NORTHWESTERN UNIVERSITY

# Human-Machine Communication in Assistive and Rehabilitation Robotics

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Mechanical Engineering

By

Aleksandra Ola Kalinowska

EVANSTON, ILLINOIS

September 2023

# ABSTRACT

Human-Machine Communication in Assistive and Rehabilitation Robotics

Aleksandra Ola Kalinowska

Assistive robots have the potential to support motor recovery after an injury as well as to alleviate the burden of physical tasks for both impaired and able-bodied individuals. Over the last few decades assistive robots have become increasingly more capable. Many can now provide task-specific assistance and can be safe enough to physically interact with people, but they are still incapable of versatile collaboration.

While effective assistance requires robots to adapt to and infer intent from their human partners, adaptation and intent inference are particularly challenging when the tasks are novel and/or dynamic. As a result, existing robots have difficulty engaging in activities that involve dynamic motion, such as catching a falling object, as well as tasks that are unfamiliar to the robot. One of the main contributing factors are the limited communication capabilities available to the human-robot pair. Unlike a pair of people, the human-robot pair does not have the ability to communicate flexibly to coordinate interaction, so the robot often relies on passively inferring intent from human movement or interpreting inputs from a joystick-like interface. There is a need for flexible and comprehensive languages for human-machine communication.

In this thesis, I present mathematical formulations and algorithmic tools that enable the robot to interpret dynamic motion as a communication signal in the context of a task. I begin by characterizing metrics of dynamic performance and show how they can be used to study post-stroke impairments. I then introduce methods that enable human-robot teams to co-create communication protocols from interaction. Lastly, I design algorithms that parse demonstrations of dynamic tasks to learn task definitions that enable the robot to recreate a new dynamic skill. In each case, I show how the algorithmic tools can improve robotic assistance by testing them on robots interacting with people. I conclude this thesis with a discussion of future research directions that will help facilitate flexible human-machine communication for versatile assistance and rehabilitation.

# Acknowledgements

First and foremost, I would like to thank my advisor Todd Murphey. Thank you for your willingness to always find time to listen, discuss, inspire, and provide feedback, for your contagious enthusiasm for research, and for your advice, both solicited and unsolicited, about anything from research ideas to personal decisions. You have emboldened me to define and pursue my own research questions and given me the (sometimes frustrating) freedom to start figuring out how to do it. Your unwavering support and mentorship are greatly appreciated.

I would also like to thank my committee members: Jules Dewald and Brenna Argall. Big thank you to Jules, for enabling me to work with stroke survivors, who have been the biggest supporters of our research and incredible teachers of gratitude and humility, and for enabling me to learn from the field of movement sciences. Thank you, Brenna, for pushing me to think about the bigger picture by asking tough but thoughtful questions, and for being a role model both inside and outside the lab.

To my research colleagues and collaborators, my PhD would not have been the same without you. Thank you for being willing to question my assumptions, for helping me find errors in my code and in my thinking, and for being there to share both the excitement of new results and the woe of things not working. Special thank you to Milli Schlafly, without you I would have given up on quantifying motion bandwidth long before there were enough results to write a paper, as well as to Kyra, Tommy, Ahalya, Katie, and Elnaz for your willingness to engage in interesting new ideas and co-design experiments to explore if they work.

To my lab mates, thank you for your kindness, intelligence, and friendships. I will miss the mid-afternoon conversations which often started as short thoughts about research and ended as a discussion about books, politics, or the philosophy of life—they are some of my fondest memories from my PhD. Thank you to Ana Pervan and Hannah Janssen (an honorary lab mate), for being my friends and PhD companions from the very first day until (hopefully) long after the last.

To my family and my support crew, both in Poland and in the Chicagoland area. Thank you for motivating me to pursue a PhD in the first place, for being there for me along the way, and for always encouraging me to be an independent thinker, in addition to being a proficient engineer and scientist. I wouldn't be here without you.

And lastly, thank you to Korbinian Riener. Thank you for choosing to call Chicago home for the last few years. Thank you for your continuous support and encouragement, for the mostly unwavering patience, and for your consistent (even though sometimes unsuccessful) attempts to put my PhD struggles into perspective. I am endlessly grateful and could not be happier for having you by my side.

I am truly fortunate to have had such remarkable individuals in my life during the PhD, and long before it. Even though the dissertation marks the end of a chapter, I hope for it to be only the beginning.

# Table of Contents

# List of Tables

# List of Figures

CHAPTER 1

# Introduction

Assistive robots hold promise of being versatile collaborative companions—helping with daily tasks, with manually intensive chores or with learning and re-learning motor skills after an injury. Many existing robots are already capable of task-specific collaboration. To provide assistance, these robots are designed to interpret and anticipate their human partners' actions in the context of a pre-specified task, and to coordinate their behaviors based on the inferred intent of the human partner. This works well for known quasistatic tasks, but is particularly difficult to do during dynamic tasks that are sensitive to the timing of coordinated motion, or novel tasks that had previously not been encountered by the human-robot pair. Dynamic tasks necessitate the robot to be able to interpret dynamic motion and anticipate motion dynamics. Novel tasks rely on the human-robot pair's ability to quickly and efficiently coordinate joint action and learn from new experiences.

During most tasks, people move quasistatically—they use overdamped motion making their movement trajectories reasonably predictable in the context of a task. Reaching for an object or sitting down onto a chair are examples of a quasistatic activity. However, time pressure in a task or the task dynamics can lead to underdamped, dynamic motion that is much more difficult to predict. As an example, a person might compensate for tripping on a curb to maintain balance, or they might reach to catch a moving object to prevent it from falling. Dynamic response is an inevitable component of everyday activities and, consequently, of versatile human-robot interaction. It is therefore necessary for an assistive robot to reason about dynamic motion and to be able to flexibly communicate with a person about a coordinated dynamic response of the human-robot system.

At present, collaborative robots—even those designed for working with people—have limited capabilities to communicate with their human partners. In many assistive applications, robots rely on teleoperation, where the robot is directly guided by the commands of a human user, possibly with some autonomous capabilities. Examples include powered wheelchairs or surgical robot arms. Teleoperated robots can be designed to provide feedback to the human operator, e.g., through a screen-based display or through haptics that mimic touch sensations at the robot's end-effector. These communication channels improve performance of the human-robot team. However, existing collaborative robots still lack the capability to flexibly engage in bidirectional communication with a human partner, limiting their collaborative potential.

Being a *versatile* collaborative companion requires a robot to safely and effectively interact with people across an assortment of daily activities. While recently there has been a lot of progress in creating large language models, such as chatGPT, capable of versatile verbal dialogue, such communication is often too slow and too cognitively taxing for coordinating joint action during physical human-robot interaction (pHRI). There is a need for human-robot pairs to be able to develop *flexible*, *nonverbal* communication protocols. In this thesis, I design algorithmic approaches for human-machine communication that enable robotic assistance during tasks that are dynamic and/or novel.

## 1.1. Main Contributions

This thesis develops algorithmic approaches for human-robot communication that anticipate dynamic responses during task-oriented pHRI. Specifically, it explores three facets of pHRI. In

Chapters 2 and 3, I develop mathematical specifications of dynamic motion and evaluate their ability to characterize dynamic deficit due to a neuromotor disorder, such as stroke. In Chapter 4, I create robot policies capable of co-creating communicative conventions with people from pHRI—enabling flexibility in coordinating joint actions. In Chapters 5 and 6, I define task representations that can be learned from task demonstrations and used to provide robotic assistance during dynamic tasks. Throughout this thesis, I introduce novel algorithmic formulations, implement them in simulation and/or hardware, and validate them experimentally in studies with human participants. The goal of the thesis is to design and validate control methods for pHRI that enable the robot to reason about a person's task-oriented dynamic response and to reliably adapt its behavior during tasks involving dynamic motion.

### 1.1.1. Robot-assisted assessment of dynamic motion

Robots that provide physical assistance for rehabilitation or during daily tasks depend on their ability to quantitatively assess movement. While many activities of daily living (ADLs) require dynamic motion—fast, controlled, and timing-sensitive movements where the continuous modulation of motor activity is required to respond to real-time stimuli—existing assessments focus on quantifying static and quasi-static characteristics of motion, such as strength and reaching range. As a result, existing metrics of motion often do not capture the functional limitations that manifest themselves during ADLs, hindering progress in robotic assistance and rehabilitation. There is an unmet need for a mathematical specification of dynamic motion which would enable human-robot communication about the timing and level of required assistance during dynamic tasks.

In this work, I develop virtual dynamic tasks and design frequency-based metrics of motion that quantify motion bandwidth. I show that the frequency-based metrics can be used to assess dynamic performance with different types of feedback in unimpaired individuals as well as to detect impairment across a range of tasks in chronic survivors of stroke.

**The contributions of this research effort are as follows:**

(1) I design robot-assisted virtual tasks (i.e., the ball-in-bowl task and the nail-and-hammer task) that require dynamic motion in the upper limb. I develop metrics that assess dynamic capabilities by quantifying the frequency content of movement during these tasks. I refer to this approach as QADR—Quantitative Assessment of Dynamic Response.

(2) I validate QADR in a study with able-bodied individuals (n=6). Results show that the ball-in-bowl task (a virtual task with a resonant frequency) can be used to quantify a person's motion bandwidth. Haptic feedback improves the robot's ability to elicit a dynamic response at a specific frequency compared to visual-only feedback.

(3) I employ QADR to characterize dynamic performance in chronic survivors of hemiparetic stroke (n=13+48). I show that a frequency-based metric of motion generalizes across three dynamic tasks, in many aspects similar to activities of daily living: the ball-in-bowl task, the nail-and-hammer task, as well as an object hit task from prior work [**183**].

(4) I find that stroke causes a significant decline in dynamic response at frequencies above 1.5Hz in the paretic upper limb, with the degree of functional loss dependent on the clinically assessed severity of impairment. Results suggest that one of the underlying

mechanistic changes that cause a reduction in motion bandwidth is a shift in reliance from corticospinal and corticobulbar pathways to corticoreticulospinal pathways.

In this work, I led the conceptualization and experimental validation of a novel approach to quantify and interpret dynamic motion. I would like to acknowledge Millicent Schalfly (MS), Kyra Rudy (KR), and Michael Doody (MD) for their contributions to this work. MS designed and ran experiments, conducted data analysis, visualized and interpreted results, and supported the writing of the final manuscripts. KR implemented the initial version of the ball-in-bowl task, as well as helped with designing and running experiments, interpreting results, and writing the final manuscripts. MD created an engaging visualization for the ball-in-bowl task that was used in the final study with post-stroke individuals. This work was published in [**98**] and [**101**]. A final publication is in preparation for the Journal of NeuroEngineering and Rehabilitation [**100**].

### 1.1.2. Emergence of communicative conventions

The vocabulary and communicative conventions necessary for human-machine communication are most often defined and encoded in the robot prior to use. Sometimes the algorithmic setup allows for initial customization of the user-generated vocabulary, e.g., tailoring movement-based commands to the physical capabilities and limb size of the person. Even so, the burden falls on the human operator to learn to generate and interpret the communicative signals prior to human-robot interaction. As an example, a person might learn joystick inputs (i.e., forward, back, left, right tilt of the joystick) to command a powered wheelchair to move in the corresponding directions. In the case of a directly controlled powered wheelchair, there is an intuitive way to specify the mapping of a continuous 2D control space onto the analogous 2D action space of the robot, making the language easy to learn. However, if a person has to use shoulder shrugs to control movements of a robot arm or sips and puffs to maneuver a wheelchair because of limb loss or paralysis, the mapping is no longer as intuitive. Enabling the communicative conventions to emerge from interaction, similar to how verbal languages have emerged between people, can help to make the symbols both more functionally optimal and easier to learn.

Communication emergence is studied extensively in people and, more recently, in artificial agents. It is often investigated using the Lewis signaling game, where a speaker and a listener rely on communication to solve an object identification task. In this task, the communication is usually free and guaranteed to the agents. As a result, artificial agents who learn to exchange information in this environment are not incentivized to do so in a concise and efficient manner. In contrast, people try to be as informative as possible, and give only as much information as is needed. If artificial agents are to communicate with people, they should be able to learn concise communication protocols.

In this work, I design artificial agents that can use reinforcement learning (RL) to learn how to communicate with other agents in a concise manner. I implement agents, similar to a collaborative mobile robot, that learn to navigate a simulated maze-like environment. I test the agents' behavior during task-oriented interaction with other agents, and with people (n=2). My findings serve as a proof-of-concept that robots can be designed to co-create communicative conventions during interaction with people. Deep RL is a potential algorithmic approach that enables communication emergence, and—with an appropriate reward design and training structure—can enable emergence of concise communication and on a timescale feasible for people.

**The contributions of this research effort are as follows:**

(1) I create artificial agents, modelled using neural networks and RL, that learn how to communicate during task-oriented collaboration. I develop a simulated cooperative navigation task that serves as a testbed for communication emergence.

(2) I introduce situated communication in deep RL. I design agents to develop a policy that reasons about taking both communicative and noncommunicative actions, in contrast to agents reasoning over communicative and noncommunicative actions separately. I find that situated communication enables concise communication, in line with the Gricean maxim of quantity.

(3) I compare situated communication against other forms of incentivizing conciseness and find that, with situated communication, agents converge on a shared communication protocol more quickly, while learning to exchange information most sparingly.

(4) I show that an agent designed using neural networks and supervised learning can learn symbolic vocabulary from physical interfaces, such as a sip-n-puff, operated by a person. I find that with just a handful of demonstrations per symbol, the agent can learn to recognize symbols from continuous signal, even if the symbols are temporally distorted compared to the original demonstrations.

In this work, I conceptualized and experimentally validated a collaborative artificial agent that can learn to communicate concisely through task-oriented interaction. I tested a similar artificial agent to learn vocabulary from a human-operated interface, i.e., the sip-n-puff, to navigate a simulated wheelchair. I would like to acknowledge Elnaz Davoodi (ED), Florian Strub (FS), Kory Mathewson (KW), Ivana Kajic (IK), Michael Bowling (MB), and Marshall Johnson (MJ) for their contributions to the project. ED supported the conceptualization of the project, its implementation, the interpretation of results, and the writing of the final manuscripts. FS, KW, IK, and MB provided feedback on the experimental setup, interpretation of results, and final writing of the manuscript. MJ implemented the wheelchair simulation with human input and ran initial tests (n=2). This work was published in [**93**]. A final publication is under revision at the Journal of Artificial Intelligence Research [**94**].

### 1.1.3. Learning of task representations

Most of human-robot communication and assistance takes place in the context of a task. As such, it is beneficial for the robot to be able to flexibly build and grow a library of tasks during use. Learning from demonstrations (LfD), or imitation learning, are categories of algorithms that enable users to teach robots new tasks by providing task demonstrations. In these methods, robots learn a task representation—in the form of a task trajectory, reward function, or task policy—by extracting salient information from demonstrated task executions.

It is often challenging for the users to provide high-quality task demonstrations, because the task itself could be challenging to perform (e.g., cart-pole inversion due to unintuitive dynamics) or because the low-dimensional control interface may be limiting for the users (e.g., joysticks or planar keyboard controllers for 7-DoF robot arms). These challenges only grow when the user suffers from a neuromotor impairment, because the individual is limited in the modalities they can use for communication and the generated signal is often more noisy. Similarly, dynamic tasks are often difficult to demonstrate, because the person has to account for robot dynamics as well as the timing of a robot's actions when providing demonstrations—these characteristics of robot motion tend to be unintuitive and difficult to control, especially for novice users.

In this work, I propose two methods for learning task representations from human demonstrations. The first method enables learning subtasks from continuous demonstrations of a complex task. The second method enables learning subtask definitions from imperfect and failed demonstrations. Importantly, both approaches avoid defining tasks in terms of optimal movement trajectories, but rather focus on capturing movement dynamics and task statistics, respectively.

**The contributions of this research effort are as follows:**

(1) I develop a dynamics-based clustering algorithm for creating symbolic representations of sequential tasks from continuous task demonstrations.

(2) I validate the segmentation approach for the task of walking and show that the learned symbolic representation can be used in real-time to provide task-based assistance.

(3) I create an algorithm, called ergodic imitation, for learning robust task definitions without the need for near-optimal demonstrations. In this framework, I represent tasks as feature-embedded distributions and show that negative demonstrations——demonstrations of what not to do——can be incorporated into creating a task definition.

(4) I demonstrate the efficacy of ergodic imitation in a human subject study (n=24), where novice participants are asked to define the task of inverting a cart-pendulum. Ergodic imitation helps novices create effective task definitions; negative demonstrations improve task learning. With data from one expert participant, I show that ergodic imitation works with simulated tasks of target reaching and table cleaning with a 7-degree-of-freedom robot arm.

In this work, I conceptualized and experimentally validated algorithmic approaches for learning task representations from human demonstrations. I would like to acknowledge Thomas Berrueta (TB), Ahalya Prabhakar (AP), Adam Zoss (AZ), and Kathleen Fitzsimons (KF) for their contributions. TB implemented the algorithm for segmenting continuous motion trajectories, and contributed to writing the final manuscript. AP conducted robot-arm experiments to validate ergodic imitation, analyzed the experimental data, and contributed to writing the final manuscript. KF provided feedback on the design of ergodic imitation and edited the final manuscript. AZ helped with the conceptualization of the project and collected walking data using the Ekso Bionics exoskeleton. This work was published in [**92**] and [**97**].

### 1.1.4. Embodied Communication

Early research on physical human-robot interaction (pHRI) had necessarily focused on device design—the creation of compliant and sensorized hardware, such as exoskeletons, prostheses, and robot arms, that enabled people to safely come in contact with robotic systems and to communicate about their collaborative intent. As hardware capabilities have become sufficient for many applications, and as computing has become more powerful, algorithms that support fluent and expressive use of pHRI systems have begun to play a more prominent role in determining the systems' usefulness. In a recent review article [**96**], I summarize the field of algorithms for human-machine communication. I lay out current state-of-the-art algorithmic approaches and existing application domains, as well as discuss unmet needs and future directions for the field. In the review, I describe a selection of representative algorithmic approaches that regulate and interpret communication in pHRI. I discuss the progression from algorithms based on physical analogies, such as admittance control, to computational methods based on higher-level reasoning, which take advantage of multimodal communication channels.

Throughout the review, I emphasize the need for new algorithms that will facilitate versatile human-machine communication. I describe the need for research on communication emergence, such as the work in Chapter 4. I also note a need for quantitative metrics that assess and interpret task-oriented motion trajectories, particularly when the motion is dynamic. Much of my thesis work aims to address this gap. While existing algorithmic approaches largely enable task-specific pHRI, they do not generalize to versatile human–robot collaboration. In the review and in my discussion of next steps in Chapter 7, I therefore propose focusing research efforts on facilitating emergent embodied dialogue——bidirectional, multimodal communication that can be learned through continuous human-robot interaction.

## 1.2. Thesis Outline

Chapter 2 will introduce a quantitative assessment method of dynamic response (QADR). It will present results from a human subject study with able-bodied individuals, measuring the interaction bandwidth of a human-machine system. Chapter 3 will discuss using QADR to assess motion bandwidth in chronic survivors of hemiparetic stroke. It will present a quantitative analysis of post-stroke dynamic deficit as well as empirical evidence for its potential causes. Chapter 4 will present a method for learning communicative conventions through task-oriented interactions. It will discuss results from a simulation study between two artificial agents as well as show preliminary evidence that the approach would be feasible for human-machine interaction. Lastly, Chapters 5 and 6 will discuss learning task representations from human demonstrations of tasks, such as walking in a lower-limb exoskeleton or cleaning a table with a robot arm. Chapter 5 will describe an approach that facilitates contrastive learning based on task statistics. Chapter 6 will introduce an algorithm for learning substasks from continuous task demonstrations without supervision or *a priori* assumptions about the task structure. In Chapter 7, I summarize my contributions and propose future research directions.

CHAPTER 2

# Measuring motion bandwidth using pHRI

An assistive robot can augment human performance by providing physical assistance or motion guidance. In human-robot collaboration, it is important to know the bandwidth of an individual's ability to generate motion in response to external stimuli, such as visual or haptic cuing. This becomes particularly relevant in timing-sensitive tasks, such as walking or catching a falling object. In this work, we propose a frequency-based assessment of motion that enables us to measure the bandwidth of physical human-robot interaction (pHRI)—quantifying how fast individuals can respond to stimuli on a continuous basis. We introduce a robot-assisted virtual dynamic task with a tunable resonant frequency. A human subject study with seven participants shows that our task can elicit a dynamic response in a participant at frequencies of 0.5Hz, 1Hz, 1.5Hz and 2.5Hz at the arm. Using the virtual task, we test whether haptic cues improve motion timing. At all tested frequencies, we find that haptic stimuli help guide timing of dynamic movement and improve performance compared to visual-only cuing. By quantifying the interaction bandwidth for other pHRI systems—particularly when the human collaborators have neuromotor impairments—our method can help assistive robots adapt to an individual. Moreover, our results highlight the importance of incorporating haptic feedback into pHRI for dynamic tasks—haptics can provide guidance around motion timing, such as in assistive robots used for assessment and physical rehabilitation.

## 2.1. Introduction

Physically interacting robots, such as an exoskeleton, can be useful in a number of ways: they can alleviate physical burden during manual tasks, help with learning or re-learning motor skills, or provide precise assessment of physical abilities. Although many relevant motor skills and daily tasks are highly dynamic, assistive robots largely operate in a static or quasi-static regime. As an example, robot-assisted assessment of motor skills typically uses tasks such as defining reachable workspace [**64**, **178**] or tracking a pre-defined trajectory [**187**]. However, recent work shows interest in assessing motor coordination through dynamic tasks, e.g., a timing-sensitive hitting task with moving objects [**183**], because ability to generate dynamic movements plays an important role in one's ability to complete activities of daily living.

When the human-robot interaction involves dynamic movements, the human-robot pair benefits from continuous communication [**166**, **167**]. Processing times of communicative cues and the ability to generate a motor response may vary between populations: younger vs. older individuals [**53**], able-bodied vs. physically impaired [**40**]. To increase collaborative success, it would be beneficial to understand the bandwidth of an individual to react to stimuli and generate motion as a response. Our prior work attempted to assess motor coordination during dynamic tasks post-stroke [**98**]. Here, we build on this work and propose a method for measuring interaction bandwidth during physical human-robot interaction (pHRI).

To measure interaction bandwidth, we use a virtual ball-in-bowl task, inspired by the real-world task of quickly moving a cup of water without spilling. The ball-in-bowl task has a specified resonant frequency (dependent on the size of the bowl), which defines the default movement frequency that is needed to succeed at the task. Because of this property, we can learn about an individual's bandwidth from the frequency content in their motion. Unlike in existing assessments, we do not assess motion quality through jerk [**8**, **155**], error [**187**] or time to motion initiation [**40**], but rather we look at the frequency decomposition of motion. In particular, we quantify the energy exerted by an individual during task completion around task resonance. Evaluating frequency content in motion during the ball-in-bowl task (a dynamic task with a resonant frequency) while a person is physically coupled with a robot allows us to quantify the individual's interaction bandwidth with the robotic system.

Prior work shows that real-time perception and processing of sensory information is crucial for effective motor coordination [**73**]. While motor coordination is controlled by both visual and proprioceptive feedback loops [**186**], precise movements can overwhelm the visual sensory channel—motor tasks that demand a high degree of accuracy are consequently more dependent on kinesthetic information [**199**]. Thus, robot-mediated haptic feedback should be expected to improve human performance, particularly in highly dynamic tasks that require accuracy with respect to motion timing. We use our approach to study one's ability to generate dynamic motion when relying on two different modalities for sensory feedback: visual and haptic cuing.

There are a number of studies that show the value of haptic feedback in human-robot interaction. It is well-established that haptic feedback is beneficial for teleoperation in quasistatic/non-dynamic settings, such as robotic surgery [**147**, **186**], a peg-in-hole insertion task [**141**] or a pick-and-place task [**165**], particularly in a cluttered environment [**14**]. Moreover, there is recent research that explicitly studies robotic assistance with haptic feedback during dynamic tasks. Ozen et al. found that training with haptic feedback enhances motor learning for the task of inverting a virtual pendulum [**140**]. Other work has shown positive impact of vibrotactile feedback on task performance during balancing an inverted pendulum [**184**] or while using a teleoperated robot to balance an object on a tray [**166**]. While prior work shows that haptic feedback is helpful in improving joint human-robot performance, no study to date has shown *why* haptic feedback improves performance. In this work, we show that haptic stimuli improve people's timing accuracy across a range of motion frequencies. Improved motion timing is likely a significant contributor to an increase in overall task performance during human-robot collaborative tasks.

In summary, we contribute a frequency-based method for assessing interaction bandwidth—individuals' ability to continuously generate a dynamic response (controlled and timing-sensitive movements in response to real-time stimuli). In a robot-assisted virtual environment, we assess dynamic response under two feedback conditions: visual and combined visuo-haptic cuing. We test the task on a group of able-bodied individuals (n=7). Through our experiments, we show that:

- Using the ball-in-bowl task—a virtual task with resonance—we can elicit motion at a specific frequency.
- Frequency content in motion during this dynamic task can quantify interaction bandwidth.
- Haptic feedback improves motion timing.

Given the validation study presented here, our method can be used to assess motion bandwidth in individuals with neuromotor impairments as well as to inform the design of physically coupled human-robot systems.

## 2.2. Experimental setup

We have developed a virtual task that elicits controlled, timing-sensitive movements. In our experimental setup, we couple the virtual environment with a stationary upper-limb exoskeleton, capable of rendering haptic feedback and translating forearm motion into activity in the virtual task (see Fig. 2.1). Using this setup, we conduct a human subject study with seven able-bodied individuals.

### 2.2.1. The physical environment

A haptic environment is created using the Arm Coordination Training 3-degree-of-freedom device (ACT-3D) [**98**]. The ACT-3D is the combination of an admittance controlled HapticMASTER robot, a 6-degree-of-freedom load cell (JR3 load cell, Woodland, CA) at the robot's end-effector, and a Biodex chair. The participant's forearm is attached to the load cell using a forearm-wrist-hand orthosis. This setup, as shown in Fig. 2.1, allows the participant to directly control the location of the end-effector with movements of their arm. Active movements are captured as force readings on the load cell and, using the robot's internal controller, translated into movements of the end-effector in 3D space.

The robot is able to provide partial or full support of the arm against gravity by modulating abduction loading at the shoulder. It is also able to render haptic objects and forces in the upper-limb workspace of the individual. The forces can be updated in real-time. Given that people can process haptic feedback at a temporal resolution up to 20Hz [**26**], we chose 20Hz as our update rate.

### 2.2.2. The virtual environment

A virtual task is visualized on a screen in front of the participant. We use a ball-in-bowl task, loosely inspired by the real-world activity of moving a cup of water without spilling. The task was introduced in our prior work [**98**] and modified for this study to incentivize dynamic

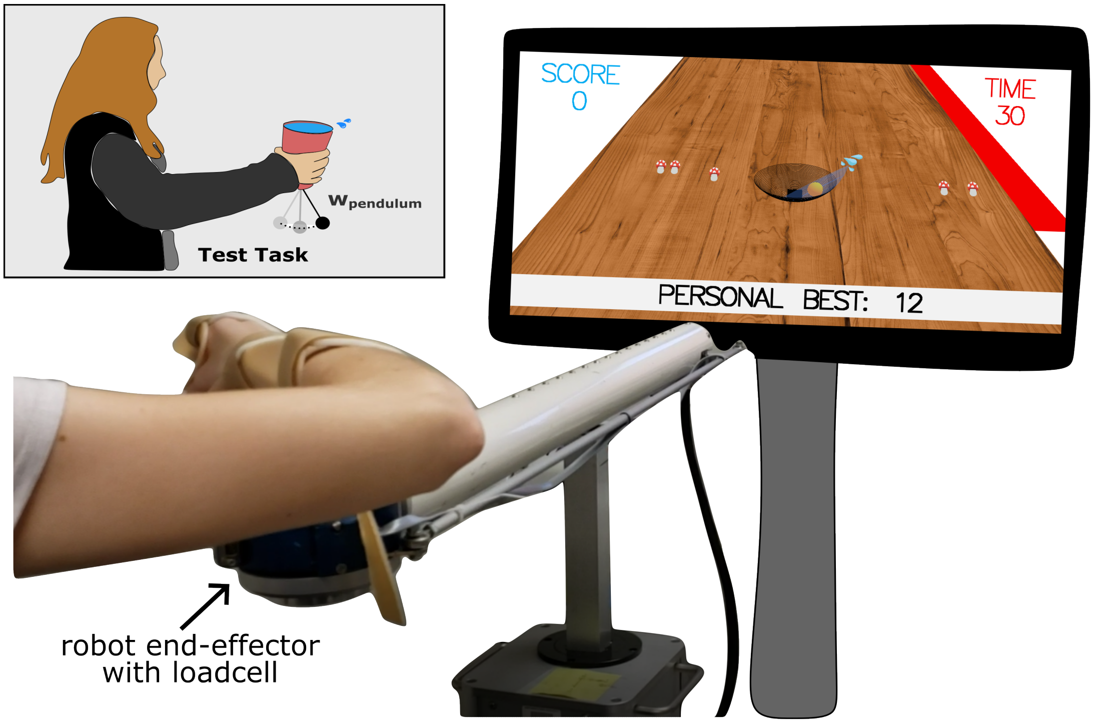

Figure 2.1. **Experimental setup and ball-in-bowl task.** In the top left, we illustrate the real-world task of carrying a cup of water without spilling that served as inspiration for the ball-in-bowl task. In the bottom right, we visualize the experimental setup with the robot.

motion. The current version of the task has been open-sourced and is available online under the MIT license [**99**].

In the ball-in-bowl task, the location of the robot end-effector is mapped to the location of a bowl on a virtual table. There is a virtual ball rolling around inside the virtual bowl and the participant experiences haptic feedback corresponding to the interaction force between the bowl and the ball. There is no friction in the simulated ball-in-bowl system—consequently only normal forces are rendered haptically to the user. While participants exert forces in a range of values throughout a trial (rarely exceeding $10N$), the haptic force feedback is scaled and consistently rendered as a $0.5N$ vector in the $xy$-plane.

The motion of the ball is simulated using a modified dynamics model of a 3D pendulum, where the pendulum's acceleration $\ddot{\theta}$ is calculated using independent components $\ddot{\theta}_x$ and $\ddot{\theta}_y$ defined as

$$\ddot{\theta}_i = (\frac{g}{h} sin(\dot{\theta}_i) - u_i cos(\dot{\theta}_i)).$$

The variable $g = 9.81\frac{m}{s^2}$ represents the gravity constant, $h$ is the pendulum length, and $i$ indexes the coordinate (either $x$ or $y$). The variable $u_i$ represents the individual's input to the system—in our experimental setup, it is the acceleration of the end-effector in the $x$ and $y$ directions, as calculated from load cell measurements of force. When the participant moves in synchrony with the ball, they amplify the ball's oscillations and allow it to gain energy. When the participant counteracts the ball's movements, they dampen out its energy and prevent it from falling out of the bowl. There is no damping or friction in the simulated ball-in-bowl system. This design choice disincentivizes the individual from waiting for the ball to lose energy and settle on its own. Only active movements of the end-effector can affect the ball's oscillations. Unlike in a real-world 3D pendulum, $\ddot{\theta}_i$ is only influenced by $u_i$—this simplification of the system dynamics was a design choice meant to improve explainability and participants' agency over the ball's movements.

Participants can move in 3 dimensions. Their $xy$-motion is mapped directly onto the location of the virtual bowl. In the z direction, participants start from a home position $z = 0$, resting on a haptic table. During task attempts, they are asked to keep their arm lifted anywhere above the haptic table to avoid imposing dynamic constraints on the ball-in-bowl system. In turn, to prevent fatigue, their arm weight is fully supported against gravity—the load cell readout in the z direction is near zero, making the arm feel buoyant in space. Movement in the z direction does not affect the simulation.

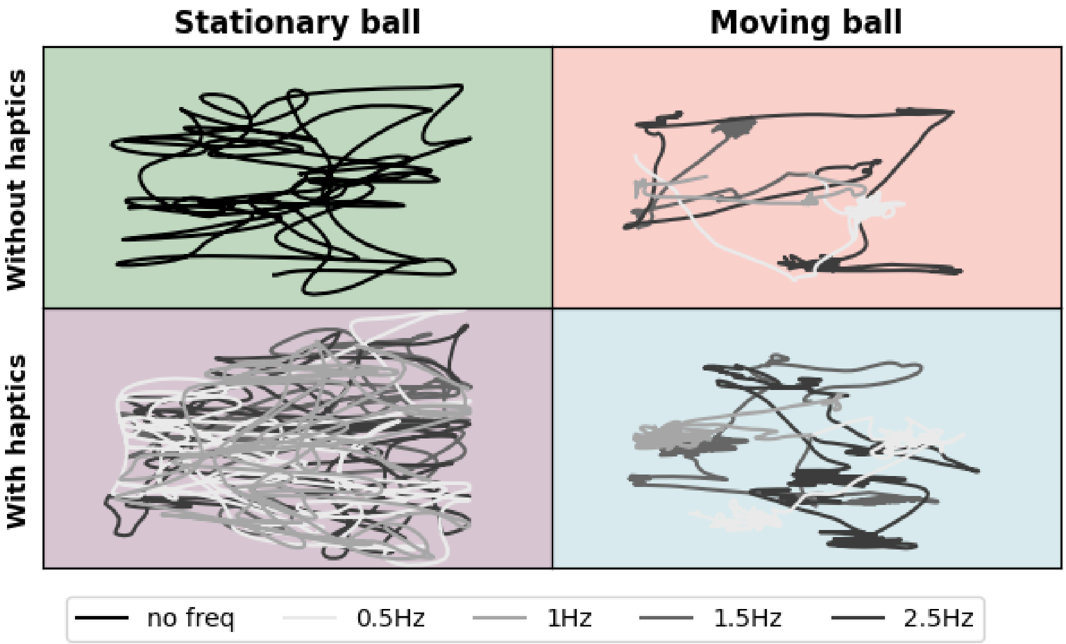

Figure 2.2. **Example trajectories across workspace (*xy*-plane) of a participant for 13 different trials.** Each line represents one 30-second trial. Note that when the ball is stationary, participants traverse the workspace frequently to collect targets. When the ball is moving, participants alternate between moving quickly between targets and trying to settle the ball.

### 2.2.3. The virtual task

The goal of the virtual task is to collect as many targets as possible within a 30-second window. There are three conditions that have to be met for an individual to be able to collect a target: (1) The individual's arm must be lifted above the haptic table, indicated to the participant by the bowl turning blue. (2) The virtual ball's total energy must be low enough to be oscillating no higher than one third of the bowl's height, indicated to the participant by the ball turning green. (3) The center of the virtual bowl must be aligned with the target location.

At any given moment there are 5 target mushrooms spread out randomly on the virtual table within the participant's reachable workspace. During each attempt at the task, participants are instructed to manipulate the location of the virtual bowl to collect as many targets as possible in

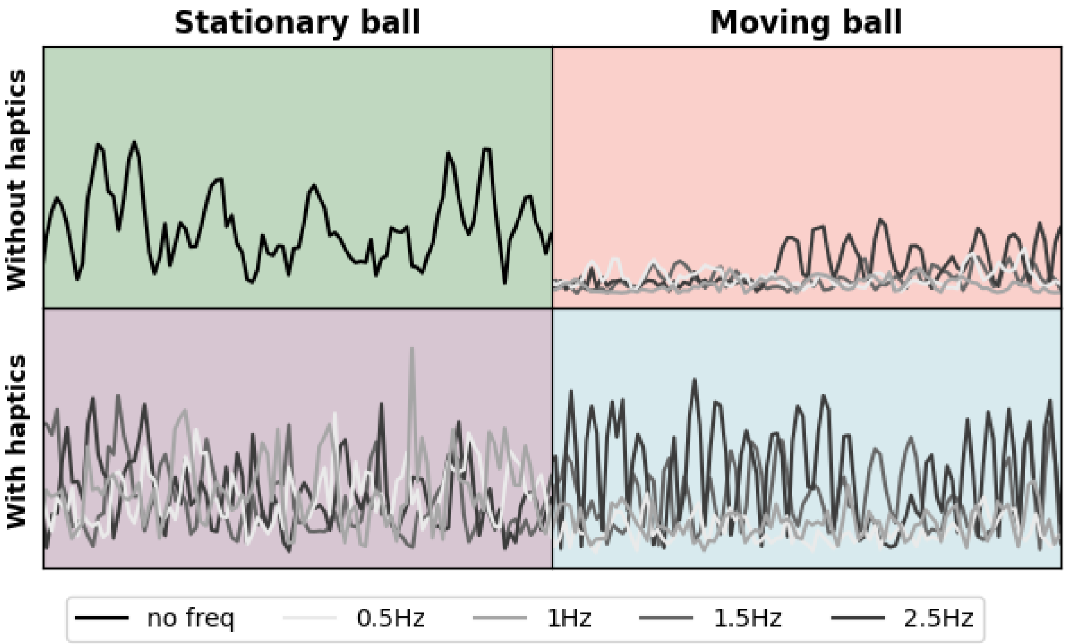

Figure 2.3. **Five seconds of time-series data of a participant's acceleration during 13 different trials of the ball-in-bowl task.** Note that the time-series data is difficult to evaluate for success and comparison between trials. As we describe in this work, frequency content in motion can be used as a more informative, time-independent quantity to assess motion quality.

30 seconds. Every time an individual collects a target, three things happen: (1) the participant receives a point, (2) a new target appears in a different location, so that 5 targets are visible at any given time, and (3) the ball is injected with a bit of energy, so that again the participant must generate movement to counteract the ball's oscillations and reduce its energy before collecting another target. The score and remaining time are visualized on the left and right side of the screen, respectively.

The simulated ball-in-bowl task has a software-defined natural frequency. We modulate the resonant frequency by changing the length of the modelled pendulum. There is an inverse correlation between the length of the pendulum $h$ and task resonance $f_{task}$:

$$f_{task} = \frac{1}{2\pi}\sqrt{\frac{g}{h}}.$$

Modulating task resonance enables us to assess ability to generate motion at a specific frequency. In our experiment, we use pendulum lengths $h = [0.995, 0.249, 0.111, 0.04]$ that correspond to natural frequencies $f_{task} = [0.5\text{Hz}, 1\text{Hz}, 1.5\text{Hz}, 2.5\text{Hz}]$. We chose 2.5Hz as our maximum frequency, so that it would be achievable for all study participants [**26**]. Because we noticed that changes in frequency were not perceived by participants linearly—namely, a change from 0.5Hz to 1Hz was more perceptible than a change from 2Hz to 2.5Hz—we tested more frequencies near the lower bound of the range.

### 2.2.4. Study participants

We recruited seven able-bodied individuals to complete the study. All participants expressed verbal and written consent to participate. The study protocol was approved by the Northwestern University Review Board under IRB STU00021840. Participating individuals were screened for physical disabilities and known abnormalities in motor control—none were reported. The group of tested individuals was 22-28 years of age (with an average of 26). There were 4 males and 3 females; all were right-hand dominant. We randomly assigned the arm with which individuals performed our experimental tasks—4 of our participants used their right arm while 3 used their left arm during the experiment.

### 2.2.5. Experimental Procedure

An experimental session starts with measuring arm weight, so that during the task we can support the arm against gravity. We then define the individual's reachable workspace by asking each participant to "clean" the virtual table. Participants are instructed to cover the biggest area on the table that they can reach without allowing their shoulder to leave the back of the Biodex chair. During later task attempts, targets are placed only within this reachable area.

We then proceed to a training period, when we explain the ball-in-bowl task. We allow numerous attempts at the task until the participant feels comfortable with the experimental setup and they seem to understand the task objective. We offer tips and guidance as well as encourage the participant to ask questions. The training period usually takes 15-20 minutes with at least 10 task attempts.

All individuals complete the ball-in-bowl task at four frequencies (0.5Hz, 1Hz, 1.5Hz, and 2.5Hz) and under four task conditions (with/without the ball moving, with/without haptic feedback). During trials without the ball moving, the ball rests at the bottom of the bowl, while the participant is asked to collect targets without needing to consider the ball's movements. The goal of the no-ball trials is two-fold: (1) without haptic feedback, to evaluate the baseline frequency spectrum of movement during the ball-in-bowl task, and (2) with haptic feedback, to confirm that the haptic forces themselves, without active participant movement, do not result in peaks at resonance. Each experimental set of conditions is attempted by the participant 24 times: 6 times at each task frequency in 2 sets of 3 in random order to control for potential effects of learning and/or fatigue. There is one exception: when the ball is stationary and no haptic feedback is provided, there is no notion of task frequency—participants attempt this experimental condition 6 times. Each study volunteer performs a total of 78 task attempts. We take

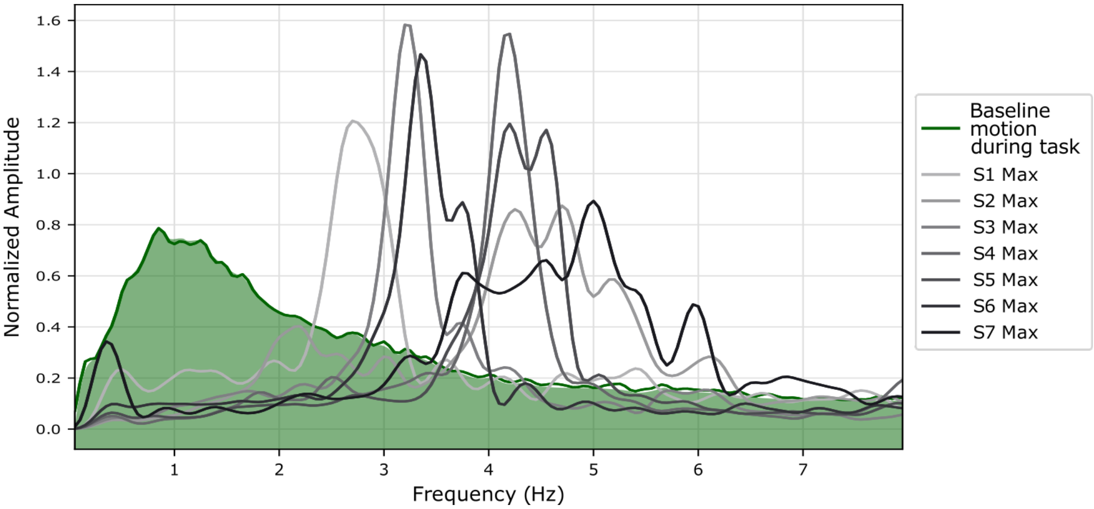

Figure 2.4. **Upper bound of motion bandwidth during open-loop movements (grey) in comparison to baseline closed-loop motion during the ball-in-bowl task (green).** All tested participants are able to generate open-loop motion up to 2.5Hz—their maximum open-loop motion frequency ranges from 3 to 5Hz. During the ball-in-bowl task, when collecting targets without a moving ball, most of participants' motion is exerted around 1Hz.

two 5-minute breaks after 27 and 54 trials, as well as shorter breaks throughout the experiment as needed.

We end the experiment with a test of open-loop motion—motion that relies on internal proprioception without incorporating external stimuli. While coupled with the robot, participants are asked to move back and forth in the transverse plane as fast as they can, using small amplitude motion. Unlike during closed-loop motion in the ball-in-bowl task, they are not generating motion in response to any task-specific stimuli. We collect three 10-second efforts.

### 2.2.6. Motion metrics

We assess motion by looking at the spectral properties of the forces introduced into the system by the participant. While the load cell measures forces in the x-direction and y-direction in the transverse plane, a discrete fast Fourier transform (FFT) is performed on each trial to obtain the amplitude of the signal for a range of frequencies up to 4Hz. In a frequency spectrum, the relative amplitude of a signal at a certain frequency provides us with insight about participants' ability to switch movement direction at that frequency. The signal in the frequency domain is normalized by the energy introduced into the ball-in-bowl system by a participant throughout a task attempt. We use the following standard definition of a signal's energy: $E = \sum A(\omega)^2 d\omega$ where $\omega$ and $A$ are the frequency and amplitude, respectively. After normalization by $E$, the total energy introduced into the system during each task attempt is equal to one, allowing us to do trial-to-trial and subject-to-subject comparisons. To obtain a single estimate of the frequency content of movement in the 2D plane, we add the x and y frequency spectra. The resultant spectrum is re-normalized and visualized throughout this chapter.

The primary frequency metric presented in this chapter is energy@resonance. It refers to the energy exerted by the participant at the task frequency $\pm 0.2$Hz (equivalent to a window size of $w_f = 0.4$Hz). We compute the energy@resonance measure for each trial with a moving ball. Additionally, regardless of the task frequency, we compute an energy@frequency metric—the energy at $0.5 \pm 0.2$Hz, $1 \pm 0.2$Hz, $1.5 \pm 0.2$Hz, and $2.5 \pm 0.2$Hz—to validate that the game elicits movement at task resonance. For the last analysis, we compute the percentage difference in energy@resonance between trials with haptic forces and without haptic forces, i.e., $(e_{haptic\ feedback} - e_{no\ haptics})/e_{no\ haptics}$, where $e$ is the energy at resonance.

