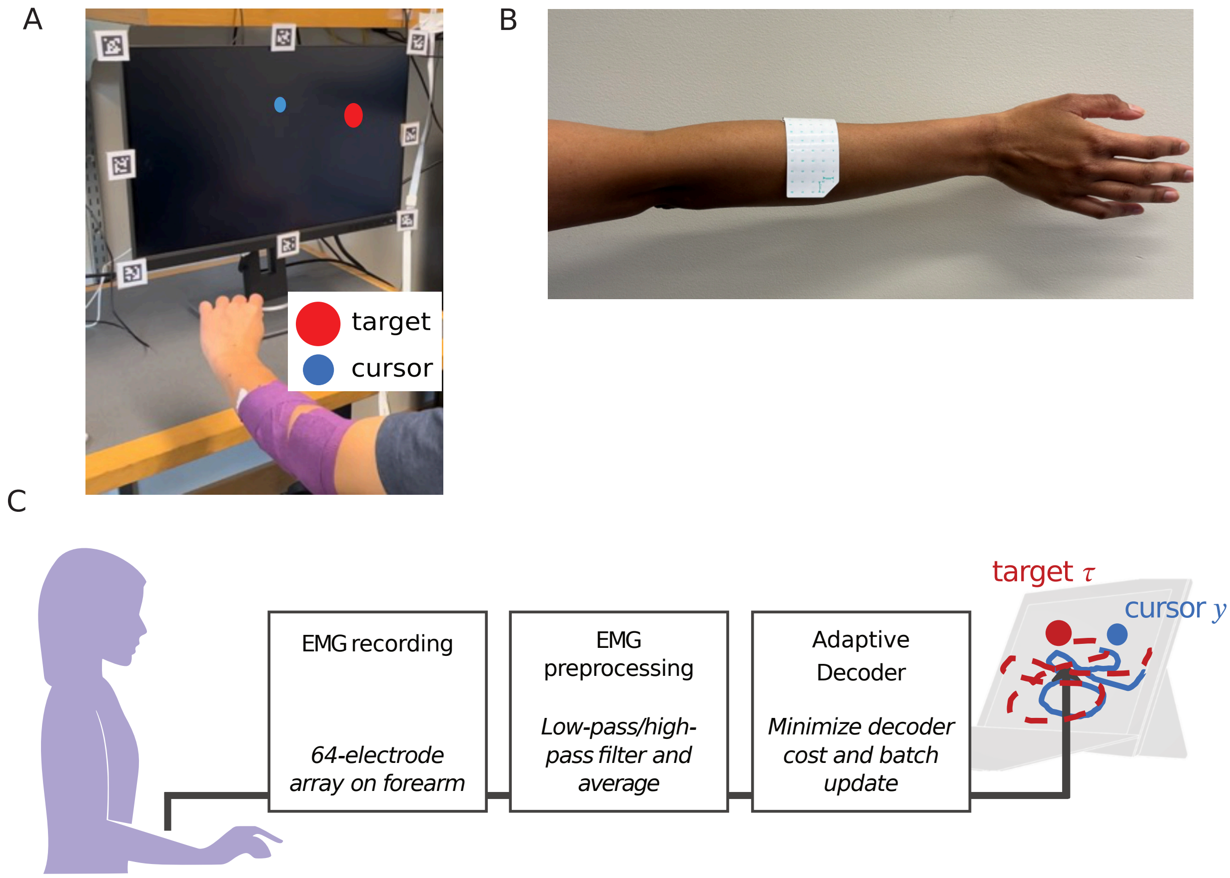

Figure 3.2: Experimental test-bed.

A. Photograph of experiment. The user tracks the target (dashed red) by controlling the velocity of a 2D cursor (solid blue) with their forearm muscle activity. Muscle activity is collected via surface EMG electrodes that are placed on the participant's dominant forearm and wrapped with Coban tape.

B. Electrode placement on dominant forearm. Surface EMG electrodes are placed on user dominant forearm to target Extensor Carpi Radialis.

C. Schematic of EMG preprocessing and decoding pipeline. EMG signals recorded from the user forearm are preprocessed and then input to the adaptive decoder. The decoder output is the cursor velocity that is integrated to display cursor position (blue) on the screen.

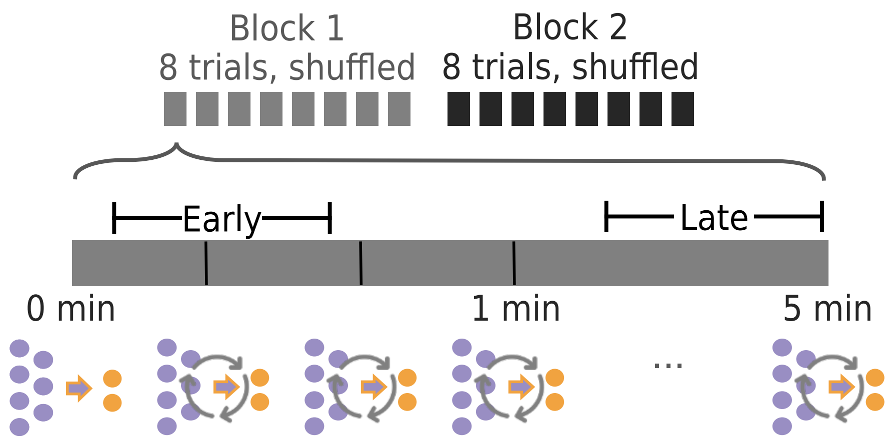

Figure 3.3: Experimental session schematic. Participants completed eight trials with different decoder conditions in both block 1 and 2 (random order). Each trial was 5 minutes. The decoder was initialized to random weightings and adapted every 20 seconds within trials.

Experimental design

Participants were asked to control a cursor on the screen – using muscle activity from their forearms measured via electromyography (EMG) – to follow a 2-D continuous target trajectory as closely as possible (Fig. 3.2A). Participants were told that they might not be able to control the cursor at the beginning of the trial but to expect that their cursor control would improve as the trial progressed. If the participant's cursor was stuck in the corner or edge of the screen for longer than 3.33 seconds, the cursor was automatically reset to the center of the screen. Participants sat in a chair with no restraints facing a computer screen (HP Compaq L2206tm, 1900$\times$1600, 46.5cm$\times$24.5cm). The computer screen displayed a red target circle (RGB: 1, 0, 0). The cursor was a blue circle (RGB: 0, 0, 1). The target was three times the area of the cursor (Fig. 3.2A).The target and cursor positions were updated and displayed at 60 Hz. The task was programmed using the `pygame` and `LabGraph` Python packages.

Following prior work (Yamagami et al., 2020; Yang et al., 2021; Yamagami et al., 2021), the target traversed a pseudorandom trajectory, which was generated by a sum-of-sinusoids with randomized phases with frequencies that were prime multiples of 0.05 Hz (x-axis frequencies = 0.10 and 0.25 Hz, y-axis frequencies = 0.15 and 0.35 Hz). We randomized the phase of the sum-of-sines so that the reference trajectory would be unpredictable to the participants. Distinct prime multiples were chosen in each direction to provide separability in the x- and y-axes for any frequency-based analysis. To ensure constant signal power, the magnitude of each frequency component was normalized by the frequency squared. The trajectory was different every trial. Each trial was 5 minutes with a 5-second ramp period during which the cursor and target speed slowly increased from stationary to the experimentally-prescribed speeds. This ramp period followed prior experiments (Yamagami et al., 2021) and was added to give participants time to recognize the starting cursor and target movements. Participants completed 16 trials of 5-minutes each in 2 blocks; each block consisted of 8 trials. Participants were given a 5-minute break in between each block but their EMG array and placement did not change throughout the experiment (Fig. 3.3).

EMG signal collection and processing

EMG signals were obtained using a Quattrocento system (Bioelettronica, Italy). A 64-channel high-density surface EMG electrode array (8mm inter-electrode spacing, 5×13 electrode rectangular layout) was placed on the dominant forearm of each participant, targeting the Extensor Carpi Radialis (Fig. 3.2B). Electrodes were placed on the dominant arm for each participant. Once placed, the electrode array was wrapped with Coban self-adherent wrap (3M, Saint Paul, Minnesota). The electrode cables from the Quattrocento to the array were secured to minimize motion artifacts.

EMG signals were acquired using Biolite Software (Bioelettronica, Italy) at 2048 Hz with Differential Mode with the built-in low-pass filter of 130 Hz and a high-pass filter of 10 Hz.We filtered and rectified the EMG data following prior pre-processing techniques to compute the EMG linear envelope (Yamagami et al., 2018; Chvatal and Ting, 2013): a 40-

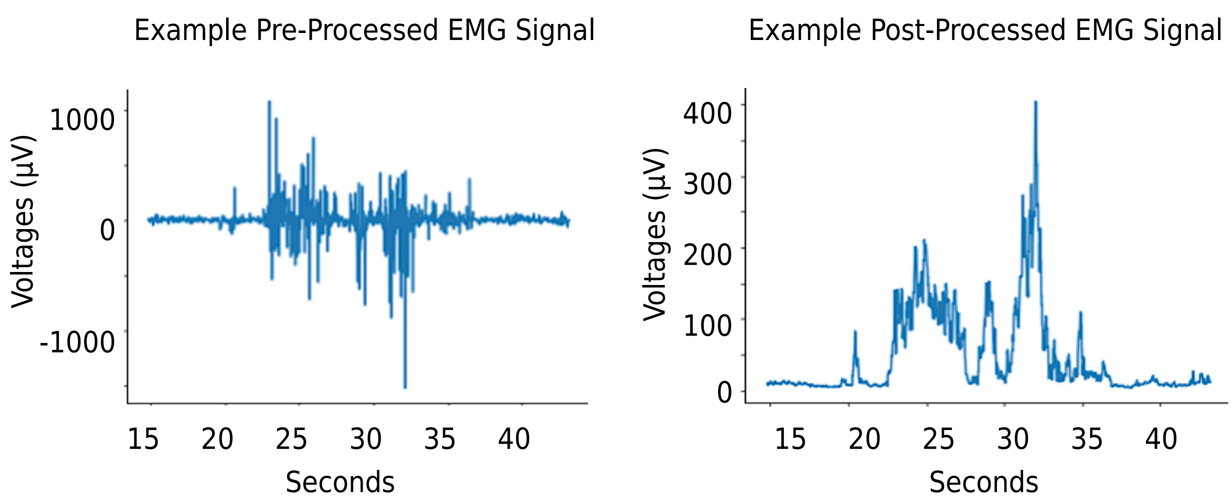

Figure 3.4: Example pre-processed and post-processed EMG signal. (Left) Example EMG signal acquired from one channel of high-density sEMG electrode via Quattrocento Bioelettronica. **(Right)** Example of same EMG signal after filtering and rectification.

Hz high-pass filter, rectification, and a 40-Hz low pass filter. A moving average filter was applied to downsample EMG signals from 2048Hz to 60Hz (Fig. 3.4).

3.2.2 Myoelectric Interface Decoder

Real-time myoelectric control was implemented with a velocity-controlled Wiener filter that output a cursor velocity $v_t \in \mathbb{R}^2$,

$$v_t = D \cdot u_t \,, \tag{3.1}$$

where $u_t \in \mathbb{R}^N$ is the processed EMG signals of $N = 64$ channels and $D \in \mathbb{R}^{2\times N}$ is the decoder mapping. The cursor velocity is integrated to output a cursor position $y_t \in \mathbb{R}^2$,

$$y_t = y_{t-1} + v_t \Delta t \,. \tag{3.2}$$

where Δt is the time between cursor updates.

Myoelectric Interface Decoder Adaptation

Decoder adaptation consisted of two steps:

Step 1: Calculate optimal decoder D^*

Calculate the optimal decoder D^* by minimizing the cost function c_D based on the previous 20 seconds of user and trial data:

$$D^* = \min_D c_D(D)\,. \tag{3.3}$$

Step 2: Update next decoder D

Then, update the next decoder D following the SmoothBatch approach (Orsborn et al., 2012), which uses a weighted combination of the prior decoder D^- and the optimal decoder D^*:

$$D = \alpha D^- + (1-\alpha)D^*, \tag{3.4}$$

where α is the learning rate.

The decoder cost function was minimized using `SciPy`. To satisfy real-time timing constraints on decoder cost minimization, we initialized $D_0^* = (\tau - y) \cdot s^+$ using the previous 20 seconds of data (Ch. 2). At the start of each trial, the decoder was initialized by randomizing the decoder weightings from a uniform distribution (using `NumPy.random.rand`) and multiplying by a scalar factor. Each user had two different decoder initializations, referred to as *D1* and *D2*. Initial decoders for each participant's trial were set to either D1 or D2 only. Note, while we programmed the decoder to update every 20 seconds, due to slight software imprecision, the decoder update actually occurred approximately every 18 seconds.

Decoder Cost

The decoder cost function was formulated in earlier work (Ch. 2) and aims to minimize both the tracking error and decoder effort as the decoder adapts. Minimizing velocity error is a common goal in user-machine interfaces, seen widely across neural interfaces (Orsborn et al., 2014; Wodlinger et al., 2015), body-machine interfaces (Seáñez and Mussa-Ivaldi, 2013), and myoelectric interfaces (Hahne et al., 2015; Radhakrishnan et al., 2008). Decoder effort is considered as part of the decoder cost since our prior theoretical analysis suggested that a regularization term in the user and decoder costs is necessary to ensure convergence in the user-decoder co-adaptation game to stable stationary points (Madduri et al., 2021).

The decoder cost c_D is constructed as a linear combination of the task *error* and decoder *effort*:

$$\text{error} = \|D \cdot u - \frac{1}{\Delta t}(\tau - y)\|_2^2\,,\ \text{effort} = \|D\|_F^2\,; \tag{3.5}$$

here $\|x\|_2$ denotes the 2-norm of the signal $x : [0, t] \to \mathbb{R}^d$, and $\|X\|_F$ denotes the Frobenius norm of the matrix $X \in \mathbb{R}^{m \times n}$. The decoder cost c_D is then,

$$c_D = \|D \cdot u - \frac{1}{\Delta t}(\tau - y)\|_2^2 + \lambda_D \|D\|_F^2, \tag{3.6}$$

where λ_D is the penalty term of the decoder effort.

Decoder Conditions

We varied the decoder cost function and parameters of decoder adaptation to determine how it influenced system performance and user behavior. Specifically, the role of (1) decoder learning rate α, (2) decoder penalty term λ_D, and (3) decoder initialization on co-adaptive system performance was tested.

1. **Decoder Initialization:** Two randomized initializations of the D decoder matrix were used, D1 and D2: *positive* (matrix elements chosen uniformly at random in the range $[0, 10^{-2}]$) and *negative* (elements chosen uniformly at random in $[-10^{-2}, 0]$). The gain (10^{-2}) of the decoder initialization was chosen by empirically testing whether the initial cursor velocity was controllable or not. If the initial gain of the decoder was too high, the initial cursor velocity would be too high for participants to even attempt control.

2. **Decoder Learning Rate:** Two different decoder learning rates were tested: *slow* ($\alpha = 0.75$) and *fast* ($\alpha = 0.25$). A slow learning rate meant that the new decoder had more emphasis on the previous decoder, D^{-}. A fast learning rate meant that the new decoder had more emphasis on the optimal decoder, D^{*}. Initially, three learning rates were tested in pilot experiments: $\alpha = 0.25$, $\alpha = 0.5$, and $\alpha = 0.75$. We empirically found that only the fastest learning rate, $\alpha = 0.25$, negatively impacted control, and the other two learning rates did not have effects that were significantly different from each other. Thus, we chose the two extreme learning rates to test for our experiments: $\alpha = 0.75$ and $\alpha = 0.25$.

3. **Decoder Penalty Terms:** Two decoder penalty terms were tested: *low* ($\lambda = 10^2$) and *high* ($\lambda = 10^3$). These were the regularization terms in Eq. 3.6. In the case with the higher decoder penalty term, minimizing the decoder cost meant that there was more emphasis on minimizing the decoder effort and less emphasis on minimizing the task error as compared to the lower decoder penalty term scenario. Similar to the learning rates, we empirically tested multiple different penalty terms before running our experiments. Since we wanted to primarily test the impact of the decoder penalty term on the decoder and user's efforts, we chose two different penalty terms that participants could still control and would not impact task error.

We tested all combinations of learning rates, penalty terms, and initializations, leading to a total of eight different decoder conditions.

3.2.3 Data Analysis

Our primary metric for quantifying performance was the tracking error, calculated as the Euclidean distance between the target and the cursor, $\|\tau - y\|_2$. We assessed changes in task performance within a trial by comparing the mean tracking error in the first (early) and last (late) 30 seconds of the trial (excluding ramp-up time). Because participants' proficiency in tracking varied, we quantified improvements over time by calculating the relative error: $\frac{error_{fi}}{error_i} \times 100\%$, where error$_{fi}$ = error$_{final}$—error$_{initial}$.

3.2.4 Statistical Analyses

All analyses treated participants as individual data points and computed the mean across decoder conditions (learning rate, penalty terms, initialization) and trials. To assess statistical significance across time and conditions for each participant, we used a two-sided Wilcoxon signed-rank test (`scipy.stats.wilcoxon`), which is a paired, non-parametric test. We chose a non-parametric statistical test because of subject-to-subject variability. Figure boxplots were plotted with `Matplotlib.pyplot.boxplot` (center line, median; box limits, upper and lower quartiles; whiskers, 1.5x interquartile range; fliers not plotted).

3.3 Results

3.3.1 Trajectory tracking performance improved

Task performance, as measured by time-domain tracking error, improved within individual trials (Fig. 3.5A): comparing tracking error *early* (first 30 sec) and *late* (last 30 sec) in trials across all conditions (learning rate, initialization, decoder effort weight, and trial block), we found that performance significantly improved within trials (Fig. 3.5B, C).

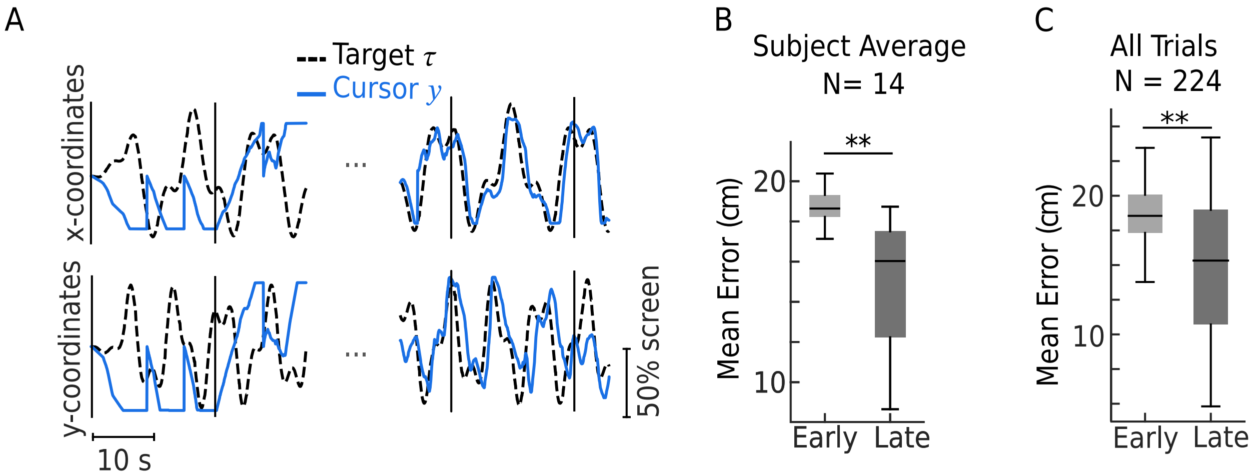

Figure 3.5: Target and Cursor Trajectory Tracking. Comparison of the target (red dashed trace, τ) and cursor (blue straight trace, y) positions over time for one example trial for one subject. The y-axis is normalized coordinates, and the x-axis is the time in minutes within the trial. The black vertical lines represent decoder adaptations. Horizontal (x) and vertical (y) positions are shown in the top and bottom plots, respectively.

A. Example cursor y (solid blue) and target τ (dashed black) trajectories over time for one example trial for one subject. Vertical black lines represent decoder updates. The y-axis is normalized coordinates, and the x-axis is the time in minutes within the trial. The black vertical lines represent decoder adaptations. Horizontal (x) and vertical (y) positions are shown in the top and bottom plots, respectively.

B. Mean tracking error (N = 14, median with shading indicating interquartile range) early (first 30 sec) and late (last 30 sec) in trials. Error ($\|\tau - y\|$) was averaged across trials for each participant. Wilcoxon signed-rank test, **p < 0.001.

C. Mean tracking error for all trials (N = 224, median with shading indicating interquartile range) early (first 30 sec) and late (last 30 sec) in trials. Error ($\|\tau - y\|$) was computed for each trial. Wilcoxon signed-rank test, **p < 0.001.

3.3.2 Decoder initialization did not affect interface performance

Performance improved as the decoder adapted during a trial and these performance improvements were not impacted by decoder initialization (Fig. 3.6), similar to invasive brain-computer interface adaptive decoder studies (Gilja et al., 2012; Orsborn et al., 2012; Dangi et al., 2013).

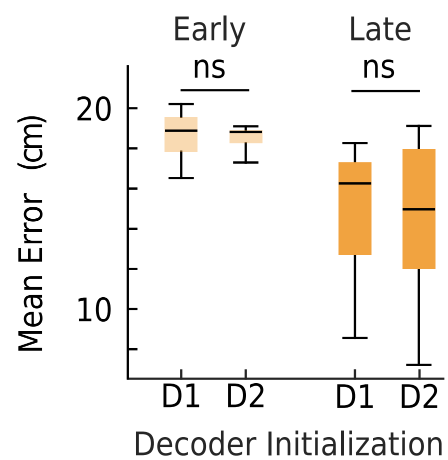

Figure 3.6: Performance improved regardless of decoder initialization. Mean error (N = 14, median with shading indicating interquartile range) separated by decoder initialization (D1, D2) early and late in trials. Error ($\|\tau - y\|$) was averaged across trials for each participant. Wilcoxon two-sided signed-rank test, ns > 0.05.

3.3.3 Decoder learning rate affected performance

The slow learning rate yielded better tracking performance (Fig. 3.7): comparing the relative error between the *early* and *late* 30 seconds of the trial, we found a significantly greater improvement in performance for the slow learning rate than for the fast learning rate (Fig. 3.8).

