Technical University of Munich
Department of Electrical and Computer Engineering
Chair of Automatic Control Engineering

Natural Human-Robot Interaction in Close Proximity: A Human-in-the-Center Approach

Özgür S. Öğüz

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzende/-r: Prof. Dr.-Ing. Hans-Georg Herzog

Prüfende/-r der Dissertation:

1. Prof. Dr.-Ing. habil. Dirk Wollherr
2. Prof. Abderrahmane Kheddar, Ph.D.

Die Dissertation wurde am 04.07.2018 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 04.10.2018 angenommen.

Acknowledgments

This thesis summarizes my research conducted at the Chair of Automatic Control Engineering, LSR, of the Technical University of Munich, Germany. I'm grateful that I have experienced all the facets of an academic environment, including independent research, student supervision, proposal writing, project management, and more, thanks to the responsibilities given by the Chair.

First and foremost, I would like to warmly thank my advisor Dirk Wollherr for his advice and encouragement during my time at TUM. Besides, I want to thank Martin Buss for creating this resourceful research environment.

I would also like to thank my collaborator Stefan Glasauer for his support and advice. Many thanks to Marion Leibold for always being available to provide her help. I would like to acknowledge all of the team members in LSR for providing a fun and friendly environment to work. Special thanks to Siemens AG for partially funding this research.

Many of the experiments and analysis presented in this thesis were made possible by the help of the students I supervised. I'm grateful for working with such bright and hard-working young minds.

Most importantly, I would like to thank my family, and especially my wife Gökçe, for being patient, offering support, and having fun together at all times which made this journey a breeze.

Abstract

As robots become ubiquitous in our daily lives, they are expected to interact in close proximity to humans naturally and safely. Humans interact with each other in a confined workspace effectively by adapting to the movement of their partner. Similarly, robot motion has to be agile by predicting the movement of the dyadic partner. However, due to the intra- and inter-personal motion variations, predicting human movement behavior and reproducing a compatible dyadic interaction skill for the robot are nontrivial tasks. Generating reactive motion planners for the robotic agent alone would not be able to provide the expected interaction dynamics between partners. A feasible methodology for constructing such versatile interaction skills is to follow a multifaceted approach; first by focusing on human motion models, both in isolation and during dyadic cooperation, and second, by developing robot interaction skills constrained on natural dyadic movement behaviors. In this thesis, I discuss several model-based and data-driven approaches to model and predict human movement behaviors. I also propose a learning framework, a stochastic trajectory optimization formulation, and a policy improvement algorithm for robot motion generation during close proximity dyadic interaction. The human-robot interaction (HRI) experiments and the simulation results show how human movement models can be used to acquire agile and naturalistic robot interaction skills. In essence, this thesis signifies that for providing natural, safe, and effective close proximity HRI by autonomous agents, human-in-the-loop learning and control approaches have to be built by combining model-based and data-driven methods.

Zusammenfassung

Je allgegenwärtiger Roboter in dem täglichen Leben werden, umso mehr steigt der Bedarf nach sicherem und natürlichem Interaktionsverhalten in unmittelbarer Nähe des Menschen. Menschen interagieren auf engstem Raum effektiv miteinander, indem sie sich an die Bewegungen ihres Partners anpassen. In ähnlicher Weise müssen Roboterbewegungen agil sein, durch schnelles vorhersagen der Bewegungen des dyadischen Partners. Aufgrund von intra- und inter-personalen Bewegungsvariationen sind jedoch das Vorhersagen des menschlichen Bewegungsverhaltens und das Reproduzieren einer kompatiblen dyadischen Interaktionsfertigkeit durch den Roboter nicht-triviale Aufgaben. Das einseitige anwenden von reaktiven Bewegungsplanern durch den Roboter ist nicht ausreichend um die erwartete Interaktionsdynamik zwischen den Partnern bereitzustellen. Daher ist ein vielschichtiger Ansatz eine praktikable Methode zum Aufbau solch vielseitiger Interaktionsfertigkeiten; erstens durch die Fokussierung auf menschliche Bewegungsmodelle, jeweils isoliert und während der dyadischen Kooperation, und zweitens durch die Entwicklung von Roboterinteraktionsfertigkeiten, die an ein natürliches dyadisches Bewegungsverhalten gebunden sind. In dieser Dissertation werden mehrere modellbasierte und datengesteuerte Ansätze zur Modellierung und Vorhersage des menschlichen Bewegungsverhaltens diskutiert. Darüber hinaus werden ein Lern-Framework, eine stochastische Trajektorien Optimierung Formulierung, und ein Policy Improvement Algorithmus als Ansätze für die Erzeugung von Roboterbewegungen während der dyadischen Interaktion auf engstem Raum vorgeschlagen. Die Mensch-Roboter-Interaktion Experimente und Simulationsergebnisse zeigen wie menschliche Bewegungsmodelle verwendet werden können, um agile und naturalistische Roboterinteraktionsfertigkeiten zu erlangen. Zusammengefasst zeigt diese These, dass um autonomen Agenten natürliche, sichere und effektive Mensch-Roboter-Interaktion auf engstem Raum zu ermöglichen, human-in-the-loop Lern- und Steuerungsansätze basierend auf einer Kombination aus modellbasierten und datengesteuerten Methoden nötig sind.

Contents

II Modeling Human-Human Interaction Behavior

List of Figures

List of Tables

Notation

Acronyms and Abbreviations

HRI	Human-robot interaction
HRC	Human-robot collaboration
HHI	Human-human interaction
CNS	Central nervous system
OCP	Optimal control problem
EPH	Equilibrium point hypothesis
NLP	Nonlinear program
ULP	Upper level program
LLP	Lower level program
MIM	Multiple internal model
MIIM	Multiple inverse internal model
IOC	Inverse optimal control
DTW	Dynamic time warping
RL	Reinforcement learning
IRL	Inverse reinforcement learning
PCA	Principal component analysis
RNN	Recurrent neural network
DRL	Deep reinforcement learning
MP	Movement primitive
DMP	Dynamic movement primitive
ProMP	Probabilistic movement primitive

PTOMP	Progressive trajectory optimization-based motion planning
GPR	Gaussian process regression
PIBBO	Policy improvement through black-box optimization
AI	Artificial intelligence
LfD	Learning from demonstration
PbD	Programming by demonstration
SMP	Social movement primitive
AS^{2}M	Associative social skill memory

1 Introduction

"Will robots inherit the earth? Yes, but they will be our children."

— Marvin A. Minsky

Recently, interest in endowing robots and autonomous agents with complex, responsive dyadic interaction skills has been increasing for both manufacturing and household settings. However, despite remarkable progress in optimal control, optimization, motion planning, and reinforcement learning, neither robots nor autonomous agents possess behavioral capabilities that can compare to the naturalness, flexibility, and responsiveness of human interaction skills. Humans collaborate with each other naturally, and efficiently in close proximity without causing safety risks by being attentive to each other. In that regard, the analysis of human motion behaviors might provide the models for enabling the robots and autonomous systems to acquire the essential skills to become effective dyadic partners.

The high-level goal of this dissertation research is to construct human-in-the-loop motion planning and control frameworks for robotic agents. Two mainstream approaches, model-based and data-driven, have been followed simultaneously to complement each other for achieving intuitive close proximity human-robot interaction (HRI). Model-based methods are *descriptive*, and thus enable methodical analysis of systems and computation of optimal behaviors, e.g. for human motor control models (Ch. 2) and dyadic interaction behavior (Ch. 4), as well as planning of optimal robot motion (Ch. 6). On the other hand, data-driven approaches are *generative*, and in general computationally more efficient. They facilitate online-capable algorithms and learning from demonstrations of behavioral features which may not be captured solely by model-based methods, e.g. learning and predicting online person dependent movement variations (Ch. 3) and interaction behaviors (Ch. 5), as well as adapting to human preferences during interaction (Ch. 7). In this thesis, I discuss how such multifaceted approaches have been used together for three subproblems: (*H*) human motion understanding, (*HH*) human-human dyadic interaction modeling, and (*HR*) safe and effective close proximity human-robot interaction.

Human sensorimotor control has been observed to achieve near-optimal performance for stereotypical movements [1] and well-practiced skills [2]. In general, these optimality conditions have been used to identify relevant optimal control models to describe human movement in free-space and mostly for 2D planar motions [3]–[5]. This leads to the questions of whether there exist similar principles that can allow us to model and explain (*i*) human arm motions in 3D, and (*ii*) human dyadic interaction behaviors in a generalized way. If such optimal control models can be found and transferred to autonomous systems, the robotic agent can also follow similar strategies to interact with humans.

Unfortunately, the modeling of human movement behavior and the construction of effective and generalizable control policies, such as the movement control observed in dyadic human interaction, pose significant challenges. For explaining the remarkable ability of central nervous system (CNS) to learn, adapt and control motor skills, the multiple internal models (MIMs) hypothesis has been formed as a unified architecture.

Nevertheless, a model-based computational framework that connects optimality principles with the MIMs hypothesis is missing. For the analysis and modeling of human motion control, a multitude of optimal control models has been proposed, each corresponding to a specific task. However, a single generalized optimal control model that can predict human motions for different tasks is desired. The inverse optimal control (IOC), or inverse reinforcement learning (IRL), techniques can discover cost functions used in such an optimal control problem (OCP) to model and predict human movement behavior for a broader range of motions.

For dyadic interaction, there have been studies focusing solely on the safety aspect, while the naturalness and readability of movement has been overlooked [6]. On the other hand, generating *legible* and *predictable* robot motions has been investigated [7], [8], and isolated human motion behaviors have been modeled and transferred to robots [9]. However, close proximity interaction behaviors have been neglected. In particular, a comprehensive analysis on human motion models, both in isolation and during dyadic interaction, and a unified framework that can integrate both physical and psychological constraints into robot motion generation are two crucial missing components. Similar to analyzing human movement in isolation by IOC, dyadic interaction behavior control can also be modeled by IRL as composite cost models.

The cost functions found can be integrated into trajectory optimization methods to imitate human movements, while the motion predictions of humans using those models become additional constraints for the optimization to provide safe interaction for robot motion generation during close proximity. In essence, rather than focusing on task specific human motion models and tackling only a single aspect of dyadic interaction, I propose a comprehensive approach in this thesis, including optimal control, trajectory optimization, movement primitives and reinforcement learning to realize effective HRI in close proximity.

In this thesis, I discuss methodologies for ($M1$) identifying optimal control models for human motion in free space and also during dyadic interaction, and ($M2$) optimization of robot motion planning in HRI, along with ($M3$) learning interaction behavior skills automatically from example demonstrations as well as by trial and error. For human motion modeling ($M1$), single human and human-human interaction (HHI) movement data were recorded, and optimal control models were computed by IOC and IRL, respectively. In regard to motion planning ($M2$), an online stochastic trajectory optimization was developed for close proximity HRI. For interaction skill acquisition ($M3$), HHI data was used to train a neural network end-to-end to imitate humans, and a policy improvement algorithm was implemented to update the robot policy with respect to the human preferences during interaction.

Considering the three subproblems mentioned earlier, the key contributions of this thesis are:

- (H) Using an inverse optimal control approach, composite models are identified for human motor control that supports the MIMs hypothesis.
- (H) A hybrid framework, combining model-based and data-driven methods, is proposed that predicts human motions and computes control models simultaneously to account for intra- and inter-personal movement variations.
- (HH&HR) An ontology for HHI scenarios is constructed, and control policies for dyadic interaction are learned from HHI demonstrations.
- (HR) An online capable trajectory optimization algorithm that provides safe and effective close proximity HRI is constructed.
- (HR) A human guided policy improvement algorithm is developed for generating readable, yet effective robot motion trajectories.

The central themes of this thesis are introduced in the following sections. First, I explain how MIMs are discovered by using IOC for understanding and modeling human motor control, followed by the effective motion prediction framework we constructed by using movement primitives (MPs). Then, I present a novel ontology for the classification and analysis of human-human dyadic interaction motions. Using IRL, unique control policies are learned for each dyadic interaction scenario. An end-to-end learning framework is introduced that enables acquiring effective control policies for the robot interaction skill by imitating human dyadic movement behaviors. In the last section, I summarize how the previously developed human related motion models are incorporated into a stochastic trajectory optimization algorithm and how robot interaction policies are improved by human guidance during interaction for realizing natural and safe HRI.

1.1 Human Motion Understanding and Prediction

One prerequisite of effective robot motion planning in close proximity human-robot interaction (HRI) is to understand human movement behavior. Thanks to the remarkable capability of the central nervous system (CNS) on controlling biomechanical properties, humans can learn and use a wide repertoire of motor skills. Adaptation to the environmental conditions, such as agile arm reaching motions closer to a dyadic partner, is not only possible but also almost unconsciously achieved with such sensorimotor skills. Despite the recent advancements on sensor technologies [10], it is not within our reach to observe and decode the internal mechanisms of how the CNS coordinates learning, adaptation, and control of our actions. Is it then possible to identify or model control strategies by only observing human movement?

If such control models are available, they can be used for predicting human motions so that the robotic agent proactively take corresponding actions. Moreover, those models can also be transferred to robot motion generation and control procedures in order to imitate human movement behavior. All of these capabilities would enable the robotic agent to effectively work in close proximity to humans, and allow natural and safe HRI.

Two main approaches have been proposed to explain the underlying principles for human motor control. The first one, which is closest to the current robotics engineering perspective, asserts that motion control is carried out on the level of muscle force generation. Control laws are sought for human motions based on an optimal control problem (OCP) formulation using a dynamical model of the human biomechanical system, and an optimality criterion defined by a cost function. However, different cost functions within an OCP lead to disparate movement trajectories. In essence, even though a cost function (e.g. min-jerk [3], min-torque-change [11]) is identified to describe a certain behavior, a generalized optimal control model is missing that can capture a wide range of human movement behavior [12]. On the other hand, the equilibrium point hypothesis (EPH) aims for a unifying approach by using the stability properties of the human arm, while considering the posture and the motion together [13]. The limitations of EPH have been discussed in the literature, including its inability to explain the adaptation of human motor control to environmental changes and to account for motions with low arm stiffness (for a detailed discussion, see [14]).

Model-based approaches, such as optimal control formulations, are *descriptive* as they explain a possible control strategy by only analyzing the observed state of a system. Such an understanding is useful for a multitude of problems, e.g. motor performance evaluation for detecting disabilities due to neural disorders by comparing control models of patients and healthy subjects [15]; detection of deviations of personal motion behaviors w.r.t. the previously identified motor control models, e.g. due to exhaustion [16]. Specifically, for human–robot interaction (HRI), person-specific control models enable the robot to detect the underlying cause of behavioral anomalies for providing better assistance and safety. On the other hand, for predicting human motion effectively, it is not necessary to identify such control mechanisms provided that humanlike motion trajectories can be modeled accurately, and generated during interaction efficiently. In essence, a framework that can both *describe* and *generate* human movement behavior is crucial to develop competent human-in-the-loop robot control and motion generation systems.

Next, I discuss the inverse optimal control (IOC) approach we used to find evidence for the hypothesized multiple internal models within the CNS. The hybrid model, which is constructed as a combination of model-based and data-driven methods, is also introduced to efficiently predict human motions while learning person specific motor control models. Our analysis provides a new perspective to understand and model human sensorimotor control, and these formulations enable the integration of those models into HRI.

1.1.1 Human motor control and inverse optimal control[1]

The main research question in this line of my work is whether motor control strategies can be identified by only analyzing human movement. In this thesis, the focus was on stereotypical point-to-point arm reaching movements, which can be considered approximately optimal after some learning phase [1]. Accordingly, the optimality principle has been followed. However, rather than finding a single cost function, a composite cost model, inspired by the multiple internal models (MIMs) hypothesis [17], was formed to define the OCP. The composite model was formed as a convex combination of multiple cost functions, each of which was shown to represent a certain aspect of human arm reaching motions accurately in a given scenario. Each cost function can be regarded as forming a distinct OCP, hence, solving the composite model corresponds to minimizing all the costs simultaneously while satisfying the constraints. Since our goal is to find a control law given a single reaching task, this entails an IOC problem, for which we developed a bi-level

[1] This work has previously appeared in the following publication: [235]

optimization formulation comprising an upper (ULP) and a lower level program (LLP). Human movement data was used to find the best combination of cost functions for describing the control policy within the CNS when optimality is assumed.

The LLP forms an OCP with multiple cost functions, each of which can be considered to form an optimal control model. The ULP computes the optimal combination of those models that result in a movement profile as similar as possible to humans'. However, the weighting factors of this combination is solved for each movement task with a distinct initial and final posture, and the solution to each particular movement results in a different composite model. This leads to the question of whether there exists a mapping from task parameters to cost combinations, or in other words, optimal control models. In addition, is there a high-level mechanism that regulates the contribution of different controllers depending on the task? In motor control literature, MIMs hypothesis has been formed to explain this remarkable generalization capability of the CNS [17]. However, only a few probabilistic computational models have been proposed [18]–[21]. The focus has been on the learning and switching of multiple competing models for different manipulation skills, and human reaching motions have not been evaluated extensively. In addition, the biomechanical properties has not been considered. In essence, they fail to provide a descriptive model to explain stereotypical human arm reaching behaviors.

The developed IOC framework is used in order to find the combination of internal models, which are hypothesized to exist within the CNS, and to explain how this combination changes for different reaching tasks. We conducted an experiment where participants executed a comprehensive set of free-space reaching motions. The results show that there is a trade-off between kinematics and dynamics based controllers depending on the reaching task. In addition, this trade-off depends on the initial and final arm configurations, which in turn affect the musculoskeletal load to be controlled. Given this insight, we further provide a discomfort metric to demonstrate its influence on the contribution of different inverse internal models. This formulation together with our analysis not only supports the MIM hypothesis but also suggests a hierarchical framework for the control of human reaching motions by the CNS.

Understanding and modeling the control of human arm reaching is the first step towards establishing one of the foundations for achieving human-in-the-loop control and motion planning for close proximity HRI. In the next section, I discuss how those optimal control models are combined with movement primitives (MPs) to predict human motions effectively.

1.1.2 Movement primitives for human motion prediction[2]

As discussed previously, humans exhibit remarkable versatility to adapt to each other's motion and work effectively in close proximity. Human upper body motions, specifically the arm motions, are fast and require agile and responsive motion generation by the robotic agent during dyadic interaction. Even though the optimal control formulation is informative and an important step for understanding and modeling how humans control their movements (Sec. 1.1.1), it does not provide efficient computational performance for predicting human motions during interaction.

For building a *descriptive* model, the motor control redundancy and the nonlinear characteristic of the human arm as a dynamical system pose significant challenges. The latent biomechanical

[2] This line of my research has previously appeared in the following publications: [236]–[239]

properties that the CNS uses for human motor control are still not fully understood (Sec. 1.1.1). The IOC formulation provides a suitable approach to fit control models to the observed human arm reaching behaviors. However, solving an optimization problem constrained on a dynamical system is computationally slow, whereas dyadic interaction necessitates online-capable *generative* models. In addition, there are intra- and inter-personal movement variations due to physiological differences, sensorimotor noise, and learning experiences [5], which in turn complicate modeling human movement behavior even further.

A multifaceted approach can tackle those challenges. As mentioned earlier, (inverse) optimal control formulations are able to identify the underlying principles of human motor control for reaching task. Data-driven methods, on the other hand, afford learning from demonstrations in a compact and computationally efficient fashion. In addition, it is easier to capture the variations in human movement with probabilistic methods. Then, a hybrid approach, combining model-based and data-driven methods, enables acquiring task-dependent and person-specific control models while predicting human motions online.

In this thesis, such a hybrid approach is proposed. Previously developed IOC approach (Sec. 1.1.1), which determines the combination of cost functions that governs a motion execution, is augmented with a data-driven probabilistic movement primitives (ProMPs) method. This hybrid model allows predicting human motions online while still taking into account motor variability and the interpersonal differences. In effect, it affords both a *descriptive* and a *generative* model of human reaching motions that can be effectively utilized online for human-in-the-loop robot control and task execution.

1.2 Modeling Human-Human Interaction Behavior

Even though modeling human motor control informs us about the characteristics of human movement, non-verbal close proximity human-human interaction (HHI) behavior requires further investigation due to the dynamic nature of motion planning. In essence, there is a permanent action-perception loop running for both partners during dyadic interaction. Humans rely on not only some physical constraints, but also social signals during collaboration. A natural dyadic interaction cannot be achieved solely by generating collision free trajectories. As the robotic agent is expected to collaborate with humans skillfully similar to how humans interact with each other, these latent features have to be incorporated into robot control and motion generation frameworks. Similar to finding control models for human movement in isolation, can such effective interaction behavior skills be learned for autonomous systems?

Without intuitive interaction skills, a robotic agent would only act as a helper for a human partner. In effect, this leads to leader-follower (or master-slave) architectures, which has been investigated frequently in physical human-robot interaction (HRI) [22], haptic assistance [23], and teleoperation tasks [24]. Considering the diversity in relative positioning of partners and task setup, human capability to adapt and efficiently collaborate with others in close proximity is extraordinary. This is only possible by a generalized and/or highly flexible decision making and motion control models.

Human factors involved in dyadic interaction necessitate an interdisciplinary approach. Perspectives from human sensorimotor control research together with physiology and cognitive science studies can collectively help us investigate such flexibility and effectiveness of human interaction skills. Few studies have looked into dyadic interaction motions [25]–[27], but focusing solely on learning movement models for a specific task from data. Human factors have mostly been investigated in social sciences, and cognitive science. Particularly for attention theory studies, the role of visual perception and what people pay attention to have been investigated [28]. Those studies provide key directions to systematically analyze dyadic interaction. However, identifying the interaction related features and modeling the movement behavior have to be considered concurrently with such methodical analysis.

My research contributes a novel categorization of interaction scenarios and provides two imitation learning formulations to model close proximity interaction movements. These studies encourage exploring non-physical joint movement and building control policies for dyadic interaction systematically and comprehensively in a unified way in future studies.

1.2.1 Human-human interaction & inverse reinforcement learning[3]

Here, I briefly discuss our analysis on human-human interaction in close proximity. By abstracting tasks and using insights from attention theory, this analysis results in an *ontology* that defines the types and properties of dyadic interactions, as well as the relationships between them. The variations in task demands and influence of perception signals during close proximity interaction between two humans guide our first steps to build an *ontology* systematically. For instance, visual field of a person alters the attention, which in turn affects the human interaction behavior. Furthermore, dyad positioning and task regions are critical in terms of constraining human joint movement. The experimental setup was devised to cover these variations in human perception and task properties to take into account their influence on movement behavior.

This *ontology* enables us to group interaction behaviors into separate cases, each of which can be represented by a graph representation. A single case is defined by the relative positioning of the partners and the location of task related regions in a way that these combined properties result in different movement strategies. The features extracted during interaction are used to classify cases quantitatively. Using inverse reinforcement learning, we find unique interaction policies, represented as combination of cost functions based on distance features, for each case.

The *ontology* has been tested on human-human interaction data that covers a wide range of dyadic movement behaviors. The results show that it can be used to classify and model human dyadic movements accurately, while the learned policies enable generating the appropriate robot trajectories similar to humans' during interaction. Such a thorough coverage and systematic analysis of HHI cases serve as a base and reference for later studies to propose and benchmark new learning algorithms, and evaluate close proximity HRI behaviors for a wide range of scenarios.

1.2.2 Learning dyadic interaction behaviors from demonstrations[4]

In the previous section, I introduced a systematic analysis of HHI behaviors which leads to an *ontology*. Nonetheless, it is unclear which properties of joint movement behavior defines a natural dyadic interaction dynamics, which in turn impedes modeling such a dynamical system explicitly.

[3] This work has been submitted and under review for publication: [240]

[4] This work has previously appeared in the following publication: [241]

The implemented inverse reinforcement learning (IRL) formulation (Sec. 1.2.1) depends on the selection of features we assumed relevant for interaction. However, it is very likely that there are latent features involved in this dynamic motion planning. With the recent advancements in (deep) learning algorithms and tools, neural networks have been successfully used to discover and represent such latent properties within complex data. Hence, given recorded HHI data, these features can be captured while learning a mapping from interaction related quantities (e.g. distance values, artificial forces) to control signals.

We developed an end-to-end learning framework to imitate human interaction behavior control. Using HHI data, a recurrent neural network (RNN) is trained to learn a control policy for a robotic agent to generate interaction behavior similar to humans'. A novel activation function has been introduced, and evaluated exhaustively along with the state-of-the-art network architectures. The results have been analyzed in terms of humanlikeness (i.e., w.r.t. the temporal and spatial features of generated vs. recorded motions), convergence and reachability properties, and noise suppression.

By using this end-to-end framework, a generalized interaction control policy is learned for the robot end effector without making any modeling assumptions on the dyadic motion behavior. This relaxation enables capturing latent features related to interaction movement. In addition, the framework allows imitation of human motions with a varying level of detail depending on the network complexity. In that regard, it is now possible to build complex interaction behavior models depending on the task by hierarchically combining different types of policies.

1.3 Close Proximity Human-Robot Interaction

While learning control policies for human movement allows us to predict and imitate human motions, they can be exploited fully only if they are incorporated into robot motion plans effectively. For achieving natural interaction, human perception of a robotic agent's movement has to be considered as well. In addition to the safety and efficiency aspects of robot movement, it should be predictable by the human partner. In essence, it is necessary to integrate not only physical but also social constraints into robot motion planning. Current motion planners are largely ignorant of this crucial objective.

Motion planners with a human-in-the-loop focus enable vast opportunities to introduce robotic partners in daily life, e.g. as household service robots and collaborative robots in manufacturing. Recently, there has been an increased interest in close proximity HRI. Safety aspect of such interactions has been one of the initial concerns [6], [27], [29]. However, human partner has mostly been treated as a dynamic obstacle to be avoided, while the robotic agent has been indifferent to how humans perceive the interaction. As mentioned earlier, few studies investigate motion models for the robot that can be perceived as legible [7], predictable [8], and understandable [30] by the human partner. Yet, the close proximity interaction, where all those aspects are highly interdependent, has been neglected. This dependency requires the integration of human factors into the autonomous motion planning skill of the robotic agent.

There are several challenges for constructing interaction behavior control for robotic agents. The inter- and intra-personal movement variations cause a predicament to predict human motions and to construct generalized motion controllers that can handle such uncertainty. For this same reason, reactive motion planners, which are solely designed to avoid obstacles, cannot provide intuitive and safe interaction. Trajectory optimization methods are favorable candidates as safety criteria can be handled as constraints or additional cost functions. However, the computational efficiency presents a challenge for fast paced close proximity dyadic interaction. Still, the robotic agent is expected to generate natural interaction behavior while considering the safety aspects. Finding a priori model of intuitive interaction behavior is an open research question. Instead of following this path, I ask in this thesis whether it is possible for the robot to improve its policy online w.r.t. the human partner's preferences during interaction.

My research introduces (*i*) a stochastic trajectory optimization framework for safe and effective dyadic interaction, and (*ii*) a policy improvement formulation for the robot motion adaptation w.r.t. the feedback of the human partner. These motion planners have become available by consolidating model-based optimization with data-driven methods and also by updating the policy online using the perception data obtained during interaction. Overall, these two novel approaches verify the significance of human-in-the-loop planning for close proximity safe and intuitive human-robot interaction (HRI). In addition, they pave the way for a long-term learning framework for personalized assistance.