Aggregate frequency spectra presented in this chapter are averaged across participants, re-normalized, and filtered with a low-pass Butterworth filter (and a cutoff frequency of 5Hz) for visual clarity. In creating the boxplot figures, we average across trials of a participant under the same trial conditions so that each boxplot demonstrates the spread in performance across the seven participants. All statistical analyses, except the window size analyses, are performed on unaggregated and unfiltered frequency spectra and metrics.

### 2.2.7. Study design and statistical analyses

Our experiment aims to test two hypotheses: (H1) the ball-in-bowl game encourages participants to move at the task's resonant frequency, and (H2) haptic feedback further encourages movement at task resonance.

To assess H1, we perform two statistical analyses. First, we perform a set of one-way repeated measures ANOVAs (rm-ANOVAs) and post-hoc t-tests with Bonferroni corrections to determine whether the ball's resonant frequency affects the energy exerted by participants around specific frequencies when completing the task without haptic feedback. This determines whether, for example, participants completing trials using a ball with a resonant frequency of 0.5Hz exert more energy around 0.5Hz than while completing trials using a ball with a resonant frequency of 1Hz, 1.5Hz, or 2.5Hz. We repeat the same analysis using trials with haptic feedback. For the second statistical analysis, we compare energy@resonance during trials with a moving ball to trials with a still ball (and hence, no incentive to move at the resonant frequency). We use a two-way rm-ANOVA with within-subject factors for the trial condition (moving vs. stationary ball) and ball frequency, followed by one-way rm-ANOVAs at each frequency with trial condition (still vs. moving ball) as the within-subject factor.

To assess H2, we perform two statistical analyses. First, we perform a two-way rm-ANOVA with within-subject factors for ball frequency and trial condition (with vs. without haptic feedback) across trials with a moving ball, followed by one-way rm-ANOVAs at each frequency with trial condition (with vs. without haptic feedback) as the within-subject factor. Secondly,

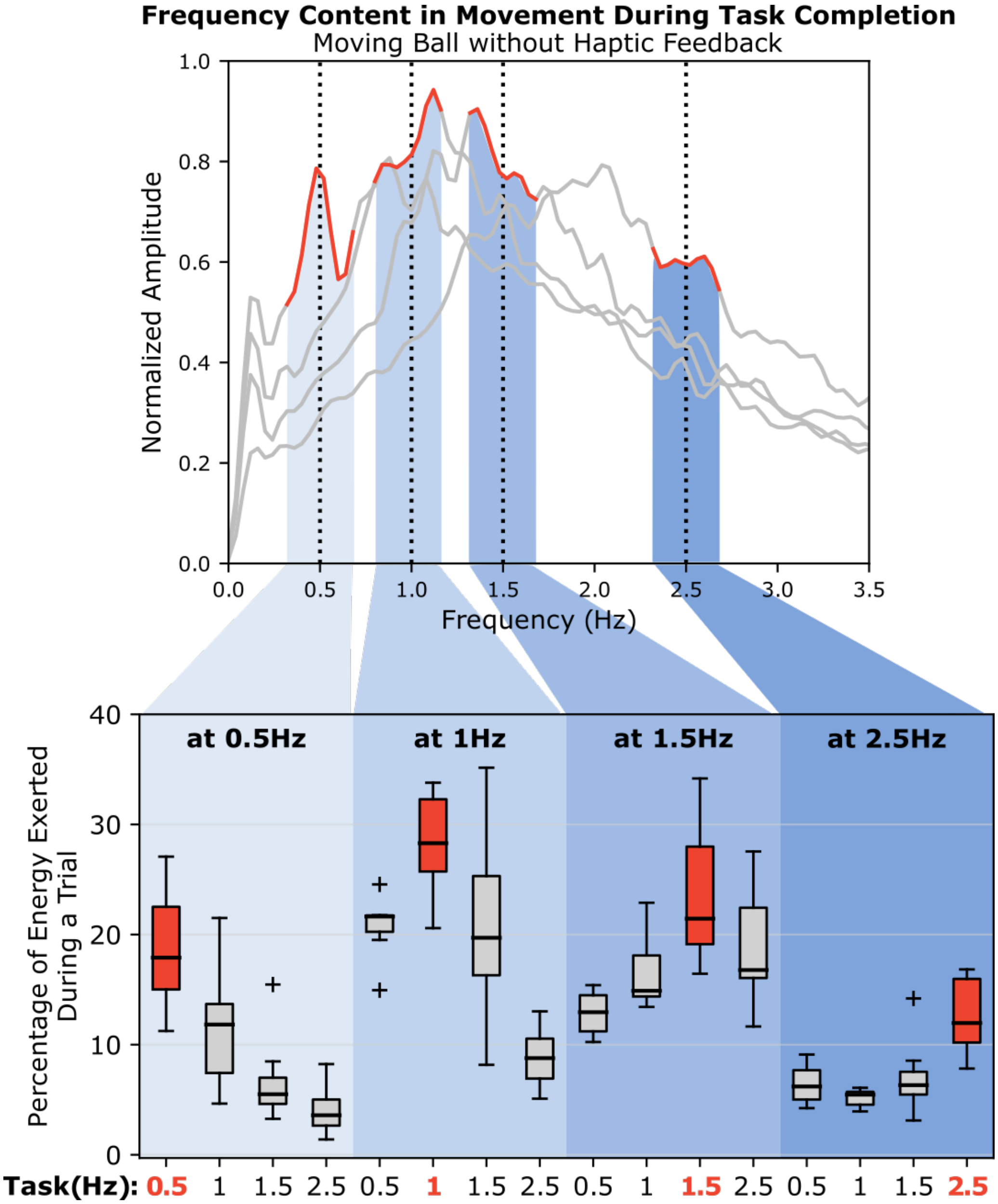

Figure 2.5. **Game elicits movement at task resonance.** Energy@resonance (red boxes) is higher than energy@frequency (grey boxes) for all 4 tested task frequencies, denoted by the 4 shades of blue. Aggregate results (n=7). Boxes in the bottom plot represent the area under the curve of sections in the top plot (highlighted in blue) across all participants.

we look at the percentage difference in energy@resonance between aggregated trials with haptic forces and without haptic forces. We perform a two-way rm-ANOVA with within-subject factors for task frequency and window size, followed by one-way rm-ANOVAs at each frequency with window size as the within-subject factor.

For each statistical test, we evaluate the assumptions using the Shapiro-Wilk test for normality and Mauchly's test for sphericity. If the sphericity assumption is violated, we report the Greenhouse–Geisser correction, indicated by $p_{GG}$.

## 2.3. Results

Using the ball-in-bowl task, we show that frequency decomposition of motion during a dynamic task can be used to assess motion quality, revealing information not easily obtained from the typically studied time-series data. In a small human subject study with able-bodied individuals (n=7), we validate that the ball-in-bowl game is able to elicit active movement at specific frequencies. We show that haptic cues improve motion timing and increase frequency content in motion at task resonance.

### 2.3.1. Quantifying motion bandwidth during open-loop movements

As a baseline, we evaluate open-loop motion. As visible in Fig. 2.4, participants generate open-loop motion at frequencies ranging from 3 to 5Hz depending on the individual. Consequently, we conclude that all participants are physically able to generate open-loop motion up to 2.5Hz—the highest frequency in our experimental protocol. Given this result, the dynamic response we are measuring during the ball-in-bowl task is limited mainly by the capabilities of participants' reasoning about motion timing rather than a physical inability to move at the tested frequencies.

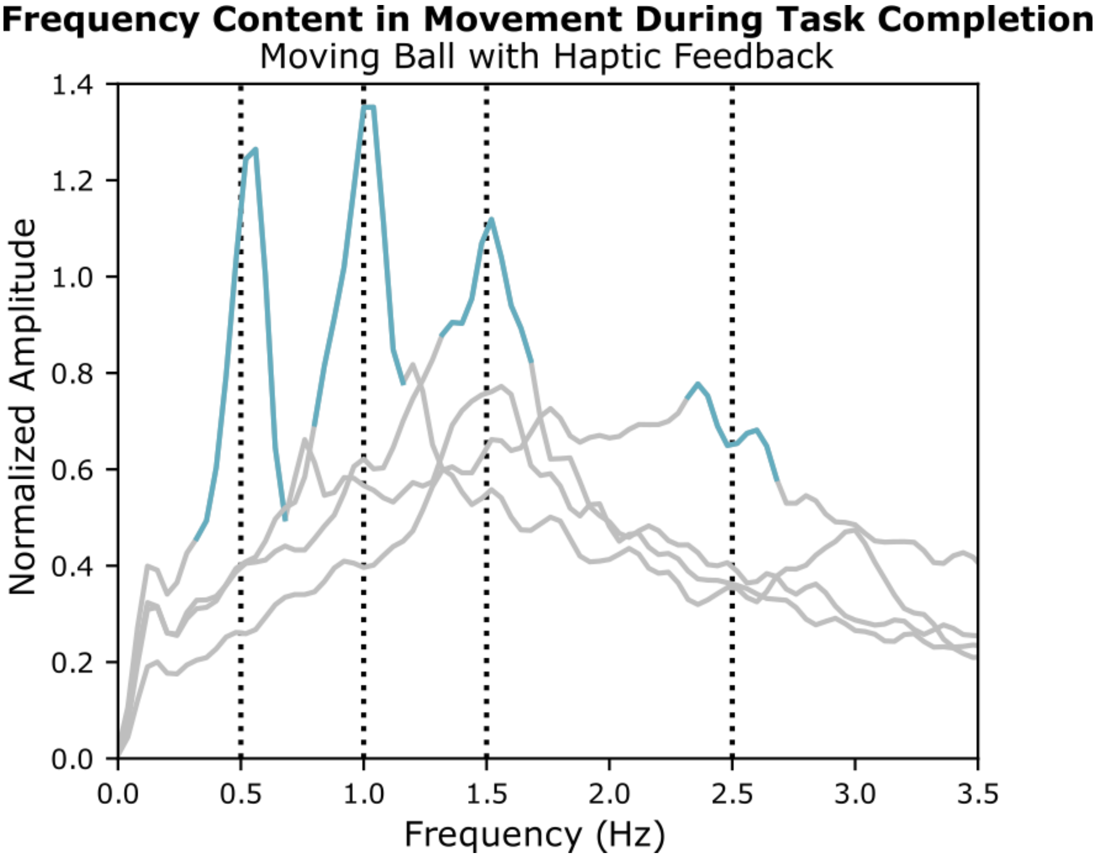

Figure 2.6. **Haptic feedback encourages movement at task resonance.** The peaks at resonance (indicated with a blue color) are higher when haptic feedback is rendered to participants than when only visual cuing is provided. Aggregate results (n=7).

In Fig. 2.2 and Fig. 2.3, we illustrate the motion of one of our study participants during an example task attempt under each of the four experimental conditions. In Fig. 2.2, we visualize participant trajectories as a function of x and y. Note that when the ball is stationary, the participant moves quickly through their workspace, solely focused on collecting targets. This baseline motion has a characteristic frequency decomposition (green curve in Fig. 2.4)—if no other incentives are provided, the motion focused on collecting targets during the ball-in-bowl task largely centers around 1Hz. When the ball is active, participants alternate between generating high-frequency movement to settle the ball and traversing the workspace to collect targets.

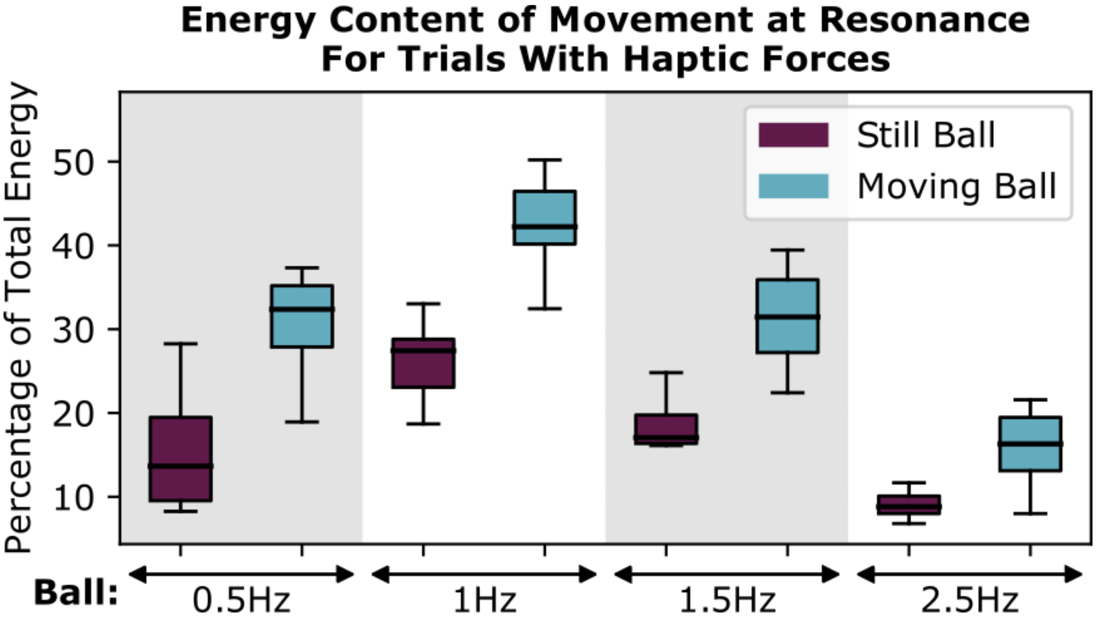

Figure 2.7. **Game is measuring active movement.** Energy@resonance is lower when participants are not actively trying to balance the ball.

The motion focused on settling the active ball introduces peaks around various frequencies, corresponding to the task's resonance, allowing us to assess an individual's interaction bandwidth in the physical human-robot system.

In Fig. 2.3, we illustrate participant movement as a function of time. Note that qualitatively it is difficult to compare participant's performance between experimental conditions and task frequencies. In contrast, by looking at the frequency content in motion, we can more easily reason about a participant's performance both quantitatively and qualitatively.

### 2.3.2. Measuring interaction bandwidth during a dynamic task

By using a task with a resonant frequency, we can quantify an individual's ability to generate a dynamic response at a specific frequency (see Fig. 2.5 for aggregate results). The energy exerted

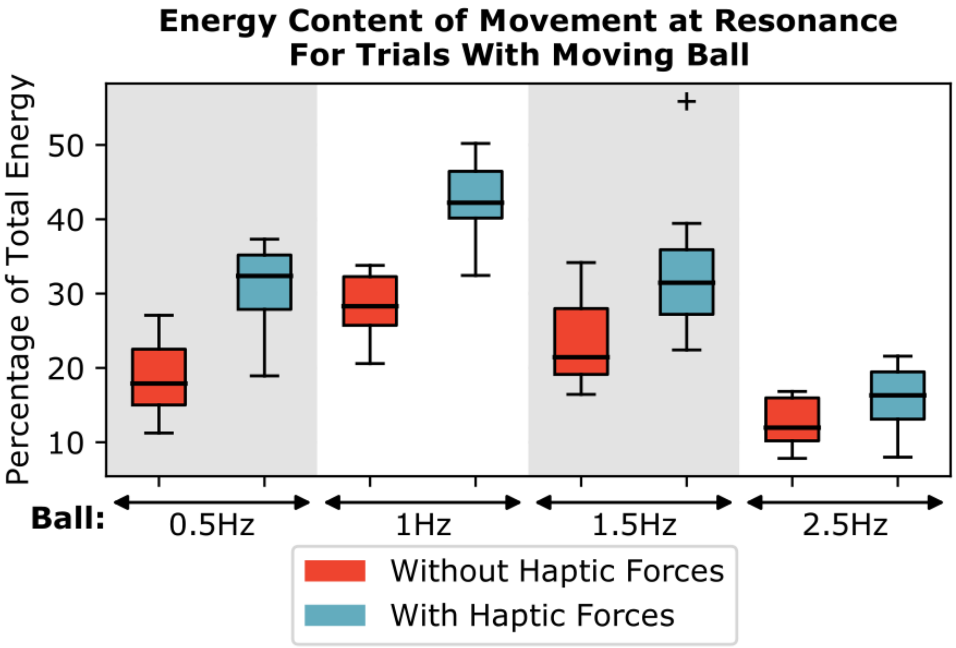

Figure 2.8. **Haptic forces improve movement timing compared to no forces.** Energy@resonance is higher for trials with visuo-haptic feedback than with only visual cuing.

at the task's resonant frequency is consistently higher than energy exerted at that frequency during tasks with a different resonant frequency. Repeated measures ANOVAs reveal that, when only visual feedback is provided, task frequency significantly affects energy exerted at 0.5Hz ($p < 0.001$, $F = 18.94$), 1Hz ($p < 0.001$, $F = 15.28$), 1.5Hz ($p = 0.009$, $F = 5.23$), and 2.5Hz ($p < 0.001$, $F = 11.23$), respectively. The ANOVA is performed for the energy@frequency metric, delineated by different shades of blue in Fig. 2.5. Moreover, post-hoc t-tests with Bonferroni corrections reveal that for each task frequency, the energy@resonance metrics (red boxes in Fig. 2.5) are significantly higher than the energy@frequency metrics for the same frequency (grey boxes in Fig. 2.5) with all p-values less than 0.004.

The same analysis is performed for trials with a moving ball and haptic feedback. Again, we observe that individuals exert significantly more energy at task resonance (0.5Hz ($p < 0.001$,

$F = 72.10$), 1Hz ($p < 0.001$, $F = 57.85$), 1.5Hz ($p < 0.001$, $F = 8.57$), and 2.5Hz ($p < 0.001$, $F = 15.19$)) than they would if the task did not encourage movement at this frequency. Moreover, post-hoc t-tests with Bonferroni corrections reveal that energy@resonance (energy under the blue curves in Fig. 2.6) is significantly higher than energy@frequency for trials with a difference resonant frequency, with all p-values less than 0.001.

When the ball is stationary, participants are no longer incentivized to generate movement at a specific frequency to settle the ball. Thus, we observe that participants produce significantly less movement at the ball's resonant frequency (see Fig. 2.7) when the ball is still. The repeated measures ANOVA shows that trial condition (still vs. moving ball) significantly affects energy@resonance ($p < 0.001$, $F = 78.92$) across frequencies and at each frequency—0.5Hz ($p = 0.003$, $F = 22.59$), 1Hz ($p = 0.001$, $F = 48.53$), 1.5Hz ($p = 0.030$, $F = 8.05$), and 2.5Hz ($p = 0.045$, $F = 6.41$). Since haptic forces are present in both experimental conditions, this analysis confirms that a peak in the energy exerted at the task's resonant frequency cannot be fully attributed to an artifact of the haptic feedback. Instead, these results suggest participants are actively moving at the task's resonant frequency in order to settle the ball and succeed at the task.

Using the ball-in-bowl task, we are able to elicit a dynamic response at specific frequencies and quantitatively assess individuals' ability to perform movements at a chosen frequency. Results confirm that the game elicits active movement at the tested frequency.

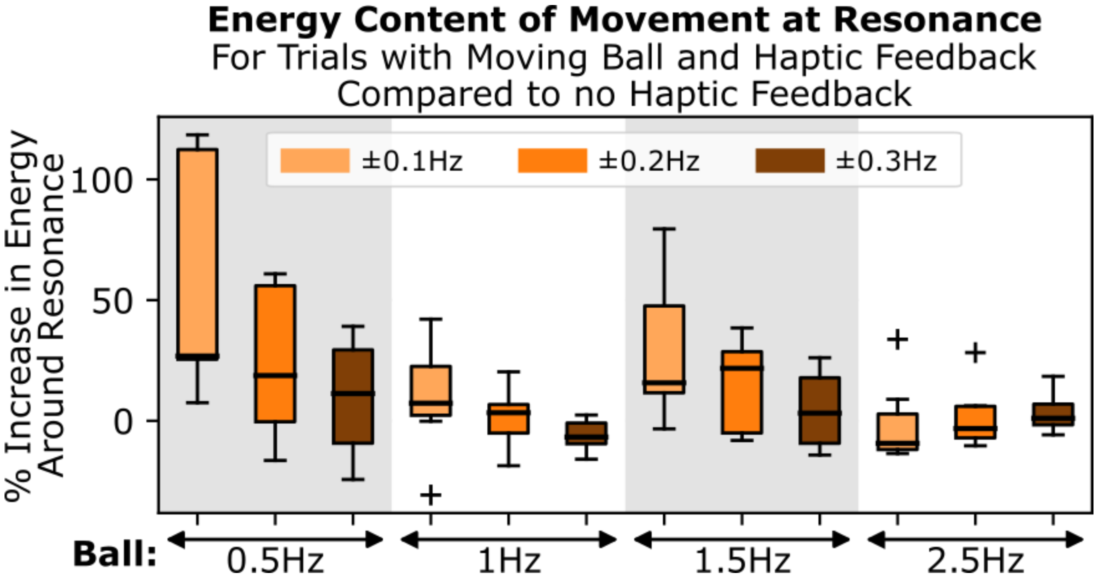

Figure 2.9. **Haptic forces guide precise timing of movement.** Haptic forces improve energy@resonance more the smaller the window size $w_f$. The effect might only be true up to a certain frequency. At 2.5Hz, we no longer observe this trend.

### 2.3.3. Impact of haptic feedback on motion timing

The haptic feedback we provide during experiments makes the task more interactive and perceptually real. Moreover, in our results we find that haptic feedback improves the timing of individuals' motion—when completing the ball-in-bowl task, participants exert more effort near task resonance than during trials without haptic feedback.

When participants are provided both visual and haptic feedback, we observe that the frequency spectrum peaks (blue curves in Fig. 2.6) become more pronounced than when participants are provided only visual feedback (red peaks in Fig. 2.5). Accordingly, the statistical analysis reveals that participants exert more energy@resonance with haptic feedback compared to without haptic feedback (refer to Fig. 2.8). The repeated measures ANOVA shows that trial condition (haptic vs. no haptic feedback) significantly affects energy@resonance ($p = 0.002$, $F = 29.72$) across frequencies and at most frequencies individually—0.5Hz ($p = 0.002$, $F =$

26.43), 1Hz ($p = 0.012$, $F = 12.82$), 1.5Hz ($p = 0.031$, $F = 7.85$)—with the energy@resonance at 2.5Hz being marginally significant ($p = 0.067$, $F = 5.01$). There is also an interaction effect between ball frequency and trial condition ($p = 0.003$, $F = 3.81$), which may reflect a stronger impact of haptic feedback at lower frequencies compared to higher frequencies.

Lastly, we find that haptic feedback enables the participant to more closely match the task's resonant frequency with their motion. A repeated measures ANOVA reveals that window size ($w_f \in$ [0.2Hz, 0.4Hz, 0.6Hz]) significantly impacts how much haptic feedback improves the energy exerted around resonance ($p < 0.001$, $F = 15.83$) across frequencies (see Fig 2.9). Additionally, there is a significant interaction effect between window size and ball frequency ($p < 0.001$, $F = 9.71$), reflecting a stronger trend at 0.5Hz ($p_{GG} = 0.003$, $F = 20.75$), 1Hz ($p_{GG} = 0.107$, $F = 3.51$), and 1.5Hz ($p_{GG} = 0.026$, $F = 7.93$) compared to a weaker trend at 2.5Hz ($p_{GG} = 0.315$, $F = 1.23$). These results could indicate that people have a limited perception bandwidth for haptic feedback, meaning that the temporal resolution of haptic perception is not high enough to allow us to internalize haptic feedback above a certain frequency (between 1.5Hz and 2.5Hz). However, the results are not conclusive, because the within-frequency trends are statistically significant (with a p-value $< 0.05$) only at 0.5Hz and 1.5Hz. Moreover, visual feedback has been show to dominate proprioceptive feedback [**68**]—it would be interesting to do a follow-up experiment with haptic but no visual feedback.

## 2.4. Discussion & Future Work

For effective human-robot collaboration, it is beneficial to understand people's bandwidth for motion and physical interaction. The scientific community outside of robotics has investigated these questions before. As an example, studies report that able-bodied individuals can generate a hand grasp with a delay of 200ms in response to a sound. If these upper-limb reactions are generated repeatedly, a delay of 200ms suggests an interaction bandwidth of 5Hz. However, continuous interaction is mechanistically different from a one-off reaction in that it includes motion planning and re-planning in the motor cortices, cerebellum, and basal ganglia as well as motion termination and re-initiation at the level of the muscle. These processes may slow down interaction bandwidth in a complex, dynamic task.

Thus far, most research has investigated one-off reaction times [**107**, **134**, **168**, **185**] rather than continuous interaction bandwidth. Using frequency analysis in a dynamic task, such as the ball-in-bowl task, we can evaluate the bandwidth of repeated interaction available to an individual in a specific context. In our experiments, we evaluate upper-limb movements in response to visual and haptic stimuli. By adjusting the feedback provided to the participant, we could further study interaction bandwidth under different modalities of sensory inputs. By using a different type of robot, we could study movements of other body parts, such as hands or legs.

In many neuromotor impairments, such as stroke, reaction times are delayed [**168**]. Our method can be used for quantitative assessment of motion and/or interaction bandwidth for individuals with neuromotor impairments. We can use it to examine deficits in receiving and interpreting sensory feedback due to a neuromotor disorder and determine how these deficits affect dynamic performance. Given this information, robotic systems focused on assistance or rehabilitation can tailor the interaction to the capabilities of that particular individual. In our ongoing work, we are looking at the impact of hemiparetic stroke on motion bandwidth post-injury.

While dynamic motion is part of many daily activities, such as walking, carrying a bag of groceries, or catching a falling object, current clinical assessments largely focus on static and quasi-static movements, because we do not have well-established methods for quantifying dynamic performance. A reliable, quantitative assessment of dynamic motion can have significant positive implications for translational research and clinical practice. In research, it can improve our ability to study the underlying causes of dynamic deficit. In clinical practice, it can be used for (1) precise tracking of disease progression, (2) studying therapeutic efficacy, and (3) early diagnosis of deficits in movement coordination and motion bandwidth. Long-term, our method can become the foundation of robot-assisted rehabilitation focused on re-training dynamic movements.

## 2.5. Conclusions

In this work, we propose a method for measuring interaction bandwidth during pHRI. To validate the method, we run a human subject study (n=7) and test frequencies 0.5-2.5Hz—all within the normal range of a dynamic response for an able-bodied individual. We show that our game successfully elicits dynamic motion at the tested frequencies. Secondly, we show that haptic feedback helps guide the timing of participants' motion compared to only visual cuing. The method can be further used to analyze interaction bandwidth of human-robot systems, particularly to tailor interaction parameters to the individual capabilities of people with neuromotor impairments. Our ongoing work is applying the method to quantify deficiencies in dynamic motion after a hemiparetic stroke.

CHAPTER 3

# Robotic assessment of motion bandwidth and dynamic deficit

The lack of sensitive tools to quantify dysfunction hinders our ability to study the underlying causes of impairment and makes it difficult to demonstrate the effectiveness of therapeutic approaches. Consequently, it slows down progress on developing novel treatment protocols, including personalized disease-targeted interventions. While there exist well-established metrics of static and quasi-static motion, such as reaching range, little emphasis has been placed on quantifying dynamic response—controlled and timing-sensitive movements where the continuous modulation of motor activity is required to respond to real-time stimuli. In this study, we employ robot-assisted virtual tasks that require dynamic motion in the upper limb, and develop metrics that assess dynamic capabilities by quantifying the frequency spectra of movement during these tasks. We assess chronic survivors of hemiparetic stroke (n=13+48) across three dynamic tasks. We find that stroke causes a significant decline in dynamic response at frequencies above 1.5Hz in the paretic upper limb, with the degree of functional loss dependent on the clinically assessed severity of motor impairment. Frequency-based metrics of dynamic motion could be used to assess performance during everyday activities, such as brushing teeth, cooking, or cleaning. Versatile quantitative assessment of dynamic motion can accelerate progress in understanding and rehabilitating functional dynamic motion in individuals with neuromotor disorders.

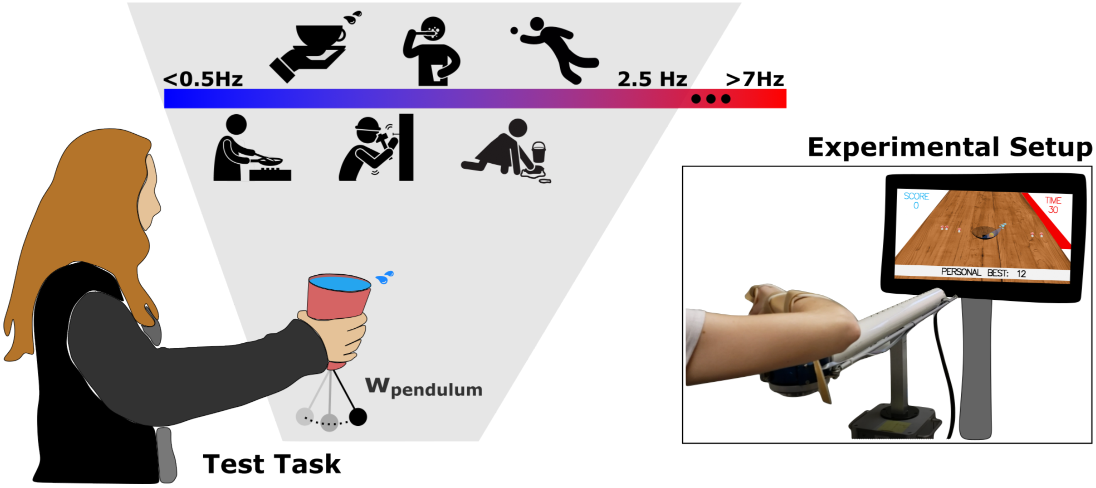

Figure 3.1. **Tasks with resonant frequencies enable assessment of an individual's ability to generate dynamic motion.** Dynamic motion that requires fast but controlled actions plays a key role in activities of daily living, such as brushing teeth, catching a falling object, or moving a cup of water without spilling. The ball-in-bowl task can be set to test for a dynamic response at any frequency. In our experiments, we assess motion at frequencies between 0.5Hz and 2.5Hz—encompassing the range that able-bodied individuals are capable of and use in their daily activities.

## 3.1. Introduction

Many activities of daily living (ADLs), such as self-feeding or self-cleaning, involve dynamic motion—fast, controlled, and timing-sensitive movements where the continuous modulation of motor activity is required to respond to real-time stimuli. For fast coordinated motion, able-bodied individuals rely on an intact lateral corticospinal tract—the fastest and most direct pathway from the brain to distal limbs. During neuromotor disorders, fibres in the corticospinal tract can be significantly severed. Moreover, individuals might experience damage to other parts of the nervous systems that control motor activity, such as the motor cortices, the cerebellum or the internal capsule. While the effect of a neuromotor disorder on dynamic motion can vary depending on how and where the damage occurs, the underlying changes can significantly impede patients' ability to perform routine ADLs [**115**].

To better understand the changes in dynamic performance in impaired individuals, both clinicians and researchers rely on established assessments of impairment, such as the Fugl-Meyer Assessment (FMA) or the Unified Parkinson Disease Rating Scale (UPDRS). Of note, clinical assessments primarily assess static and quasi-static movements—quantifying movement range or independent joint control—and devoting little focus to functional dynamic motion. For example, during the Fugl-Meyer Assessment for the Upper Extremity (UE-FMA)—a clinical assessment of upper-limb performance after a stroke—only 6 out of 66 points are allocated to assess speed and coordination—qualities of motion that explicitly affect the patient's ability to generate dynamic motion [**76**]. Moreover, similar to other observer-based clinical assessments, the FMA is known to be subjective and relatively coarse, making it difficult to reliably quantify subtle but clinically important functional changes [**164**]. Due to the lack of reliable metrics to assess dynamic movement, the impact of neuromotor impairments on dynamic performance is not well understood. There is a need for improved assessment methods for the evaluation of dynamic motion in neuromotor disorders.

With the development of wearable sensors and robotic technology, there has been significant progress in designing and validating assessment methods [**21**, **101**, **135**, **183**], as we describe in more detail in Section 3.2. With sensing technology becoming more affordable and more widespread (many individuals already wear a motion tracker on their wrist), there is potential for continuous, quantitative tracking of motion quality. Robotic technology has already made a profound impact on our understanding of sensorimotor function as well as motor learning and re-learning [**66**, **75**, **195**]. Using quantitative metrics of motion, wearable sensors and robotic devices can help address the limitations of current clinical measures, providing a reliable, quantitative approach for measuring losses of dynamic function associated with sensorimotor impairments.

In this work, we employ a novel methodology for assessing dynamic performance called QADR—Quantitative Assessment of Dynamic Response, and use it to assess a decrease in motion bandwidth due to hemiparetic stroke. We assess dynamic performance by evaluating the frequency response in motion during completion of three virtual tasks. Our first task is loosely based on the real-world activity of carrying a cup of water without spilling, similar to the virtual tasks used in prior studies [**135**,**173**,**177**]. It can assess an individual's ability to generate motion at a prescribed frequency [**101**], offering an estimate of motion bandwidth—the body's ability to execute motor commands in response to a desired change in speed and/or direction of motion. The other two tasks give the individual freedom to move and change direction at frequencies at the upper bound of their motion bandwidth. The second task involves virtual hammering of nails into a wall, while the third task requires an individual to hit objects that randomly appear in their virtual environment at an increasing pace. The tasks are meant to be representative of other dynamic activities that we perform in our daily lives, such as cooking, cleaning or tooth-brushing, as illustrated in Fig. 3.1. In our experiments, the virtual task environment is paired with an assistive robot that is able to provide (1) task-specific haptic feedback as well as (2) variable support for the arm against gravity. In future studies, the assessment could be performed using a wearable sensor, e.g., a bracelet with an inertial measurement unit (IMU) for at-home monitoring [**124**].

While stroke is a leading cause of disability in developing countries [**32**], post-stroke changes in dynamic performance are not well-understood. Due to the prevalence of stroke [**13**] and the nervous system's capacity for neuroplasticity [**50**], stroke survivors are well-positioned to benefit from a better understanding of a post-stroke loss in dynamic motion capabilities. Using quantitative assessments, it has been shown that stroke affects static and quasi-static metrics of motion—stroke survivors experience a decrease in reachable workspace [**65**], increased error in precision of motion [**47**], as well as delays in initiation and termination of movement [**40**, **168**]. At the same time, they experience a less-studied difficulty with dynamic motion. As an example, they have lower ability to adapt when moving inside a force field, where an internal model of motion dynamics is required to compensate for the external forces [**180**]. They also experience difficulty when asked to tap in rhythm with auditory cues at 1.25Hz—a task that requires continuous modulation of muscle activity to succeed [**30**]. Here, we aim to quantify changes in motion bandwidth within reachable workspace of the upper limb due to a hemiparetic stroke. We propose a versatile, quantitative evaluation of dynamic motion that could offer a task-independent approximation of the functional challenges that individuals with stroke experience in everyday life. The proposed results can begin to give insight into the scientific causes of impaired dynamic movement, and pave the way for novel rehabilitation protocols focused on recovering functional dynamic capabilities.

**As our first contribution, we show that a post-stroke impairment can be identified and quantified by evaluating the frequency spectra of movement during dynamic tasks.** We show that frequency-based metrics can detect post-stroke impairment and that the assessment generalizes across three dynamic tasks. Moreover, we show that there is a significant moderate correlation between clinically determined levels of impairment (via UE-FMA) and a reduction in motion bandwidth (measured using frequency-based metrics).

**As our second contribution, we show that stroke decreases motion bandwidth, affecting motion above 1.5Hz.** Across tasks, we find that moderate-severe stroke hinders dynamic response at higher frequencies ($>$ 1.5Hz) without affecting dynamic response at lower frequencies ($<$ 1.5Hz). Moreover, stroke lowers patients' default frequency of fast directional movement in free-form dynamic tasks—stroke survivors exert more energy at lower frequencies with their paretic arm than their nonparetic arm in the tested tasks.

**As our third contribution, we show that a shift in reliance on descending pathways for motor control might be contributing to the decrease in dynamic function.** While shoulder abduction loading has been correlated with an increased reliance on indirect pathways for motor control [**60**, **129**], we observe that loading impacts participants' performance around 1.5Hz—a range of motion frequencies where stroke survivors' motion begins to be affected by stroke.

## 3.2. Background: Quantitative Metrics of Neuromotor Disorders

Clinical assessments of impairment act as the gold standard in clinical practice and in translational research. For stroke, these assessments include: the FMA [**76**], Chedoke-McMaster Assessment (CMSA) [**77**], MRC Scale of Muscle Power, the Wolf Motor Function Test (WMFT), or the Action Arm Research Test (ARAT). While clinical assessments focus on evaluating static and quasi-static movement characteristics (e.g. movement range or grip strength), many include a small component that evaluates dynamic motion, e.g., finger/toe/heel tapping in Parkinson's Disease [**139**,**146**] or the nose-to-knee test in the FMA for stroke [**76**]. For example, in the knee-to-nose test, individuals are asked to use their index finger to move between their knee and nose 5 times as fast as they can while seated in a chair. Some tests are specifically designed to assess functional abilities that combine many aspects of movement. Examples include the WMFT or the ARAT. While useful in providing an approximate rating of impairment level, clinical tests require a highly trained clinician to administer and their execution is time-consuming; they provide a subjective coarse-grained measurement with a bias in the scores from floor/ceiling effects and inter-rater variability [**164**].

To provide more consistent and objective measures of impairment, significant research efforts have been made to introduce reproducible, quantitative metrics. Established quantitative assessments evaluate reaches towards a stationary target, characterizing parameters, such as error from target, movement speed, and reaction time [**157**]. Other evaluations measure reachable workspace—the area that an individual can reach with their hand without movement of the trunk [**66**, **116**]. Although quantitative and reproducible, the tests often analyze movements that isolate individual degrees of freedom, overlooking functional motion. They put little emphasis on understanding the effect of impairment on an individual's ability to perform functional dynamic tasks. As a result, the scores do not adequately reflect a patient's ability to engage in ADLs, such as self-cleaning or eating and drinking.

More recently, research efforts have been made to assess dynamic-movement capabilities. A large body of work studies post-stroke changes in the speed of motor reflexes [**36**, **86**]. Another body of work investigates how people with neurological impairments generate corrective movements—in reaction to perturbations of applied forces. As an example, Takahashi *et al.* study adaptation when moving inside a force field [**180**]. Lowrey *et al.* characterize delays in the initiation of corrective movements when a perturbation is applied to the hand during reaching [**121**]. However, both reflexive and corrective movements are known to rely on different mechanisms than conscious, voluntary motion [**48**].

To investigate dynamic deficits in volitional motion, researchers have developed robot-assisted dynamic tasks that require coordinated, timing-sensitive movements. Calautti *et al.* assess how well stroke survivors generate movements in synchrony with a repeating auditory cue [**30**]. Tyryshkin *et al.* introduce the object hit task [**183**], which we describe in more detail in Section 3.3.1. Using the same KINARM platform, Bourke *et al.* proposes and validates the object hit-and-avoid task [**21**], increasing task complexity. Our prior work [**101**] has validated a virtual ball-in-bowl task, enabling the characterization of interaction bandwidth in able-bodied individuals. Similarly, Nayeem *et al.* have proposed a physical ball-in-bowl platform for assessing functional object interactions after stroke during reaches between two targets [**135**]. Using these types of tasks, researchers can begin to quantitatively assess functional dynamic motion.

As a baseline, researchers can evaluate task success, similarly to how it is done during task-based clinical assessments, e.g., the WMFT or ARAT. Additionally, by sensorizing the patient during these tasks, researchers can record movement trajectories and characterize an individual's motion qualities. Continuous measures of dynamic movement can help assess functional deficits beyond relying on task success. While multiple metrics of motion have been proposed in prior work [**96**], to date no metric has gathered enough evidence to warrant wide adoption. Researchers have proposed metrics that quantify jerk or motion smoothness [**135**], assuming that smooth motion is desirable. Others have built predictive models of motion that enable real-time quantification of motion predictability [**81**, **122**]. In robot-assisted rehabilitation, human movement is often evaluated based on the error of an individual's position and/or velocity from an optimal solution over time [**22**, **149**]. Of note, it is not always straightforward to identify the optimal, or even adequate, time-based trajectory for comparison [**71**]; any theoretically optimal control policy cannot *a priori* account for the unique physical and cognitive limitations experienced by an individual with neuromotor impairments. In contrast, QADR evaluates motion in the frequency domain, providing a quantitative assessment of motion bandwidth. Unlike existing methods, QADR allows for flexibility in performance over time and does not make *a priori* assumptions about the task solution. It allows for multi-joint, multi-degree-of-freedom assessment with or without support of the limb against gravity. Given encouraging results from the current study, which characterizes frequency-based assessments across three dynamic tasks (two without a resonant frequency), QADR shows promise as a an assessment tool for quantifying dynamic motion capabilities during daily activities.

## 3.3. Quantitative Assessment of Dynamic Response (QADR)

We assess post-stroke deficits using frequency-based metrics across three dynamic tasks, as illustrated in Fig. 3.2. We design and collect data for two tasks as part of our study, i.e., the ball-in-bowl task and the nail-and-hammer task. Additionally we analyze data from another dynamic task, i.e., the object hit task, designed and validated by Tyryshkin *et al.* during prior work [**183**].

### 3.3.1. Dynamic tasks

**The ball-in-bowl task.** The ball-in-bowl task is loosely inspired by the real-world activity of moving a cup of water without spilling. In the task, the location of the robot end-effector is mapped to the location of a bowl on a virtual table. There is a virtual ball rolling around inside the virtual bowl and the participant experiences haptic feedback corresponding to the interaction force between the bowl and the ball. The goal of the task is to collect as many targets as possible within a 30-second window. There are three conditions that have to be met for an individual

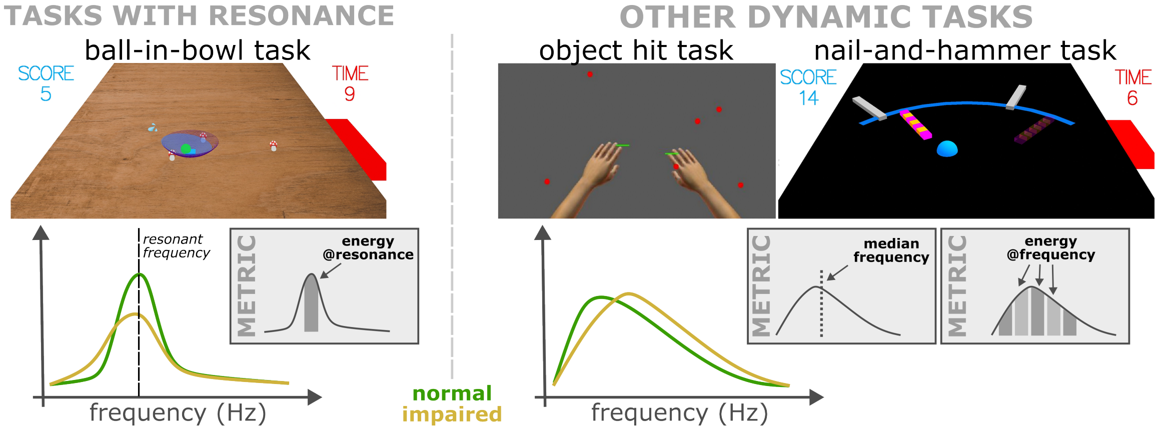

Figure 3.2. **Frequency-based assessment is a versatile metric, relevant to quantifying impairment during functional tasks.** During tasks with resonance (e.g., the ball-in-bowl task), interactive haptic probing enables measuring one's dynamic response at specific frequencies. During other dynamic tasks without a natural frequency (e.g., the object hit or nail-and-hammer tasks), frequency assessment quantifies an overall shift in motion bandwidth.

to be able to collect a target: (1) The individual's arm must be lifted above the haptic table, indicated to the participant by the bowl turning blue. (2) The virtual ball's total energy must be low enough to be oscillating no higher than one third of the bowl's height, indicated to the participant by the ball turning green. (3) The center of the virtual bowl must be aligned with the target location.

The main challenge of the task is to dampen out the ball's energy and prevent it from falling out of the bowl. The ball starts oscillating inside the bowl and is injected with a small amount of energy every time that a target is collected. When the participant moves in synchrony with the ball, they amplify the ball's oscillations and allow it to gain energy. When the participant counteracts the ball's movements, they dampen out its energy and prevent it from falling out of the bowl. To succeed, the participant must generate timely movements that counteract the ball's oscillations. The simulated ball-in-bowl task has a software-defined natural frequency, which we modulate by changing the length of the modelled pendulum. Modulating task resonance enables us to assess ability to generate motion at a specific frequency. In our experiment, we test four natural frequencies $f_{task} = [0.5\text{Hz}, 1\text{Hz}, 1.5\text{Hz}, 2.5\text{Hz}]$. The task was introduced in our prior work [**98**, **101**]. The current version of the task has been validated with able-bodied individuals [**101**] and is available online under the MIT license [**99**].

**The nail-and-hammer task.** The nail-and-hammer task is very similar to the real-world activity of hammering nails into a wall. In the task, the nails are located at one of 11 angles between $-50$ degrees and $+50$ degrees from center, as visualized in Fig. 3.5. Each nail requires 7 hits to be fully completed. For a nail to move into a wall, the participant's motion has to be aligned parallel to the nail $\pm 20$ degrees. At any given moment one nail is "active", indicated with bright colors on the screen and haptically rendered in the participant's workspace. Once fully hammered, the nail becomes "inactive" and a new nail appears. We instruct participants to hammer nails as fast as they can during each 30-second trial. The game requires fast coordinated motion.