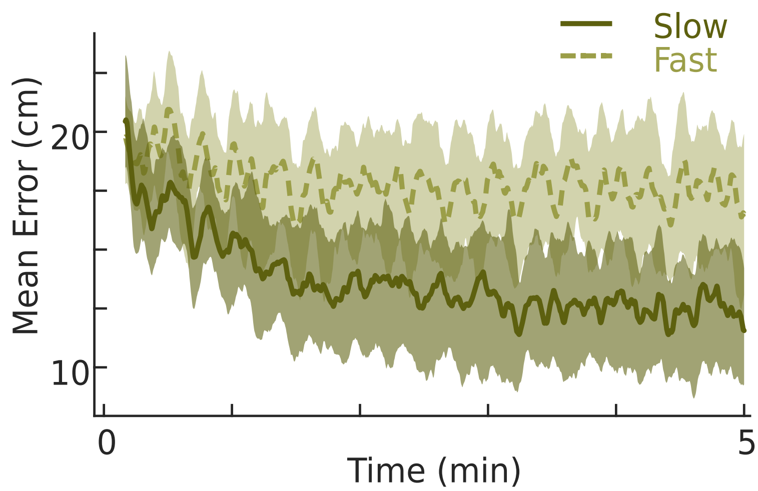

Figure 3.7: Tracking Error by Learning Rate. Median tracking error per subject ($N = 14$) over the five-minute trial, separated by decoder learning rate (slow vs. fast). Error was calculated at each time point and smoothed with a low-pass filter over 5 seconds for visualization. Solid lines show the median, and shading shows 25% interquartile.

3.3.4 Only decoder learning rate affected performance

Comparing the relative error between the *early* and *late* 30 seconds of the trial, we found that only learning rate significantly impacted performance. Neither decoder initialization nor penalty significantly affected performance (Fig. 3.8).

3.3.5 Decoder penalty term affected decoder effort

Comparing the relative error of the *first* to the *last* 20 seconds between the *high* and *low* decoder cost weights, we found no significant effect on task performance (Fig. 3.8c). However, we did observe an effect on the Frobenius norm of the decoder based on the decoder cost weight (Fig. 3.9). The decoder Frobenius norm is overall higher for the *low* decoder cost weight than for the *high* decoder cost weight.

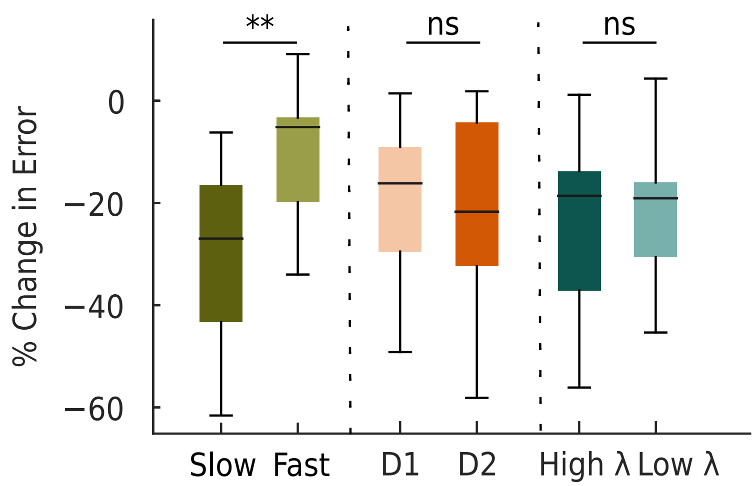

Figure 3.8: Median Relative Error Across Decoder Conditions. Median relative error across all blocks and for each learning rates for each subject (slow—dark green, fast—light green), for each decoder initialization for each subject (D1—light orange, D2—dark orange), and for each decoder penalty term λ for each subject (high λ - dark blue, low λ - light blue); statistical comparisons across conditions ($N = 14$) with a Wilcoxon two-sided signed-rank test (slow vs. fast: $p = 0.0001$, D1 vs. D2: $p = 0.6698$, high λ vs low λ: $p = 0.326$). Only (a) is statistically significant ($p < 0.05$)).

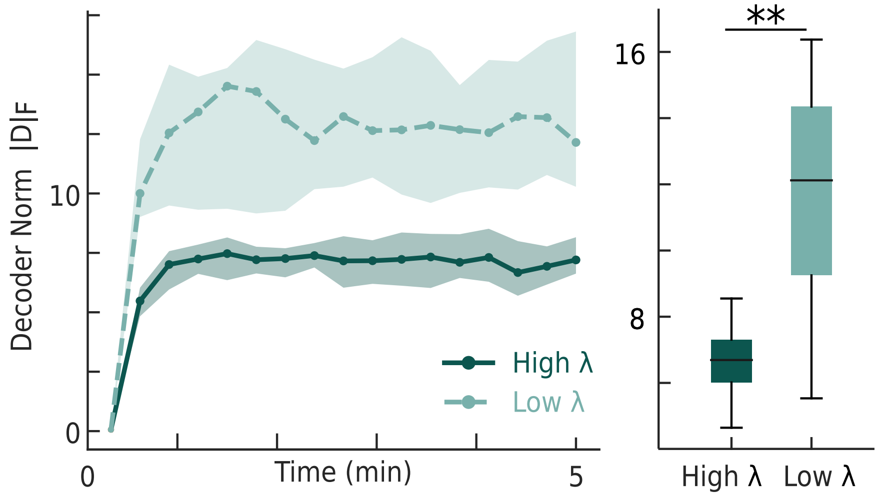

Figure 3.9: Decoder Norm Distributions Across Penalty Terms. (Left) Decoder norm ($||D||_F$) distributions across trial (lines show the median, shading shows 25% interquartile region) for median decoder cost weight per subject ($N = 14$) across conditions (blocks, learning rate, decoder initialization). Solid dark blue trace indicates high penalty, light dashed blue trace indicates low penalty. Markers indicate decoder updates. **(Right)** Decoder norm averaged across trial then averaged across subject. Wilcoxon two-sided signed-rank test, ns > 0.05.

3.4 Discussion

The aim of this study was to compare the performance of a new adaptive decoder with performance from prior CLDA experiments. We observed that updating a randomly-initialized decoder to minimize *task error* and *decoder effort* enabled users to control a 2-dimensional trajectory-tracking task with no calibration. These performance results were in line with prior research (Orsborn et al., 2014). Performance was affected by decoder learning rate but not by initialization nor by cost weights. However, decoder initialization and cost weights influenced the final decoder, suggesting a potential avenue to influence user learning.

In contrast to prior work that optimized exclusively for task error (Orsborn et al., 2012, 2014; Müller et al., 2017) or interaction efficiency (De Santis, 2021), our experiments employed a new decoder adaptation scheme that optimized a cost function that captures the trade-offs of a two-learner human-decoder interface. Similar to previous studies (Orsborn et al., 2012), our co-adaptive framework yielded rapid calibration independent of initialization. But our framework can potentially shape both the decoder *and* the human by changing parameters such as decoder learning rate, initialization, and cost weights.

Similarly to previous studies of learning rate on human-decoder co-adaptation that optimized for task error (Müller et al., 2017), we found that a slow learning rate resulted in a lower task error than a fast learning rate. This may be because slow decoder adaptation is less affected by random fluctuations in myoelectric activity or changes in human effort. Performance outcomes could differ with longer trials. Studying variability and performance over longer time scales is important for future work.

Our preliminary analysis suggests that decoder initialization location did not affect task performance. The robustness of task performance despite varying decoder initialization locations is congruent with prior co-adaptive algorithms that solely optimized for task performance (Orsborn et al., 2012; Shanechi et al., 2016). Initial performance is a result of the random decoder initialization, so low variance in initial performance across subjects does not reflect subject performance. Final performance is due to decoder adaptation and user performance, hence the larger variance across subjects in final versus initial performance. Lastly, the theoretical analysis in Madduri et al. (2021) suggested that decoder initialization may bias the system towards different equilibria, which we may have observed in this study. Additional analyses and longer-term experiments may help quantify the system equilibria and the effect of decoder initialization location on the final decoder location. We also acknowledge that experiments with a higher number of subjects would be valuable to strengthen these results.

Lastly, our experiments suggest that decoder penalty terms influenced the learned decoder without impacting task performance. This finding that a higher decoder cost weight led to a lower decoder Frobenius norm is consistent with theoretical expectations of minimizing the decoder cost. Task performance being unaffected by different decoder norms suggests that users may be able to learn multiple decoders. Users might be potentially compensating for different decoders in their learned strategies. A particularly compelling question for future work is whether the user's encoding or control strategy is biased by the decoder parameters of the learning rate, initialization, and penalty terms.

Designing neural interfaces that can adapt to a wide range of users is key to improving neural interface adoption and usability. Neural interfaces that adapt to individual users and shape user learning through co-adaptation could lead to individualized and robust neural interfaces. We approach the analysis and synthesis of co-adaptive neural interfaces from a game-theoretic perspective that treats the human and decoder as two independent agents in a game. This chapter informs an initial exploration of the effect of a game-theoretic adaptive decoder framework and differing parameters on user learning. We found that decoding parameters systematically influence the interface performance and also match the expected predictions from the game-theoretic model. We continue to analyze these predictions through the remaining chapters.

Table 3.1: Participant Demographics

	Responses
Gender	Women: 9 Men: 4 Prefer Not to Answer: 1
Weight (mean $\pm$ std, lbs.)	140 $\pm$ 21.6
Height (mean $\pm$ std, in.)	65 $\pm$ 3.4
Age (mean $\pm$ std, yrs.)	23 $\pm$ 3.9
Forearm Circumference (mean $\pm$ std, in.)	10 $\pm$ 0.8
Handedness	Right-Handed: 12 Left-Handed: 2
Level of Ambidexterity	Dominant: 8 Mostly Dominant: 5 Ambidextrous: 1

Chapter 4

ENCODER ESTIMATION METHODS REVEAL CLOSED-LOOP INTERACTIONS

4.1 Introduction

Distinguishing user and decoder contributions in an interface is key to understanding and designing closed-loop, co-adaptive systems. As we begin to tune interfaces for different users and purposes, we need to be able to measure inputs from both users and decoders. Understanding whether and what changes in interface performance, for example, are due to user adjustments or decoder updates is key to programming and optimizing future interfaces. While decoders are known exactly and can be easily interpretable, the user model must be estimated. Decoding algorithms inherently depend on the user's encoding. Analyzing if and how a user's encoder changes in relation to the task and to the decoder output provides approaches to designing decoders that can influence or shape users encoders. In this chapter, we build upon control theory methods in continuous trajectory tracking to estimate a user's model and demonstrate how this model can be used to quantify user-decoder interactions.

Direction-tuning analysis is often used with neural and myoelectric interfaces (e.g., Georgopoulos et al. 1986; Couraud et al. 2018; Radhakrishnan et al. 2008) to characterize user activity in a task-relevant manner. However, direction-tuning analysis does not provide a model for the user and, therefore, limits our ability to examine user-decoder interactions. We used methods from control theory, a discipline focused on analysis and synthesis of feedback systems, to model the behavior of users and the closed-loop system. We refer to our model of the user as the user's "encoder". Prior work shows that control theory can estimate changes in user trajectory-tracking behavior by separating their feedback and feedforward controllers (McRuer and Jex, 1967; Drop et al., 2013; Zhang et al., 2018; Yamagami et al., 2020). We extended these techniques to our multi-input, multi-output system to analyze co-adaptive outcomes.

4.1.1 Related Work

Encoder models have provided insight to how users adapt and learn across different types of interfaces, including invasive neural interfaces (e.g. Golub et al. 2015; Merel et al. 2015; Liang and Kao 2020 and non-invasive myoelectric interfaces (Yamagami et al., 2020). Golub et al. 2015 use BCIs to probe the formation of encoder models (referred to as internal models in the paper) and show that encoder models are formed to reflect learning from sensory feedback in a closed-loop context. Golub et al.'s internal model framework uses a time-based scheme to estimate the user's process of internally prediction cursor state. Importantly, Golub et al. show that internal models of users can predict the user's intention and that the internal model changes with learning. Similarly, Yamagami et al. 2020 compare user behavior and performance in manual and myoelectric interfaces by estimating user internal models. Leveraging control theory techniques, Yamagami et al. develops methods to measure *feedforward (user)* and *feedback (error correction)* controllers in closed-loop interfaces. These frequency-domain techniques have been previously used in characterizing user behavior in trajectory-tracking tasks (McRuer and Jex, 1967; Drop et al., 2013; Zhang et al., 2018) and can be extended to tasks with more complex (acceleration-based) dynamics and to disturbance-rejection tasks. Changes in users' feedback and feedforward controllers can be used to reveal user learning in closed-loop systems. In a similar vein, the encoder model introduced in this chapter is also used to understand and describe user learning within and across trials.

Encoder models also provide opportunities to simulate closed-loop user-decoder behavior and can help de-risk experimental setups. Merel et al. 2015 model encoder-decoder dynamics for co-adaptive interfaces to find an optimal decoder that the encoder could learn. Merel et al. use a split Linear Quadratic Gaussian system to model the decoder as a Linear Quadratic Estimator (the Kalman Filter) and the encoder as a Linear Quadratic Regulator. Also using linear methods, Willett et al. 2021 develop a piece-wise-linear model to simulate online BCI performance. In their paper, Willet et al. model the user with a state-based feedback control model — the user employs a control policy for a forward model to estimate the cursor velocity and position. The user's forward model is then updated with delayed visual feedback of the decoded cursor velocity and position. The Willet methods could be used to predict closed-loop gain and smoothing parameters of the decoder before experimentation, demonstrating the utility of encoder models in developing future neural interfaces. our encoder model provides methods to measure and quantify encoder-decoder interactions observed during experimentation and is particularly useful in our work comparing different decoder designs.

4.1.2 *Contributions of this Chapter*

In this chapter, we develop encoder estimations methods that we can then use to measure and computer encoder-decoder interactions in high-density myoelectric interfaces. This chapter demonstrates that:

1. control theory can be used to develop models for users' encoders and for encoder-decoder interactions in stable closed-loop systems;
2. encoder-decoder interactions in experimental data can be measured;
3. the encoder estimation model can reveal changes in users encoders within- and across-trials.

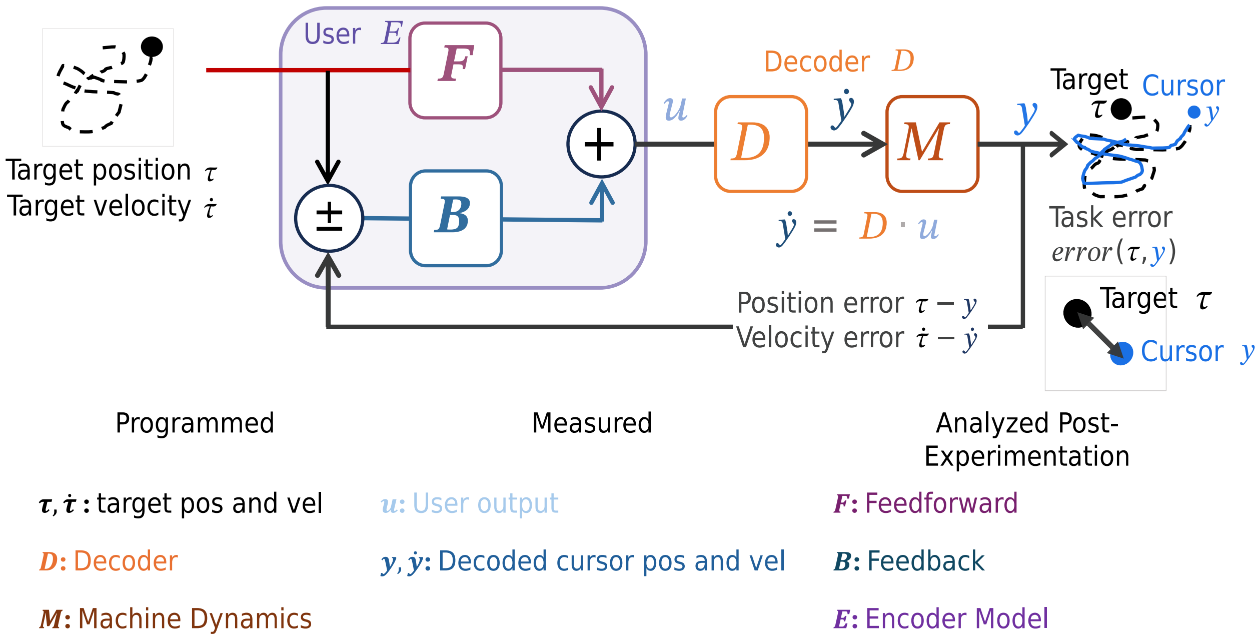

Figure 4.1: User-Machine Interface Closed-Loop Control System. The user's *Encoder*, E, outputs myoelectric or neural activity u to the decoder D to follow the target's position τ and the target's velocity $\dot{\tau}$. The output of the decoder D is a velocity, which is then integrated by the machine M to a cursor position. The user sees their cursor moving with some velocity $\dot{y}$ to position y. The error between the cursor position and target position Figure modified from Yamagami et al. 2020.

4.2 Methods

4.2.1 Encoder Estimation Methods

We can consider our myoelectric interface as a control loop in the format of Fig. 4.1. We constructed a u matrix of EMG signals of $t = 1200$ samples (corresponding to 20 seconds of data at 60 Hz) and a matrix from task information (target position $\tau \in \mathbb{R}^{2 \times t}$, target velocity

$\dot{\tau}$, position error $\tau - y$. and velocity error $\dot{\tau} - \dot{y}$),

$$u = E \cdot \begin{bmatrix} \tau \\ \dot{\tau} \\ \tau - y \\ \dot{\tau} - \dot{y} \end{bmatrix} + \beta \cdot 1_t^\top, \tag{4.1}$$

where $1_t^\top = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{1\times t}$, $u \in \mathbb{R}^{N\times t}$, $E \in \mathbb{R}^{N\times f}$, N is the number of signals, d is the dimension of task information, and β is an offset factor for estimation ($\beta \in \mathbb{R}^{N\times 1}$). In this paper, $N = 64$ EMG channel inputs, $d = 8$ (target position, target velocity, position error, and velocity error for both x and y dimensions), and $t = 1200$ (20 seconds at 60 Hz sampling rate). The encoder E was estimated using linear regression (`sklearn.linearmodel.LinearRegression`) with a y-intercept offset, so there was one E estimate for each 20 seconds of data. This was also checked with the Moore-Penrose pseudo-inverse (`NumPy.linalg.pinv`).

4.2.2 Assessing Fit of Encoder Estimation

We estimated the accuracy of the myoelectric encoder model by reconstructing EMG signals from the encoder estimation and compared the coefficient of determination (R^2) for the cursor velocity and position decoded from the reconstructed EMG to the actual cursor velocity and position recorded in the trial. To establish a baseline for the predictive power of our encoder model, we also computed the accuracy of cursor velocity and position decoded from time-shuffled EMG data (Fig. 4.2). The time-shuffled EMG data was created by permuting the EMG signals within each trial, so that the EMG data came from the same subject and trial but different time points.

Note, the results shown in this section estimated the encoder across $t = 1200$ samples (corresponding to 20 seconds of data at 60 Hz). Accuracy of the encoder estimation method does suffer if the encoder is estimated across $t = 3600$ samples (corresponding to 60 seconds of data at 60 Hz), as shown in Fig.

4.2.3 *Analytical Predictions of Decoder-Encoder Dynamics*

We used control theory principles to predict relationships between elements of the encoder E and decoder D. There are two conditions that are desirable for the encoder and decoder:

1. perfect trajectory tracking, where the encoder's output is translated by the decoder to perfectly track the target, and
2. closed-loop stability, which ensures predictable responses and an ability to withstand small perturbations.

Trajectory tracking conditions correspond with feedforward control, whereas closed-loop stability conditions correspond with feedback control. We regarded the encoder and decoder as linear time-invariant (LTI) transformations and analyzed them in the frequency-domain with Laplace variable s.

Perfect Trajectory Tracking

We can say that this system will have perfect trajectory tracking when user's feedforward input is the perfect inverse of the decoder and machine. In the frequency-domain, the machine transformation is $M(s) = \frac{1}{s}I(s)$ where I is the *identity* transformation, that is, $I(s)$ is an identity matrix for each $s \in \mathbb{C}$. We can define the user's feedforward transformation as,

$$F(s) = F_0 + F_1 \cdot s. \tag{4.2}$$

Perfect trajectory tracking can occur only if

$$\begin{aligned} I &= M \cdot D \cdot F \\ &= \frac{1}{s} \cdot D \cdot F_0 + D \cdot F_1. \end{aligned} \tag{4.3}$$

From there, we can compute the following desirable conditions for perfect trajectory tracking:

$$F_0 = 0 \tag{4.4}$$

and

$$D \cdot F_1 = I. \tag{4.5}$$

Extending this to the case where $D \cdot F_0 \in \mathbb{R}^{2\times 2}$, $D \cdot F_0 = 0 \in \mathbb{R}^{2\times 2}$ and $D \cdot F_1 = \mathbb{I}^{2\times 2}$.

Closed-loop Stability

To ensure the second condition of closed-loop stability, we look at the user's feedback transformation, which we can define as,

$$B(s) = B_0 + B_1 \cdot s. \tag{4.6}$$

It suffices that $D \cdot B_0$ and $D \cdot B_1$ are negative-definite diagonal matrices for closed-loop stability (Åström and Murray, 2008, Chapter 5).