1.3.1 Trajectory optimization for robot motion planning[5]

For close proximity HRI, a novel **P**rogressive **T**rajectory **O**ptimization-based **M**otion **P**lanning (PTOMP) approach is proposed. Such a motion planner has to be responsive to the human partner's movement behavior and intent, rather than just acting reflexively. This requires an adaptive planning by taking into account not only the current state but also the likely future motion trajectory of human body. Adaptation, in turn, poses computational efficiency challenges for the optimization algorithm, as its hyperparameters induce a trade-off between accuracy and computation time. By complementing the model-based optimization algorithm with a data-driven approach, a balance between those criterion has been achieved.

PTOMP allows safe and responsive robot motion generation thanks to the online parameter adaptation of the optimization algorithm, and the effective person-specific motion prediction. A mapping from interaction related features (e.g. distance between human and robot, time to collision, etc.) to feasible hyperparameters for the optimization algorithm has been learned through Gaussian process regression (GPR). The previously introduced probabilistic movement primitives (ProMPs) formulation estimates a trajectory distribution over human motion, that was learned in an offline phase (Sec. 1.1.2). The proposed stochastic optimization-based planning algorithm progressively acquires feasible hyperparameters from the learned regression model and re-plans the motion online, that ensures collision avoidance, while minimizing the task-related trajectory cost. Several cost functions, which are found to describe human point-to-point reaching motions (Sec. 1.1.1), are also combined to form a composite model that mimics humanlike arm reaching control. PTOMP has been tested on simulation and compared to several state-of-the-art algorithms for various HRI scenarios. It provides safer close proximity interaction, whereas the other trajectory optimizers fail frequently.

This work contributes a new approach to optimal online motion planning for human-robot interac-

[5] This work has previously appeared in the following publication: [242]

tion scenarios. The model-based formulation with a novel parameter adaptation and the human-in-the-loop focus allow the robotic agent to respect the model constraints and to be perceptive about its partner, which in turn produce safer interaction without sacrificing task fulfillment. Online adaptation is feasible even in more demanding real-world scenarios by PTOMP since its parallelization capability offers a highly scalable framework.

1.3.2 Intuitive human-robot interaction and policy improvement

Even though trajectory optimization enables effective and safe close proximity HRI, for a natural interaction, the robotic agent is also expected to attune its planning strategy according to the human partner's perception. For instance, humans should also be able to understandable the robot motion and its intent, yet what makes the intention of motion *readable* by humans is not clear. By combining interaction dependent costs with task and movement related cost functions, robot motion can be optimized for generating not only safe but also intuitive interaction behavior. However, people might have different expectations and preferences on how such a *readable* movement should be realized by their robotic partners.

My hypothesis is that such movement features can be learned by the robotic agent during interaction through the feedback obtained from humans. We implemented a policy improvement formulation, that is both generic and adaptive to new users and tasks, to find the optimal policy for safe yet understandable robot motion generation. An experiment was conducted, where subjects cooperated with the robot to perform a pick-and-place task in close proximity. Similar to PTOMP framework (Sec. 1.3.1), human motions are predicted for proactive avoidance behavior (Sec. 1.1.2). Response and reaction times of subjects, along with their accuracy, and explicit feedback on the readability of the robot partner's motion are taken into account for improving the control policy of the robot online. The results show that the robot is able to adapt its behaviors to personal preferences allowing the humans to recognize the robot's intentions while performing the joint tasks confidently.

The proposed online policy adaptation with the generic composite cost model has proven to be an effective approach to provide intuitive close proximity HRI. It can directly be used for different tasks, e.g. even for navigation of an autonomous agent among people, where the robot shares a workspace with humans. Furthermore, the generic formulation together with online improvement of the control policy facilitates long-term continuous learning and personalization of robotic assistance technologies.

1.4 Contributions and Thesis Overview

In the rest of this thesis, I will present in detail my interdisciplinary approach to enable safe and intuitive close proximity human-robot interaction (HRI) by combining (*i*) human sensorimotor control models, (*ii*) human-human interaction (HHI) dynamics, and (*iii*) novel learning and optimization algorithms for robot interaction behavior generation. As the goal is to build human-in-the-loop learning and control formulations for robot interaction skill acquisition, the primary focus is always on human motion behaviors.

Starting with this focal point, Chapter 2 explains the first contribution of the thesis, which is the identification of control policies for human reaching motions as a composition of several optimal control models. This is the first study that provides evidence for the multiple internal models (MIMs) hypothesis from an optimality of movements perspective. A trade-off between those models, that depends on the task parameters, have been discovered, which led to our proposition of a hierarchical control structure within the central nervous system (CNS).

Chapter 3 builds upon this inverse optimal control (IOC) approach (Ch. 2) by combining the data-driven probabilistic movement primitives (ProMPs) method to construct human movement representations. With this hybrid approach, not only the person specific composite control models are learned, but also intrapersonal movement variations are captured. Furthermore, human motions are predicted online with high accuracy, that allows building effective human-in-the-loop motion controllers for robotic agents.

Chapter 4 focuses on classification and learning of human-human interaction behaviors. A novel graph-based ontology is constructed that defines the types and properties of close proximity interactions as well as the relationships between them. In addition, interaction behavior models are learned by an inverse reinforcement learning (IRL) method. This unified and general framework can be used as a reference for future dyadic interaction analysis and learning studies. Chapter 5 also presents a learning approach with recurrent neural networks for cross-domain sequence learning of control signals from positional data acquired for the previous work discussed in Chapter 4. In essence, a generalized policy for dyadic interaction behavior is extracted and transferred from human-human data to control an autonomous agent during close proximity HRI.

In Chapter 6 and 7, two algorithms are presented that enable constructing dyadic interaction skills for autonomous agents to provide intuitive HRI. First, a stochastic trajectory optimization algorithm is proposed that progressively adjust hyperparameters to proactively plan the robot movement depending on the human partner's predicted motion (Ch. 3). Cost functions identified for human motor skills (Ch. 2) are used for computing these locally optimal motion plans. This integrated approach results in a natural, safe and effective close proximity interaction. Second, a generic policy together with a composite cost model is formulated, that enables an autonomous robotic agent to learn intuitive dyadic movement skills online during interaction by a policy improvement algorithm.

In sum, the trajectory of this dissertation demonstrates the necessity of an interdisciplinary approach to tackle human-in-the-loop learning and control problems for close proximity HRI. Due to the multiplicity of requirements, which might have conflicting constraints, model-based methods have to be integrated with data-driven approaches.

Part I

Human Motion Understanding, Modeling, and Prediction

2

An Inverse Optimal Control Approach to Explain Human Arm Reaching Control Based on Multiple Internal Models[1]

Human motor control is highly efficient in generating accurate and appropriate motor behavior for a multitude of tasks. This work examines how kinematic and dynamic properties of the musculoskeletal system are controlled to achieve such efficiency. Even though recent studies have shown that the human motor control relies on multiple models, how the CNS controls this combination is not fully addressed. In this study, we utilize an IOC framework in order to find the combination of those internal models and how this combination changes for different reaching tasks. We conducted an experiment where participants executed a comprehensive set of free-space reaching motions. The results show that there is a trade-off between kinematics and dynamics based controllers depending on the reaching task. In addition, this trade-off depends on the initial and final arm configurations, which in turn affect the musculoskeletal load to be controlled. Given this insight, we further provide a discomfort metric to demonstrate its influence on the contribution of different inverse internal models. This formulation together with our analysis not only support the MIMs hypothesis but also suggest a hierarchical framework for the control of human reaching motions by the CNS.

2.1 Introduction

Human motor control shows a remarkable ability to regulate stereotypical human motions that are observed for a broad range of tasks in daily life. On the one hand, an influential idea discussed in literature is that, the central nervous system (CNS) utilizes internal models, which hypothetically represent the control of dynamics and kinematics of movement, to achieve such an efficient motor behavior [17], [31]. On the other hand, another major line of work hypothesizes that, after an initial learning phase, the stereotypical human motions are approximately optimal with respect to an unknown criterion. Hence, modeling of human motor behavior has been posed as an optimal

[1] This work has previously appeared in the following publication: [235]

control problem and several cost functions have been proposed [11], [32]–[35]. If one interprets an optimal control problem with different cost functions as different control models, the combination of them corresponds to the multiple inverse internal models formulation. Depending on the task, the combinations of those models can be adapted accordingly. If we assume the space of 3D free space motion control models is represented as a combination of internals models within the cerebellum, then the follow-up question is whether there exists a hierarchical control where the contribution of those different internal models are regulated. Analysis of such a hierarchical control mechanism can improve our understanding and interpretation of how the CNS might be internally representing and utilizing some critical metrics to implicitly control stereotypical movements for a multitude of tasks. In this work, we search for such internal models and their combinations to support the multiple internal model (MIM) hypothesis by focusing on the arm reaching motions in a wide range of the 3D space. Subsequently, how those internal models are combined for arm motions and why their contributions change depending on the reaching task are investigated.

Searching for the physiological foundations of motor control and internal models, various experiments are discussed in the literature [36]–[38]. Resulting hypotheses are that the posterior parietal cortex is updating the motor plan, while the cerebellum might capture the feedforward control signals. Further structures of the human nervous system might contain the state estimator and comparator [37]. In this regard, feedforward and feedback control mechanisms along with learning and adaptation processes for motor control has been studied extensively [16], [39]–[43]. These results suggest that computational approaches to motor learning and control should include two separate performance errors rather than one [44], and that the brain learns multiple internal models that can be combined as required by the circumstance [45]. Several studies investigated whether multiple internal models could be learned concurrently, and switched depending on the task [46], [47]. It was shown that two learned models could be additively combined [48], and a modular controller selection architecture was proposed which relied on a linear combination of the outputs of the multiple inverse internal models [18]. However, verification of those composite models for a broad range of tasks is missing and how they are combined and controlled efficiently by the CNS are still unclear.

Before even identifying multiple models utilized by the CNS, finding a single internal model is a challenging computational problem. A common feature of motor control is that the task requirements can be met by infinitely many diverse movements. Thus, providing only the boundary conditions of the motion for given dynamics leads to an ill-defined problem. The ambiguity caused by this problem can be resolved if an optimality principle is applied. Accordingly, the basis of many scientific theories on human motor control is formed by optimality principles. A large number of models of open-loop motor control exist and each model claims to describe human motion, but several models are incompatible with others. The starting point for the derivation of a cost function are characteristics of the human arm movements and the human as a biological entity. Human motor control has been speculated to minimize the sequence of control signals [49]–[51], or limb states [2]. These minimization strategies are related to physiological and task variables such as smoothness of the hand path [11], [32], accuracy [33], [52] or error and effort [5], [53]–[56]. However, these single models appear to be not descriptive enough for a broader range of tasks. Hence, recent studies focus on finding a combination of such models. These methods solve an inverse optimal control (IOC) problem where the contribution of different optimal control models are computed [57]–[59]. The best combination of models that results in a trajectory as similar as to the recorded human motion data is identified through iteratively comparing the calculated

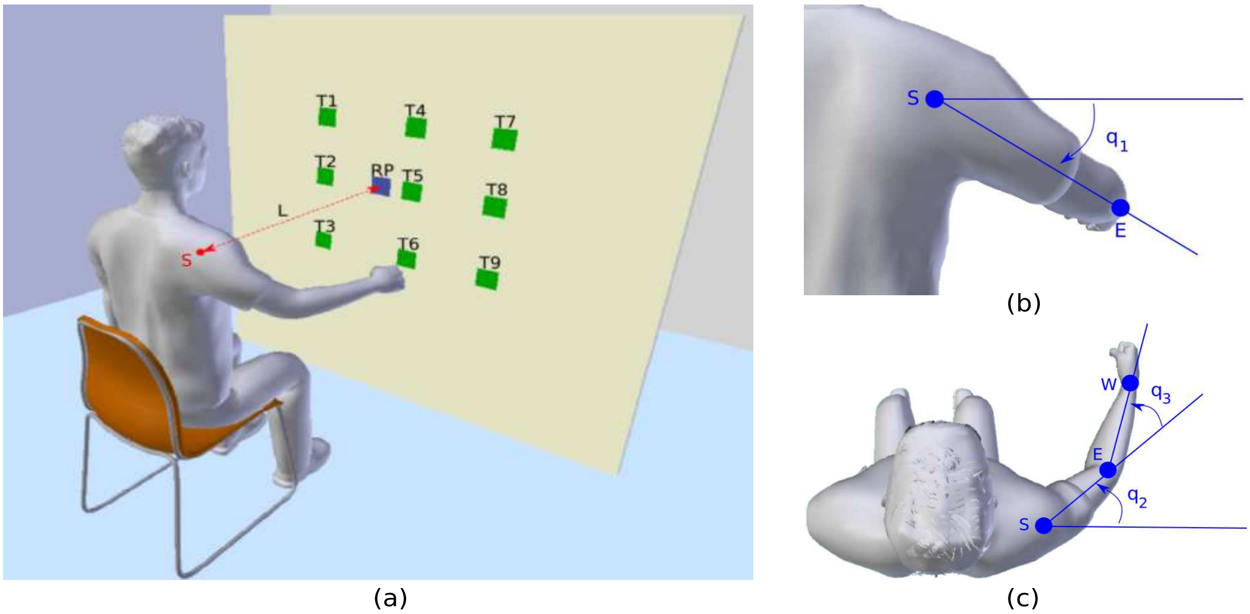


Figure 2.1: Experimental setup. (a) Experimental overview. T1 to T9 indicate nine target areas. RP stands for the reference point, which is used to adjust subject's position. The distance L between the center of the shoulder joint S and the RP is selected as 80% of the total arm length. Subjects are required to reach nine target areas from nine different starting arm postures. (b)-(c) Three rotations defined in our arm model. S, E, W are the positions of shoulder, elbow, and wrist joints, respectively. When the arm is in full stretched out posture, q_1, q_2 and q_3 all have zero rotations.

motion trajectory with the observation, while still satisfying all individual control models w.r.t. the dynamical system and the constraints. IOC formulation describes human motion better than the previous single models, yet it forms a more complex computational problem, which emphasizes the importance of a better understanding of why and how those models are efficiently utilized by the CNS.

In this study, we focus on finding evidence for MIMs hypothesis and understanding how these models are efficiently combined by the CNS. Considering that internal models mimic the transformations between system states, motor commands, and sensory signals, an optimal control problem (OCP) can be regarded as an inverse internal model by providing the necessary control signals to carry out the reaching task [17], [44], [60]. Our hypothesis is that, if a comprehensive analysis is done on a rich set of human motion, the change in the contribution of the underlying motor control structures can be observed in a way to reveal the factors that explain such an adjustment. We limit ourselves to examining simple, 3D free-space arm reaching movements. Simple movements can be considered to make up a basis set of which more complicated movements are composed [61]–[63], and as such, studying them provides insight into the control of more complex movements. We devised an experiment in which a wide range of 3D reaching motions are recorded (Fig. 2.1). For the arm reaching task we focused, a single forward model, i.e. the arm dynamics, is paired with multiple inverse internal models, i.e. the composite of OCPs where each OCP is associated with a specific cost function.

This study presents both a comprehensive optimality analysis on human arm reaching motions in 3D space and a trade-off between dynamics and kinematics based controllers depending on the reaching motion type. The main contributions of our work are three-fold:

- Multiple inverse internal models, represented as a combination of different controllers, are discovered to be collectively controlling the 3D free-space arm reaching movements.
- We explain quantitatively how the contribution of these inverse internal models change w.r.t. the task parameters.
- Finally, a hierarchical control model that regulates MIMs for controlling reaching motions is proposed.

2.2 Methods

Here, first the musculoskeletal model used for the human arm is described. Then a brief formulation of an optimal control problem (OCP) and the numerical solution method are provided. This OCP together with a derivative-free optimizer forms the inverse optimal control (IOC) formulation as a bi-level optimization problem to solve for the composite control models of human reaching movement. A new metric is introduced to explain how the central nervous system (CNS) might be controlling this compositeness. Lastly, the experimental task is detailed out.

2.2.1 Dynamical model of the arm

A common approach to model the arm dynamics in 3D reaching motions is to consider it as articulated rigid bodies. By ignoring the hand movements, the arm can be treated as a musculoskeletal system which consists of four degree-of-freedoms (DoFs), where the shoulder joint has three rotations and the elbow joint has one rotation. Due to the fact that in our preliminary tests, the internal/external rotation of the shoulder joint is merely activated for the given reaching tasks, it is neglected in the dynamic model. This simplification can increase the computational efficiency of the inverse optimal control calculations while preserving enough accuracy in the results. According to the classical Lagrangian formalism, the dynamics of the 3-DoF arm model can be defined as

$$\boldsymbol{\tau} = M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + G(\boldsymbol{q}), \tag{2.1}$$

where the variable $\boldsymbol{q} = (q_1, q_2, q_3)^\top$ denotes the three joint angles and $\boldsymbol{\tau} = (\tau_1, \tau_2, \tau_3)^\top$ represents the torques. Time derivatives of $\boldsymbol{q}$, i.e. $\dot{\boldsymbol{q}}$ and $\ddot{\boldsymbol{q}}$, are the joint angle velocities and joint angle accelerations, respectively. M, C and G are the inertia matrix, the Coriolis/centripetal terms and the gravitational vector, respectively. As the viscous frictions and elastic properties of the tissues are difficult to estimate, they are neglected in the dynamics. The upper arm length and the forearm length, as well as the mass, inertia and distance to the center of mass are defined as described in previous researches [64], [65]. When the arm is in fully stretched out position, q_1, q_2 and q_3 all have zero configurations. The positive rotation direction of corresponding joint angles are given in Fig. 2.1b,c. The dynamics is obtained through the AUTOLEV [66] tool, which is an interactive symbolic dynamics program for formulating motion equations.

2.2.2 Optimal control problem

The goal of OCP is to find the time-dependent state variable functions and control signal functions while minimizing a given cost function. During the calculation, several constraints have to be fulfilled, including the equality constraints, inequality constraints, and more important, the dynamic relationship between control signals and state variables, which is represented by ordinary differential equations (ODE). Thus, a classical OCP can be defined as follows.

Definition: *Let $\boldsymbol{x} \in \mathbb{R}^n$ be the time-dependent state variable functions, $\boldsymbol{u} \in \mathbb{R}^m$ be the time-dependent control signal functions, and T is the terminal time. The cost function is represented by $\boldsymbol{\phi}$, then the general OCP is defined by*

$$\min_{\boldsymbol{x},\boldsymbol{u}} \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u}) := \phi_b(\boldsymbol{x}(0), \boldsymbol{x}(T)) + \int_0^T \phi_I(\boldsymbol{x}(t), \boldsymbol{u}(t))dt, \tag{2.2}$$

where ϕ_b is the terminal cost term and ϕ_I is the integral cost term. Note that, $\boldsymbol{x}$, and $\boldsymbol{u}$ are subject to the dynamic relationship constraint which is represented by n ordinary differential equations as

$$\dot{\boldsymbol{x}}(t) = \varphi(\boldsymbol{x}(t), \boldsymbol{u}(t)), \tag{2.3}$$

and also the inequality and boundary (equality) constraints as

$$g(\boldsymbol{x}(t), \boldsymbol{u}(t)) \leq 0, \quad b(\boldsymbol{x}(0), \boldsymbol{x}(T)) = 0. \tag{2.4}$$

Note that in most cases the boundary constraints are considered as

$$\boldsymbol{x}(0) = \boldsymbol{x}^s, \quad \boldsymbol{x}(T) = \boldsymbol{x}^e \tag{2.5}$$

for given start values $\boldsymbol{x}^s \in \mathbb{R}^n$ and end values $\boldsymbol{x}^e \in \mathbb{R}^n$ of state variables.

The OCP is named according to its cost function. If the cost function is a combination of a terminal cost term and an integral cost term, then it is called *Bolza-problem*. If it only has the terminal cost term or only the integral cost term, it is called *Lagrange* or *Mayer*-problem, respectively. There are two main approaches to solve OCPs, *indirect*, and *direct*. Here, we focus on the *direct multiple shooting* approach used within our formulation.

2.2.3 Direct multiple shooting approach

Direct approach to continuous optimal control finitely discretizes the infinite dimensional decision variables, such as the control signals $\boldsymbol{u}(t)$ (Fig. 2.2). This allows approximating the original problem as a finite dimensional nonlinear program (NLP), which can be solved by structure exploiting numerical NLP solution methods. There are three different direct approaches: *direct single shooting*, *direct multiple shooting* and *direct collocation*. The difference between each approach is how they transcribe the OCP into a finite NLP. In this work, the direct multiple shooting is used.

The direct multiple shooting method first utilizes the piecewise control discretization. The control signal is chosen to be

$$\boldsymbol{u}(t) = \boldsymbol{q}_i, \quad \text{for} \quad t \in [t_i, t_{i+1}]. \tag{2.6}$$

After the discretization, the ordinary differential equations are solved in each separate time interval $[t_i, t_{i+1}]$ with an artificial initial starting values $\boldsymbol{s}_i$ as:

$$\dot{\boldsymbol{x}}_i(t; \boldsymbol{s}_i, \boldsymbol{q}_i) = \varphi(\boldsymbol{x}_i(t; \boldsymbol{s}_i, \boldsymbol{q}_i), \boldsymbol{q}_i), \quad \boldsymbol{x}_i(t; \boldsymbol{s}_i, \boldsymbol{q}_i) = \boldsymbol{s}_i, \quad t \in [t_i, t_{i+1}]. \tag{2.7}$$

This generates the state trajectory $\boldsymbol{x}_i(t; \boldsymbol{s}_i, \boldsymbol{q}_i)$ for each time interval. Thus the integral of the cost function for each time interval can be calculated individually as

$$l_i(\boldsymbol{s}_i, \boldsymbol{q}_i) = \int_{t_i}^{t_{i+1}} \phi_I(\boldsymbol{x}_i(t_i; \boldsymbol{s}_i, \boldsymbol{q}_i), \boldsymbol{q}_i) dt, \tag{2.8}$$

then the inequality constraints can be checked in divided time grid which is normally the same as the grid for control signals and states. The NLP generated by multiple shooting approach can be described as follows.

$$\min_{\boldsymbol{s},\boldsymbol{q}} \sum_{i=0}^{N-1} l_i(\boldsymbol{s}_i, \boldsymbol{q}_i) + \phi_b(\boldsymbol{s}_0, \boldsymbol{s}_N) \tag{2.9}$$

where the following constraints should be satisfied:

- initial value:

$$\boldsymbol{x}_0 - \boldsymbol{s}_0 = 0, \tag{2.10}$$

- continuity:

$$\boldsymbol{x}_i(t_{i+1}; \boldsymbol{s}_i, \boldsymbol{q}_i) - \boldsymbol{s}_{i+1} = 0, \quad i = 0, \dots, N-1, \tag{2.11}$$

- discretized inequality constraints:

$$g(\boldsymbol{s}_i, \boldsymbol{q}_i) \leq 0, \quad i = 0, \dots, N-1, \tag{2.12}$$

- terminal constraints:

$$b(\boldsymbol{s}_N) \leq 0. \tag{2.13}$$

The continuity conditions can also be interpreted as a discrete time dynamic system $\boldsymbol{s}_{i+1} = \varphi_i(\boldsymbol{s}_i, \boldsymbol{q}_i)$ by setting $\varphi_i(\boldsymbol{s}_i, \boldsymbol{q}_i) := \boldsymbol{x}_i(t_{i+1}; \boldsymbol{s}_i, \boldsymbol{q}_i)$. Then the NLP can be solved by a sparsity exploiting NLP solver. Compared to single shooting, the multiple shooting approach has several advantages. e.g. the state trajectory can be initialized and it has better local convergence properties for unstable dynamic systems.

2.2.4 Inverse optimal control as a bi-level optimization problem

The purpose of IOC is to identify the formulation of the OCP, specifically the cost function it optimizes, which best reproduce the observations. A numerical method for solving an IOC problem is to reformulate it as a bi-level optimization problem [58]. This method relies on the assumption that, the optimal cost function is a composite of several plausible basic cost functions. The contribution of each basic cost function is defined through a weight vector, and this weight vector is

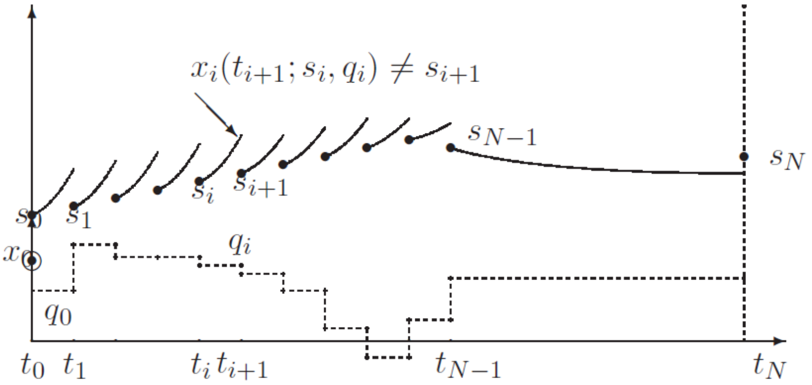


Figure 2.2: The NLP variables in the direct multiple shooting approach (taken from [67])

identified by using the bi-level optimization framework presented in (2.14).

$$
\begin{aligned}
&\text{Upper level program:} \quad \min_{\boldsymbol{\alpha}} \Phi(\boldsymbol{x}^*_{\boldsymbol{\alpha}}, \boldsymbol{x}^{obs}), \quad \text{with} \quad \sum_{i=1}^{N} \alpha_i = 1, \\
&\qquad\qquad\qquad\qquad \Updownarrow \\
&\text{Lower level program:} \quad \min_{\boldsymbol{x},\boldsymbol{u}} J(\boldsymbol{x}, \boldsymbol{u}|\boldsymbol{\alpha}) := \sum_{i=1}^{N} \alpha_i J_i, \quad \text{s.t.} \quad g(\boldsymbol{x}, \boldsymbol{u}) \leq 0, \quad h(\boldsymbol{x}, \boldsymbol{u}) = 0.
\end{aligned} \tag{2.14}
$$

2.2.4.1 Lower level program

The lower level program of the bi-level optimization is a direct OCP [68] given by

$$
\min_{\boldsymbol{x},\boldsymbol{u}} J(\boldsymbol{x}, \boldsymbol{u}|\boldsymbol{\alpha}) := \sum_{i=1}^{N} \alpha_i J_i, \quad \text{s.t.} \quad g(\boldsymbol{x}, \boldsymbol{u}) \leq 0, \quad h(\boldsymbol{x}, \boldsymbol{u}) = 0. \tag{2.15}
$$

The goal of OCP is to find the optimal trajectory which minimizes a given cost function J. Here J is assumed to be a linear combination of N basic cost functions J_i $(i = 1 \ldots N)$ which are weighted by the weight vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_N)$. The variables $\boldsymbol{x}$ and $\boldsymbol{u}$ are the vector of system states and control signals, respectively. With above explained arm model, the system states in this work are given as $\boldsymbol{x}^\top = (\boldsymbol{q}^\top, \dot{\boldsymbol{q}}^\top, \ddot{\boldsymbol{q}}^\top)$. Since the joint torques $\boldsymbol{\tau}$ are smoothly generated by muscle contractions [11], the control signals are defined as the time derivative of torques $\boldsymbol{u} = \dot{\boldsymbol{\tau}}$. Thus the OCP of reaching motions can be stated mathematically as: *find the admissible system state trajectory $\boldsymbol{x}^*_{\alpha}(t)$ and control signal trajectory $\boldsymbol{u}^*_{\alpha}(t)$ in time T, which minimize the cost function J with respect to a given weight vector $\boldsymbol{\alpha}$, while satisfying the system dynamics and the task constraints.* For reaching motions, the task constraints contain two parts: the initial condition $\boldsymbol{x}(0) = \boldsymbol{x}_s$ and the final condition $\boldsymbol{x}(T) = \boldsymbol{x}_e$ as the boundary constraints; limitations on joint angles $\boldsymbol{q}_{\min} \leq \boldsymbol{q}(t) \leq \boldsymbol{q}_{\max}$ as the inequality constraint. The constraints of joint angle velocities and joint angle accelerations are set to a large range since during the preliminary analysis they are identified to be merely active.