**The object hit task.** The object hit task requires participants to hit away virtual balls moving towards them in their workspace using virtual paddles attached to each hand. The objective of the task is to hit away as many balls as possible. The balls appear at random from 10 equally-spaced bins at the top of the workspace, with a total of 30 balls released from each bin during a 2-minute trial. The number of balls that appear on the screen at a given moment and their speed increase throughout the task such that one single, slow-moving ($\sim$10 cm/s) ball is visible on the screen at the beginning of the task and a maximum of 16 fast-moving ($\sim$50 cm/s) balls are present towards the end of the task. To simulate feedback of object contact, the robot generates a 50 ms force pulse, based on the ball's acceleration, following contact with the paddle. The task is described in more detail by Tyryshkin *et al.* in a previously published article [**183**].

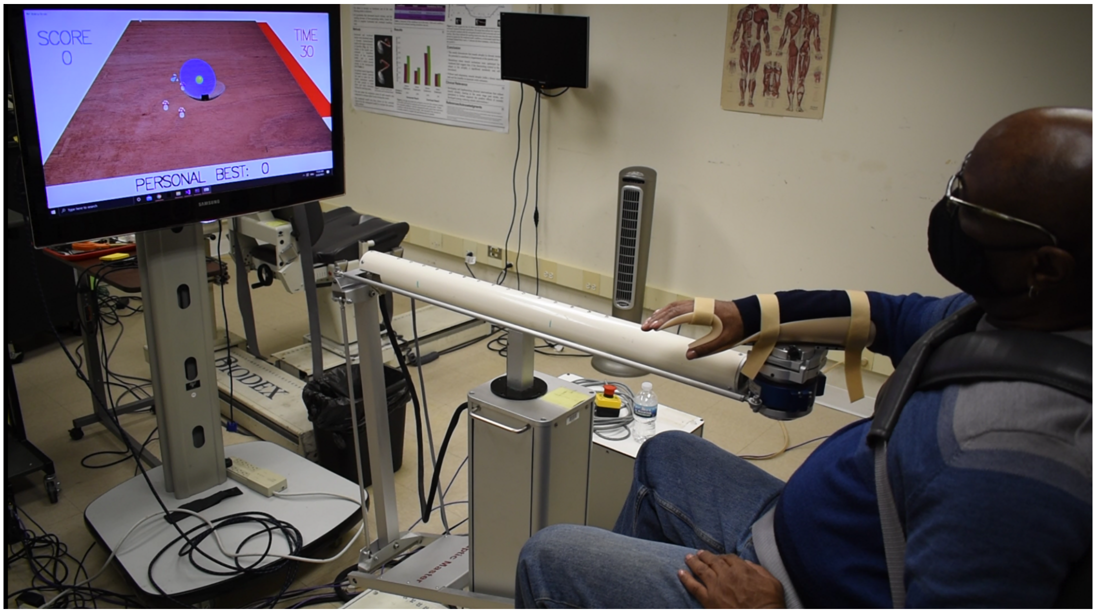

Figure 3.3. **Experiment 1 setup.** Participants are seated in a Biodex chair with their arm strapped into a forearm orthosis and connected to a robot end-effector via a load cell. A virtual game is visualized on a screen in front. The robot provides haptic feedback about the virtual game environment and can be used to support the arm against gravity.

### 3.3.2. Frequency-based metrics

We assess motion by looking at the spectral properties of forces exerted by a participant during each task. In this study, we analyze forces exerted by the upper limb in the x- and y-direction in the transverse plane. A discrete fast Fourier transform (FFT) is performed on each task trial to obtain the amplitude of the signal for a range of frequencies up to 4Hz. In a frequency spectrum, the relative amplitude of a signal at a certain frequency provides us with insight about participants' ability to switch movement direction at that frequency. The signal in the frequency domain is normalized by the energy exerted by a participant throughout a task attempt. We use the following standard definition of a signal's energy: $E = \sum A(\omega)^2 d\omega$ where $\omega$ and $A$ are the frequency and amplitude, respectively. After normalization by $E$, the total energy introduced into the system during each task attempt is equal to one, allowing us to do trial-to-trial and subject-to-subject comparisons. To obtain a single estimate of the frequency content of movement in the 2D plane, we add the x and y frequency spectra. The resultant spectrum is re-normalized and visualized throughout this chapter.

**Energy near resonance.** In the ball-in-bowl task, the primary frequency metric is energy@resonance [**101**]. This metric quantifies the energy exerted by the participant at the task frequency ±0.3Hz (equivalent to a window size of $w_f = 0.6\text{Hz}$). We compute the energy@resonance measure for each ball-in-bowl trial, at four task frequencies: 0.5Hz, 1Hz, 1.5Hz, 2.5Hz. Timesteps, where the participant is resting their arm on the haptic table or when the ball is "green" (has little oscillations), are not counted towards the energy@resonance metric. The metric is designed for tasks with resonance that require movement at a specific frequency that is known *a priori*.

**Percent loss in function.** As a secondary metric in the ball-in-bowl task, we use the percentage decrease in energy@resonance. This metric quantifies the loss in function in the paretic arm compared to the nonparetic arm, and is defined as follows:

$$\frac{E@R_{nonparetic} - E@R_{paretic}}{E@R_{nonparetic}} \times 100\%,$$

where E@R is the energy@resonance. A positive percentage indicates worse performance on the paretic side, while a negative percentage indicates better performance on the paretic side. The frequency spectra for each participant's movement with and without loading are additively combined and renormalized prior to computing the percent loss in function at each task frequency.

**Median frequency.** For the other two tasks that do not have a natural frequency, we use a primary metric based on the median frequency of motion. Specifically, we find the frequency value that defines the median frequency of movement generated by a participant in the range 0Hz to 4Hz. This is a versatile metric that could be used to assess motion in a range of dynamic tasks beyond those included in this study.

**Average frequency amplitude.** As a secondary metric for the nail-and-hammer and object hit tasks, we introduce average amplitude. This metric, similar to the energy@resonance, quantifies the energy exerted by the participant at a specific frequency ±0.5Hz (equivalent to a window size of $w_f = 1\text{Hz}$). We calculate an average frequency amplitude for the motion range 0Hz to 3.25Hz in increments of 0.25Hz (with overlapping windows) to find bands in motion bandwidth that are affected by stroke during free-form dynamic tasks without a resonant frequency.

### 3.3.3. Clinical metrics

We use the Fugl-Meyer Assessmenet for the Upper Extremity (UE-FMA) as the main clinical metric of stroke severity. Th UE-FMA offers a score in the range 0 to 66, with higher scores indicating better performance. Two versions of the UE-FMA score are used: (1) full UE-FMA, and (2) UE-FMA (arm only), where only shoulder/arm components of the assessment are considered. The UE-FMA (arm only) score excludes explicit assessment of the hand and wrist, offering a score in the range 0 to 42. Moreover, we separately consider the knee-to-nose component of the UE-FMA, defined as the time required to move one's index finger from knee to nose 5 times using the paretic arm. In our analysis, two versions are considered: (1) knee-to-nose (paretic) and (2) knee-to-nose (normalized difference), calculated as

$$\frac{\text{time}_{\text{paretic}} - \text{time}_{\text{nonparetic}}}{\text{time}_{\text{paretic}}}.$$

Table 3.1. Participant demographics for experiment 1.

| Ball-in-Bowl | Nail-and-Hammer | Age | Stroke Side | Hand Dominance | UE-FMA | knee-to-nose |
|---|---|---|---|---|---|---|
| Y | Y | 68 | left | right | 51 | 4.77 |
| Y | Y | 64 | right | right | 49 | 5.1 |
| N | Y | 63 | right | right | 31 | 13.86 |
| Y | Y | 55 | left | right | 30 | 6.87 |
| Y | Y | 65 | right | left | 52 | 5.87 |
| N | Y | 57 | right | right | 58 | 3.84 |
| Y | N | 54 | left | right | 17 | – |
| Y | Y | 72 | right | right | 13 | – |
| Y | Y | 51 | left | right | 32 | 29.4 |
| N | Y | 61 | right | right | 26 | 11.94 |
| Y | N | 63 | right | right | 7 | – |
| Y | Y | 75 | right | left | 9 | – |
| Y | Y | 66 | left | right | 21 | 12.9 |

The knee-to-nose test is one of the few components of the UE-FMA that assesses speed and coordination, necessary for dynamic motion.

## 3.4. Experimental methods

We analyze data from two experimental studies. In experiment 1, we recruit chronic survivors of stroke and have them complete the ball-in-bowl and nail-and-hammer tasks (n=13). In experiment 2, we analyze data from a prior study [**183**]. This study assesses survivors of stroke in the acute inpatient setting (<100 days since stroke) in the object hit task (n=48).

### 3.4.1. Experiment 1: Participants

We recruited thirteen individuals to complete the study. All participants expressed verbal and written consent to participate. The study protocol was approved by the Northwestern University Review Board under IRB STU00021840. The group of tested individuals was 51-75 years of age (with an average of 63). There were 4 females and 9 males; 11 were right-hand dominant, 2 were left-hand dominant.

All individuals were chronic survivors of hemiparetic stroke. The time from their stroke ranged from 4 to 27 years (with an average of 15 years). We recruited participants with varying levels of impairment, as assessed by the UE-FMA. The group of study participants included 4 individuals with mild impairment (UE-FMA $> 45$) and 9 individuals with moderate-severe impairment (UE-FMA $\leq 45$).

All participants completed the same amount of trials on each side. To avoid a bias from fatigue and/or task learning, we randomly assigned the arm with which individuals began performing our experimental tasks. Due to time constraints, two participants did not complete the nail-and-hammer task, as noted in Table 3.1. Due to an inability to understand the task, three participants were not included in the ball-in-bowl analysis, as noted in Table 3.1 and explained in more detail in the Appendix.

### 3.4.2. Experiment 1: Setup

A haptic environment is created using the Arm Coordination Training 3-degree-of-freedom device (ACT-3D) [**101**]. The ACT-3D is the combination of an admittance controlled HapticMASTER robot, a 6-degree-of-freedom load cell (JR3 load cell, Woodland, CA) at the robot's end-effector, and a Biodex chair. The participant's forearm is attached to the load cell using a forearm-wrist-hand orthosis. This setup, as shown in Fig. 3.3, allows the participant to directly control the location of the end-effector with movements of their arm. Active movements are captured as force readings on the load cell and, using the robot's internal controller, translated into movements of the end-effector in 3D space.

The robot is able to provide partial or full support of the arm against gravity by modulating abduction loading at the shoulder. It is also able to render haptic objects and forces in the upper-limb workspace of the individual. A virtual task is visualized on a screen in front of the participant. Forces exerted by study participants are recorded at 20Hz for the ball-in-bowl task and 50Hz for the nail-and-hammer task.

Participants can move in 3 dimensions. Their $xy$-motion is mapped directly onto the location of the virtual bowl or virtual hammer. In the z direction, participants start from a home position $z = 0$, resting on a haptic table, rendered by the robot. During task attempts, participants are asked to keep their arm lifted anywhere above the haptic table to avoid imposing dynamic constraints on the ball-in-bowl system. Their arm weight is either fully supported against gravity—the load cell readout in the z direction is near zero, making the arm feel buoyant in space—or they experience a force equivalent to 35% of their maximum shoulder abduction—on average for a stroke survivor this loading is equivalent to supporting arm weight. Movement in the z direction does not affect the simulation.

### 3.4.3. Experiment 1: Procedure

Each participant comes in for two sessions. During one of the sessions, a licensed physical therapist conducts a Fugl-Meyer Assessment for the Upper Extremity. We work with the same therapist for the whole study to limit variability in the clinical scores from inter-rater variability.

An experimental session starts with measuring arm weight and shoulder strength using the robot. We position the participant with their shoulder resting on the backrest, shoulder abduction at 80 degrees, horizonal shoulder abduction at 40 degrees, and elbow flexion at 90 degrees. We ask the participant to attempt to lift their forearm as hard as they can, while we set the robot to emulate a very stiff spring (k=1000). We repeat this measurement 3 times and record the highest measured force in the Z-direction. We then define the individual's reachable workspace by asking each participant to "clean" the virtual table. Participants are instructed to cover the biggest area on the table that they can reach without allowing their shoulder to leave the back of the Biodex chair. During later task attempts, targets are placed only within this reachable area.

We then proceed to a training period, when we explain the tasks. We allow numerous attempts at the tasks until the participant feels comfortable with the experimental setup and they seem to understand the task objective. We offer tips and guidance as well as encourage the participant to ask questions. The training period usually takes 15-20 minutes with at least 10 task attempts.

All individuals complete the ball-in-bowl task at four frequencies (0.5Hz, 1Hz, 1.5Hz, and 2.5Hz) and under two task conditions (with/without loading). Each experimental set of conditions is attempted by the participant 9 times: in 1 set of 5 during the first session and 1 set of 4 during the second session. Task frequency and loading level are selected in random order to control for potential effects of learning and/or fatigue. Each study volunteer performs a total of 154 task attempts with each attempt lasting 30 seconds. We end the second experimental session with the nail-and-hammer task. Each participant performs 24 attempts at the task (6 with each arm at each loading level); one attempt lasts 20 seconds. We take at least two 5-minute breaks during each session, as well as shorter breaks throughout the experiment as needed.

### 3.4.4. Experiment 2: Participants

A total of 48 subjects were analyzed for the object hit task. All participants were individuals with acute hemiparetic stroke within the first 100 days of the incident and an average of two weeks from the incident. Participants included patients recruited from Providence Care (St. Mary's of the Lake Hospital, Kingston, ON). The study was approved by the Queen's University Health Sciences and Affiliated Teaching Hospitals Research Ethics Board (#ANAT-024-05) and subjects provided informed consent. More information about subject recruitment and demographic data can be found in prior work [**183**].

### 3.4.5. Experiment 2: Setup & Procedure

The behavioural task was performed using a bimanual exoskeleton robot which measures limb motion (KINARM, BKIN Technologies Ltd, Kingston, ON, Canada). Participants sat in a modified wheelchair base, and their arms were fitted in supports permitting movement in the horizontal plane. Arm supports were adjusted such that the robot's linkages aligned with the subject's elbows and shoulders. Subjects received visual feedback from a virtual reality system which displayed fingertip position and virtual objects in the same plane as arm motion via a two-way mirror. Direct vision of the hands and arms was occluded. The experimental setup is visualized in Fig. 3.6. Functionally, the setup was similar to the setup in experiment 1 while allowing for bimanual motion. Positions and velocities of the hands and balls were recorded with a sampling frequency of 200 Hz.

### 3.4.6. Data analysis

Aggregate frequency spectra presented in this chapter are averaged across participants and renormalized. In creating the boxplot figures, we average across trials of a participant under the same trial conditions so that each boxplot demonstrates the spread in performance across

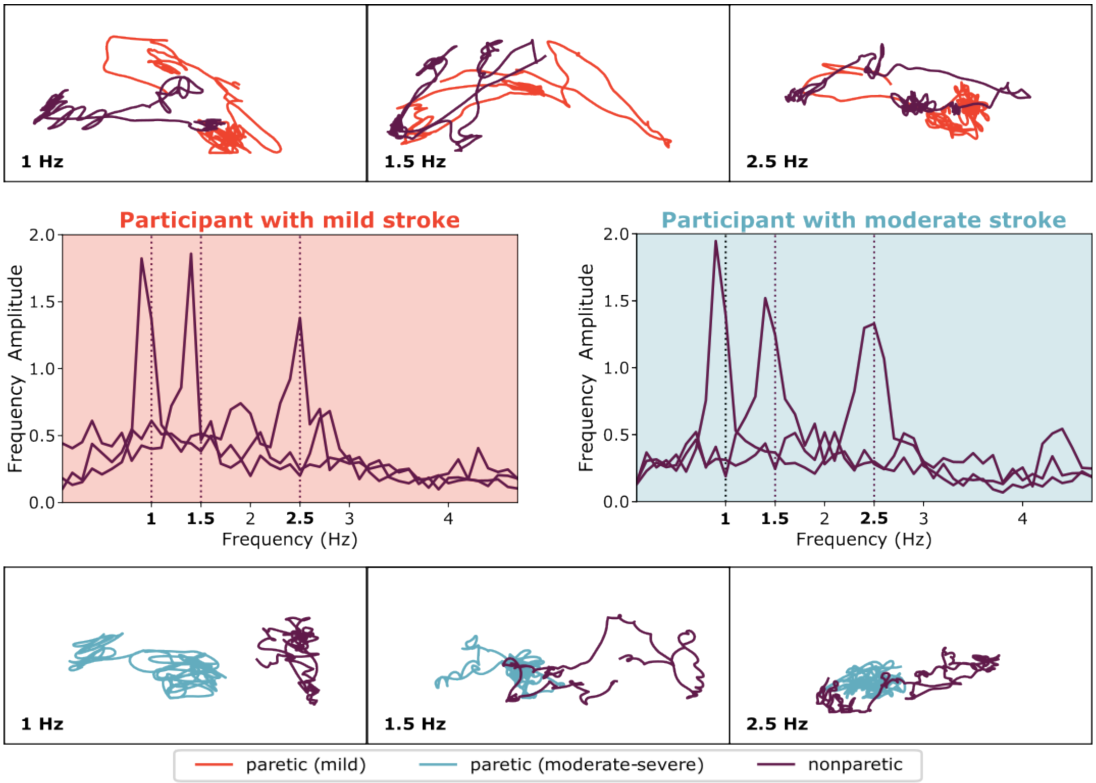

Figure 3.4. **Example trajectories and frequency spectra for individual trials during the ball-in-bowl task.** While individuals with a more severe impairment (blue curves) have a visibly smaller workspace, their movement within the workspace looks qualitatively similar. Frequency-based metrics can help assess motion within the workspace by characterizing movement dynamics—note that there are visible peaks in energy exerted near resonance for each trial. All trials were completed in the fully supported condition; for the frequency spectra, resonant frequencies for a given trial are indicated with a dotted line. Performance during trials with the nonparetic arm is used to confirm that a participant understands the task, as explained in more detail in Appendix B.1.

participants. All statistical analyses are performed on unaggregated and unfiltered frequency spectra and metrics.

Our experiment aims to test four hypotheses: (H1) frequency-based metrics capture a reduction in motion bandwidth due to a stroke across a range of dynamic tasks, (H2) there is a relationship between clinically determined levels of impairment (as characterized using existing clinical assessments) and a reduction in motion bandwidth (measured using QADR), (H3) stroke affects motion bandwidth, reducing ability to generate motion at higher frequencies (above 1.5Hz), and (H4) loading exacerbates loss in dynamic function after a stroke.

To assess H1, we perform two statistical analyses. We use a one-way rm-ANOVA for the median frequency metric for the nail-and-hammer task and a one-way ANOVA for the object hit task with arm (paretic/nonparetic) as a factor.

To assess H2, we perform three statistical analyses. We repeat the rm-ANOVA for the median frequency metric for the nail-and-hammer task with stroke severity (mild/moderate-severe) as a factor, in addition to arm (paretic/nonparetic) and loading level. We run a post-hoc t-test with a Bonferroni correction for multiple comparisons to compare the median frequency on the paretic side for mild and moderate-severe participants. Secondly, we calculate the Pearson R and Spearman rank correlation coefficients for the nail-and-hammer task—we test the correlations between clinical assessment scores and frequency-based task metrics.

To assess H3, in the ball-in-bowl task we compare energy@resonance during trials completed using moderate-severe participants' paretic arm to trials completed with their nonparetic arm. For each task frequency, we fit a linear mixed model using the lmer function [**11**] in R [**150**] following Winter's guidance [**194**] for handling repeated measures experiments. The linear mixed model provides the flexibility to exclude 19 trials with less than 1.25s where the ball is unsafe and the arm is lifted. We use the following expression:

$$\text{E@R}\sim\text{Arm*Loading}+(1+\text{Arm*Loading}\,|\,\text{Participant}),$$

with fixed effects for arm, loading, and the interaction between arm and loading as well as random intercepts and slopes within each participant and experimental condition. We use Wald

Chi-squared tests to evaluate for statistical significance; similar to an ANOVA, the Wald Chi-squared test evaluates whether a given factor explains some of the variation in an outcome measure. Secondly, we perform a one-way rm-ANOVA for the effect of task frequency on the percent loss in function measure in the ball-in-bowl task, followed by paired post-hoc t-tests with a Bonferroni correction for multiple comparisons. In the other two tasks, we use paired t-tests on the average amplitude metric comparing performance on the nonparetic side to performance on the paretic side across frequencies between 0.5Hz and 2.75Hz. For the nail-and-hammer task, we run separate t-tests for participants with mild and moderate-severe stroke. For the object hit task, we run an ANOVA with arm (paretic/nonparetic) and trial half (first/second) as factors.

To assess H4, we look at participants with severe stroke and the 1.5Hz task. For each arm, we fit a linear mixed model using the expression,

$$\mathrm{E@R} \sim \mathrm{Loading} + (1 + \mathrm{Loading} \,|\, \mathrm{Participant}),$$

with a fixed effect for loading as well as random intercepts and slopes within each participant and loading condition. We use Wald Chi-squared tests to evaluate for statistical significance.

For each statistical test, we evaluate the assumptions using the Shapiro-Wilk test for normality and Mauchly's test for sphericity.

## 3.5. Results

### 3.5.1. Frequency-Based Metrics Capture Post-Stroke Reduction in Motion Bandwidth

We use QADR to quantify changes in motion bandwidth due to a hemiparetic stroke. As part of experiment 1, participants with chronic stroke (n=10) perform the ball-in-bowl task under three task frequencies: 1Hz, 1.5Hz, and 2.5Hz. Example task trajectories are shown in Fig. 3.4. While individuals with a more severe impairment have a visibly smaller workspace in the plots in Fig. 3.4, their movement within the workspace looks qualitatively similar. As described throughout this article, frequency-based metrics can help assess motion within the participant's reachable workspace. The frequency spectra for the ball-in-bowl trials are easier to interpret than time-series trajectories—we see discernible peaks near resonance when visualizing task trajectories in the frequency domain (refer to Fig. 3.4). By quantifying the magnitude of the peaks near resonance in the ball-in-bowl task, we observe that a moderate-severe motor impairment following a stroke impacts participants' ability to generate motion within their reachable workspace at frequencies above 1Hz. This change in motion bandwidth is visible in the aggregate frequency spectra visible in Fig. 3.7 and is discussed in more detail in later subsections.

Secondly, we ask stroke survivors to complete the nail-and-hammer task. In Fig. 3.5, we illustrate the aggregate frequency spectra of all participants (n=11). Moderate-to-severe stroke (indicated with the color blue in Fig. 3.5) causes participants to exert less energy in the 2-4Hz bandwidth and more in the 0.5-2Hz bandwidth. For participants with mild stroke (plotted in red in Fig. 3.5), motion frequency is less affected, particularly for frequencies above 2Hz.

For each trial, we calculate the median frequency of motion during that trial. We observe that in this free-form task, the median hammering frequency decreases from values around 2Hz for the nonparetic arm to values around 1.7Hz for the paretic arm, as illustrated in the boxplot on the right side of Fig. 3.5. In a rm-ANOVA, we observe that the median frequency of motion is significantly reduced following stroke ($F = 11.86$, $p = 0.007$). These results suggest that a frequency-based metric during a dynamic task, i.e., median frequency of motion during nail hammering, can capture motion impairment due to a hemiparetic stroke.

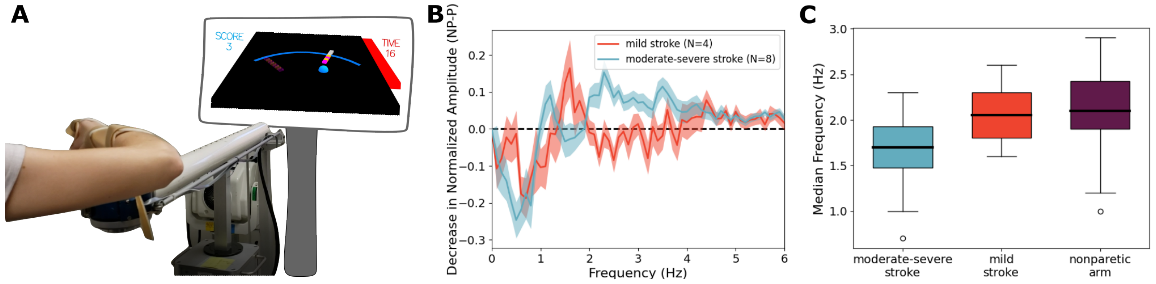

Figure 3.5. **Stroke lowers the default frequency of fast directional movement.** (A) Experimental setup for the nail-and-hammer task. (B) Stroke survivors, particularly those with moderate-severe stroke (blue curve), exert more high-frequency motion (above 2Hz) and less low-frequency motion (below 1Hz) with their nonparetic arm, when completing the nail-and-hammer task. Individuals with mild stroke (red curve) are less affected. P indicates paretic; NP indicates nonparetic. (C) We observe that stroke significantly reduces the median frequency of motion ($p < 0.001$) during a simulated nail-hammering task. Mild stroke survivors are less affected than moderate-severe individuals post-stroke ($p < 0.001$).

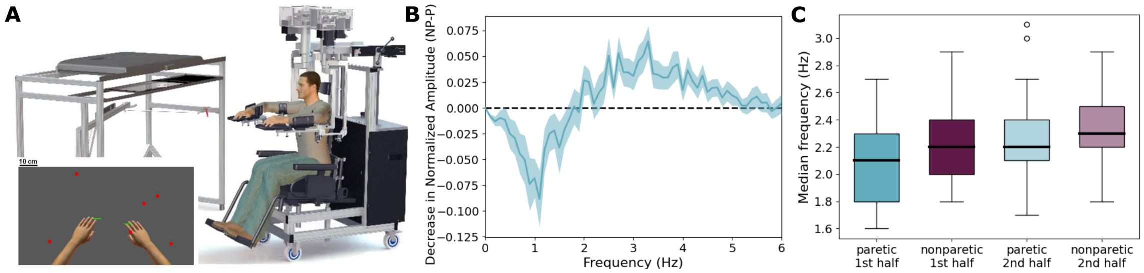

Figure 3.6. **Even in a task that was not designed for frequency analysis of movement, frequency-based metrics capture dynamic loss due to a stroke.** (A) Experimental setup for the object hit task. (B) Stroke survivors exert more high-frequency motion (above 2Hz) and less low-frequency motion (below 1Hz) with their nonparetic arm, when completing the object hit task. (C) We observe that stroke significantly reduces the median frequency of motion ($p < 0.001$) during the object hit task (introduced by Tyryshkin *et al.* [**183**]). Stroke affects movement differently during the first half of the task, when targets appear slowly starting at 0.5Hz, than the second half of the task, when targets appear at a higher rate up to 3.5Hz ($p = 0.01$).

As part of experiment 2, we analyze motion from the object hit task from a prior study [**183**]. Aggregate frequency spectra are visualized in Fig. 3.6. Again, note that study participants exert less energy below 2Hz and more energy above 2Hz with their paretic arm than with their nonparetic arm. For each trial, we calculate the median frequency of the forces exerted by the participants (n=48). In a rm-ANOVA, arm (paretic/nonparetic) is a significant factor for

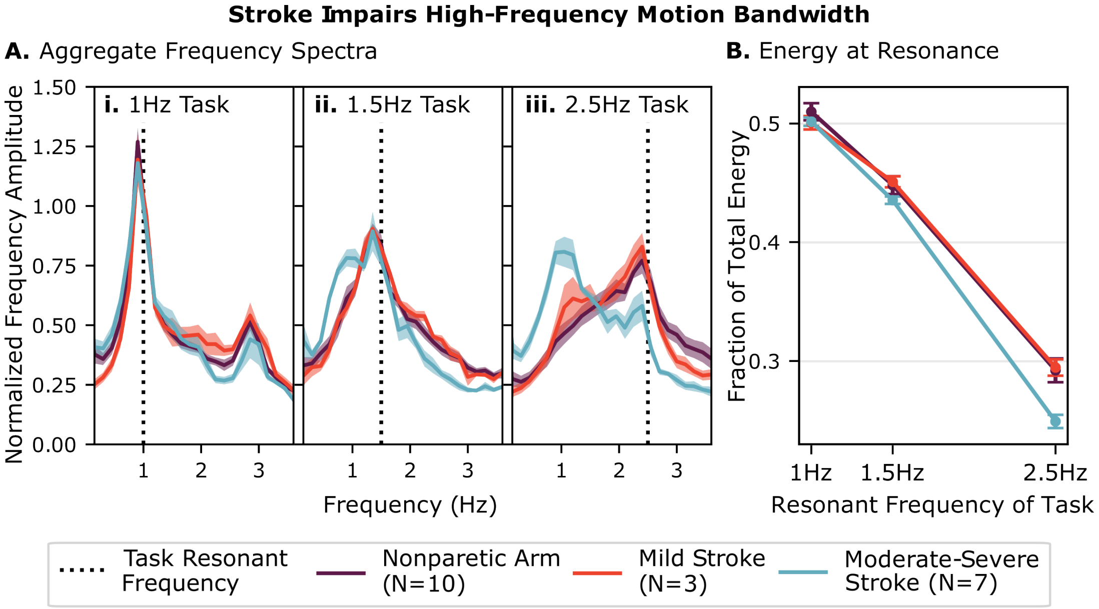

Figure 3.7. **Stroke affects dynamic response at higher frequencies ($\geq$1.5Hz).** (A) All participants are able to generate motion at the resonant frequency of the task with their nonparetic arm (purple curve). Participants with mild stroke achieve similar performance at moving at the task's resonant frequency using their paretic arm (red curve) as their nonparetic arm (purple curve). Participants with moderate-severe stroke (blue curve) have difficulty generating motion above 1Hz. Even when the task frequency increases, moderate-severe participants continue exerting a significant proportion of their energy around 1Hz. (B) The energy@resonance metric quantifies the decline in motion bandwidth. At 2.5Hz, arm significantly affects energy at resonance for moderate-severe participants ($p < 0.001$). Error bars represent standard error across participants.

the object hit task ($F = 9.19, p = 0.004$). These results suggest that a frequency-based metric can be used to quantify impairment even during a dynamic task that was not designed for a frequency-based analysis of motion.

**Across all three dynamic tasks, frequency-based metrics are sensitive to loss of function due to a stroke.**

### 3.5.2. Frequency-based metrics correlate nonlinearly with clinical assessments

As another validation of a frequency-based metric, we test for its sensitivity to clinically diagnosed levels of impairment. At a gross level, we observe that the primary outcome metrics

Table 3.2. **Correlations between frequency-based metrics in the nail-and-hammer task and clinical assessments.** The table includes Spearman and Pearson correlation coefficients; p-values are listed in brackets.

| | paretic | | $\delta$(nonparetic-paretic) | |
|---|---|---|---|---|
| | Pearson | Spearman | Pearson | Spearman |
| UE-FMA | **-0.58 (0.05)** | **-0.74 (0.008)** | **0.6 (0.04)** | **0.65 (0.02)** |
| UE-FMA<br>(arm only) | -0.54 (0.07) | **-0.63 (0.03)** | **0.62 (0.04)** | **0.71 (0.01)** |
| knee-to-nose<br>(normalized difference) | 0.47 (0.12) | **0.59 (0.04)** | **-0.57 (0.05)** | **-0.64 (0.02)** |
| knee-to-nose<br>(paretic) | -0.37 (0.32) | 0.1 (0.8) | -0.26 (0.5) | -0.14 (0.7) |

in the nail-and-hammer task (median frequency) are significantly lower for individuals with moderate-severe stroke than for individuals with mild stroke ($t = -7.2, p < 0.001$). In a rm-ANOVA, severity (mild/moderate-severe) and arm (paretic/nonparetic) have an interaction effect ($F = 3.58, p = 0.07$). These results suggest that the frequency-based metrics are sensitive to clinical levels of impairment.

Secondly, we compare the median frequency in the nail-and-hammer task to participants' UE-FMA scores. Four versions of the UE-FMA score are considered: (1) full UE-FMA, (2) UE-FMA (arm only), (3) knee-to-nose (paretic), and (4) knee-to-nose (normalized difference). Two versions of the median frequency are considered: (1) paretic and (2) normalized difference between nonparetic and paretic. We do not calculate correlations for the object hit task, because UE-FMA scores were not available. We also do not calculate correlations for the ball-in-bowl task, because unlike the UE-FMA it does not offer a holistic assessment of motion bandwidth but rather a quantification of performance at a specific frequency.

Both frequency-based task metrics display significant moderate Pearson and Spearman correlations with the full UE-FMA (see Table 3.2). Many task metrics significantly correlate with the partial UE-FMA and knee-to-nose assessment. Our initial hypothesis was that there would be a higher correlation of task metrics with the dynamic knee-to-nose metric compared to the full UE-FMA. This is not the case. One plausible explanation is that some participants' impairment was so significant that they were not able to complete the knee-to-nose movement. Hence, we have less statistical power for the knee-to-nose correlation analysis. Overall, these results offer promising validation for frequency-based metrics, suggesting their adequacy for assessing changes in motion quality due to a neuromotor disorder.

Of note, loss in dynamic function seems to correlate *nonlinearly* with clinical measures. Across the board, the Spearman coefficients (which test for a monotonic relationship between two variables) are higher than Pearson correlations (which test for a linear relationship between two variables). While we observe higher levels of clinically diagnosed impairment to correlate with lower levels in dynamic performance, we do not observe frequency-based metrics to have an equally strong *linear* dependence on the UE-FMA score. This trend could be suggestive of the fact that there are aspects of dynamic movement that are not captured by the clinical assessment and that loss of dynamic function does not directly correlate to loss of static motion capabilities.

A similar trend is observable in the ball-in-bowl and nail-and-hammer analyses. We find that for individuals with mild stroke, in contrast to moderate-severe stroke, none of the primary metrics are significantly different for the paretic vs. nonparetic arm. As an example, we see in Fig. 3.7 that the energy@resonance metric overlaps for the mild paretic and nonparetic arms across all three tested task frequencies. Similarly, we find that the median frequency of nail

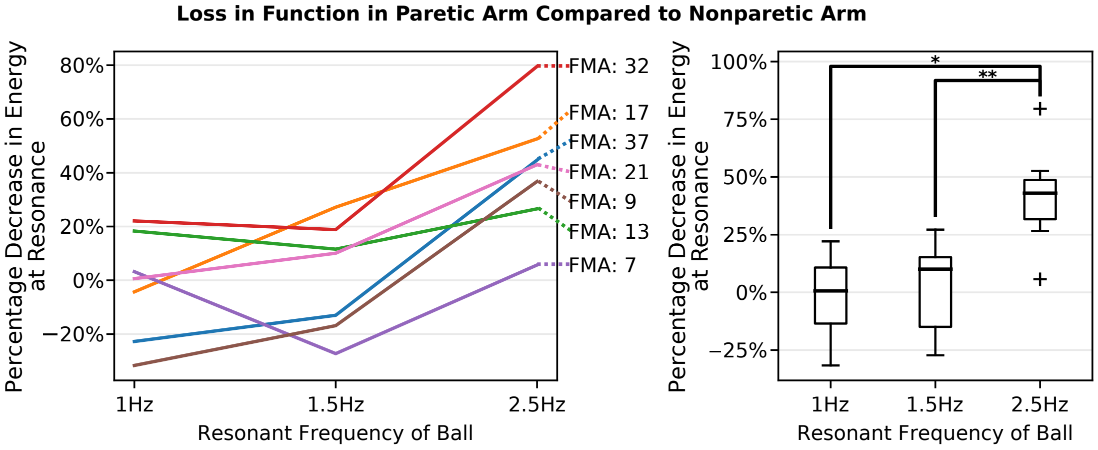

Figure 3.8. **Stroke survivors experience an increasing loss in function in their paretic upper limb with increasing frequency.** The whiskers on the box plots illustrate inter-subject variability. Note that with increasing frequency, functional loss increases. [TBD] Report statistics here. Maybe a paired t-test for 1Hz, 1.5Hz, 2.5Hz?

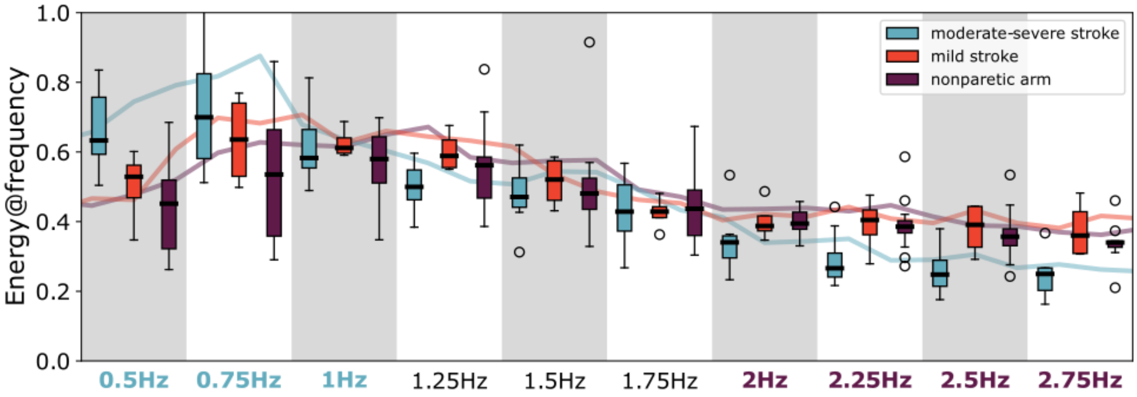

Figure 3.9. **In a task without resonance, we observe a decrease in movement at frequencies above 1Hz.** In the nail-and-hammer task, individuals with moderate-severe stroke exert less energy above 2Hz on their paretic side than their nonparetic side (blue vs. purple boxes), and more energy below 1Hz. Motion bandwidth is mostly preserved for individuals with mild stroke (red vs. purple boxes).

hammering is near 2Hz for both paretic and nonparetic arms for individuals with mild stroke, in contrast to individuals with moderate-severe stroke (refer to Fig. 3.5).

**Impairment severity correlates moderately with loss in dynamic function. The correlation appears to be nonlinear.**

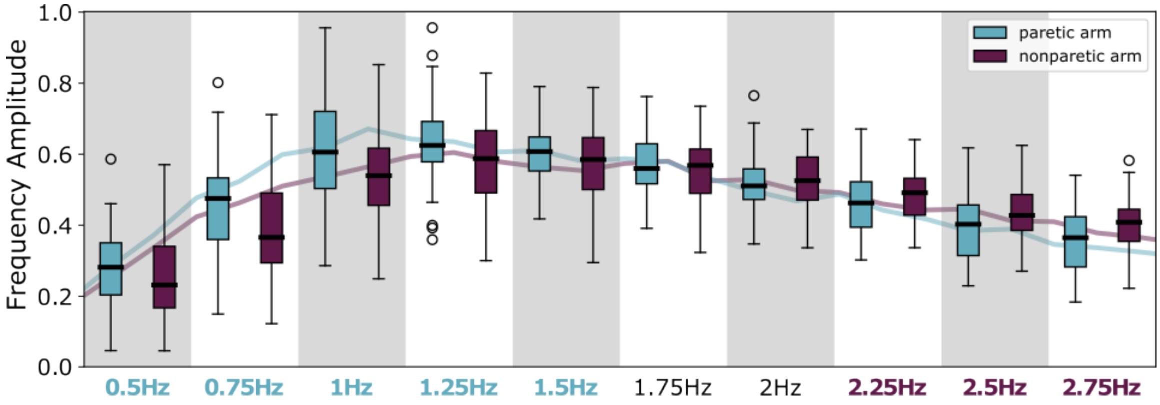

Figure 3.10. **In a task not designed for frequency analysis, we observe a decrease in movement at frequencies above 1.5Hz.** In the object-hit task, individuals with stroke exert less energy above 2.25Hz on their paretic side than their nonparetic side (blue vs. purple boxes), and more energy below 1.5Hz.

### 3.5.3. QADR Reveals a Post-Stroke Effect on High-Frequency Motion

We calculate energy@resonance for each participant during individual trials for three task frequencies in the ball-in-bowl task (see Fig. 3.7). Arm has a significant effect on the energy at resonance for the 2.5Hz task ($\chi^2 = 12.4, p < 0.001$), while it does not result in a significant effect on the lower frequency tasks (1Hz ($\chi^2 = 0.0, p = 0.970$) and 1.5Hz ($\chi^2 = 0.2, p = 0.645$)). As visible in Fig. 3.7C, individuals with moderate-severe stroke exhibit a peak to the left of 2.5Hz, suggesting they are experiencing difficulty moving fast enough to match the 2.5Hz resonant frequency. Similar behavior is exhibited by individuals with moderate-severe stroke for the 1.5Hz task (see Fig. 3.7B), even though there is not yet a statistically significant decrease for energy@resonance. Lastly, there is an interaction effect between arm (paretic/nonparetic) and task frequency ($\chi^2 = 15.9, p = 0.001$), suggesting that a unilateral brain injury due to a stroke affects performance differently at different frequencies.

As a secondary metric for the ball-in-bowl task, we analyze the percentage decrease in energy@resonance in the paretic arm compared to the nonparetic arm. Fig. 3.8 visualizes the metric for participants with moderate-severe stroke. Overall, we find that the percent loss increases with increasing task frequency. While there is some variability among participants, the median loss in function is around 0% at 1Hz, around 10% at 1.5Hz and around 40% at 2.5Hz, as illustrated in Fig. 3.8. An ANOVA confirms that the resonant frequency of the ball has a statistically significant effect on the percent loss in function ($F = 16.9, p < 0.001$). The percent loss in function is significantly higher at 2.5Hz than it is at 1Hz ($p = 0.017$) or at 1.5Hz ($p = 0.003$).

In the nail-and-hammer task, we calculate an average amplitude metric at ten task frequencies: 0.5Hz, 0.75Hz, 1Hz, 1.25Hz, 1.5Hz, 1.75Hz, 2Hz, 2.25Hz, 2.5Hz, 2.75Hz (see Fig. 3.9). In a series of paired t-tests (reported in Table 3.3), we observe similar trends in performance as in the ball-in-bowl task. At frequencies up to 1Hz, participants with moderate-severe stroke exert more energy on their paretic side. At 2Hz, the trend flips—participants with moderate-severe stroke exert more energy with their nonparetic arm compared to their paretic arm at frequencies above 2Hz.

The trend is again confirmed using data from experiment 2. The object hit task starts out as a low-frequency task (with targets appearing at a rate of 0.5Hz) and gradually becomes a high-frequency task (with objects appearing at a rate of 3.5Hz). In our analysis, we separately calculate metrics for the first half of a trial and for the second half of a trial. As expected, in a rm-ANOVA, arm (paretic/nonparetic) is a significant factor ($F = 9.19$,$p = 0.004$), as is trial half (first/second) ($F = 42.0$,$p < 0.001$). Interestingly, there is an interaction effect between arm and half of trial ($F = 3.72$,$p = 0.06$), suggesting that stroke affects performance in the first half (low-frequency) differently than in the second half (high-frequency).

In the object hit task, again we calculate an average amplitude metric at ten task frequencies: 0.5Hz, 0.75Hz, 1Hz, 1.25Hz, 1.5Hz, 1.75Hz, 2Hz, 2.25Hz, 2.5Hz, 2.75Hz (see Fig. 3.10). In a

Table 3.3. **Paired t-tests for average amplitude metrics in the nail-and-hammer and object hit tasks.** We list the difference in means between the nonparetic and paretic arm; p-values are listed in brackets.

| | 0.5Hz | 0.75Hz | 1Hz | 1.25Hz | 1.5Hz | 1.75Hz | 2Hz | 2.25Hz | 2.5Hz | 2.75Hz |
|---|---|---|---|---|---|---|---|---|---|---|
| nail-and-hammer task (mild) | **-0.06 (0.01)** | **-0.1 (<0.001)** | **-0.06 (0.01)** | -0.003 (0.92) | 0.06 (0.06) | 0.05 (0.06) | 0.006 (0.72) | -0.015 (0.23) | -0.02 (0.16) | 0.002 (0.86) |
| nail-and-hammer task (mod-sev) | **-0.18 (<0.001)** | **-0.15 (<0.001)** | -0.04 (0.06) | 0.01 (0.57) | -0.02 (0.35) | -0.008 (0.59) | **0.05 (<0.001)** | **0.1 (<0.001)** | **0.12 (<0.001)** | **0.12 (<0.001)** |
| object hit task | **-0.02 (0.02)** | **-0.05 (<0.001)** | **-0.07 (<0.001)** | **-0.04 (0.002)** | **-0.02 (0.02)** | -0.007 (0.49) | 0.1 (0.16) | **0.02 (0.02)** | **0.3 (0.006)** | **0.04 (<0.001)** |

series of paired t-tests (reported in Table 3.3), we observe similar trends in performance as in the other two tasks. At frequencies up to 1.5Hz, participants with moderate-severe stroke, exert more energy on their paretic side. At 2.25Hz, the trend flips—participants with moderate-severe stroke exert more energy with their nonparetic arm compared to their paretic arm at frequencies above 2.25Hz.