4.3 Results

4.3.1 Encoder Model Can Estimate Experimental Details

We define our model of the user's encoder as the closed-loop mapping between task information and EMG activity in our myoelectric interface (Fig. 4.1). Our encoder model, $E \in \mathbb{R}^{64\times 8}$, considers feedforward and feedback inputs to the user, which are derived from closed-loop task information presented to the user — target position ($\tau \in \mathbb{R}^2$), target velocity ($\dot{\tau} \in \mathbb{R}^2$), position error ($\tau - y \in \mathbb{R}^2$) and velocity error ($\dot{\tau} - \dot{y} \in \mathbb{R}^2$). We formulate

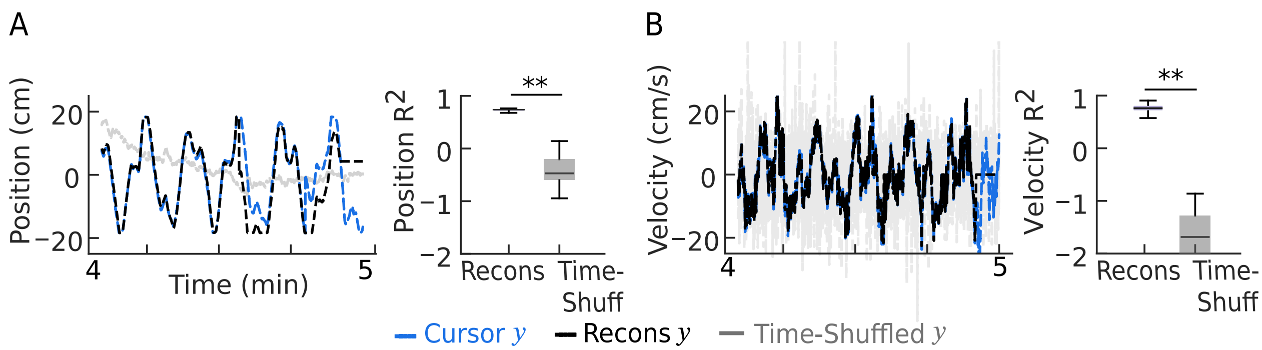

Figure 4.2: Encoder estimation can reconstruct cursor position and velocity data. Example reconstructions of cursor position **(A)** and velocity **(B)** from encoder estimations (dashed black), compared to the actual position and velocity (dashed blue) with a time-shuffled baseline (solid gray). Distributions of average R^2 values between reconstructed and time-shuffled control for (A) position (average reconstructed position $R^2 = 0.72$, average time-shuffled position $R^2 = -.44$), and (B) velocity (average reconstructed velocity $R^2 = 0.75$, average time-shuffled position $R^2 = -2.33$) (N = 14, median with interquartile range; Wilcoxon signed-rank test, **p < 0.001).

an encoder model with these feedforward and feedback elements that depend linearly on target position, cursor error, and an offset β (to capture resting activity):

$$u = E \cdot \begin{bmatrix} \tau \\ \dot{\tau} \\ \tau - y \\ \dot{\tau} - \dot{y} \end{bmatrix} + \beta \cdot 1_t^\top, \tag{4.7}$$

where $1_t^\top = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{1 \times t}$, and t indicates time.

The encoder matrix has both feedforward (F) and feedback (B) components:

$$E = \begin{bmatrix} F_0 & F_1 & B_0 & B_1 \end{bmatrix} \tag{4.8}$$

such that each component of task information, τ, $\dot{\tau}$, $\tau - y$, $\dot{\tau} - \dot{y}$, respectively corresponds with an element of the encoder, $F_0 \in \mathbb{R}^{64\times2}$, $F_1 \in \mathbb{R}^{64\times2}$, $B_0 \in \mathbb{R}^{64\times2}$, $B_1 \in \mathbb{R}^{64\times2}$. The subscripts 0 and 1 represent the 0^{th}- and 1^{st}-order dynamics of position and velocity, respectively. The output of the encoder is an EMG activity time series $u \in \mathbb{R}^{64\times t}$ which is input to the decoder, $D \in \mathbb{R}^{2\times 64}$. The decoder outputs a cursor velocity, which is integrated by the system dynamics M to a cursor position y.

We estimated the user's encoders $E \in \mathbb{R}^{64\times 8}$ within a trial with linear regression using 20-second batches of data, corresponding to the time intervals where the decoder was held constant (see Sec. 4.2.1). This model appeared to capture useful aspects of user behavior, yielding reconstructed cursor position and velocity trajectories that were correlated with the actual cursor movements (Fig. 4.2B).

4.3.2 Decoder-Encoder Interactions Can Be Measured Using Encoder Model

We then used our encoder model to quantify decoder-encoder interactions and analyze performance of the closed-loop system. One common challenge in co-adaptive systems is convergence to a stable solution or performance (Dangi et al., 2013; Müller et al., 2017; Hsieh and Shanechi, 2018). Performance-based analysis from above (See Ch. 3) suggested that our co-adaptive interface may have converged, but does not prove stability nor ideal performance. We exploited the separation of the feedforward and feedback pathways in our encoder model to empirically measure the stability and tracking error properties of the closed-loop encoder-decoder system. Perfect trajectory tracking is obtained when the user's feedforward input F becomes the pseudo-inverse of the decoder D and system dynamics M, which results in the following conditions: $D \cdot F_0 = 0 \cdot \mathbb{I}$ and $D \cdot F_1 = s \cdot \mathbb{I}$, where s is the frequency-domain Laplace variable. (See Sec. 4.2.3). We used the known decoder values and estimated user encoders to quantify these values in our experimental data, which revealed that $D \cdot F_0$ and $D \cdot F_1$ approach the analytically-ideal values on average across all trials (Fig. 4.3A). Similarly, closed-loop stability can be achieved if the feedback components of the decoder-encoder product ($D \cdot B_0$ and $D \cdot B_1$) are negative-definite diagonal matrices (Åström and Murray, 2008), which was also observed experimentally (Fig. 4.3B).

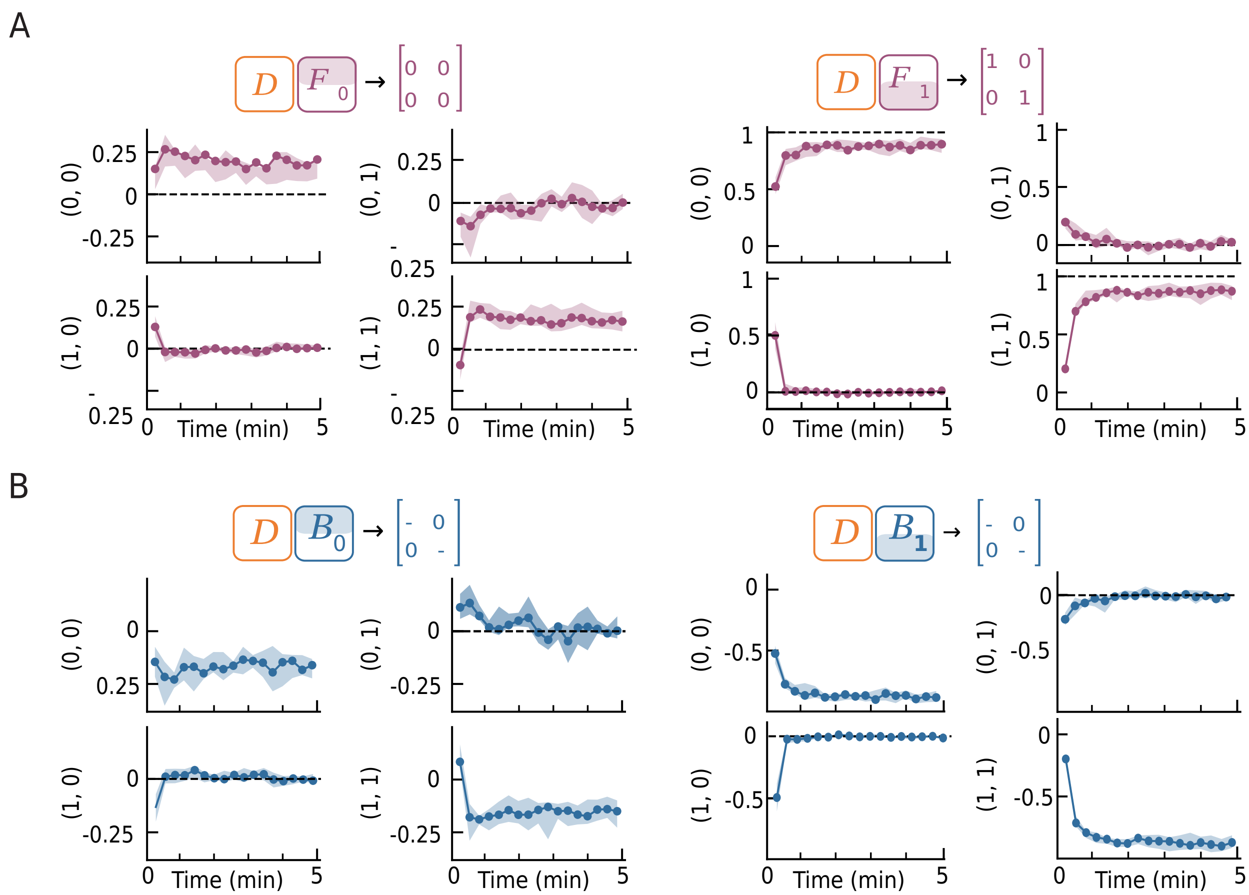

Figure 4.3: Encoder estimation reveals decoder-encoder interactions.

A. Product of the decoder matrix with feedforward contributions of the encoder matrix (N = 14, median with shading indicating interquartile range). Black dashed lines represent values that yield perfect trajectory tracking.

B. Product of the decoder matrix with feedback contributions of the encoder matrix (N = 14, median with shading indicating interquartile range). Black dashed lines represent values that yield closed-loop stability.

4.3.3 Encoder Estimation Model Quantifies Within-trial and Across-trial User Adaptation

We also used our encoder estimation model to revisit user learning-related changes in experiments. We observed that the change in user encoders between the start and end of

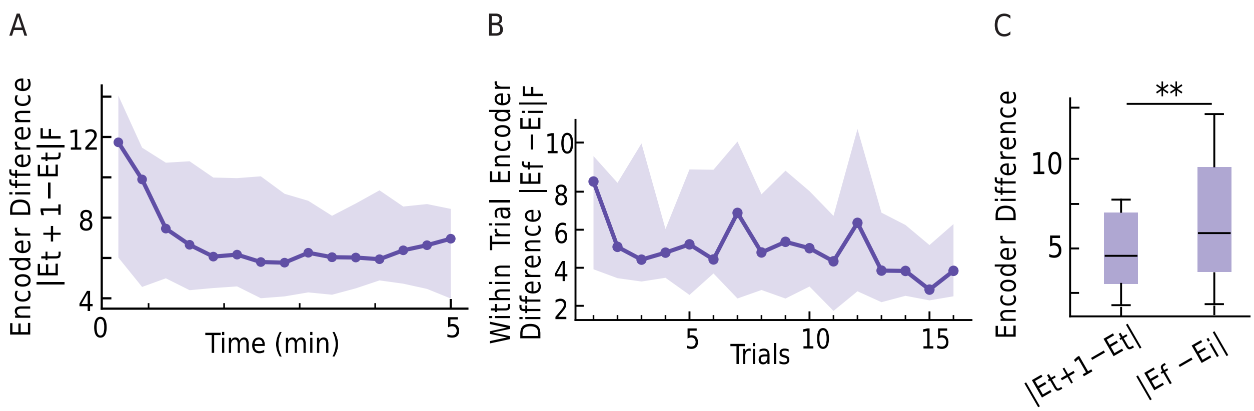

Figure 4.4: Encoder estimation shows user adaptation within trials and across trials.

A. Average change (norm difference) in consecutive encoders ($\|E_{t+1} - E_t\|$) over the course of a trial (N = 14, median with shading indicating interquartile range)

B. Average change (norm difference) in the initial versus final encoder within a trial ($\|E_f - E_i\|$) as a function of trial number (N = 14, median with shading indicating interquartile range).

C. Average change in encoders (norm difference) across a full trial ($\|E_f - E_i\|$) and consecutive encoders in a trial ($\|E_{t+1} - E_t\|$) (N = 14, median with interquartile range; Wilcoxon signed-rank test; **p < 0.001).

a trial decreased through the trial (Fig. 4.4A), suggesting encoders converge within a trial. Furthermore, the within-trial change in user encoders decreased across sequential trials (Fig. 4.4B). We found that the change in user encoders between the start and end of a trial was significantly larger than the change in sequential pairs of encoder estimates (Wilcoxon signed-rank test, $p < 0.001$) (Fig. 4.4C), consistent with a directed change over the course of an experiment (see Methods). Our models therefore revealed evidence of co-adaptation between the encoder and decoder that was consistent with our EMG-based analyses. By modeling the user, our control-theoretic encoder model allowed us to more precisely resolve user-decoder interactions within the closed-loop, co-adaptive interface.

4.4 Discussion

We created control theory-based estimations of user encoder transformations that allowed us to quantify critical closed-loop properties like trajectory-tracking and stability (Fig. 4.3). Prior work has shown that encoder-decoder stability likely contributes to interface performance and user learning (Orsborn et al., 2014; Taylor et al., 2002; Danziger et al., 2009), but efforts to quantify stability in adaptive neural interfaces have focused on the decoder alone (Dangi et al., 2013; Hsieh and Shanechi, 2018). By analyzing encoder-decoder pairs, we were able to characterize stability in the full co-adaptive system (Fig. 4.3) and use these methods as a way to measure learning.

4.4.1 Extending Encoder Estimation Methods to Other Modalities

Our encoder estimation technique was derived for closed-loop systems and tested with EMG data, but the methods are not limited to EMG data. These methods provide ways to represent encoder-decoder interactions within co-adaptive interfaces and to assess the stability of these interactions. Engineering co-adaptive interfaces that are stable is critical for usability and predictability regardless of the modality of the interface. Prior work (Liang and Kao, 2020; Merel et al., 2015; Golub et al., 2015) has shown the value of estimating users encoders from intracortical neural activity as these encoders could provide insights into internal model formation and closed-loop interactions. A potential extension of this work here is estimating encoders in intracortical neural interfaces through this control theory-based method and assessing encoder-decoder dynamics within co-adaptive neural interfaces.

4.4.2 Extending to Nonlinear Encoder Estimation Methods

The goal of this chapter was to model the user's transformation between task-space and output signals. The techniques proposed in this chapter leverage control theory and linear methods, which are highly interpretable. The inputs, outputs, and resulting behavior of the encoder model can be explained through commonly-used linear regression methods. However, as pointed earlier in the discussion, the encoder model does not generalize well and may not be able to synthesize user activity prior to experimentation. Deep learning and non-linear techniques have been growing in popularity for insights about neural mechanisms and dynamics (e.g., Pospisil et al. 2018; Wimalasena et al. 2022). Non-linear encoder techniques have been used to simulate closed-loop interaction prior to experimentation (Liang and Kao, 2020) and could add additional predictive power to these encoder models. Extending the linear encoder estimation techniques to non-linear ones is another potential extension to the work in this chapter.

Chapter 5

SHAPING USER-DECODER INTERACTIONS IN CO-ADAPTIVE NEURAL INTERFACES

5.1 Introduction

Neural interfaces can restore or augment human capabilities (Serruya et al., 2002; Taylor et al., 2002; Carmena et al., 2003; Hochberg et al., 2006; Pandarinath and Bensmaia, 2022; Dadarlat et al., 2023). In a neural interface, signals from the user are translated via a decoder algorithm to control a device. Decoder algorithms that adapt to users during interface operation can improve performance and enable personalization (Taylor et al., 2002; Danziger et al., 2009; DiGiovanna et al., 2009; Jarrassé et al., 2012; Orsborn et al., 2014; Shenoy and Carmena, 2014; Hahne et al., 2017; Brandman et al., 2018; Silversmith et al., 2020; Rizzoglio et al., 2021; Gigli et al., 2022; Hu et al., 2023). Users also adapt as they learn to control the interface because they receive real-time feedback (Carmena et al., 2003; Ganguly and Carmena, 2009; Fetz, 2007; Jackson and Fetz, 2011; Albert and Shadmehr, 2016). Introducing adaptive algorithms into neural interfaces therefore creates a co-adaptive system. Prior work has demonstrated the benefits of leveraging co-adaptation to guide or assist user adaptation (Orsborn et al., 2014; Oby et al., 2019; De Santis and Mussa-Ivaldi, 2020) and maintain performance over time (Orsborn et al., 2014; Benabid et al., 2019; Silversmith et al., 2020).

However, co-adaptive systems are challenging to design because they involve dynamic, two-learner interactions: both the user and decoder may adapt simultaneously and in response to one another. Experiments have exposed the importance of identifying appropriate decoder parameters, such as the rate of decoder adaptation, to ensure stable and predictable interface control (Danziger et al., 2009; Müller et al., 2017; Madduri et al., 2022). Methods to design and analyze two-learner systems are an active area of research in neural engineering (Dangi et al., 2013; Hsieh and Shanechi, 2018) and dynamic game theory (Chasnov et al., 2020), but co-adaptive neural interfaces are largely designed and implemented empirically. Theoretically-grounded and experimentally-validated techniques to model and quantify co-adaptive dynamics will unlock new ways to design robust interfaces that harness the full potential of user and decoder adaptation.

5.1.1 Related Work

Existing frameworks for designing co-adaptive interfaces highlight the promise of model-based approaches but either do not capture the full range of dynamic interactions that can arise between users and decoders, or do not make predictions about interaction outcomes. Past work on neural interfaces focused on the user adapting to a leading, optimal (fixed) decoder (Merel et al., 2015) or the decoder adapting to follow the user (De Santis, 2021), and most experimental neural interfaces are constrained to this limited range of co-adaptive interactions (Taylor et al., 2002; Orsborn et al., 2014; Oby et al., 2019; Silversmith et al., 2020). We instead seek methods that will be able to accommodate a wide spectrum of user-decoder interactions, where the device may shift from 'leading' to 'following' the user over time (Madduri et al., 2023). Game theory has been used to analyze the joint responses of two learners in human-human motor interactions (Braun et al., 2009; Jarrassé et al., 2012) and human-machine interactions (Müller et al., 2017; Li et al., 2019). We built on these ideas, adapting tools from control theory and game theory to create a flexible, experimentally-validated framework for analysis and synthesis of co-adaptive outcomes of continuous interactions in neural interfaces.

This chapter builds on the models and methods discusses in prior chapters. To build experimentally-validated frameworks for co-adaptive neural interfaces, we developed a myoelectric interface experimental platform (see Ch. 3). We modeled this closed-loop system using control theory methods to quantify interactions between users and the device (see Ch. 4). This experimental platform and analysis toolkit allowed us to test analytical predictions about user-machine behaviors derived from a model treating the user and decoder as two agents in a game.

5.1.2 *Contributions of This Chapter*

In this chapter, we show how the experimental results revealed that user learning can be influenced through real-time interface adaptation and that these outcomes can be predicted using game-theoretic models.

This chapter demonstrates that:

1. users adapted their behavior alongside an adaptive decoder to control a cursor via their muscle activity, creating a co-adaptive system.
2. our game-theoretic model provided predictions about how decoder learning rates, penalty terms, and initializations impact system performance and user-decoder interactions that were also seen in experimentation;
3. decoder learning rates disrupted user-decoder interactions and performance;
4. decoder penalty terms impacted both decoder and encoder "effort";
5. decoder penalty also revealed individual preferences for cursor speed vs. effort;
6. decoder initialization did not affect system performance but might have biased encoders learned by users.

Together, our findings establish new methods to analyze and synthesize co-adaptive systems that can enable next-generation approaches for designing and personalizing neural interfaces.

5.2 Methods

5.2.1 Game Theory Methods

We extended our game theory framework presented in Ch. 2 to include our encoder estimation from Ch. 3. Ch. 2 considers the encoder as an affine function with only one dependent variable as an input. The game theory formulation here considers the encoder as a function of multiple dependent variables. Importantly, the predictions and intuitions about the user-decoder interactions remain the same.

Game formulation

We modeled the co-adaptive system in a game theory framework involving two agents: the user and decoder. The user and decoder decision variables were the entries in the matrices E and D, respectively. Each agent was modeled as having a cost function defined by a linear combination of *task error* and *agent effort*. Agent effort was quantified using the square of the Frobenius norms of the matrices E and D. Task error was defined in cursor velocity space as the squared 2-norm of $e = D \cdot u - (\dot{\tau} - \dot{y}) = D \cdot E \cdot z - (\dot{\tau} - \dot{y})$ where

$$z = \begin{bmatrix} \tau \\ \dot{\tau} \\ \tau - y \\ \dot{\tau} - \dot{y} \end{bmatrix}. \tag{5.1}$$

As in Ch. 2, we modeled agent adaptation using gradient descent on their individual cost function and assessed convergence of gradient descent to stationary points by linearizing discrete-time gradient descent dynamics and computing eigenvalues of the linearization.

Solving for stationary points

To help make analysis tractable, we focused on a 1-dimensional version of the task (simplifying the analysis to mirror methods in Ch. 2 and in Sec. 7.1) and assumed the effort penalty was identical for the encoder and decoder (i.e. $E = D$):

$$c_E(E, D) = \|e(E, D)\|_2^2 + \lambda\|E\|_F^2,\ c_D(E, D) = \|e(E, D)\|_2^2 + \lambda\|D\|_F^2, \tag{5.2}$$

where $D \in \mathbb{R}$ and $E = [F_0, F_1, B_0, B_1] \in \mathbb{R}^{1\times 4}$.

With the full terms of the encoder E in the cost function, the error term of the cost becomes

$$e(E, D) = D \cdot F_0 \cdot \tau + D \cdot F_1 \cdot \dot{\tau} + D \cdot B_0 \cdot (\tau - y) + D \cdot B_1 \cdot (\dot{\tau} - \dot{y}), \tag{5.3}$$

so there are four encoder decision variables that aim to minimize the position error (the four components of E). The decoder and encoder are playing a game in each one of the encoder decision variables.

To further simplify, we analyzed the effect of varying only one variable in E while holding the others constant. For each choice of $X \in \{F_0, F_1, B_0, B_1\}$, the error term has the form $e(X, D) = D \cdot X \cdot a + b$ for some scalars a and b. For example, in the $X = B_1$ and $a = \dot{\tau} - \dot{y}$ case,

$$\|e(E, D)\|_2^2 = \|D \cdot B_1(\dot{\tau} - \dot{y}) - (1/\Delta t)(\tau - y) + k\|_2^2. \tag{5.4}$$

For simplicity, we can set $b = (1/\Delta t)(\tau - y) + k$.

We solved the nonlinear equations $\partial_X c_E(X^*, D^*) = 0$ and $\partial_D c_D(X^*, D^*) = 0$ to compute stationary points (X^*, D^*). In the scalar case, there are three real stationary points of this system:

$$D^* = E^* = \sqrt{\frac{ab - \lambda}{a^2}}; D^* = E^*_X = -\sqrt{\frac{ab - \lambda}{a^2}}; D^* = E^* = 0 \tag{5.5}$$

Again, in the $X = B_1$ case:

$$\begin{gathered} D^* = B_1^* = \sqrt{\frac{(\dot{\tau} - \dot{y})(\frac{1}{\Delta t}(\tau - y) + k) - \lambda}{(\dot{\tau} - \dot{y})^2}}; \\ D^* = B_1^* = -\sqrt{\frac{(\dot{\tau} - \dot{y})(\frac{1}{\Delta t}(\tau - y) + k) - \lambda}{(\dot{\tau} - \dot{y})^2}}; \\ D^* = B_1^* = 0. \end{gathered} \tag{5.6}$$

However, the stationary point at $(D^{*}, E^{*}) = (0, 0)$ assumes that the decoder's transformation is 0, which is not experimentally realistic. The stationary points also shift as lambda changes, highlighting the dependence of the stationary points on the penalty terms. As only the decoder penalty term can be specified, the decoder penalty can shift the stationary points and, by extension, the encoder.