One classical method to solve OCP is to first transform it into a nonlinear programming (NLP) problem with constraints, then solve it by using structure exploiting numerical NLP solution methods. In our work, we utilize the multiple shooting method [67] with ACADO toolkit [69] to resolve OCPs.

Criterion	Equation
Hand jerk	$J_{HJ} = \int_0^T \dddot{x}^2 + \dddot{y}^2 + \dddot{z}^2 dt$
Joint angle acceleration	$J_{JA} = \int_0^T \ddot{q}_1^2 + \ddot{q}_2^2 + \ddot{q}_3^2 dt$
Joint angle jerk	$J_{JJ} = \int_0^T \dddot{q}_1^2 + \dddot{q}_2^2 + \dddot{q}_3^2 dt$
Torque change	$J_{TC} = \int_0^T \dot{\tau}_1^2 + \dot{\tau}_2^2 + \dot{\tau}_3^2 dt$
Torque	$J_{Tor} = \int_0^T \tau_1^2 + \tau_2^2 + \tau_3^2 dt$
Absolute work (energy)	$J_{Enr} = \int_0^T (\sum_{i=1}^3 \lvert \dot{q}_i \tau_i \rvert) dt$
Geodesic	$J_{Geo} = \int_0^T (\dot{\boldsymbol{q}}^\top M \dot{\boldsymbol{q}})^{1/2} dt$

Table 2.1: Cost functions proposed in literature. Variables x, y, z are the hand positions in Cartesian space. M denotes the inertia matrix. Corresponding references for the proposed criteria are given as: minimum hand jerk [32], minimum joint angle acceleration [70], minimum joint angle jerk [71], minimum torque change [11], minimum torque [2], minimum absolute work [72], [73] and minimum geodesic [74].

2.2.4.2 Selection of basic cost functions

The core part of the IOC framework is to select a set of reasonable basic cost functions. For arm movements, several cost functions were proposed in the past. These cost functions can be categorized into subjective and objective cost functions. Subjective cost functions refer to the decision from a subject, such as the minimum hand jerk [32], while objective cost functions are task-related. Since the integration of objective cost functions into OCP is difficult, only subjective cost functions are considered in this work. In literature, various subjective cost functions are proven to be useful in explaining human reaching motions (see Table 2.1). Generally, these cost functions can be grouped as two classes: (a) *kinematic cost functions*: the minimum hand jerk [32], the minimum joint angle acceleration [70] and the minimum joint angle jerk [71] are typical ones; and (b) *dynamic cost functions*: the minimum torque change [11], the minimum torque [2] and the minimum absolute work [72], [73] (also referred as minimum energy throughout this work) belong to this class; and finally the minimum *geodesic* criterion [74] is a junction of kinematic and dynamic cost functions, which yields the shortest path in configuration space while taking the kinetic energy into consideration. An example of the optimal end-effector trajectories solved from

OCPs with respect to different basic cost functions is given in Fig. 2.3. Among these proposed cost functions, we select five of them as the basic cost functions for IOC, which are the minimum hand jerk J_{HJ}, the minimum joint angle jerk J_{JJ}, the minimum torque change J_{TC}, the minimum energy J_{Enr} and the minimum geodesic J_{Geo}. The minimum joint angle acceleration is ignored since it gives quite similar solution to the minimum joint angle jerk, then the identification between these two cost functions is difficult. In addition, the minimum torque criterion is also neglected because in our preliminary tests we found it has the largest error in describing the reaching motions. Thus the combined cost function J for the direct OCP is defined as

$$J = \alpha_1 J_{HJ} + \alpha_2 J_{JJ} + \alpha_3 J_{TC} + \alpha_4 J_{Geo} + \alpha_5 J_{Enr}. \tag{2.16}$$

One more important issue in combining the basic cost functions, due to the different units, is that the range of the objective values of different cost functions are usually considerably different, thus they cannot directly be equally compared in Eq. (2.16). To overcome this problem, we introduce another scalar factor vector $\boldsymbol{S}$, with the purpose to balance the objective values of selected basic cost functions to the same range. Thus Eq. (2.16) is transformed into

$$J = \sum_i S_i \alpha_i J_i, \quad i \in \{HJ, JJ, TC, Geo, Enr\}. \tag{2.17}$$

To obtain the scalar factor vector for a given reaching task, five optimal trajectories $\boldsymbol{x}_i^*$ with respect to each basic cost function are first computed by solving the corresponding OCPs. Based on the results, the range of the objective value of each basic cost function can be defined through the minimal and maximal values as $Range_i = [J_{i,\min}, J_{i,\max}]$. Since all selected basic cost functions are integral cost terms and always produce positive values during the optimization, the minimal values are zeros for all cost functions $J_{i,\min} = 0$. Then the scalar factor vector can be generated directly by comparing the maximal values $J_{i,\max}$. In our experimental data, we found that the minimum joint angle jerk J_{JJ} tends to have the largest maximal objective value, therefore we set the scalar factor of the minimum joint angle jerk to 1, then the ratios between other basic cost functions and the minimum joint angle jerk are chosen to be the corresponding scalar factors.

$$S_i = \frac{J_{i,\max}}{J_{JJ,\max}}. \tag{2.18}$$

Note that the scalar factor vector varies when at least either the initial condition $\boldsymbol{x}_s$ or the final condition $\boldsymbol{x}_e$ changes. Thus before running the IOC for each given observation, the scalar factor vector needs to be determined in order to ensure the accuracy of the result.

2.2.4.3 Upper level program

The purpose of the upper level program is to find the optimal weight vector $\boldsymbol{\alpha}^*$ which minimizes the distance error between the optimal trajectory $\boldsymbol{x}_{\boldsymbol{\alpha}}^*(t)$ obtained from the lower level program and the observation $\boldsymbol{x}^{obs}$. This optimization problem can be represented as

$$\min_{\boldsymbol{\alpha}} \Phi(\boldsymbol{x}_{\boldsymbol{\alpha}}^*, \boldsymbol{x}^{obs}), \quad \text{with} \quad \sum_{i=1}^{N} \alpha_i = 1, \tag{2.19}$$

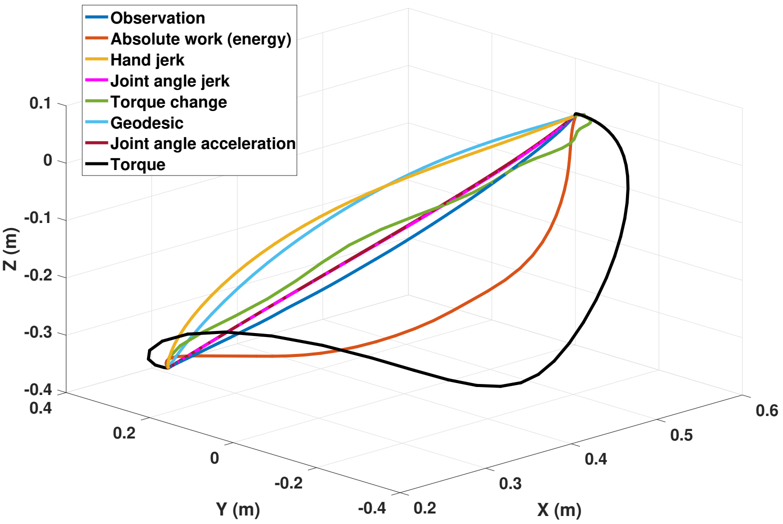


Figure 2.3: An example of the optimal end-effector trajectories solved from OCPs with respect to different basic cost functions. The variance in their predictions is clear. Only exception is the similarity of the predicted trajectories by minimum joint angle acceleration and the minimum joint angle jerk as they overlap in the figure.

where Φ is a metric which measures the distance error.

Selecting a good metric Φ is crucial in the bi-level optimization framework since it highly affects the decision on the optimal weight vector. The recorded observations are usually the position trajectories in Cartesian space represented by x, y, z coordinates. These observations cannot be directly compared by Φ because, on the one hand, the system states $\boldsymbol{x}$ are defined as joint angles, on the other hand, the position trajectories usually contain uncertainties, which come from: 1) the error from the torso movement, 2) the difference between the subject's actual arm length and the defined musculoskeletal system's arm length. No consistent results can be derived if a direct comparison to the position trajectories is implemented in Φ.

To address this problem, we transform the recorded position trajectories to the *relative position trajectories in arm model coordinate system* through the following steps:

1. Record the Cartesian position trajectories of the shoulder joint $\boldsymbol{t}_s = (t_{s,x}, t_{s,y}, t_{s,z})$, the elbow joint $\boldsymbol{t}_e = (t_{e,x}, t_{e,y}, t_{e,z})$ and the wrist joint $\boldsymbol{t}_w = (t_{w,x}, t_{w,y}, t_{w,z})$.
2. Derive the observed joint angle trajectory through the arm geometry as $\boldsymbol{q}^{obs} = G(\boldsymbol{t}_s, \boldsymbol{t}_e, \boldsymbol{t}_w)$. Since the roll rotation of the shoulder joint is neglected in our work, the translation function G can be easily obtained.
3. Compute the relative position trajectory (end-effector trajectory) in arm model coordinate system by using the kinematic relationship of the proposed arm model as $\boldsymbol{t}^{obs} = \delta(\boldsymbol{q}^{obs})$, where δ represents the function of the forward kinematics.

The relative end-effector trajectory $\boldsymbol{t}^{obs}$ eliminates the error caused by different arm lengths and the torso movements, thus it can be compared to the solution calculated from the lower level program.

Based on the feature compared in Φ, two different types of the distance metric can be formulated: one is *the joint angle metric*, where the observed joint angle trajectory $\boldsymbol{q}^{obs}$ is compared to the optimal system states trajectory $\boldsymbol{x}^*_{\boldsymbol{\alpha}}$, which also contains the joint angle trajectory $\boldsymbol{q}^*_{\boldsymbol{\alpha}}$; another is *the end-effector trajectory metric*, where at first the optimal end-effector trajectory $\boldsymbol{t}^*_{\boldsymbol{\alpha}}$ is computed from the optimal joint angle trajectory $\boldsymbol{q}^*_{\boldsymbol{\alpha}}$ by using the same forward kinematics function δ, then the distance error is calculated between the relative end-effector observation $\boldsymbol{t}^{obs}$ and $\boldsymbol{t}^*_{\boldsymbol{\alpha}}$.

In our preliminary tests we found that the end-effector trajectory metric has a better performance than the joint angle metric. Possible reason is that the three joint angles actually have different degrees of influence on the reaching motions [75]. However, it is not straightforward to determine the contribution of different joint angles, which could introduce further uncertainties and errors. Similar problem also occurs when combining the joint angle metric and the end-effector metric, since they have different units and it is difficult to balance them into the same range. Therefore in our work the distance metric of the upper level program only considers the end-effector trajectories, which can be treated as comparing two 3-dimensional signals. The dynamic time warping (DTW) algorithm [76] is implemented to calculate the distance error. In time series analysis, DTW is used for measuring the similarity between two temporal sequences which may vary in speed. The sequences are first warped in the time dimension and then compared to each other. With this, Eq. (2.19) can be stated as

$$\min_{\boldsymbol{\alpha}} \Phi(\boldsymbol{x}^*_{\boldsymbol{a}}, \boldsymbol{x}^{obs}) := \min_{\boldsymbol{\alpha}} D(\boldsymbol{t}^*_{\boldsymbol{\alpha}}, \boldsymbol{t}^{obs}), \tag{2.20}$$

where D denotes the DTW calculation.

To solve Eq. (2.20), common gradient-based methods and stochastic optimization algorithms are not applicable because of two reasons. First, the metric Φ is non-differentiable with respect to the weight vector $\boldsymbol{\alpha}$; second, before each calculation of Φ, a direct OCP must be solved in advance, thus it usually takes a few minutes for one evaluation. Specifically, the stochastic optimization algorithms (e.g. particle swarm optimization [77]) are also not suitable here, since they require more samples which will result in infeasible computation time. Hence, the upper level program is optimized by a robust derivative-free optimization (DFO) method. Here we use the method called CONDOR [78] for COnstrained, Non-linear, Direct, parallel optimization, which is a parallel extension of the Powell's method [79] based on the trust region algorithm [80]. Through a local approximation of Φ, it can find the optimal solution more efficiently than the common pattern search and stochastic optimization techniques. To reduce the computation time, the initial value of $\boldsymbol{\alpha}$ should be set properly before the optimization. Since among the five elements of $\boldsymbol{\alpha}$ only four are actually independent, and OCPs corresponding to the costs $J(\boldsymbol{\alpha})$ and $J(\lambda\boldsymbol{\alpha})$, $\lambda > 0$ are identical, a practical strategy is to fix one element to one and then adjust the remaining components with respect to it [57]. As all the basic cost functions are scaled into the same range, the value of other components can be restricted to stay in $[0, 1]$. During the optimization process, if any element is found larger than one, the optimization should be restarted with setting this element to one. In our experimental data, setting the weight of joint angle jerk to one gives the best results in most cases. After around 100 iterations, the algorithm converges to a local minimum. Note that due to the high non-linearity of the problem formulation, the global minimum is not available in the bi-level optimization [59]. In order to get more accurate results while keeping a reasonable computation time, we set the initial value of $\boldsymbol{\alpha}$ to $(0.5, 1, 0.5, 0.5, 0.5)$ and solve it three times with different initial search radii [79] as 0.15, 0.3 and 0.45, so that most range is covered within three IOC calculations.

The one results in the minimum distance error is considered as the final optimal weight vector $\boldsymbol{\alpha}^*$ and is normalized for later analysis.

2.2.5 Discomfort metric

The motion parameters of each reaching task can be represented as the initial joint angle configuration $\boldsymbol{q}_s = (q_{1s}, q_{2s}, q_{3s})$ and the final joint angle configuration $\boldsymbol{q}_e = (q_{1e}, q_{2e}, q_{3e})$. The proposed discomfort metric is a linear combination of six joint angles as

$$Discomfort = \beta_1 \frac{90 - q_{1s}}{180} + \beta_2 \frac{q_{2s}}{180} + \beta_3 \frac{q_{3s}}{180} + \beta_4 \frac{q_{1e} - q_{1s}}{180} + \beta_5 \frac{q_{2e} - q_{2s}}{180} + \beta_6 \frac{q_{3e} - q_{3s}}{180}, \quad (2.21)$$

where each parameter is scaled with its approximate joint angle limits and balanced with the normalized weight vector $\boldsymbol{\beta} = (\beta_i), i = 1 \dots 6, ||\boldsymbol{\beta}|| = 1$. The reason of introducing $\boldsymbol{\beta}$ is that, each joint angle possibly has different degrees of influence on the reaching motions [75]. Assuming the discomfort metric can explain the trade-off between the contributions of controllers we observed, the optimal weight vector $\boldsymbol{\beta}^*$ for a given set of reaching motions can be determined by utilizing the contribution of dynamics based controllers α_{Dyn} through the following processes:

1. For each given $\boldsymbol{\beta}$, a set of discomfort values can be derived from each reaching motion as $\boldsymbol{Discomfort} = (Discomfort_i), i = 1 \dots N$, where N is the number of total recorded reaching motions.
2. Along with the set of corresponding values of α_{Dyn} obtained from IOC calculations as $\boldsymbol{\alpha_{Dyn}} = (\alpha_{Dyn,i}), i = 1 \dots N$, the coefficient of determination R^2 can be calculated from the data set $(\boldsymbol{Discomfort}, \boldsymbol{\alpha_{Dyn}})$.
3. The optimal weight vector $\boldsymbol{\beta}^*$ is obtained by maximizing R^2.

Note that, rather than a deterministic mathematical explanation of the relationship between task demand and the contribution of dynamics based controllers α_{Dyn}, the proposed discomfort metric should be treated as a linear regression estimation of this relationship but with multi-dimensional input.

2.2.6 Model-fitting analysis

For the comparison and analysis of the composite models used to find the best fit to the observed data, we computed Akaike and Bayesian information criterion (AIC and BIC, respectively) metrics. As the IOC formulation provides a deterministic model, residual sum of squares (RSS) was used for the computations. Note that, this is only valid under the assumption that the model errors are independent and identically distributed according to a normal distribution. The Shapiro-Wilk test was used to test the normality assumption on the error, and the null-hypothesis was not rejected in any of the cases. The RSS values for each model were computed by the dynamic time warping algorithm.

2.2.7 Experimental task

During the experiment, participants are required to sit in front of a vertically placed plastic board and perform reaching tasks with their right arm. Nine target areas along with one reference point are marked on the board surface as squares with the side length set to 5 cm. Distance between each target area can be seen in Fig. 2.4b. The sitting position of the participant is determined by setting the line between the reference point and the center of the shoulder joint vertical to the board surface, and the distance is selected as 80% of the total arm length (see Fig. 2.1a).

Every participant is asked to reach nine target areas from nine different initial arm postures. While reaching, the participant should hold a fist and use the frontal surface of the fist to touch the target area, in order to reduce the noise caused by finger movements. Before the recording, the initial arm configuration is determined by measuring the joint angles with a protractor. The nine selected initial postures are displayed in Fig. 2.4. After the measurement, a set of reference tools is utilized to designate the corresponding initial posture. This set of tools consists of two bars with adjustable lengths and positions. The end points of the bars indicate the positions of the elbow and wrist joints for the given posture. The reference tools are placed in appropriate configuration so that during the reaching motion they do not block any potential motion trajectory. The participants are given the following instructions. First, to avoid the decision-making process of target selection, the subjects need to reach the nine target areas in a fixed order as from target one to target nine. Second, the participants should put their arm in the given initial posture as accurate as possible before executing the follow-up reaching tasks. Third, in order to minimize the influence of locating target during the movement, the participants should look at the target area before performing the reaching motion. Fourth, the participants are asked to avoid using the internal/external rotation of the shoulder joint, which is ignored in our arm model. In addition, no instruction about the reaching speed is given, in order to ensure the recorded reaching motions are natural movements.

All participants are trained before the recording to get familiar with the experimental setup and the tasks. If any unintended motion is detected during the experiment, corresponding tasks are executed again. The participants are given enough rest time to avoid fatigue. Each reaching task is performed three times, thus a total of 3645 (9 initial postures $\times$ 9 targets $\times$ 3 times $\times$ 15 subjects) trajectories are recorded. For the IOC calculations, one average trajectory is obtained through the corresponding three trials, hence 1215 IOC calculations are performed in total.

2.2.8 Data collection

A total of fifteen subjects (11 males, age: 27 ± 4; weight: 67 ± 9 kg, height: 172 ± 5 cm) conducted the experiments and gave their written informed consent for their participation. All participants are right-handed with normal vision ability. None of them received any information about the purpose or hypotheses of the experiment. The study was approved by the ethics committee of the Technical University of Munich School of Medicine. The procedures were carried out in accordance with the relevant guidelines and regulations. The trajectories are recorded by using the multicamera motion capture system Qualysis [81] with eight cameras at a frequency of 250 Hz. Nine tracking markers are placed on the subject: seven of them are attached to the right arm (three on the shoulder, two on the elbow, two on the wrist) and two to the torso. The position trajectories are smoothed with the built-in filter function of Qualysis. The movement onset and terminal time of each reaching

Posture	$q_{1s}(^\circ)$	$q_{2s}(^\circ)$	$q_{3s}(^\circ)$	Target	$q_{1e}(^\circ)$	$q_{2e}(^\circ)$	$q_{3e}(^\circ)$
P1	11.0 ± 5.0	6.6 ± 4.7	12.7 ± 3.5	T1	-10.3 ± 3.6	82.0 ± 6.9	23.7 ± 7.2
P2	11.2 ± 5.7	8.8 ± 9.5	33.4 ± 6.5	T2	7.2 ± 4.2	80.5 ± 7.6	29.4 ± 10.2
P3	11.9 ± 3.7	31.9 ± 5.8	13.2 ± 3.7	T3	27.5 ± 4.7	83.5 ± 7.0	24.5 ± 9.1
P4	-22.3 ± 5.2	12.5 ± 5.2	14.1 ± 3.9	T4	-10.2 ± 3.7	66.6 ± 6.1	19.0 ± 5.7
P5	-23.5 ± 5.7	15.8 ± 6.4	37.9 ± 7.5	T5	6.7 ± 3.8	64.3 ± 7.0	24.0 ± 8.7
P6	-22.9 ± 5.3	37.3 ± 7.9	16.1 ± 5.1	T6	26.6 ± 4.3	66.5 ± 7.0	20.4 ± 8.3
P7	42.2 ± 6.2	7.0 ± 7.4	12.3 ± 4.6	T7	-9.1 ± 3.4	55.0 ± 5.0	14.4 ± 4.5
P8	40.2 ± 4.4	7.1 ± 5.3	35.4 ± 5.6	T8	5.2 ± 3.4	53.9 ± 5.4	14.9 ± 5.4
P9	35.4 ± 5.1	36.1 ± 5.6	10.1 ± 5.8	T9	23.6 ± 3.6	54.5 ± 5.9	13.0 ± 5.4

Table 2.2: Actual initial and final joint angle configurations calculated among all 15 subject's data. Mean values and the corresponding standard derivations are presented. P1 to P9 are nine different initial arm postures. T1 to T9 indicate nine target areas. (q_{1s}, q_{2s}, q_{3s}) are the three initial joint angles while (q_{1e}, q_{2e}, q_{3e}) are the final joint angles for each target area.

task are selected as the instant at the 5% of the peak velocity of the hand. These 3D position trajectories are utilized in the IOC calculations. The joint angle trajectories are calculated through inverse kinematic technique.

2.3 Results

We have three main results, which support our hypothesis on hierarchical control of multiple internal models within the CNS. First, there are discernible behavioral differences, such as velocity profiles and reaching time, depending on the reaching motion type. Such differences in movement features signal the existence of different motion control structures. Second, we identified that the combination of different optimal controllers explains a broad range of human reaching motions in 3D free space better than a single one. As each optimal control problem (OCP) with a given cost function defines a single control mechanism and thus a single inverse internal model, having several controllers supports the multiple inverse internal models component of the multiple internal models (MIMs) hypothesis. Finally, the combination of those internal models changes depending on the initial and final arm postures. This dependency, we claim, is resolved by the central nervous system (CNS) by a hierarchical control mechanism to actively coordinate the contribution of those independent internal models. To this end, we propose a metric, that is a function of motion features, as a higher level control parameter since it can describe the dependence of model contributions to the motion parameters.

2.3.1 Reaching motion behavior analysis

Initial and final postures The experimental task is demonstrated in Fig. 2.1. Every subject starts the reaching tasks from nine different initial arm postures and reaches nine target areas (9 initial postures × 9 targets = 81 reaching tasks). The initial postures are selected as different combinations of three joint angles, where q_1 has three values corresponding to the upper, middle, down arm postures, q_2 and q_3 have two values which stand for the zero rotation and a rotation approximately has $40°$, respectively. The actual initial joint angles averaged from all subjects' data are listed in Table 2.2. The results show that for the same reaching task, participants share similar initial joint angle configurations which satisfies our experimental design. An illustration of the initial arm postures and the corresponding reach endpoints on the board surface for a representative subject (subject 3) is presented in Fig. 2.4.

Since our focus is on stereotypical arm reaching characteristics that are observed on the trajectory level, and the corresponding feedforward controllers that might result in such trajectories, the experiment conducted is designed to allow the subjects to reach their hands to a region rather than a single point target. To investigate whether the initial arm posture has influence on the selection of reach endpoints and final arm postures, one-way repeated measures ANOVAs (using SPSS statistics) are conducted separately for each target area. Corresponding p-values are presented in Table 2.3. The results show that, no significant difference of the initial arm posture conditions on the Cartesian coordinates of the reach endpoints x, y, z is found ($p_{x,y,z} > 0.05$ for all target areas). This suggests that, the selection of reach endpoints, which exhibit elliptical distributions inside each target area, is unaffected by the initial arm postures.

However, due to the redundancy of arm kinematics, given only the target area might result in numerous possible final arm joint angle configurations which can satisfy the task requirements. The results of one-way repeated measures ANOVAs on the three final joint angles also support this possibility. The redundancy mainly happens in the ulnar/radial rotation of the shoulder joint q_{2e} and the extension/flexion rotation of the elbow joint q_{3e}, which are found to be influenced by the initial arm postures ($p_{q_{2e}} < 0.05$ except target three, $p_{3e} < 0.05$ except target two, three and six), whereas the elevation/depression rotation of the shoulder joint q_{1e} remains irrelevant to the initial arm postures ($p_{1e} > 0.05$ for all target areas).

These results may also be due to the fact that, in our experimental design, participants are asked to avoid using the internal/external rotation of the shoulder joint during the reaching movements, and thus the height of reach endpoints can only be determined by q_1. By specifying the height of target areas, the choice of q_{1e} is quite limited, so that no significant influence of the initial arm posture on q_{1e} can be found. In contrast, same target area can be reached by different combinations of q_2 and q_3 which exhibit similar hand positions in x, y direction. As a consequence, the initial joint angle configurations affect the selection of final arm postures, as subjects show different tendencies in this selection. Some of them prefer a straight arm posture with low rotation in the elbow joint, while others would like to keep the rotation of the elbow joint the same as the initial configuration. The averaged final arm postures among all subjects' data are also presented in Table 2.2.