**Across all three dynamic tasks, we observe that individuals with moderate-severe stroke are able to generate motion up to 1Hz; they begin to experience difficulty with motions in the range 1.25-2Hz, while motion above 2Hz is significantly impacted.**

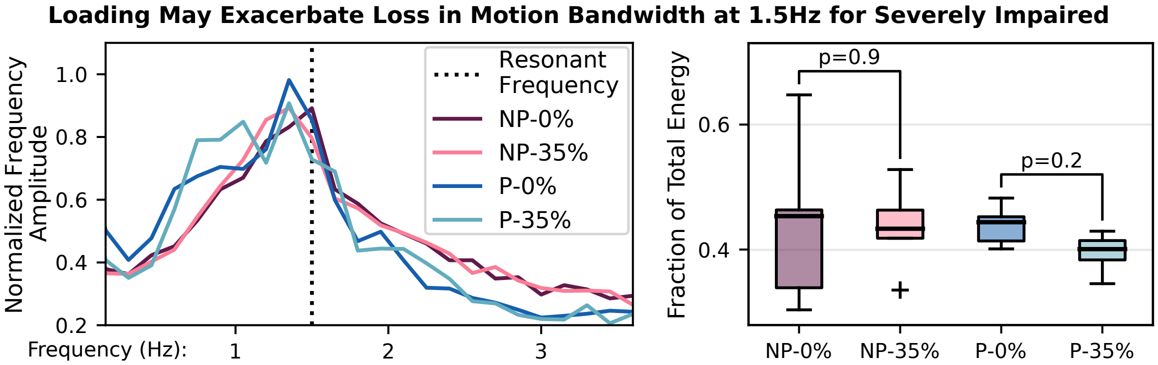

Figure 3.11. **Shoulder abduction loading—previously shown to cause an increased recruitment of indirect pathways—may be contributing to a decrease in motion bandwidth.** Here, we plot the aggregate frequency spectra for the 1.5Hz ball-in-bowl task, separated by loading condition; we choose 1.5Hz because it is the only tested frequency in the ball-in-bowl task where stroke affects performance but participants are still able to complete the task. Note that while the peak for the paretic side without loading (dark blue curve) overlaps with the nonparetic peaks (purple and pink curves), the paretic peak with loading (light blue curve) is less pronounced with a secondary peak below resonance. Energy@resonance on the paretic side with loading is on average lower than energy@resonance on the paretic side without loading (p=0.2). These results suggest that an increased use of indirect pathways might be contributing to a reduction in motion bandwidth. P indicates paretic; NP indicates nonparetic.

### 3.5.4. An Increased Reliance on Indirect Pathways Could Be Contributing to a Decrease in Motion Bandwidth

Lastly, we analyze the impact of loading on dynamic performance. To study the impact of loading, we study participants with severe stroke (n=5) in the ball-in-bowl task with a resonant frequency of 1.5Hz. In the prior analyses, we observed that at this frequency, moderate-severe stroke survivors start to experience loss of function in their paretic limb but they are still able to generate some motion at resonance. In contrast, at 1Hz all stroke survivors are able to complete the task—there is no difference in energy@resonance for the paretic or nonparetic arm ($p = 0.97$). Conversely, at 2.5Hz, stroke survivors are unable to complete the task—energy@resonance is significantly lower for the paretic arm than the nonparetic arm ($p < 0.001$). The 1.5Hz task is the only experimental condition in which moderate-severe participants can reliably complete the task with their paretic upper limb but are beginning to experience the impact of stroke. We assume that this trend is even stronger for severe-only participants, increasing sensitivity to the loading condition.

In our analysis of the 1.5Hz task, we observe that loading may exacerbate loss in motion bandwidth due to a hemiparetic stroke. The impact of loading is qualitatively visible in Fig. 3.11. Note that while the peak for the paretic side without loading (dark blue curve) overlaps with the nonparetic peaks (purple and pink curves), the paretic peak with loading (light blue curve) is less pronounced with a secondary peak visible at a frequency lower than the resonant frequency. The statistical analysis confirms this trend. In the 1.5Hz task, we observe that loading does not impact energy@resonance on the nonparetic side ($\chi^2 = 0.02, p = 0.914$); on the paretic side, the effect of loading on energy@resonance is stronger ($\chi^2 = 1.65, p = 0.199$), although not statistically significant.

**These results suggest that an increased use of indirect pathways contributes to a reduction in motion bandwidth, in line with our initial findings in a prior study [98]. Future work will investigate this further.** In prior work, we tested six severely impaired stroke survivors in the ball-in-bowl task with a task frequency of 1.88Hz and loading equivalent to over 50% of maximum shoulder abduction, finding statistical differences in dynamic performance [**98**]. In a follow-up study, the task design and experimental procedure were improved and experimentally validated with a group of unimpaired individuals [**101**]. Future experiments—that aim to evaluate the impact of loading on dynamic performance—should focus on assessing severely impaired participants at frequencies between 1.5-2Hz with loading levels at ~30-50%. It is likely that mild participants would benefit from being evaluated at higher frequencies above 2.5Hz at loading levels at ~50%.

### 3.5.5. Discussion

The functional loss in motion bandwidth that we quantify using QADR is a manifestation of an underlying change in motor control due to a stroke. There exist at least three well-studied mechanistic changes post-stroke that could be contributing to a reduction in motion bandwidth.

After a stroke, the cortico-reticulospinal tract becomes more active due to cortical reorganization [**45**]. Prior work suggests that stroke survivors recruit indirect cortico-reticulospinal pathways for control of voluntary motion as a way to compensate for the severed lateral corticospinal and corticobulbar projections [**129**]. Unlike lateral corticospinal projections, ipsilateral cortico-reticulospinal projections pass through the reticular formation in the brainstem and spinal cord, which consists of an interconnected network of synapses [**190**]. Due to this difference in structure of the motor pathways, the lag time between a cortical signal to initiate movement and muscle contraction is longer for reticulospinal pathways than for corticospinal projections [**123**].

Moreover, after initial engagement of muscle fibres, closed-loop dynamic tasks require fast repeated relaxation and re-engagement of the motor units. Prior work has found that motor unit firing rate modulation [**132**] and presynaptic inhibition [**112**] differs in individuals with stroke compared to able-bodied individuals, making it harder for stroke survivors to execute fast changes in motion direction where rapid muscle relaxation and re-activation are necessary.

In parallel, individuals with brain injury experience losses of cerebellar input to the cortex [**160**] due to lesions of cerebello-thalamo-cortical projections and cortical reorganization. This can impact dynamic performance, because the cerebellum plays an important role in determining timing of motion and overseeing motor coordination [**24**, **125**]. The cerebellum might also be involved in learning an inverse model of inter-joint dynamics [**12**], making it difficult for stroke survivors to incorporate limb dynamics into motion planning.

While multiple factors likely contribute to the observed changes in motion bandwidth post-stroke, further studies are necessary to examine the relative impact and establish a causal relationship with an observed functional decrease in dynamic performance. QADR and the frequency-based metrics introduced in this study can help with future research on the mechanistic causes of a decreased motion bandwidth after a unilateral stroke.

## 3.6. Future Work: Implications for Neuromotor Disorders

### 3.6.1. QADR and ADLs

While in this study we use QADR in a laboratory setting, frequency-based metrics of motion show promise of translating between clinical assessments and functional tasks. For one, the frequencies required to complete the ball-in-bowl task directly correlate with frequencies required to balance a cup of coffee during walking [**128**], carry a bag of groceries, or brush one's teeth. Based on a quantitative assessment of motion bandwidth, QADR can help predict which functional tasks will cause difficulty, as illustrated in Fig. 3.1. With existing clinical assessments, clinicians have limited insight about a patient's capacity for dynamic tasks. Secondly, a versatile metric of motion can equip clinicians with a continuous assessment tool for everyday tasks. As an example, the frequency-based metrics could be calculated using data from a wearable sensor for at-home monitoring during ADLs [**124**]. A quantitative, objective, and continuous measure for dynamic task performance that generalizes to in-clinic assessments and everyday tasks could support progress in the development of strategies for rehabilitation and assistance.

### 3.6.2. Understanding causes of dynamic deficit

Although many neuromotor conditions manifest themselves in similar functional deficits, including a decrease in motion bandwidth, they often have differing underlying causes that are not fully understood. We can use QADR in combination with other noninvasive diagnostic techniques to characterize abnormal processes occurring during motor control and correlate them with changes in dynamic performance. As an example, we can measure electromyographic (EMG) or electroencephalographic (EEG) signals during task completion. Similar to how clinical assessments, such as the FMA, allow us to correlate the degree of impairment with physiological changes, we can use QADR to study factors contributing to a decrease in motion bandwidth, enhancing our understanding of the mechanistic causes of neuromotor disorders.

The proposed approach can also be used to better understand dynamic response and motion bandwidth in able-bodied individuals. We know that real-time perception and processing of sensory information is crucial for effective motor coordination [**73**]. As such, we can study the speed of neural circuitry used to generate dynamic motion when relying on different modalities for sensory feedback. By adjusting the feedback provided to the participant—currently visual and haptic, we can compare the motion bandwidth available to able-bodied individuals under different modalities of sensory inputs. In impaired individuals, we can examine deficits in receiving and interpreting sensory feedback and determine how these deficits affect dynamic performance. Lastly, prior work suggests that each hemisphere contributes differently to motor control [**158**]. Motor lateralization causes differences in post-stroke deficits and might have implications for effective rehabilitation. Our platform enables us to further test the effect of the lesioned hemisphere on motor deficits and how it relates to limb dominance.

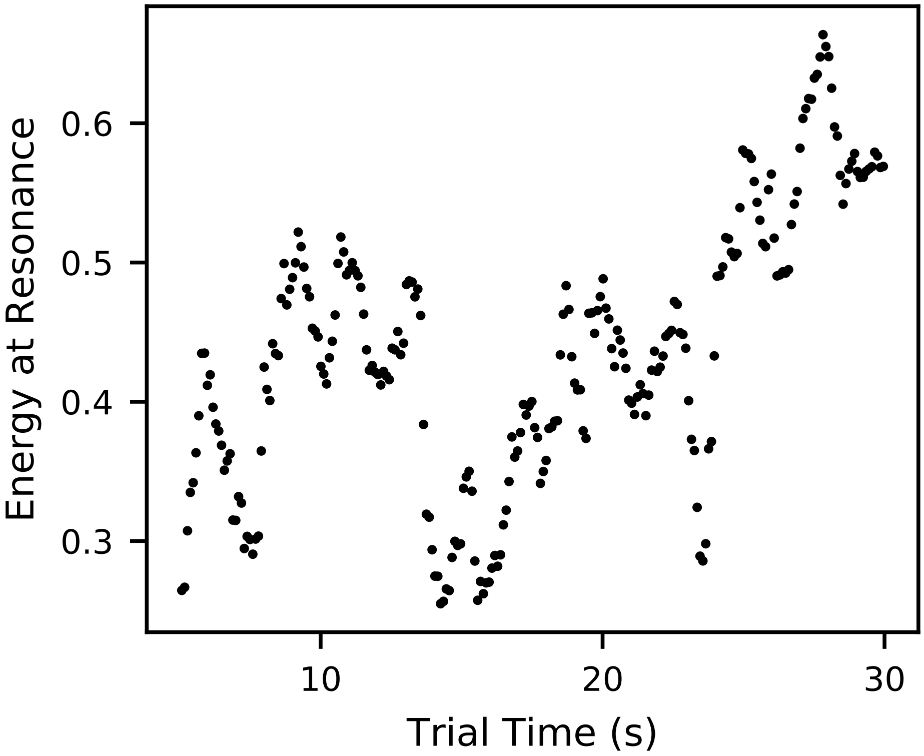

Figure 3.12. **A frequency-based metric could be used in real-time to adjust assistance.** While assist-as-needed paradigms have been shown to increase patient engagement during robot-assisted therapy, we lack real-time metrics of motion that could be used to adjust assistance during dynamic tasks. Frequency-based metrics are a promising candidate, because (a) they are feasible to compute in real-time, and (b) they do not constrain the task solution in time, compared to metrics such as error that assume an optimal solution trajectory.

By quantitatively assessing motion bandwidth, the QADR platform can improve our understanding of the underlying causes of impairment. In clinical practice, it can be used for (1) precise tracking of disease progression, (2) studying therapeutic efficacy, as well as (3) early diagnosis of deficits in movement coordination and motion bandwidth.

### 3.6.3. Therapeutic intervention

Stroke is the second highest cause of death globally and a leading cause of disability in developing countries [**32**]. Despite the development of various training protocols for motor recovery after stroke, rehabilitation treatments are still minimally effective, especially in more impaired individuals [**130**, **148**]. When used as an intervention, the QADR platform can become the foundation of therapeutic paradigms that are targeted to the impairment based on its underlying cause and assist-as-needed based on individual performance. The virtual environment can make therapy cognitively engaging for the patient and translational to real-world functional tasks.

QADR can quantify one's ability to generate dynamic movement in real time, making it suitable for assist-as-needed rehabilitation paradigms (as visualized in Fig. 3.12). For one, it is strategy-agnostic, making it possible to assist with movement without over-constraining it. Imagine a stirring task: one individual may prefer to stir clockwise while another may prefer to stir counterclockwise; both strategies are correct but present themselves differently in time. Secondly, frequency-based metrics are high-resolution—they can quantify one's ability to generate dynamic movement at a specific frequency or position in the workspace. Real-time, quantitative metrics make possible therapeutic strategies that target each patient's unique deficits, focusing on movements that are most affected by the disorder.

Adaptive assistance paradigms have been used to increase active engagement during therapy in many neuromotor disorders, such as stroke, spinal muscular atrophy (SMA) or cerebral palsy (CP). Providing targeted assistance while maintaining individuals' active engagement in movement is key to exploiting neuroplasticity and facilitating a successful recovery using remaining neural resources [**75**, **191**]. Because QADR is coupled with an assistive robot, we can modulate shoulder abduction loading. By providing partial or full support against gravity, our platform expands an individual's workspace and enables dynamic movements that are inaccessible in the unsupported condition [**178**]. In prior work on stroke, therapeutic interventions that rely on adaptive robotic assistance have led to improved rehabilitation outcomes compared to conventional therapy [**18**,**151**].

Finally, by centering physical therapy around functional tasks and employing a virtual environment, we can make therapy cognitively engaging for the patient [**34**] and directly relevant to daily activities. Frequency-based assessments naturally extend to a range of dynamic tasks—by employing virtual tasks with resonant frequencies (e.g. the ball-in-bowl task), we exercise dynamic response at specific frequencies; during free-form tasks without resonance (e.g. the nail-and-hammer or object hit task), we exercise maximum motion bandwidth for controlled directional motion. The proposed measure of dynamic performance is task-agnostic and easy to compute, supporting the creation of other real-world tasks for therapeutic settings. Most importantly, the measure is representative of functional motion—if an individual is unable to move at 2.5Hz to perform the virtual tasks presented in this chapter, they will not be capable of scrubbing a dirty dish or clapping in sync with a large crowd [**181**]. Therapies centered around dynamic tasks have the potential to improve quality of life for individuals with neuromotor impairments [**161**], particularly for survivors of stroke or traumatic brain injuries where the spinal cord and brainstem are still intact. Using the QADR platform as part of an intervention protocol, we can increase access to therapy, improve patient engagement, and enable targeted interventions—ultimately improving rehabilitation efficacy.

### 3.7. Conclusions

In this work, we contribute a methodology for quantifying an individual's ability to generate a *dynamic response*—controlled and timing-sensitive movements where the continuous modulation of motor activity is required to respond to real-time stimuli. We employ the method with post-stroke individuals and evaluate how unilateral brain injury due to a stroke impacts motion bandwidth within reachable workspace. We find that it causes a significant decrease at frequencies above 1.5Hz.

Our method provides a platform to evaluate how neuromotor impairments affect motion bandwidth more broadly, enabling the study of the underlying causes of a decrease in dynamic performance. QADR could also be used strictly as an assessment tool to diagnose delays in motor development in childhood [**171**] and to track disease progress or therapeutic efficacy in neuromotor disorders [**164**]. Long-term, it can form the basis for personalized robot-assisted interventions.

CHAPTER 4

# Emergence of communicative conventions

While it is known that communication facilitates cooperation in multi-agent settings, it is unclear how to design artificial agents that can learn to effectively and efficiently communicate with each other. Much research on communication emergence uses reinforcement learning (RL) and explores *unsituated* communication in one-step referential tasks—the tasks are not temporally interactive and lack time pressures typically present in natural communication. In these settings, agents may successfully learn to communicate, but they do not learn to exchange information concisely—they tend towards over-communication and an inefficient encoding. Here, we explore *situated* communication in a multi-step task, where the acting agent has to forgo an environmental action to communicate. Thus, we impose an opportunity cost on communication and mimic the real-world pressure of passing time. We compare communication emergence under this pressure against learning to communicate with a cost on articulation effort, implemented as a per-message penalty (fixed and progressively increasing). We find that while all tested pressures can disincentivise over-communication, *situated* communication does it most effectively and, unlike the cost on effort, does not negatively impact emergence. Implementing an opportunity cost on communication in a temporally extended environment is a step towards embodiment, and might be a pre-condition for incentivising efficient, human-like communication.

## 4.1. Introduction

Effective communication is a key skill for collaboration in a multi-agent setting [**44**]. As humans, we share communication protocols and cooperative conventions that have evolved over thousands of generations to optimize communication efficiency. As an example, we communicate in accordance with cooperative principles, such as Grice's maxims of conversation [**78**]. In line with the maxim of quantity, we are known to try to be as informative as possible, giving only as much information as is needed [**78**]. If future artificial systems are to cooperate with humans, it will be beneficial for their communication protocols to follow these patterns [**51**, **175**]. As a result, understanding the process of communication emergence and the pressures that shape the emergent communication protocols is of interest to the scientific community.

With a recent increase in available computational power, the field has seen a lot of progress with communication successfully emerging between reinforcement learning (RL) agents in a range of learning environments [**113**, **189**]. However, prior work shows that the emerged communication protocols often do not share properties of natural languages [**108**] and that artificial agents tend towards an anti-efficient encoding [**38**]. This likely happens because in many of the studied environments, communication is not *situated*—the action space of the agent does not include both communicative and environmental actions [**52**, **189**]. Agents do not learn to reason about *whether* to communicate; instead, communication is guaranteed and free to the agents. When free, excessive use of communication does not negatively affect the outcome of the game (or cause agent frustration), as it might in a real-world situation [**176**]. As a possible solution, we explore *situated* communication and show it is possible to obtain concise communication protocols by providing the agent with an action-communication trade-off.

In this work, we compare pressures that can incentivise conciseness during communication emergence. As our testbed, we use a cooperative multi-step navigation task with two RL agents. In the task, a speaker provides navigation hints to help a listener reach a goal within a gridworld maze. We explore three training regimes: (i) *unsituated* communication (cheap talk); the speaker sends a message to the listener at each timestep without any cost, similar to the communication paradigm in existing work [**38**, **114**, **119**], (ii) *unsituated* communication with a per-message penalty; the speaker experiences a cost on communication effort (either fixed or progressively increasing), similar to prior work [**153**] when fixed, (iii) *situated* communication as introduced by Wagner et al. (2003); the listener has to forgo an action to solicit information from the advising agent, experiencing an opportunity cost on communication. Using the collaborative navigation task, we evaluate how the different pressures in the three training regimes can incentivise sparse information sharing during communication emergence. We find that *situated* communication (regime iii) outperforms an internal cost on communication effort (regime ii) in terms of both conciseness of the emerged communication protocol and overall task performance.

## 4.2. Background

Initially, emergent communication between RL agents was largely studied in one-step referential games, such as the Lewis task [**38**,**43**,**90**,**114**,**119**]. The Lewis task [**118**] is a cooperative game, where the speaker sees an artifact (e.g., an image) and communicates a message from a fixed vocabulary (e.g., a symbol) to the listener, who then interprets the message to select a target item from among a set of distractors. This type of learning environment is known to successfully enable language development [**103**]. However, this setting does not recreate the temporal aspects of a real-world environment, which may influence the structure of the emergent languages.

Recent work increasingly explores multi-step tasks that enable temporally extended dialogue, more similar to real-world environments [**20**, **35**, **55**, **62**, **69**, **88**, **131**]. Cao et al. (2018) investigate multi-step negotiation. Jaques et al. (2019) test communication emergence during Sequential Social Dilemmas in environments, such as Harvest or Cleanup. Evtimova et al. (2018) propose a multi-modal, multi-step referential task (a modified version of the Lewis task). Bouchacourt and Baroni (2019) introduce a fruit-tool matching game, similar to a multi-step Lewis task but with a preference-based reward. Here, we build on this body of work by introducing a multi-step navigation task in a gridworld environment. In Appendix C.2, we elaborate on how our task compares to existing work and on the direct impact of introducing multi-step communication in this task.

The literature predominantly considers two types of communication: (i) communication through a cheap talk channel [**52**,**70**] and (2) communication through environmental actions [**170**]. In cheap talk, agents have a designated communication channel where they share messages at every timestep—the messages are free to the agents and agents do not reason about *whether* to communicate. Many authors find that selfish agents do not learn to effectively use an ungrounded cheap talk communication channel [**35**, **43**]. However, one can incentivise communication in a cheap talk channel by rewarding an intrinsic motivation to influence others [**88**] or by introducing a pro-social reward [**74**, **114**], as we do in our work. On the other hand, when communicating through actions, agents' environmental actions are made visible to other agents, enabling the environmental actions to be used for communication. Note that environmental actions are binding—they impact the state of the environment and the resulting reward, so agents do not have an inconsequential way to exchange information as with a cheap talk channel. In these scenarios, cooperation has a high success rate [**35**, **37**, **138**], but the communication is inflexible and directly connected with the task—there is mixed opinion whether agents are actually communicating [**120**]. In this work, we evaluate a third type of communication that we refer to as *situated* communication as introduced by Wagner et al. (2003) and used e.g. by Lowe et al. (2019) in matrix games, such as the Prisoner's Dilemma. When using *situated* communication, the listener has to actively choose between communicative and environmental actions. While only environmental actions directly affect the environment and allow the agent to obtain a reward, the agent can choose to forgo an environmental action to communicate.

In most studies, the emerged language structures are analyzed for shared commonalities with natural languages, such as compositionality or encoding efficiency. Although desired, it is nontrivial for such properties to emerge spontaneously [**108**]. As a result, researchers introduce pressures during training and/or structure the learning process to incentivise specific language properties. To encourage compositionality in the emergent communication protocol, one can introduce populations of agents [**39**, **154**] or the need for a language to be easily teachable [**119**]. Efficient communication can be incentivised by modifying the agents' reward structure, e.g., by adding an internal cost of articulation [**153**] or by prioritizing messages based on a metric of confidence [**198**]. Sparse communication has also been shown to emerge in mixed and competitive settings, where communicating too much might harm agent performance, e.g., in the case of prey coordinating an escape from predators [**172**]. In parallel, researchers have studied communication patterns during competitive games—e.g., auctions—where the capacity of the

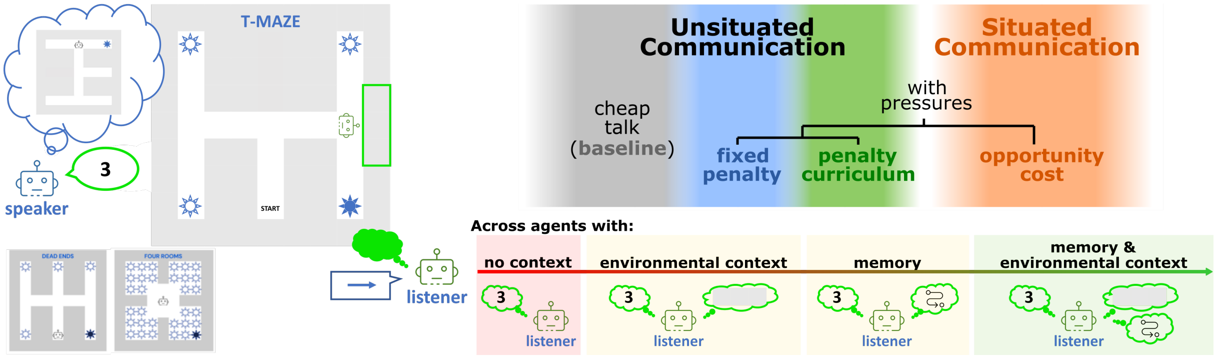

Figure 4.1. **Experimental setup.** (left) Gridworld environment & navigation task, (right) experimental conditions.

communication channel is externally bounded [**19**]. Here, we propose *situating* the communication in a multi-step task as a mechanism to shape the properties of the emergent communication protocol.

In our experiments, we study the emergence of communication in a cooperative multi-step navigation task. Like humans or robots that can only observe a small part of the world in their proximity, the listener has a limited view of its environment and has to rely on the speaker for guidance [**59**]. The task is an abstraction of a real-world task [**56**] where a person receives guidance from an oracle while navigating towards a goal. As our primary testbed, we consider a maze with T-junctions (i.e. junctions with a left/right turn) [**57**]. The well-defined decision points enable us to more easily quantify the amount of information exchanged by the agents.

### 4.3. Environment

We define a cooperative navigation task as a Markov Decision Process (MDP) with two RL agents. We test three environments: T-maze, dead ends, and four rooms, as shown in Fig. 4.1. The T-maze serves as our primary testbed. All three environments are set up as a pixel-based gridworld with a maze inside. Features of the world are encoded with binary

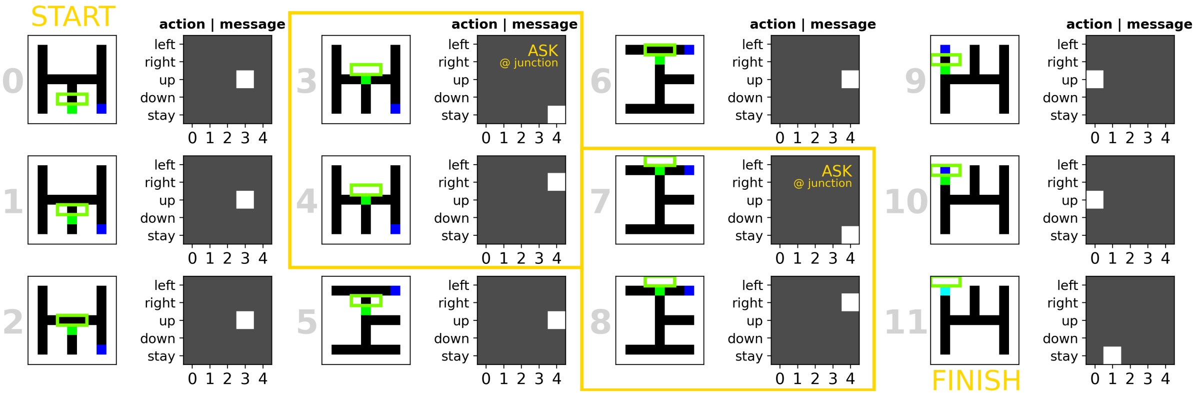

Figure 4.2. **Step-by-step walk-through of an example episode with *situated* communication.** The listener learns to solve the task optimally, deciding to stay and ask for information when at a junction (twice during the episode). At each of the 11 timesteps, we visualize (left) the speaker's view of the board with an overlaid green box indicating the listener's view, and (right) the speaker's message and listener's action at that step. The gridworld colors correspond to the color encoding included in the agents' observation vectors during the task.

vectors and represented with colors: walls are black ($[1,1,1]$), the maze is white ($[0,0,0]$), the agent is green ($[0,1,0]$), and the target is blue ($[0,0,1]$), as shown in Fig. 4.2 in the Appendix. The full environmental representation is defined as a $9 \times 9 \times 3$ matrix (height by width by RGB).

**The agents.** There are two independent RL agents, a speaker and a listener (i.e. acting agent). The speaker does not reside within the gridworld and cannot take environmental actions (i.e. navigate the maze) but instead can communicate information to the listener. The speaker's action (i.e. message) space spans 5 symbols $[0,1,...,4]$. We refer to the message $m_t = 0$ as a null message, and $m_t \in [1,2,3,4]$ as non-zero messages. At each timestep, the speaker can see the entire gridworld, including the location of the agent and the location of the goal. The speaker's view of the world map is rotated to align with the direction that the listener is facing. The listener is embedded in the gridworld and can take actions to move through the maze. The action space of the listener spans 5 actions [move up, move down, move right, move left, stay in place]. The listener's observation consists of the environmental view (if any) concatenated with the message from the speaker. We test the listener under two conditions: (1) with no visibility, where the listener's observation consists solely of the speaker's message, and (2) with partial visibility, where the listener can see the 3 pixels directly in front of them. The second variant gives the listener some environmental context to take actions without needing to rely solely on communication. We test agents with and without memory. Agents without memory have to rely only on their current observations to take an action. Agents with memory have an internal representation of the history of an episode—they can use accumulated knowledge from prior timesteps to make decisions in the current timestep.

**The task.** The goal of the agents is to cooperate so that the listener reaches the target. In each experimental episode, both agents receive a reward $R = 1$ if the listener reaches the target before the episode terminates. Episode timeout is set to 100 steps with a gamma discount factor of $\gamma = 0.99$. In all three environments, the goal locations are randomly assigned to one of the pixels indicated with a star in Fig. 4.1. There are 4, 5, and 32 possible goal locations in the T-maze, dead ends, and four rooms environments, respectively. In each episode, the listener agent starts from the 'START' cell indicated in Fig. 4.1. In the T-maze, the agent always starts from the bottom middle cell and the goal locations are randomly assigned to one of the 4 corners of the maze.

## 4.4. Situated vs. Unsituated Communication

Wagner et al. (2003) classify simulated environments for communication emergence based on the action space of the simulated agents. They define two types of actions: (1) communicative (i.e. sending or receiving signals)—actions that do not affect the state of the world or other agents, and (2) non-communicative (i.e. environmental)—actions that affect the environment and/or modify the agent's own internal state. Depending on the actions available to the agents, they define *situated* and *unsituated* simulations of communication emergence:

(1) *unsituated*: an agent's actions are only communicative (or only non-communicative)

(2) *situated*: an agent's actions include *both* communicative and non-communicative actions.

Given these definitions, we define two modes of communication based on the actions available to the listener. During *unsituated* communication in our task, the speaker generates a 1-token message at every timestep and the message is broadcasted to the listener before they choose an action. We refer to the listener using this mode of communication as a passive listener, because it passively receives the speaker's message at every timestep. When messages are free, this communication mode is equivalent to cheap talk. In *situated* communication, the listener can actively choose between (i) taking an environmental action and (ii) soliciting a message. The message is only broadcasted to the listener after they ask for information—we refer to this listener as active. The active listener can solicit to receive information in a following timestep by choosing a *stay in place* action at the current timestep. The active listener experiences an opportunity cost to communication—they have to forego an environmental move (that could bring them closer to the target) to obtain information and make an informed decision. As a result, they have to learn *whether* to communicate at all.

Similar to much of the existing work in the field [**113**], the communication in our experiments is asymmetrical (i.e. the action space of the speaker and listener are different). During *unsituated* communication, we simulate a passive listener that can take environmental actions and receive signals from a speaker but cannot send signals itself. If the agents' action spaces were symmetrical (the scenario considered by Wagner et al., 2003), this would be a somewhat degenerate case—communication would never emerge between two agents who can only receive signals and take environmental actions. As a result, our definition of *unsituated* communication deviates slightly from the definition introduced by Wagner et al. (2003).

## 4.5. Experimental Setup

**Agent architectures.** The speaker and the listener are designed as two independent RL agents. Both agents have the same architecture without sharing weights or gradient values. They both have a 2-layer Convolutional Neural Network (CNN) that generates an $8-32$ bit representation $s$ of the environment. In the case of the listener, this representation of the environment gets concatenated with the message received from the speaker. In both cases, the vector gets passed into a fully connected layer that generates the agent's action. Agents with memory, have an additional single-layer LSTM [**85**] after their fully connected layer.

We train the agents using neural fitted Q learning [**152**], with an Adam optimizer [**102**] and $Q_t(\lambda)$ where $\lambda = 0.9$. The Q values are updated using temporal difference (TD) error where the bootstrapped $Q_t(\lambda)$ is defined as follows:

$$Q_t(\lambda) = (1-\lambda)\sum_{n=1}^{\infty} \lambda^{n-1} Q_t^{(n)}$$

During training, agents use an $\varepsilon$-greedy policy with the exploration rate set as $\varepsilon = 0.01$.

**Hyperparameters.** For each experiment, we run a hyperparameter sweep over learning rates of the speaker and listener $\alpha \in \{10^{-5}, 10^{-6}\}$ and over the size of the environmental representation $s \in \{8, 16\}$, which denotes the size of the output vector of the convolutional layers of the agents' networks. We run the simulation with each hyperparameter setting 10 times with different random seeds. For each experiment, we present the best mean over the 10 replicas. The best mean is selected based on the metric of solution optimality (the normalized reward per step). When we plot metric means, we include the standard error of the mean.

**Message penalties.** In *unsituated* communication, we test the consequences of introducing a cost of articulation for the speaker. In the baseline scenario (i.e. cheap talk), all messages are free to the agent. With a cost of articulation, each non-zero message incurs a penalty, while a null message (the symbol 0) remains free to the agent. We test two penalty mechanisms: a fixed per-message penalty and a progressive per-message penalty. When fixed, the per-message penalty remains constant throughout the entire duration of training. We test fixed penalties with values $m_p \in \{0.01, 0.05, 0.1\}$.

As the second mechanism, we introduce a penalty curriculum with a scheduler that progressively increases the penalty between stages. The penalty scheduler is a mapping from the curriculum stage to the penalty in that stage. We test two curriculum implementations with the following mappings $m_{p1} = \{0:0, 1:0.01, 2:0.02, 3:0.03, 4:0.04, 5:0.05, ...\}$ and $m_{p1} = \{0:0, 1:0.01, 2:0.05, 3:0.1, 4:0.2, 5:0.3\}$. Both curricula start with no per-message penalty in stage 0, allowing cost-less exploration of a successful communication protocol. In the next stage, both curricula impose a small penalty that progressively increases over the time of training. Intuitively, the first curriculum is more gradual, enabling a more stable solution; the second curriculum is more rapid, possibly speeding up the learning process. The agents can progress to the next stage of the curriculum after $2M$ or $5M$ training steps if they achieve a performance threshold or if they spend $15M$ steps in their current stage. We test 3 progression thresholds based on the agents' success rate in reaching the target: 92%, 95%, and 97%.

**Evaluation metrics.** We evaluate agent performance according to three criteria: (i) task success (via the mean return per episode), (ii) solution optimality (via the normalized reward per step), and (iii) communication efficiency (via communication sparsity). The metric of task success is calculated after each episode as

$$M_t(n) = (\sum_{i=1}^{n} R_i)/n, \tag{4.1}$$

where $n$ is the number of episodes thus far and $R_i$ is the reward for the $i$th episode. The metric represents the likelihood of the agents succeeding at reaching the target before episode timeout. When it converges to 1, agents are reliably reaching the target in each episode. The normalized reward per step quantifies the optimality of the path taken to solve the task. If the task is solved in the optimal number of steps ($s_{opt} = 9$), agents obtain a per-step reward of 1. Formally, it is calculated as

$$M_o(n) = s_{opt} \sum_{i=1}^{n} R_i/s_i, \tag{4.2}$$

where $s_i$ is the number of steps the agent took in episode $i$. Finally, the metric of communication sparsity quantifies the efficiency of information exchanged between the collaborating agents. We define it as the average negative logarithm of the number of non-zero messages generated by the speaker per episode, such that

$$M_s(n) = (\sum_{i=1}^{n} -\log m_i)/n, \tag{4.3}$$

where $m_i$ is the number of non-zero messages in episode $i$. Agent pairs that converge to a higher value form a more efficient communication protocol—they exchange fewer messages per episode. If agents were able to solve the task using a single message, their sparsity metric would equal 0. As an example, in the T-maze, depending on the listener's characteristics: partial or no visibility, the optimal number of messages is equal to two ($M_s = -0.7$) or nine ($M_s = -2.2$) messages per episode for agents without memory, and one ($M_s = 0$) message per episode for agents with memory.

## 4.6. Results

We experimentally compare two pressures that can incentivise a concise exchange of information during communication emergence. Specifically, we consider two types of pressure:

- A per-message penalty: similar to a cost of articulation. During *unsituated* communication, we impose a cost on effort through a penalty on non-zero messages. We test (a) a fixed penalty, and (b) a penalty curriculum with a progressively increasing per-message penalty.
- An opportunity cost on communication: mimicking the pressure of time in a real-world situation. We study agents using *situated* communication, where at each step the listener faces an action-communication trade-off.

In our experiments, we train agent pairs communicating (1) using cheap talk (baseline), (2) using *unsituated* communication with a fixed penalty or a penalty curriculum, and (3) using *situated* communication. In all experiments, we train agents under two visibility and two memory conditions (see Sections 4.3 & 4.5 for more details). Without visibility or memory, the acting agent relies solely on the speaker's most recent message to take an action. With visibility and memory, the acting agent has increasingly more contextual information, allowing for sparser and more efficient communication.

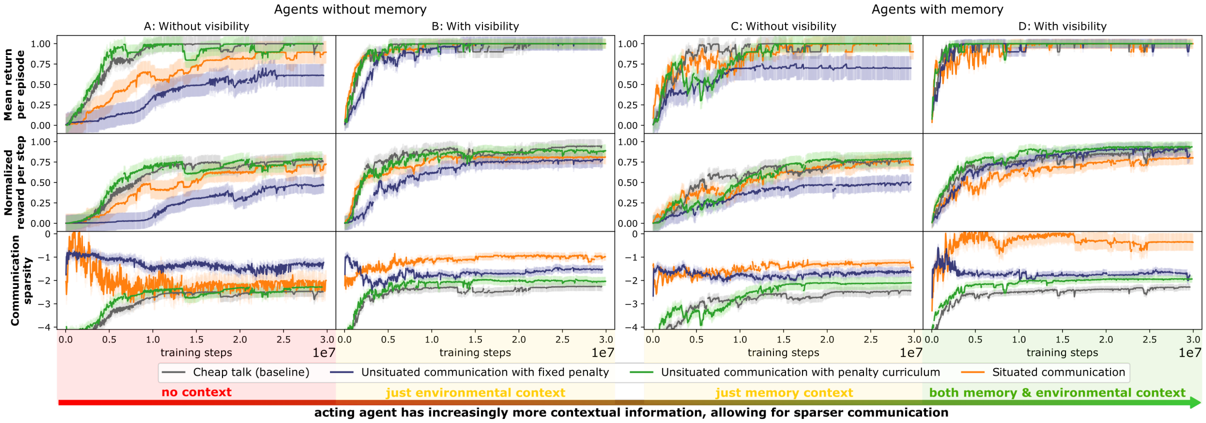

Figure 4.3. **Comparison of pressures for avoiding over-communication (mean over 10 seeds) in the T-maze.** Top row illustrates an average metric of success—under almost all conditions agents are able to find a solution to the task. Middle row illustrates an average metric of optimality—with context (i.e., visibility and/or memory) agents require fewer steps to reach the target. Bottom row illustrates communication conciseness—*situated* agents are able to communicate most sparingly.

### 4.6.1. Cheap talk (baseline)

**Without pressures to be concise, agents successfully learn to solve the task.** We start by analyzing the learning pattern and behavior of agents without communication constraints—we use their performance as a reference in our subsequent analysis. When allowed to communicate, all agents in the T-maze environment (grey lines in Fig. 4.3) learn to solve the task—their mean return per episode converges to 1. Best agent pairs find an optimal solution—their normalized reward per step converges to 1. With both memory and partial visibility for context, even the average performance is close to optimal (refer to Fig 4.3 column D, row 2). Moreover, we analyze the sparsity of the established communication protocol. With *unsituated* communication and no pressures (cheap talk), agents have no incentive to be sparse. As a result, for all tested conditions, their sparsity metric is equal to the number of steps needed to solve the task, and lowest compared to agents with added pressures (refer to the grey lines in Fig. 4.3 row 3).

For the best-performing agent pairs, we analyze the communication protocol qualitatively as illustrated in Fig. 4.4. Best-performing agent pairs agree on unambiguous meanings of messages and, in some cases, learn synonyms to signal the same environmental action. Under partial visibility (refer to the bottom middle heatmap in Fig. 4.4), the meaning of messages depends on the environmental context (e.g. message 1 at the corridor might be consistently interpreted by the listener as ‘move up’ but at the junction as ‘move right’). All else equal, successful agents without memory converge to a just-in-time protocol, where at each time step the listener can unambiguously interpret the speaker’s message. Interestingly, memory influences timing in the established communication protocols. The best-performing agents with memory learn a look-ahead communication protocol. As an example, a speaker with memory might broadcast the same message for the first 4 steps of a T-maze episode, alerting the listener to make a left or right turn at the junction (see the bottom right heatmap in Fig. 4.4).

Lastly, we analyze agent performance in two other environments. The learning curves are illustrated in Fig. 4.6. In four rooms, baseline agents achieve similar performance to the T-maze environment. The dead ends maze is more difficult—not all agents are able to solve the task. Importantly, compared to the T-maze, both dead ends and four rooms are significantly different from the listener perspective. In dead ends, when the listener sees a corridor (i.e. wall-path-wall alternating pixels), it can no longer learn to always go straight—4 out of 5 times it has to turn when seeing a corridor at the first junction. As a result, it has to rely more on the messages from the speaker. Similarly, in four rooms, there are many possible goal locations (32 compared to 4 in the T-maze) and the environmental context is ambiguous—visibility does not help the listener with decision-making as much as it does in the T-maze. Again, the listener has to rely more on the messages. Because the dead ends and four rooms environments are more ambiguous than

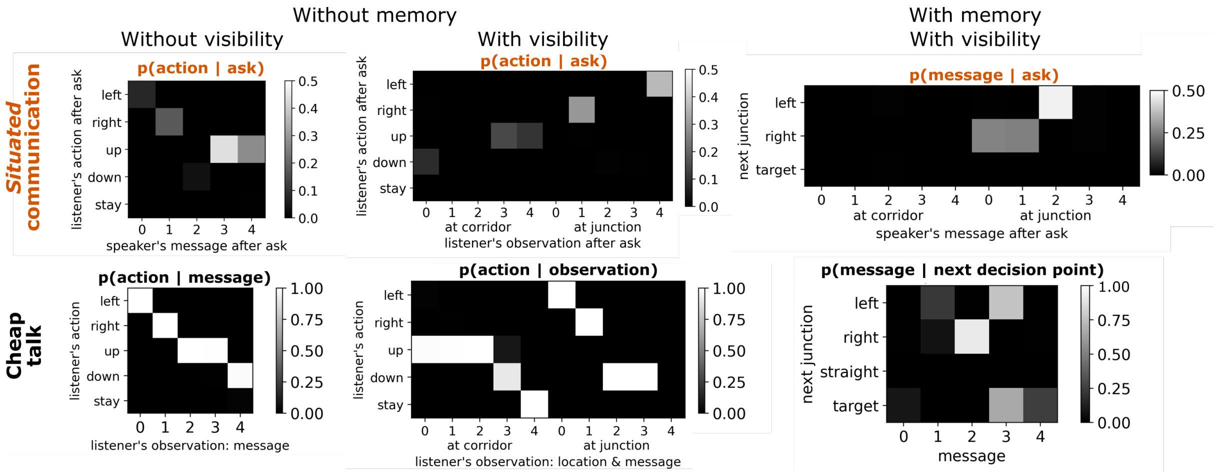

Figure 4.4. **Communication protocols for successful agent pairs in a T-maze environment.** Communication protocols are analogous when the listener has no context (left-most column). With context, agents with *situated* communication learn to exchange information sparsely, communicating mostly at the junctions.

the T-maze environment from the listener perspective, we only consider listeners with context (i.e. memory and partial visibility). With added pressures, this allows the agent pair to learn to communicate more sparingly and still succeed at the task.

### 4.6.2. *Unsituated* communication with a penalty (fixed and progressive)

**A fixed penalty improves communication sparsity but makes communication emergence more difficult.** We test four fixed penalty values to evaluate the impact of a fixed penalty on communication emergence. In Fig. 4.5, we illustrate the learning curves for agents training with a fixed penalty. Note that very few agent pairs learn to reliably solve the task. Only a few agent pairs find a quasi-optimal solution. Secondly, note in the bottom plots in Fig. 4.5 that when a cost of articulation is introduced, speakers send few non-zero messages at the beginning of training. The decrease in early exploration of a common language likely makes it harder for agents to establish a successful communication protocol and consequently causes difficulty in collaborating to reach the target.

In summary, when agents converge to a successful communication protocol, the per-message penalty incentivises sparse communication. However, overall convergence rates are low. Out of the three tested penalty values, a fixed penalty $= 0.05$ seems to perform best. In Fig. 4.3 and Fig. 4.6 for comparison with other pressures, we include agents trained with a fixed penalty $= 0.05$.

**A penalty curriculum does not negatively impact performance but it does not as strongly incentivise conciseness.** To mitigate the issue of stifled early exploration, we implement penalty curricula, in which agents start training with no per-message penalty. Only after they reach a performance threshold (as described in Section 4.5), the speaker begins to experience a penalty for every non-zero message. As visible in Fig. 4.3, this strategy significantly improves convergence rates—agents reliably learn to communicate *on par* with baseline agents. However, communication is not sparse. The agents exchange only slightly fewer messages than baseline agents (refer to the green vs. grey lines in Fig. 4.3 row 3). Once they find a communication protocol that works, they seem unlikely to change it even with an increasing penalty—the sparsity metric remains relatively constant throughout training. The trend is very similar in the other two mazes (see Fig. 4.6).