Stability at Stationary Points

Stability at the stationary point can be assessed by linearizing about the stationary point and evaluating the eigenvalues of the Jacobian.

When $\lambda_E = \lambda_D = 0.5$ and $a_X = b_X = 1$ for simplicity, the eigenvalues are simplified to $\{1 - 2\alpha_E, 1 - 2\alpha_D\}$. The encoder-decoder dynamics converge if all the eigenvalues of the discrete-time Jacobian are within the unit circle. This occurs when $0 < \alpha_E < 1$ and $0 < \alpha_D < 1$, constraining the learning rates of the encoder and decoder adaptation. Learning rates that are too fast or too slow may limit the rate of convergence to stationarity. When $\lambda_E = \lambda_D = 0$, the eigenvalues are simplified to $\{1, 1 - 2\alpha_E a_X b_X - 2\alpha_D a_X b_X\}$, which are unstable, meaning that λ_E and λ_D are required for stable stationary points.

5.2.2 Data Analysis Methods

EMG Analysis

To quantify the relationship between user EMG activity and cursor movement, we computed EMG direction tuning curves, a well-established analysis used for neural and myoelectric interfaces (Georgopoulos et al., 1986; Radhakrishnan et al., 2008; Couraud et al., 2018). We first down-sampled and grouped intended user cursor velocities into ten equally-spaced directions and then fit raw EMG activity to the binned directions to form EMG tuning curves. These EMG tuning curves were created for each participant, each trial, and each EMG signal (Fig. 5.2A, B). Note, since the EMG channels are differentially-recorded, we selected 63 EMG signals for the EMG tuning analysis.

To quantify the within-trial EMG changes from early to late trial segments (Fig. 5.2C) we computed the norm difference between the early and late EMG tuning curves:

$$\|\Delta EMG\|_2 = \|EMG_{dir,i} - EMG_{dir,j}\|_2 \tag{5.7}$$

where EMG_{dir} is the average EMG activity per direction and i and j indicate time points. These early and late tuning curves were calculated by reconstructing 30-second segments of trial data that equally sampled all cursor directions. To measure how the EMG activity changed across the entire experiment per participant (Fig. 5.1A), we averaged the $\|\Delta EMG\|_2$ across all channels for each participant to compute a mean $\|\Delta EMG\|_2$ per participant per trial.

To find the preferred direction of the EMG activity, we performed cosine-fitting analysis (Georgopoulos et al., 1986). Specifically, average EMG activity for each intended cursor direction was modeled as:

$$EMG = B_1 cos\theta + B_2 sin\theta + B_3 \tag{5.8}$$

where θ represents intended cursor direction and B_1, B_2 and B_3 are model coefficients. The preferred direction (PD) is:

$$PD = arctan2(B_2/B_1). \tag{5.9}$$

The change between early and late preferred direction (Fig. 5.1D) was calculated as $|\Delta PD| = |PD_{late} - PD_{early}|$.

Statistical Analyses

All analyses treated participants as individual data points and computed the mean across decoder conditions (learning rate, penalty terms, initialization) and trials. To assess statistical significance across time and conditions for each participant, we used a one- or two-sided Wilcoxon signed-rank test (`scipy.stats.wilcoxon`), which is a paired, non-parametric test.

We chose a non-parametric statistical test because of subject-to-subject variability. Figure box-plots were plotted with `Matplotlib.pyplot.boxplot` (center line, median; box limits, upper and lower quartiles; whiskers, 1.5x interquartile range; fliers not plotted).

5.3 Results

5.3.1 An Adaptive Myoelectric Interface Created a Co-adaptive Experimental Platform

These results use data from the myoelectric interface experiment introduced in Ch. 3. To determine whether the myoelectric interface was co-adaptive, we examined whether users adapted during each trial. We asked whether the user's behavior changed while controlling the interface by estimating the relationship between user sEMG activity and the direction of intended movement – i.e., an EMG tuning curve (Georgopoulos et al., 1986; Radhakrishnan et al., 2008; Couraud et al., 2018) (see 5.2.2). EMG tuning curves from a sample participant revealed within-trial changes in user behavior (Fig. 5.1A, B). We saw similar changes in other participants (Fig. 5.2). We quantified within-trial changes for each participant by calculating the magnitude of the difference between EMG tuning curves (Fig. 5.1C) and angular differences in the preferred direction (Fig. 5.1D) for the first versus last 30 seconds in the trial across all electrodes (see 5.2.2). All participants showed significant changes in the magnitude and tuning angles of EMG activity over the course of the trial, suggesting that users adapted their behavior during the 5-minute trial (Wilcoxon signed-rank test with a Bonferroni correction of $n = 14, p < 0.001$).

Interestingly, we observed that within-trial differences in EMG tuning curves decreased as trials progressed (Fig. 5.1E), demonstrating that users also adapted across the duration of the experimental session. We hypothesized that these longer timescale changes reflected users acquiring useful strategies for the myoelectric interface during the experiment. This suggests that performance may improve over time, regardless of the decoder. We tested this prediction by comparing performance in the first and second blocks, which contained trials with matched decoder adaptation conditions. We indeed saw that participants' average mid-trial and final task error was lower in block 2 compared to block 1 (Wilcoxon signed-rank test, $p < 0.05$), suggesting that task performance trajectories improved more and more quickly in block 2 than in block 1 (Fig. 5.1E,F). Taken together, these results suggest that users adapt alongside the decoder in our myoelectric interface, both within and across trials.

5.3.2 Decoder Learning Rates Can Disrupt Co-adaptation

Our game-theoretic model emphasizes the importance of both the user and decoder learning rates for convergence to stationary points. One agent adapting faster than the other may limit or prevent convergence to stationarity (Fig 5.3). Our model predictions echo results from previous models (Müller et al., 2017) and experimental findings from both invasive neural and non-invasive body-machine interfaces (Danziger et al., 2009; Orsborn et al., 2011). We validated our model's predictions by testing the effect of decoder learning rates in our co-adaptive myoelectric interface system. Because the user's learning rate was unknown and could not be estimated *a priori*, we varied the decoder learning rate, testing two extremes: slow and fast (see Ch. 3 Methods). As explored in Ch. 3, decoders for both learning rate conditions were initialized randomly, matching all other conditions, and then adapted according to the specified cost function. Task performance was worse for co-adaptive interfaces where the decoder adapted with a fast learning rate compared to the slow learning rate throughout all trials (Fig. 5.4A), and performance improved more within the trial in the slow condition compared to the fast condition (Wilcoxon signed-rank test, $p < 0.001$) (Fig. 5.4B).

We found that the decoder learning rate also affected the user's encoders. We used our encoder estimates from Ch. 4 to quantify how much users' encoders changed over the course of the 5-minute trial in each decoder condition. User encoders changed less within trials when the decoder adapted fast compared to slow (Wilcoxon signed-rank test, $p < 0.001$) (Fig. 5.4C), suggesting that the decoder may have adapted too fast for the user's encoder

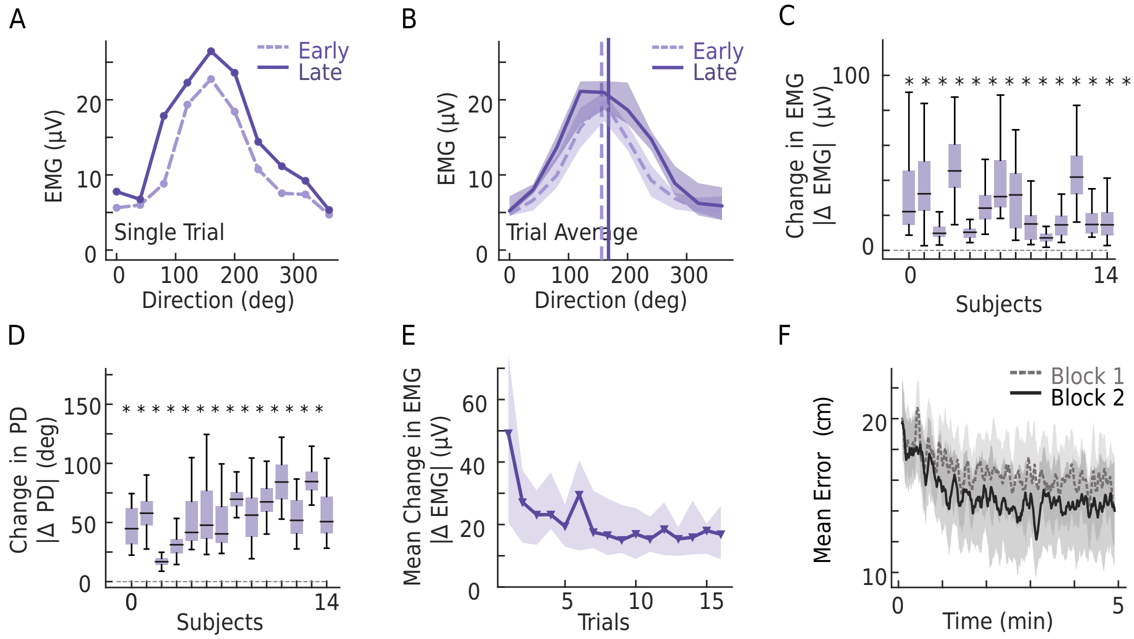

Figure 5.1: Co-adaptation in myoelectric interfaces.

A. EMG tuning curve for one channel, early (dashed) and late (solid) within a trial.

B. Average EMG tuning curve for one channel early and late across all trials (N = 16). Preferred directions for each curve shown with vertical lines.

C. Norm difference of the average early versus late EMG tuning curves ($\|\Delta EMG\| = \|EMG(late) - EMG(early)\|$) compared to a difference of zero (dashed gray) for all channels, computed for each subject (N = 63, Wilcoxon signed-rank test with Bonferroni correction of n = 14; *p < 0.001).

D. Magnitude of angular change in EMG PD early versus late ($\|\Delta PD\| = \|PD(late) - PD(early)\|$) compared to difference of zero (dashed gray) for all channels, computed for each subject (N = 63, Wilcoxon signed-rank test with a Bonferroni correction; *p < 0.001).

E. Average change in EMG tuning curves within a trial ($\|\Delta EMG\|$ as C.) for all subjects as a function of trial number (N = 14). Average change in EMG of first trial vs. last trial (Wilcoxon signed-rank test, *p < 0.05).

F. Average error as a function of trial time in block 1 (dashed gray) and block 2 (solid black) (N = 14). Black boxes represent comparisons of mid-trial and end-of-trial task error between block 1 and block 2 (Wilcoxon signed-rank test, **p < 0.001, *p < 0.05)

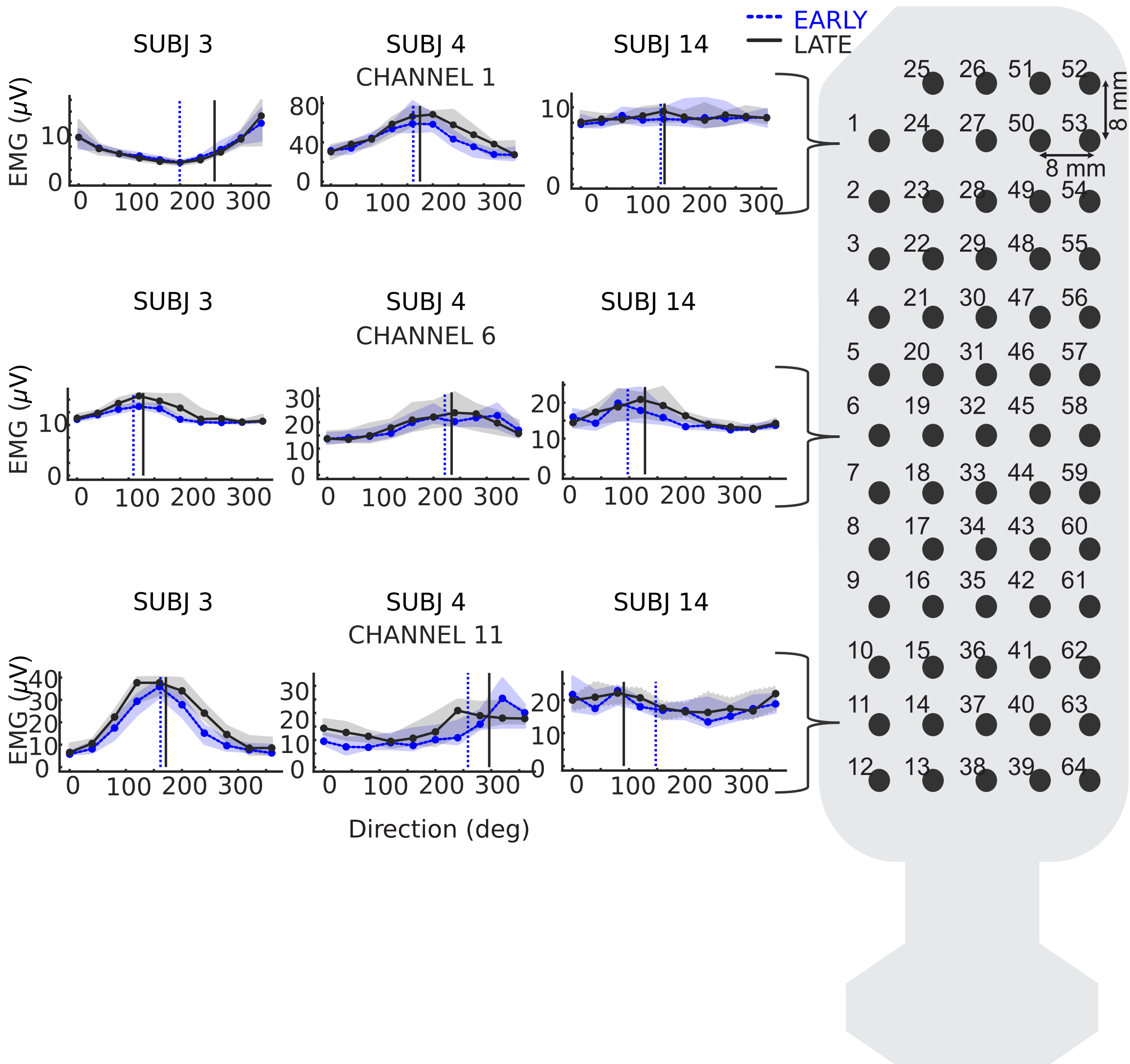

Figure 5.2: Surface EMG electrode and tuning curves.

Diagram of surface EMG electrode and sample EMG tuning curves from representative subjects taken from early (dashed blue) and late (solid black) segments within trial. Vertical lines represent preferred direction from early (dashed blue) and late (solid black) segments within trial.

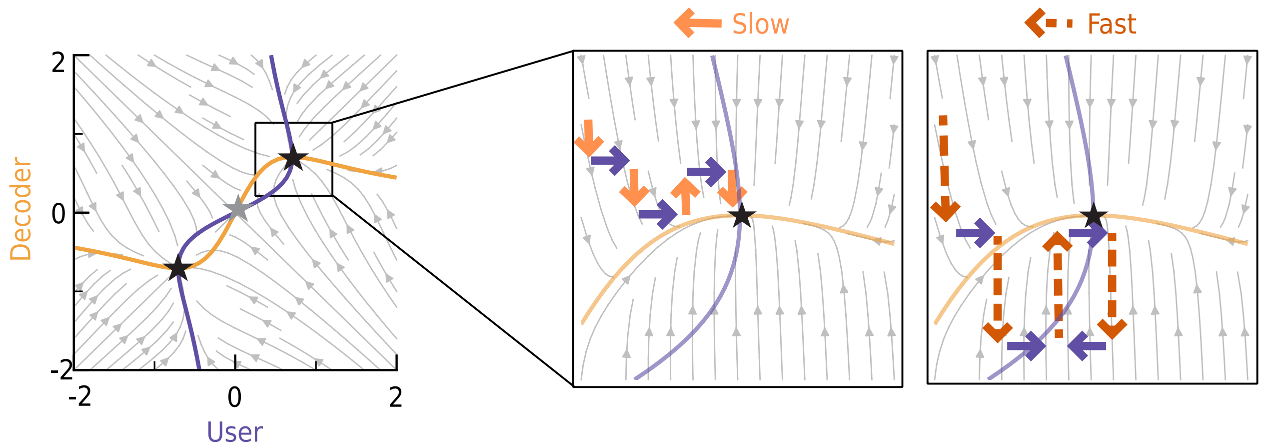

Figure 5.3: Game theory model predicts learning rate affects convergence to equilibria. Model predictions. All panels show the gradient field of the user and decoder cost functions. Purple (user) and orange (decoder) curves show nullclines (where the agent's cost gradient equals 0) that intersect at stationary points (black stars). Slow (middle) and fast (right) panels zoom in on one stationary point and illustrate changes in the decoder (shades of orange) and encoder (purple) for different decoder learning rates.

to match. We then analyzed user-decoder interactions, which confirmed that rapid decoder adaptation disrupted co-adaptation. The decoder-encoder pairs in the slow condition approached the system ideal for perfect trajectory tracking and closed-loop stability, but the fast condition decoder-encoder pairs deviated from the system ideal state in both the first-order feedforward (Fig. 5.4D) and feedback conditions(Fig. 5.4E), showing that the fast learning rate negatively impacted the co-adaptive interactions for both trajectory tracking and closed-loop stability. Our experimental results corroborated our game-theoretic prediction that decoder learning rates will influence system behavior.

5.3.3 Decoder "Effort" Penalty Influenced User "Effort" Without Changing Performance

Our game-theoretic framework includes the regularization terms, $\|D\|_F$ and $\|E\|_F$, which correspond to what we refer to as "effort". Including regularization terms enforces con-

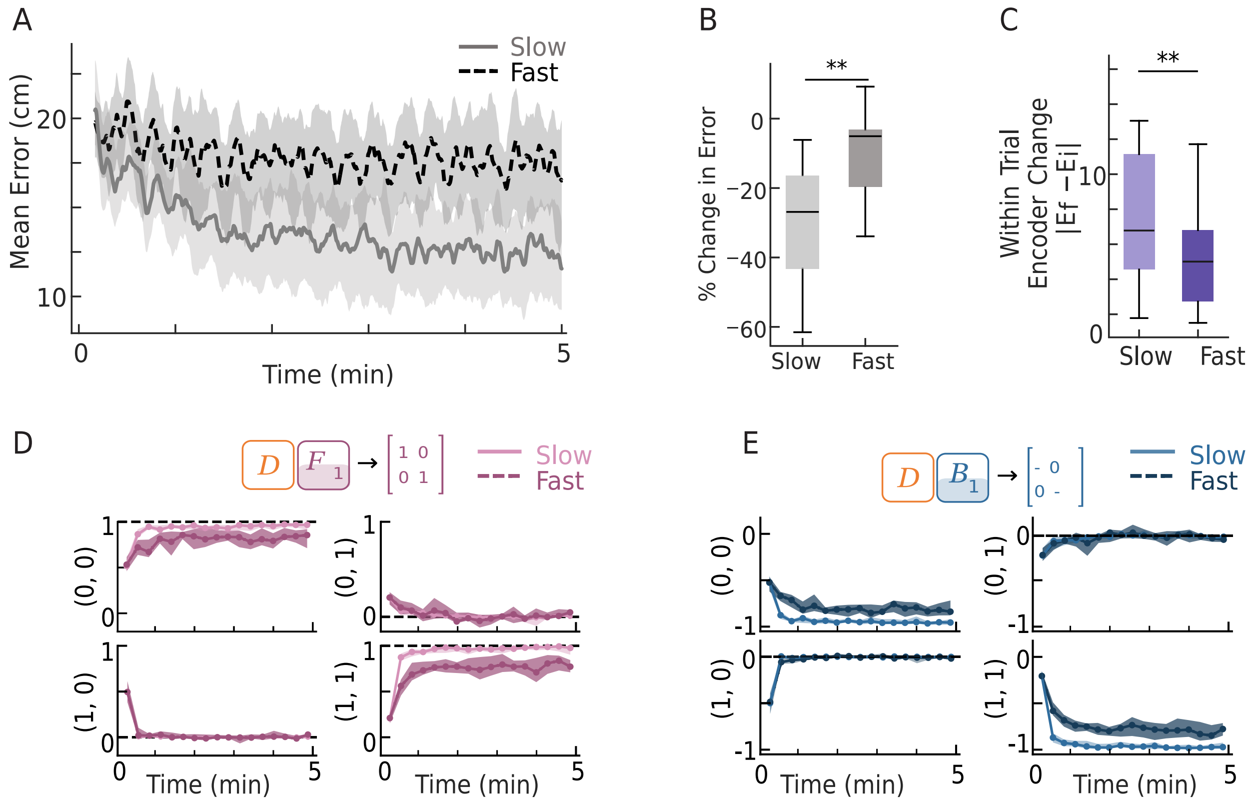

Figure 5.4: Decoder learning rate influences interface performance and user learning.

A. Average error as a function of time within the 5-minute trial for fast (dashed dark orange) and slow (solid light orange) decoder learning rates (N = 14, median with shading indicating the interquartile range).

B. Percent change in error from the start of the trial (first 30 seconds) to the end of the trial (last 30 seconds), separated by fast (dashed black) and slow (solid gray) learning rates (**p < 0.001).

C. Average change in encoder within trials ($\|E_f - E_i\|$) for the slow and fast learning rates (**p < 0.001).

D. Product of the decoder matrix with first-order feedforward (F_1) contributions of the encoder matrix for fast (dashed dark pink) and slow (solid light pink) decoder learning rates. Black dashed lines represent values that yield perfect trajectory tracking.

E. Product of the decoder matrix with first-order feedback (B_1) contributions of the encoder matrix for fast (dashed dark blue) and slow (solid light blue) decoder learning rates. Black dashed lines represent values that yield closed-loop stability.