Movement duration The movement durations averaged among all subjects' data of reaching tasks starting from initial arm posture one are presented in Table 2.4. The full table of all 81 tasks is given in the supplementary material. We first investigate the relationship between the

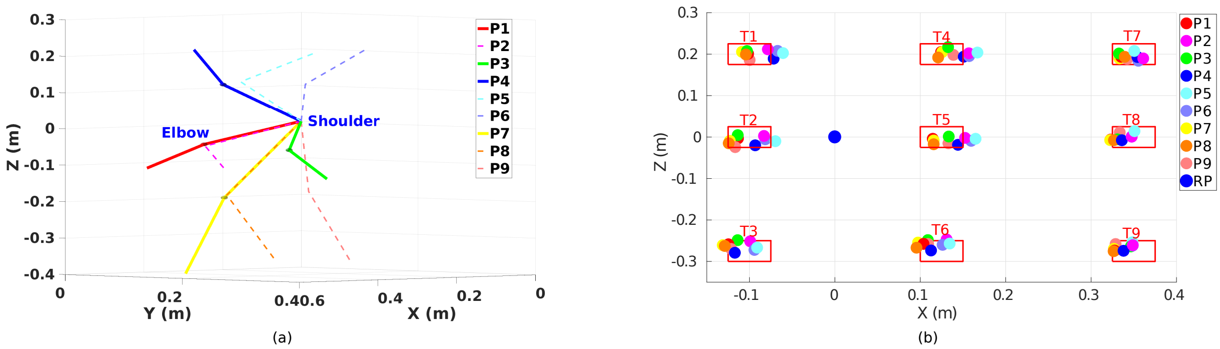


Figure 2.4: Example of the experimental design. (a) An illustration of nine initial arm postures. P1 to P3 are three postures in the middle level, while P4 to P6 and P7 to P9 are postures in the upper and lower level, respectively. Differences between P1 and P2, P4 and P5, P7 and P8 are that q_3 increases while q_2 remains constant, while between P1 and P3, P4 and P6, P7 and P9, q_3 remains constant and q_2 increases. (b) Positions of the nine target areas and the reference point (RP) on the board surface. Target areas are squares with side length equals to 5 cm. An example of the reach endpoints for subject 3 is also presented. P1 to P9 indicate the initial arm posture for each corresponding reach endpoint.

movement duration and the target distance which is defined as the distance between the initial hand position and the reach endpoint. The distribution of target distances and movement durations of all subjects' data is presented in Fig. 2.5a. The result of correlation analysis (using MATLAB *corrcoef* function) indicates that there is a weak correlation between target distance and movement duration ($r = 0.245$, $p < 0.001$) among all the data. However, different behaviors are observed among subjects. Two representative subjects, one with strong correlation (subject 3, in Fig. 2.5b, $r = 0.815$, $p < 0.001$) and another without correlation (subject 8, in Fig. 2.5c, $r = -0.069$, $p = 0.543$), are presented. This indicates that, different subjects may have different motor control strategies in their natural reaching movements, i.e. some prefer a task-dependent movement duration while others adapt the velocity so that the durations of different reaching tasks are on the same level. Individual analysis shows that most of the subjects adapt their velocities with respect to the target distance (see supplementary material), hence in general the observed correlation is weak.

In addition, two-way repeated measures ANOVAs are performed to investigate whether the initial arm posture and target area conditions have influence on the movement duration. The results show significant differences both in target area conditions ($F(8, 112) = 18.5$, $p < 0.001$) and the interaction between target area and initial posture conditions ($F(64, 896) = 4.4$, $p < 0.001$). However, the initial posture conditions exhibit no significant difference ($p = 0.084$). This suggests that, the movement duration is partially affected by the target distance, but in the meantime is also influenced by other reaching motion parameters, e.g. arm configurations.

Kinematic features All subjects exhibit similar kinematic features during the reaching motions. The hand velocity and angular velocity profiles are bell-shaped and demonstrate classical patterns in point-to-point reaching tasks. A partial table of the kinematic features averaged among all subjects is presented in Table 2.4 (see supplementary material for full data). Typical kinematic features of a representative subject (subject 3) are presented in Fig. 2.6. To

Target	p-values					
	x	y	z	q_{1e}	q_{2e}	q_{3e}
T1	0.212	0.803	0.999	0.902	0.041	0.003
T2	0.280	0.793	0.999	0.938	0.041	0.066
T3	0.324	0.717	0.999	0.731	0.053	0.057
T4	0.422	0.761	0.999	0.654	0.004	0.001
T5	0.619	0.556	0.999	0.946	0.004	0.043
T6	0.609	0.447	0.999	0.825	0.017	0.071
T7	0.591	0.738	0.999	0.105	0.007	0.007
T8	0.624	0.496	0.999	0.474	0.004	0.003
T9	0.634	0.349	0.999	0.394	0.009	0.013

Table 2.3: One-way repeated measures ANOVAs of the initial arm posture conditions on the Cartesian coordinates of the reach endpoints x, y, z and final arm joint angle configurations (q_{1e}, q_{2e}, q_{3e}) separately for each target area. Due to the space limitation, only p-values are given here.

further investigate the influence of initial arm posture and target area conditions on the kinematic features, two-way repeated measures ANOVAs are conducted on the average hand velocity and the peak hand velocity. Significant differences are found in the initial posture conditions ($F_{average}(8, 112) = 14.3$, $p_{average} < 0.001$; $F_{peak}(8, 112) = 9.5$, $p_{peak} < 0.001$), the target area conditions ($F_{average}(8, 112) = 107.3$, $p_{average} < 0.001$; $F_{peak}(8, 112) = 87.1$, $p_{peak} < 0.001$) and the interaction between initial posture and target area conditions ($F_{average}(64, 896) = 31.1$, $p_{average} < 0.001$; $F_{peak}(64, 896) = 37.9$, $p_{peak} < 0.001$).

Considering the target distance as another parameter of the reaching task, similar results are also found in the correlation analysis between the target distance and the velocities. As explained in the movement duration part, subjects would adapt their reaching velocity according to the target distance, hence relative stronger correlations between the velocities and the target distance are found ($r_{average} = 0.694$, $p_{average} < 0.001$; $r_{peak} = 0.654$, $p_{peak} < 0.001$). Corresponding distributions of the same two subjects are presented in Fig. 2.5. It can be observed that, both subjects demonstrate good correlations between the velocities and the target distance (Fig. 2.5e, $r_{average} = 0.914$, $p_{average} < 0.001$, Fig. 2.5h, $r_{peak} = 0.908$, $p_{peak} < 0.001$ for subject 3; Fig. 2.5f, $r_{average} = 0.864$, $p_{average} < 0.001$, Fig. 2.5i, $r_{peak} = 0.786$, $p_{peak} < 0.001$ for subject 8). Note that, unlike most of the subjects, the movement durations of subject 3 do not decrease even though the dependency is reported between the velocity and the distance. This may be due to a different path planning strategy adapted by the subject.

All these adaptations, including the final arm postures, velocity profiles and hand trajectories, suggest that the reaching motion behavior changes depending on the reaching task. Thus, it is possible that the CNS also changes its motor control strategy based on the reaching motion parameters. To further investigate the underlying control strategy of reaching motions, an analysis based on the inverse optimal control results is presented in the following sections.

Target	Movement duration (s)	Target distance (m)	Average hand velocity (m/s)	Peak hand velocity (m/s)
T1	1.73 ± 0.36	0.813 ± 0.044	0.53 ± 0.11	1.00 ± 0.23
T2	1.63 ± 0.35	0.775 ± 0.051	0.54 ± 0.12	1.04 ± 0.30
T3	1.70 ± 0.37	0.779 ± 0.058	0.52 ± 0.12	0.98 ± 0.26
T4	1.57 ± 0.32	0.689 ± 0.043	0.48 ± 0.10	0.91 ± 0.22
T5	1.49 ± 0.30	0.645 ± 0.050	0.48 ± 0.11	0.88 ± 0.22
T6	1.57 ± 0.37	0.650 ± 0.060	0.47 ± 0.12	0.86 ± 0.22
T7	1.52 ± 0.30	0.574 ± 0.037	0.41 ± 0.09	0.76 ± 0.18
T8	1.44 ± 0.25	0.535 ± 0.045	0.40 ± 0.08	0.72 ± 0.18
T9	1.47 ± 0.25	0.533 ± 0.052	0.39 ± 0.06	0.72 ± 0.19

Table 2.4: Movement duration and kinematic features of reaching tasks starting from initial arm posture one towards nine target areas. Mean values and standard derivations are presented. The data is averaged among all subject's data.

2.3.2 Identification of multiple internal models

Internal models are neural mechanisms that control a specific property of human movement by imitating the input/output characteristics of the motor apparatus. However, even simple human motions, such as reaching a cup, require a complex system to control the kinematics and dynamics properties of the human arm. To identify the components of such a control mechanism, we first examine the existence of multiple internal models for a broad range of human reaching movements. In our experiment, stereotypical 3D free-space arm reaching task is the focus, and accordingly, in our formulation there is a single forward internal model, i.e. the arm dynamics model, that solves for the system states given the control input, but it is paired with different inverse models. Inverse optimal control results show that the combinations of optimal controllers, i.e. multiple inverse models, result in a better fit than a single optimal control based solutions. Additionally, the contribution of internal models is found to be dependent on the movement type.

Reconstruction errors By means of forward optimal control, motion trajectories are simulated to calculate the reconstruction errors of the solutions with respect to the single model and the composite model control strategies. In the single model cases, the single basic cost functions (hand jerk, joint angle jerk, torque change and energy, see method section) are used individually to predict human motion trajectories for all the reaching tasks we have considered. Then, in the composite model case, by using the optimal composite cost function obtained from the inverse optimal control (IOC) calculations, we simulate the reaching trajectories. Thereafter, the reconstruction errors of those single models and the composite model are compared by computing the distance error between the simulated trajectories and the observations.

Two different kinds of errors are measured through the dynamic time warping, one is the end-effector Cartesian coordinate error, another is the joint angle error. As presented in Fig. 2.7a, the composite model can simulate the reaching motions with smaller end-effector trajectory error (1.2 ± 0.9 cm) compared to all other single models (hand jerk: 3.5 ± 2.0 cm, joint angle jerk:

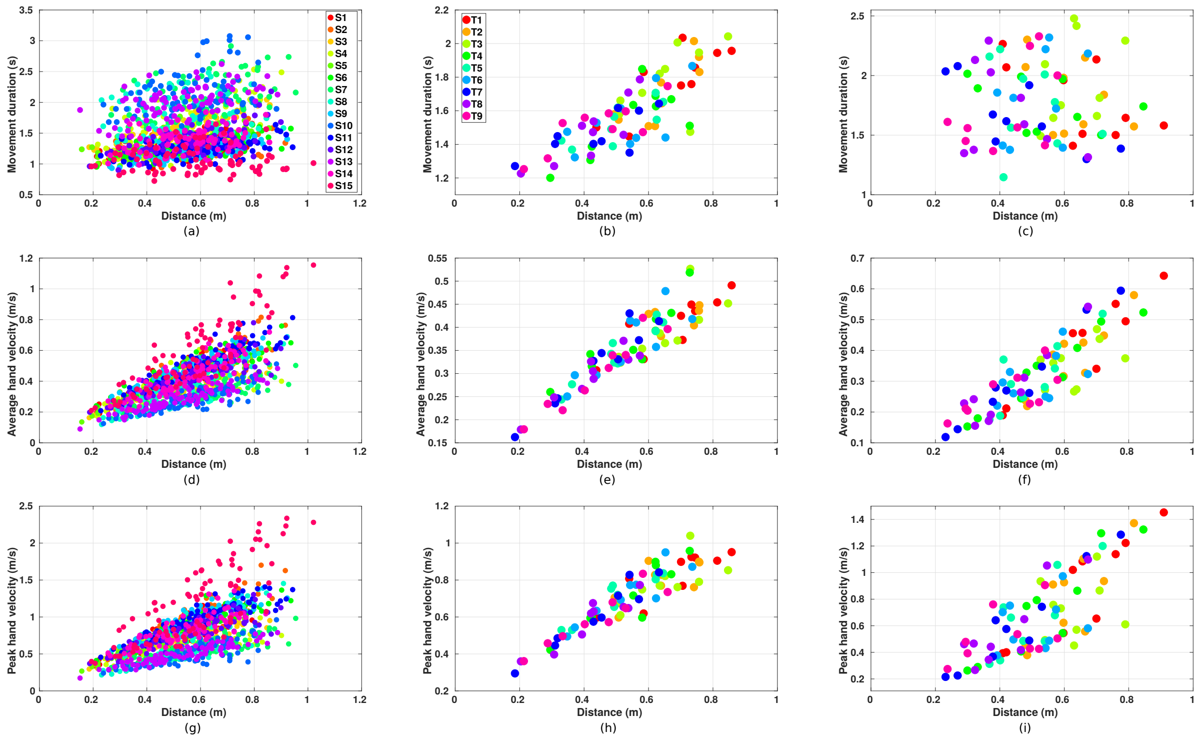


Figure 2.5: Relationship between target distance and movement duration, average hand velocity, peak hand velocity. (a) The distribution of target distance and movement duration for all subjects' data. Points are grouped by 15 subjects as from S1 to S15. (b) The relationship between target distance and movement duration for subject 3. Points are grouped by the target areas for the corresponding reaching tasks. (c) The relationship between target distance and movement duration for subject 8. (d)-(f) The distribution of target distance and average hand velocity for all subjects' data, subject 3, and subject 8, respectively. (g)-(i) The distribution of target distance and peak hand velocity for all subjects' data, subject 3, and subject 8, respectively.

1.9 ± 1.4 cm, torque change: 7.3 ± 3.7 cm, energy: 7.1 ± 4.2 cm). One-way repeated measures ANOVAs are performed separately on different single models and the composite model. The results demonstrate significant differences between the composite model and each other single model (hand jerk vs composite: $F(1, 1200) = 2080.9$, $p < 0.001$; joint angle jerk vs composite: $F(1, 1200) = 863.2$, $p < 0.001$; torque change vs composite: $F(1, 1200) = 3641.0$, $p < 0.001$; energy vs composite: $F(1, 1200) = 2732.8$, $p < 0.001$). Similar results can also be observed in the joint angle errors, which are illustrated in Fig. 2.7b. One-way repeated measures ANOVAs also show significant differences between the composite model and each single model (hand jerk vs composite: $F(1, 1200) = 1723.4$, $p < 0.001$; joint angle jerk vs composite: $F(1, 1200) = 111.0$, $p < 0.001$; torque change vs composite: $F(1, 1200) = 4170.8$, $p < 0.001$; energy vs composite: $F(1, 1200) = 3175.3$, $p < 0.001$). Even the joint angle trajectory is not considered in the presented IOC formulation (see method section), the composite model still exhibits better performance in describing the angular trajectories with an average error of $3.4 \pm 3.1°$ compared to other single models (hand jerk: $12.9 \pm 8.4°$, joint angle jerk: $3.9 \pm 3.2°$, torque change: $14.6 \pm 7.1°$, energy: $11.1 \pm 5.8°$). The smaller reconstruction errors of the composite model suggest that, rather than a single model, the CNS may use multiple models to control its reaching motions.

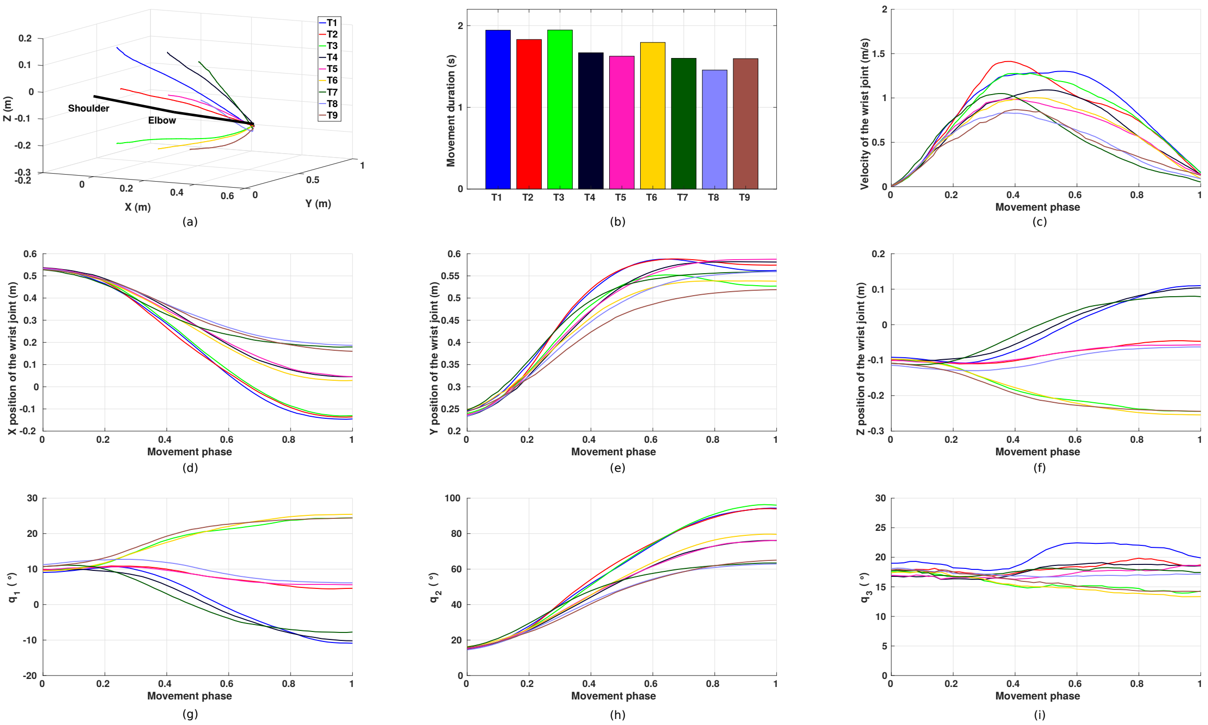


Figure 2.6: Examples of nine reaching tasks from initial arm posture one for subject 3. The subplots show (a) end-effector trajectories, (b) movement durations, (c) velocity of the wrist joint, where the trajectories are presented with their movement phase, which starts at 0 and ends at 1, (d)-(f) x, y, z coordinates of the wrist joint, (g)-(i) trajectories of the three joint angles.

For the model fitting analysis, Akaike and Bayesian information criterion (AIC and BIC, respectively) were computed for the models with 2, 3, and 4-cost function combinations. A representative subject's data is used to compute the composite models for each combination. The composite model with the 2-cost functions used joint-angle-jerk (kinematics) and torque-change (dynamics), whereas for the 3-cost function model energy (dynamics) cost was also included together with the previous two. The resulting AIC and BIC comparisons favored the 4-cost function model compared to the other two models. The end-effector Cartesian coordinate error was also computed to compare these models in terms of their accuracy in representing the human behavior. This analysis demonstrated a statistically significant difference (for both 2- vs. 4-cost function model, and 3- vs. 4-cost function model, $p < 0.001$), favoring the 4-cost function composite model (see supplementary material).

Contribution of models Having identified that the composite model as the controller describes human reaching motion behaviors better than the single models, we also investigate whether the combination of different basic models changes depending on the reaching tasks. Two-way repeated measures ANOVAs are performed separately on the contribution of hand jerk α_{HJ}, joint angle jerk α_{JJ}, torque change α_{TC} and energy α_{Enr} controller models with respect to the initial arm posture and target area conditions. The results show significant differences both in the target area conditions for all four base models ($F_{HJ}(8, 112) = 6.8$, $p_{HJ} < 0.001$; $F_{JJ}(8, 112) = 3.9$, $p_{JJ} < 0.001$; $F_{TC}(8, 112) = 10.6$, $p_{TC} < 0.001$; $F_{Enr}(8, 112) = 15.0$, $p_{Enr} < 0.001$), as well as

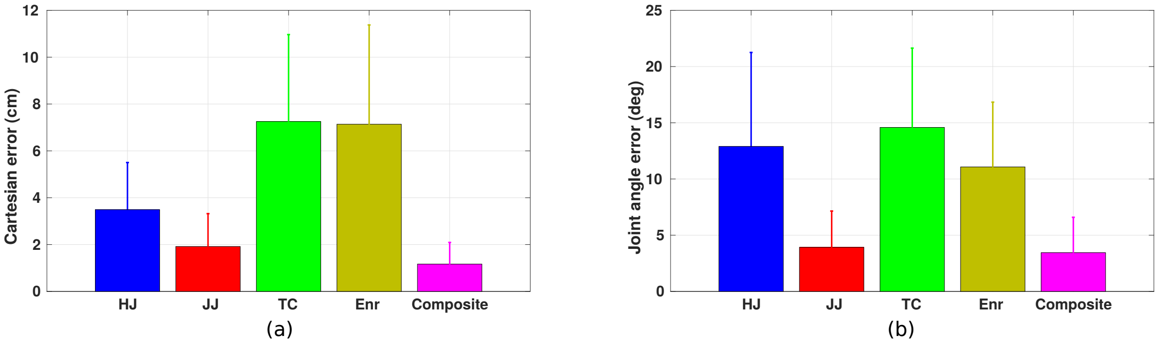


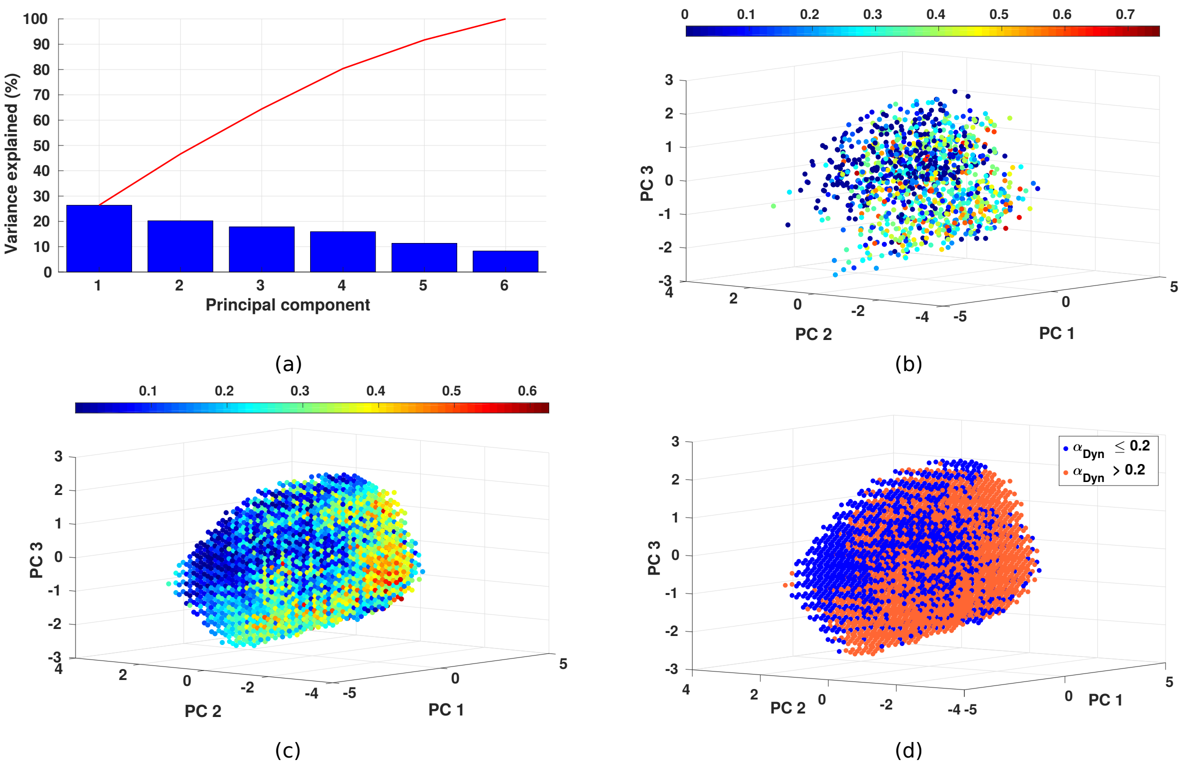
Figure 2.7: Reconstruction errors normalized with the number of data points. (a) The Cartesian errors of reconstructions by using the minimum hand jerk (HJ), minimum joint angle jerk (JJ), minimum torque change (TC), minimum energy (Enr) and the composite model (Composite), respectively. (b) The joint angle errors of reconstructions.

in the interaction between the initial arm posture and target area conditions ($F_{HJ}(64, 896) = 2.6$, $p_{HJ} < 0.001$; $F_{JJ}(64, 896) = 1.5, p_{JJ} < 0.01$; $F_{TC}(64, 896) = 1.4, p_{TC} < 0.05$; $F_{Enr}(64, 896) = 4.4$, $p_{Enr} < 0.001$). For the initial arm posture conditions, except the contribution of joint angle jerk controller ($p = 0.199$), significant differences are found for other three basic models ($F_{HJ}(8, 112) = 17.4$, $p_{HJ} < 0.001$; $F_{TC}(8, 112) = 5.3$, $p_{TC} < 0.001$; $F_{Enr}(8, 112) = 12.6$, $p_{Enr} < 0.001$). The reason for α_{JJ} being not affected by the initial posture conditions may be that α_{JJ} is more consistent in different reaching tasks. Particularly, the joint angle jerk controller plays a dominant role in the motor control of reaching motions, which has an average value of $\alpha_{JJ} = 0.45 \pm 0.14$, while other three basic controllers have smaller contributions ($\alpha_{HJ} = 0.31 \pm 0.18; \alpha_{TC} = 0.09 \pm 0.12; \alpha_{Enr} = 0.15 \pm 0.16$). This is supported by the reconstruction errors exhibited in Fig. 2.7, where the joint angle jerk model can simulate the trajectories with smaller errors compared to other three models, and the results are also similar to the composite model. This can be explained by the fact that, the recorded reaching motions mostly have bell-shaped velocity profiles, while the optimal control solution to the joint angle jerk cost function also produces bell-shaped velocities, hence the joint angle jerk based controller can represent the reaching motions with better performance.

In addition, we also group the four base controllers into two types: the dynamics and the kinematics related control models (see method section). Two-way repeated measures ANOVAs also indicate significant differences in the contribution of dynamics based controllers α_{Dyn} with respect to the initial arm postures conditions ($F(8, 112) = 15.1$, $p < 0.001$), the target area conditions ($F(8, 112) = 11.7$, $p < 0.001$) and the interaction between the initial arm posture and target area conditions ($F(64, 896) = 3.03, p < 0.001$). The contribution of kinematics based controllers α_{Kin} demonstrates the same results since the sum of α_{Dyn} and α_{Kin} equals to one. To simplify the analysis and reduce the dimension of possible variables, we focus on analyzing the contribution of dynamics based controllers in the following section.

2.3.3 Hierarchical control of internal models

As the contribution of dynamics based controllers is found to be influenced by the initial arm posture and target area conditions, we further investigate whether this influence follows a certain



(a)

(b)

(c)

(d)

Figure 2.8: Results of principal component analysis (PCA). (a) Variance explained by each principal component. Red line indicates the total explained variances summed up from corresponding principal components. (b) Original distribution of α_{Dyn} with respect to the first three principal components by using all subjects' data. PC 1 to PC 3 indicate the first three principal components, the color of the points stands for the contribution of dynamics based controllers. (c) The distribution of α_{Dyn} after the interpolation by using all subjects' data. (d) Two groups of the points divided from Fig. 2.8c by using the threshold as $\alpha_{Dyn} = 0.2$.

criterion. For each given reaching task, the motion parameters can be defined as the initial arm posture with three initial arm joint angles $\boldsymbol{q}_s = (q_{1s}, q_{2s}, q_{3s})$ and the final arm posture (represents target area) with the corresponding final arm joint angles $\boldsymbol{q}_e = (q_{1e}, q_{2e}, q_{3e})$. The purpose of this section is to identify if there is a relationship between the contribution of dynamics based controllers and the motion parameters.

Principal component analysis As each reaching task is defined through six joint angles, due to this high dimensionality, it is not straightforward to determine whether the relationship between the motion parameters and the contribution of dynamics related controllers exists. Hence, we first use the principal component analysis (PCA) to transfer the original motion parameters to the principal components (PCs) by using all subjects' data. For visualization purposes, we select the first three PCs, which only explain 64% of the variance (Fig. 2.8a), as a new motion parameter representation for each reaching task, and then investigate the distribution of α_{Dyn} with respect to these PCs.