All in all, introducing a penalty (fixed or progressive) makes the communication sparser compared to baseline. However, the sparsity of communication is still suboptimal.

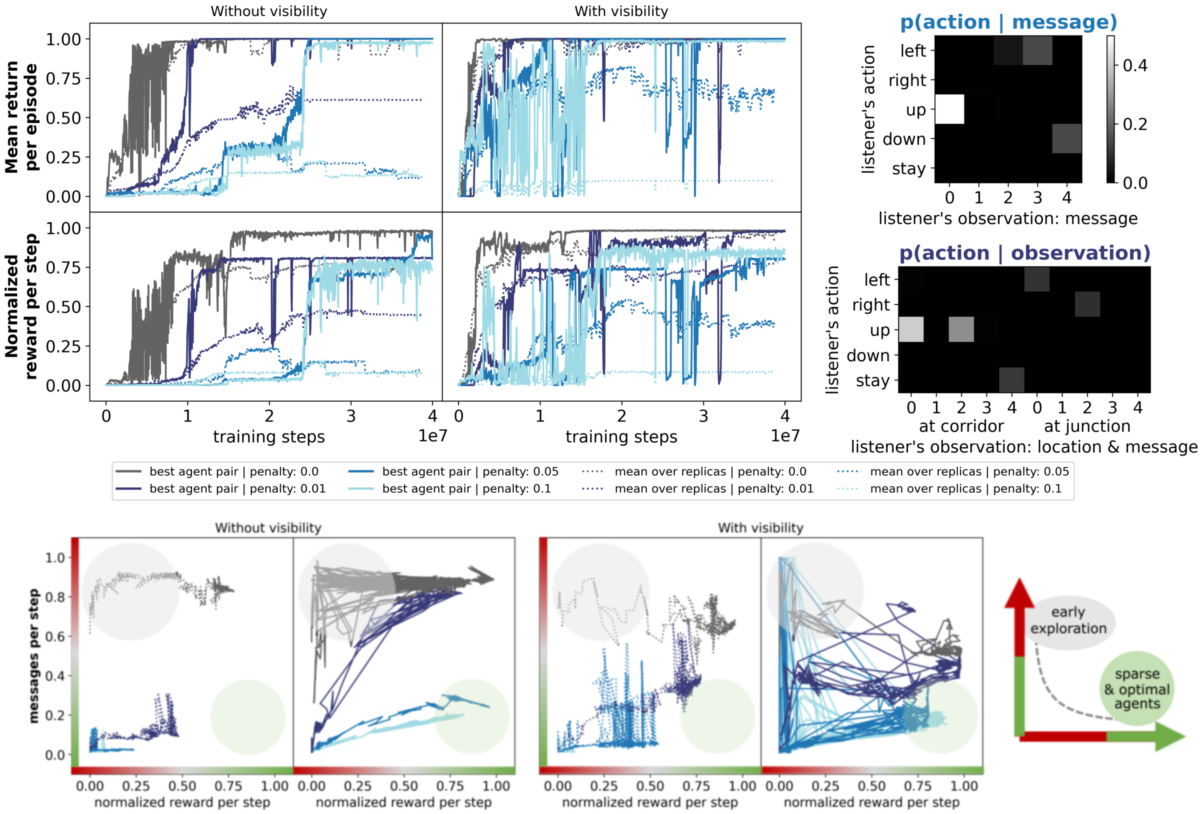

Figure 4.5. **Communication emergence with a cost on communication effort in the T-maze.** Agents without memory. (top) The additional pressure makes learning more difficult—task performance for blue lines is lower than for the grey lines (agents using cheap talk). (bottom) The immediate cost of non-zero messages stifles early exploration.

### 4.6.3. *Situated* communication

**The pressure of time in a multi-step interaction can incentivise sparse communication without compromising communication emergence.** In the last set of experiments, we evaluate the impact of *situated* communication on language emergence. We find that *situated* communication incentivises conciseness without stifling early exploration. Active listeners learn over time that communication comes at a cost, adjusting *when* and *whether* they solicit information, rather than avoiding communication early on (refer to the orange lines in Fig. 4.3 row

3). After training, agents using *situated* communication achieve highest communication sparsity. Importantly, the sparsity does not result in a loss in task performance—agents with an active listener follow learning patterns similar to baseline agents (refer to Fig. 4.3 row 1 & 2).

In a T-maze environment, the active listener can learn to near optimally solicit information, asking ~2.06 and ~9.76 times per episode given partial and no visibility, respectively. In order to be theoretically optimal under the no visibility condition, an active listener without memory requires 9 messages per episode (1 message per step) and, under partial visibility, 2 messages per episode (1 at each junction). The heatmaps in Fig. 4.4 illustrate example communication protocols of successful agent pairs. Under the no memory and no visibility condition, the listener queries the speaker at each timestep, in line with the optimal strategy. Under the partial visibility condition, information solicitation takes place mostly at the junctions, where the acting agent has a choice between two viable environmental actions.

Interestingly, when we test *situated* communication between agents with memory, agents continue to ask for information at the junctions when it is immediately actionable, as visible in the top right heatmap in Fig. 4.4. This is non-obvious—given memory, the active listener could ask for information at any point in the maze. This result suggests that it is easier for agents to succeed at the task when they exchange information when it is immediately actionable. There might be benefit to allowing agents to reason about the timing of communication in multi-step interactions. We show more results that support this thesis in Appendix C.2.

Lastly, we test the robustness of *situated* communication across two other environments: dead ends and four rooms. As visible in Fig. 4.6, across all three environments, active listeners learn to reliably solve the task—they achieve a mean return per episode equal to 1—while requiring the least amount of communication—their sparsity metric is closest to 0. In some

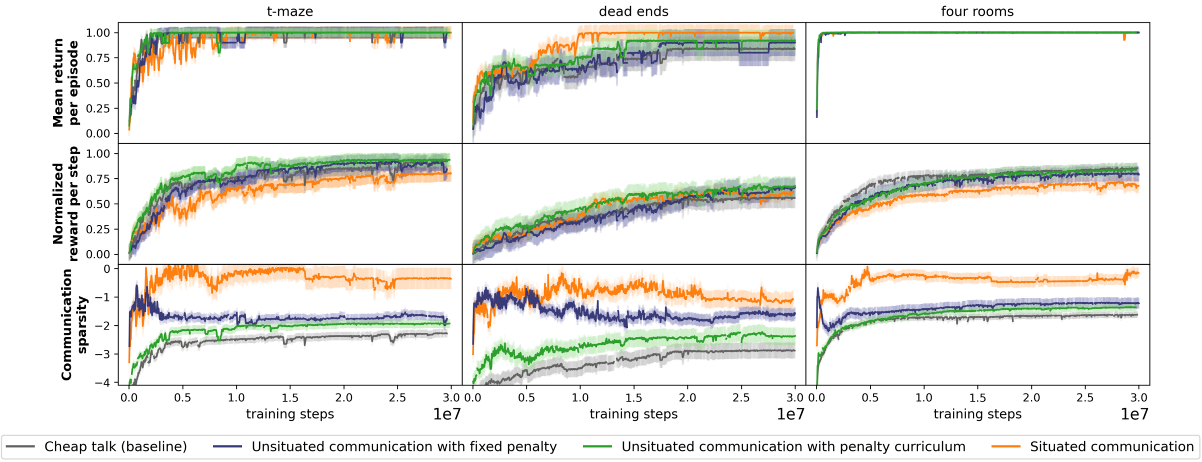

Figure 4.6. **Comparison of pressures for avoiding over-communication (mean over 10 seeds) in the 3 environments.** All agents have memory; listener has partial visibility. Across all 3 environments, situated agents agree on the most concise communication protocols without sacrificing their ability to solve the task.

mazes, active listeners converge to solutions that are slightly suboptimal compared to agents using other modes of communication, meaning that on average they take more steps to reach the target. Future work will investigate further how agents using *situated* communication can learn to solve the task optimally. A symmetrical implementation of *situated* communication, where both agents can act in the environment, might offer a robust solution.

## 4.7. Conclusions & Discussion

We find that giving the listener agency to choose *whether* to communicate enables agents to learn how to concisely exchange task-relevant information. By *situating* the communication in the task, we allow the functional pressure of time to shape the emergent communication protocol to be sparse, in line with the Gricean maxim of quantity. A cost of articulation, implemented in the form of a per-message penalty, is not as effective.

As common in related work and described in Section 4.4, the current implementation of communication is asymmetrical—only the speaker has privileged information and only the listener can take non-communicative actions in the environment. Ongoing work will expand this implementation and situate both the speaker and listener in the environment, enabling both agents to communicate and take non-communicative actions in the maze-bazed world. Moreover, we would like to explore more complex cooperative tasks, such as those in the Overcooked game environment [**37**, **104**]. One of the objectives of achieving concise communication protocols is to design RL agents that can effectively communicate with people.

## 4.8. Future Work: Communication Emergence during pHRI

In this work, we designed and tested situated RL agents that are capable of establishing communicative conventions through task-oriented interaction. Moreover, we showed that they can use these communication protocols to cooperatively complete a task while remaining concise in how they exchange information. The goal of this work is to deploy these agents with people in the loop to enable robot users to co-create symbolic languages with their collaborative robot companions. In the current implementation, agents learn to communicate using a pre-defined set of discrete tokens, and the learning cycle is relatively long—it takes thousands of interactions for agents to converge on a communication protocol. These two limitations are a potential challenge for human-in-the-loop deployment. As such, we run a follow-up, proof-of-concept experiment to test whether agents with a similar architecture can learn non-discrete symbols from smaller amounts of data. For these experiments, we simplify the learning setup—we replace learning using reinforcement from free-form interaction with supervised learning during a structured curriculum.

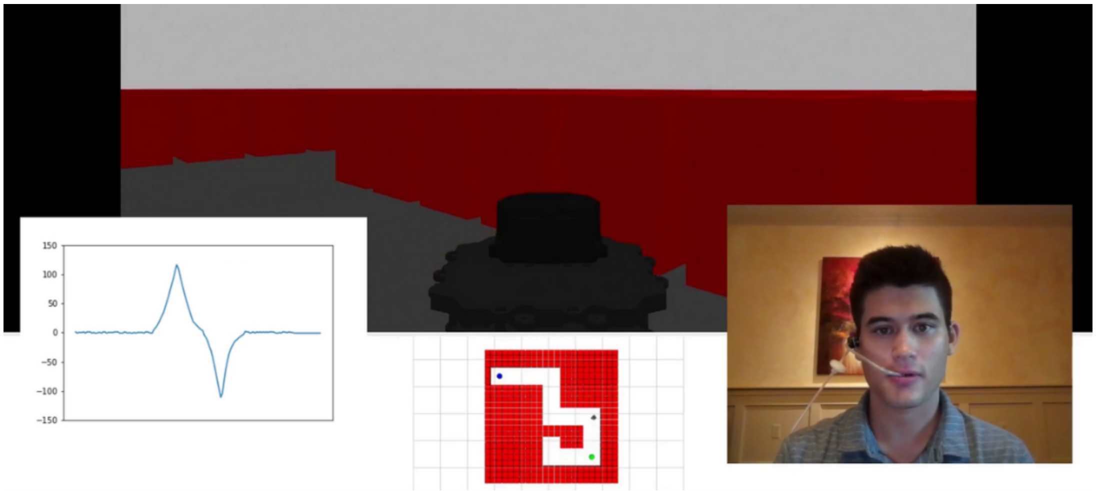

Figure 4.7. **Experimental setup.** The person sees a simulated mobile robot inside a maze, including a first-person view (top) and a top-down view (bottom middle). The person is using a SNP device to generate communicative symbols by controlling the air pressure in the SNP straw. An example symbol is plotted on the bottom left.

### 4.8.1. Experimental Setup

We create a simulated environment, where a person can use a sip-n-puff (SNP) device to communicate with a simulated mobile robot to navigate a virtual environment (see Fig. 4.7. The SNP device is a sensorized straw that a user can sip and puff into—it then outputs a 1D continuous signal, based on the recorded pressure inside the straw. For the simulated environment, we design a set of mazes that can be viewed through RViz and/or Gazebo, as visible in Fig. 4.8. The mazes are categorized by levels of difficulty, depending on the anticipated complexity of maze navigation. For example, easy mazes only include right-angle turns and straight movements, whereas mazes with a medium level of difficulty introduce diagonal movements. The starting point is indicated by a green circle, while the end is marked by a blue circle.

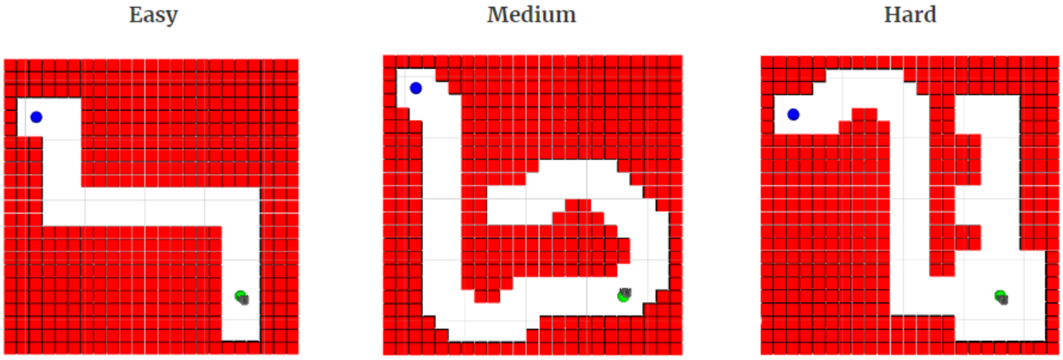

Figure 4.8. **Example mazes at three levels of difficulty.** Mazes are randomly generated and categorized by difficulty level. For example, easy mazes only include right-angle turns and straight movements, whereas mazes with a medium level of difficulty introduce diagonal movements. The starting point is indicated by a green circle, while the end is marked by a blue circle.

### 4.8.2. Learning Symbolic Vocabulary

We collect data using a simulated maze curriculum. The mobile robot is programmed with a pre-defined control sequence to navigate the maze at a constant velocity. During training, the user is asked to generate signals through the SNP interface to label the movement of the robot. Though the user is providing control inputs, the robot action sequence is pre-defined based on the generated maze. The SNP data is auto-labelled based on tags applied to the maze during the maze-generation process.

We augment the collected data to enable learning from fewer interactions. For each SNP symbol created by the user, we first smooth the signal to reduce noise. We also center the signal in time to normalize variability from the timing of the symbol relative to the movement of the robot. We then apply one of three distortions: (1) we compress/expand the signal in time and add noise, (2) we increase/decrease the signal's amplitude and add noise, or (3) we shift the signal left/right in time and add noise. We start from 20 samples per user (5 per symbol) and we apply 1000 transformations to obtain 20000 samples per user.

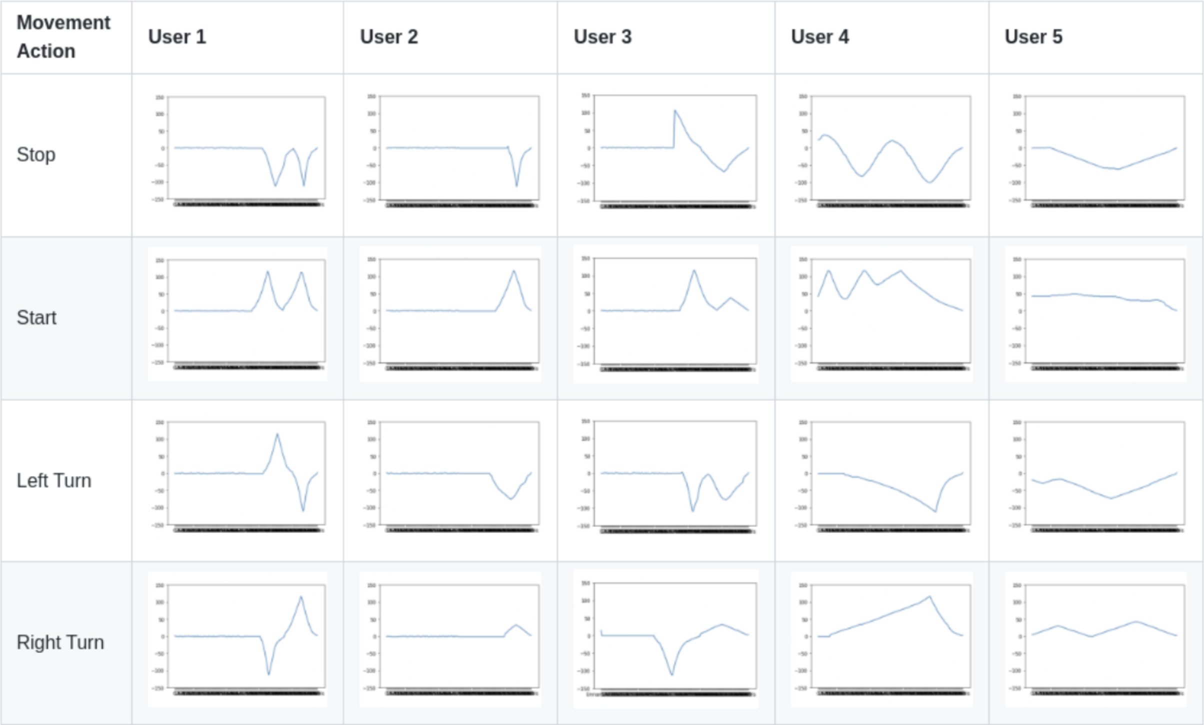

Figure 4.9. **Example sets of symbolic vocabulary.** Sets of symbolic vocabulary were generated by two people: users 1-4 were generated by a person very familiar with the setup; user 5 was generated by a person unfamiliar with the setup or using a SNP device. All user profiles led to successful robot control at test time, with a symbol classification accuracy between 83% and 95%.

The augmented dataset is used to train a 1-dimensional Convolutional Neural Network (CNN). The CNN is trained to classify SNP inputs into four discrete actions: forward, left, right, stop. We train a seperate model for each user. Using the trained network, the user can communicate with the simulated mobile robot and use the SNP device to navigate mazes in the virtual environment.

### 4.8.3. Experimental Results

A graphical representation of five sets of symbols is shown in Fig. 4.9. Users 1 through 4 were profiles created by a collaborator; user 5 was not familiar with the setup prior to use and was given no instructions other than to use the SNP device to guide the robot through the simulated maze.

During testing, the user was asked to navigate a number of randomly generated mazes, similar to the mazes used for data collection. This time, however, their input was continuously processed by the trained CNN and classified as one of the four movement actions (stop, start, left turn, right turn). In every trial, the trained model was able to successfully predict movement actions from SNP signal; the user was able to navigate a maze using their custom set of symbols.

All 5 profiles, visualized in Fig. 4.9, enabled the user to successfully navigate a test maze with minimal error. During testing, the model predicted correct movement actions with 91% accuracy for user 1, 83% accuracy for user 2, 87% accuracy for user 3, 95% accuracy for user 4, and 87% accuracy for user 5. Profile 4 led to to best performance. The most common error was caused by the symbol for stop being misclassified as a symbol for turning left. Interestingly, the system performance was comparable for user 5 who was unfamiliar with the setup to the average performance of the other four profiles.

### 4.8.4. Conclusions and Next Steps

In this work, we showed that it is possible to use deep learning to learn symbols from a human-operated physical interface, such as an SNP device, with just five demonstrations per symbol. Future work will aim to extend this approach to incorporate reinforcement learning and enable learning from task-oriented interaction.

One possible implementation includes directly combining the RL approach presented in this chapter with the data augmentation technique used in our SNP setup. While the RL approach was slow to converge when agents were initialized with random weights, we noticed that if the ‘oracle’ agent is pre-trained, two simulated agents can converge to a communication protocol in tens of interactions (orders of magnitude less than when the agents are randomly initialized). To explain, currently the ‘oracle’ agent observes a map of the maze along with a target target location color-coded in RGB pixels. During pre-training, the agent can learn to interpret an RGB map as well as identify a policy for navigating the maze. A pre-trained agent can then provide more informative communication tokens during a cooperative task attempt with another agent, speeding up the convergence process. During human-agent interaction, we can assume that the person would be more similar to a pre-trained agent rather than a randomly initialized one, and that they would provide at least somewhat informative commands from the beginning of the task-oriented interaction.

Secondly, we would need to transition from discrete tokens to continuous communication symbols. To do so, we could expand the agent architecture with an additional CNN that interprets continuous signals and outputs discrete tokens which are then considered as input to the action policy. We could take advantage of the data augmentation process introduced above. For one, we could pre-train the acting agent with a handful of commands for basic action primitives as we did in our current example. Moreover, we could learn from each interaction multiple times by distorting the human-generated signals during task-oriented interaction like we did during supervised training in our SNP setup.

Lastly, we can further adapt the human-in-loop training process of the RL agent to further optimize learning efficiency. As an example, we can create a dynamically adapting maze curriculum. The objective function for autonomously generating subsequent mazes could maximize information gathering around symbol boundaries. This would help users to refine their symbol definitions to make them increasingly unique and differentiated as well as to generate new symbols while avoiding excessive repetition of established ones. The ultimate goal is to enable the human-robot pair to continually expand their shared library of communicative symbols during task-oriented interaction.

CHAPTER 5

# Defining task representations using contrastive learning

With growing access to versatile robotics, it is beneficial for end users to be able to teach robots tasks without needing to code a control policy. One possibility is to teach the robot through successful task executions. However, near-optimal demonstrations of a task can be difficult to provide and even successful demonstrations can fail to capture task aspects key to robust skill replication. Here, we propose a learning from demonstration (LfD) approach that enables learning of robust task definitions without the need for near-optimal demonstrations. We present a novel algorithmic framework for learning tasks based on the ergodic metric—a measure of information content in motion. Moreover, we make use of negative demonstrations—demonstrations of what *not* to do—and show that they can help compensate for imperfect demonstrations, reduce the number of demonstrations needed, and highlight crucial task elements improving robot performance. In a proof-of-concept example of cart-pole inversion, we show that negative demonstrations alone can be sufficient to successfully learn and recreate a skill. Through a human subject study with 24 participants, we show that consistently more information about a task can be captured from combined positive and negative (*posneg*) demonstrations than from the same amount of just positive demonstrations. Finally, we demonstrate our learning approach on simulated tasks of target reaching and table cleaning with a 7-DoF Franka arm. Our results point towards a future with robust, data-efficient LfD for novice users.

## 5.1. Introduction

Many assistive robots being deployed in people's homes or on factory floors are capable of performing a variety of tasks. As such, it is beneficial for end users to be able to customize these robots by teaching them tasks specific to their needs. However, it is often not possible to provide high-quality task demonstrations. This could be because the task is challenging for a person to perform, e.g., cart-pole inversion due to its unintuitive dynamics, or the person is limited by a low-dimensional control interface, such as a joystick, for providing demonstrations to a 7-DoF robotic arm. Although successful approaches exist for imitation learning, including Dynamic Motion Primitives (DMPs) [**87**], inverse reinforcement learning (IRL) [**1**], and others—as we describe in more detail in Section 5.2—few of the LfD frameworks allow for reliable learning from novice task demonstrations.

Our approach stems from the idea of evaluating how much information about a task is encoded in motion—quantifying this using a measure of ergodicity. We propose ergodic imitation for robust learning from imperfect demonstrations. We define tasks through spatial distributions in state-based feature space. Through successive demonstrations, we learn the underlying distribution corresponding to a task and generate robot behavior via ergodic control [**127**] with respect to the learned distributions. This learning framework allows us to combine multiple novice demonstrations into a successful objective and use model predictive control (MPC) to recreate trajectories for new, previously unencountered scenarios. It is worth noting that ergodic imitation does not focus on imitating trajectories directly—instead it emphasizes imitating trajectory *statistics*. As a result, the method learns well from imperfect demonstrations and is robust to noise in individual demonstrations (e.g., corrective motions or perturbations).

Moreover, we propose imitation learning using negative demonstrations. In some cases, it might be easier for a person to demonstrate what *not* to do rather than to provide an exemplary task execution. Demonstrating things to avoid is something that people already intuitively do when teaching new skills to others. Robotic LfD can also largely benefit from incorporating negative demonstrations into the learning process. Ergodic imitation is a particularly suitable algorithmic framework, because it enables combining positive and negative demonstrations into a well-posed task objective.

As part of this study, we validate our learning approach on two test beds: a virtual 2-D cart-pole system and a simulated 7-DoF robotic arm. We find that ergodic imitation (1) enables robust skill reconstruction that outperforms the provided demonstrations and (2) generalizes to different robot tasks. Additionally, we test the utility of negative demonstrations in an experiment with 24 participants. Our results show that there is consistent benefit to soliciting combined *posneg* demonstrations compared to only positive demonstrations.

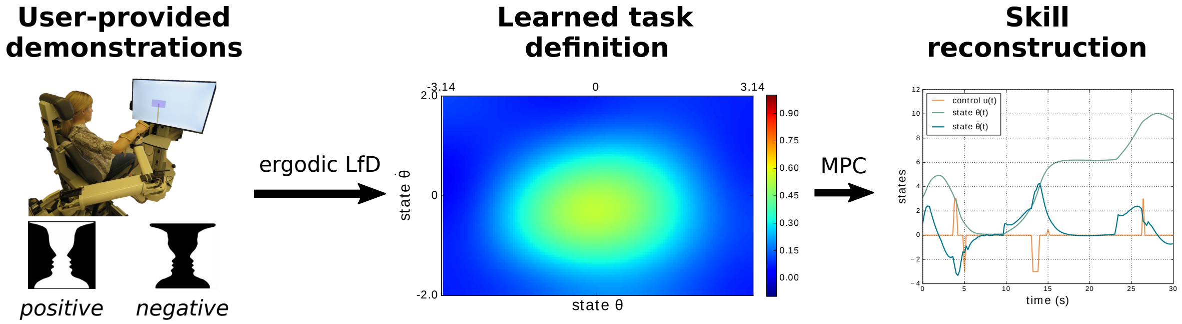

Figure 5.1. An overview of the learning process using ergodic imitation on the example of the cart-pole inversion task. Positive and negative demonstrations are combined to form *posonly*, *negonly*, or *posneg* task definitions.

## 5.2. Related Work

Most inverse reinforcement learning (IRL) and inverse optimal control (IOC) methods assume that the demonstrations representing the task are generated by an expert demonstrator providing optimal (or near-optimal) strategies for the task in order to generate feasible solutions [**1**, **84**, **117**]. Other approaches—such as those that use probabilistic methods to learn a task [**31**, **89**, **111**, **145**, **159**]—also rely on highly skilled demonstrators, accounting for imperfections with relatively small-scale noise in the probabilistic representation. To enable task learning that more closely captures human preferences and accounts for imperfect or incomplete demonstrations, active learning methods have been developed [**9**, **10**, **17**, **133**, **174**]. In these approaches, the human is treated as an oracle that the autonomy can query, improving learning quality. Alternatively in [**196**], the person provides ratings (or confidence scores) to a subset of demonstrations, quantifying the demonstrations' quality and increasing learning reliability. However, there is an inherent cost of time and effort to querying the user for input and so the user-in-the-loop learning process can be prohibitively frustrating [**5**, **29**]. The user tends to have a preference towards online learning approaches (e.g., [**174**]) that do not require *post-hoc* corrections to the learned robot policies. Therefore, in this work, we improve task learning during run-time by soliciting different types of demonstrations from the human teacher.

Suboptimal and failed demonstrations have been used in LfD before and have been shown to improve the learning process [**49**, **169**]. In [**169**], the proposed approach learns faster and generalizes better than the original IRL method, because it is able to learn from failure. In [**49**], authors demonstrate that accommodating failed demonstrations improves learning particularly for multimodal tasks (a result we also observe in our experiments). In [**79**, **80**], researchers propose a method that can learn solely from failed demonstrations, acknowledging that combining successful and failed demonstrations would likely deliver highest quality results. Building on this prior work, we demonstrate the value of soliciting negative demonstrations—explicit demonstrations of what *not* to do. We present an LfD algorithm that allows for combining positive (successful but possibly suboptimal) demonstrations to be combined with negative (unsuccessful and/or explicitly suboptimal) demonstrations together in the learning process.

While the above mentioned methods can successfully learn various skills, many cannot generate safety guarantees for the learned policy nor guarantee the dynamic feasibility of the generated trajectories [**49**, **80**, **169**]. In contrast, IOC methods, such as [**54**, **67**], which use optimal control to generate actions, can have provable guarantees on feasibility and performance. Similarily, the proposed algorithm inherits formal properties from ergodic control and standard MPC methods. (1) The ergodic cost is globally convex w.r.t. distributions so long as the metric used is on a Sobolev space. Here, we use the spectral approach as in [**126**]. (2) Ergodic imitation inherits asymptotic convergence from ergodic control [**127**]. In the cart-pole example, this implies that when the goal distribution is defined as a delta function at the unstable equilibrium, the statistics of the trajectory will asymptotically approach the delta function—the pole could occasionally fall, but the amount of time spent at the inverted equilibrium will approach 100% as $t \rightarrow \infty$. (3) Safety sets can be specified through the use of barrier functions.

In this chapter, we propose and validate ergodic imitation as one possible approach that allows for learning from both suboptimal and negative demonstrations by allowing demonstration-based distributions to be simply added together, and it maintains desired properties of existing

IOC methods. We obtain these properties by defining objectives as state distributions and employing ergodic control for closed-loop trajectory generation. While other methods, such as IRL, could be adapted to learn an objective function over distributions, the nominal complexity of IRL is already exponential ($O(n^2 log(nk))$) [**105**]. If one were to perform IRL over the set of distributions, the algorithm would further increase in computational complexity likely becoming intractable. In ergodic imitation, the learning step (in Eq. (5.3)) has linear time complexity, allowing us to avoid the computational complexity of IRL. As documented in prior work, ergodic control scales well to relatively high-dimensional spaces—it has been implemented on the half cheetah example with 26 observables and continuous actions [**3**].

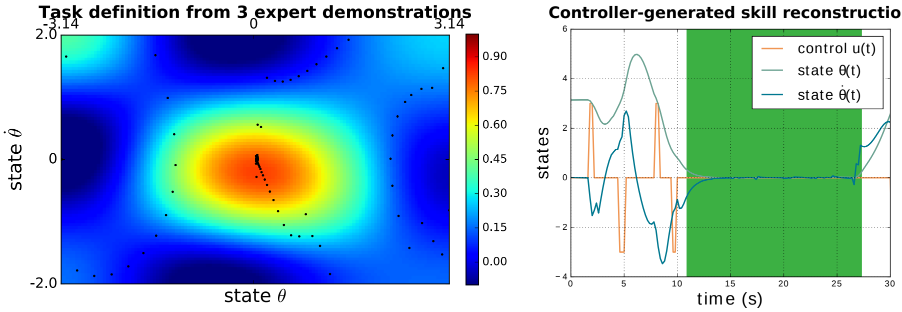

Figure 5.2. Example task definition (left) and skill reconstruction (right) learned from 3 expert trajectories. An optimal controller is used to recreate the task given the learned goal distribution. Green indicates success—time when the cart-pole is inverted. The controller-generated trajectory is plotted on the right and overlayed on the task distribution on the left with black dots—note that the trajectory closely represents the underlying distribution subject to constraints imposed by system dynamics.

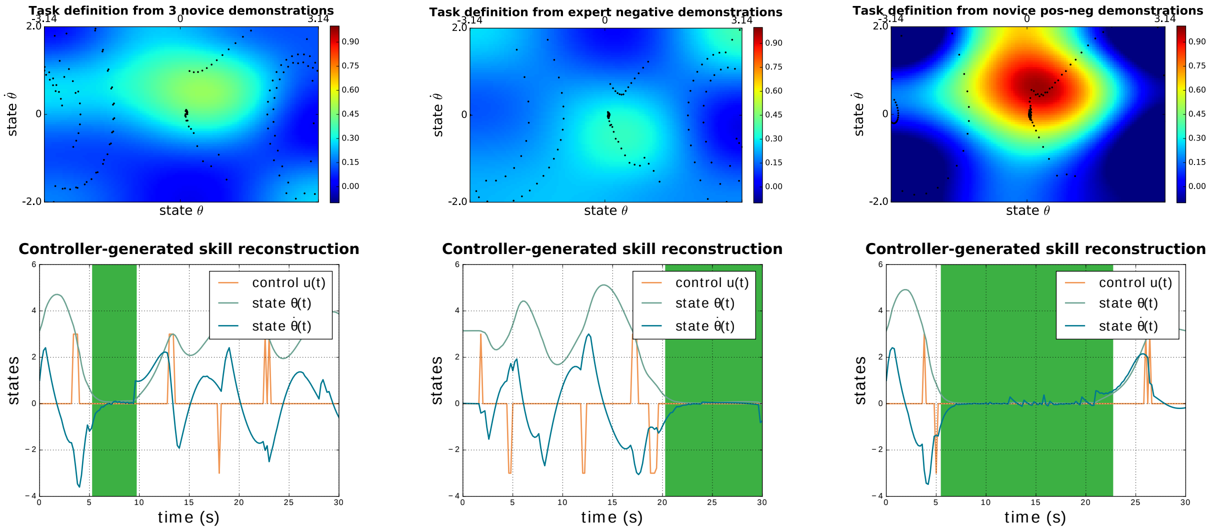

Figure 5.3. (left) Example task definition and skill reconstruction learned from 3 novice trajectories from Subject 6. Note that the controller just barely succeeds (for less than 5 seconds), exhibiting comparable performance to the original demonstrations, which had an average success time of 5.6 seconds. (center) Example task definition and skill reconstruction learned from 3 *negative* demonstrations. Note that a negative demonstration includes only failed task attempts and—with this low-dimensional task—suffices for learning a sub-optimal, yet successful task definition. (right) Example task definition and skill reconstruction learned from positive and negative demonstrations from Subject 6. Note that while the *posonly* controller exhibits performance similar to the original demonstrations, the *posneg* controller significantly outperforms them. Although Subject 6 is still a novice at this task, we can learn a task representation comparable to the one learned from an expert trajectory (see Fig. 5.2) by soliciting both positive and negative demonstrations.

## 5.3. Methods

### 5.3.1. Ergodic Task Definitions and Control

In ergodic imitation, we generate a representation of an unknown task using spatial statistics. Since we avoid specifying temporal dependencies, we synthesize robotic controls that successfully achieve a task without necessarily replicating the demonstration's time-evolving trajectory. We define a single demonstration, represented as $d_i$, as the distribution of points in the state space making up the the state trajectory $x(t)$ for a given set of time $t \in [t_0, t_f]$, and the set of demonstrations as $D = d_1, ..., d_m$. This set may contain both positive and negative demonstrations, so we also store a label array $E = e_1, ..., e_m$ corresponding in length to $D$.

**Learning from positive demonstrations.** A positive demonstration is defined to be a person's attempt at a task that is at least somewhat successful. Oftentimes user-provided positive demonstrations are incomplete or highly sub-optimal with multiple attempts at the task and corrective actions within the demonstration. This was the case in the user studies used in this work.

During the task learning process, we use the demonstration trajectories to generate a task definition $\phi(x)$ by representing the spatial statistics of each demonstration trajectory $d$ with the Fourier decomposition $\phi_k$, as described in Eq. 5.2. We then average the $\phi_k$ values of the demonstrations to represent the collective spatial statistics of all the demonstrations. Regions of the state space where more time is spent in the trajectories have a higher density in the distribution than regions where less time is spent. Examples of the cart-pole inversion task learned from positive demonstrations—of an expert and a novice—are showed in Fig. 5.2 and Fig. 5.3, respectively.

To generate the distribution $\phi(x)$ from the demonstration trajectories $x(t)$, we calculate spatial Fourier coefficients of $x(t)$ using Fourier basis functions of the form

$$F_k(x) = \frac{1}{h_k} \prod_{i=1}^{n} cos\left(\frac{k_i \pi}{L_i} x_i\right), \tag{5.1}$$

where $k$ is a multi-index over $n$ dimensions, $h_k$ is a normalizing factor [**126**], and $L_i$ is a measure of the length of the dimension. We then compute the coefficients of a time-averaged trajectory using Eq. 5.2.

$$c_k = \frac{1}{T}\int_0^T F_k(x(t))dt \tag{5.2}$$

The coefficients of the demonstration trajectories are combined to form the coefficients that describe the task definition. We use a weighted average

$$\phi_k = \sum_{j=1}^{m} w_j c_{k,j}, \tag{5.3}$$

where the weighting factor $w_j$ normalizes each demonstrations based on either the length of the trajectory or the relative quality of the demonstration. In the examples in this chapter, we normalize by the length. Note that other representations of a distribution could be used instead of Fourier coefficients, including wavelets or Gaussian Mixtures, as long as a comparison metric of two distributions can be defined and meets the conditions for global convexity.

**Learning from negative demonstrations.** In this work, we employ negative demonstrations, defined as both unsuccessful task attempts and explicit demonstrations of what *not* to do. Negative demonstrations can include good-faith attempts at a task where the demonstrator performs poorly, or explicit examples of actions that are far from the desired behavior. In the case of reaching a target with object avoidance, a negative demonstration might repeatedly circle the area of the object without reaching the target. Negative demonstrations add most complementary information to the positive (successful) demonstrations.

As with positive demonstrations, the demonstrated trajectories are represented by the Fourier decomposition $c_k$ calculated using Eq. 5.2. However, they are combined through subtraction—$w_j < 0$ in Eq. 5.3—such that regions of the state space where more time is spent in the trajectories have a lower density in the distribution than regions where less time is spent. When only negative demonstrations (*negonly*) are used, a uniform distribution is introduced into the demonstration set D and given a positive weight, $w_j > 0$.

**Algorithm 1** Ergodic Control Algorithm for LfD

```
Input: initial time $t_0$, initial state $x_0$, set of demonstrations $D = \{d_1, ..., d_m\}$ with positive/negative labels $\{e_1, ..., e_m\}$, final time $t_f$
Output: ergodic trajectory $x(t) \to X$
  Define: ergodic cost weight $Q$, highest order of coefficients $K$, control weight $R$, search domain bounds $\{L_1, ...L_n\}$, sampling time $t_s$, time horizon $T$
  Initialize: nominal control $u_{nom}$, $i = 0$
  Generate distribution $D(s)$ from demonstration set $D$.
  Calculate $\phi_k$ from distribution $D(s)$
  while $t_i < t_f$ do
    Compute $u_i^*$ using MPC
    Apply $u_i^*$ for $t \in [t_i, t_i + t_s]$ to get $x \forall t \in [t_i, t_i + t_s]$.
    Define $t_{i+1} = t_i + t_s, x_{i+1} = x(t_{i+1})$
    $i \leftarrow i + 1$
  end while
```

**Ergodic Control.** Once the task is learned—by combining demonstrations using Eq. (5.3)—we use a model predictive controller (MPC) to synthesize controls that generate a trajectory to match the spatial statistics of the distribution representing the demonstration set. In defining the task objective, we use ergodicity, which relates the temporal behavior of a signal to a pre-defined distribution. Ergodicity can be measured by several metrics [**162**, **163**]; here we use the spectral approach [**126**], which characterizes ergodicity by comparing spatial Fourier coefficients of $x(t)$ to coefficients of $\phi(x)$. Assume we have an autonomous agent whose movements are governed by a dynamic model that is either known *a priori* or learned from data and is of the form

$$\dot{x} = f(x, u) = g(x) + h(x)u \tag{5.4}$$

where $x \in \mathbb{R}^n$ is the state of the agent and $u \in \mathbb{R}^m$ is the control input or "actions" the robot can take. A trajectory $x(t)$ is ergodic with respect to a distribution $\phi(x)$ if, for every neighborhood

$\mathscr{N} \subset \mathscr{X}$, the amount of time $x(t)$ spends in $\mathscr{N}$ is proportional to the measure of $\mathscr{N}$ provided by $\phi(x)$. On a long enough time horizon, measuring a perfectly ergodic $x(t)$ gives a complete description of $\phi(x)$. Here, we ask that $x(t)$ be maximally ergodic, by introducing a metric on the distance from ergodicity into the objective function, so that when $x(t)$ captures the statistics of $\phi(x)$ in a specified time horizon $T$, the metric is lower. Ergodicity can be quantified as the sum of the weighed square distance between Fourier coefficients of the distribution $\phi_k$ and the coefficients representing the trajectory $c_k$:

$$\varepsilon = \sum_{k_1=0}^{K} \dots \sum_{k_n=0}^{K} \Lambda_k |c_k - \phi_k|^2, \tag{5.5}$$

where there are $n$ dimensions and $K+1$ coefficients along each dimension, and the coefficients $c_k$ can be calculated using Eq. 5.2. The coefficient $\Lambda_k = \frac{1}{(1+||k||^2)^s}$, where $s = \frac{n+1}{2}$, places larger weights on lower frequency information.

We define the task objective as

$$J = q\varepsilon + \int_0^T \frac{1}{2} u(t) R u(t) dt \tag{5.6}$$

with a cost to minimize the ergodic metric and a cost on the control effort used over time.

Now, using the ergodic objective function defined above, we frame the control problem as an MPC problem, following work in [**127**]. The full algorithm is outlined in Algorithm 1.

### 5.3.2. Experimental platforms

We use two simulated experimental platforms and three benchmark tasks for algorithm validation. Similar to [**27, 182, 197**], we employ a cart-pole system for initial validation with end users. Inverting and balancing the cart-pole is a great example of a task that is reliably difficult

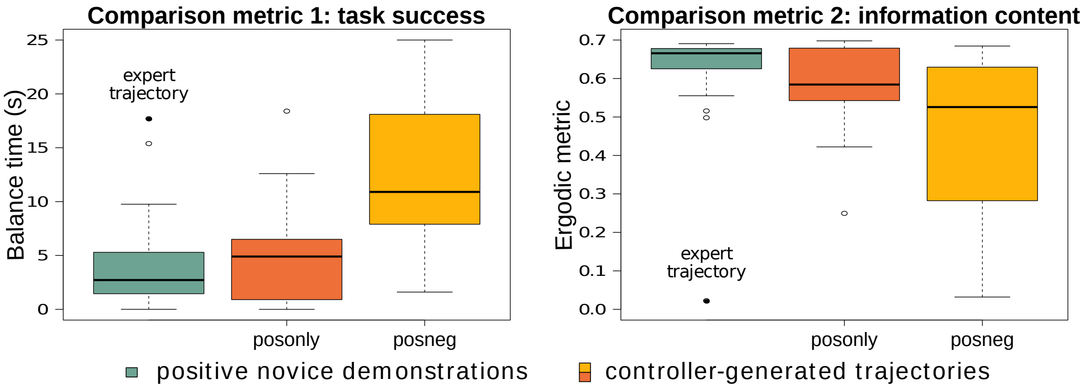

Figure 5.4. Comparison of best task executions from 24 novice participants and skill reconstructions based on learned objectives, using only positive demonstrations (*posonly*, orange) and using both positive and negative demonstrations (*posneg*, yellow). We employ two performance metrics for the comparison: task success time (left) and the ergodic metric [**71**] (right), which measures information captured about the task in the learned distributions by comparing it to the true task definition. For both metrics, *posonly* skill reconstructions achieve performance comparable to or better than the novice demonstrations. *Posneg* trajectories significantly outperform the provided novice demonstrations in both metrics: 1 (F=9.07, p=5.7e-10) and 2 (F=1.2, p=3.8e-5)—in fact they provide skill reconstructions comparable to expert task executions.

for people, particularly novices, to accomplish. We also test two household tasks on a robot arm. These include reaching with object avoidance (similar to [**16, 143**]) as well as cleaning or wiping a surface (similar to [**4, 63**]). These are good examples of real-world assistive tasks that encounter high variability in task execution during demonstrations.

**Cart-pole System.** A simulated cart-pole with state vector $x = [\theta, \dot{\theta}, x_c, \dot{x_c}]$ and input $\ddot{x_c}$ was used in a previous study of 24 participants. Participants were each given 3 sets of 30 30-second attempts to invert the pole from its resting state to the unstable equilibrium. The data from this experiment—details of which can be found in [**95**] and [**72**]—are used as the novice task demonstrations in this work.

For cart-pole inversion, a demonstration is defined as successful when during the 30-second attempt the participant reaches a state near the unstable equilibrium, specifically $|\theta| < 0.4$ rad and $|\dot{\theta}| < 0.75$ rad/s. We take the best demonstrations each user provided in set 3 (on average the best set) as positive demonstrations and take unsuccessful demonstrations from set 1 (on average the worst set) as negative demonstrations. Finally, we also test our approach on expert demonstrations—the positive expert demonstrations are generated using an optimal controller, whereas the negative expert demonstrations are generated by one of the authors. The true task definition for cart-pole inversion is defined as a Dirac delta function around $[\theta, \dot{\theta}] = [0, 0]$.

**Robot Arm Simulator.** We develop a pybullet simulation of the Franka Emika Panda Robot Arm to evaluate ergodic imitation on basic table-top tasks, specifically target reaching and table cleaning. In the simulation, we generate demonstrations for robot motion using a keyboard control interface. Keyboard keys control the desired end-effector position of the robot in the $[x, y]$ dimensions at a fixed end-effector height $z_d$. Demonstrations consist of the end-effector trajectories, from which we learn a task distribution. After that task definition is learned, we use ergodic MPC as a motion planner for the end-effector by generating desired end-effector trajectories $[x, y, z_d]$. For the ergodic controller, we model the system as a double-integrator with state $X = [x, y, \dot{x}, \dot{y}]$ and $U = [\ddot{x}, \ddot{y}]$. We use the an IK solver to generate joint states corresponding to the target trajectory and employ a low-level joint controller to execute the trajectory on the robot arm.