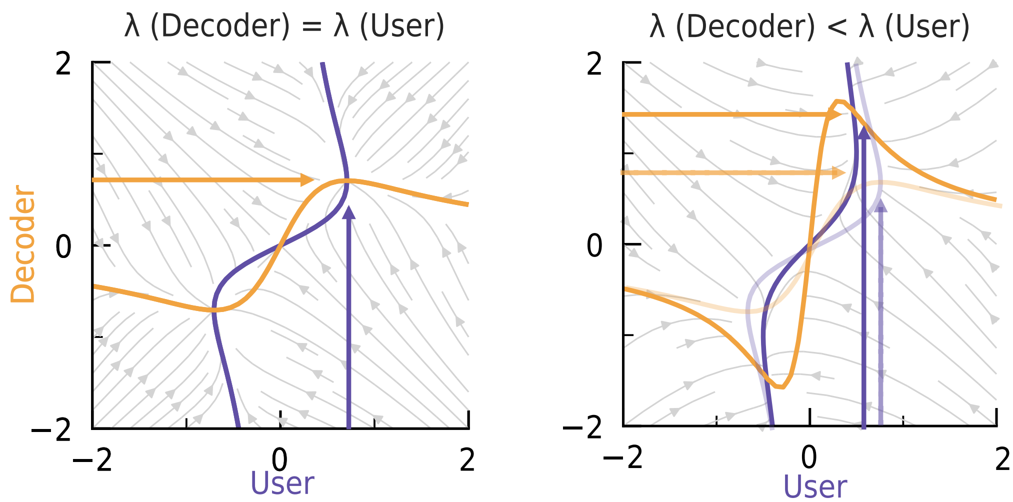

Figure 5.5: Game theory model predicts penalty parameters affect equilibria.
Model predictions. Both panels show gradient fields of the user and decoder cost functions when the decoder penalty term is equal to that of the user (left) or less than that of the user (right). Arrows depict expected stationary points of user (purple) and decoder (orange).

vergence to a stable stationary (See Sec. 5.2.1). In the model, the scaling of the decoder regularization term, controlled by λ_D, influences the decoder effort. In turn, the user effort is also influenced by λ_D due to the linear relationship between the decoder and cursor velocity. As the decoder penalty changes in our model, the stationary point and encoder effort correspondingly shift (Fig. 5.5). Our model therefore predicts that changing decoder effort will not impact convergence to stationarity in the co-adaptive interface, but will influence user behavior by shifting their effort.

We tested the impact of two penalty terms – low penalty and high penalty – on decoder-encoder co-adaptation in our myoelectric interface experiments. Decoders for both penalty conditions were initialized randomly, matching all other conditions, and then adapted according to the specified cost function (see Methods). Performance was not impacted by the penalty terms (Wilcoxon signed-rank test, $p > 0.05$) (Fig. 5.6A), suggesting the co-adaptive systems converged similarly for both decoder conditions. Decoder effort was influenced by

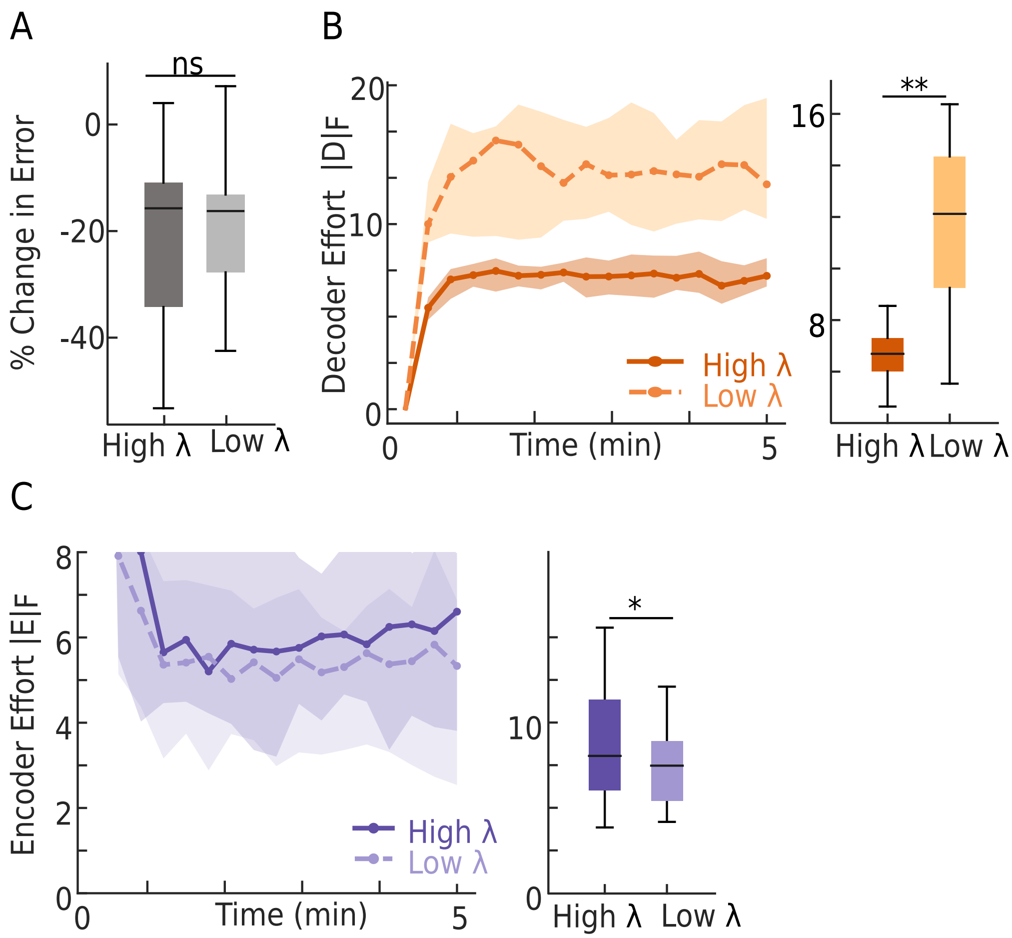

Figure 5.6: Decoder penalty affects user effort but not performance.

A. Percent change in error from the start of the trial (first 30s) to the end of the trial (last 30s), separated by high (dark gray) and low (light gray) decoder penalty term conditions (N=14, median with shading indicates interquartile range; Wilcoxon signed-rank test, ns > 0.05).

B. Left: Average magnitude of the decoder matrix (norm) as a function of time in the trial for low (dashed light orange) and high (solid dark orange) decoder penalty terms (N = 14). Right: Box plots are average decoder effort across the trial for each subject (**p < 0.001).

C. Left: Average magnitude of the user encoder matrix (norm) as a function of time in the trial for low (dashed light purple) and high (solid dark purple) decoder penalty terms. Right: Box plots are average effort for each subject across the trial (*p < 0.05).

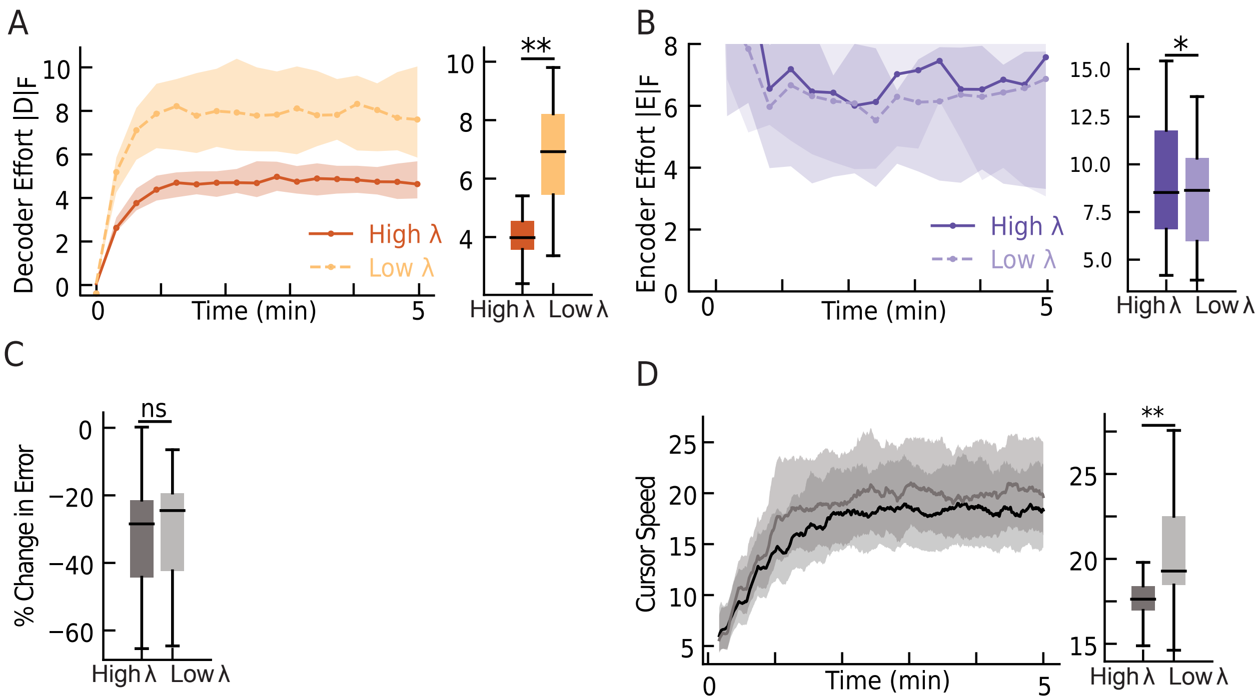

Figure 5.7: Effect of penalty parameter in slow learning rate condition only.

A. Left: Average magnitude of the decoder matrix (norm) as a function of time in the trial for low (dashed light orange) and high (solid dark orange) decoder penalty terms (N = 14, median with shaded area indicating the interquartile range). Right: Boxplots are average decoder effort across the trial for each subject (N=14, median with shading indicates interquartile range; Wilcoxon signed-rank test, **p < 0.001).

B. Left: Average magnitude of the user encoder matrix (norm) as a function of time in the trial for low (dashed light purple) and high (solid dark purple) decoder penalty terms. Right: Boxplots are average effort for each subject across the trial (*p < 0.05).

C. Percent change in error from the start of the trial (first 30 seconds) to the end of the trial (last 30 seconds), separated by high (dark gray) and low (light gray) decoder penalty term conditions (ns > 0.05).

D. Left: Average cursor speed as a function of time in the trial for low (light gray) and high (dark gray) decoder penalty terms. Right: Boxplots are average cursor speed across the trial for each subject (**p < 0.001).

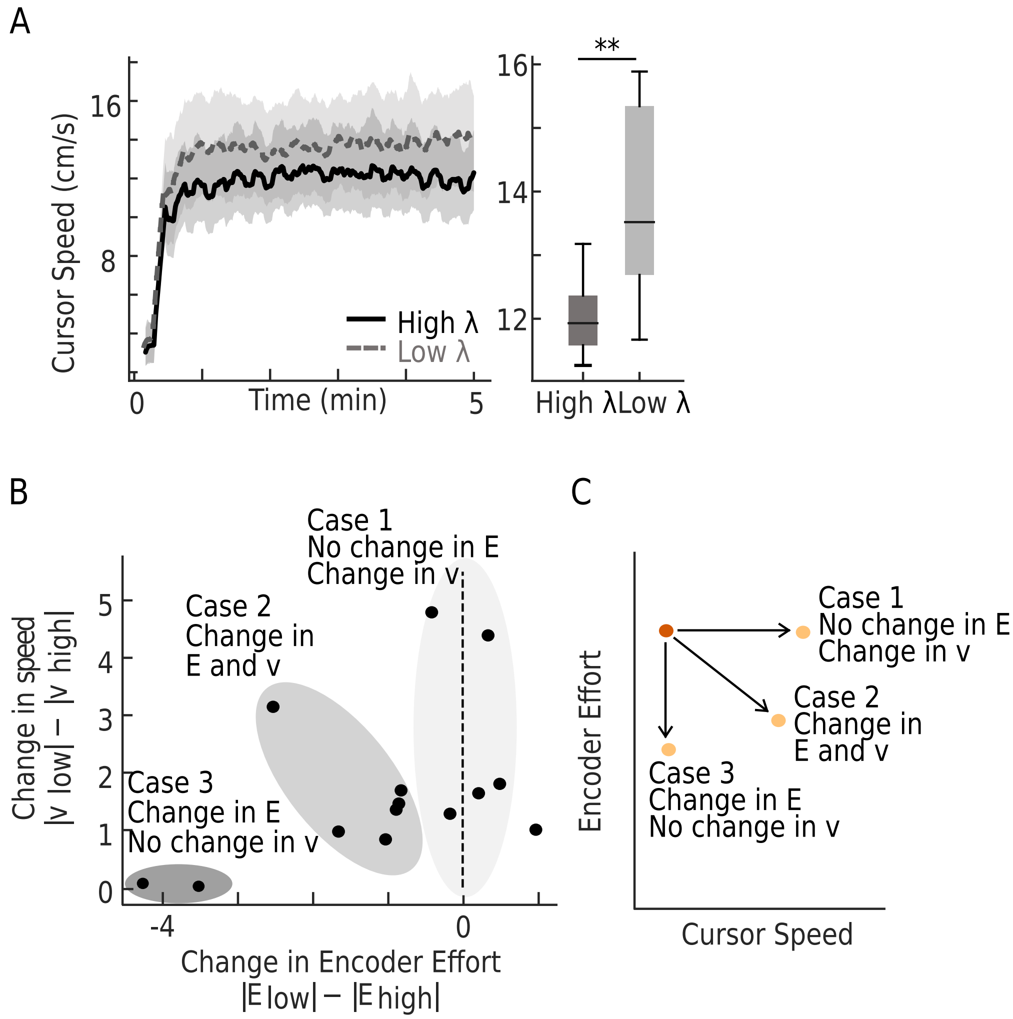

Figure 5.8: Individual choices seen in cursor speeds.

A. Left: Average cursor speed as a function of time in the trial for low (dashed light gray) and high (solid dark gray) decoder penalty terms (N = 14, median with shaded area indicating the interquartile range). Right: Box-plots are average cursor speed across the trial for each subject (N=14, median with shading indicates interquartile range; Wilcoxon signed-rank test, **p < 0.001).

B. Average difference in cursor speed between the low and high decoder penalty conditions plotted against the average difference in encoder efforts between the low and high decoder penalty conditions for each individual participant. Shading to illustrate qualitatively identified groupings.

C. Schematic illustrating the relationship between encoder effort and cursor speed when decoder effort changes.

the penalty term as expected: higher decoder penalty resulted in decreased decoder effort (Wilcoxon signed-rank test, $p < 0.001$) (Fig. 5.6B). As predicted by our model, the decoder effort affected the encoder effort to reflect the trade-off in effort between the two learners. Thus, higher decoder penalty terms resulted in increases in the encoder effort (Wilcoxon signed-rank test, $p < 0.05$)(Fig. 5.6B). These results held regardless of the learning rate of the decoders (Fig. 5.7).

While our experimental results matched model predictions, we noticed that an increase in decoder magnitude did not correspond to a similar degree of decrease in magnitude for the encoder (Fig. 5.6B vs. 5.6C). Importantly, decoder "effort" controls the magnitude of D, which defines the gain between user EMG and cursor velocity. If users do not change their effort to perfectly mirror changes in the decoder effort, the cursor's speed—an aspect of task performance not captured in our model—would change. We found that decoder penalty terms influenced the cursor speed during tracking. The lower decoder penalty term led to faster cursor speeds on average across users (Wilcoxon signed-rank test, $p < 0.001$)(Fig. 5.8A). Interestingly, these trade-offs differed across our participants (Fig. 5.8B), potentially capturing individual preferences. Our data suggest users fell into approximately three different groups: some users chose to decrease their effort and maintain a similar cursor speed, others elected to maintain a similar effort to go faster, and others changed both effort and speed to varying degrees (Fig. 5.8C). Our experimental results confirmed our game-theoretic model predictions and highlighted the potential of decoder adaptation to influence the user's encoder during closed-loop co-adaptation.

5.4 Discussion

Our game-theoretic model predicted the importance of decoder learning rates in co-adaptive systems. Experimentally, we found that a faster decoder learning rate yielded worse overall performance (Fig. 5.4A,B) and altered trajectory-tracking and stability in encoder-decoder pairs (Fig. 5.4D,E) in our myoelectric interface. Our results add to prior experimental observations from co-adaptive kinematic (Danziger et al., 2009) and brain-computer interfaces (Orsborn et al., 2011; Müller et al., 2017). Interestingly, many high-performance adaptive algorithms in invasive neural interfaces adapt rapidly (Shanechi et al., 2016; Brandman et al., 2018; Hsieh and Shanechi, 2018), whereas the fast decoder learning rate condition disrupted co-adaptive dynamics in our interface. Our model highlights that the learning rates of both the user and decoder determine system dynamics (See Sec. 5.2.1). Possibly, our naive participants may have adapted at different rates than the more experienced participants in invasive brain-computer interface studies. User learning rates may also vary across different bio-signal modalities (Jackson and Fetz, 2011). Future co-adaptive interfaces may, then, benefit from personalizing decoder learning rates based on measures of user learning rates. Methods to estimate properties of user learning, such as learning rate, in real-time will open new ways to more fully harness co-adaptation.

Our game theory model and experimental results also showed that decoder "effort" can influence user behavior (Fig. 5.5, Fig. 5.6B,C). In myoelectric interfaces, this manipulation may be helpful to encourage users to learn encoders that use less muscle activity to conserve energy or to increase muscle activation for motor rehabilitation. Interestingly, we saw that users responded differently to varying decoder penalty terms, potentially reflecting personal preferences for effort versus cursor speed (Fig. 5.8F). Customizing interfaces based on dynamic user preferences is beneficial in human-machine interfaces like exoskeletons (Ingraham et al., 2022). Individual user preferences for cursor dynamics have also been seen in invasive neural interfaces (Willett et al., 2016). Extending our game-theoretic frameworks to include user preferences will enable design of fully personalized, custom neural interfaces, which may make these technologies more widely accessible (Yamagami et al., 2023).

Our game theory model highlighted that multiple decoder-encoder stationary points exist in neural interfaces. This raises the possibility that the initial decoder can affect the final stationary point and, in turn, the final user encoder, without significantly impacting performance (Fig. 5.9A). Our experimental results confirmed that performance was not impacted by decoder initialization (Fig. 5.9D), consistent with prior work testing random decoder initializations in neural interfaces (Orsborn et al., 2012). While performance was not different, we found that initialization did subtly impact the stationary points as identified by the final decoder-encoder matrices (Fig. 5.10). Our findings suggested that decoder initialization might bias the encoders learned by users, which could be particularly beneficial for rehabilitation applications, which aim to shape users' behavior towards a particular goal. Our results also suggest that it may be critical to carefully consider initial training protocols for any neural interface, as it may influence user learning trajectories.

Our experiments revealed occasional deviations from model predictions, including deviations from strict encoder-decoder trade-offs (Fig. 5.6) and only subtle impacts of the decoder initialization (Fig. 5.10). This finding suggests that the actual user's cost might differ from our model's assumptions. For instance, our model predicts that decoder initialization would lead users to completely invert the relationship between EMG/movement and cursor velocity (Fig. 5.9A), which ignores the likely possibility that users have some bias in strategies to control the interface. Additionally, there may be multiple ways to represent "effort", and these definitions may become unclear in less embodied invasive neural interfaces. Prior work has considered that users minimize effort (Flash and Hogan, 1985; Todorov and Jordan, 2002) via muscle activation (Emken et al., 2007), metabolic cost (Finley et al., 2013; McDonald et al., 2022), and motor variability (Harris and Wolpert, 1998). Studies into how users learn to control neural interfaces suggest that other features, such as correlations between neurons, may be key factors (Sadtler et al., 2014). Deeper insights into how users learn to control different kinds of interfaces will improve our ability to model their cost functions. Alternatively, user costs may not need to be explicitly modeled. Recent work demonstrates an interface that adapts using measurements of the human actions to steer the co-adaptive system to reach a variety of different game theory equilibria without the need to estimate or measure the human's cost (Chasnov et al., 2023). Extending such data-driven approaches to neural interfaces is a promising future direction.

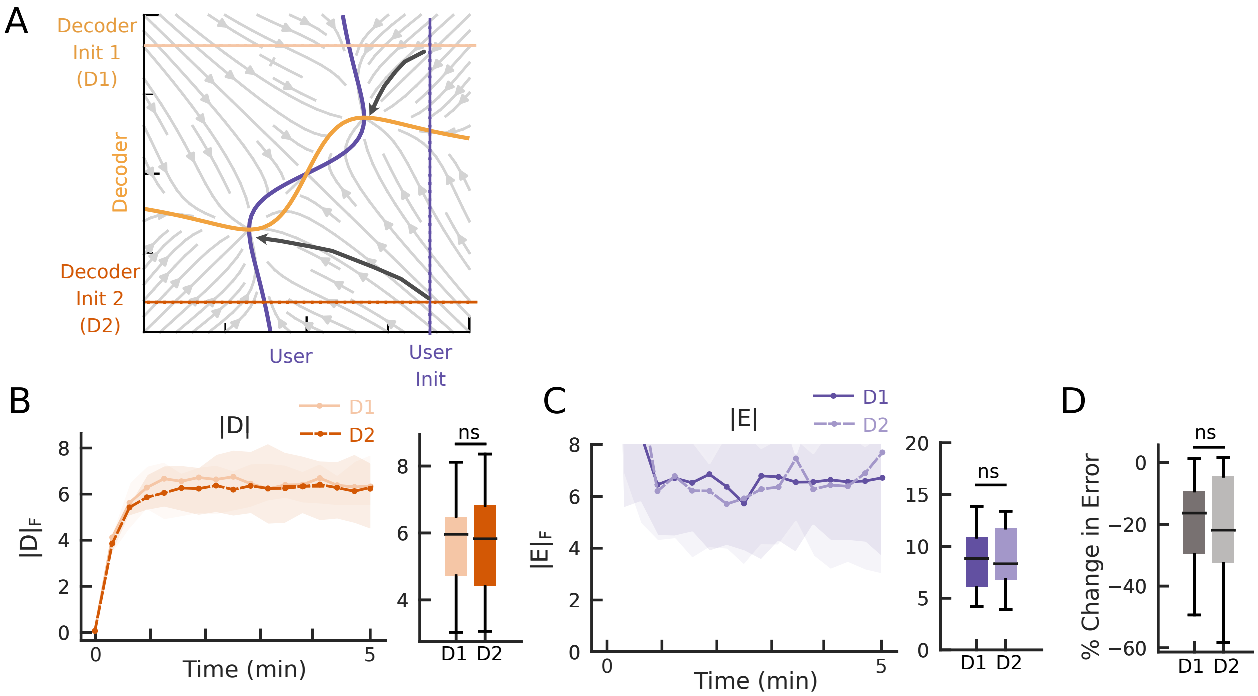

Figure 5.9: Decoder initialization does not affect interface performance.

A. Model predictions. Decoder initialization D1 (light orange) and initialization D2 (dark orange) are noted on the vertical axis of the diagram. Assumed user initialization (purple) is noted on the horizontal axis.