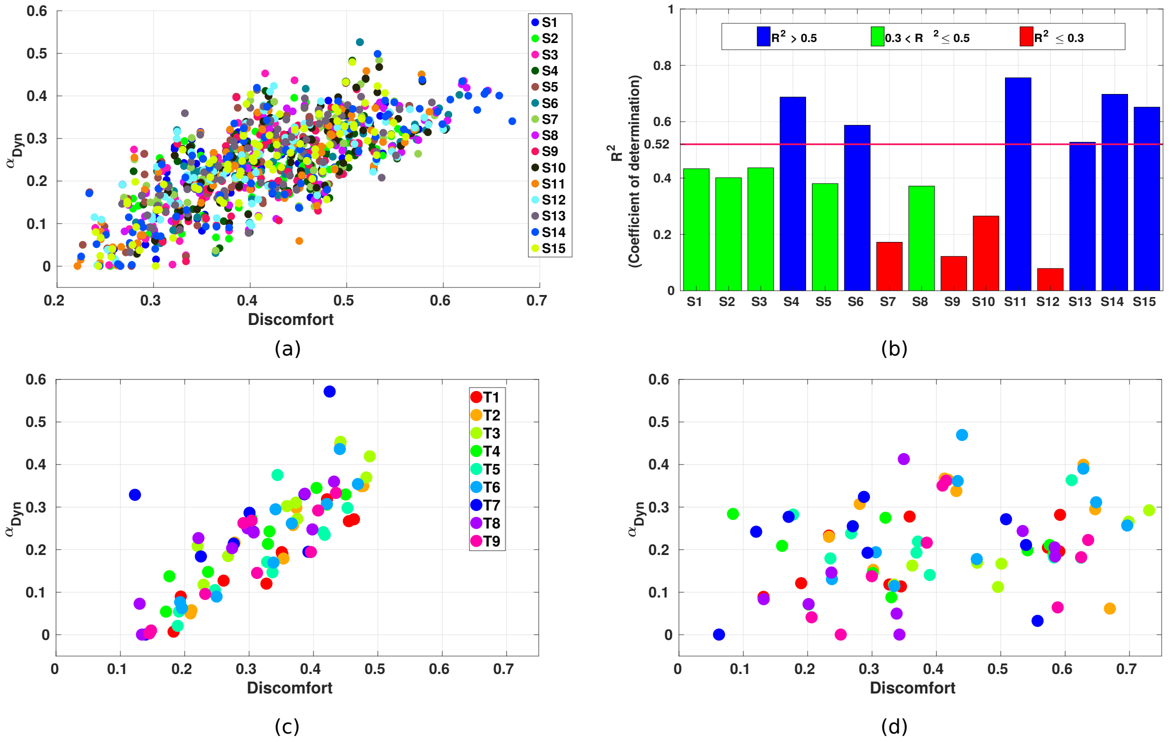
The original IOC results of all subjects' data are illustrated in Fig. 2.8b, where the x, y, z coordinates of each data point are the corresponding values of the first three PCs, and the color indicates the value of α_{Dyn}. Due to the infeasibility of the global optimum in the IOC formulation [59], the results of α_{Dyn} prone to noise. Besides, the insufficient data amount (9 initial postures × 9 targets × 15 subjects = 1215 motions in total) also limits the determination of the relationship between PCs and α_{Dyn}. In order to overcome this issue and have a better visualization, the original data is interpolated with a 3D interpolation method (using MATLAB *griddata* function) and the corresponding results are presented in Fig. 2.8c. It is observed from the given perspective of the figure that, upper-left points on the point cloud usually have smaller α_{Dyn}, while the points with high values of α_{Dyn} mostly appear in the opposite region. A more clear illustration is presented in Fig. 2.8d, where the data points are divided into two groups with the threshold as $\alpha_{Dyn} = 0.2$. Hence through visual inspection, even it only represents a subspace of 6 PCs, a trend that indicates a dependence between the motion parameters of the reaching task and the distribution of α_{Dyn} is observed. In addition, as the sum of α_{Kin} and α_{Dyn} is one, the observed distribution suggests a trade-off between the kinematics and dynamics related controllers based on the type of the reaching task.

Note that, the first three PCs account only for 64% of the total variance. Thus, in order to explain the distribution more accurately, we opt for using all six motion parameters to identify a criterion, which we refer to as *discomfort* metric, in order to describe the distribution of α_{Dyn}.

Discomfort metric As found in a recent study, for reaching motions the contribution of interaction torque to net torque changes depending on the load on the arm [82]. This result hints that the distribution of α_{Dyn} may be related to the musculoskeletal loading of the arm during the movements. Considering the musculoskeletal loading as the criterion to describe the discomfort of the reaching motions [83], [84], the fully stretched down arm posture can be treated as the most comfortable posture. Then the more rotations the arm requires to execute the reaching task from the fully stretched down posture, the more uncomfortable the motion is. Based on this, we propose a discomfort metric (see method section) to explain the distribution of α_{Dyn}.

We first investigate the overall behavior by using all subjects' data. Due to the same noise issue explained in previous part, during the calculation of discomfort metric, the data is smoothed by using LOESS [85] (LOcal regrESSion), with the span as 1% of the total data points (using MATLAB *smooth* function). By assuming a linear relationship between the motion related parameters and the controller contribution, we computed a linear regression model, which can arguably explain the variance in the contribution of dynamics based controllers ($R^2 = 0.52$, see Fig. 2.9a). This suggests that, for the reaching motions with high discomfort values, i.e. more rotations are required and higher musculoskeletal loading is generated, the contribution of dynamics based controllers increases while the contribution of kinematics based controllers decreases. Hence, there is a trade-off between the dynamics and kinematics related controllers depending on the reaching tasks. This trade-off implies that the CNS may utilize a hierarchical control structure. As such, depending on the motion parameters, human motor control first regulates the contribution of each internal model to form a composite model by using a higher level controller. Then, the reaching motion is controlled with respect to this task specific composite model.

To investigate this idea further, we also calculate the discomfort values separately by using individual subject's data (smoothed as well with span = 5% due to less amount of data). Corresponding values of the coefficients of determination between the discomfort values and α_{Dyn} for each subject



(a)

(b)

(c)

(d)

Figure 2.9: Results of the discomfort metric. (a) The distribution of α_{Dyn} and the discomfort values calculated from all subjects' data. The points are grouped by different subjects as from S1 to S15. (b) The coefficient of determination between α_{Dyn} and the discomfort values calculated by using each subject's data separately. The line indicates the overall coefficient of determination from all subjects' data presented in Fig. 2.9a ($R^2 = 0.52$). (c)-(d) The distribution of α_{Dyn} and the discomfort values for subject 15 and subject 12, respectively. The points are grouped by the target areas for the corresponding reaching tasks.

are presented in Fig. 2.9b. Among 15 subjects, 3 different clusters are observed (Fig. 2.9b). For the first group, the variation in their motion behavior can be explained by our discomfort metric ($R^2 > 0.52$) better than others. The second group consists of people whose motion behavior can still be accounted for with the proposed model ($0.3 < R^2 < 0.5$). However, for the third group, the model we proposed fails to capture the variation of motion characteristics ($R^2 < 0.3$). Two representative subjects are given in Fig. 2.9c (subject 15) and Fig. 2.9d (subject 12), where one subject's data shows similar trade-off compared to the overall results and the other one demonstrates a discrepancy. Possible reasons for the outliers might be, first, the simplicity of our model as the discomfort values we proposed is a linear combination of joint angles. However, the actual metric utilized by the higher level controller might have a different structure, e.g. nonlinear combination of motion parameters. In addition, the distribution α_{Dyn} possibly depends on other motion parameters besides the joint angles. The second reason could be the fact that subjects show interpersonal variance in their behaviors, e.g. the variance as we observed in the movement duration analysis. Hence, due to the interpersonal variance, it might be difficult to find a single control structure which can explain every subject's motions.

2.4 Discussion

In this study, we look for the evidence to support multiple internal models hypothesis for a broad range of free space reaching motions. There are two possibilities to consider for the use of different internal models. Either internal models act independently, or they are blended together to achieve a single goal-oriented task. We take the latter claim as our foundation and lay out the analysis of human reaching motion as a bi-level optimization problem. Here, the lower level is treated as a standard optimal control problem (OCP), where each OCP with a distinct cost function corresponds to an inverse internal model, and the upper level is utilized as a parameter identification process to compute the contribution of each inverse internal model to the observed motion behavior. In this way, we identify not only the change in the contribution level of different models, but also how this change indicates a trade-off on the utilization of kinematics and dynamics related biomechanical properties, and thus controllers, depending on the initial arm posture and target position. In essence, our analysis suggests a hierarchical control structure, where a higher level controller is responsible for the regulation of the lower level independent internal models.

Optimal control problem as an inverse internal model Internal models in sensorimotor integration are empirical constructs to describe learning and control of motor behavior. Even if one assumes that the central nervous system (CNS) uses internal models, forward models predicting the arm motion for a specified control has to be distinguished from inverse models that "can calculate necessary feedforward motor commands from desired trajectory information", as Kawato emphasizes [86]. Experimental results supporting both types of models are discussed in literature, cf. [87], [88]. As we look into stereotypical motions, one central idea discussed in literature is that, after an initial learning phase, such motions are approximately optimal with respect to an unknown criterion and thus several of optimal control models with different cost functions have been proposed [11], [32], [33]. In that regard, optimal control theory provides a mathematical framework to describe learning control processes in biological systems [44], [89]. Solving an OCP is to determine *what to do*, i.e. the optimal control signal, in order to achieve certain goals optimally, defined by the cost function, with the given dynamics of the system and a set of constraints. In essence, each OCP with a distinct cost function specifies a unique controller for the movement. This optimal controller -among infinitely many possible controllers- can be considered as an inverse internal model representation within the CNS [18], [48], [60], [86]. Given our formulation and results, human arm reaching motions can be explained as the outcome of a combination of several such optimal controllers and thus multiple inverse internal models. This supports the multiple internal models hypothesis for human arm reaching tasks. Furthermore, such a controller not only finds the optimal control sequence, but also acts as a planner by computing the optimal trajectory. However, note that the control law found by inverse optimal control (IOC) provides an open-loop model, hence there is no feedback consideration. However, as we focus only on stereotypical motions, the effects of sensory update are considered minimal [16].

Single vs. multiple internal models Both multiple internal models (MIMs) hypothesis and the recently introduced IOC framework rely on the fact that a combination of controllers describe human motor behavior better than single models. One fundamental reasoning for the coexistence of multiple controllers is the CNS's striking efficiency in learning to control human motions by adapting to changes in the environmental conditions. Each controller is suitable and responsible for one or a small set of motor behavior and depending on the occasion, some of the controllers are appropriately chosen to generate the required motor command. As a result, frameworks with MIMs comprising both forward and inverse models have been introduced, e.g. [17], [45], [86]. Similarly for (inverse) optimal control framework, since a single model describes a controller for a specific feature of motor behavior, e.g. smoothness or effort, a composite of control models are necessary to capture diverse human movement characteristics simultaneously [57], [58], [65]. In that sense, our composite OCP formulation can be regarded as a framework for describing and supporting MIMs hypothesis from optimal control point of view. Each OCP is associated with a single cost function, and thus provides a modular structure. Such a modular structure is also in accordance with the motivation behind MIMs. Especially, considering the variety of interaction cases a human may face with (e.g. depending on the object being interacted, the environmental effects, and their possible combinations), such a modular structure allows for providing appropriate motor commands effectively rather than a single controller that needs to take all the external signals into account and solves for control commands on each occasion.

Forward and inverse internal models Considering that internal models mimic the transformations between system states, motor commands, and sensory signals, an OCP can be regarded as an inverse internal model [44], [60]. In essence, for the stereotypical movements we investigate, solution of an OCP provides the necessary control signals to carry out the reaching task, which is consistent with the inverse internal model idea proposed by Kawato and Wolpert [17]. Even though the bi-level optimization formulation uses the combination of costs in the upper level program to optimize the weighting factors, the lower level program still solves an OCP, for which each constituent OCP model -each associated with a specific cost function- needs to be satisfied simultaneously. Wolpert and Kawato propose a general model to account for motor control's ability to handle various tasks (e.g. reaching for grasp) concurrently, for which multiple forward and inverse models are paired [17]. As we concentrate on a specific arm reaching task, in our formulation we have a single forward model, i.e. the arm dynamics model that describes how system states evolve given the control input, but it is paired with different inverse models. Hence, the combination of multiple OCPs to control the execution of a single task offers a model-based formulation for the multiple inverse internal models proposed in Wolpert and Kawato [17], which has been treated in later studies as a black-box function approximator [18], [20].

Composition of internal models There have been other studies which focus on multiple models for motor control, learning and adaptation [90]. The focus of these prior work was whether multiple internal models can be learned concurrently [46]–[48] and switched depending on the context [91]–[94], whereas in our work, free-space stereotypical arm reaching movements are assumed to be already learned, and controlled as a combination of multiple internal models and this combination is sought with a systematic approach by the IOC formulation. A similar study by Davidson and Wolpert analyzed whether two internal models could be learned concurrently for gripping task [48]. They proposed a linear combination of objects' weights as an internal model representation, and their results verified that dynamic internal models can be additively combined. However, they also acknowledged the necessity for further studies to verify whether similar compositions could be realized for more complex internal models [48]. In our study, a combination of optimal control models (via a composite cost) is proposed which can be applied on tasks which require position and force control at the same time. In addition, it is verified that the composite model explains human movement behavior better than single models, and this compositeness changes with respect to the task parameters.

Trade-off between dynamics and kinematics In the controller contribution analysis, we observe a trade-off between the usage of dynamics and kinematics based controllers with regard to the task demands. In theory, an OCP with kinematics related cost functions results in trajectories with high smoothness [32], [71], while the optimization of dynamics related costs minimizes the motor control effort for given reaching tasks [11], [73]. It has been acknowledged and verified that motor control involves consideration of different performance measures [44], [58], [82]. In Liu and Todorov, the focus was on closed-loop control behaviors for reaching tasks, and the change in the control strategy during movement was investigated [12]. However, our main focus is on stereotypical motions for which an open-loop control is assumed, similarly to the prior work in this field [57], [58], [65]. Composite models were discovered for planar arm reaching motions in a similar work by Berret et al. [58], however, the combination of different models was not analyzed in depth to explain how and why such compositeness is utilized. A similar result was also reported in a recent work by Vu et al., where the contribution of the interaction torque to the net torque was found to depend on the load on the forearm, and a trade-off between kinematic and kinetic variables was suggested [82]. Besides, for different arm postures, the musculoskeletal load varies and affects the feeling of comfort [83], [84]. This indicates the possibility of the musculoskeletal load as a criterion to describe the trade-off between kinematics and dynamics based models for different reaching tasks. However, an inquiry on how the control models change depending on the task parameters has been neglected. In this work, a discomfort metric is introduced to explain the change in the model contributions with respect to the movement type. In essence, the angular rotations required to execute the reaching task are considered as a representation of the musculoskeletal load. As a result, the trade-off between controllers is revealed by the task demands, and the contribution of dynamics based controllers are found to increase with higher load. For reaching tasks with higher effort requirements, the CNS might be planning the motion depending more on the dynamical effects, while for tasks with low dynamics demands, the smoothness plays a dominant role in the motion planning.

Hierarchical control The adaptive usage of smoothness and effort suggests a hierarchical control structure of the CNS, which can be described as a task-estimation phase and an optimal execution phase. For a given reaching task, the CNS first evaluates the motion with respect to a higher level criterion, e.g. the proposed discomfort metric, then based on the estimation, it controls the contributions of different internal models. This phase can be considered as a task-estimation phase which is similar as the responsibility estimator discussed in the MIMs hypothesis [17], [86]. However, in prior studies [17], [18], the output control signals from multiple inverse models are linearly combined with a responsibility signal, whereas in our model, even though the weights are linear in the cost functions, which is solved in the upper level program of the bi-level optimization, the underlying optimal control problems and thus their outputs, which are solved by the lower level program, are nonlinear in terms of weights and states, especially in the presence of kinematics, and dynamics constraints. In other words, finding the composite model in an inverse optimal control formulation involves solving the optimization problem that simultaneously minimize multiple costs to satisfy each control model constrained on the dynamical system, i.e. a highly nonlinear, high-dimensional and non-convex solution space. Note that, the estimation relies on personal experience and preference, thus different subjects have different attitudes towards the same task, e.g. the task treated as comfortable for one subject might be uncomfortable for the other subject. This would result in an interpersonal discrepancy during the regulation of internal models. After the task-estimation, the CNS plans and executes the motion with respect to the optimality criterion defined as the regulated contributions of different internal models. This execution phase can be treated as the realization of an OCP. The usage of this hierarchical control structure may help explain how humans achieve optimality in different reaching tasks. Due to the high degrees of freedom of human arm, it is not possible to learn the optimal movement behaviors case by case [17]. It is more likely that, the CNS first learns a set of (optimal) controllers, i.e. inverse internal models, for typical movements, and maintains a functional mapping from task requirements to weighting of these models, similar to the responsibility estimator proposed in Wolpert and Kawato [17].

Model fitting We provide an analysis on the model fitting by using Akaike and Bayesian Information Criterion (AIC/BIC) in the Results section. Here, we also touch upon a couple of points regarding the IOC formulation and the models found. Our analysis shows statistically significant difference, on most cases, between the results of the composite model and the rest considering not only the hand trajectory but also the joint angle trajectory (note that, the inverse problem is solved only on the end-effector (hand) trajectory level, i.e. there is no fitting in joint-angle and/or torque space). In addition, a recent study by Vu et al. shows that, even though a single OCP model can match human-like end-effector trajectories, it cannot capture the torque profiles accurately [82]. Hence, it is remarkable that such composite models can predict both kinematics and dynamics related outcomes better than single control models. Additionally, in terms of the fitting quality of the composite model, our bi-level optimization formulation along with a comprehensive greedy search on the parameter space allow for discarding models if necessary, i.e. the weighting factors could be zero. The upper level program (the trust-region based optimizer) in the bi-level formulation provides regularization effect on the weighting parameters, which is commonly applied for inverse optimal control and reinforcement learning problems to prevent overfitting [27], [95].

Limitations There are some limitations in the presented work. First, due to the high nonlinearity of the IOC formulation, the global optimum is usually not available [59]. To address this issue, we solve the same IOC problem several times and use the best local optimum to approximate the global optimum. However, there is no guarantee about the error between the approximated and the real global optimum. This partially explains the observed variance in the controller contribution analysis. Second, we propose a discomfort metric to simulate the higher level controller utilized in the task-estimation phase. When formulating the discomfort values, the final arm postures are assumed to be known. However, during the task-estimation phase, it is reasonable to think that there is a bias between the estimation and the real posture. This variability might be inherent to movement or internal noise, as there is still on-going debate on whether there exists a Donders' like law for arm movements [96]. Hence rather than a deterministic expression, the high level controller may utilize a more flexible strategy to minimize this influence. With the proposed simplistic discomfort metric, we intend to present a proof-of-concept formulation for the existence of the hierarchical control structure. It requires further investigation on how the higher level controller is actually formulated, as besides the initial and final arm postures, it might also include other motion parameters or even subjective factors for the given task.

The experiment conducted is designed to allow the subjects to reach their hands to a region rather than a single point target. Hence, our focus is on stereotypical arm reaching characteristics that are observed on the trajectory level, and the corresponding feedforward controllers that might result in such trajectories. We believe this is still an important aspect of motor control, since in our daily lives there are many cases where we interact with our environment in an open-loop fashion, and also it can be seen as an approximation of non-deterministic control due to the stationary environment as well as the fast and stereotypical nature of the motions we investigated. Even though there is evidence for open-loop control of specific tasks in human motor system [97]–[99], in general the necessity of feedback control for motor tasks is clear [44]. A combination of both feedback and feedforward processes is likely to be involved for most optimal movement control tasks [87], [100], [101]. Especially in the context of adaptation to new tasks or new dynamical environments feedback is needed. In that regard, a feedback loop is essential for the analysis of human learning strategies, but the derivation of optimal control strategies is proved to be more complex for the closed-loop approach than for the open-loop one [12]. In essence, we assume that learning processes for such reaching tasks are completed and that “the sensorimotor control is best described as being near optimal” [1] for an unknown cost function subject to the dynamics of the plant. Furthermore, such analysis would also help us better understand and build closed-loop systems to model more complex movement behaviors from control theoretical point-of-view. It is clear that the current IOC formulation can be extended with a non-deterministic formulation and analysis. Considering the signal-dependent noise in motor control, the end-point variance can be tackled with a stochastic optimal feedback controller [5], [33], [102]. However, integrating stochastic components to an IOC formulation is still an open research question in control theory to effectively solve such problems for complex systems [103], [104].

3

A Hybrid Framework for Understanding and Predicting Human Reaching Motions[1]

Robots collaborating naturally with a human partner in a confined workspace need to understand and predict human motions. For understanding, a model-based approach is required as the human motor control system relies on the biomechanical properties to control and execute actions. This model-based control models explain human motions descriptively, which in turn enables predicting and analyzing human movement behaviors. Prior to a human-robot interaction (HRI) scenario, the tasks can be pre-analyzed and configured, and similarly, during task execution the robot motion can be planned and adjusted accordingly to provide a less demanding working environment for the human. In motor control, reaching motions are framed as an optimization problem. However, different optimality criteria predict disparate motion behavior. Therefore, the inverse problem – finding the optimality criterion from a given arm motion trajectory – is not unique. This work uses the inverse optimal control (IOC) approach described in Chapter 2 to determine the combination of cost functions that governs a motion execution. The results indicate that reaching motions depend on a trade-off between kinematics and dynamics related cost functions. However, the computational efficiency is not sufficient for online prediction to be utilized for HRI. In order to predict human reaching motions with high efficiency and accuracy, we combine the IOC approach with a probabilistic movement primitive (ProMP) formulation. This hybrid model allows an online-capable prediction while taking into account motor variability, and the interpersonal differences. The proposed framework affords a descriptive and a generative model of human reaching motions which can be effectively utilized online for human-in-the-loop robot control and task execution.

3.1 Introduction

As robots become more present in our social lives, the necessity for interaction and collaboration between humans and robots is becoming more apparent. Although there are several major facets of

[1] This work has previously appeared in the following publication: [236]

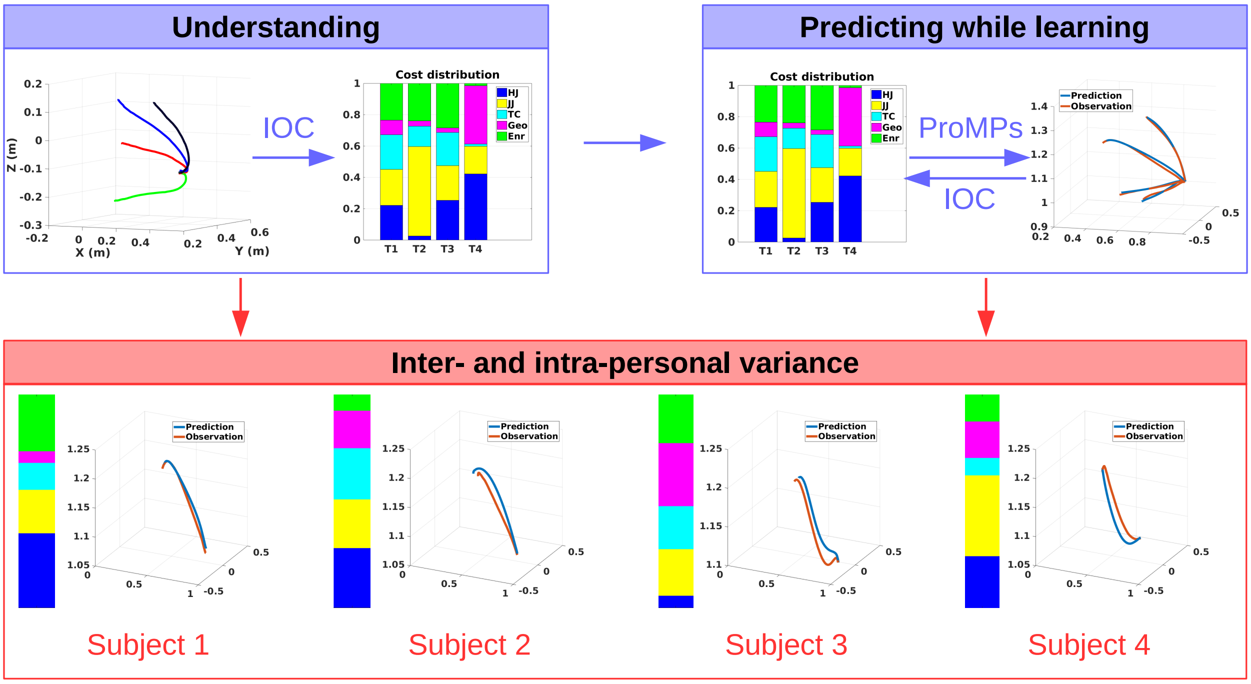


Figure 3.1: The overall framework, where the focus is two-fold: understanding (upper left), and prediction (upper-right) of human motion behaviors. For understanding, biomechanically inspired cost function distributions are learned from demonstrations by model-based inverse optimal control; and for online prediction, data-driven probabilistic movement primitives are used. The two approaches are interconnected to each other in order to account for the inter-and intra-personal movement behavior variations in terms of both motion trajectories and also the cost distributions.

providing robots with such capability, e.g. motion planning or decision making, the human aspect has to be prioritized and integrated into robot interaction skills. Requirements for such a human-in-the-loop formulation is two-fold: describe (*understand*) how human motions are controlled, and generate (*predict*) human-like motions. A descriptive model help us understand how the biomechanical properties are used by the central nervous system (CNS) for controlling human body to execute a vast collection of motor behaviors. Such an understanding is useful for a multitude of problems, e.g. motor performance evaluation for detecting disabilities due to neural disorders by comparing control models of patients and healthy subjects [15]; sports performance evaluation by analyzing the identified control models of athletes [105]; detection of deviations of personal motion behaviors w.r.t. the previously identified motor control models e.g. due to exhaustion [16]. Specifically, for human-robot interaction (HRI), the robot can plan its motions in a way to allow the human partner to rely more on energy-efficient control models. In addition, person specific control models enable the robot to detect the underlying cause of behavioral anomalies for providing better assistance and safety.