For the target-reaching task, we define success as reaching a target location without colliding with an obstacle. For the cleaning task, success $m$ is evaluated as a continuous variable based on both workspace coverage and object avoidance. If the controller-generated trajectory comes too near the object, the trial is considered a failure ($m = 0$). Otherwise, the cleaning is assessed by calculating the workspace visited by the end-effector, as approximated on a $5 \times 5$ grid.

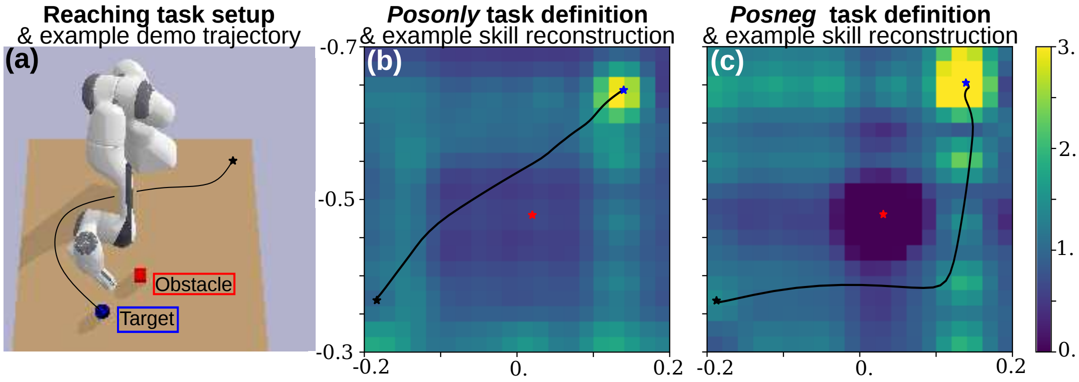

Figure 5.5. (a) Reaching towards a target (shown in blue) while avoiding an obstacle (shown in red) with a robot arm. Task definition and resulting robot end-effector trajectory generated with an ergodic controller using the positive-only demonstrations (b) and using combined positive and negative demonstrations (c). Negative demonstrations more effectively reflect the region of avoidance, representing what *not* to do.

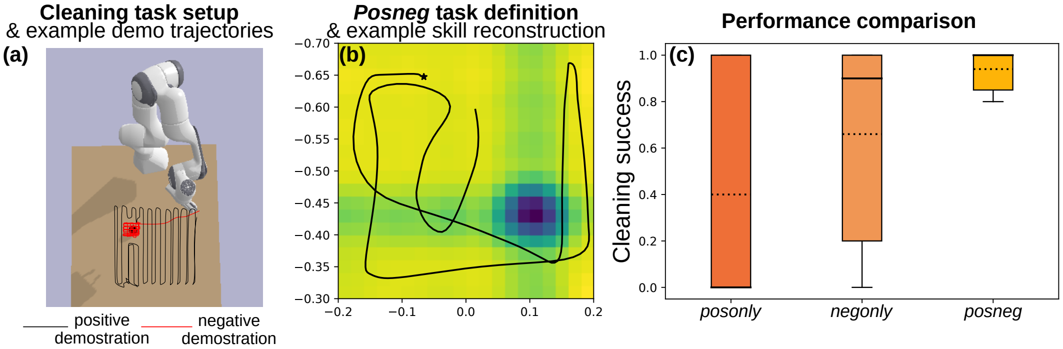

Figure 5.6. Cleaning around an object with a robot arm. (a) An example positive demonstration trajectory is shown in black and a negative demonstration is shown in red. (b) Task definition and resulting robot end-effector trajectory generated using the ergodic controller with positive and negative demonstrations. (c) Comparison of success between results from task definitions generated with positive, negative, and *posneg* demonstrations. The black line represents the median result and the dotted line represents the mean. The *posneg* definition results in significantly better performance than either *posonly* or *negonly* definitions, capturing both the desired cleaning and object avoidance goals.

## 5.4. Experimental Results

### 5.4.1. Ergodic imitation enables learning from imperfect demos

We show that ergodic imitation can be used to infer the cart-pole inversion task from imperfect novice demonstrations and that it can recreate the skill on average better than presented during demonstrations. More specifically, a t-test comparison shows that trajectories generated using ergodic imitation have higher success times (F=0.24, p=0.06) and are more ergodic w.r.t. the true task definition (F=0.68, p=0.002) than the provided demonstrations (see Fig. 5.4). This means that when learning from only positive demonstrations, our trained controller will on average perform comparably or outperform the provided task demonstrations.

### 5.4.2. Negative demos consistently improve learning

Furthermore, we show that negative demonstrations add more value than numerous positive demonstrations, allowing data-efficient learning. For each of the 24 participants, we learn a task definition from 3 positive and 3 negative demonstrations. Again, we compare the controller-generated trajectories with the provided trajectories, using success time and the ergodic metric. Results of a t-test show that trajectories generated using ergodic imitation have higher success times (F=9.07, p=5.7e-10) and are more ergodic with respect to the true task definition (F=1.2, p=3.8e-5) than the provided demonstrations. They also have higher success times (F=1.4, p=4.9e-7) and are more ergodic (F=0.79, p=0.003) than the trajectories generated using *posonly* demonstrations. Finally, note that the effect sizes are significantly larger than in the earlier comparison as visible in Fig. 5.4.

In the event that an end-user cannot generate any successful demonstrations, we also demonstrate the ability to define a successful task specification from just negative demonstrations. This is visible in Fig. 5.3, where we note that the skill reconstruction achieves inversion quite late around $t = 20s$. Although it is likely impractical for most tasks to learn from only negative demonstrations, this interesting result illustrates that valuable information about a task can be conveyed through negative demonstrations.

### 5.4.3. Ergodic imitation with negative demonstrations works for multimodal and multi-objective tasks

Ergodic imitation with *posneg* demonstrations extends to a variety of tasks. It is particularly useful for open-ended tasks, such as cleaning, where multiple successful task executions exist, or for multi-objective tasks, such as reaching a target while avoiding an object, where an added safety constraint is present. Negative demonstrations can be helpful, particularly when trying to represent constraints in the environment.

We use ergodic imitation to learn to reach a target while avoiding an object. In Fig. 5.5, we present an example trajectory generated from random initial conditions based on a task definition learned from 13 positive demonstrations and combined *posneg* demonstrations (13 positive + 3 negative), all between 10-15 seconds in length. Note that with positive demonstrations, the goal location is successfully reflected, but the region of obstacle avoidance is only starting to appear. When we add negative demonstrations, the region of avoidance is more clearly defined.

We use ergodic imitation to clean a surface around an object. We compare the results for 10 controller-generated trajectories from random initial states for each type of task definition (*posonly*, *negonly*, and *posneg*)—generated from 5 positive, 2 negative, and 3+2 combined *posneg* demonstrations, respectively. The positive demonstrations were 40-50 seconds in length whereas the negative ones—18-25 seconds. As visible in Fig. 5.6, the combination of positive and negative demonstrations offers best performance, highlighting the region to avoid while still representing the rest of the cleaning task. The *posonly* controller-generated trajectories sometimes result in collisions. The *negonly* skill reconstructions result in very few object collisions but the overall workspace coverage is low. The *posneg* task definition significantly outperforms both other task specifications, resulting in no failures and a median 100% success rate.

## 5.5. Conclusions & Discussion

This chapter introduces ergodic imitation for learning from novice robot users and illustrates the value of negative demonstrations—reflecting what *not* to do—in imitation learning. While positive-only demonstrations can result in successful skill reproduction, the combination of positive and negative demonstrations can help to efficiently generate task definitions for difficult tasks. Ergodic imitation is particularly well suited for multi-objective and open-ended tasks, where either multiple goals are equally important (e.g., moving a cup without spilling) or different motion trajectories can accomplish the same task.

There is potential to extend the approach to applications with a focus on safety constraints and user preferences, such as assisted driving—similarly to [**156**] but without the need for preference querying. Moreover, in future work, the ergodic learning framework could be further automated by combining it with feature selection algorithms, such as [**110**, **137**, **142**]. Overall, the presented results are promising—the proposed algorithmic framework and negative demonstrations have potential to enable demonstration-efficient LfD from imperfect demonstrations for a range of robotic applications.

CHAPTER 6

# Learning task representations of sequential tasks

Hybrid systems, such as bipedal walkers, are challenging to control because of discontinuities in their nonlinear dynamics. Little can be predicted about the systems' evolution without modeling the guard conditions that govern transitions between hybrid modes, so even systems with reliable state sensing can be difficult to control. We propose an algorithm that allows for determining the hybrid mode of a system in real-time using data-driven analysis. The algorithm is used in conjunction with data-driven dynamics identification to enable model predictive control based entirely on data. Two examples—a simulated hopper and experimental data from a bipedal walker—are used. In the context of the first example, we are able to closely approximate the dynamics of a hybrid SLIP model and then successfully use them for control in simulation. In the second example, we demonstrate gait partitioning of human walking data, accurately differentiating between stance and swing, as well as selected subphases of swing. We identify contact events, such as heel strike and toe-off, without a contact sensor using only kinematics data from the knee and hip joints, which could be particularly useful in providing online assistance during walking. Our algorithm does not assume a predefined gait structure or gait phase transitions, lending itself to segmentation of both healthy and pathological gaits. With this flexibility, impairment-specific rehabilitation strategies or assistance could be designed.

## 6.1. Introduction

In order to implement predictive control on an autonomous system or to provide robotic assistance to a human, we require the ability to identify system dynamics. Lots of work has been done in this area, allowing one to learn the dynamics of an autonomous robot [**2**, **58**, **136**, **144**, **192**] or a joint human-machine system [**25**] from collected state data.

Hybrid systems are more challenging. Even if we have reliable sensors to real-time detect the switching times of the system dynamics, we require a closed-form mapping from states to hybrid mode to implement any form of predictive control. And, oftentimes, one would prefer to avoid installing supplementary sensors altogether because of the cost, unreliability or inconvenience of additional hardware.

A lot of work in this domain has been done in the context of gait analysis and gait phase identification. Real-time, closed-form expressions for distinguishing between hybrid dynamic modes can be obtained using supervised machine learning techniques, such as neural networks [**42**, **188**], Hidden Markov Models [**179**], or Gaussian Mixture Models [**46**] from selected sensory information, such as ground contact forces, joint positions, inertial data, or muscle signals. Most of these studies rely heavily on ankle data for phase identification, which is often not present in lower-limb assistive devices, such as the one used in this study. Moreover, work in this area often requires assumptions to be made about the number of dynamic modes, phase transition times, or phase durations, limiting the algorithms' ability to adequately partition abnormal gait. Finally, to the authors' best knowledge, no gait partitioning strategies have been shown to have the potential to generate control for assistance in real-time.

Here, we propose a novel algorithm that synthesizes a closed-form expression for hybrid mode switching conditions, allowing for real-time switching time predictions. For a simple system—a simulated one-legged hopper—we show that the data-derived guard conditions can be employed for control. For a bipedal walker (a human in an Ekso Bionics® exoskeleton), we demonstrate accurate gait segmentation using data only from knee and hip joints. We validate our gait partitioning by comparing it to pressure data from heel and toe contact sensors and demonstrate the ability to reliably predict heel strike and toe-off events without use of impact sensors. We do not pre-label transitions in our training data or pre-define the expected number of phases, which allows us to identify a range of recurring movement patterns in the gait cycle and shows promise for meaningful partitioning of abnormal walking.

This chapter lays the groundwork for applications of real-time assistance. In Section 6.2, we give a detailed description of our algorithmic approach and methods used. In Sections 6.3 and 6.4, we validate our methodology through two examples:

- control of a simulated hopper without any *a priori* knowledge of its dynamics or guard conditions
- gait partitioning of experimental data from a bipedal walker.

Finally, we conclude with a discussion of the results and opportunities for future work.

## 6.2. Methods

Our procedure for approximating the dynamics of a hybrid system for control is split up into two parts: synthesizing switching conditions between dynamically distinct modes, and estimating the continuous dynamics of each mode from data. For the first part, we use Nonparametric Clustering of Dynamics (NCD), where we locally approximate system dynamics at each several points in time, apply a nonparametric clustering algorithm onto those dynamical models, and train a classifier to obtain a mapping from system states to dynamically distinct modes. We later use this mapping to identify a system's hybrid mode in real-time. For the second part, we use Koopman operators [**2**], which allow us to generate an approximation of the within-mode continuous dynamics. Both of these approaches are described in more detail in the subsections below.

### 6.2.1. Nonparametric Clustering of Dynamics

Hybrid dynamical systems evolve according to distinct dynamics in different regions of the state-space manifold and as a result are often represented as finite automata with discrete nodes [**83**]. Here we propose NCD—a system identification algorithm that allows us to synthesize finite representations of dynamical systems from data. Since hybrid systems express distinct dynamics based on state-space dependencies, local estimates of the dynamics differ about the boundaries specified by the system's guard equations. NCD generates partitions in the state-space manifold, where local estimates of the system dynamics differ. It numerically approximates boundaries on the manifold that delineate which mode governs the system at a given point and, as a result, identifies transitions between modes in hybrid systems, otherwise defined by guard equations.

Given a dataset representative of all dynamic modes in a hybrid system, NCD first segments the dataset into subsets and then generates local estimates of the system dynamics from each subset. Subsets can have various temporal lengths, they can be independent or overlap; however, they should always be continuous in time and have the same dimensionality. While any function approximation technique that produces a numerical closed-form model of the system dynamics can be used, we use generalized linear regression to generate Koopman operators (see Section 6.2.2), because their representation makes them easy to use in control settings [**193**]. In Koopman operators, states $x \in \mathbb{R}^n$ are lifted into a higher dimensional space through a nonlinear transformation of the original state-space by basis functions $\Psi(x) = [\psi_1(x), ..., \psi_N(x)]^T \in \mathbb{R}^N$ $s.t.$ $\psi_i(x) \in \mathbb{R}$.

The collection of local dynamic models is then made into a list, where we apply unsupervised learning techniques to divide the list into classes of distinct dynamic models based on a distance metric. For models generated by generalized regression, the list contains the weight matrices from each local model, $L = [W_0, .., W_{S-1}]$ from $S$ subsets of data. For other kinds of parametrized models, the list may consist of parameter vectors. If the number of hybrid modes is known ahead of time, one can apply parametric clustering techniques and prespecify the expected number of clusters to form. Nonparametric clustering techniques are useful for analyzing systems where the partitions are not as well-defined. We apply Hierarchical Density-Based Clustering for Applications with Noise (HDBSCAN) to perform nonparametric clustering [**33**].

Once all dynamic models in the list have been assigned a class label by the clustering algorithm, one can extend the class labels to each model's corresponding subset of data. To map the class labels onto the state-space, we train a Support Vector Machine (SVM) on the labeled data to generate an indicator function $\Phi(x)$ [**15**]. In particular, we generate an indicator function based on nonlinear transformations of the data, $\Phi(\Psi(x)) = i$, where $i \in \{0, ..., B-1\}$ of $B$ discerned dynamic modes. The indicator function specifies what mode the system is currently in, similar to the guard equations in hybrid systems. Algorithm 1 summarizes the NCD procedure for generating data-driven hybrid mode transition boundaries generally.

**Algorithm 2** Nonparametric Clustering of Dynamics (NCD)

```
Input: Dataset $X = [x_0, ..., x_M]$, function approximation technique with closed-form model. Here, we use generalized linear regression to generate Koopman operators, described in Section 6.2.2.
Procedure:
    Split $X$ into $S$ subsets
    Estimate system dynamics locally for each subset of the dataset $X_{a:b} = \{X[a], ..., X[b]\}$ using $W = FunctionApprox(X_{a:b})$
    Construct list of dynamic models $L = [W_0, ..., W_{S-1}]$
    Apply nonparametric clustering to $L$ and label all $W_i$'s with one of $B$ discerned classes $\{C_0, ..., C_{B-1}\}$
    Label all points in $X$ with the label $l \in \{0, ..., B-1\}$ of the subset they were applied to
    Train an SVM, $\Phi(x)$, to project class labels directly onto the state-space
Return: Trained SVM indicator function $\Phi(x)$
```

The indicator function gives a closed-form estimate of the regions drawn by the guard equation boundaries, which allows one to predict switching times in hybrid systems via model predictive control. If the hybrid mode dynamics are not known *a priori*, one can generate data-driven models of each hybrid mode by segmenting the dataset according to the partitions designated by NCD and learning a model for each dynamic subsystem.

### 6.2.2. Koopman Operators

Koopman operators have been shown to be effective in modeling observable dynamical systems [**2**]. Formally, Koopman operators describe the time-evolution of dynamical systems in an infinite-dimensional function space [**28**,**106**]. While the infinite-dimensional Koopman operator is valuable as a theoretical construct, it is impractical for numerical applications. Through generalized linear regression, one can synthesize finite-dimensional approximations of the Koopman operator by considering nonlinear basis functions of state $\Psi(x) = [\psi_1(x), ..., \psi_N(x)]^T$ and their evolution in time [**193**]. The regression generates an operator $K$ that minimizes the residual $r(x_k)$ in $\Psi(x_{k+1}) = K\Psi(x_k) + r(x_k)$ through the least-squares optimization

$$\min_K \frac{1}{2} \sum_{k=1}^{M-1} ||\Psi(x_{k+1}) - K\Psi(x_k)||^2. \tag{6.1}$$

The optimization has closed-form solution $K = AG^\dagger$, where † denotes the Moore-Penrose pseudoinverse, and the matrix components $A$ and $G$ are

$$\begin{aligned} G &= \frac{1}{M} \sum_{k=1}^{M-1} \Psi(x_k)\Psi(x_k)^T \\ A &= \frac{1}{M} \sum_{k=1}^{M-1} \Psi(x_{k+1})\Psi(x_k)^T. \end{aligned} \tag{6.2}$$

We can apply NCD in conjunction with finite-dimensional Koopman operators to generate data-driven estimates of the state-space boundaries designated by the guard equations and to synthesize the dynamical models of each hybrid mode.

## 6.3. Example 1: Control of a Simulated SLIP

The spring-loaded inverted pendulum (SLIP), often used as a simplified model of human running, is an example of a hybrid system with known dynamics. Although governed by relatively simple equations, it is unstable if left unassisted. Here, we use a simulated SLIP to demonstrate the joint capabilities of NCD and Koopman operators for predicting hybrid mode switching and identifying bimodal dynamics for control. In simulation, we successfully use the data-derived approximation of the SLIP's hybrid dynamics to generate forward motion using model predictive control.

### 6.3.1. Switching Time and Dynamics Identification

For all simulations, we use a 2D SLIP model described by a state vector $x = [x_m, \dot{x}_m, z_m, \dot{z}_m, x_t]$, where $x_m$ and $z_m$ are the coordinates of the mass, and $x_t$ is the coordinate of the toe, and a control vector $u = [u_{stance}, u_{flight}]$, where $u_{stance}$ is the leg thrust applied during stance and $u_{flight}$ is the toe velocity control applied during flight. SLIP dynamics as described in [**95**] are used. We begin by generating 30 seconds of training data using a simulation of the SLIP hopper controlled by a model predictive controller with knowledge of the correct SLIP dynamics.

We proceed with system identification by employing NCD, described in Section 6.2.1. The algorithm generates an indicator function that maps from states to SLIP hybrid modes. We verify the mapping by directly comparing against the solution of the analytical guard equation [**95**] and find that the data-driven indicator function is able to detect hybrid modes with near perfect accuracy for two tested trajectories—it is 100% accurate for constant-velocity forward hopping and 99.5% accurate for varying-velocity hopping with directional changes. A fragment of the

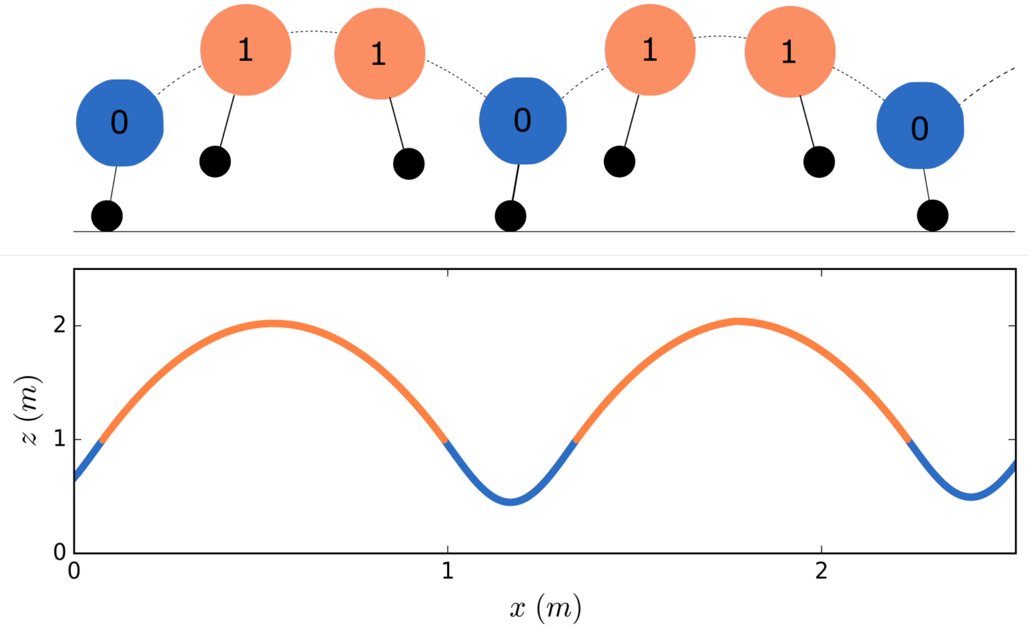

Figure 6.1. SLIP trajectory segmented according to an NCD-generated mapping. The data-driven indicator function is able to detect the SLIP's hybrid mode with near perfect accuracy for hopping on flat ground.

constant-velocity trajectory color-coded according to the SLIP's current hybrid mode is shown in Fig. 6.1.

Finally, using the NCD-generated mapping, we synthesize two separate Koopman operators for the SLIP's flight and stance modes. This obtained, entirely data-derived representation of the SLIP dynamics is then used for control.

### 6.3.2. Model Predictive Control

A model predictive controller (MPC) similar to [**6**] is used. Note that, although we make that choice, any model predictive controller that is capable of completing the task can be used. For

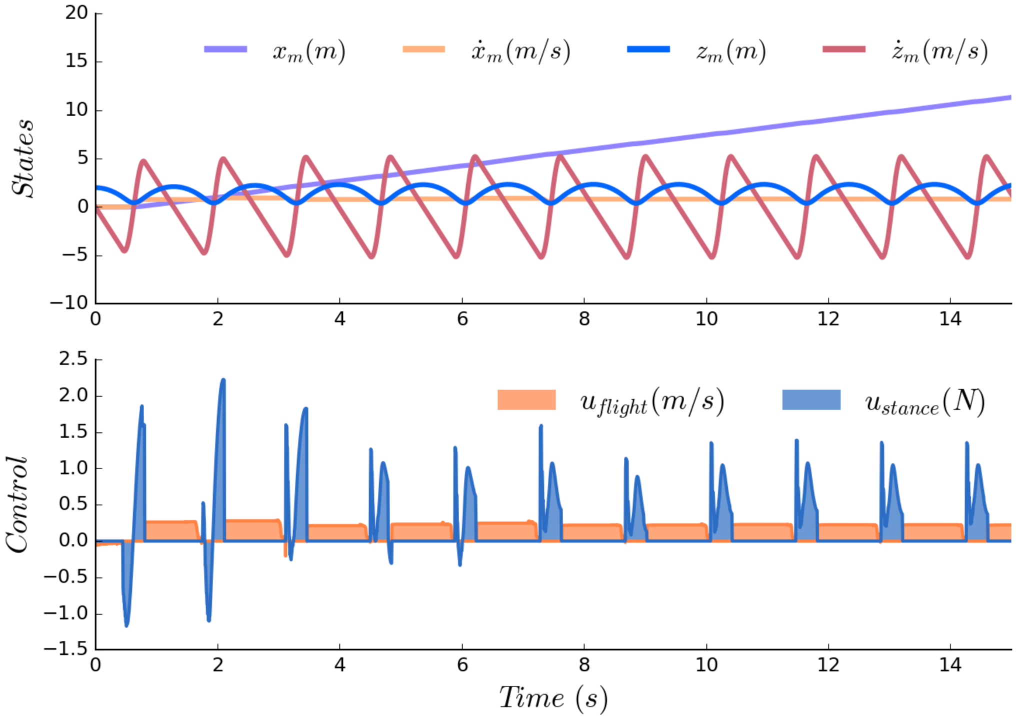

Figure 6.2. State trajectories and control history from a SLIP model simulation—the guard equation and system dynamics used in the simulation were learned solely from data. Note that the controller is able to keep the SLIP upright and moving forward at an average speed of $0.37m/s$, close to the desired $0.4m/s$.

the MPC, we define an objective function of the form

$$J = \frac{1}{2}\int_{t_o}^{t_f} \|x(t) - x_d(t)\|_Q^2 + \|u(t)\|_R^2 \delta t, \tag{6.3}$$

with $Q \geq 0$ and $R \geq 0$ being cost metrics on state error and control effort, respectively, and $x_d(t)$ being the desired trajectory.

For an example trial, we run the simulation with a desired trajectory $x_d(t) = [0, 0.4, 1.6, 0, 0]$, and a diagonal $Q$ matrix with $Q_{diag} = [0, 50, 100, 0, 0]$. As such, the controller tries to maintain the SLIP center of mass at a height of $1.6m$ and a forward velocity of $0.4m/s$. In this trial, the SLIP starts at a height $z_m = 2m$ and no forward velocity ($\dot{x}_m = 0m/s$). As shown in Fig. 6.2, the controller is able to keep the SLIP upright and moving forward at a velocity of $0.37 m/s$, close to the desired $0.4 m/s$, while having knowledge solely of the data-driven dynamics.

## 6.4. Example 2: Gait Partitioning for a Biped

Gait partitioning is an area of interest due to its promising applications in improving control of lower-limb assistive devices as well as in generating individually tailored physical therapies. Here, we segment the gait cycle into phases using the NCD algorithm, described in Section 6.2.1, and interpret obtained mode transitions based on established gait events. With control generation in mind, we are particularly interested in determining impact events, such as heel strike and toe-off, because they mark transitions between dynamically distinct modes. We validate the accuracy and latency of our predictions against external pressure sensors. Our results demonstrate successful gait partitioning for healthy flat-ground walking.

### 6.4.1. Walking Data Acquisition

Data was collected using EksoGT™—a robotic exoskeleton from Ekso Bionics, Richmond, CA, USA, visible in Fig. 6.3. When not actively in assistance mode, the device offers freedom to move in the sagittal plane, and to a limited extent in the frontal plane. It provides assistance solely in the sagittal plane. Both knee and hip joints can be used for assistance, where angular position and angular velocity can be measured at $500 Hz$ by encoders in all four joints. The ankle joints are passive and no sensory data is available.

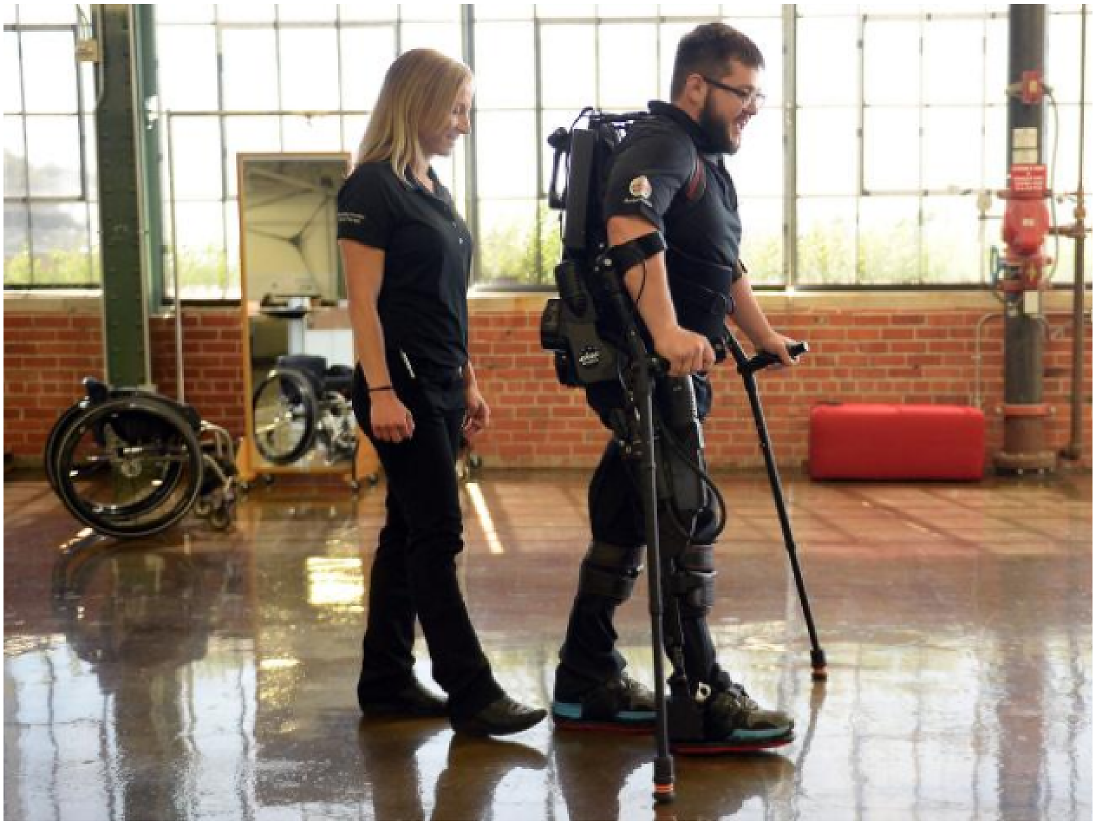

Figure 6.3. Ekso Bionics exoskeleton, EksoGT™, used for data collection. We use two sets of its sensors in this study: hip and knee encoders to generate kinematics-based dynamical models using the NCD algorithm, and foot-mounted pressure sensors at the heel and toe to validate our models against ground-contact events after partitioning.

For the purposes of this study, two minutes of data were collected of straight flat-ground walking from one healthy subject with previous experience walking in the exoskeleton. No assistance or resistance was provided to the wearer through motor activity; any perceptible resistance was passive from the mechanical structure of the device. A total of sixteen variables were recorded. Twelve of them (right/left knee angles, right/left knee angular velocities, right/left hip angles, right/left hip angular velocities, right/left knee motor currents, and right/left hip motor currents) were used for analysis. The additional four variables (right/left toe sensors and

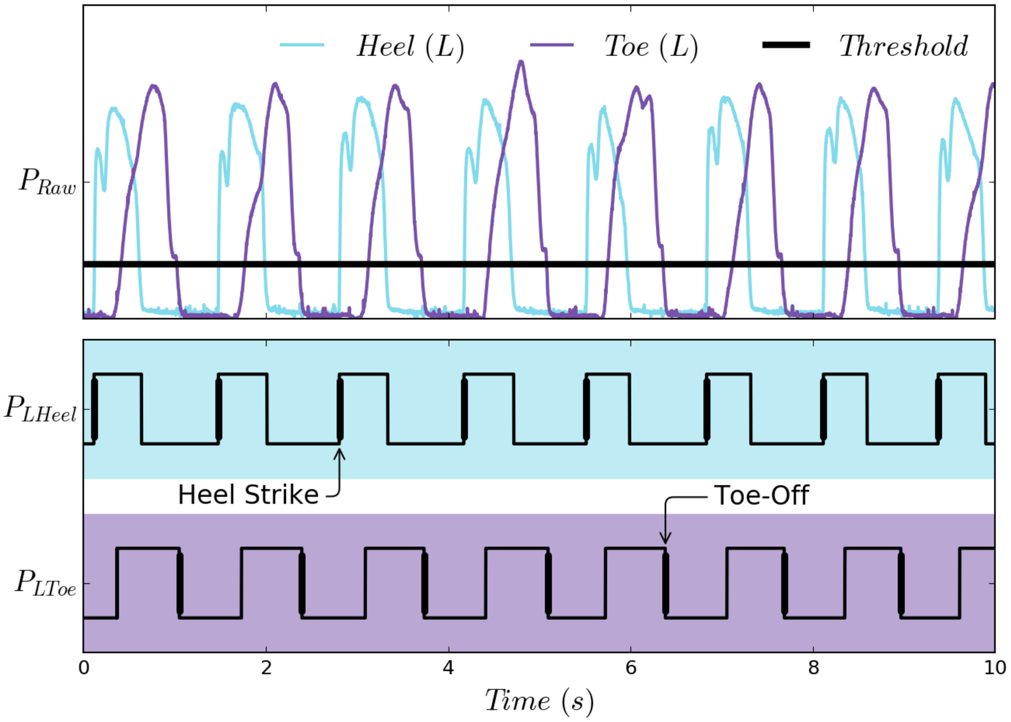

Figure 6.4. Example signals from two foot-mounted pressure sensors of the exoskeleton in Fig. 6.3. The top plot shows raw signals from the left heel and left toe sensors for several gait cyclces. These raw signals are thresholded to generate binary digital signals, visible underneath, which allow us to detect gait events such as heel strikes and toe-offs on rising edges of the heel signal and falling edges of the toe signal, respectively.

right/left heel sensors) were excluded from analysis and used solely for validation and verification of NCD-generated gait partitions.

### 6.4.2. Ground Truth for Gait Partitions

Gait cycles are generally defined from one foot strike to the subsequent foot strike on the same side. Clinically, they are often partitioned separately for each leg based on functionally critical events for that leg [**41**]. For this study, we are interested in generating gait partitions that capture critical changes in the behavior of both legs. We do not impose symmetry, but we choose states and basis functions symmetrically for right and left legs to allow the algorithm to remain equally

Table 6.1. Gait Decompositions

| | **2-mode decomposition** | **4-mode decomposition** | **6-mode decomposition** |
|---|---|---|---|
| **Phases** | right/left step | right/left pre-swing + swing and stance | right/left initial swing, terminal swing, and stance |
| **Events** | heel strike | heel strike and onset of knee buckling | heel strike, toe-off, and foot clearance at peak swing |
| **Classification rate*** | 100% | 100% | 100% |
| **Average offset*** | $11ms \pm 8.5ms$ | $22ms \pm 10.2ms$ | $36ms \pm 9.7ms$ |

*Classification rate and average offsets were calculated only for predictions of impact events, specifically heel strikes and toe-offs. These events were verified against signal from external pressure sensors not used in generating the data-driven gait partitions.

sensitive to recurring patterns on both sides. Moreover, we look for gait events representing contact with the ground, specifically heel strike and toe-off, because these impact events indicate transitions between dynamically distinct modes that are important for generating control in a robotic assistive device. As a tertiary objective, we are interested in sub-dividing swing, because the majority of active assistance during walking takes place during the swing phase. To generate data-driven approximations for transitions between the described dynamically distinct gait phases, we use the proposed algorithm—NCD.

In order to validate NCD-generated gait partitions, we utilize foot-mounted pressure sensors at the heel and toe. We collect analog signals from the pressure sensors and threshold them to obtain binary readings of whether the heel and toe are in contact with the ground. These processed sensor readings allow us to directly record heel strikes and toe-offs, establishing a notion of ground truth for transitions between stance and swing phases, as demonstrated in Fig. 6.4. As a result, for each gait decomposition, we can find mode transitions that correspond to these ground-contact events, and measure the offset between the prediction of the event and

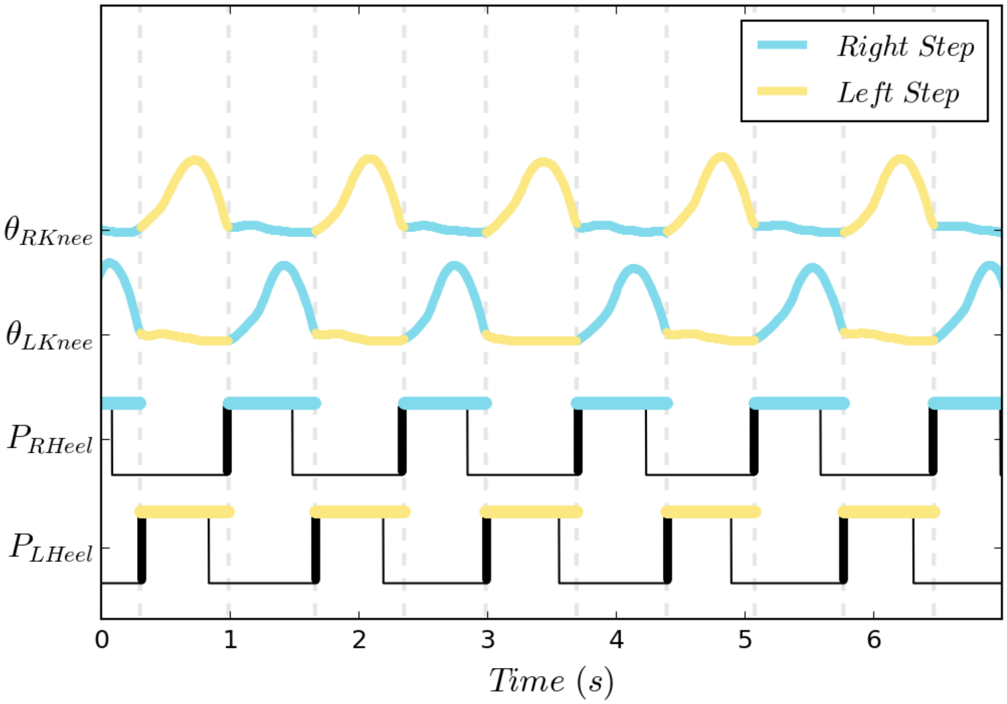

Figure 6.5. Two-phase segmentation of the gait cycle plotted against heel pressure sensors. Dotted gray lines depict NCD predictions of heel strikes, while bold black segments show ground truth. Identified phases are superimposed onto knee trajectories depicting their relationship to the gait cycle. We detect heel strikes without misclassifications and an average offset of $11ms \pm 8.5ms$. Note that pressure signals are not used in training for any decomposition.

the ground truth from the pressure sensors. The offset measurement is dependent on NCD reliably recognizing specific mode transitions and thus conveys both the precision and accuracy of event detection. We report the offset, or latency, as our validation statistic for each of the obtained gait decompositions.

### 6.4.3. Multi-Phase Gait Partitions Using NCD

Using NCD, we can segment gait into a range of decompositions from kinematic information, where each decomposition is determined by a set of distinct transition conditions that are recurrent throughout the gait cycle. In this subsection, we report gait partitions obtained from

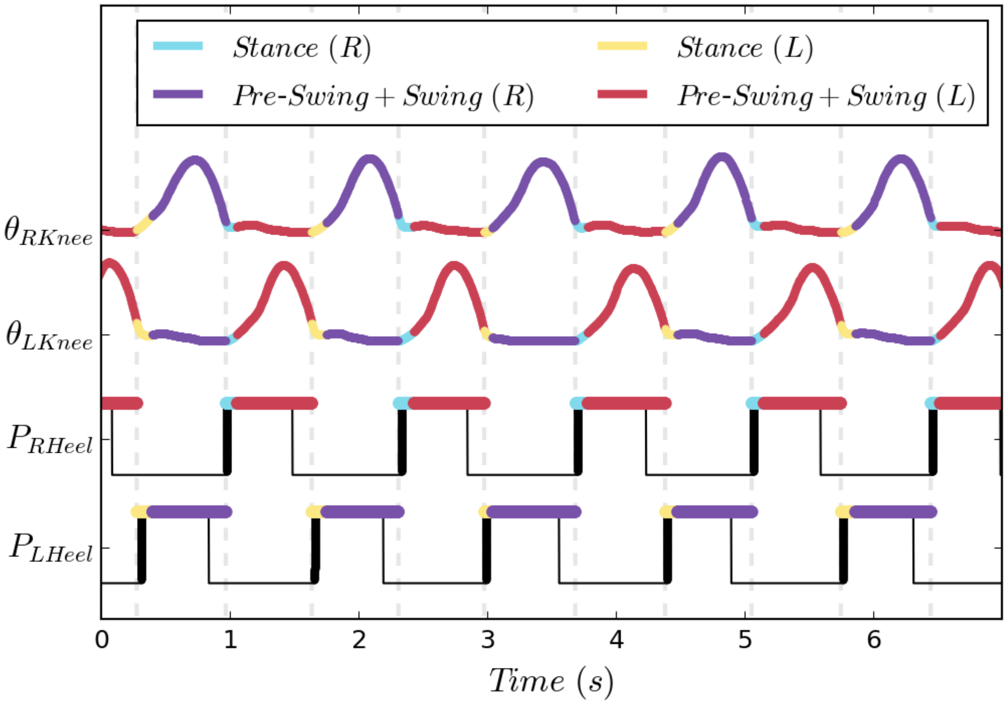

Figure 6.6. Four-phase segmentation of the gait cycle plotted against under-the-heel pressure sensors. Here, we are able to detect heel strike and the onset of knee buckling. We identify heel strikes with no misclassifications and an average offset of $22ms \pm 10.2ms$.

the same data using NCD with different basis functions. Specifically, we use 30-second fragments of the collected walking data (refer to Section 6.4.1 for details). For each partitioning, we expand the state space through quadratic, cubic, and/or trigonometric functions of the original 12 states. Depending on the choice of basis functions, and consequently the recognized transition events, we segment gaits into 2, 4, and 6 phases, where in each case the phase transitions correspond to easily interpretable gait events. We summarize our findings in Table 6.1.

In the 2-phase segmentation, we identify transitions that correspond to heel strikes. The gait cycle gets split into a right and left step, as shown in Fig. 6.5. We verify the accuracy of heel strike against pressure sensor signal and observe no misclassifications with $\sim 11ms$ latency.

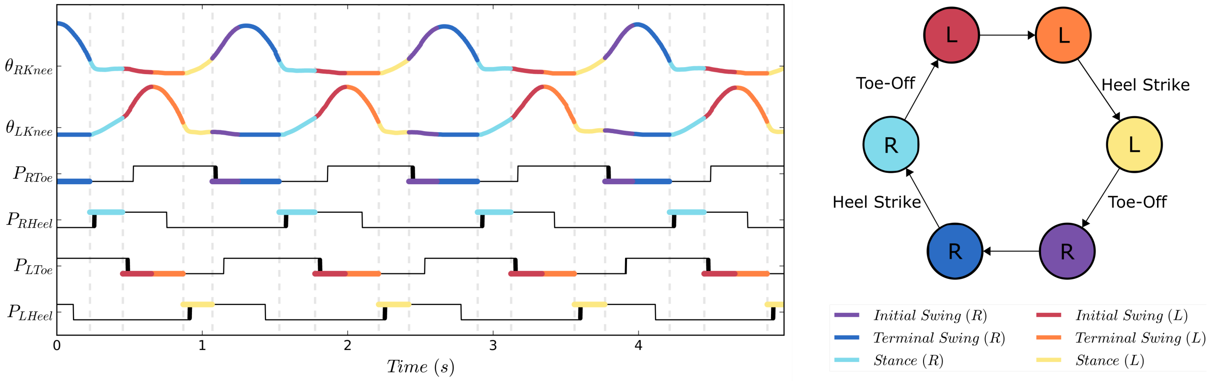

Figure 6.7. Six-phase segmentation of the gait cycle against foot-mounted pressure sensors. Dotted gray lines indicate NDC predictions of heel strikes and toe-offs, while the bold black segments indicate ground truth measurements. NCD generated reliably accurate predictions of phase-transition events with no misclassified transitions, and an average absolute prediction offset of 36*ms* with standard deviation of 9.7*ms*.

In the 4-phase partitioning, we identify transitions that correspond to heel strike and the onset of knee buckling, when the knee is ready to start bending but toe-off has not yet occurred. This decomposition, visible in Fig. 6.6, combines pre-swing and swing into one mode with a shortened stance as the second mode. Interestingly, this decomposition could be particularly relevant for control, because transition out of stance and into pre-swing can be interpreted as the cue for assistance, while heel strike (transition out of swing into stance) can be a signal for assistance to pause. Again, we verify the accuracy of heel strike predictions against heel strikes recorded via pressure sensors and observe no misclassifications with $\sim 22ms$ latency.

The 6-phase partitioning is the most complex of the ones reported here; we visualize it in Fig. 6.7. This partitioning allows us to identify both heel strikes and toe-offs, splitting the gait into right/left swing and stance. In addition, it divides swing into two segments of initial and terminal swing. Transition from initial into terminal swing corresponds to the clinically recognized foot clearance—when the swing leg passes the stance leg—and near maximal knee flexion. This segmentation could also be extremely useful for assistance, because it allows us to detect the start of swing. In situations when we want the subject to independently initiate a step, we might want them to complete pre-swing without assistance and wait for toe-off to apply control. As before, we verify the toe-offs and heel strikes against pressure sensors, detecting no misclassifications and an average offset of 36*ms*.