B. Left: Average magnitude of the decoder matrix (norm) as a function of time in the trial for D1 (solid light orange) and D2 (dashed dark orange) initializations (N = 14, median with shaded area indicating the interquartile range). Right: Boxplots are average decoder effort across the trial for each subject (N=14; Wilcoxon signed-rank test, ns > 0.05).

C. Left: Average magnitude of the user encoder matrix (norm) as a function of time in the trial for D1 (solid dark purple) and D2 (dashed light purple) decoder initializations (N = 14). Right: Boxplots are average effort for each subject across the trial (N = 14; Wilcoxon signed-rank test, ns > 0.05).

D. Percent change in error from the start of the trial (first 30 seconds) to the end of the trial (last 30 seconds), separated by high (dark gray) and low (light gray) decoder penalty term conditions (N=14, Wilcoxon signed-rank test, ns > 0.05).

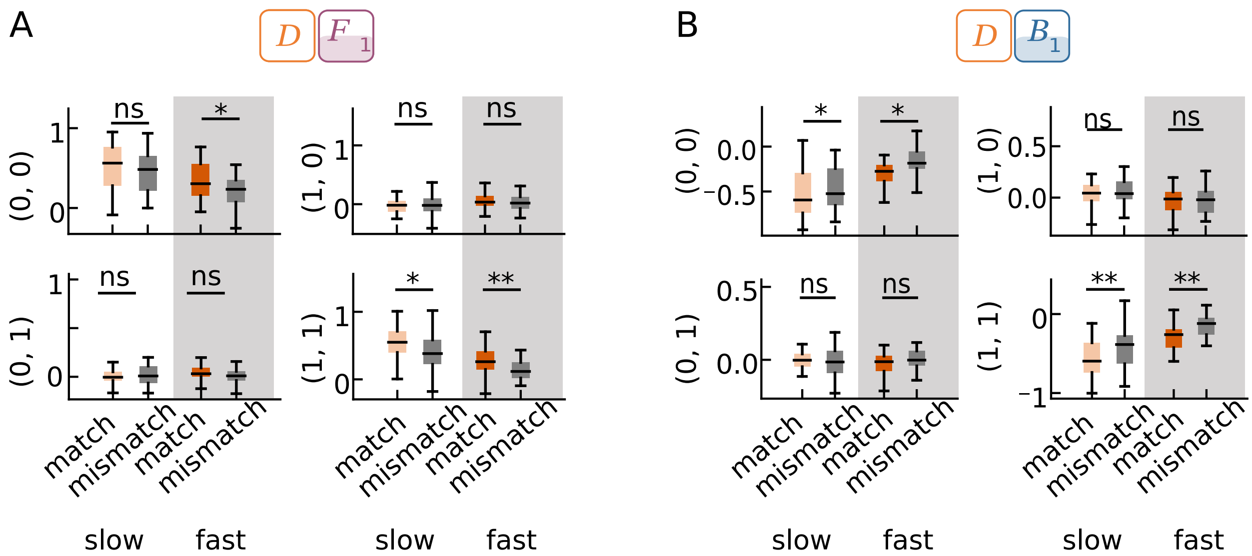

Figure 5.10: Decoder initialization slightly influences decoder-encoder pairs.

A. Last minute in trial of product of the average decoder matrix of each initialization with first-order feedforward (F_1) contributions of the average encoder matrix of each initialization separated by learning rate. Fast learning rate conditions are shaded gray. The matched conditions are the decoders and encoders of the same initialization and the mismatched conditions are the decoders and encoders of different initializations (N=28; Wilcoxon signed-rank test, ns > 0.05; *p < 0.05; **p < 0.001).

B. Last minute in trial of product of the average decoder matrix of each initialization with first-order feedback (B_1) contributions of the average encoder matrix of each initialization, separated by learning rate. Fast learning rate conditions are shaded gray. The matched conditions are the decoders and encoders of the same initialization and the mismatched conditions are the decoders and encoders of different initializations (N=28; Wilcoxon signed-rank test, ns > 0.05; *p < 0.05; **p < 0.001).

Chapter 6

DISCUSSION

Using biological signals as inputs to devices provides exciting new opportunities to treat neurological disorders and expand how humans can interact with technology. Wide adoption of these technologies will require ensuring interfaces work reliably for diverse users and applications. However, engineering reliable high-performance user-machine interfaces requires grappling with the rich closed-loop dynamics inherent in these systems, where users and machines can both adapt. Co-adaptation provides avenues to creating interfaces that can be personalized to each user and that can be adapted across tasks and contexts. Designing these co-adaptive systems to be stable and widely usable requires new techniques.

The overarching motivation of this thesis was to devise experimentally-validated methods to characterize and design co-adaptive user and decoder interactions in closed-loop interfaces. Creating tools specific to co-adaptation is key to design and, eventually, optimize future, co-adaptive neural interfaces. With this aim, we proposed a game-theoretic framework for modeling user-decoder co-adaptation in *Chapter 2*, verified this framework with myoelectric interfaces in *Chapter 3*, developed a control-theoretic tool for measuring user encoders in closed-loop interfaces in *Chapter 4*, and showed that decoder adaptation could shape user behavior in a principled way in *Chapter 5*. Below, we will summarize each chapter and then conclude with collaborations and future directions of the chapter.

6.1 Summary of Chapters

In *Chapter 2*, we presented a framework for modeling user-decoder co-adaption through game-theory with the aim of informing future co-adaptive decoder design. Co-adaptive interfaces come with complex user-decoder dynamics, and our game-theoretic methods allowed us to construct a versatile framework to consider these two-learner dynamics. Game theory is a field focused on understanding the decisions and responses of agents to each other, and it provides a suite of useful tools to analyze these interactions. We adopted the perspective that the user and decoder are playing a dynamic game (Başar and Olsder, 1999), treating the user and decoder as two agents adapting according to individual cost functions. This framework allows us to analyze the stability of user-decoder learning dynamics, and suggests that learning rates, decoder cost functions, and initial decoder parameters can all be used to influence overall system behavior.

In *Chapter 3*, we implemented our game-theoretic framework in myoelectric interfaces. In an experiment with 14 participants, we tested a new decoder adaptation scheme. This decoder scheme worked as prior adaptive decoders (Orsborn et al., 2012) had: performance improved regardless of initialization. We also varied the decoder parameters determined to be relevant for convergence: decoder initialization, learning rate, and penalty parameters. We found that these parameters influenced system performance as the model predicted, and learning rate was the only parameter to influence performance significantly. Our myoelectric interface also provided us with a platform to systematically test a new decoder and decoder design choices.

In *Chapter 4*, we leveraged control theory to construct closed-loop estimations of user encoders. Changes in task performance in co-adaptive interfaces are not a clear indicator of whether the user, decoder, or both are adapting. Tools that can separate the user's internal model, the *encoder*, from the decoder's model are important to quantify behavior in these interfaces. In this chapter, we showed how to frame the co-adaptive interface as a multi-input multi-output closed-loop system and construct the user's encoder based on key *feedforward* and *feedback* contributions. We estimated users encoders from our myoelectric experiment and found that our estimation was able to reconstruct task-relevant information from the trial. Furthermore, our estimation method was easily interpretable and required minimal data, lending itself as a potential tool for online encoder estimation. Measuring encoders during closed-loop control, through methods like ours, opens up avenues for quantifying and gauging learning while users are operating the interface.

Finally, in *Chapter 5*, we used our novel quantitative modeling techniques to reveal how decoding algorithms can shape user adaptation in neural interfaces. We showed that users learned within- and across-trials in our myoelectric experiment, suggesting that our interface was co-adaptive. This learning was affected and potentially disrupted by a mismatch in learning rates, as evidenced by the faster learning rate leading to poorer performance and less ideal decoder-encoder conditions. Changing the decoder penalty parameter influenced the encoder effort, as expected, and also affected the cursor speed. Noticing that the effects on cursor speed were more significant than the effects on encoder effort, we found that each participant had individual preferences in trading off between speed and effort. Taken together, our results in this chapter demonstrated how decoder design could influence user adaptation and could emphasize individual preferences in co-adaptive interfaces.

6.2 Game-theoretic Co-adaptive Framework Offers Insights for Novel Decoder Design

Throughout this thesis, we frame co-adaptation as a game between the user and decoder and formalize this interaction as a potential game. The user and decoder each adapt to minimize task error and their own individual effort. Our game-theory framing leveraged a general-purpose optimization tool to design interfaces with multiple objectives, balancing task error with decoder effort, which in turn influenced user effort. Current neural interfaces largely aim to maximizing performance without considering potentially important aspects like user effort. Future interfaces might be designed to consider additional objectives like stability, robustness, or performance across multiple tasks. We foresee our framework becoming especially useful as the functionalities of neural interface tasks grow, where we need systematic techniques for decoder design. In a collaboration led by Si Jia Li (PhD Candidate, UW Bioengineering), we tested this by extending the game-theoretic framework to a task with both continuous and discrete dynamics. Specifically, we considered how to use our game formulation to encourage decoders for different tasks to adapt to orthogonal spaces, with the goal that users would then learn two distinct control spaces. We believe that this framing can be a powerful way of shaping user behavior to learn advantageous mappings through closed-loop decoder adaptation.

While the game-theoretic framework provides a promising approach to model and implement rich co-adaptive dynamics, this and other existing frameworks are still limited. For example, current models are confined to 1 or 2 degrees of freedom of task control and simple low-order machine dynamics. Advancing models and computational tools for analyzing co-adaptation in high-dimensional interfaces will be critical for practical deployment in real-world contexts like assistive devices or rehabilitative interventions. Additionally, these frameworks rely on models of user learning that are relatively simplistic and require further validation. For instance, incorporating models of how different forms of sensory feedback influence user learning may be critical for methods to shape user strategies and encoders.

6.3 Co-adaptation Can Create Individualized, Generalizable Neural Interfaces

Improving interface performance is still a critical goal across neural interfaces, especially for ensuring interface accessibility and usability across users. In a myoelectric experiment with 14 participants, we tested a new decoder adaptation scheme in Chapter 3. Each participant had different control strategies, distinct physiologies, and varying expertise with EMG interfaces. In fact, participant strategies included treating the wrist as a joystick to control the direction of the cursor, opening and closing fists for control, engaging their pointer finger or thumb to point in their desired direction, or even waving their hands to attempt to activate the forearm muscles. Participants even cited exploring different strategies or solidifying to a strategy after multiple trials. Importantly, interface control and performance improved during our 5-minute trial, showing that our adaptive decoder scheme could customize interfaces to be usable by a diverse set of participants and strategies. Individualization of interfaces can improve interface accessibility, especially for individual with motor impairments (Yamagami et al., 2023). Closed-loop decoder adaptation improves interface performance, but, critically, it provides an avenue to enable neural interface control for a wide space of users.

Future directions of this work can lead to interfaces that are not only personalized for each user but can even adapt as users attempt different tasks. In a collaboration led by Amber Chou (PhD Candidate, UW ECE), we have been measuring user intent through eye movements instead of the task-based approaches used in this thesis. We can update the decoder to follow the user's eye gaze, which enables us to create interfaces that can adapt as users conduct unstructured tasks – such as drawing or scrolling on the screen. Additionally, eye gaze can provide key insights into changes in user behavior or strategies as users operate these neural interfaces across longer timelines and different environments.

Additionally, given that learning rate was such a key parameter for performance, we were curious to see if personalizing the learning rate to each user could affect interface control. In research led by Annika Pfister during Annika's undergraduate research experience with the Burden Lab, we adapted learning rate as users attempted to control the myoelectric interface. Determining whether to speed up or slow down the learning rate as users improve control is still an open question – speeding up the learning rate might achieve a more "optimal" decoder faster, but slowing the learning rate might enable users to improve control of their current decoder. This level of customization of decoder parameters holds promise as we consider how to engineer more complex interfaces across different modalities.

6.4 Curriculum-based Decoder Adaptation Can Strategically Shape User Learning

By developing new computational and experimental methods to study co-adaptive systems, our results reveal how interface algorithms can influence user behavior. Specifically, our model and experimental findings show that decoder learning rate, penalty terms, and initialization influence encoder-decoder dynamics in co-adaptive interfaces. These observations are consistent with recent work suggesting that adaptive algorithms can influence neural representations learned in brain-computer interfaces (Rajeswaran et al., 2024).

Neural interface control is a task that some users find challenging (Zhang et al., 2018), and our results hint at the possibility of designing a “curriculum” to help naive users learn to control neural interfaces. In fact, in research conducted by Sasha Burckhardt during Sasha’s undergraduate researcher role, we demonstrated the value of testing and designing curriculum for EMG activation and control. Prior to experimentation, we showed subjects an EMG-based “radar plot”, which visually depicted the activation of EMG channels with 64 spikes radiating from the center of a circle that dynamically adjusted based on the amplitude of the channel. The hypothesis was that the radar plot visualization training would discourage movements that did not change EMG measurements and lead to performance improvements. Contrary to our expectations, this pre-experimentation training confused participants and hurt performance. However, this work encapsulates the importance of appropriate curricula for users of neural interfaces. In this vein, we could leverage decoder adaptation to encourage user learning and exploration. Decoder learning rates, for example, may need to be significantly slower early in learning when users are widely exploring strategies (Herzfeld et al., 2014; Athalye et al., 2017; Dhawale et al., 2017).

Decoder adaptation is a powerful tool for enabling initial control, but it can also be a strategic technique for biasing users towards particular strategies. For example, a common aim of rehabilitation is to guide users to adopt specific actions and movements. Goal-directed decoder manipulations that encourage users to learn a particular behavior would provide a principled tool to use co-adaptive interfaces for rehabilitation (De Santis and Mussa-Ivaldi, 2020; Hung et al., 2021). We can borrow concepts from training neural networks (Bengio et al., 2009) to envision a decoder adaptation curriculum to intelligently shape user learning, fully harnessing the power of co-adaptive interfaces.

6.5 Conclusions

Neural interfaces can restore or augment human sensorimotor capabilities by converting high-bandwidth biological signals into control signals for an external device via a decoder algorithm. Leveraging user and decoder adaptation to create co-adaptive interfaces presents opportunities to improve usability and personalize devices. However, we lack principled methods to model and optimize the complex two-learner dynamics that arise in co-adaptive interfaces. The theme of this thesis was to provide new paths to harness the complex interactions between users and algorithms in closed-loop interfaces. We developed and experimentally-validated new computational methods based on control theory and game theory to analyze and generate predictions for user-decoder co-adaptive outcomes in continuous interactions. Importantly, our computational frameworks predicted the outcome of co-adaptive interface interactions and revealed how interface properties can shape user behavior. This thesis developed methods for developing new co-adaptive decoders, highlighted tools to design individualized neural interfaces, and demonstrated the potential for using strategic decoder schemed to guide user learning. Together, this thesis contributed versatile and principled ways to model and design the next generation of smart, personalized adaptive neural interfaces.

Chapter 7

APPENDIX

7.1 Game Theory Methods: Step-by-Step

For reference, we also include the step-by-step methods that yielded the results in Chapter 2. Explanation and background of the equations are in Chapter 2 Methods, but we will reiterate the equations here for completeness.

As in Chapter 2, signals are denoted with lower-case non-bolded symbols (e.g. x) and parameterizations of the transformations (i.e., matrices) are capitalized (e.g. X). Time is represented by t and indexed by the subscript $t = 1, 2, \ldots$. Task dimensionality is indexed by the subscript $d = 1, 2, \ldots$. The channels of biosignal data are by N and each channel n is indexed by the superscript $n \in \{1, \ldots, N\}$. We will suppress the sub- and/or super-scripts when they are clear from context.

Step 1: Equations and Symbols

User and Decoder Transformations

The user is represented as

$$u_t = f(E_t, \tau_t) = E_t \tau_t. \tag{7.1}$$

At each new time sample t, the EMG signals u are updated and are inputs to the decoder, D. The decoder outputs the cursor position:

$$y_t = f(D_t, u_t) = D_t u_t. \tag{7.2}$$

Table 7.1: Symbols Used in Game Theory Methods

	Symbol
Time	t
Task Dimensionality	d
Number of Biosignal Channels	N
User Activity	$u \in \mathbb{R}^{N\times 1}$
Encoder/User	$E \in \mathbb{R}^{N\times d}$
Decoder	$D \in \mathbb{R}^{d\times N}$
Target Position	$\tau \in \mathbb{R}^{d\times 1}$
Cursor Position	$y \in \mathbb{R}^{d\times 1}$

User and Decoder Cost Functions

The user and decoder aim to minimize their task error, which is calculated as the quadratic loss of the cursor and target positions (Héliot et al., 2010; Müller et al., 2017):

$$\varepsilon_t(\tau_t, y_t) = \|\tau_t - y_t\|_2^2. \tag{7.3}$$

The cost function of the user and decoder are:

$$\begin{aligned} c_E(E, D) &= \varepsilon(\tau, y) + \lambda_E \|E\|_2^2 \\ c_D(E, D) &= \varepsilon(\tau, y) + \lambda_D \|D\|_2^2. \end{aligned} \tag{7.4}$$

Potential Functions

The user and decoder form a *potential game*. If there exists a continuously differentiable function $\phi : \mathbb{R}^{n\times d} \mapsto \mathbb{R}$ such that $\frac{\partial \phi}{\partial E} = \frac{\partial C_E}{\partial E}$ and $\frac{\partial \phi}{\partial D} = \frac{\partial C_D}{\partial D}$, this function ϕ can be considered to be a potential function (Hespanha, J., 2017, Ch. 12). The potential function is:

$$\phi(E, D) = \varepsilon(\tau, y) + \lambda_E \|E\|_2^2 + \lambda_D \|D\|_2^2. \tag{7.5}$$

Step 2: Solving for Stationary Points of Potential Function

We start by re-writing the potential function with the error:

$$\phi(E, D) = \|\tau_t - y_t\|_2^2 + \lambda_E\|E\|_2^2 + \lambda_D\|D\|_2^2. \tag{7.6}$$

Since $y_t = D_t u_t$ (Eq. 7.2) and $u_t = E_t \tau_t$ (Eq.7.1), we can say

$$\phi(E, D) = \|\tau_t - D_t E_t \tau_t\|_2^2 + \lambda_E\|E\|_2^2 + \lambda_D\|D\|_2^2. \tag{7.7}$$

Now, we simplify to consider the scalar case, so the potential function simplifies to a function with only scalar variable (i.e. all variables $\in \mathbb{R}^1$):

$$\phi(E, D) = (\tau_t - D_t E_t \tau_t)^2 + \lambda_E(E)^2 + \lambda_D(D)^2. \tag{7.8}$$

The stationary points E^*, E^* are determined by solving $\frac{\partial \phi}{\partial E}\Big|_{E=E^*} = 0$ and $\frac{\partial \phi}{\partial D}\Big|_{D=D^*} = 0$.

The derivative of the potential function is

$$\begin{aligned} \frac{\partial \phi}{\partial E} &= 2(\tau - DE\tau)(-D\tau) + 2\lambda_E E \\ \frac{\partial \phi}{\partial D} &= 2(\tau - DE\tau)(-E\tau) + 2\lambda_D D. \end{aligned} \tag{7.9}$$

Solving for when the derivatives $\frac{\partial \phi}{\partial E} = 0$ and $\frac{\partial \phi}{\partial D} = 0$ with the assumption that $\lambda_D = \lambda_E = \lambda$ yields the following stationary points:

$$E^* = D^* = +\sqrt{1 - \frac{\lambda}{\tau^2}}, E^* = D^* = -\sqrt{1 - \frac{\lambda}{\tau^2}}, \text{ and } E^* = D^* = 0. \tag{7.10}$$

We can solve for the task error that is converged to by solving Eq. 7.3 at the stationary points:

$$\varepsilon_t(\tau, y) = \|\tau - (1 - \frac{\lambda}{\tau^2})\tau\|_2^2. \tag{7.11}$$

Step 3: Continuous-Time Stability Conditions at Stationary Points

Now, we will find the stability at the stationary points found in Eq. 7.10 assuming continuous-time dynamics. Continuous-time dynamics assume that the user and decoder adapt according to their continuous-time gradient, $\dot{E} = -\frac{\partial \phi}{\partial E}$ and $\dot{D} = -\frac{\partial \phi}{\partial D}$.

We next find the Jacobian of the continuous-time dynamics:

$$\begin{bmatrix} -2\lambda_D - 2\tau^2 E^2 & -2\tau^2 DE + 2\tau(-\tau DE + \tau) \\ -2\tau^2 DE + 2\tau(-\tau DE + \tau) & 2\lambda_E - 2\tau^2 D^2 \end{bmatrix}, \tag{7.12}$$

and then evaluate the Jacobian at the fixed points. We will evaluate the Jacobian at the fixed point $E^* = D^* = +\sqrt{1 - \frac{\lambda}{\tau^2}}$ and set $\lambda_D = \lambda_E = \lambda$:

$$\begin{bmatrix} -2\lambda - 2\tau^2(1 - \frac{\lambda}{\tau^2}) & -2\tau^2(1 - \frac{\lambda}{\tau^2}) + 2\tau(-\tau(1 - \frac{\lambda}{\tau^2}) + \tau) \\ 2\tau^2(1 - \frac{\lambda}{\tau^2}) + 2\tau(-\tau(1 - \frac{\lambda}{\tau^2}) + \tau) & -2\lambda - 2\tau^2(1 - \frac{\lambda}{\tau^2}) \end{bmatrix}. \tag{7.13}$$

The real part of all eigenvalues must be less than zero for continuous-time stability at the fixed points. Thus, we find that the eigenvalues of the Jacobian are:

$$\left\{-4\lambda, 4\lambda - 4\tau^2\right\}. \tag{7.14}$$

This provides us with bounds on the stationary points - $\lambda > 0$ and $\frac{\lambda}{\tau^2} < 1$ for stability at the stationary points.