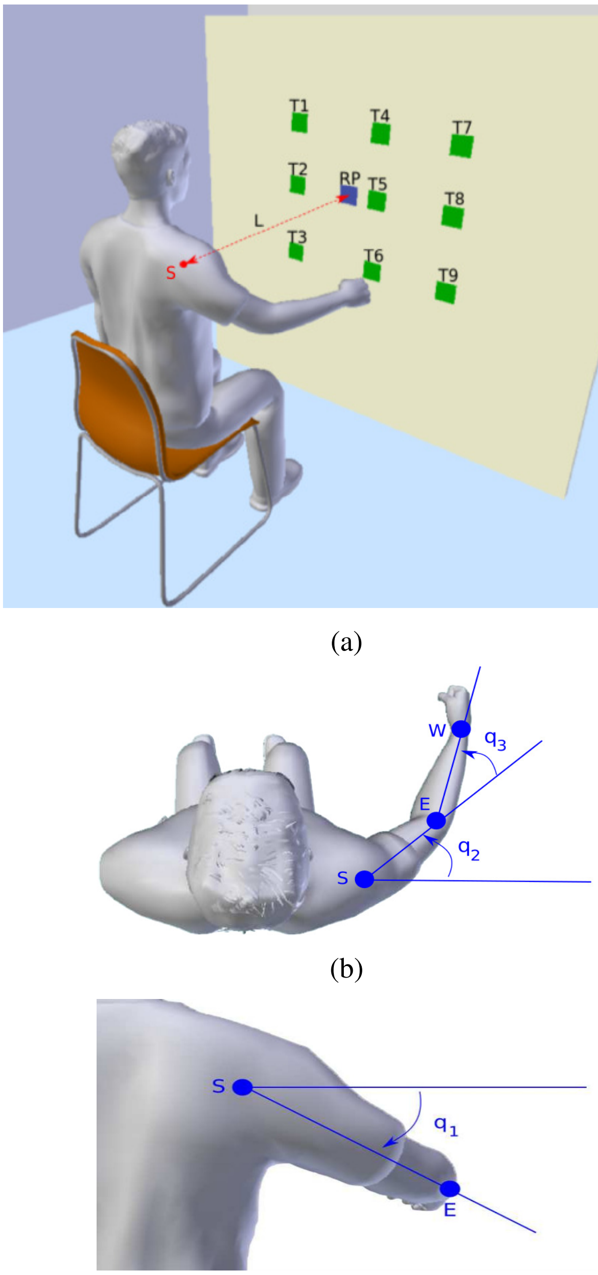
A generative model allows estimating human-like motion trajectories. In this work, the focus is using such models to *predict* human motions, rather than transferring them to robots to generate human-like movement behaviors. For close dyadic collaboration, where the partners share a workspace with the possibility of overlapping motions, they should be able to predict each other's intent and the required motion that can support this intention. Considering how swiftly two humans work together in a confined workspace, the challenges for a human-robot team become obvious; the robot has to take into account human partner's intention and movement in order to control its own motion for achieving effective cooperative task executions. In essence, early prediction of the human motion allows an immediate initiation of the replanning process and an early adaptation of the robot motion [237], [239], [242]. Therefore, the ability of understanding and predicting human motions effectively is the key to achieving swift close human-robot collaboration.

The focus in this work is two-fold. First, descriptive models of human reaching motions are investigated and experimentally evaluated. Second, a hybrid framework is proposed, which combines those descriptive models with a data-driven probabilistic approach and realizes online-capable human motion prediction (Fig. 3.1). Such a framework not only enables effective robot control for human-in-the-loop scenarios, but they can also be directly used for controlling the robot.

Currently, there is no commonly accepted model that explains how the human CNS controls human motions and the latent biomechanical properties of the human motion are not fully understood. Knowing the underlying principles of human motion execution is essential for reproducing human-like motion behaviors accurately in a given setting. However, not every single person exhibits the same motion patterns. These differences might be due to their learning experiences and physiological differences [106]. Moreover, even the motion behaviors of the same person show variations due to motor noise [5]. Considering all those intricacies, finding motion behavior models, even for simple reaching tasks, poses challenging research questions.

As the observations of the human motions' behavioral aspect suggest an appealing modeling problem, the human body as a biomechanical system introduces challenges in terms of formulating methods for finding those models. Motor control redundancy, and the non-linear characteristic of the human arm as a dynamical system are the most important problems to tackle. A common feature of motor control is that the task requirements can be met by infinitely many diverse movements. Thus, stating only the boundary conditions of the motion for given dynamics leads to an ill-defined problem. The ambiguity caused by this problem can be resolved if an optimality principle is applied. Accordingly, the basis of many scientific theories on human motor control is formed by optimality principles [107]. A large number of models of open-loop motor control exist and each model claims to describe human motion, but several models are incompatible with others [44]. The characteristics of the human arm movements and the human as an organism define the starting point for the derivation of a cost function. Many cost functions have been proposed to model human reaching motions, however, all of those methods are only verified for specific settings, mostly in 2D [11], [32], [33]. Hence their generalization capability to a wider range of human reaching motion behavior in 3D space is unclear. Moreover, as some recent studies suggest, humans might be optimizing two class of cost functions, one for kinematics, and the other for dynamics [58], [59]. However, finding the contribution of such multiple cost functions is also not trivial as it is a non-linear optimization problem.

Building on the results of prior research studies, and their insights, we hypothesize humans utilize multiple models, rather than a single one, to control their motions. Since kinematics is essential for producing smooth motions, and the human arm is a dynamical system, it is reasonable to consider kinematics and dynamics related costs in combination. Hence, we identify possible costs from literature to account for both aspects. In order to find the contribution of each model for the realization of human motion behaviors, we frame such an inverse optimal control (IOC) problem as a bi-level optimization formulation. However, this formulation treats the human motion generation



(a)

(b)

(c)

Figure 3.2: Experimental setup same as previous work (Ch. 2). (a) Overview of the experimental setup. T1 to T9 denote the nine target areas. RP means the reference point used to adjust the sitting position of the subject. S is the center of the shoulder joint and L is the distance between S and RP which is defined as 80% of the subject's arm length. (b) Top view of the subject. S, E, W are the shoulder joint, the elbow joint and the wrist joint respectively. As in the arm model defined, q_2 and q_3 are the yaw rotation of the shoulder joint and the pitch rotation of the elbow joint. (c) Back view of the subject. q_1 is the pitch rotation of the shoulder joint.

as a deterministic problem. In essence, it is only suitable for modeling average behavior over a group of humans. In order to afford both intra- and inter-personal motion variability, we propose a hybrid framework by extending the IOC formulation with a data-driven probabilistic method. Specifically, by utilizing probabilistic movement primitives (ProMPs), our framework allows for integrating person specific variations into the IOC-based average motion behavior models during online interaction. Therefore, we can learn a distribution of motion behavior per person, and roll-out predictive trajectories from this distribution online, while updating at the same time the multiple model representation to describe the person specific cost optimization behavior.

The initial data was acquired from the comprehensive experiment conducted for the previous study (Fig. 3.2, Ch. 2) that covers significantly more cases than prior studies [9], [82]. This extended experiment provides us with critical insights on the interplay between the parameters of the reaching tasks and the contribution of kinematics and dynamics related models. We identify a trade-off between those models with respect to the initial and final joint angle configurations. With the proposed hybrid framework, we are able to determine personal preferences as well as the motor variability per person. It also enables accurate and computationally efficient online prediction of human motion behaviors, which can be integrated into any human-robot collaboration scenario.

In this work, we focus on building descriptive as well as generative models for human motion behavior. By utilizing such models, we aim for efficient and accurate prediction of human motions during human-robot collaboration to realize a natural interaction between partners. This work, and similar hybrid frameworks have been integrated into our other studies for robot action selection [237], [238] and motion planning [242] algorithms. The main contributions of this specific research study are:

- We propose a hybrid framework, consisting of a model-based approach and a data-driven probabilistic method, for predicting human motions.
- We identify a trade-off between kinematics and dynamics related costs depending on the reaching task.
- Our hybrid framework takes into account interpersonal differences and person specific motor variability during online observations.

3.2 Related Work

Many experimental studies have revealed that arm motions exhibit invariant parameters which do not significantly change with movement speed, load or direction [108]–[110]. For motor control, these parameters are utilized to describe point-to-point reaching motions [111]. It is assumed that the CNS follows some specific principles when planning the motions [107]. Therefore, optimal control theory becomes the central mathematical formulation to model, describe, and understand motor control by the CNS [44], [68], as it emphasizes the optimality of biological movements by minimizing some performance criteria. In literature, several optimal control models have been proposed to describe the point-to-point arm movements, e.g. the minimum hand jerk [32], the minimum torque change [11] and the minimum variance [33]. These models are proven to be efficient in representing the experimental data. However, they are only verified within specific settings, and exhibit, in some cases, dissimilar patterns. Hence, the exact variables optimized in the brain still remain unclear. Later studies suggest that, instead of a single cost function, the CNS might actually consider a weighted combination of costs during the optimization [112]–[116]. It has already been verified that the trade-off between the objective (task-related) and the subjective (subject-related) cost functions exists in the CNS [12], however, there is still no clear explanation about how the subjective costs are combined in reaching motions. In [58], this cost combination hypothesis was tested in point-to-bar reaching motions on a vertical 2D plane. An inverse optimal control framework, which was initially proposed in [57] for locomotion planning, was applied to identify the contribution of different cost functions. Though their results support the idea of the combined cost functions, an in-depth analysis on how this combination is formed in 3D reaching motions and whether there is a relationship between the degree of contribution and the reaching task parameters is still missing.

Inverse reinforcement learning (IRL), also sometimes used synonymously with inverse optimal control (IOC), is another line of formulation to find control models, or optimal policies given some demonstrations or observations. However, most of the state-of-the-art methods operate on features rather than raw states, and without relying on the dynamical system as a hard constraint on the optimization problem. In essence, the best combination of features, which are extracted during an agent interacting with the environment, is solved for minimizing a pre-defined cost function [27], [117]–[120]. A recent approach by [121] extends such IRL formulation by tackling the requirement on defining informative features with using neural networks to parameterize the cost function. Essentially, this approach learns nonlinear cost functions from user demonstrations, at the same time as learning a policy to perform the task. This formulation can be applied to complex, nonlinear cost function representations and high dimensional problems. However, this is still not directly comparable to solving optimal control problems where the dynamical system is a constraint at each time step, and hence the resulting behaviors are not guaranteed to be generated by the underlying model.

In contrast to creating an optimal control model, another approach to predict human motions is to use data-driven methods. These methods focus more on finding a representation from a given data-set [122], [123]. Statistical approaches require training data to discover patterns for different arm motions. In that sense, a rigorous and time consuming data collection process is unavoidable. Other data-driven approaches which do not rely on statistical formulations, e.g. dynamic movement primitives (DMPs) [124], require only a minimal set of training data. In an earlier work, we combined optimal control models with DMPs to predict human reaching motion behaviors while avoiding obstacles [238]. In that regard, Interaction Primitives (IPs) were developed based on DMPs as a compact representation of a dyadic activity to predict and plan interaction behaviors [125]. IPs are learned as a distribution over DMP parameters by training on two interacting partners' trajectories. These IPs encode reciprocal dependencies of dyad movements during the execution of a specific task. The robot then mimics one partner by using the learned model to interact with a human in a similar task. In essence, the learned distributions are conditioned on an observed partial trajectory to predict a human partner's movement for the rest of the task, which in turn enables the robot to correlate its own motion w.r.t. the human to achieve a successful cooperation. Recently, Environment-adaptive Interaction Primitives (EaIPs) were proposed by extending the IPs with the integration of environmental parameters of the task [126]. Hence, EaIPs enable inferring movement behavior by conditioning on not only the partner trajectory but also the task and environment related features. However, these are pure data-driven approaches, and thus, they can neither elicit the underlying principles of human interaction movement control, nor provide any means to analyze optimality of observed movements. In addition, our proposed hybrid framework is flexible to integrate such interaction primitives as the data-driven part of the formulation to predict human motions, which can further be integrated into a trajectory optimizer for the robot motion planning in HRI scenarios [242].

Lastly, human motor control by the CNS is recognized as a stochastic system [5], thus the variance of the motion should be considered in the trajectory prediction. In [127], a probabilistic movement primitives (ProMPs) approach was proposed with the ability to encode the variance in a general probabilistic framework for representing and learning movement primitives [128]. The ProMPs has been successfully implemented in human-robot interaction [129] and human-robot collaboration [130], [131] scenarios for controlling the robot motion. For a close cooperation between the robot and human, a precise prediction of the human behavior is essential [123]. However, while predicting human motions with the ProMPs, the integration of the kinematics and dynamics of the human arm is not straightforward. Our work combines an optimal control model with the ProMPs, in order to make use of the advantages from both methods.

3.3 Hybrid Online Prediction Framework

In literature, many prediction methods for human motion were proposed. Among them, two classes of the methods are widely used: 1) *model-based methods*, where a motion model is created based on minimizing a criterion, such as the minimum hand jerk model [32], the minimum joint angle jerk model [71] and the minimum variance model [33]. Then the solution to the model is considered as the prediction; 2) *data-driven methods*, where a set of data (observations) should be available before building a generative model for predicting human motions. The characteristic of the motion can be learned from the data and then the prediction is generated by reproducing this characteristic and in some cases with variation. Gaussian Mixture Models [132], [133], dynamic movement primitives [124] and probabilistic movement primitives [127] are typical data-driven methods. In this section, we propose a hybrid online prediction framework for reaching motions by combining a model-based method and a data-driven method. Instead of using the motion model with single cost function, a composite model is obtained by the IOC framework described in Chapter 2.2.4. In order to account for the motor variability of the reaching motion [5], this composite model is combined with the ProMPs. ProMPs are selected due to both their capability on learning a model with a very small amount of observations (in our experiments 5-10 samples seem to be enough), and also their computational efficiency for rolling-out predictive trajectories online. Especially, it is known that GMMs tend to perform poorly in high-dimensional spaces when few datapoints are available [134]. In the rest of this section, first the regression model used to approximate the IOC model parameters is explained, followed by a brief explanation of the ProMPs. Then, a comparison between the predictions of the composite model and the ProMPs is discussed. Lastly, the hybrid prediction framework is explained in detail.

3.3.1 Representation of the reaching motions

From the IOC formulation, we acquire a weighted combination of cost functions, which specifies the contribution of each model for the realization of a reaching motion. For each specific motion behavior, one composite model needs to be found. However, we can only have a limited number of different composite models, due to the computational time limit. To utilize the composite model in general cases, a mapping from the motion parameters to the contribution of cost functions is required. According to the results of the initial experiment we conducted, which is detailed out in section 3.4.1.3, a correlation between the initial and the final joint angles ($\boldsymbol{q}_s, \boldsymbol{q}_e$) and the optimal weight vector $\boldsymbol{\alpha}^*$ is identified. Here we use the Gaussian Process Regression (GPR) model [135] to represent the mapping as

$$\boldsymbol{\alpha}^* = GPR(\boldsymbol{q}_s, \boldsymbol{q}_e), \tag{3.1}$$

where GPR denotes the GPR model. The optimal weight vector returned by the GPR model is a distribution with mean and variance. Note that the GPR model can be replaced by other similar stochastic models, but we find that the GPR model is more suitable in our case since it requires less data. This GPR model provides a connection between the IOC formulation and the ProMPs in our hybrid online prediction framework.

3.3.2 Probabilistic movement primitives

The ProMPs is a probabilistic formulation for movement primitives. It is able to capture the variance information of trajectories and represent the behavior in stochastic systems. Given a discrete trajectory $\boldsymbol{X} = \{x_t\}, t = 0 \dots T$ defined by states x_t over time T, a weight vector $\boldsymbol{\omega}$ is used to represent the trajectory as

$$\boldsymbol{y}_t = [x_t, \dot{x}_t]^\top = \boldsymbol{\Phi}_t^\top \boldsymbol{\omega} + \boldsymbol{\epsilon}_y, \tag{3.2}$$

where $\boldsymbol{\Phi}_t = [\phi_t, \dot{\phi}_t]$ denotes the $n \times 2$ dimensional time-dependent basis matrix for states x_t and the velocities $\dot{x}_t$. n is the number of basis functions and $\boldsymbol{\epsilon}_y \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_y)$ is zero-mean independent and identically distributed Gaussian noise. The mean of the trajectory can be obtained by weighting $\boldsymbol{\Phi}_t$ with $\boldsymbol{\omega}$. The probability of observing a trajectory $\boldsymbol{X}$ with a given $\boldsymbol{\omega}$ is represented by a linear basis function model as

$$p(\boldsymbol{X}|\boldsymbol{\omega}) = \prod_t \mathcal{N}(\boldsymbol{y}_t|\boldsymbol{\Phi}_t^\top \boldsymbol{\omega}, \boldsymbol{\Sigma}_y). \tag{3.3}$$

In order to capture the variance, a Gaussian distribution $p(\boldsymbol{\omega}; \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\omega}|\boldsymbol{\mu}_\omega, \boldsymbol{\Sigma}_\omega)$ over the weight vector $\boldsymbol{\omega}$ is introduced with parameters $\boldsymbol{\theta} = \{\boldsymbol{\mu}_\omega, \boldsymbol{\Sigma}_\omega\}$. Then the distribution of $\boldsymbol{y}_t$ at time t is given by

$$p(\boldsymbol{y}_t; \boldsymbol{\theta}) = \int \mathcal{N}(\boldsymbol{y}_t|\boldsymbol{\Phi}_t^\top \boldsymbol{\omega}, \boldsymbol{\Sigma}_y)\mathcal{N}(\boldsymbol{\omega}|\boldsymbol{\mu}_\omega, \boldsymbol{\Sigma}_\omega)d\boldsymbol{\omega} = \mathcal{N}(\boldsymbol{y}_t|\boldsymbol{\Phi}_t^\top \boldsymbol{\mu}_\omega, \boldsymbol{\Phi}_t^\top \boldsymbol{\Sigma}_\omega \boldsymbol{\Phi}_t + \boldsymbol{\Sigma}_y). \tag{3.4}$$

With Eq. (6.3), the mean and the variance of the states for any time point t can be derived. If a set of observations is available, the parameters $\boldsymbol{\theta}$ can be learned by using the maximum likelihood estimation [136]. In reaching motions, the distribution $p(\boldsymbol{\omega}; \boldsymbol{\theta})$ can be considered as a representation of the motor variability. For more details of the ProMPs please refer to [127].

Perspective	Composite model	ProMPs
Underlying principle	Yes	No
Optimality	Yes	No
Computation time	High	Low
Motor variability	No	Yes

Table 3.1: Different perspectives of the composite model prediction and the ProMPs prediction.

3.3.3 Comparison between the composite model and the ProMPs prediction

Both the composite model formulation and the ProMPs framework have clear advantages, and drawbacks, but they are also complementary. By combining them into a hybrid prediction framework, the advantages of both methods can be exploited at the same time (Table 3.1).

The composite model represents the underlying principles of reaching motion control. Several motion models have been proven to be accurate in describing the movements, such as the minimum hand jerk model on some tasks, and the minimum torque change model on others, in 2D reaching motions. The composite model we proposed inherits those capabilities and extends it to the 3D reaching motions. It helps us explain how humans execute and control their reaching motions, while extracting such information from the data-driven methods is not trivial. However, the biggest obstacle in implementing the composite model prediction in online case is the computation time. Before rolling out the optimal trajectory, an OCP needs to be solved, which usually takes several minutes, even when the state-of-the-art solvers are used [67]. However, in real world settings, the reaching motions take no longer than a few seconds, thus the data-driven methods are more suitable in the online case, as they are computationally more efficient.

Another important reason of using the ProMPs as the data-driven method in the hybrid prediction framework is that it allows describing the motor variability given sample demonstrations [127]. As explained in [5], human motor control is a stochastic system with signal-dependent noise [33], thus reaching motions are expected to show variance. Since it is not straightforward to consider the variance within an IOC problem, we formulate our composite model as a deterministic OCP. On the other hand, as the ProMPs formulation employs a probabilistic function to represent the motion, the obtained model is not a single trajectory but a distribution of trajectories. Hence, while the composite model describes an optimal average behavior as an initial guess, the ProMPs enables capturing the multiplicative noise due to motor control. However, to understand the control model due to such noise, the model-based IOC computation and a follow-up GPR update is still required.

3.3.4 Prediction framework

The idea of the hybrid prediction framework is, for a given reaching task, to use the composite model to generate the initial training data for the ProMPs. Then in the online phase, the ProMPs can roll out predicted trajectories with high efficiency while also learning the variance by using

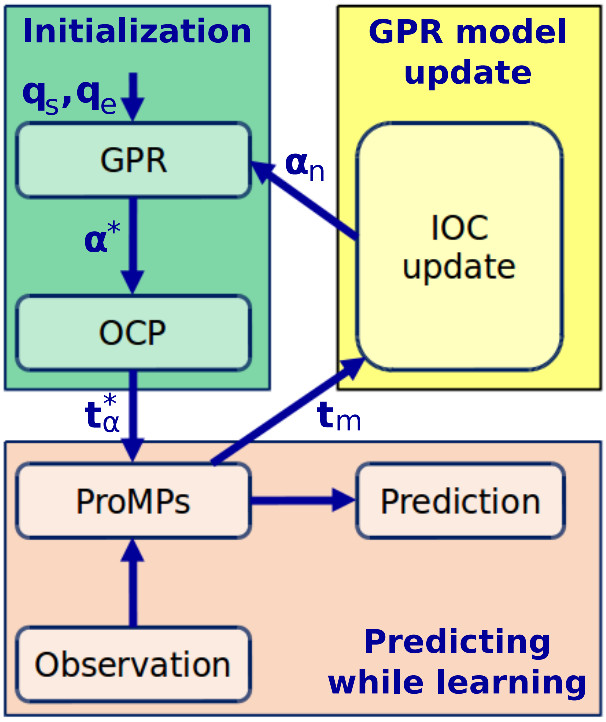


Figure 3.3: Overview of the prediction framework (upper-right in Fig. 3.1). $\boldsymbol{q}_s$ and $\boldsymbol{q}_e$ are the initial and final joint angle configurations. $\boldsymbol{\alpha}^*$ is the estimated optimal weight vector and $\boldsymbol{t}_\alpha^*$ is the corresponding optimal solution from OCP. $\boldsymbol{t}_m$ denotes the mean of the converged trajectory distribution extracted from the ProMPs, $\boldsymbol{\alpha}_n$ is the new obtained optimal weight vector, which is used to update the GPR model.

each motion observation as new training data. After several observations, the parameters of the ProMPs converge (the details is explained in section 3.4.2.2), then the mean of the converged trajectory distribution is calculated to update the composite model. An overview of the framework is given in Fig. 3.3, and the details of this hybrid model is explained next.

3.3.4.1 Initialization with the composite model

Usually for a given reaching task, the starting position and the target position are known. Through the inverse kinematics, the initial joint angle configuration $\boldsymbol{q}_s$ and the final joint angle configuration $\boldsymbol{q}_e$ can be approximated. By using the GPR model trained on the IOC results, a distribution of the estimated optimal weight vector is available. However, due to the limited amount of data for training the GPR model, the variance cannot be learned accurately. Thus only the mean value of the distribution $\boldsymbol{\alpha}^*$ is used here. After solving the OCP with respect to $\boldsymbol{\alpha}^*$, the optimal joint angle trajectory $\boldsymbol{q}_\alpha^*$ and its corresponding end-effector trajectory $\boldsymbol{t}_\alpha^*$ are obtained. $\boldsymbol{t}_\alpha^*$ is considered as the training data for the ProMPs. As the OCP gives a deterministic solution, no variance information can be derived. Hence the ProMPs is initialized by learning the parameters from the optimal trajectory $\boldsymbol{t}_\alpha^*$, while setting the variance to a large value.

3.3.4.2 Predicting while learning

During online prediction, a trajectory along with the variance for each time point is generated by the ProMPs. This variance information is useful for human-robot interaction scenarios where the robot should also consider the uncertainties of human behaviors. The observations recorded during the prediction are utilized to update the ProMPs to get a more accurate representation of the variance. After each movement, the observation is added to the data storage which contains all the previous observations. Subsequently, the ProMPs update their parameters from the new data storage. With the incorporation of each motion observation, parameters of the ProMPs as well as the variance information converge.

3.3.4.3 GPR model updating

Once the ProMPs system becomes stable, the mean of the converged trajectory distribution $\boldsymbol{t}_m$ can be extracted. This trajectory can be considered as the average behavior of the recorded subject for this reaching task. Then in a separate updating process, $\boldsymbol{t}_m$ is used by the IOC framework to get the corresponding optimal weight vector. The new optimal weight vector $\boldsymbol{\alpha}_n$ is used to update the GPR model. Therefore, with more information returned from the real recordings, the GPR model also becomes more accurate in describing the mapping from the initial and final joint angles to the optimal weight vector.

3.3.5 Motor variability and interpersonal variance

The motor variability is essential in describing human behaviors [5], as it can be considered as the uncertainties of human motions (e.g. the noise in motor command). It represents the fact that for a given reaching task, even the same subject is expected to execute the motion in slightly different trajectories. This phenomenon has been reported in sensorimotor control by demonstrating such variability on observed experimental data for a multitude of tasks, e.g. locomotion [137], writing [138], pointing [139], reaching [140], grasping [141]. Usually for simple tasks this difference is not large and can be modeled as a probabilistic distribution [142], [143]. However, such probabilistic models cannot explain the underlying cause of observing such motor variability, which is known to be due to additive and multiplicative noise in the motor control and is treated as the intra-subject variance in this work. Apart from the motor variability, there are also motion behavior differences between subjects [144], which we call interpersonal variance in this work. The existence of such a disparity can be verified through the contribution of basic cost functions, as shown in the next section. The interpersonal variance suggests that humans plan their motions in a personal way, which reflects the dissimilarity of the control structure due to learning, and adaptation effects, along with biomechanical differences. Thus the updated GPR model from the hybrid prediction framework is actually a person specific model.

3.4 Experiments and Results

In this section, two experiments and their corresponding results are presented. One is designed for the IOC framework with the purpose of understanding the characteristics of human reaching motions, and the other is used to test the performance of the hybrid online prediction framework.

3.4.1 Experiment for the IOC framework

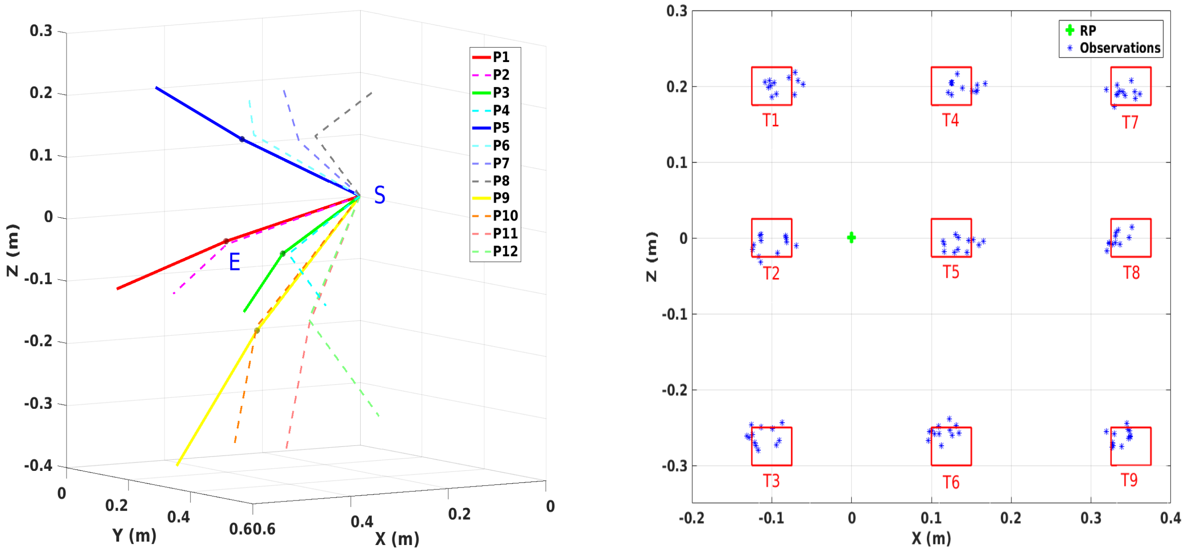
To cover the reaching motions in a relatively large range, we designed an experiment for point-to-point reaching tasks consisting of 12 starting postures and 9 target regions. The recorded trajectories were analyzed through the IOC framework. Based on the obtained optimal weight vectors, we find that the contribution of basic cost functions has a relationship with the initial and final joint angle configurations. Besides, the composite cost function is proven to have less error in describing the reaching motions in almost all tasks compared to the single cost models. This result encourages us to use the composite model in the prediction rather than a model with single cost function. In the rest of this subsection, at first the details about the experimental setup are presented, then the results from the IOC framework are discussed.