The list of possible gait partitions presented here is not exhaustive and is meant to illustrate the capabilities of the algorithm. Additional gait segmentations can be obtained, depending on what gait phase transitions are important in a particular application.

## 6.5. Discussion and future work

We demonstrate that NCD in combination with a data-driven dynamics identification technique, such as the Koopman operator, can be used to infer mode transitions and hybrid dynamics of a system. We learn a data-driven model of a SLIP hopper and successfully use it in simulation to generate control. We further show the ability to complete gait partitioning for normal walking, where switching times for gait phases correspond to gait events measured independently through ground-reaction forces.

What's worth noting is that our method could be particularly well-suited for modeling abnormal gaits. For one, we do not parametrize the gait partitioning prior to application of NCD—we do not make assumptions about either the number of dynamic modes or phase durations. Using an unsupervised learning technique, we have the flexibility to identify any recurrent movement pattern and, as a result, partition gaits with varying granularity. We can identify motion patterns in individuals, even when the movements are not part of the "correct" mode sequence in a specified task. Secondly, we do not require pre-labeling of phase transitions in the training data, which much prior work has relied on [**42**, **61**], and are able to train our segmentation algorithm on small amounts of data (experiments in this study used 30-second sets of walking data sampled at $500Hz$). This means that gait partitions could be generated for individual patients and updated continually as their impairment changes over time. Personalized gait partitions from impaired individuals could then be used to generate tailored therapies or to create personalized control patterns for assistive devices.

Finally, the algorithm could further be used to identify higher level behaviors, such as stair climbing, marching, or walking down an incline. Through real-time identification of a person's activities based on kinematic data, NCD could facilitate providing task-relevant assistance without the need for manual task specification. Future work will seek experimental validation of our algorithms for generating assistance in lower-limb exoskeletons—both for flat-ground walking and for task-varying movement.

## 6.6. Conclusion

The presented data-driven methods are a step towards generating real-time model predictive controllers for hybrid systems, such as lower-limb assistive exoskeletons. We show the ability to model and control a hybrid system with no *a piori* knowledge of the transition conditions or mode dynamics. We further show the ability to generate control-consistent gait partitions in a healthy individual. Long-term, our approach can lead to developing personalized models of gait dynamics for targeted assistance and/or rehabilitation in individuals with a range of walking impairments.

CHAPTER 7

# Conclusions and Future Directions

In this thesis, I present algorithmic approaches for improving the efficacy and functionality of assistive and rehabilitation robots by improving human-machine communication. I test the algorithms in simulation as well as in hardware with able-bodied and impaired individuals. In a recent review article in Annual Reviews, I describe the current state-of-the-art algorithms and open challenges in the field of human-machine communication [**96**]. Below, I summarize this thesis's contributions in the context of the field and outline directions for future work.

## 7.1. Emergence of Multimodal Dialogue

Modern collaborative robots often have some capability to interpret human intent. They can infer intent from biological signals, such as movement, EEG, or EMG, or from a direct human-generated input, such as joystick signal. These setups successfully facilitate one-directional communication from the person to the robot and enable task-specific interactions. However, they often enable little feedback to be provided back to the person and they do not facilitate bi-directional dialogue during pHRI. Flexible bi-directional communication protocols would enable more versatile human-robot collaboration.

In Chapter 4, I propose algorithms that enable multiple agents to create and agree on communicative conventions from task-oriented interaction. In this work, I use neural networks and reinforcement learning to facilitate the emergence of concise and effective dialogue between two artificial agents. As an extension of the project, I test the feasibility of neural networks to learn symbolic vocabulary from physical interfaces, such as a sip-n-puff, operated by a person. I find that with just a handful of demonstrations per symbol, the algorithm can learn to recognize the symbol from a continuous human-generated signal, even if the symbol is temporally distorted compared to the original demonstration.

The methods presented in Chapter 4 are a departure from conventional algorithms for personalizing human-machine communication paradigms. Using RL, I enable communicative conventions to *emerge* during task-oriented collaborative interaction. In contrast, existing methods either provide no adaptation or they adapt the mapping of command inputs onto robot actions during an initial calibration period. As described at the end of the chapter, follow-on work should accommodate learning temporally extended symbols. This could improve the person's ability to use a low-dimensional control interface to control a higher-dimensional action space of a robot. Future work should also expand the experimental work to accommodate multi-modal and bi-directional communication, e.g., through haptics, gestures, or bioelectrical activity, with both the person and robot actively generating and receiving communicative signals.

## 7.2. Learning New Tasks

Because robotic communication and assistance often take place in the context of a task, task teachability is an important feature for a collaborative robot. In Chapters 5 and 6, I propose two methods for learning task representations from human demonstrations. Method 1 enables learning subtasks from continuous demonstrations, method 2 enables learning from negative demonstrations—presentations of what not to do. Importantly, both approaches avoid defining tasks in terms of optimal movement trajectories, but rather focus on capturing movement dynamics and task statistics, respectively.

Both of the methods presented in this thesis rely on a well-defined set of features that can capture task success. Features can include the xyz-position of a robot end-effector, the velocity of a robot's joints, locations of objects in the robot's workspace, or any nonlinear transformations of the observed states. The proposed methods would be more generalizable if the relevant features could also be learned from demonstration. Extensions of this work could enable negative demonstrations to be employed to select task-relevant features. As an example, we can quantify the variability of each feature among positive demonstrations as well as between positive and negative demonstrations. If the feature varies largely among positive demonstrations, it is likely irrelevant to the task. Contrarily, if it varies between positive and negative demonstrations, it is likely key to task success. Contrastive learning could help determine a set of task-relevant features.

Lastly, follow-on work should combine learning task definitions with the process of communication emergence. This research should include investigating how to enable continual learning of new task definitions, so that new skills can continue to be added to the robot's task library throughout use. It could also explore how to accelerate the emergence of communicative conventions given familiar tasks and task-oriented coordinated behaviors. Robust human-robot communication could in turn help the robot learn representations of new tasks.

## 7.3. Human-Machine Communication for Neurorehabilitation

As described by John Krakauer and David Reinkensmeyer in a recent publication [**109**], there is ongoing debate about the value of robotic movement training devices in clinical rehabilitation practice. In spite of decades of research, there are few studies that show improved patient outcomes as a result of robot-assisted therapy compared to conventional training. That said, the evidence is clear that robots can be useful scientific tools in the study of neuromotor disorders and in the development of novel rehabilitation protocols [**109**]. My doctoral work focuses on two aspects of human-machine communication for rehabilitation:

- I create measures of dynamic motion to quantitatively assess dynamic motion capabilities and to study their variation under conditions of reproducible, titrated assistance;
- I evaluate how to provide adaptive feedback and assistance to accelerate learning and relearning of dynamic motor skills.

### 7.3.1. Robot-Assisted Assessment of Movement

In Chapters 2 and 3, I propose and validate a novel methodology, called QADR, for assessing an individual's motion bandwidth and their ability to generate a dynamic response. I use the proposed approach to quantify dynamic deficit after hemiparetic stroke and begin to identify the mechanistic causes of loss in dynamic function due to a stroke. My results show that stroke survivors experience a reduced motion bandwidth within their reachable workspace. The loss in function is likely due to, among others, a shift in reliance from direct to indirect pathways for voluntary motor control.

While most quantitative assessment techniques measure quasistatic motion capabilities, such as strength or reaching range, QADR aims to quantify dynamic performance. A reliable and accurate measure of dynamic motion enables novel investigation of motion bandwidth in both able-bodied and impaired individuals. QADR can help characterize the neural circuitry responsible for high-bandwidth motion—it can quantify the bandwidth for incorporating different types of sensory inputs during voluntary motion or help to evaluate the role of motor lateralization during dynamic motion.

In its current setup, QADR can provide titrated support of the arm against gravity, which I have used to study the pathways responsible for motor control after a stroke. Future work should use QADR to further our understanding of the mechanistic causes of post-stroke dynamic deficit as well as to explore dynamic deficit across a wider range of neuromotor disorders.

### 7.3.2. Robot-Assisted Motor Learning

A number of independent studies show that traditional robot-assisted training—focused on repetitive, assisted execution of motion trajectories—does not significantly improve the efficacy of physical therapy regimens [**109**]. However, there is significant promise in task-oriented, adaptive therapy that is specifically targeted towards neural mechanisms underpinning impairment.

The assessment tools presented in Chapters 2 and 3 offer a quantitative measure of dynamic capabilities that can be used to continuously adapt assistance during task-based training. A frequency-based metric can be calculated in real time and used in closed loop to update robotic assistance as needed based on dynamic performance. In my master's thesis [**91**], I show that forceful interaction—adjusted in real-time based on task performance—can accelerate learning of a dynamic task in unimpaired individuals [**72, 95**]. Future work should evaluate the efficacy of adaptive interventions designed using the tasks and quantitative measures presented in this thesis.

Moreover, I show that frequency-based metrics generalize across 3 tasks for quantifying dynamic deficit. Future work could adapt the proposed assessment methods to assess dynamic capabilities using wearable sensors during everyday tasks. Ideally, robots would enable active rehabilitation and therapeutic benefits while also assisting with the completion of everyday tasks—combining assistance with active rehabilitation [**7**].

# APPENDIX A

## Measuring motion bandwidth using pHRI

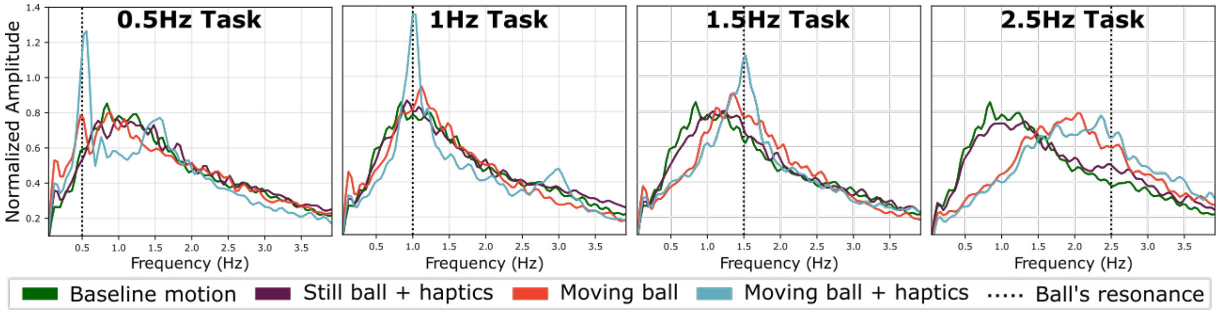

Figure A.1. Frequency decomposition of motion during ball-in-bowl task. Aggregate spectra for all subjects across four tasks and four experimental conditions.

Fig. A.1 includes aggregate frequency spectra for all subjects across four experimental conditions and four tasks. The comparisons relevant to our study hypotheses are included in the main text of the thesis (Chapter 2). We include the aggregate data here to allow the curious reader to refer to it for context.

APPENDIX B

# Robotic assessment of motion bandwidth and dynamic deficit

## B.1. Participant inclusion/exclusion from aggregate analysis

We recruited and consented a total of 20 stroke survivors during the study described in experiment 1. However, data from only 13 participants was included in the final analyses (10 participants for the ball-in-bowl task, 12 participants for the nail-and-hammer task). There are two reasons why participants were excluded from the final analyses:

- Participants did not complete the entire experimental protocol due to unforeseen circumstances, e.g., the robot malfunctioning during the experiment, or a participant's unwillingness to come back for a second session.
- Participants were unable to complete the ball-in-bowl task with their nonparetic arm. If in the frequency spectra for the nonparetic arm, we did not observe a peak at resonance

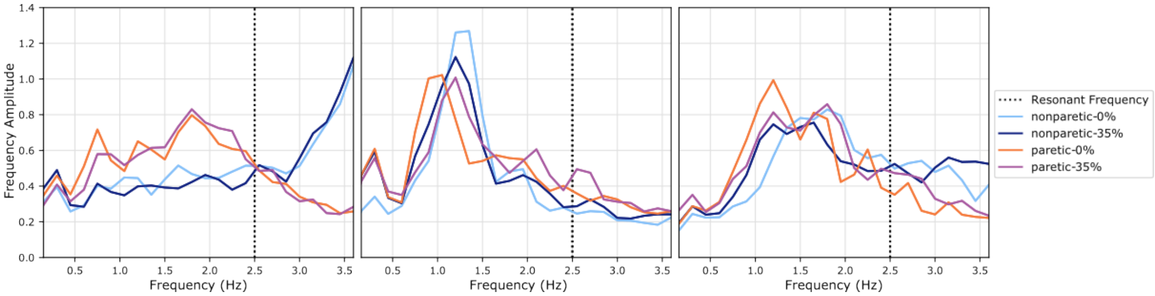

Figure B.1. **Aggregate frequency spectra for the ball-in-bowl task at 2.5Hz for 2 participants excluded from the analysis.** Note that the nonparetic peaks do not match the resonant frequency. The peaks are visibly offset from 2.5Hz.

and/or the peak was consistently shifted, we cannot be certain that the participant understood the task. Three example plots are included in Figure B.1. This happened a few times for individuals with severe impairment, likely due to cognitive dysfunction that can accompany physical symptoms in hemiparetic stroke.

## B.2. Exclusion of the 0.5Hz ball-in-bowl task

As discussed earlier in the article, the ball-in-bowl task has been validated with unimpaired individuals in a prior study [**101**]. During validation, as one of the experimental conditions,

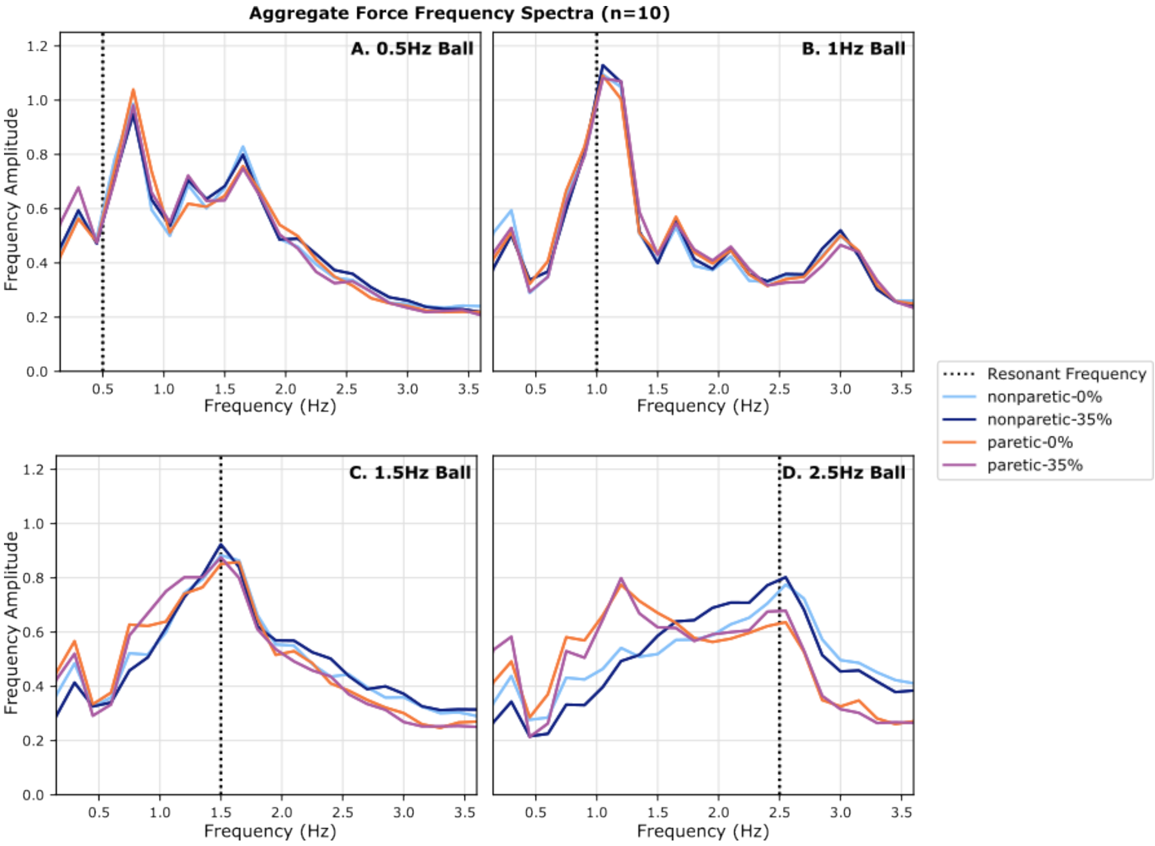

Figure B.2. **Aggregate frequency spectra for the ball-in-bowl task for all participants included in the analyses (n=10).** Note that the nonparetic peaks match resonance for tasks for all resonant frequencies except 0.5Hz. As a result, the 0.5Hz task is not included in the current analyses.

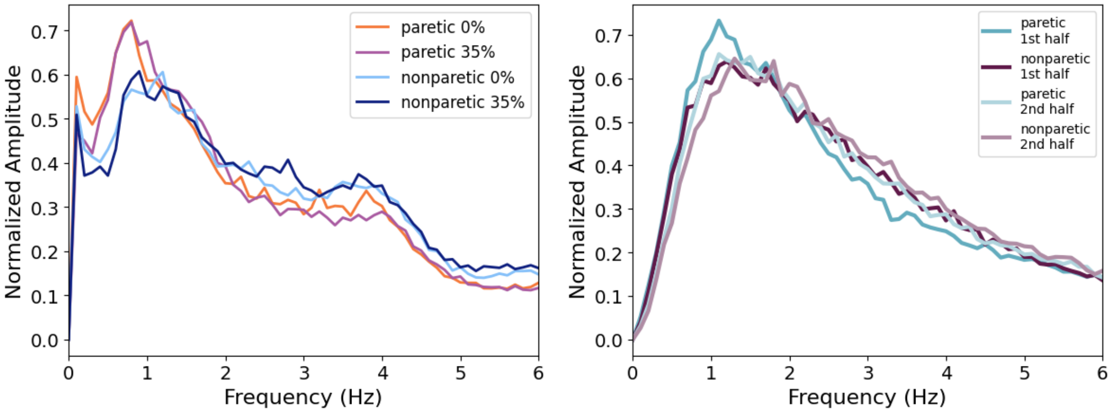

Figure B.3. **(left) Aggregate frequency spectra for the nail-and-hammer task for all participants included in the analyses (n=13). (right) Aggregate frequency spectra for the object hit task for all participants (n=48).**

participants were asked to complete trials without a moving ball—and hence without an associated resonant frequency. When participants' sole goal was to collect targets without needing to counteract the oscillations of a ball, they exhibited a default movement frequency between 1 and 1.5Hz. This result suggests that the ball-in-bowl tasks with different resonant frequencies require strategically different approaches from the participant. For the 0.5Hz task, participants need to move slower than their default frequency range to succeed at the task. For the 1Hz, 1.5Hz, and 2.5Hz tasks, they need to move within or faster than their default frequency range.

In our validation study, we find that unimpaired individuals are able to make the switch in strategy—we see statistically significant peaks near resonance for all four tested frequencies [**101**]. In contrast, the data from the current study with impaired individuals shows a different trend. We visualize the aggregate frequency spectra for all four frequencies of the ball-in-bowl task in Fig. B.2. While we see distinct peaks for the nonparetic arm near resonance for 1Hz, 1.5Hz, and 2.5Hz, we do not see a peak at resonance for the 0.5Hz task. As a result, we make the decision to not include the 0.5Hz task in our analysis.

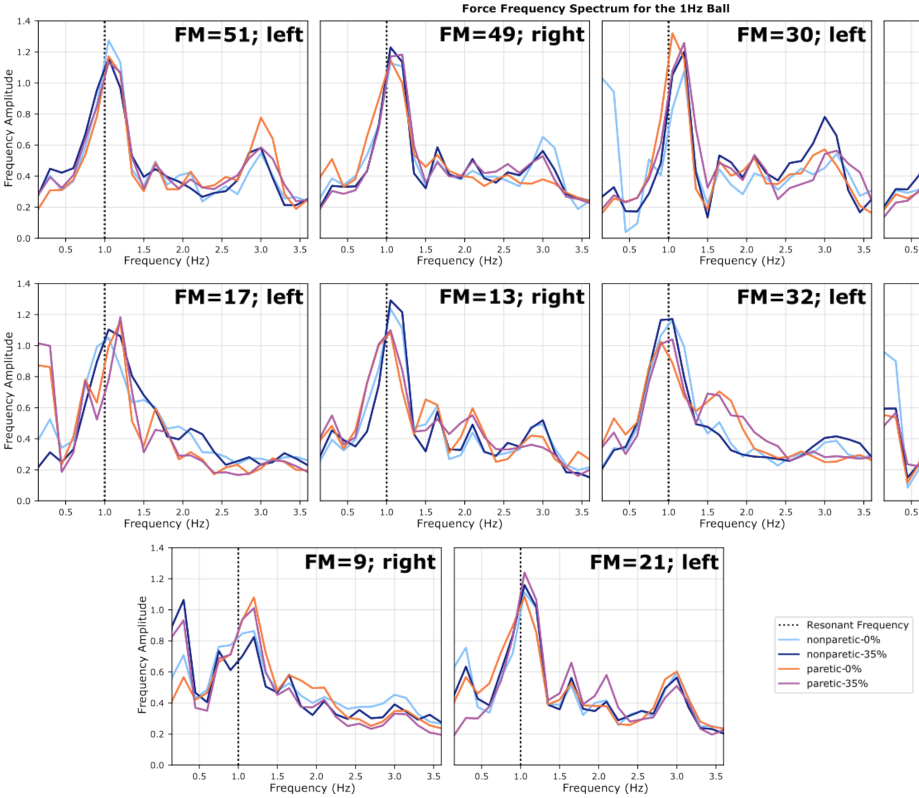

Figure B.4. **Frequency spectra for the ball-in-bowl task at 1Hz for all participants (n=10).** Each plot is labeled with the participant's FMA score and affected side (right/left arm).

Like mentioned in the previous section, one of the inclusion criteria for a participant in this study was their ability to understand and successfully complete the task, as measured by their performance at the task with their nonparetic arm. Almost all impaired participants did not exert enough energy near resonance during the 0.5Hz task to generate a visible peak near resonance, even with their nonparetic arm, as visible in Fig. B.2 (top left), to warrant inclusion. While we exclude the 0.5Hz task from the current analysis, a future study could investigate this change in behavior compared to unimpaired individuals.

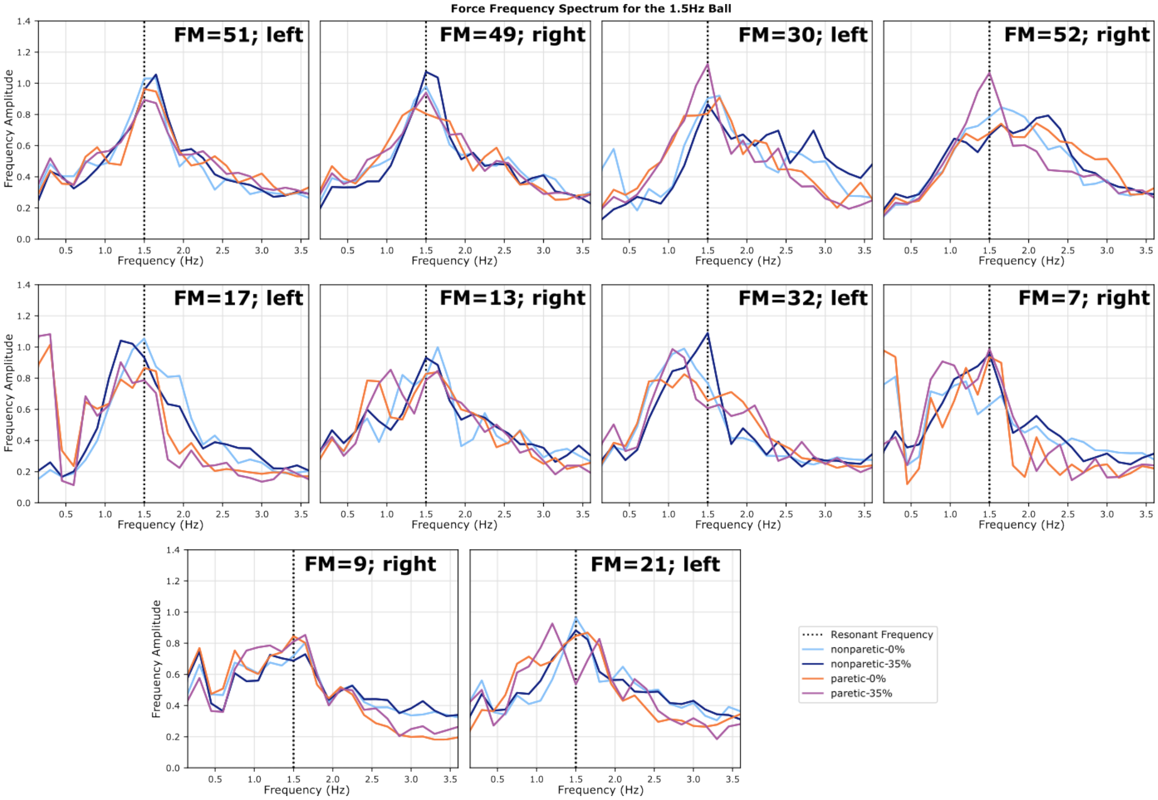

Figure B.5. **Frequency spectra for the ball-in-bowl task at 1.5Hz for all participants (n=10).** Each plot is labeled with the participant’s FMA score and affected side (right/left arm).

## B.3. Aggregate results for all tasks

For the reader’s reference, we visualize aggregate frequency spectra for all tasks. Fig. B.2 includes the aggregate frequency spectrum for the ball-in-bowl task (n=10). Fig. B.3 includes the aggregate frequency spectrum for the nail-and-hammer task (n=12) and the object hit task (n=48). The frequency spectra for the ball-in-bowl task are filtered with a low-pass Butterworth filter (and a cutoff frequency of 7Hz) for visual clarity.

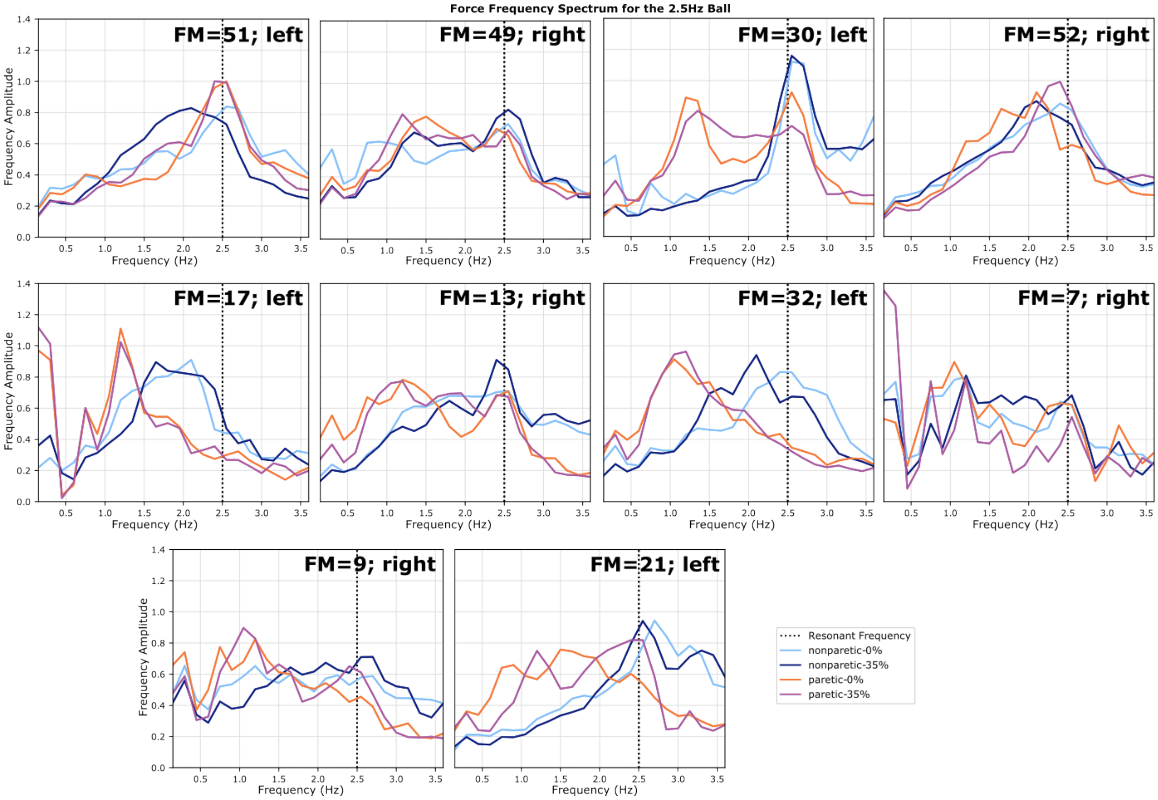

Figure B.6. **Frequency spectra for the ball-in-bowl task at 2.5Hz for all participants (n=10).** Each plot is labeled with the participant's FMA score and affected side (right/left arm).

## B.4. Individual participant data for the ball-in-bowl task

The study includes 10 participants' data for the ball-in-bowl task. For the reader's reference, frequency spectra for individual participants are visualized in Fig. B.4, Fig. B.5, and Fig. B.6 for tasks with resonant frequencies at 1Hz, 1.5Hz, and 2.5Hz, respectively. Individual frequency spectra are filtered with a low-pass Butterworth filter (and a cutoff frequency of 7Hz) for visual clarity.

APPENDIX C

# Emergence of communicate conventions

## C.1. Implementation details

Agent architecture is visualized in Fig. C.1. All models were implemented in Python using JAX [**23**] and Haiku [**82**]. For training, we used GPUs V100 and P100.

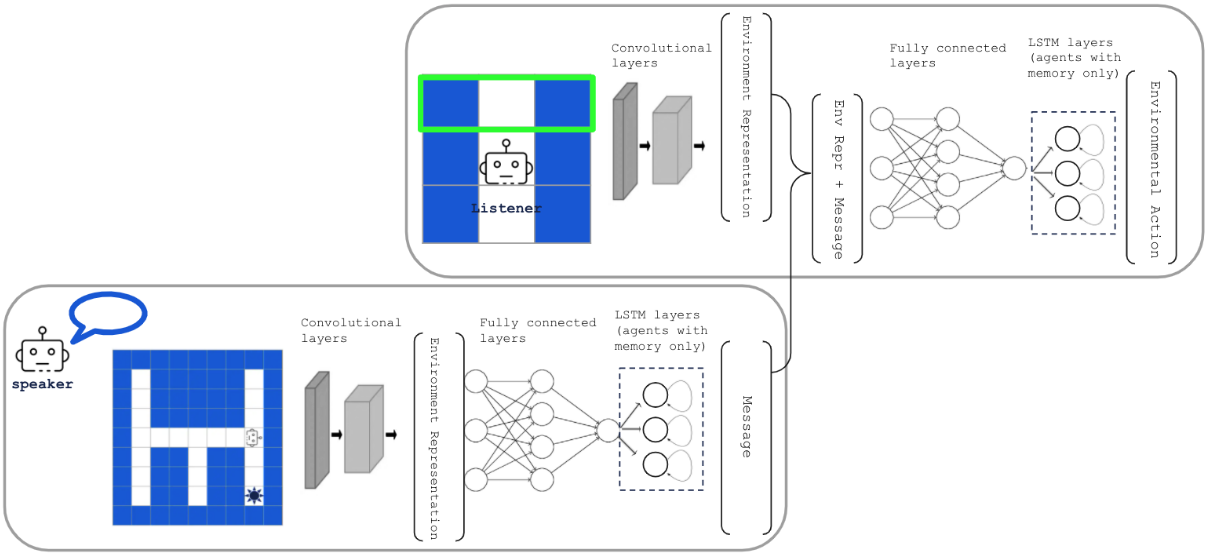

Figure C.1. **Agents' architecture and mechanism of communication.**

## C.2. Experimental comparison of upfront vs. real-time messaging

**Real-time communication improves language emergence compared to upfront messaging.** As an additional experiment, we compare real-time communication (multi-step cheap talk)

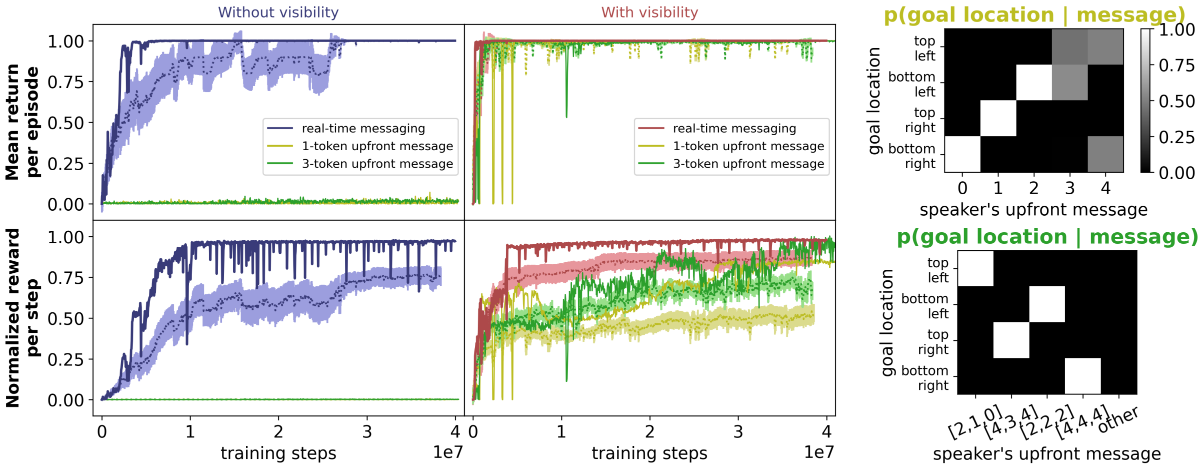

Figure C.2. **Comparison of upfront and real-time messaging.** Agents have memory. Real-time messaging improves convergence on a successful communication protocol. With upfront messaging, agents learn to solve the task before episode timeout when the listener has partial visibility. However, convergence is slow and agents are unlikely to solve the task in the optimal number of steps.

with upfront messaging. In upfront messaging, the speaker generates a 1-, 2-, or 3-token message at the beginning of each episode and that message gets broadcasted to the listener at each timestep throughout the episode. It is a form of *unsituated* communication, where the speaker can only generate one message at the beginning of an episode. In both scenarios, the theoretical capacity of the communication channel allows the agents to communicate the necessary information, whether the agents choose to communicate directions, e.g., ‘turn right’, or a goal address, e.g., ‘top left corner’. With upfront messages of length 1, 2, and 3, the speaker has 5, 25, or 125 unique messages available for communication, respectively.

In Fig. C.2, we plot the best agent pairs as well as the mean over replicas with the same hyperparameters as the best agent pair. With both real-time and upfront messaging, agents succeed in establishing a successful communication protocol when the listener has partial visibility—they converge to a mean return of 1 per episode. With no visibility for the acting agent, agent

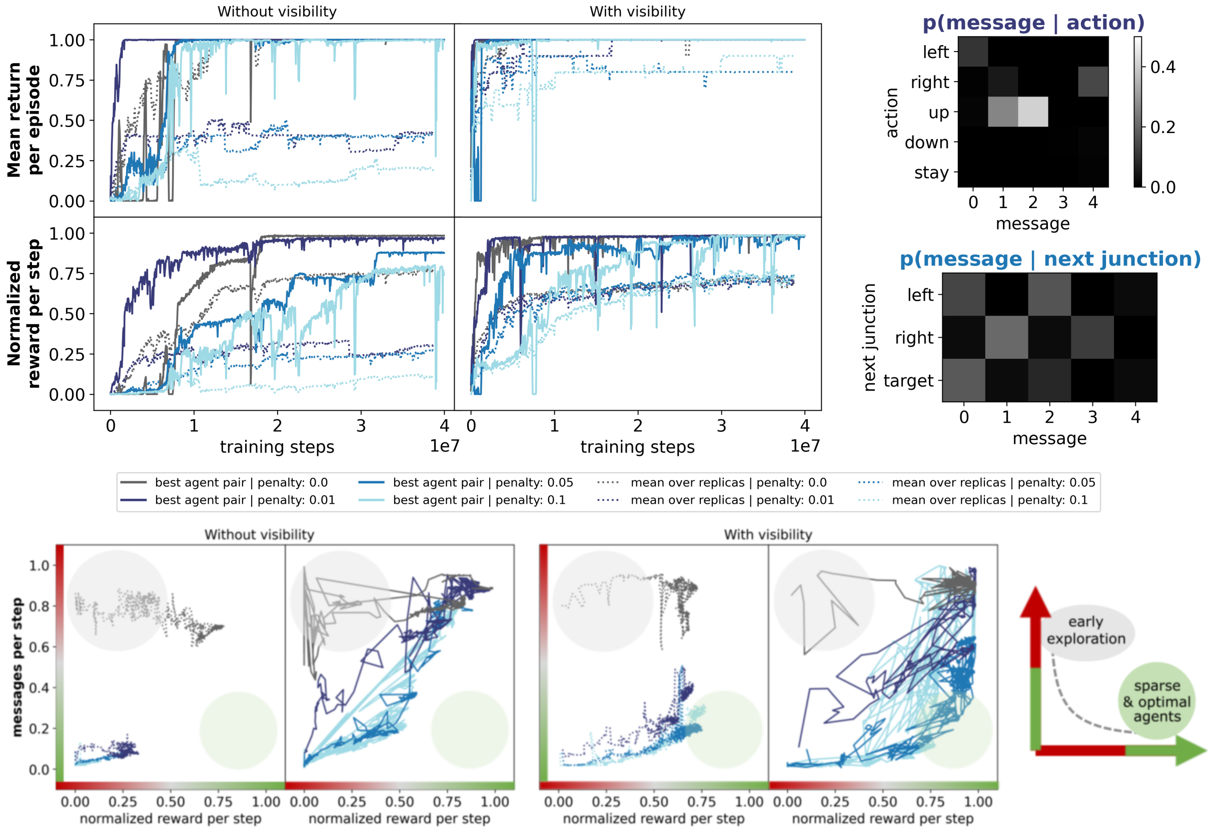

Figure C.3. **Communication emergence with a cost on communication effort.** Agents with memory; speaker experiences a per-message penalty. With memory, overall performance improves—more agent pairs converge to an optimal solution. However, relative to baseline, agents with a message penalty have more difficulty learning to jointly solve the task. In the heatmaps on the right, we illustrate the communication protocol of the best agent pairs. Note that the penalty is largely ineffective—the agents send many non-zero messages per episode.

pairs with upfront messaging do not succeed at solving the task. Moreover, the real-time agents are more likely to converge to an optimal solution, being able to solve the T-maze task in 9 moves. With 1 upfront token, even the best agents learn to at-best solve the task in 12 steps. These agents seem to reliably learn unique messages to encode the action required at the first turn or the right/left part of the address, but they do not establish a unique encoding for the top/bottom portion of the address, as visible in the top heatmap in Fig. C.2. With 3 upfront

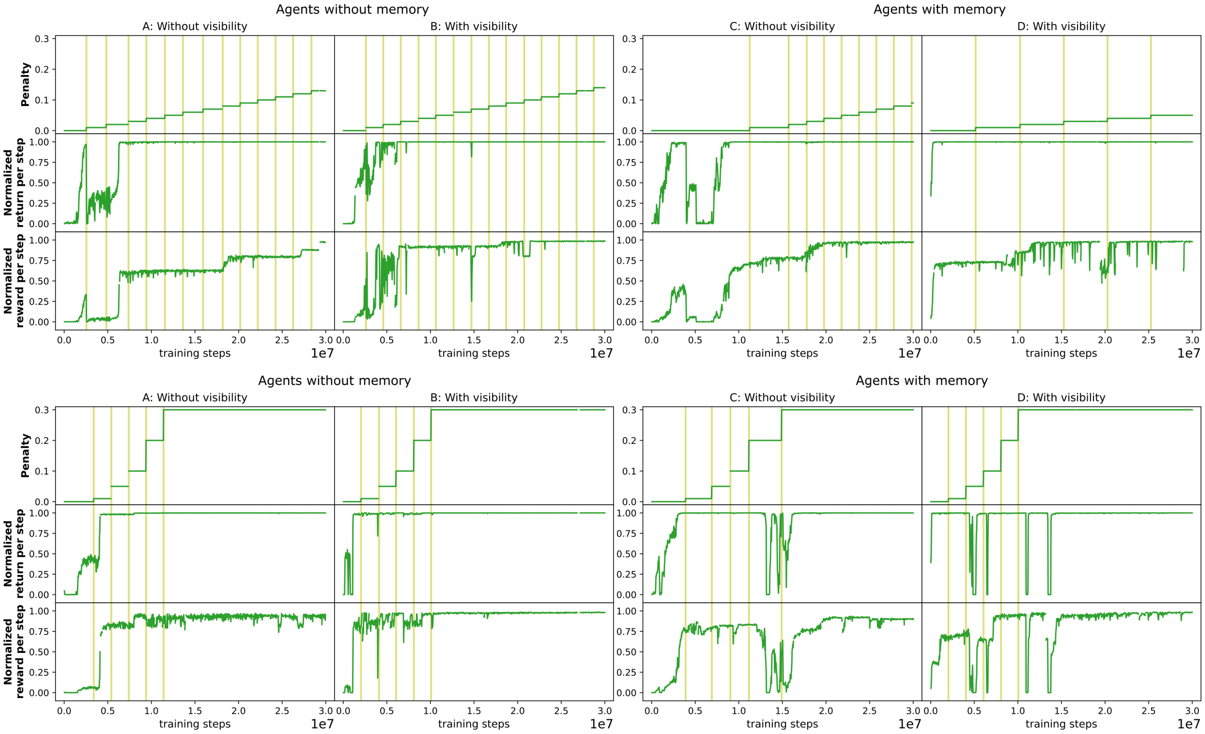

Figure C.4. **Best agent pairs training with a penalty curriculum.** (top) Curriculum with mapping $m_{p1}$ (bottom) curriculum with mapping $m_{p2}$.

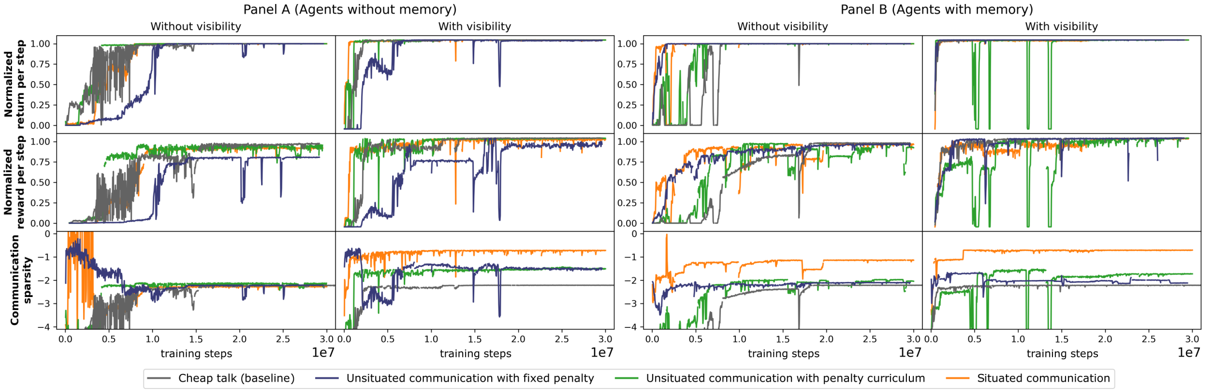

Figure C.5. **Comparison of pressures for avoiding over-communication (best agent pairs) in the T-maze.**

tokens, the best agent pair agrees on 4 distinct symbols to encode the 4 possible goal locations. However, convergence is slow and on average agent pairs perform less optimally than under the

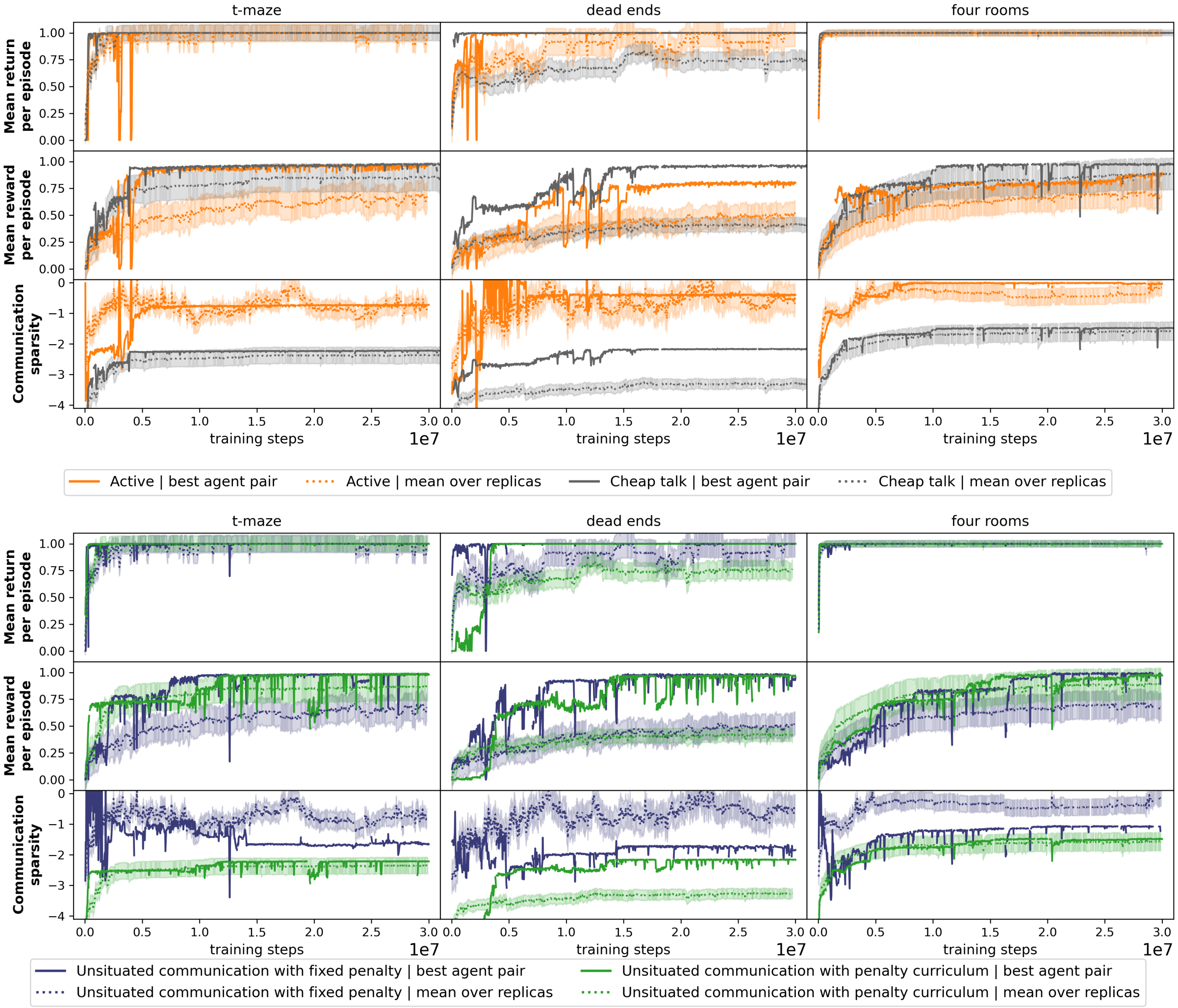

Figure C.6. **Agents in 3 environments** (top) using *situated* communication compared to cheap talk, (bottom) using *unsituated* communication with a fixed penalty and penalty curriculum.

real-time communication paradigm. We hypothesize that there are benefits to allowing communication to emerge from multi-step interactions. Our findings suggest that it is easier for agents to learn to communicate if they can exchange information when it is immediately actionable.