Step 4: Discrete-Time Stability Conditions at Stationary Points

The steps for solving the discrete-time stability are similar to **Step 3**. The discrete-time dynamics can be written as $E^+ = E - \alpha_E \frac{\partial \phi}{\partial E}$ and $D^+ = D - \alpha_D \frac{\partial \phi}{\partial D}$. After substituting the derivatives from Eq. 7.9, the discrete-time dynamics become:

$$\begin{aligned} E^+ &= E - \alpha_E(\tau - DE\tau)(-D\tau) + 2\lambda_E E \\ D^+ &= D - \alpha_D 2(\tau - DE\tau)(-E\tau) + 2\lambda_D D. \end{aligned} \tag{7.15}$$

We find the Jacobian of the discrete-time dynamics:

$$\begin{bmatrix} \alpha_D(-2\lambda_D - 2\tau^2 E^2) + 1 & \alpha_D(-2\tau^2 DE + 2\tau(\tau - \tau DE)) \\ \alpha_E(-2\tau^2 DE + 2\tau(\tau - \tau DE)) & \alpha_E(-2\lambda_E - 2\tau^2 D^2) + 1 \end{bmatrix}. \tag{7.16}$$

Note the inclusions of α in the Jacobian. The influence of the user and decoder learning rates is seen in the discrete-time dynamics. We then evaluate the Jacobian at the same fixed point $E^* = D^* = +\sqrt{1 - \frac{\lambda}{\tau^2}}$ and set $\lambda_D = \lambda_E = \lambda$. We then find the eigenvalues of the Jacobian with the following assumptions: $\lambda_D = \lambda_E = \lambda = 0.5$ and $\tau = 1$:

$$\{1 - 2\alpha_D, 1 - 2\alpha_E\}. \tag{7.17}$$

Since the discrete-time dynamics converge if the absolute value of eigenvalues is less than 1 (i.e. lie inside the unit circle), we find that

$$0 < \alpha_E < 1, 0 < \alpha_D < 1 \tag{7.18}$$

are conditions for convergence to the stationary points. Since the eigenvalue with the maximum magnitude determines the asymptotic convergence rate, the learning rates $\alpha_E = \alpha_D = \frac{1}{2}$ yield the fastest convergence rate.

7.2 Tables for Sample Prior Co-adaptive Interface Work

Please note, the tables included in this section are meant to be helpful references and used as a starting point for future literature review. These tables are not exhaustive.

Table 7.2: Co-adaptive Interface Experiments

Paper	Experimental Interface Subjects	Decoder	Decoder Adaptation Method
Taylor et al. 2002	Intracortical BCI NHPs	Population Vector Analysis	Iterative refinement based on tracking neural tuning properties
Gage et al. 2005	Intracortical BCI rats	Kalman Filter	Sliding window of batch maximum likelihood estimations
Danziger et al. 2009	Kinematic interface humans	Nonlinear mapping of sensors o angles	Least mean squares or Moore-Penrose inverse
Orsborn et al. 2012, 2014	Intracortical BCI NHPs	Kalman Filter	Exponentially weighted sliding average (SmoothBatch)
Hahne et al. 2015	Myoelectric Interface humans	Linear Regression	Exponential recursive least squares
Rouse et al. 2016	Intracortical BCI NHPs	Weiner filter	L1-regularized least-squares regression model
De Santis et al. 2018	Kinematic interface humans	PCA	Recursive update PCA projection to 2D subspace with the highest signal variance
Silversmith et al. 2020	Intracortical BCI human (tetraparesis)	Kalman Filter	Recursive maximum-likelihood estimations
Rizzoglio et al. 2021	Kinematic Interface humans	Non-linear Autoencoder	Update according to latent manifold of user's input

Table 7.3: Mathematical Models for Co-adaptive Interfaces

Paper	User Model	Decoder Model	Adaptation Rule
Merel et al. 2015	Linear Quadratic Regulator (LQR)	Kalman Filter (Linear Quadratic Estimation)	Minimize task error through optimal control policy
Müller et al. 2017	Affine function	Affine function	Minimize a joint loss function through stochastic gradient descent
De Santis 2021	Non-stationary Gaussian generative process	Linear mapping for dimensionality reduction	Reward-based learning

BIBLIOGRAPHY

Laura Acqualagna, Loic Botrel, Carmen Vidaurre, Andrea Kübler, and Benjamin Blankertz. Large-Scale Assessment of a Fully Automatic Co-Adaptive Motor Imagery-Based Brain Computer Interface. *PLOS ONE*, 11(2):e0148886, February 2016. ISSN 1932-6203. doi: 10.1371/journal.pone.0148886. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0148886. Publisher: Public Library of Science.

Scott T. Albert and Reza Shadmehr. The neural feedback response to error as a teaching signal for the motor learning system. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 36(17):4832–4845, 2016. ISSN 1529-2401. doi: 10.1523/JNEUROSCI.0159-16.2016.

Vivek R. Athalye, Karunesh Ganguly, Rui M. Costa, and Jose M. Carmena. Emergence of coordinated neural dynamics underlies neuroprosthetic learning and skillful control. *Neuron*, 2017. doi: 10.1016/j.neuron.2017.01.016.

Tamer Başar and Geert Jan Olsder. *Dynamic noncooperative game theory*. Number 23 in Classics in applied mathematics. SIAM, Philadelphia, 2nd ed edition, 1999. ISBN 978-0-89871-429-6.

Alim Louis Benabid, Thomas Costecalde, Andrey Eliseyev, Guillaume Charvet, Alexandre Verney, Serpil Karakas, Michael Foerster, Aurélien Lambert, Boris Morinière, Neil Abroug, Marie-Caroline Schaeffer, Alexandre Moly, Fabien Sauter-Starace, David Ratel, Cecile Moro, Napoleon Torres-Martinez, Lilia Langar, Manuela Oddoux, Mircea Polosan, Stephane Pezzani, Vincent Auboiroux, Tetiana Aksenova, Corinne Mestais, and Stephan Chabardes. An exoskeleton controlled by an epidural wireless brain-machine interface in a tetraplegic patient: a proof-of-concept demonstration. *The Lancet. Neurology*, 18(12): 1112–1122, 2019. ISSN 1474-4465. doi: 10.1016/S1474-4422(19)30321-7.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 41–48. Association for Computing Machinery, 2009. ISBN 978-1-60558-516-1. doi: 10.1145/1553374.1553380. URL https://dl.acm.org/doi/10.1145/1553374.1553380.

David M. Brandman, Tommy Hosman, Jad Saab, Michael C. Burkhart, Benjamin E. Shanahan, John G. Ciancibello, Anish A. Sarma, Daniel J. Milstein, Carlos E. Vargas-Irwin, Brian Franco, Jessica Kelemen, Christine Blabe, Brian A. Murphy, Daniel R. Young, Francis R. Willett, Chethan Pandarinath, Sergey D. Stavisky, Robert F. Kirsch, Benjamin L. Walter, A. Bolu Ajiboye, Sydney S. Cash, Emad N. Eskandar, Jonathan P. Miller, Jennifer A. Sweet, Krishna V. Shenoy, Jaimie M. Henderson, Beata Jarosiewicz, Matthew T. Harrison, John D. Simeral, and Leigh R. Hochberg. Rapid calibration of an intracortical brain-computer interface for people with tetraplegia. *Journal of Neural Engineering*, 15(2):026007, 2018. ISSN 1741-2552. doi: 10.1088/1741-2552/aa9ee7.

Daniel A Braun, Pedro A Ortega, and Daniel M Wolpert. Nash equilibria in multi-agent motor interactions. *PLoS computational biology*, 5(8):e1000468, 2009.

Steven L Brunton and J Nathan Kutz. *Data Driven Science & Engineering*. Cambridge University Press, 2017.

Jose M. Carmena. Advances in Neuroprosthetic Learning and Control. *PLoS Biology*, 11 (5):e1001561, May 2013. ISSN 1545-7885. doi: 10.1371/journal.pbio.1001561. URL https://dx.plos.org/10.1371/journal.pbio.1001561.

Jose M Carmena, Mikhail A Lebedev, Roy E Crist, Joseph E O'Doherty, David M Santucci, Dragan F Dimitrov, Parag G Patil, Craig S Henriquez, and Miguel A. L Nicolelis. Learning to Control a Brain–Machine Interface for Reaching and Grasping by Primates. *PLoS Biology*, 1(2), October 2003. ISSN 1545-7885. doi: 10.1371/journal.pbio.0000042. URL https://dx.plos.org/10.1371/journal.pbio.0000042.

Steven M. Chase, Andrew B. Schwartz, and Robert E. Kass. Bias, optimal linear estimation, and the differences between open-loop simulation and closed-loop performance of spiking-based brain-computer interface algorithms. *Neural Networks: The Official Journal of the International Neural Network Society*, 22(9):1203–1213, November 2009. doi: 10.1016/j.neunet.2009.05.005.

Benjamin Chasnov, Lillian Ratliff, Eric Mazumdar, and Samuel Burden. Convergence analysis of Gradient-Based learning in continuous games. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 115 of *Proceedings of Machine Learning Research*, pages 935–944. PMLR, 2020. URL http://proceedings.mlr.press/v115/chasnov20a.html.

Benjamin J. Chasnov, Lillian J. Ratliff, and Samuel A. Burden. Human adaptation to adaptive machines converges to game-theoretic equilibria, 2023. URL http://arxiv.org/abs/2305.01124.

Stacie A. Ph D. Chvatal and Lena H. Ph D. Ting. Common muscle synergies for balance and walking. *Frontiers in Computational Neuroscience*, 7, May 2013. ISSN 1662-5188. doi: 10.3389/fncom.2013.00048. URL https://www.frontiersin.org/articles/10.3389/fncom.2013.00048.

Jennifer L. Collinger, Brian Wodlinger, John E. Downey, Wei Wang, Elizabeth C. Tyler-Kabara, Douglas J. Weber, Angus JC McMorland, Meel Velliste, Michael L. Boninger, and Andrew B. Schwartz. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 381(9866):557–564, February 2013. ISSN 0140-6736, 1474-547X. doi: 10.1016/S0140-6736(12)61816-9. URL https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(12)61816-9/fulltext. Publisher: Elsevier.

M Couraud, D Cattaert, F Paclet, P Y Oudeyer, and A de Rugy. Model and experiments to optimize co-adaptation in a simplified myoelectric control system. *Journal of Neural*

Engineering, 15(2):026006, 2018. ISSN 1741-2560, 1741-2552. doi: 10.1088/1741-2552/aa87cf. URL https://iopscience.iop.org/article/10.1088/1741-2552/aa87cf.

Maria C. Dadarlat, Ryan A. Canfield, and Amy L. Orsborn. Neural Plasticity in Sensorimotor Brain–Machine Interfaces. *Annual Review of Biomedical Engineering*, 25(Volume 25, 2023), June 2023. ISSN 1523-9829, 1545-4274. doi: 10.1146/annurev-bioeng-110220-110833. URL https://www.annualreviews.org/content/journals/10.1146/annurev-bioeng-110220-110833.

Siddharth Dangi, Amy L. Orsborn, Helene G. Moorman, and Jose M. Carmena. Design and Analysis of Closed-Loop Decoder Adaptation Algorithms for Brain-Machine Interfaces. *Neural Computation*, 25(7):1693–1731, July 2013. ISSN 0899-7667, 1530-888X. doi: 10.1162/NECO_a_00460. URL https://www.mitpressjournals.org/doi/abs/10.1162/NECO_a_00460.

Z. Danziger, A. Fishbach, and F.A. Mussa-Ivaldi. Learning Algorithms for Human–Machine Interfaces. *IEEE Transactions on Biomedical Engineering*, 56(5):1502–1511, May 2009. ISSN 0018-9294, 1558-2531. doi: 10.1109/TBME.2009.2013822. URL http://ieeexplore.ieee.org/document/4776455/.

Dalia De Santis. A Framework for Optimizing Co-adaptation in Body-Machine Interfaces. *Frontiers in Neurorobotics*, 15, 2021. doi: 10.3389/fnbot.2021.662181.

Dalia De Santis and Ferdinando A. Mussa-Ivaldi. Guiding functional reorganization of motor redundancy using a body-machine interface. *Journal of NeuroEngineering and Rehabilitation*, 17(1):61, May 2020. ISSN 1743-0003. doi: 10.1186/s12984-020-00681-7. URL https://doi.org/10.1186/s12984-020-00681-7.

Dalia De Santis, Patrycja Dzialecka, and Ferdinando A. Mussa-Ivaldi. Unsupervised Coadaptation of an Assistive Interface to Facilitate Sensorimotor Learning of Redundant Control. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics*

(Biorob), pages 801–806, August 2018. doi: 10.1109/BIOROB.2018.8487912. ISSN: 2155-1782.

Alan D. Degenhart, William E. Bishop, Emily R. Oby, Elizabeth C. Tyler-Kabara, Steven M. Chase, Aaron P. Batista, and Byron M. Yu. Stabilization of a brain-computer interface via the alignment of low-dimensional spaces of neural activity. *Nature Biomedical Engineering*, 4(7):672–685, 2020. ISSN 2157-846X. doi: 10.1038/s41551-020-0542-9.

Ashesh K. Dhawale, Maurice A. Smith, and Bence P. Ölveczky. The role of variability in motor learning. *Annual review of neuroscience*, 40:479–498, 2017. ISSN 0147-006X. doi: 10.1146/annurev-neuro-072116-031548. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6091866/.

Jack DiGiovanna, Babak Mahmoudi, Jose Fortes, Jose C. Principe, and Justin C. Sanchez. Coadaptive brain–machine interface via reinforcement learning. *IEEE Transactions on Biomedical Engineering*, 56(1):54–64, 2009. ISSN 1558-2531. doi: 10.1109/TBME.2008.926699.

Frank M. Drop, Daan M. Pool, Herman J. Damveld, Marinus M. van Paassen, and Max Mulder. Identification of the feedforward component in manual control with predictable target signals. *IEEE Transactions on Cybernetics*, 43(6):1936–1949, 2013. ISSN 2168-2275. doi: 10.1109/TSMCB.2012.2235829.

Jeremy L. Emken, Raul Benitez, Athanasios Sideris, James E. Bobrow, and David J. Reinkensmeyer. Motor adaptation as a greedy optimization of error and effort. *Journal of Neurophysiology*, 97(6):3997–4006, 2007. ISSN 0022-3077. doi: 10.1152/jn.01095.2006. URL https://journals.physiology.org/doi/full/10.1152/jn.01095.2006.

Eberhard E Fetz. Volitional control of neural activity: implications for brain–computer interfaces. *The Journal of Physiology*, 579:571–579, 2007. ISSN 0022-3751. doi: 10.1113/jphysiol.2006.127142.

James M. Finley, Amy J. Bastian, and Jinger S. Gottschall. Learning to be economical: the energy cost of walking tracks motor adaptation. *The Journal of Physiology*, 591(4):1081–1095, 2013. ISSN 1469-7793. doi: 10.1113/jphysiol.2012.245506.

T. Flash and N. Hogan. The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of Neuroscience*, 5(7):1688–1703, 1985. ISSN 0270-6474, 1529-2401. doi: 10.1523/JNEUROSCI.05-07-01688.1985. URL https://www.jneurosci.org/content/5/7/1688.

Drew Fudenberg and David K. Levine. *The Theory of Learning in Games.* MIT Press, 1998.

Gregory J Gage, Kip A Ludwig, Kevin J Otto, Edward L Ionides, and Daryl R Kipke. Naïve coadaptive cortical control. *Journal of Neural Engineering*, 2(2):52–63, June 2005. ISSN 1741-2560, 1741-2552. doi: 10.1088/1741-2560/2/2/006. URL https://iopscience.iop.org/article/10.1088/1741-2560/2/2/006.

Karunesh Ganguly and Jose M. Carmena. Emergence of a stable cortical map for neuroprosthetic control. *PLOS Biology*, 2009. doi: 10.1371/journal.pbio.1000153.

A. P. Georgopoulos, A. B. Schwartz, and R. E. Kettner. Neuronal population coding of movement direction. *Science*, 233(4771):1416–1419, 1986. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.3749885. URL https://science.sciencemag.org/content/233/4771/1416.

Andrea Gigli, Arjan Gijsberts, and Claudio Castellini. Unsupervised myocontrol of a virtual hand based on a coadaptive abstract motor mapping. In *2022 International Conference on Rehabilitation Robotics (ICORR)*, pages 1–6. IEEE, 2022. ISBN 978-1-66548-829-7. doi: 10.1109/ICORR55369.2022.9896414. URL https://ieeexplore.ieee.org/document/9896414/.

Vikash Gilja, Paul Nuyujukian, Cindy A. Chestek, John P. Cunningham, Byron M. Yu, Joline M. Fan, Mark M. Churchland, Matthew T. Kaufman, Jonathan C. Kao, Stephen I. Ryu, and Krishna V. Shenoy. A high-performance neural prosthesis enabled by control algorithm design. *Nature Neuroscience*, 15(12):1752–1757, 2012. doi: 10.1038/nn.3265.

Matthew D Golub, Byron M Yu, and Steven M Chase. Internal models for interpreting neural population activity during sensorimotor control. *eLife*, 4:e10015, December 2015. ISSN 2050-084X. doi: 10.7554/eLife.10015. URL https://doi.org/10.7554/eLife.10015. Publisher: eLife Sciences Publications, Ltd.

Andrea M. Green and John F. Kalaska. Learning to move machines with the mind. *Trends in Neurosciences*, 34(2):61–75, February 2011. ISSN 0166-2236. doi: 10.1016/j.tins.2010.11.003. URL https://www.sciencedirect.com/science/article/pii/S0166223610001682.

Janne M. Hahne, Sven Dähne, Han-Jeong Hwang, Klaus-Robert Müller, and Lucas C. Parra. Concurrent adaptation of human and machine improves simultaneous and proportional myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(4):618–627, 2015. ISSN 1558-0210. doi: 10.1109/TNSRE.2015.2401134.

Janne M. Hahne, Marko Markovic, and Dario Farina. User adaptation in Myoelectric Man-Machine Interfaces. *Scientific Reports*, 7(1):4437, June 2017. ISSN 2045-2322. doi: 10.1038/s41598-017-04255-x. URL https://www.nature.com/articles/s41598-017-04255-x.

Christopher M. Harris and Daniel M. Wolpert. Signal-dependent noise determines motor planning. *Nature*, 394(6695):780–784, 1998. ISSN 1476-4687. doi: 10.1038/29528. URL https://www.nature.com/articles/29528.

Sara Hendren. *What Can a Body Do?* Riverhead Books, 2020. ISBN 978-0-7352-2000-3.

David J. Herzfeld, Pavan A. Vaswani, Mollie K. Marko, and Reza Shadmehr. A memory of errors in sensorimotor learning. *Science*, 345(6202):1349–1353, 2014. ISSN 0036-

8075, 1095-9203. doi: 10.1126/science.1253138. URL https://science.sciencemag.org/content/345/6202/1349.

Hespanha, J. *Noncooperative Game Theory: An Introduction for Engineers and Computer Scientists*. Princeton University Press, 2017. URL doi:10.2307/j.ctt1vwmgbh.

Leigh R. Hochberg, Mijail D. Serruya, Gerhard M. Friehs, Jon A. Mukand, Maryam Saleh, Abraham H. Caplan, Almut Branner, David Chen, Richard D. Penn, and John P. Donoghue. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 442(7099):164–171, 2006. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature04970. URL http://www.nature.com/doifinder/10.1038/nature04970.

Leigh R. Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y. Masse, John D. Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S. Cash, Patrick van der Smagt, and John P. Donoghue. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485(7398):372–375, 2012. doi: 10.1038/nature11076.

Han-Lin Hsieh and Maryam M. Shanechi. Optimizing the learning rate for adaptive estimation of neural encoding models. *PLOS Computational Biology*, 14(5):e1006168, May 2018. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1006168. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1006168. Publisher: Public Library of Science.

Xuhui Hu, Aiguo Song, Hong Zeng, Zhikai Wei, Hanjie Deng, and Dapeng Chen. Bridging human-robot co-adaptation via biofeedback for continuous myoelectric control. *IEEE Robotics and Automation Letters*, 8(12):8573–8580, 2023. ISSN 2377-3766. doi: 10.1109/LRA.2023.3330053. URL https://ieeexplore.ieee.org/abstract/document/10306285/authors.

Na-Teng Hung, Vivek Paul, Prashanth Prakash, Torin Kovach, Gene Tacy, Goran Tomic, Sangsoo Park, Tyler Jacobson, Alix Jampol, Pooja Patel, Anya Chappel, Erin King, and Marc W. Slutzky. Wearable myoelectric interface enables high-dose, home-based training in severely impaired chronic stroke survivors. *Annals of Clinical and Translational Neurology*, 8(9):1895–1905, 2021. ISSN 2328-9503. doi: 10.1002/acn3.51442.

C. Hutchinson. An example of the equivalence of the Kalman and Wiener filters. *IEEE Transactions on Automatic Control*, 11(2):324–324, April 1966. ISSN 0018-9286. doi: 10.1109/TAC.1966.1098326. URL http://ieeexplore.ieee.org/document/1098326/.

Rodolphe Héliot, Karunesh Ganguly, Jessica Jimenez, and Jose M. Carmena. Learning in Closed-Loop Brain–Machine Interfaces: Modeling and Experimental Validation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(5), October 2010. ISSN 1941-0492. doi: 10.1109/TSMCB.2009.2036931.

K A Ingraham, C D Remy, and E J Rouse. The role of user preference in the customized control of robotic exoskeletons. *Science robotics*, 7(64):eabj3487, March 2022. ISSN 2470-9476. doi: 10.1126/scirobotics.abj3487. URL http://dx.doi.org/10.1126/scirobotics.abj3487.

Andrew Jackson and Eberhard E. Fetz. Interfacing with the Computational Brain. *Ieee Transactions on Neural Systems and Rehabilitation Engineering*, 19(5):534–541, October 2011. ISSN 1534-4320. doi: 10.1109/TNSRE.2011.2158586. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3372096/.

Beata Jarosiewicz, Steven M. Chase, George W. Fraser, Meel Velliste, Robert E. Kass, and Andrew B. Schwartz. Functional network reorganization during learning in a brain-computer interface paradigm. *Proceedings of the National Academy of Sciences of the United States of America*, 105(49):19486–19491, December 2008. ISSN 0027-8424. doi: 10.1073/pnas.0808113105. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2614787/.

Beata Jarosiewicz, Nicolas Y. Masse, Daniel Bacher, Sydney S. Cash, Emad Eskandar, Gerhard Friehs, John P. Donoghue, and Leigh R. Hochberg. Advantages of closed-loop calibration in intracortical brain–computer interfaces for people with tetraplegia. *Journal* *of Neural Engineering*, 10(4), July 2013. ISSN 1741-2552. doi: 10.1088/1741-2560/10/4/046012. URL https://dx.doi.org/10.1088/1741-2560/10/4/046012. Publisher: IOP Publishing.

Beata Jarosiewicz, Anish A. Sarma, Daniel Bacher, Nicolas Y. Masse, John D. Simeral, Brittany Sorice, Erin M. Oakley, Christine Blabe, Chethan Pandarinath, Vikash Gilja, Sydney S. Cash, Emad N. Eskandar, Gerhard Friehs, Jaimie M. Henderson, Krishna V. Shenoy, John P. Donoghue, and Leigh R. Hochberg. Virtual typing by people with tetraplegia using a self-calibrating intracortical brain-computer interface. *Science Translational Medicine*, 7(313):313ra179–313ra179, November 2015. ISSN 1946-6234, 1946-6242. doi: 10.1126/scitranslmed.aac7328. URL http://stm.sciencemag.org/lookup/doi/10.1126/scitranslmed.aac7328.