3.4.1.1 Experimental setup and data collection

A visualization of the experimental setup is presented in Fig. 3.2a. Participants were required to sit before a board which was placed vertical to the ground surface. Nine target areas and one reference point were marked on the board as square regions with the side length equal to $5cm$. The distances between the target areas and the reference point are shown in Fig. 3.4b. Before the experiment, the sitting height of the participant was adjusted by setting a straight line between the reference point and the center of the shoulder joint vertical to the board surface. Then the distance between the center of the shoulder joint and the board surface was selected as 80% of the arm length. These distances were chosen to ensure that the participants can reach all nine targets easily without moving their torso.

Since we want to cover a large range of reaching motions, every participant was asked to reach the nine targets from 12 different starting arm postures. According to the joint angle limits we defined in the arm model, these starting postures were chosen from the combination of three different q_1, two different q_2 and two different q_3 $(3 \times 2 \times 2 = 12)$ configurations (see Table 3.2). As shown in Fig. 3.4a, the pitch rotation of the shoulder joint q_1 is selected as three configurations: up, middle and down respectively. The yaw rotation of the shoulder joint q_2 and the rotation of the elbow joint q_3 are chosen from the stretched to the side configurations and a configuration in the middle of the joint angle limits. With nine targets for each starting posture, 108 (12 starting postures $\times$ 9 targets) cases of the reaching motions were considered in the experiment.

Before the recording, the arm posture was determined by measuring all three joint angles to ensure all participants shared the same starting joint angle configuration. The participants were given the following instructions. First, in order to discard the decision-making process of target selection, the subject needs to reach the nine targets in a fixed order as from target one to target nine. Second, the participant should strictly put his arm in the previously set starting posture before executing the follow-up reaching task. A set of special reference tools were prepared and put beside the participants. These tools consist of two bars and their end points indicate the positions of the elbow and wrist joints for the given starting posture. Reference tools were placed in appropriate positions so that during the reaching motion they do not block any potential motion trajectory. Third, in order to eliminate the effect of locating targets during the movement, before the execution of the reaching tasks, the participants should look at the targets rather than the reference tool. Fourth, the participants were told to avoid using the roll rotation of the shoulder joint, which is ignored in our arm model. In addition, all participants were trained before the experiments to get familiar with



(a)

(b)

Figure 3.4: Experimental setup. (4a) Visualization of 12 starting joint angle configurations. P1 to P4 are the postures with q_1 in the middle (no rotation), while P5 to P8 are the postures with q_1 in the up region and P9 to P12 with q_1 in the down region. (4b) Target areas on the board surface. RP denotes the reference point. Observations are the actual positions where the 108 averaged trajectories terminate on the board surface.

the setup and the task. If any unintended motion was detected during the recording, corresponding tasks were executed again. Between each starting posture, enough rest time was provided for avoiding fatigue. To reduce the noise, every target in every starting posture was reached two times, thus a total of 216 (108 cases $\times$ 2 times) trajectories were recorded for one participant.

The data were collected from fifteen subjects (11 males, age: 27 ± 4; weight: $67 \pm 9\ kg$, height: $172 \pm 5\ cm$) who all gave written informed consent for their participation. All the participants were right-handed with normal vision ability. None of them received any information about the purpose of the experiment. The study was approved by the ethics committee of the Technical University of Munich School of Medicine. The reaching motions were recorded by the multicamera motion capture system Qualisys at a frequency of $250Hz$. With the built-in filter function, the smooth position trajectories of the shoulder, elbow and wrist joints can be directly obtained from the tracking system, and used for the IOC calculations.

3.4.1.2 Average motion behavior

In our IOC framework, we are interested in the control structures for the human reaching motion behavior in a general sense, rather than the individual differences. We also intend to provide a base model to be extended for person specific motion behaviors during prediction. Hence, we compute the average trajectories from all 15 subjects, and the IOC problems are solved for these trajectories. Besides, the averaging process also saves a lot of computation time. Since the IOC calculation for one trajectory roughly takes four hours, the analysis on all 1620 (15 subjects $\times$ 108 cases) trajectories would require an immense amount of time. Table 3.2 gives the mean values

	$\mathbf{q_{1,S}}(^\circ)$		$\mathbf{q_{2,S}}(^\circ)$		$\mathbf{q_{3,S}}(^\circ)$	
Posture	Mean	Std	Mean	Std	Mean	Std
P1	10.95	5.01	6.58	4.66	12.72	3.49
P2	11.21	5.73	8.78	9.47	33.39	6.51
P3	11.93	3.70	31.90	5.82	13.15	3.74
P4	13.00	6.37	34.45	6.80	37.92	8.53
P5	-22.29	5.21	12.46	5.18	14.11	3.86
P6	-23.47	5.71	15.82	6.41	37.88	7.51
P7	-22.89	5.33	37.31	7.91	16.10	5.09
P8	-23.64	5.66	41.07	8.42	35.75	7.88
P9	42.15	6.16	6.98	7.43	12.28	4.58
P10	40.22	4.40	7.08	5.28	35.40	5.59
P11	35.36	5.09	36.14	5.61	10.06	5.76
P12	35.14	5.45	36.88	6.85	43.44	6.69

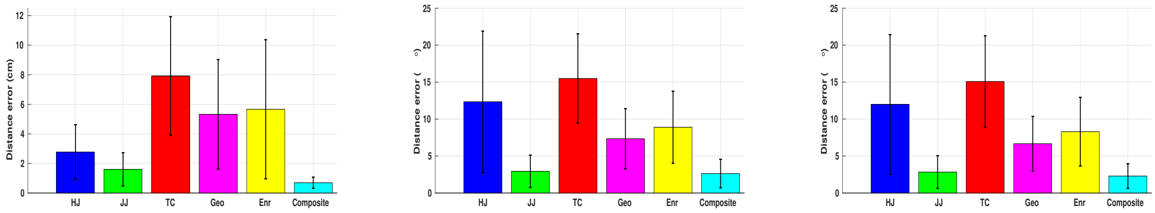
Table 3.2: Actual starting joint angle configurations. P1 to P12 are the 12 predefined starting postures. $q_{1,S}$, $q_{2,S}$ and $q_{3,S}$ are the three starting joint angles with respect to the stretched out posture. The values are computed by using all 15 subjects' data.

and the standard deviations of 12 starting joint angles calculated from all subjects' data. The standard deviations indicate that for the same starting posture, all subjects started their reaching motions with a relatively small joint angle difference, which enables the feasibility of averaging the trajectories. If not mentioned explicitly, all the IOC results presented in the following part are based on the averaged trajectories.

3.4.1.3 Results for the IOC framework

After the IOC calculations, we obtained one optimal weight vector for each reaching task. The contribution of basic cost functions in 108 different cases are analyzed next.

Performance in describing the reaching motions To verify the performance of the composite model, the optimal trajectory solved with it is compared to the optimal trajectories computed for each single basic cost function. The distance error between each optimal trajectory and the average motion behavior is measured through the DTW based comparison separately. The results show that, almost for all cases, the composite model has a better performance in describing the reaching motions. Even though the distance metric we used in the upper level program of the IOC framework only considers the end-effector trajectory, the composite model still has less errors in the joint angle trajectories. Fig. 3.5 presents the distance error averaged from all 108 cases. The p-test results indicate that, there are significant decreases on the distance error when comparing



(a) (b) (c)

Figure 3.5: Average distance error over all reaching tasks. HJ, JJ, TC, Geo, Enr and Composite are the hand jerk, the joint angle jerk, the torque change, the geodesic, the energy and the composite cost function, respectively. Mean values and the standard deviations of the errors for each cost function are presented. (5a) Distance error measured by comparing the end-effector trajectories. (5b) - (5c) Distance error measured by comparing the joint angle trajectories with and without considering target one.

the composite model to all other five basic cost functions ($\mathrm{p}_i < 0.0001, i = 1 \ldots 5$). In joint angle trajectories, except the minimum joint angle jerk cost function (p = 0.1813), we still observe significant decreases ($\mathrm{p} < 0.0001$). The reason is, in 3D reaching motions, the observed joint angle trajectories are bell-shaped, which are quite close to the results derived from the minimum joint angle jerk cost function, especially when the reaching motion enforces approaching the joint angle limits (e.g. reaching target one). After we removed the cases of reaching target one in the comparison, there is still a significant decrease ($\mathrm{p} < 0.05$), now for all the cases, on the distance error in describing the joint angle trajectories with the composite model. Furthermore, it should be noted that, optimizing only dynamics related cost functions leads to inconsistent arm trajectories in terms of joint and Cartesian displacements (a single case is shown in Fig. 2.3). In contrast, even though maximizing smoothness in joint space (angel jerk, i.e. kinematic cost) was efficient to fit the angular and Cartesian displacements, it is reported by [82] that it fails to describe the movement in torque space accurately. It appears that the composite optimality criterion comprising different biomechanical properties is the only model that can explain both kinematic and dynamic aspects of the reaching behaviors.

Influence of the initial and final conditions In order to get a deeper understanding of the human reaching motions, an analysis on identifying the possible factors which influence the contribution of basic cost functions is performed. We conduct the N-way independent analysis of variance (ANOVA) on our results with four factors, the three starting joint angles $q_{1,S}, q_{2,S}, q_{3,S}$ and the target index T. As ANOVA checks the importance of one or more factors by comparing the response variable means at different factor levels, the results obtained can be utilized to identify the factors which have statistical significant influence on the examined variable. In Table 3.3, we list the corresponding results from our ANOVA analysis when selecting the response variable as the contribution of five different basic cost functions as well as the sum of dynamics related cost functions (the minimum torque change + the minimum energy), respectively.

From ANOVA analysis it can be concluded that the starting joint angles of the two shoulder rotations have influences on the contributions of the cost functions: $q_{1,S}$ has influence on the contribution of the hand jerk (F(2,58) = 19.5487, $\mathrm{p} < 0.0001$), the joint angle jerk (F(2,58) = 10.7701, $\mathrm{p} < 0.001$), the torque change (F(2,58) = 12.7500, $\mathrm{p} < 0.0001$), the energy (F(2,58) = 7.7557,

RV : Factor	Sum.Sq.	Mean.Sq.	F	p
HJ : $q_{1,S}$	0.9624	0.4812	19.5487	0.0000
HJ : $q_{2,S}$	0.1872	0.1872	7.6063	0.0078
HJ : $q_{3,S}$	0.0068	0.0068	0.2750	0.6020
HJ : T	0.1635	0.0204	0.8303	0.5796
JJ : $q_{1,S}$	0.4115	0.2058	10.7701	0.0001
JJ : $q_{2,S}$	0.0150	0.0150	0.7830	0.3799
JJ : $q_{3,S}$	0.0176	0.0176	0.9223	0.3409
JJ : T	0.2026	0.0253	1.3255	0.2494
TC : $q_{1,S}$	0.1005	0.0503	12.7500	0.0000
TC : $q_{2,S}$	0.0081	0.0081	2.0525	0.1573
TC : $q_{3,S}$	0.0004	0.0004	0.1092	0.7423
TC : T	0.0603	0.0075	1.9122	0.0753
Geo : $q_{1,S}$	0.1202	0.0601	3.0653	0.0543
Geo : $q_{2,S}$	0.0056	0.0056	0.2844	0.5959
Geo : $q_{3,S}$	0.0232	0.0232	1.1812	0.2816
Geo : T	0.0894	0.0112	0.5702	0.7980
Enr : $q_{1,S}$	0.2760	0.1380	7.7557	0.0010
Enr : $q_{2,S}$	0.1525	0.1525	8.5667	0.0049
Enr : $q_{3,S}$	0.0331	0.0331	1.8596	0.1779
Enr : T	0.2721	0.0340	1.9113	0.0755
Dyn : $q_{1,S}$	0.6702	0.3351	19.3833	0.0000
Dyn : $q_{2,S}$	0.2308	0.2308	13.3516	0.0006
Dyn : $q_{3,S}$	0.0411	0.0411	2.3760	0.1287
Dyn : T	0.3356	0.0420	2.4267	0.0246

Table 3.3: Results of ANOVA tests. RV denotes the response variable, selected as the contribution of each basic cost function (HJ: hand jerk, JJ: joint angle jerk, TC: torque change, Geo: geodesic, Enr: energy) and the dynamic related cost functions (Dyn: dynamics, which is the sum of the minimum torque change and the minimum energy). Four variables are considered as the factors, which are the three starting joint angles $q_{1,S}, q_{2,S}, q_{3,S}$ and the target index T. RV:Factor indicates the ANOVA result of the influence of the factor on the corresponding response variable (e.g. HJ: $q_{1,S}$ means the influence of $q_{1,S}$ on the contribution of the minimum hand jerk cost function). Sum.Sq. and Mean.Sq. are the sum of squares due to each source and the mean squares for each source, respectively. F is the F-statistic, p are the p-values, which represents the probability that the F-statistic can take a value larger than a computed test-statistic value.

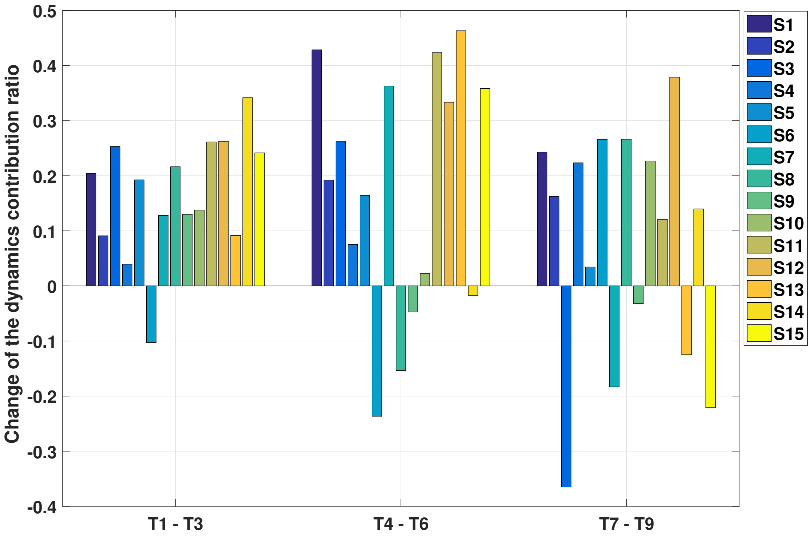
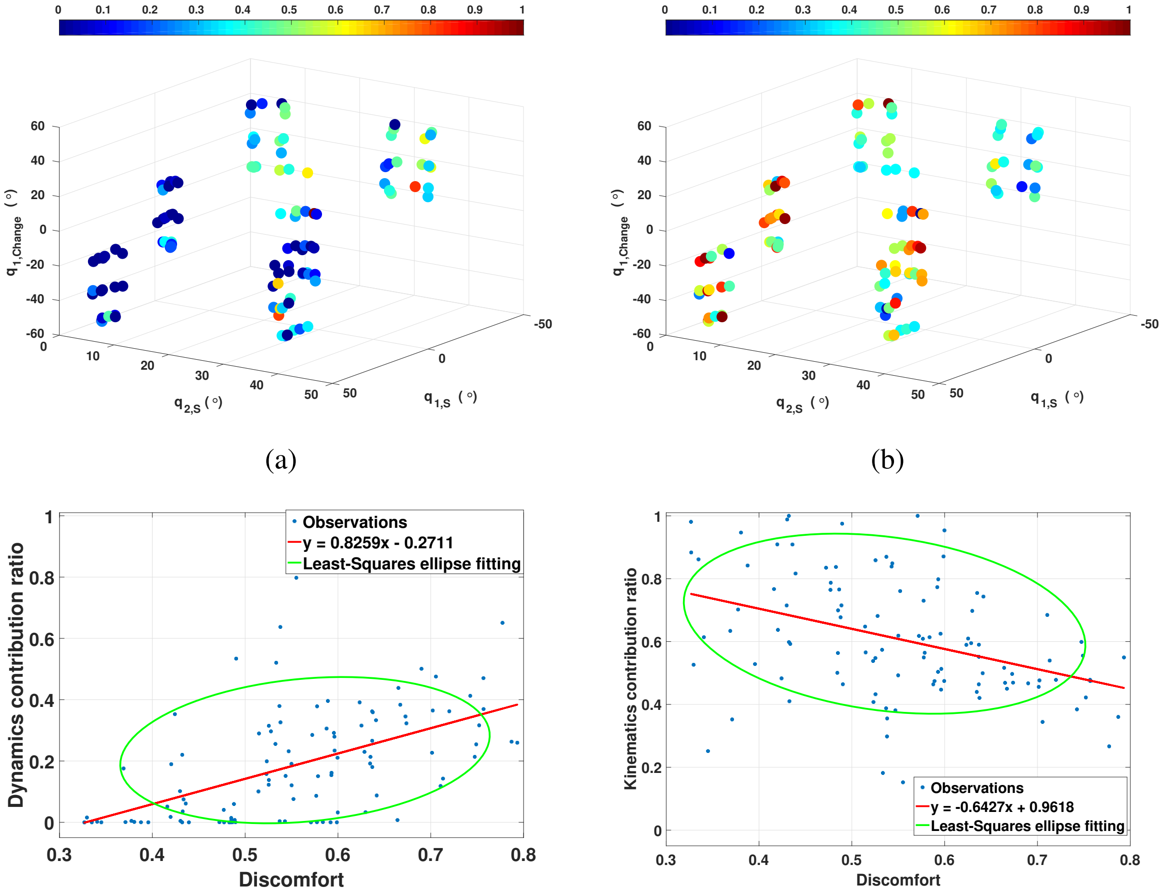


Figure 3.6: Results of the individual analysis. S1 to S15 represent 15 subjects. The change of the dynamics related cost is derived by subtracting C_{top} from C_{bot}, where C_{top}, C_{bot} are the contribution of dynamics related cost for targets in the top and bottom row, respectively. (e.g. T1 - T3 means subtracting the contribution of the dynamics to the target three from to the target one).

p < 0.001) and the dynamics (F(2,58) = 19.3833, p < 0.0001); while $q_{2,S}$ has influence on the hand jerk (F(1,58) = 7.6063, p < 0.01), the energy (F(1,58) = 8.5667, p < 0.01) and the dynamics (F(1,58) = 13.3516, p < 0.001). For the target position, only the dynamics is affected (F(8,58) = 2.4267, p < 0.05). Lastly, the starting joint angle of the elbow rotation $q_{3,S}$ has no influence on the contribution of basic cost functions (all p > 0.05).

In order to identify how the target position, which can be expressed by the three final joint angles $q_{1,E}$, $q_{2,E}$, $q_{3,E}$, affects the contribution of the dynamics, an individual analysis is conducted on the trajectories of each subject with one starting posture (fully stretched out posture P1) and six targets (top row: T1, T4 and T6, bottom row: T3, T6 and T9). Thus 90 (15 subjects × 6 trajectories) IOC calculations are performed. Then p-test is utilized to find if there is a significant difference between different final joint angles. The results suggest that only $q_{1,E}$ has influence on the contribution of the dynamics related cost, which indicates that only the height of the targets matters. This can be verified in Fig. 3.6, where we compare the contributions of the dynamics related cost between two sets of targets (top vs bottom row). From these results, the interpersonal variance can also be observed, where the changes are different for each subject, and sometimes this difference can be considerably large.

Transition between different reaching tasks According to the previous results, three factors are identified to be related to the contribution of basic cost functions, which are the two starting joint angles of the shoulder joint $q_{1,S}$, $q_{2,S}$ and the change of the pitch rotation of the shoulder joint $q_{1,Change} = q_{1,E} - q_{1,S}$. In order to identify how exactly these factors affect the contribution, two 3D scatter plots are given in Fig. 3.7a - 3.7b. Considering the musculoskeletal loading as the cri-



(a)

(b)

(c)

(d)

Figure 3.7: Transition between different cases. (7a) - (7b) Contribution of the dynamics (left) and the kinematics (right) related costs with respect to three factors. $q_{1,S}$, $q_{2,S}$ are the two starting joint angles of the shoulder rotations, $q_{1,Change}$ is the change between the final and the initial angle of the pitch rotation of the shoulder joint. The colors indicate the contribution ratio of corresponding cost. (7c) - (7d) Relationship between the proposed discomfort metric and the contribution of the dynamics and the kinematics related costs. Red lines are the linear regression models created based on the discomfort value with respect to the optimal weights (β_1^*, β_2^*). Another Least-Squares ellipse fitting is also presented to demonstrate the trend with variance.

terion to describe the comfortableness of the reaching motions [83], [84], the fully stretched down posture can be treated as the most comfortable posture. Then the more rotations required to execute the reaching tasks from the fully stretched down posture, the more uncomfortable the motion is. It can be observed that, for comfortable reaching motions (left-down region of the figures), the dynamics related cost function has less contribution while the kinematics has higher, compared to the uncomfortable reaching tasks (right-up region of the figures) where the opposite trend is observed. Based on this, we propose a *discomfort metric* by combining the three factors along with their corresponding joint angle limits as

$$Dis = \left(\frac{90 - q_{1,S}}{180}\right) + \beta_1 \frac{q_{2,S}}{180} + \beta_2 \left(\frac{q_{1,Change}}{180}\right), \tag{3.5}$$

where Dis denotes the discomfort value calculated by a linear combination of the three factors by using the weights β_1 and β_2. Then for a given pair of weights (β_1, β_2), a set of discomfort values can be derived for all 108 reaching tasks Dis_i $(i = 1 \ldots 108)$. Each discomfort value has its corresponding contribution value of the dynamics related cost function C_i $(i = 1 \ldots 108)$, hence a simple linear least square regression model can be created from the data set (Dis_i, C_i) $(i = 1 \ldots 108)$ as

$$y = \theta_1 + \theta_2 x. \tag{3.6}$$

By changing the weights, different linear regression models $y_{\beta_1,\beta 2}$ are obtained. The coefficient of determination [145] R^2 for each model is given by

$$R^2_{\beta_1,\beta_2} = 1 - \frac{\sum_{i=1}^{108}(C_i - y_{i,\beta_1,\beta_2})^2}{\sum_{i=1}^{108}(C_i - \bar{C})^2}, \tag{3.7}$$

where C_i is the actual contribution value, y_{i,β_1,β_2} represents the calculated contribution value from the linear regression model $y_{\beta_1,\beta 2}$, $\bar{C}$ is the mean value of C. R^2 measures of how well a model can represent the data, and falls between 0 and 1. The higher the value of R^2, the better the model is at predicting the data. Therefore, the optimal pair of the weights is derived by maximizing R^2

$$(\beta_1^*, \beta_2^*) = \max_{\beta_1,\beta_2} R^2_{\beta_1,\beta_2}. \tag{3.8}$$

Solving Eq. (3.8) with respect to the contribution of the dynamics yields the optimal weights as $\beta_1^* = 0.8150$ and $\beta_2^* = -0.4477$. By using the discomfort values derived with this optimal weights, the contribution of the kinematics related cost function can also be explained. Corresponding results are presented in Fig. 3.7c - 3.7d.

Since human motor control is considered as a stochastic system and we do not know exactly how these factors are combined (e.g. linear or non-linear), the discomfort metric presented here is a proof-of-concept of the transition between different reaching tasks. Due to the absence of the description of the variance, the results contain noise, but the trade-off between the dynamics and the kinematics is still observable. This finding supports the idea to use a GPR model to describe the mapping from the initial and final joint angle configurations to the optimal weight vector.

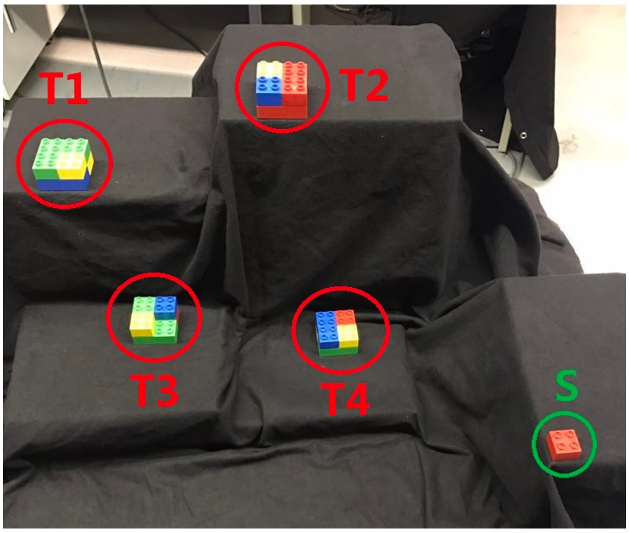


Figure 3.8: Experiment for the hybrid online prediction framework. S is the starting position and T1 to T4 are the four target regions. Each region consists of four possible placing positions as four corners of a square for the LEGO-bricks.

3.4.2 Experiment for hybrid prediction framework

In this subsection, an experiment designed to test the performance of the proposed hybrid online prediction framework is presented. The experiment is based on a simple pick-and-place task with one picking position and four targets. The accuracy of the ProMPs predictions as well as the updating process are analyzed here.