## C.3. Additional results for agents with a per-message penalty

In the main part of the thesis, we present results for experiments on agents without memory. In Fig. C.3, we present the same experiment on agents with memory. Overall, agents with memory perform better than agents without memory. However, the trends remain the same—the penalty negatively impacts convergence and the best-performing agents that find optimal task solutions do not exhibit sparse communication protocols.

In Fig. C.4, we visualize the learning curves for the two curricula in parallel with the agents' progression through the curriculum stages. The curriculum with mapping $m_{p1}$ achieves better overall performance and is included in the comparison plots in the main part of the thesis (Fig. 4.3 and Fig. 4.6).

## C.4. Additional results on the impact of pressures on communication conciseness

In the main text, we plot mean learning curves selected based on the highest mean normalized reward per step for a given set of hyperparameters. Here, we include results from the same experiments as in the main text, but instead we plot learning curves for the best agent pair. Fig. C.5 includes the same comparison as Fig. 4.3 in the main text; Fig. C.6 includes the same comparison as Fig. 4.6. In Fig. C.6, we present the best performing agent pairs from our hyperparameter sweep as well as the mean over the 10 replicas with the same hyperparameters as the best performing pair. The best performing agent pairs are selected based on the metric of solution optimality (the normalized reward per step). Overall, the trends of the best performing agent pairs are consistent with the mean trends described in Chapter 4.

# References

[1] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In Int. Conf. on Machine learning (ICML), page 1. ACM, 2004.

[2] Ian Abraham, Gerardo De La Torre, and Todd David Murphey. Model-based control using Koopman operators. Robotics: Science and Systems (RSS), 2017.

[3] Ian Abraham, Ahalya Prabhakar, and Todd D Murphey. An ergodic measure for active learning from equilibrium. Transactions on Automation Science and Engineering, 2020.

[4] Tohid Alizadeh, Sylvain Calinon, and Darwin G Caldwell. Learning from demonstrations with partially observable task parameters. In Int. Conf. on Robotics and Automation (ICRA), pages 3309–3314. IEEE, 2014.

[5] Saleema Amershi, Maya Cakmak, William Bradley Knox, and Todd Kulesza. Power to the people: The role of humans in interactive machine learning. AI Magazine, 35(4):105–120, 2014.

[6] Alexander R Ansari and Todd D Murphey. Sequential action control: closed-form optimal control for nonlinear and nonsmooth systems. IEEE Transactions on Robotics, 32(5):1196–1214, 2016.

[7] Brenna D Argall. Autonomy in rehabilitation robotics: An intersection. Annual Review of Control, Robotics, and Autonomous Systems, 1:441–463, 2018.

[8] Michelangelo Bartolo, Alessandro Marco De Nunzio, Fabio Sebastiano, Francesca Spicciato, Paolo Tortola, Jan Nilsson, and Francesco Pierelli. Arm weight support training improves functional motor outcome and movement smoothness after stroke. Functional Neurology, 29(1):15, 2014.

[9] Chandrayee Basu, Erdem Biyik, Zhixun He, Mukesh Singhal, and Dorsa Sadigh. Active learning of reward dynamics from hierarchical queries. In Int. Conf. on Intelligent Robots and Systems (IROS). IEEE, 2019.

[10] Chandrayee Basu, Mukesh Singhal, and Anca D Dragan. Learning from richer human guidance: Augmenting comparison-based learning with feature queries. In Int. Conf. on Human-Robot Interaction, pages 132–140. IEEE, 2018.

[11] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. Fitting linear mixed-effects models using lme4. Journal of Statistical Software, 67(1):1–48, 2015.

[12] Randall F Beer, Julius PA Dewald, and W Zev Rymer. Deficits in the coordination of multijoint arm movements in patients with hemiparesis: evidence for disturbed control of limb dynamics. Experimental Brain Research, 131(3):305–319, 2000.

[13] Emelia J Benjamin, Salim S Virani, Clifton W Callaway, Alanna M Chamberlain, Alexander R Chang, Susan Cheng, Stephanie E Chiuve, Mary Cushman, Francesca N Delling, Rajat Deo, et al. Heart disease and stroke statistics—2018 update: a report from the american heart association. Circulation, 137(12):e67–e492, 2018.

[14] Joao Bimbo, Claudio Pacchierotti, Marco Aggravi, Nikos Tsagarakis, and Domenico Prattichizzo. Teleoperation in cluttered environments using wearable haptic feedback. In Int. Conf. on Intelligent Robots and Systems, 2017.

[15] C.M. Bishop. Pattern Recognition and Machine Learning. Springer, 2006.

[16] Erdem Bıyık, Dylan P Losey, Malayandi Palan, Nicholas C Landolfi, Gleb Shevchuk, and Dorsa Sadigh. Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences. preprint arXiv:2006.14091, 2020.

[17] Erdem Bıyık, Malayandi Palan, Nicholas C. Landolfi, Dylan P. Losey, and Dorsa Sadigh. Asking easy questions: A user-friendly approach to active reward learning. In Conf. on Robot Learning (CoRL), pages 1177–1190, 2019.

[18] Amy A Blank, James A French, Ali Utku Pehlivan, and Marcia K O'Malley. Current trends in robot-assisted upper-limb stroke rehabilitation: promoting patient engagement in therapy. Current Physical Medicine and Rehabilitation Reports, 2(3):184–195, 2014.

[19] Liad Blumrosen, Noam Nisan, and Ilya Segal. Auctions with severely bounded communication. Journal of Artificial Intelligence Research, 28:233–266, 2007.

[20] Diane Bouchacourt and Marco Baroni. Miss tools and mr fruit: Emergent communication in agents learning about object affordances. Annual Meeting of the Association for Computational Linguistics, 2019.

[21] Teige C Bourke, Catherine R Lowrey, Sean P Dukelow, Stephen D Bagg, Kathleen E Norman, and Stephen H Scott. A robot-based behavioural task to quantify impairments

in rapid motor decisions and actions after stroke. Journal of NeuroEngineering and Rehabilitation, 13(1):1–13, 2016.

[22] Stuart A Bowyer, Brian L Davies, and Ferdinando Rodriguez y Baena. Active constraints/virtual fixtures: A survey. IEEE Transactions on Robotics, 30(1):138–157, 2013.

[23] James Bradbury et al. Jax: composable transformations of python+ numpy programs. http://github.com/google/jax, 2020.

[24] Assaf Breska and Richard B Ivry. Context-specific control over the neural dynamics of temporal attention by the human cerebellum. Science Advances, 6(49), 2020.

[25] Alexander Broad, Todd Murphey, and Brenna Argall. Learning models for shared control of human-machine systems with unknown dynamics. Robotics: Science and Systems (RSS), 2018.

[26] Thurston L Brooks. Telerobotic response requirements. In Int. Conf. on Systems, Man, and Cybernetics, pages 113–120. IEEE, 1990.

[27] Tim Brys, Anna Harutyunyan, Halit Bener Suay, Sonia Chernova, Matthew E Taylor, and Ann Nowé. Reinforcement learning from demonstration through shaping. In Int. Joint Conf. on Artificial Intelligence, 2015.

[28] Marko Budišić, Ryan Mohr, and Igor Mezić. Applied Koopmanism. Chaos: An Interdisciplinary Journal of Nonlinear Science, 22(4):047510, 2012.

[29] Maya Cakmak, Crystal Chao, and Andrea L Thomaz. Designing interactions for robot active learners. Transactions on Autonomous Mental Development, 2(2):108–118, 2010.

[30] C Calautti, PS Jones, N Persaud, J-Y Guincestre, Marcello Naccarato, EA Warburton, and Jean-Claude Baron. Quantification of index tapping regularity after stroke with tri-axial accelerometry. Brain Research Bulletin, 70(1):1–7, 2006.

[31] Sylvain Calinon. A tutorial on task-parameterized movement learning and retrieval. Intelligent Service Robotics, 9(1):1–29, 2016.

[32] Bruce CV Campbell, Deidre A De Silva, Malcolm R Macleod, Shelagh B Coutts, Lee H Schwamm, Stephen M Davis, and Geoffrey A Donnan. Ischaemic stroke. Nature Reviews Disease Primers, 5(1):1–22, 2019.

[33] R. Campello, D. Moulavi, and J. Sander. Density-based clustering based on hierarchical density estimates. In Advances in Knowledge Discovery and Data Mining, pages 160–172. Springer, 2013.

[34] Colleen G Canning, Natalie E Allen, Evelien Nackaerts, Serene S Paul, Alice Nieuwboer, and Moran Gilat. Virtual reality in research and rehabilitation of gait and balance in Parkinson disease. Nature Reviews Neurology, 16(8):409–425, 2020.

[35] Kris Cao et al. Emergent communication through negotiation. Int. Conf. on Learning Representations, 2018.

[36] Anthony N Carlsen and Dana Maslovat. Startle and the startreact effect: physiological mechanisms. Journal of Clinical Neurophysiology, 36(6):452–459, 2019.

[37] Micah Carroll et al. On the utility of learning about humans for human-AI coordination. Advances in Neural Information Processing Systems, 2019.

[38] Rahma Chaabouni et al. Anti-efficient encoding in emergent communication. Advances in Neural Information Processing Systems, 2019.

[39] Rahma Chaabouni et al. Emergent communication at scale. Int. Conf. on Learning Representations, 2021.

[40] John Chae, Guang Yang, Byung Kyu Park, and Ihab Labatia. Delay in initiation and termination of muscle contraction, motor impairment, and physical disability in upper limb hemiparesis. Muscle & Nerve, 25(4):568–575, 2002.

[41] Henry G Chambers and David H Sutherland. A practical guide to gait analysis. Journal of the American Academy of Orthopaedic Surgeons (JAAOS), 10(3):222–231, 2002.

[42] Chunjie Chen, Duxin Liu, Xuesong Wang, Can Wang, and Xinyu Wu. An adaptive gait learning strategy for lower limb exoskeleton robot. In IEEE Int. Conf. on Real-Time Computing and Robotics (RCAR), pages 172–177, 2017.

[43] Edward Choi, Angeliki Lazaridou, and Nando De Freitas. Compositional obverter communication learning from raw visual input. Int. Conf. on Learning Representations, 2018.

[44] Amit K Chopra, Munindar P Singh, et al. An evaluation of communication protocol languages for engineering multiagent systems. Journal of Artificial Intelligence Research, 69:1351–1393, 2020.

[45] Supriyo Choudhury, A Shobhana, Ravi Singh, Dwaipayan Sen, Sidharth Shankar Anand, Shantanu Shubham, Mark R Baker, Hrishikesh Kumar, and Stuart N Baker. The relationship between enhanced reticulospinal outflow and upper limb function in chronic stroke patients. Neurorehabilitation and Neural Repair, 33(5):375–383, 2019.

[46] Jun-Uk Chu, Kang-Il Song, Sungmin Han, Soo Hyun Lee, Ji Yoon Kang, Dosik Hwang, Jun-Kyo Francis Suh, Kuiwon Choi, and Inchan Youn. Gait phase detection from sciatic nerve recordings in functional electrical stimulation systems for foot drop correction. Physiological Measurement, 2013.

[47] MC Cirstea and Mindy F Levin. Compensatory strategies for reaching in stroke. Brain, 123(5):940–953, 2000.

[48] Tyler Cluff and Stephen H Scott. Online corrections are faster because movement initiation must disengage postural control. Motor Control, 20(2):162–170, 2016.

[49] Adria Colome and Carme Torras. Dual reps: A generalization of relative entropy policy search exploiting bad experiences. Transactions on Robotics, 33(4):978–985, 2017.

[50] Steven C Cramer, Mriganka Sur, Bruce H Dobkin, Charles O'Brien, Terence D Sanger, John Q Trojanowski, Judith M Rumsey, Ramona Hicks, Judy Cameron, Daofen Chen, et al. Harnessing neuroplasticity for clinical applications. Brain, 134(6):1591–1609, 2011.

[51] Jacob W Crandall et al. Cooperating with machines. Nature Communications, 2018.

[52] Vincent P Crawford and Joel Sobel. Strategic information transmission. Econometrica: Journal of the Econometric Society, 1982.

[53] Tomas Darbutas, Vilma Juodžbalienė, Albertas Skurvydas, and Aleksandras Kriščiūnas. Dependence of reaction time and movement speed on task complexity and age. Medicina, 49(1):4, 2013.

[54] Neha Das, Sarah Bechtle, Todor Davchev, Dinesh Jayaraman, Akshara Rai, and Franziska Meier. Model-based inverse reinforcement learning from visual demonstrations. arXiv preprint arXiv:2010.09034, 2020.

[55] Abhishek Das et al. Tarmac: Targeted multi-agent communication. In Int. Conf. on Machine Learning (ICML), 2019.

[56] Harm De Vries et al. Talk the walk: Navigating New York City through grounded dialogue. arXiv preprint arXiv:1807.03367, 2018.

[57] Robert MJ Deacon and J Nicholas P Rawlins. T-maze alternation in the rodent. Nature Protocols, 2006.

[58] M. P. Deisenroth, D. Fox, and C. E. Rasmussen. Gaussian processes for data-efficient learning in robotics and control. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(2):408–423, 2015.

[59] Michel Denis et al. Spatial discourse and navigation: An analysis of route directions in the city of Venice. Applied Cognitive Psychology, 1999.

[60] Julius PA Dewald, Patrick S Pope, Joseph D Given, Thomas S Buchanan, and W Zev Rymer. Abnormal muscle coactivation patterns during isometric torque generation at the elbow and shoulder in hemiparetic subjects. Brain, 118(2):495–510, 1995.

[61] Luke Drnach, Irfan Essa, and Lena H Ting. Identifying gait phases from joint kinematics during walking with switched linear dynamical systems. In IEEE Int. Conf. on Biomedical Robotics and Biomechatronics (BioRob), 2018.

[62] Tom Eccles et al. Biases for emergent communication in multi-agent reinforcement learning. Advances in Neural Information Processing Systems, 2019.

[63] Sarah Elliott, Zhe Xu, and Maya Cakmak. Learning generalizable surface cleaning actions from demonstration. In Int. Sym. on Robot and Human Interactive Communication, pages 993–999. IEEE, 2017.

[64] Michael D Ellis, Anke I. R. Kottink, Gerdienke B. Prange, Johan S. Rietman, Jaap H. Buurke, and Julius P A Dewald. Quantifying loss of independent joint control in acute stroke with a robotic evaluation of reaching workspace. In Int. Conf. of the IEEE Eng. in Medicine and Biology Society, 2011.

[65] Michael D Ellis, Yiyun Lan, Jun Yao, and Julius Dewald. Robotic quantification of upper extremity loss of independent joint control or flexion synergy in individuals with hemiparetic stroke: a review of paradigms addressing the effects of shoulder abduction loading. Journal of NeuroEngineering and Rehabilitation, 13(1):1–11, 2016.

[66] Michael D Ellis, Theresa Sukal-Moulton, and Julius PA Dewald. Progressive shoulder abduction loading is a crucial element of arm rehabilitation in chronic stroke. Neurorehabilitation and Neural Repair, 23(8):862–869, 2009.

[67] Peter Englert, Ngo Anh Vien, and Marc Toussaint. Inverse kkt: Learning cost functions of manipulation tasks from demonstrations. Int. Journal on Robotics Research, 36(13-14):1474–1488, 2017.

[68] Marc O Ernst and Martin S Banks. Humans integrate visual and haptic information in a statistically optimal fashion. Nature, 415(6870):429–433, 2002.

[69] Katrina Evtimova et al. Emergent communication in a multi-modal, multi-step referential game. Int. Conf. on Learning Representations, 2018.

[70] Joseph Farrell and Matthew Rabin. Cheap talk. Journal of Economic Perspectives, 1996.

[71] Kathleen Fitzsimons, Ana Maria Acosta, Julius PA Dewald, and Todd D Murphey. Ergodicity reveals assistance and learning from physical human-robot interaction. Science Robotics, 4(29), 2019.

[72] Kathleen Fitzsimons, Aleksandra Kalinowska, Julius Dewald, and Todd D Murphey. Task-based hybrid shared control for training through forceful interaction. Int. Journal on Robotics Research, 39(9):1138–1154, 2020.

[73] Sharlene N Flesher, John E Downey, Jeffrey M Weiss, Christopher L Hughes, Angelica J Herrera, Elizabeth C Tyler-Kabara, Michael L Boninger, Jennifer L Collinger, and Robert A Gaunt. A brain-computer interface that evokes tactile sensations improves robotic arm control. Science, 372(6544):831–836, 2021.

[74] Jakob Foerster et al. Learning to communicate with deep multi-agent reinforcement learning. Advances in Neural Information Processing Systems, 2016.

[75] Roger Gassert and Volker Dietz. Rehabilitation robots for the treatment of sensorimotor deficits: a neurophysiological perspective. Journal of NeuroEngineering and Rehabilitation, 15(1):1–15, 2018.

[76] David J Gladstone, Cynthia J Danells, and Sandra E Black. The Fugl-Meyer Assessment of motor recovery after stroke: a critical review of its measurement properties. Neurorehabilitation and Neural Repair, 16(3):232–240, 2002.

[77] Carolyn Gowland, Paul Stratford, Maureen Ward, Julie Moreland, Wendy Torresin, Sandra Van Hullenaar, Julie Sanford, Susan Barreca, Bernadette Vanspall, and Nancy Plews. Measuring physical impairment and disability with the chedoke-mcmaster stroke assessment. Stroke, 24(1):58–63, 1993.

[78] Herbert P Grice. Logic and conversation. Speech Acts, 1975.

[79] Daniel H Grollman and Aude Billard. Donut as i do: Learning from failed demonstrations. In Int. Conf. on Robotics and Automation (ICRA), pages 3804–3809. IEEE, 2011.

[80] Daniel H Grollman and Aude G Billard. Robot learning from failed demonstrations. Int. Journal on Social Robotics, 4(4):331–342, 2012.

[81] Johannes Günther, Alex Kearney, Michael R Dawson, Craig Sherstan, and Patrick M Pilarski. Predictions, surprise, and predictions of surprise in general value function architectures. In AAAI 2018 Fall Symposium on Reasoning and Learning in Real-World Systems for Long-Term Autonomy, pages 22–29, 2018.

[82] Tom Hennigan et al. Haiku: Sonnet for JAX. http://github.com/deepmind/dm-haiku, 2020.

[83] T. A. Henzinger. The theory of hybrid automata. In Proceedings of the IEEE Symposium on Logic in Computer Science, pages 278–283, 1996.

[84] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In Advances in Neural Information Processing Systems, pages 4565–4573. Curran Associates, Inc., 2016.

[85] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural Computation, 1997.

[86] Claire Fletcher Honeycutt and Eric Jon Perreault. Planning of ballistic movement following stroke: insights from the startle reflex. PloS ONE, 7(8):e43097, 2012.

[87] Auke Jan Ijspeert, Jun Nakanishi, Heiko Hoffmann, Peter Pastor, and Stefan Schaal. Dynamical movement primitives: learning attractor models for motor behaviors. Neural Computation, 25(2):328–373, 2013.

[88] Natasha Jaques et al. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. Int. Conf. on Machine Learning, 2019.

[89] Noémie Jaquier, David Ginsbourger, and Sylvain Calinon. Learning from demonstration with model-based gaussian process. In Conf. on Robot Learning (CoRL), pages 247–257, 2020.

[90] Ivana Kajić, Eser Aygün, and Doina Precup. Learning to cooperate: Emergent communication in multi-agent navigation. Meeting of the Cognitive Science Society, 2020.

[91] Aleksandra Kalinowska. Shared control paradigms for robot-aided task training and robotic assistance. Master's Thesis, Northwestern University, 2019.

[92] Aleksandra Kalinowska, Thomas A Berrueta, Adam Zoss, and Todd Murphey. Data-driven gait segmentation for walking assistance in a lower-limb assistive device. Int. Conf. on Robotics and Automation, 2019.

[93] Aleksandra Kalinowska, Elnaz Davoodi, Kory W Mathewson, Todd Murphey, and Patrick M Pilarski. Towards situated communication in multi-step interactions: Time is a key pressure in communication emergence. Meeting of the Cognitive Science Society, 44(44), 2022.

[94] Aleksandra Kalinowska, Elnaz Davoodi, Florian Strub, Kory W Mathewson, Ivana Kajic, Michael Bowling, Todd Murphey, and Patrick M Pilarski. Over-communicate no more: Situated rl agents learn concise communication protocols. under review at Journal of Artificial Intelligence Research, 2023.

[95] Aleksandra Kalinowska, Kathleen Fitzsimons, Julius Dewald, and Todd D Murphey. Online user assessment for minimal intervention during task-based robotic assistance. In Robotics: Science and Systems, 2018.

[96] Aleksandra Kalinowska, Patrick Pilarski, and Todd Murphey. Embodied communication: How humans and robots communicate through physical interaction. Annual Reviews of Control, Robotics, and Autonomous Systems, 2023.

[97] Aleksandra Kalinowska, Ahalya Prabhakar, Kathleen Fitzsimons, and Todd Murphey. Ergodic imitation: Learning from what to do and what not to do. Int. Conf. on Robotics and Automation, 2021.

[98] Aleksandra Kalinowska, Kyra Rudy, Millicent Schlafly, Kathleen Fitzsimons, Julius Dewald, and Todd Murphey. Shoulder abduction loading affects motor coordination in individuals with chronic stroke, informing targeted rehabilitation. In Int. Conf. for Biomedical Robotics and Biomechatronics, 2020.

[99] Aleksandra Kalinowska, Kyra Rudy, Millicent Schlafly, Kathleen Fitzsimons, Julius P Dewald, and Todd D Murphey. Ball-in-bowl task. https://github.com/MurpheyLab/DynamicTasks/, 2022.

[100] Aleksandra Kalinowska, Millicent Schlafly, Kyra Rudy, Julius Dewald, and Todd David Murphey. Quantifying functional loss in neuromotor disorders using assessment of motion in dynamic tasks. in preparation, 2023.

[101] Aleksandra Kalinowska, Millicent Schlafly, Kyra Rudy, Julius PA Dewald, and Todd D Murphey. Measuring interaction bandwidth during physical human-robot collaboration. Robotics and Autonomation Letters, 2022.

[102] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. Int. Conf. on Learning Representations, 2015.

[103] Simon Kirby and James R Hurford. The emergence of linguistic structure: An overview of the iterated learning model. Simulating the Evolution of Language, 2002.

[104] Paul Knott et al. Evaluating the robustness of collaborative agents. Advances in Neural Information Processing Systems, 2021.

[105] Abi Komanduru and Jean Honorio. On the correctness and sample complexity of inverse reinforcement learning. In Advances in Neural Information Processing Systems, pages 7112–7121, 2019.

[106] B.O. Koopman. Hamiltonian systems and transformation in Hilbert space. Proceedings of the National Academy of Sciences, 17(5):315–318, 1931.

[107] Robert J Kosinski. A literature review on reaction time. Clemson University, 10(1):337–344, 2008.

[108] Satwik Kottur et al. Natural language does not emerge 'naturally' in multi-agent dialog. Conf. on Empirical Methods in NLP, 2017.

[109] John W Krakauer and David J Reinkensmeyer. Epilogue: Robots for neurorehabilitation—the debate. In Neurorehabilitation Technology, pages 757–764. Springer, 2022.

[110] Oliver Kroemer and Gaurav S Sukhatme. Feature selection for learning versatile manipulation skills based on observed and desired trajectories. In Int. Conf. on Robotics and Automation (ICRA), pages 4713–4720. IEEE, 2017.

[111] T. Kulak, J. Silvério, and S. Calinon. Fourier movement primitives: an approach for learning rhythmic robot skills from demonstrations. In Robotics: Science and Systems (RSS), 2020.

[112] Jean-Charles Lamy, Isabelle Wargon, Dominique Mazevet, Zaïd Ghanim, Pascale Pradat-Diehl, and Rose Katz. Impaired efficacy of spinal presynaptic mechanisms in spastic stroke patients. Brain, 132(3):734–748, 2009.

[113] Angeliki Lazaridou and Marco Baroni. Emergent multi-agent communication in the deep learning era. arXiv preprint arXiv:2006.02419, 2020.

[114] Angeliki Lazaridou et al. Emergence of linguistic communication from referential games with symbolic and pixel input. Int. Conf. on Learning Representations, 2018.

[115] Pi-Hsia Lee, Ting-Ting Yeh, Hsin-Yen Yen, Wan-Ling Hsu, Valeria Jia-Yi Chiu, and Shu-Chun Lee. Impacts of stroke and cognitive impairment on activities of daily living in the taiwan longitudinal study on aging. Scientific Reports, 11(1):1–9, 2021.

[116] Yong Min Lee, Seunghwan Lee, Kyeong Eun Uhm, Gregorij Kurillo, Jay J Han, and Jongmin Lee. Upper limb three-dimensional reachable workspace analysis using the kinect sensor in hemiplegic stroke patients: A cross-sectional observational study. American Journal of Physical Medicine & Rehabilitation, 99(5):397–403, 2020.

[117] Sergey Levine, Zoran Popovic, and Vladlen Koltun. Nonlinear inverse reinforcement learning with gaussian processes. In Advances in Neural Information Processing Systems, pages 19–27, 2011.

[118] David Lewis. Convention. Harvard University Press, 1969.

[119] Fushan Li and Michael Bowling. Ease-of-teaching and language structure from emergent communication. Advances in Neural Information Processing Systems, 2019.

[120] Ryan Lowe et al. On the pitfalls of measuring emergent communication. Int. Conf. on Autonomous Agents and Multiagent Systems, 2019.

[121] Catherine R Lowrey, Teige C Bourke, Stephen D Bagg, Sean P Dukelow, and Stephen H Scott. A postural unloading task to assess fast corrective responses in the upper limb following stroke. Journal of NeuroEngineering and Rehabilitation, 16(1):1–17, 2019.

[122] Paul R MacNeilage and Stefan Glasauer. Quantification of head movement predictability and implications for suppression of vestibular input during locomotion. Frontiers in Computational Neuroscience, 11:47, 2017.

[123] Stuart Maitland and Stuart N Baker. Ipsilateral motor evoked potentials as a measure of the reticulospinal tract in age-related strength changes. Frontiers in Aging Neuroscience, 13:92, 2021.

[124] Martina Mancini, Vrutangkumar V Shah, Samuel Stuart, Carolin Curtze, Fay B Horak, Delaram Safarpour, and John G Nutt. Measuring freezing of gait during daily-life: an open-source, wearable sensors approach. Journal of NeuroEngineering and Rehabilitation, 18(1):1–13, 2021.

[125] Mario Manto, James M Bower, Adriana Bastos Conforto, José M Delgado-García, Suzete Nascimento Farias Da Guarda, Marcus Gerwig, Christophe Habas, Nobuhiro Hagura, Richard B Ivry, Peter Mariën, et al. Consensus paper: roles of the cerebellum in motor control—the diversity of ideas on cerebellar involvement in movement. The Cerebellum, 11(2):457–487, 2012.

[126] George Mathew and Igor Mezić. Metrics for ergodicity and design of ergodic dynamics for multi-agent systems. Physica D: Nonlinear Phenomena, 240(4):432–442, 2011.

[127] Anastasia Mavrommati, Emmanouil Tzorakoleftherakis, Ian Abraham, and Todd D Murphey. Real-time area coverage and target localization using receding-horizon ergodic exploration. Transactions on Robotics, 34(1):62–80, 2018.

[128] Hans C Mayer and Rouslan Krechetnikov. Walking with coffee: Why does it spill? Physical Review E, 85(4):046117, 2012.

[129] Jacob G McPherson, Albert Chen, Michael D Ellis, Jun Yao, CJ Heckman, and Julius PA Dewald. Progressive recruitment of contralesional cortico-reticulospinal pathways drives motor impairment post stroke. The Journal of Physiology, 596(7):1211–1225, 2018.

[130] José del R Millán, Rüdiger Rupp, Gernot Mueller-Putz, Roderick Murray-Smith, Claudio Giugliemma, Michael Tangermann, Carmen Vidaurre, Febo Cincotti, Andrea Kubler, Robert Leeb, et al. Combining brain–computer interfaces and assistive technologies: state-of-the-art and challenges. Frontiers in Neuroscience, 4:161, 2010.

[131] Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. AAAI Conf. on Artificial Intelligence, 2018.

[132] Carol J Mottram, Charles J Heckman, Randall K Powers, William Zev Rymer, and Nina L Suresh. Disturbances of motor unit rate modulation are prevalent in muscles of spastic-paretic stroke survivors. Journal of Neurophysiology, 111(10):2017–2028, 2014.

[133] Nafee Mourad, Ali Ezzeddine, Babak Nadjar Araabi, and Majid Nili Ahmadabadi. Learning from demonstrations and human evaluative feedbacks: Handling sparsity and imperfection using inverse reinforcement learning approach. Journal on Robotics, 2020, 2020.

[134] LM Nashner and PJ Cordo. Relation of automatic postural responses and reaction-time voluntary movements of human leg muscles. Experimental Brain Research, 43(3):395–405, 1981.

[135] Rashida Nayeem, Won Joon Sohn, Julie A DiCarlo, Perman Gochyyev, David J Lin, and Dagmar Sternad. Novel platform for quantitative assessment of functional object interactions after stroke. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2022.

[136] Duy Nguyen-Tuong and Jan Peters. Model learning for robot control: a survey. Cognitive Processing, 2011.

[137] Scott Niekum, Sarah Osentoski, George Konidaris, and Andrew G Barto. Learning and generalization of complex tasks from unstructured demonstrations. In Int. Conf. on Intelligent Robots and Systems (IROS), pages 5239–5246, 2012.

[138] Michael Noukhovitch et al. Emergent communication under competition. Int. Conf. on Autonomous Agents and Multiagent Systems, 2021.

[139] Józef Opara, Andrzej Małecki, Elżbieta Małecka, and Teresa Socha. Motor assessment in parkinsons disease. Annals of Agricultural and Environmental Medicine, 24(3):411–415, 2017.

[140] Özhan Özen, Karin A Buetler, and Laura Marchal-Crespo. Towards functional robotic training: motor learning of dynamic tasks is enhanced by haptic rendering but hampered by arm weight support. Journal of NeuroEngineering and Rehabilitation, 19(1):1–18, 2022.

[141] Claudio Pacchierotti, Leonardo Meli, Francesco Chinello, Monica Malvezzi, and Domenico Prattichizzo. Cutaneous haptic feedback to ensure the stability of robotic teleoperation systems. Int. Journal of Robotics Research, 34(14), 2015.

[142] Lucia Pais, Keisuke Umezawa, Yoshihiko Nakamura, and Aude Billard. Learning robot skills through motion segmentation and constraints extraction. In HRI Workshop on Collaborative Manipulation, 2013.

[143] Malayandi Palan, Nicholas C Landolfi, Gleb Shevchuk, and Dorsa Sadigh. Learning reward functions by integrating human demonstrations and preferences. In Robotics: Science and Systems (RSS), 2019.

[144] Y. Pan and J. Wang. Model predictive control of unknown nonlinear dynamical systems based on recurrent neural networks. IEEE Transactions on Industrial Electronics, 59(8):3089–3101, 2012.

[145] Alexandros Paraschos, Christian Daniel, Jan R Peters, and Gerhard Neumann. Probabilistic movement primitives. In Advances in Neural Information Processing Systems, pages 2616–2624, 2013.

[146] Joel S Perlmutter. Assessment of Parkinson disease manifestations. Current Protocols in Neuroscience, 49(1):10–1, 2009.

[147] David Pinzon, Simon Byrns, and Bin Zheng. Prevailing trends in haptic feedback simulation for minimally invasive surgery. Surgical Innovation, 23(4):415–421, 2016.

[148] Alex Pollock, Sybil E Farmer, Marian C Brady, Peter Langhorne, Gillian E Mead, Jan Mehrholz, and Frederike van Wijck. Interventions for improving upper limb function after stroke. Cochrane Database of Systematic Reviews, 2014.

[149] Tommaso Proietti, Vincent Crocher, Agnes Roby-Brami, and Nathanael Jarrasse. Upper-limb robotic exoskeletons for neurorehabilitation: a review on control strategies. IEEE Reviews in Biomedical Engineering, 9:4–14, 2016.

[150] R Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, 2020.

[151] David J Reinkensmeyer, Eric T Wolbrecht, Vicky Chan, Cathy Chou, Steven C Cramer, and James E Bobrow. Comparison of 3D, assist-as-needed robotic arm/hand movement training provided with Pneu-WREX to conventional table top therapy following chronic stroke. American Journal of Physical medicine & Rehabilitation, 91(11 0 3):S232, 2012.

[152] Martin Riedmiller. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. European Conf. on Machine Learning, 2005.

[153] Mathieu Rita, Rahma Chaabouni, and Emmanuel Dupoux. "LazImpa": Lazy and impatient neural agents learn to communicate efficiently. Conf. on Computational Natural Language Learning, 2020.

[154] Mathieu Rita et al. On the role of population heterogeneity in emergent communication. Int. Conf. on Learning Representations, 2022.

[155] Brandon Rohrer, Susan Fasoli, Hermano Igo Krebs, Richard Hughes, Bruce Volpe, Walter R Frontera, Joel Stein, and Neville Hogan. Movement smoothness changes during stroke recovery. Journal of Neuroscience, 22:8297–8304, 2002.

[156] Dorsa Sadigh, Anca D Dragan, Shankar Sastry, and Sanjit A Seshia. Active preference-based learning of reward functions. In Robotics: Science and Systems (RSS), 2017.

[157] Mique Saes, Mohamed Irfan Mohamed Refai, Bert-Jan F van Beijnum, JBJ Bussmann, Elizabeth P Jansma, Peter H Veltink, Jaap Hilbert Buurke, Erwin EH van Wegen, Carel GM Meskers, JW Krakauer, et al. Quantifying quality of reaching movements longitudinally post-stroke: A systematic review. Neurorehabilitation and Neural Repair, 36(3):183–207, 2022.

[158] Sydney Y Schaefer, Kathleen Y Haaland, and Robert L Sainburg. Ipsilesional motor deficits following stroke reflect hemispheric specializations for movement control. Brain, 130(8):2146–2158, 2007.

[159] Markus Schneider and Wolfgang Ertel. Robot learning by demonstration with local gaussian process regression. In Int. Conf. on Intelligent Robots and Systems (IROS), pages 255–260. IEEE, 2010.

[160] Robert Schulz, Benedikt M Frey, Philipp Koch, Maximo Zimerman, Marlene Bönstrup, Jan Feldheim, Jan E Timmermann, Gerhard Schön, Bastian Cheng, Götz Thomalla, et al. Cortico-cerebellar structural connectivity is related to residual motor output in chronic stroke. Cerebral Cortex, 27(1):635–645, 2017.

[161] Nicolas Schweighofer, Younggeun Choi, Carolee Winstein, and James Gordon. Task-oriented rehabilitation robotics. American Journal of Physical Medicine & Rehabilitation, 91(11):S270–S279, 2012.

[162] Sherry E Scott. Different perspectives and formulas for capturing deviation from ergodicity. Journal on Applied Dynamical Systems, 12(4):1948–1967, 2013.

[163] Sherry E Scott, Thomas C Redd, Leonid Kuznetsov, Igor Mezić, and Christopher KRT Jones. Capturing deviation from ergodicity at different scales. Physica D: Nonlinear Phenomena, 238(16):1668–1679, 2009.

[164] Stephen H Scott and Sean P Dukelow. Potential of robots as next-generation technology for clinical assessment of neurological disorders and upper-limb therapy. Journal of Rehabilitation Research & Development, 48(4), 2011.

[165] Mario Selvaggio, Firas Abi-Farraj, Claudio Pacchierotti, Paolo Robuffo Giordano, and Bruno Siciliano. Haptic-based shared-control methods for a dual-arm system. Robotics and Automation Letters, 3(4):4249–4256, 2018.

[166] Mario Selvaggio, Jonathan Cacace, Claudio Pacchierotti, Fabio Ruggiero, and Paolo Robuffo Giordano. A shared-control teleoperation architecture for nonprehensile object transportation. Transactions on Robotics, 38(1), 2021.

[167] Mario Selvaggio, Marco Cognetti, Stefanos Nikolaidis, Serena Ivaldi, and Bruno Siciliano. Autonomy in physical human-robot interaction: A brief survey. Robotics and Automation Letters, 2021.

[168] Na Jin Seo, William Z Rymer, and Derek G Kamper. Delays in grip initiation and termination in persons with stroke: effects of arm support and active muscle stretch exercise. Journal of Neurophysiology, 101(6):3108–3115, 2009.

[169] Kyriacos Shiarlis, Joao Messias, and SA Whiteson. Inverse reinforcement learning from failure. In Int. Conf. on Autonomous Agents and Multiagent Systems, 2016.

[170] Yoav Shoham and Kevin Leyton-Brown. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, 2008.

[171] H Sigmundsson. Disorders of motor development (clumsy child syndrome). Neurodevelopmental Disorders, pages 51–68, 2005.

[172] Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. Int. Conf. on Learning Representations (ICLR), 2019.

[173] Won Joon Sohn, Rifat Sipahi, Terence D Sanger, and Dagmar Sternad. Portable motion-analysis device for upper-limb research, assessment, and rehabilitation in non-laboratory settings. Journal of Translational Engineering in Health and Medicine, 7:1–14, 2019.

[174] Jonathan Spencer, Sanjiban Choudhury, Matt Barnes, Matthew Schmitte, Mung Chiang, Peter Ramadge Ramadge, and Siddhartha Srinivasa. Learning from interventions: Human-robot interaction as both explicit and implicit feedback. In Robotics: Science and Systems (RSS), 2020.

[175] Luc Steels. Evolving grounded communication for robots. Trends in Cognitive Sciences, 2003.

[176] Luc Steels and Rodney Brooks. The Artificial Life Route to Artificial Intelligence: Building Embodied, Situated Agents. Routledge, 1995.

[177] Dagmar Sternad. From theoretical analysis to clinical assessment and intervention: Three interactive motor skills in a virtual environment. Int. Conf. on Virtual Rehabilitation (ICVR), pages 265–272, 2015.

[178] Theresa M Sukal, Michael D Ellis, and Julius PA Dewald. Shoulder abduction-induced reductions in reaching work area following hemiparetic stroke: neuroscientific implications. Experimental Brain Research, 183(2):215–223, 2007.

[179] Juri Taborri, Stefano Rossi, Eduardo Palermo, Fabrizio Patanè, and Paolo Cappa. A novel HMM distributed classifier for the detection of gait phases by means of a wearable inertial sensor network. Sensors, 14(9):16212–16234, 2014.

[180] Craig D Takahashi and David J Reinkensmeyer. Hemiparetic stroke impairs anticipatory control of arm movement. Experimental Brain Research, 149(2):131–140, 2003.

[181] Michael Thomson, Kennedy Murphy, and Ryan Lukeman. Groups clapping in unison undergo size-dependent error-induced frequency increase. Scientific Reports, 8(1):1–9, 2018.

[182] Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral cloning from observation. In Int. Joint Conf. on Artificial Intelligence, 2018.

[183] Kathrin Tyryshkin, Angela M Coderre, Janice I Glasgow, Troy M Herter, Stephen D Bagg, Sean P Dukelow, and Stephen H Scott. A robotic object hitting task to quantify sensorimotor impairments in participants with stroke. Journal of NeuroEngineering and Rehabilitation, 11(1):1–12, 2014.

[184] Emmanouil Tzorakoleftherakis, Todd D Murphey, and Robert A Scheidt. Augmenting sensorimotor control using "goal-aware" vibrotactile stimulation during reaching and manipulation behaviors. Experimental Brain Research, 234(8):2403–2414, 2016.

[185] J Valls-Solé, A Solé, F Valldeoriola, E Muñoz, LE Gonzalez, and ES Tolosa. Reaction time and acoustic startle in normal human subjects. Neuroscience Letters, 195(2):97–100, 1995.

[186] Olivier AJ Van der Meijden and Marlies P Schijven. The value of haptic feedback in conventional and robot-assisted minimal invasive surgery and virtual reality training: a current review. Surgical endoscopy, 23(6):1180–1190, 2009.

[187] Elena Vergaro, Maura Casadio, Valentina Squeri, Psiche Giannoni, Pietro Morasso, and Vittorio Sanguineti. Self-adaptive robot training of stroke survivors for continuous tracking movements. Journal of NeuroEngineering and Rehabilitation, 7(1):13, 2010.

[188] Huong Thi Thu Vu, Felipe Gomez, Pierre Cherelle, Dirk Lefeber, Ann Nowé, and Bram Vanderborght. ED-FNN: A new deep learning algorithm to detect percentage of the gait cycle for powered prostheses. Sensors, 18(7):2389, 2018.

[189] Kyle Wagner et al. Progress in the simulation of emergent communication and language. Adaptive Behavior, 2003.

[190] D Wang. Reticular formation and spinal cord injury. Spinal Cord, 47(3):204–212, 2009.

[191] Kevin B Wilkins, Julius PA Dewald, and Jun Yao. Intervention-induced changes in neural connectivity during motor preparation may affect cortical activity at motor execution. Scientific Reports, 10(1):1–13, 2020.

[192] Grady Williams, Nolan Wagener, Brian Goldfain, Paul Drews, James M Rehg, Byron Boots, and Evangelos A Theodorou. Information theoretic MPC for model-based reinforcement learning. In Int. Conf. on Robotics and Automation (ICRA), 2017.

[193] M.O. Williams, I.G Kevrekidis, and C.W. Rowley. A data–driven approximation of the Koopman operator: Extending dynamic mode decomposition. Journal of Nonlinear Science, 25:1307–1346, 2015.

[194] Bodo Winter. Linear models and linear mixed effects models in r with linguistic applications. arXiv preprint arXiv:1308.5499, 2013.

[195] Daniel M Wolpert and J Randall Flanagan. Q&a: Robotics as a tool to understand the brain. BMC biology, 8(1):1–4, 2010.

[196] Yueh-Hua Wu, Nontawat Charoenphakdee, Han Bao, Voot Tangkaratt, and Masashi Sugiyama. Imitation learning from imperfect demonstration. arXiv preprint arXiv:1901.09387, 2019.

[197] Chao Yang, Xiaojian Ma, Wenbing Huang, Fuchun Sun, Huaping Liu, Junzhou Huang, and Chuang Gan. Imitation learning from observations by minimizing inverse dynamics disagreement. In Advances in Neural Information Processing Systems, pages 239–249, 2019.

[198] Sai Qian Zhang, Qi Zhang, and Jieyu Lin. Efficient communication in multi-agent reinforcement learning via variance based control. Advances in Neural Information Processing Systems, 2019.

[199] Bin Zheng and Christine L MacKenzie. Kinematics of reaching and grasping with a tool. In Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 2007.