Nathanaël Jarrassé, Themistoklis Charalambous, and Etienne Burdet. A Framework to Describe, Analyze and Generate Interactive Motor Behaviors. *PLOS ONE*, 7(11):e49945, November 2012. ISSN 1932-6203. doi: 10.1371/journal.pone.0049945. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0049945. Publisher: Public Library of Science.

Shinsuke Koyama, Steven M. Chase, Andrew S. Whitford, Meel Velliste, Andrew B. Schwartz, and Robert E. Kass. Comparison of brain–computer interface decoding algorithms in open-loop and closed-loop control. *Journal of Computational Neuroscience*, 29(1-2):73–87, August 2010. ISSN 0929-5313, 1573-6873. doi: 10.1007/s10827-009-0196-9. URL http://link.springer.com/10.1007/s10827-009-0196-9.

Yanan Li, Keng Peng Tee, Rui Yan, Wei Liang Chan, and Yan Wu. A framework of human–robot coordination based on game theory and policy iteration. *IEEE Transactions on Robotics*, 32(6):1408–1418, 2016.

Yanan Li, Gerolamo Carboni, Franck Gonzalez, Domenico Campolo, and Etienne Burdet.

Differential game theory for versatile physical human–robot interaction. *Nature Machine Intelligence*, 1(1):36–43, 2019.

Ken-Fu Liang and Jonathan C. Kao. Deep Learning Neural Encoders for Motor Cortex. *IEEE Transactions on Biomedical Engineering*, 67(8):2145–2158, August 2020. ISSN 0018-9294, 1558-2531. doi: 10.1109/TBME.2019.2955722. URL https://ieeexplore.ieee.org/document/8911512/.

Timothy P. Lillicrap, Adam Santoro, Luke Marris, Colin J. Akerman, and Geoffrey Hinton. Backpropagation and the brain. *Nature Reviews Neuroscience*, April 2020. ISSN 1471-003X, 1471-0048. doi: 10.1038/s41583-020-0277-3. URL http://www.nature.com/articles/s41583-020-0277-3.

Maneeshika M. Madduri, Samuel A. Burden, and Amy L. Orsborn. A Game-Theoretic Model for Co-Adaptive Brain-Machine Interfaces. In *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 327–330, May 2021. doi: 10.1109/NER49283.2021.9441081. ISSN: 1948-3554.

Maneeshika M. Madduri, Momona Yamagami, A.X.T. Millevolte, Si Jia Li, S. A. Burden, and A. L. Orsborn. Co-Adaptive Myoelectric Interface for Continuous Control. In *4th IFAC Conference on Cyber-Physical Human Systems*, 2022.

Maneeshika M. Madduri, Samuel A. Burden, and Amy L. Orsborn. Biosignal-based co-adaptive user-machine interfaces for motor control. *Current Opinion in Biomedical Engineering*, 27:100462, September 2023. ISSN 2468-4511. doi: 10.1016/j.cobme.2023.100462. URL https://www.sciencedirect.com/science/article/pii/S2468451123000181.

Adam H. Marblestone, Greg Wayne, and Konrad P. Kording. Toward an Integration of Deep Learning and Neuroscience. *Frontiers in Computational Neuroscience*, 10:94, September 2016. ISSN 1662-5188. doi: 10.3389/fncom.2016.00094. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5021692/.

Eric Mazumdar, Lillian J. Ratliff, and S. Shankar Sastry. On Gradient-Based Learning in Continuous Games. *SIAM Journal on Mathematics of Data Science*, 2(1):103–131, January 2020. doi: 10.1137/18M1231298. URL https://epubs.siam.org/doi/abs/10.1137/18M1231298. Publisher: Society for Industrial and Applied Mathematics.

Kirsty A. McDonald, Joseph P. Cusumano, Andrew Hieronymi, and Jonas Rubenson. Humans trade off whole-body energy cost to avoid overburdening muscles while walking. *Proceedings of the Royal Society B: Biological Sciences*, 289:20221189, 2022. ISSN 0962-8452. doi: 10.1098/rspb.2022.1189. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9597406/.

D.T. McRuer and H.R. Jex. A review of quasi-linear pilot models. *IEEE Transactions on Human Factors in Electronics*, HFE-8(3):231–249, 1967. ISSN 2168-2852. doi: 10.1109/THFE.1967.234304. URL https://ieeexplore.ieee.org/document/1698271. Conference Name: IEEE Transactions on Human Factors in Electronics.

Josh Merel, Donald M. Pianto, John P. Cunningham, and Liam Paninski. Encoder-Decoder Optimization for Brain-Computer Interfaces. *PLOS Computational Biology*, 2015. doi: 10.1371/journal.pcbi.1004288.

Chet T. Moritz. Now is the critical time for engineered neuroplasticity. *Neurotherapeutics*, 15(3):628–634, 2018. doi: 10.1007/s13311-018-0637-0.

David A. Moses, Sean L. Metzger, Jessie R. Liu, Gopala K. Anumanchipalli, Joseph G. Makin, Pengfei F. Sun, Josh Chartier, Maximilian E. Dougherty, Patricia M. Liu, Gary M. Abrams, Adelyn Tu-Chan, Karunesh Ganguly, and Edward F. Chang. Neuroprosthesis for Decoding Speech in a Paralyzed Person with Anarthria. *New England Journal of Medicine*, 385(3):217–227, July 2021. ISSN 0028-4793. doi: 10.1056/NEJMoa2027540. URL https://www.nejm.org/doi/10.1056/NEJMoa2027540. Publisher: Massachusetts Medical Society.

Jan Saputra Müller, Carmen Vidaurre, Martijn Schreuder, Frank C. Meinecke, Paul von

Bünau, and Klaus-Robert Müller. A mathematical model for the two-learners problem. *Journal of Neural Engineering*, 2017. doi: 10.1088/1741-2552/aa620b.

Yurii Nesterov and Vladimir Spokoiny. Random Gradient-Free Minimization of Convex Functions. *Foundations of Computational Mathematics*, 17(2), April 2017. ISSN 1615-3375, 1615-3383. doi: 10.1007/s10208-015-9296-2. URL http://link.springer.com/10.1007/s10208-015-9296-2.

John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.

Emily R. Oby, Matthew D. Golub, Jay A. Hennig, Alan D. Degenhart, Elizabeth C. Tyler-Kabara, Byron M. Yu, Steven M. Chase, and Aaron P. Batista. New neural activity patterns emerge with long-term learning. *Proceedings of the National Academy of Sciences of the United States of America*, 116(30):15210–15215, July 2019. ISSN 1091-6490. doi: 10.1073/pnas.1820296116.

A. L. Orsborn, S. Dangi, H. G. Moorman, and J. M. Carmena. Closed-Loop Decoder Adaptation on Intermediate Time-Scales Facilitates Rapid BMI Performance Improvements Independent of Decoder Initialization Conditions. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(4):468–477, July 2012. ISSN 1534-4320, 1558-0210. doi: 10.1109/TNSRE.2012.2185066. URL http://ieeexplore.ieee.org/document/6144747/.

Amy Orsborn and Jose M. Carmena. Creating new functional circuits for action via brain-machine interfaces. *Frontiers in Computational Neuroscience*, 2013. doi: 10.3389/fncom.2013.00157.

Amy L Orsborn and Bijan Pesaran. Parsing learning in networks using brain–machine interfaces. *Current Opinion in Neurobiology*, 2017. doi: 10.1016/j.conb.2017.08.002.

Amy L. Orsborn, Siddharth Dangi, Helene G. Moorman, and Jose M. Carmena. Exploring time-scales of closed-loop decoder adaptation in brain-machine interfaces. *Annual Inter-*

national Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference, 2011:5436–5439, 2011. ISSN 2694-0604. doi: 10.1109/IEMBS.2011.6091387.

Amy L. Orsborn, Helene G. Moorman, Simon A. Overduin, Maryam M. Shanechi, Dragan F. Dimitrov, and Jose M. Carmena. Closed-Loop Decoder Adaptation Shapes Neural Plasticity for Skillful Neuroprosthetic Control. *Neuron*, 82(6):1380–1393, June 2014. ISSN 08966273. doi: 10.1016/j.neuron.2014.04.048. URL https://linkinghub.elsevier.com/retrieve/pii/S0896627314003638.

Chethan Pandarinath and Sliman J. Bensmaia. The science and engineering behind sensitized brain-controlled bionic hands. *Physiological Reviews*, 102(2):551–604, 2022. ISSN 1522-1210. doi: 10.1152/physrev.00034.2020.

Chethan Pandarinath, Paul Nuyujukian, Christine H Blabe, Brittany L Sorice, Jad Saab, Francis R Willett, Leigh R Hochberg, Krishna V Shenoy, and Jaimie M Henderson. High performance communication by people with paralysis using an intracortical brain-computer interface. *eLife*, 6:e18554, February 2017. ISSN 2050-084X. doi: 10.7554/eLife.18554. URL https://doi.org/10.7554/eLife.18554. Publisher: eLife Sciences Publications, Ltd.

János A Perge, Mark L Homer, Wasim Q Malik, Sydney Cash, Emad Eskandar, Gerhard Friehs, John P Donoghue, and Leigh R Hochberg. Intra-day signal instabilities affect decoding performance in an intracortical neural interface system. *Journal of Neural Engineering*, 10(3):036004, June 2013. ISSN 1741-2560, 1741-2552. doi: 10.1088/1741-2560/10/3/036004. URL http://stacks.iop.org/1741-2552/10/i=3/a=036004?key=crossref.e08f0fcb98e87e334c7231b99a1b2a9d.

Camilla Pierella, Maura Casadio, Ferdinando A. Mussa-Ivaldi, and Sara A. Solla. The dynamics of motor learning through the formation of internal models. *PLOS Computational Biology*, 2019. doi: 10.1371/journal.pcbi.1007118.

Alexandra A. Portnova-Fahreeva, Fabio Rizzoglio, Ferdinando A. Mussa-Ivaldi, and Eric Rombokas. Autoencoder-based myoelectric controller for prosthetic hands. *Frontiers in Bioengineering and Biotechnology*, 11:1134135, 2023. ISSN 2296-4185. doi: 10.3389/fbioe.2023.1134135. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10331017/.

Dean A Pospisil, Anitha Pasupathy, and Wyeth Bair. 'Artiphysiology' reveals V4-like shape tuning in a deep network trained for image classification. *eLife*, 7:e38242, December 2018. ISSN 2050-084X. doi: 10.7554/eLife.38242. URL https://doi.org/10.7554/eLife.38242. Publisher: eLife Sciences Publications, Ltd.

Saritha M. Radhakrishnan, Stuart N. Baker, and Andrew Jackson. Learning a Novel Myoelectric-Controlled Interface Task. *Journal of Neurophysiology*, 100(4):2397–2408, October 2008. ISSN 0022-3077. doi: 10.1152/jn.90614.2008. URL https://journals.physiology.org/doi/full/10.1152/jn.90614.2008. Publisher: American Physiological Society.

Pavithra Rajeswaran, Alexandre Payeur, Guillaume Lajoie, and Amy L. Orsborn. Assistive sensory-motor perturbations influence learned neural representations, 2024. URL http://biorxiv.org/lookup/doi/10.1101/2024.03.20.585972.

L. J. Ratliff, S. A. Burden, and S. S. Sastry. Characterization and computation of local Nash equilibria in continuous games. In *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, October 2013. doi: 10.1109/Allerton.2013.6736623.

David J Reinkensmeyer, Etienne Burdet, Maura Casadio, John W Krakauer, Gert Kwakkel, Catherine E Lang, Stephan P Swinnen, Nick S Ward, and Nicolas Schweighofer. Computational neurorehabilitation: modeling plasticity and learning to predict recovery. *Journal of neuroengineering and rehabilitation*, 13(1):1–25, 2016.

Fabio Rizzoglio, Maura Casadio, Dalia De Santis, and Ferdinando A. Mussa-Ivaldi. Building an adaptive interface via unsupervised tracking of latent manifolds. *Neural Networks*, 137:174–187, 2021. ISSN 0893-6080. doi: 10.1016/j.neunet.2021.01.009. URL https://www.sciencedirect.com/science/article/pii/S0893608021000174.

A. G. Rouse, J. J. Williams, J. J. Wheeler, and D. W. Moran. Spatial co-adaptation of cortical control columns in a micro-ECoG brain–computer interface. *Journal of Neural Engineering*, 13(5):056018, September 2016. ISSN 1741-2552. doi: 10.1088/1741-2560/13/5/056018. URL https://dx.doi.org/10.1088/1741-2560/13/5/056018. Publisher: IOP Publishing.

Patrick T. Sadtler, Kristin M. Quick, Matthew D. Golub, Steven M. Chase, Stephen I. Ryu, Elizabeth C. Tyler-Kabara, Byron M. Yu, and Aaron P. Batista. Neural constraints on learning. *Nature*, 2014. doi: 10.1038/nature13665.

Stefan Salminger, Heiko Stino, Lukas H. Pichler, Clemens Gstoettner, Agnes Sturma, Johannes A. Mayer, Michael Szivak, and Oskar C. Aszmann. Current rates of prosthetic usage in upper-limb amputees – have innovations had an impact on device acceptance? *Disability and Rehabilitation*, 44(14):3708–3713, July 2022. ISSN 0963-8288, 1464-5165. doi: 10.1080/09638288.2020.1866684. URL https://www.tandfonline.com/doi/full/10.1080/09638288.2020.1866684.

Gang Seo, Ameen Kishta, Emily Mugler, Marc W. Slutzky, and Jinsook Roh. Myoelectric interface training enables targeted reduction in abnormal muscle co-activation. *Journal of Neuroengineering and Rehabilitation*, 19(1):67, 2022. ISSN 1743-0003. doi: 10.1186/s12984-022-01045-z.

Mijail D. Serruya, Nicholas G. Hatsopoulos, Liam Paninski, Matthew R. Fellows, and John P. Donoghue. Instant neural control of a movement signal. *Nature*, 416(6877):141–142, March 2002. ISSN 1476-4687. doi: 10.1038/416141a. URL https://www.nature.com/articles/416141a. Publisher: Nature Publishing Group.

Ismael Seáñez and Ferdinando A. Mussa-Ivaldi. A body-machine interface for the control of a 2d cursor. *IEEE International Conference on Rehabilitation Robotics*, 2013:6650508,

2013. ISSN 1945-7898. doi: 10.1109/ICORR.2013.6650508. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4353481/.

Maryam M Shanechi, Amy L Orsborn, and Jose M Carmena. Robust brain-machine interface design using optimal feedback control modeling and adaptive point process filtering. *PLoS computational biology*, 12(4):e1004730, 2016.

Maryam M. Shanechi, Amy L. Orsborn, Helene G. Moorman, Suraj Gowda, Siddharth Dangi, and Jose M. Carmena. Rapid control and feedback rates enhance neuroprosthetic control. *Nature Communications*, 2017. doi: 10.1038/ncomms13825.

Krishna V. Shenoy and Jose M. Carmena. Combining decoder design and neural adaptation in brain-machine interfaces. *Neuron*, 2014. doi: 10.1016/j.neuron.2014.08.038.

Daniel B. Silversmith, Reza Abiri, Nicholas F. Hardy, Nikhilesh Natraj, Adelyn Tu-Chan, Edward F. Chang, and Karunesh Ganguly. Plug-and-play control of a brain–computer interface through neural map stabilization. *Nature Biotechnology*, 2020. doi: 10.1038/s41587-020-0662-5.

Dawn M. Taylor, Stephen I. Helms Tillery, and Andrew B. Schwartz. Direct Cortical Control of 3D Neuroprosthetic Devices. *Science*, 296(5574):1829–1832, June 2002. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1070291. URL https://science.sciencemag.org/content/296/5574/1829. Publisher: American Association for the Advancement of Science Section: Research Article.

Scott V Taylor and Aldo A Faisal. Does the cost function of human motor control depend on the internal metabolic state? *BMC Neuroscience*, 12(Suppl 1):P99, July 2011. ISSN 1471-2202. doi: 10.1186/1471-2202-12-S1-P99. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3240571/.

Emanuel Todorov and Michael I. Jordan. Optimal feedback control as a theory of motor coordination. *Nature Neuroscience*, 5(11):1226–1235, November 2002. ISSN 1546-1726.

doi: 10.1038/nn963. URL https://www.nature.com/articles/nn963. Number: 11 Publisher: Nature Publishing Group.

Francis R. Willett, Chethan Pandarinath, Beata Jarosiewicz, Brian A. Murphy, William D. Memberg, Christine H. Blabe, Jad Saab, Benjamin L. Walter, Jennifer A. Sweet, Jonathan P. Miller, Jaimie M. Henderson, Krishna V. Shenoy, John D. Simeral, Leigh R. Hochberg, Robert F. Kirsch, and A. Bolu Ajiboye. Feedback control policies employed by people using intracortical brain–computer interfaces. *Journal of Neural Engineering*, 14(1):016001, 2016. ISSN 1741-2552. doi: 10.1088/1741-2560/14/1/016001. URL https://doi.org/10.1088/1741-2560/14/1/016001.

Francis R. Willett, Daniel R. Young, Brian A. Murphy, William D. Memberg, Christine H. Blabe, Chethan Pandarinath, Sergey D. Stavisky, Paymon Rezaii, Jad Saab, Benjamin L. Walter, Jennifer A. Sweet, Jonathan P. Miller, Jaimie M. Henderson, Krishna V. Shenoy, John D. Simeral, Beata Jarosiewicz, Leigh R. Hochberg, Robert F. Kirsch, and A. Bolu Ajiboye. Principled BCI Decoder Design and Parameter Selection Using a Feedback Control Model. *Scientific Reports*, 9(1):8881, June 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-44166-7. URL https://www.nature.com/articles/s41598-019-44166-7. Number: 1 Publisher: Nature Publishing Group.

Francis R. Willett, Donald T. Avansino, Leigh R. Hochberg, Jaimie M. Henderson, and Krishna V. Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, pages 249–254, 2021. doi: 10.1038/s41586-021-03506-2.

Matthew S. Willsey, Samuel R. Nason-Tomaszewski, Scott R. Ensel, Hisham Temmar, Matthew J. Mender, Joseph T. Costello, Parag G. Patil, and Cynthia A. Chestek. Real-time brain-machine interface in non-human primates achieves high-velocity prosthetic finger movements using a shallow feedforward neural network decoder. *Nature Communications*, 13(1):6899, November 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-34452-w.

Lahiru N. Wimalasena, Jonas F. Braun, Mohammad Reza Keshtkaran, David Hofmann, Juan Álvaro Gallego, Cristiano Alessandro, Matthew C. Tresch, Lee E. Miller, and Chethan Pandarinath. Estimating muscle activation from EMG using deep learning-based dynamical systems models. *Journal of Neural Engineering*, 19(3):036013, May 2022. ISSN 1741-2552. doi: 10.1088/1741-2552/ac6369. URL https://dx.doi.org/10.1088/1741-2552/ac6369. Publisher: IOP Publishing.

B Wodlinger, J E Downey, E C Tyler-Kabara, A B Schwartz, M L Boninger, and J L Collinger. Ten-dimensional anthropomorphic arm control in a human brain-machine interface: difficulties, solutions, and limitations. *Journal of neural engineering*, 12(1):016011, February 2015. doi: 10.1088/1741-2560/12/1/016011.

Momona Yamagami, Keshia M Peters, Ivana Milovanovic, Irene Kuang, Zeyu Yang, Nanshu Lu, and Katherine M Steele. Assessment of dry epidermal electrodes for long-term electromyography measurements. *Sensors*, 18(4):1269, 2018.

Momona Yamagami, Katherine M. Steele, and Samuel A. Burden. Decoding Intent With Control Theory: Comparing Muscle Versus Manual Interface Performance. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–12, Honolulu HI USA, April 2020. ACM. ISBN 978-1-4503-6708-0. doi: 10.1145/3313831.3376224. URL https://dl.acm.org/doi/10.1145/3313831.3376224.

Momona Yamagami, Lauren N. Peterson, Darrin Howell, Eatai Roth, and Samuel A. Burden. Effect of handedness on learned controllers and sensorimotor noise during trajectory-tracking. *IEEE Transactions on Cybernetics*, pages 1–12, 2021. ISSN 2168-2275. doi: 10.1109/TCYB.2021.3110187. Conference Name: IEEE Transactions on Cybernetics.

Momona Yamagami, Alexandra A Portnova-Fahreeva, Junhan Kong, Jacob O. Wobbrock, and Jennifer Mankoff. How do people with limited movement personalize upper-body gestures? considerations for the design of personalized and accessible gesture interfaces. In *The 25th International ACM SIGACCESS Conference on Computers and Accessibility*, pages

1–15. ACM, 2023. ISBN 9798400702204. doi: 10.1145/3597638.3608430. URL https://dl.acm.org/doi/10.1145/3597638.3608430.

Christopher S Yang, Noah J Cowan, and Adrian M Haith. De novo learning versus adaptation of continuous control in a manual tracking task. *eLife*, 10, June 2021. ISSN 2050-084X. doi: 10.7554/eLife.62578. URL https://doi.org/10.7554/eLife.62578. Publisher: eLife Sciences Publications, Ltd.

Rui Zhang, Fali Li, Tao Zhang, Dezhong Yao, and Peng Xu. Subject inefficiency phenomenon of motor imagery brain-computer interface: Influence factors and potential solutions. *Brain Science Advances*, 6(3):224–241, 2020. ISSN 2096-5958. doi: 10.26599/BSA.2020.9050021.

Xingye Zhang, Shaoqian Wang, Jesse B. Hoagg, and T. Michael Seigler. The roles of feedback and feedforward as humans learn to control unknown dynamic systems. *IEEE transactions on cybernetics*, 48(2):543–555, 2018. ISSN 2168-2275. doi: 10.1109/TCYB.2016.2646483.

Yin Zhang and Steven M. Chase. Recasting brain-machine interface design from a physical control system perspective. *Journal of Computational Neuroscience*, 39(2):107–118, October 2015. ISSN 0929-5313, 1573-6873. doi: 10.1007/s10827-015-0566-4. URL http://link.springer.com/10.1007/s10827-015-0566-4.

Karl J. Åström and Richard M. Murray. *Feedback systems: an introduction for scientists and engineers*. Princeton University Press, Princeton, 2008. ISBN 978-0-691-13576-2.