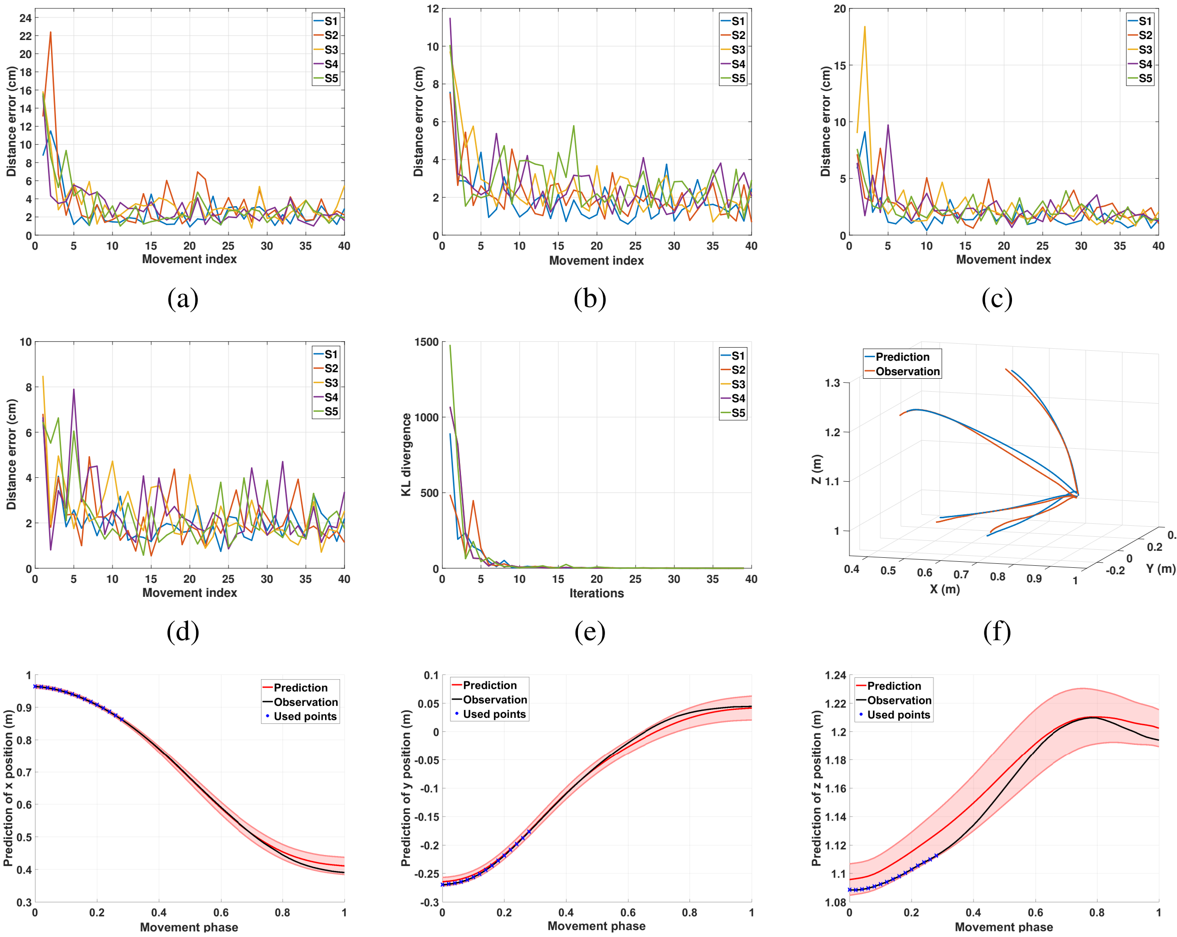
3.4.2.1 Experimental setup and data collection

As shown in Fig. 3.8, the experiment is designed as a pick-and-place task with LEGO bricks. The picking position is fixed during the experiment, and four placing regions with different heights are selected as targets. Each region consists of four possible positions as four corners of a square for placing the bricks. Experiment includes 16 pick-and-place movements (4 targets × 4 times) per subject. Every subject is required to repeat the whole experiment ten times, thus in total 160 trajectories, 40 for each target, are recorded for one subject. We collected the data from five subjects and performed the analysis on those 800 trajectories. We neglect the hand and finger movements and only predict the position of the wrist joint.

3.4.2.2 Results of the hybrid prediction framework

Here we present the corresponding results from the prediction experiment. First the prediction accuracy of ProMPs is tested by looking into the distance error between the prediction and the observation. Then the updating process for the GPR model is analyzed both to provide the evidence on the interpersonal variance, and also to demonstrate the ability of our hybrid prediction framework in describing this variance.

Performance of the predictions by ProMPs We conduct an offline analysis to investigate the performance of the ProMPs based predictions more in depth. After initialization, the ProMPs



(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

(i)

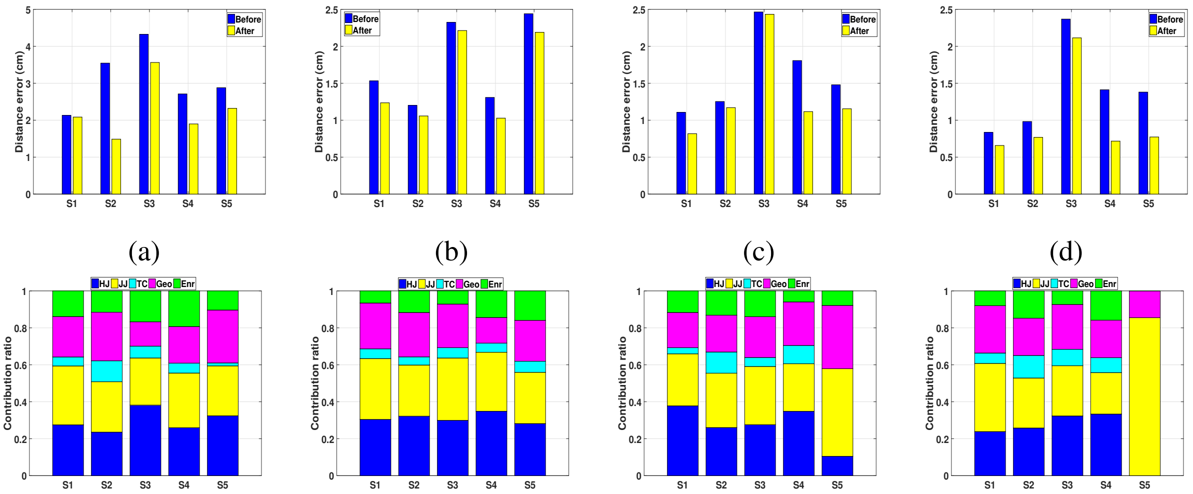
Figure 3.9: Results of predicting with the ProMPs. (9a) - (9d) Distance error between the observations and the ProMPs predictions of five subjects for target one to target four. The errors converge after several updates. (9e) KL-divergence of comparing the updated distribution with the previous one for target one. It can be observed that after ten iterations the value is quite small, which indicates that the distribution converges. (9f) The ProMPs predictions and the observations in the last iteration of subject one for all four targets. (9g) - (9i) The ProMPs predictions in the last observation of subject one for target one. Each plot presented the mean and the variance of x, y, z positions in Cartesian space respectively.

are utilized to generate predictions for the observations. For each observation, we use the first 30% of the observed points to roll out the prediction, and the distance error between the prediction and the observation is measured through DTW. After each prediction, the observation is used to update the ProMPs in order to learn the variance as well. For the next observation, the updated ProMPs is then used, and this updating process keeps running until the last observation.

The distance errors for each subject and each target are presented in Fig. 3.9a - 3.9d. The distance error is calculated between the prediction and the observation. Note that, this comparison is performed in Cartesian space, while during the initialization of the ProMPs, the trajectory generated from the composite model is a relative end-effector trajectory in arm model coordinate system (see section 2.2.4.3). Since the relative end-effector trajectory ignores the shoulder translations and the torso movements, which are not avoidable in real reaching motions, and the model's arm length is usually different than the actual arm length of the subject, the first prediction has large error. However, this initial error diminishes by later updates, and after several updates (around 5) the distance error becomes stable with a small value (around $2cm$ to $4cm$ for trajectory distance error averaged over the data points). In the end, as shown in Fig. 3.9f, the predictions get closer to the observations for each subject.

During the prediction process, the variance is also learned by updating the ProMPs. We initialized the variance to a large value, and observe that after several updates the ProMPs converges to a stable distribution. Fig. 3.9e shows the Kullback–Leibler (KL) divergence of comparing the updated ProMPs distribution with the previous one for target one. The results indicate that after around 10 iterations the distribution converges for each subject. An example of the learned distribution, which is defined by the mean values and the corresponding variances for each point in all dimensions, is presented in Fig. 3.9g - 3.9i. Hence, the motor variability is captured by person specific distribution in the ProMPs. Subsequently, the mean trajectory from the distribution is treated as the average behavior of that specific subject for the corresponding reaching task.

Updating the GPR model Due to the limited amount of available training data, the mapping represented by the GPR model is not accurate enough. Besides, because of the interpersonal variance, the error between the estimated weight vector and the actual one can be large in some cases. Thus we need to update the GPR model through a separate updating process. To do this, we first extract the mean trajectory from the converged ProMPs learned from 40 observations, and then apply the same IOC calculation on this trajectory to get a new weight vector. This new weight vector is used to update the GPR model. Note that, since we also want to model the interpersonal variance, the GPR model is updated separately with respect to each subjects' behavior. A comparison of the distance error between the observation and the optimal trajectories solved with the previous weight vector and the new weight vector is presented in Fig. 3.10a - 3.10d. As we only want to look into the distance error caused by the weight vector, the trajectories compared here are the relative end-effector trajectories, which have less error due to ignoring the shoulder translations and the torso movements. The results indicate that the error diminishes after the update. After several updates on the GPR model, the interpersonal variance can be represented in each person specific GPR model. We also observe that even for the same tasks the new weight vectors vary between different subjects (Fig. 3.10e - 3.10h). This supports the existence of the interpersonal variance while emphasizing the importance of this updating process in our framework.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3.10: Results of the GPR model updating process. (10a) - (10d) The distance error between the optimal trajectories solved with respect to the initial weight vector and the updated weight vector for target one to target four, respectively. (10e) - (10h) The contribution of basic cost functions calculated from the mean trajectories of five subjects for target one to target four, respectively.

3.5 Discussion

Facilitating efficient and safe co-existence of humans and robots is a multifaceted challenge. In this work, we focus on developing a human motion modeling and prediction framework that can be effectively used for robot control during dyadic interaction. One of the key insights of this work is that the interpersonal difference is not negligible regarding the contribution of cost functions. Even though motor variability was acknowledged in previous studies and some stochastic optimal control formulations were suggested as models for the motor control functionality of the CNS, the interpersonal variance has not been studied in such detail. The research presented in this work is a first step for combining model-based and probabilistic data-driven approaches in order to look into this topic, especially from the perspective of how this can be used for human-in-the-loop robot control. In essence, the hybrid framework enables personalized modeling and prediction of human motion behaviors, which can be integrated into robot control to provide personalized, safe, and efficient assistance to the human partner. However, there are still many aspects that need further investigation both for human motion modeling and its effective integration on robot control.

On the human-in-the-loop robot control and HRI As robots have become ubiquitous in our daily lives, the goal is to provide safe yet natural interaction between human-robot dyads. To this end, novel robot control architectures which take into account human motion behavior are required. As robots are expected to adapt their motion behaviors with respect to their human counterparts, understanding how humans control and execute their motions is critical. The outcome of human motion modeling is two-fold: On the one hand, the models learned can be used to predict human motions during interaction so that the robot can take proactive actions. On the other hand, such models enable building robot control architectures for realizing human-like motions to provide natural interaction. The proposed hybrid framework focuses on the former, and it also lays out the underlying control mechanism for human motor control while demonstrating the trade-off between kinematic and dynamic properties used for arm reaching control. Even though there were recent studies on transferring such optimal control formulations learned from human motion data to robot control (e.g. locomotion [57], reaching motion [9]), our findings would enhance such methods by building adaptive control methods to achieve a similar trade-off as human motor control seems to utilize.

The model-based optimal control formulation can further be utilized for other HRI settings, e.g. in physical HRI to provide the required assistance by the robot to the human partner in order to reduce the effort spent by the human which can be detected from the increase in dynamics related costs contribution. In addition, the trade-off analysis can be extended to understand how reciprocal influence of partners' movement affect the cost distribution, which in turn help us construct suitable control and motion planning strategies for the robot to provide optimal assistance constrained on similar cost distributions.

As humans collaborate with each other naturally and safely in close proximity, we hypothesized that one crucial requirement for dyads is to be able to estimate the collaborating partner's motions. In that regard, it is also essential for a robot to predict the motion of human partners. This prediction needs to be efficient (online-capable) in order to choose actions proactively, and to (re-)plan the motion in a way to realize a collision-free trajectory while still achieving the task. The proposed hybrid framework enables such an efficient prediction as well as an update on the cost combination per person. The ProMP-based human motion prediction component of this work has already been integrated into a stochastic trajectory optimization framework [242]. The efficiency of our motion prediction enables the robot to re-optimize its motion frequently at short intervals while considering the predicted human motion distribution as a dynamic obstacle to avoid. Hence, any changes in the expected movement can still be taken into account to achieve a responsive and safe interaction. Furthermore, since our hybrid architecture also updates personal motion models during interaction, the effect of robot movement on human partner's motion can still be captured, which is expected to increase the accuracy of predictions during the course of the interaction.

In that regard, Interaction Primitives (IPs) [125] and its extension Environment-adaptive IPs (EaIPs) [126] also provide a data-driven approach to predict a human partner's movement and then to plan the robot motion accordingly. As ProMP formulation already builds on the idea of learning a distribution over some demonstrated trajectories, it can also be extended to account for the coupling between two agents by learning a distribution over two persons' trajectories executed during a joint interaction task. Similarly, learning a joint distribution including the environment related features would be a feasible improvement. The learned human motion models can still be fed to the IOC formulation to extract the optimal cost distributions that best describes those interactive movement behaviors. The reciprocal influence of partners on their individual cost utilization poses an interesting research question that can be analyzed from the IOC perspective. Our modular hybrid framework also allows integration of any movement representation that can effectively predict human movement behaviors. In that regard, the IOC formulation can easily be integrated with (Ea)IPs to model, understand and predict human interaction behaviors.

Lastly, one critical issue has to be noted. Since those formulations only rely on data-driven formulations, there is no guarantee on a safe and effective motion generation for the robot, especially in close-proximity interaction scenarios. However, our approach has the potential to utilize underlying cost function distributions learned from human movement behaviors for robot motion generation, which can then be combined with a learning approach to achieve a generalized safe policy. In that regard, we can combine the reachability analysis [146] with our model-based optimal control formulation to ensure the safety when the robot is planning its interaction movement. In essence, by the reachability analysis, the states that lead to an unsafe situation will be eliminated, and the learning process is performed within the safe region [147]. This analysis and the required computations are based on the dynamical model of the system and may not be feasible with the purely data-driven approaches, such as IPs.

Limitations The IOC framework enables the identification of combination of basic cost functions in 3D reaching tasks. The results suggest a trade-off between the dynamics and kinematics related cost functions. With a proper definition of the system model and a set of reasonable cost functions, the IOC framework can be generalized to other problems, e.g. locomotion planning [57], car-driving [148]. However, there are several limitations of the IOC framework, one of which is the the complexity of finding the global minimum. Even though we tried to cover an extensive search range of the weight vector, the result is arguably still an approximation of the global minimum. Due to the complex non-linear formulation of the IOC framework, no efficient method has been proposed on addressing this problem yet. Second, the lack of the description of variance weakens the accuracy in terms of modeling the motion behavior. Since the IOC framework results in a deterministic solution, it cannot consider the interpersonal variance and the motor variability during the optimization. When we represent the trade-off between kinematics and dynamics related costs regarding the reaching tasks, the variance makes it hard to identify a clear relationship. Therefore, the discomfort metric we proposed is a proof-of-concept, and a deeper investigation is required to uncover how exactly the motion parameters affect the contribution of basic cost functions.

In the proposed hybrid prediction framework, we combine a model-based prediction method with a data-driven method. A GPR model is used to represent the mapping from the initial and final conditions to the optimal weight vector. However, due to the limited amount of data, the GPR model is not sufficient for representing the variance in motion behavior. It is also found to be effective only when the reaching motions are in the descriptive range of the training data. For prediction purpose, we use the trajectory obtained from the composite model to initialize the ProMPs. The reason we want to include this initialization phase other than directly using the ProMPs is that the subsequent updates on the composite models are much faster than solving the IOC problem from scratch for each person (e.g. 100 upper level optimization iterations take around 4 hours vs. 15 iterations take around half an hour). It also allows to make the prediction immediately without extra data collection. Note that, because of the fact that the arm model ignores the shoulder translation and the torso movements, which are not avoidable in real reaching motions, the current initialization process still has some errors. If a full upper-body model is considered in the IOC framework, this error could be minimized. However, this will immensely increase the computational load, hence this extension may not be feasible.

3.6 Conclusion

In this work we investigate the underlying principles of human reaching motions and propose a hybrid framework to utilize our findings in motion prediction. To uncover the criteria of the reaching motion control, we implement an inverse optimal control framework to identify the contribution of basic cost functions which can best represent the human behaviors. The IOC results indicate a trade-off between the dynamics and kinematics related cost functions depending on the reaching tasks. Then to apply the composite cost function for predicting human motions, we combine the model-based optimal control formulation with the data-driven probabilistic movement primitives method. With this hybrid prediction framework, we learn the motor variability as well as the interpersonal variance at the same time. The demonstrated high accuracy and efficiency of this hybrid framework encourages its usage in HRI settings. For human-in-the-loop robot control, a high-level planner for the robot can exploit such a hybrid model to choose its next task, plan a collision-free motion trajectory, and as a result achieve safe, efficient, and natural dyadic interaction with the human partner.

Part II

Modeling Human-Human Interaction Behavior

4

An Ontology for Human-Human Interaction and Learning Interaction Behavior Models[1]

Robots are expected to possess similar capabilities that humans exhibit during close proximity dyadic interaction. Humans can easily adapt to each other in a multitude of scenarios, ensuring safe and natural interaction. Even though there have been attempts to mimic human motions for robot control, understanding the motion patterns emerging during dyadic interaction has been neglected. In this work, we analyze close proximity human-human interaction and derive an ontology that describes a broad range of possible interaction scenarios by abstracting tasks, and using insights from attention theory. This ontology enables us to group interaction behaviors into separate cases, each of which can be represented by a graph. Using inverse reinforcement learning, we find unique interaction models, represented as combination of cost functions, for each case. The ontology offers a unified, and generic approach to categorically analyze and learn close proximity interaction behaviors that can enhance natural human-robot collaboration.

4.1 Introduction

As robots become ubiquitous in many aspects of daily life, e.g. in service robotics or manufacturing, close interaction with humans becomes a necessary functionality. Humans easily interact in shared and confined workspaces. They adapt to different interaction behaviors without compromising safety and efficiency (Fig. 4.1). Especially in manufacturing, effective interaction in a shared workspace can increase productivity, and hence, is a desirable capability for robots. However, in contrast to humans, robots still need to have a safety distance from human collaborators due to their incapability to understand human interactions. How can we achieve a non-verbal communication directly through movement that results in a natural and seamless interaction between humans and robots? For such an intuitive human-robot interaction (HRI), the robot has to (*i*) accurately predict

[1] This work has been submitted to and under review in the following publication: [240]

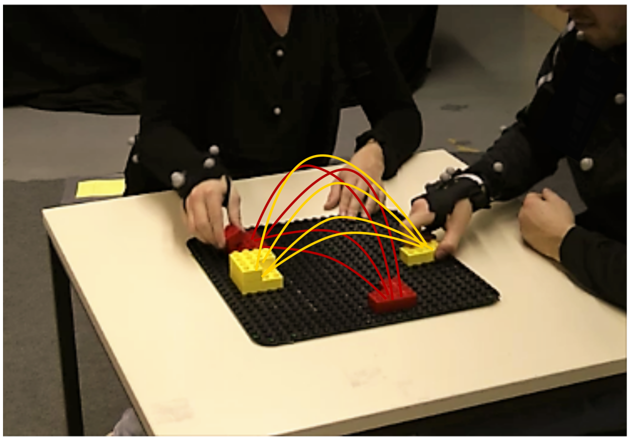


Figure 4.1: Example of a human-human interaction scenario, in which two collaborators need to adapt to each other's arm motion. The possible options for the collaborators are showcased with sample trajectories.

the human motion, and (*ii*) execute a natural responsive behavior, such that the human collaborator can easily predict the motion of the robot and adapt to it. Human movement features and the corresponding control policies during dyadic interaction can provide useful insights, since humans easily fulfill these two requirements [149]. In that regard, human-human interaction (HHI) studies are crucial to get a thorough understanding of interaction behaviors. In this work, our aim is to understand HHI by developing an ontology framework that defines the types and properties of interactions, as well as the relationships between them. Such a descriptive model can also be transferred to HRI scenarios to provide a seamless dyadic interaction.

There have been studies in neuroscience and psychology that investigated important factors for HHI. However, these studies, mainly on visual attention [28], synchronicity [149], and motion prediction [150] have specific task setups, and have rarely investigated interaction in a shared workspace with complex tasks. A recent study extensively described and analyzed the factors influencing HHI [149]. However, a systematical analysis of HHI in close proximity from the perspective of human motion behavior is still missing. In addition to descriptive analysis, there have been studies that predict and imitate human motions for predefined specific setups [236], [151]. A recent work by Mainprice et al. [27] accurately generates motions for an HHI scenario, which were comparable to real human motions. However, it remains unclear, whether their approach generalizes towards other setups and dyad positioning. Evidently, the challenges for describing and modeling HHI have been tackled in several fields in isolation. A comprehensive analysis and methods to define and learn models that generalize interaction behaviors have been neglected.

For an exhaustive analysis of close proximity HHI, the key factors from the aforementioned fields of research have to be considered together. Therefore, a generic interaction model needs to be generated that integrates insights from psychology and neuroscience, while generalizing towards a broad range of tasks. Such a formulation facilitates a systematic analysis of human interaction behaviors and enables the construction of a high-level representation of HHI scenarios. This descriptive representation can be used to classify interaction behaviors by analyzing movement related features (e.g. distance metrics). It is then possible to not only learn dyadic interaction models from HHI demonstrations, but also predict human motions during interaction. By using these interaction behavior models together with the motion prediction capability, the robotic agent can interact with human partners effectively.

In this work, human-human interactions, with a focus on shared workspace settings and repetitive tasks, is modeled as a graph based representation. By incorporating insights from attention theory, psychology, and task generalization studies, this graph structure defines the possible dyadic interaction points for a task. In a next step, the edge combinations of the graph were generated and evaluated to identify possible representations. The high number of combinations in this initial graph model was analyzed in terms of their feasibility to form the full ontology representation (20 distinct cases). Then, a reduced ontology was obtained by the quantitative analysis of the HHI experiments conducted for the full model. Realistic test settings had to be designed that represent each ontology case. These test scenarios included a variability in task setup and dyad positioning in order to verify the robustness of the ontology. The design process as well as the data analysis enabled a methodical reduction procedure to decompose initial HHI representation into seven distinct ontology cases.

The results have shown that the reduced ontology cases can be classified with high accuracy. Additional classification analysis was conducted with data unknown to the classifier, which demonstrated the generalization capability of the proposed model in terms human movement behavior. For a subset of the ontology cases, it has also been shown that the ontology representation is invariant to the dyad positioning. Lastly, a unique dyadic interaction behavior policy per ontology case has been learned by inverse reinforcement learning (IRL), enabling the transfer of those controllers to collaborative robotic agents.

This study investigates factors involved in close proximity human-human movement behaviors, and proposes an ontology for dyadic interaction. The main contributions of our work can be summed up as follows:

- We propose an ontology that comprehensively covers human-human interaction cases and is supported by a thorough experimental analysis.
- The proposed ontology provides a general, unified and reusable framework for interaction classification.
- The ontology is augmented with an inverse reinforcement learning (IRL) formulation to model human motion behaviors that enables predicting and imitating them during close proximity HRI.

The rest of the chapter is structured as follows: In section 4.2, we review related work, followed by a model for HHI in section 4.3. From the proposed model, we derive an ontology of HHI in section 4.4. In section 4.5, we introduce the inverse reinforcement learning approach for human motion prediction. The proposed ontology is validated with two experiments, as described in section 4.6. The results of the experiments are presented in section 4.7, and discussed in section 4.8. Lastly, we draw a conclusion and give an outlook on future work in section 4.9.

4.2 Related Work

In this section, we briefly explain the related work both in the field of human-robot interaction (HRI) and several relevant fields for human-human interaction (HHI) in a shared workspace, i.e. human motion prediction, as well as social psychology and neuroscience studies for interactions.

4.2.1 Human-robot interaction

Here, we review recent studies on shared workspace interaction that are closer to the settings analyzed within our work. Learning human motion behavior from recordings is a common approach to model and then predict human motions in an HRI scenario. Hayne et al. [151] presented a cost map approach, in which a voxel grid of costs is constructed. Costs are applied to each voxel, which has already been occupied by a human during a task. The robot can plan a collision free path with the help of the cost map. The advantage of this approach is its simplicity, however, it cannot consider unobserved human behavior. This drawback severely restricts the effectiveness of robot motion planning. In contrast, Mainprice et al. [26], [27] frame the interaction behavior as an optimal control problem, by assuming that the motions are optimally generated with respect to a set of cost functions. They used inverse reinforcement learning to learn such a cost function, followed by a stochastic motion planner for human motion prediction. They have shown the reliability of their approach for a few cases of HHI, and since only one interaction scenario was tested, it is still unclear if this approach can be used for other scenarios. Furthermore, how those cost functions differ from case to case, and whether they can provide deeper insights on interaction features are still interesting open questions. In this work, we analyze multiple interaction scenarios in order to extract some base interaction cases, for which the aforementioned work could be utilized.

The same problem can be solved by considering the human as a dynamic obstacle, and use motion planning algorithms [242], [152], [153]. In those studies, the focus is on planning motion behavior of the robot with respect to the changes in the environment for safety, and hence the resulting behavior is reactive. However, natural interaction does not only require safety and reactiveness but also a reciprocal responsive behavior. In our work, we focus on the analysis of the HHI scenarios to extract the features of such natural, safe yet effective interaction behaviors. In that regard, the proposed ontology will also be useful for those motion planning frameworks.

Another type of dyadic interaction is physical human-robot joint actions, where a human-robot team works on the same object, e.g. carrying a table. The perception and control problem shifts from motion prediction and adaptation to force interactions, which are transmitted through the object. These force interactions have to be interpreted carefully to realize whether they indicate that more resistive force has to be applied [154], or whether it is a signal from the collaborator to switch directions or roles [243], [155]–[157]. In our work, non-physical dyadic motion behaviors in a shared workspace are investigated instead of joint physical actions. This necessitates identification of relevant features for interaction behavior control.

A particular form of joint actions are object handover tasks between a robot and a human collaborator. The difficulty in this task is to predict the hand over pose of the human, since the robot end effector has to adapt to it as well as the intentions during the handover task. At the start, the human, as the receiver of the object, has to adapt to the pose of the robot [158]. This is followed by the robot identifying and reacting to distinct handover poses [159]. Finally, the social interaction between human and robot is optimized such that the intention of the robot to hand over an object becomes clear [160], [161]. We view handover tasks as a distinctive but still important case of HHI, which can be incorporated into our proposed ontology, as will be explained later in Section 4.3.1.

4.2.2 Human motion modeling and prediction

Human motion behavior in an interaction scenario comprises two critical components. On the one hand, the mutual adaptation has to be considered, and on the other hand, the motion related to the specific task of the individual has to be considered. The latter will be reviewed here as the learning and adaptation phases of dyadic interaction are not considered in this study.

Affordance concept is an influential formulation to investigate motion behaviors, and it is mostly used to predict possible actions on an object, while abstracting its utility. Ugur et al. [162], [163] use a continuous object representation in order to compute the effects of an action on an object by computing the difference between the representations before and after the action. The proposed framework was able to reliably predict the effects of certain actions, and also to generate plans if the desired effect is given. In a similar work, Koppula et al. [25], [164], [165] used the affordance concept to predict human motion trajectories of possible actions, which are associated with an object. Affordances were encoded in an energy function that is to be maximized. The reliability of the approach was also demonstrated in predicting human motions within a short time horizon. Even though the feasibility of this approach has been claimed for interaction scenarios if an interaction cost is added to the energy function, it has yet to be shown. Our approach is quite similar to the concept of affordances. Based on the setup of the interaction, we estimate an interaction case that enables generating a feasible interaction trajectory as the prediction. However, the difference is that we are only interested in the interaction process rather than the objects related to it.

Similar to affordances, Semantic Event Chains (SECs) approach has been proposed by Aksoy et al. [166], [167]. They encode tasks as a sequence of graphs, which represent the topology of the task at a given moment. This high-level abstraction enables grouping of similar objects and tasks, such that task learning can be simplified. In this regard, a framework has been proposed in which SECs can be used as a form of manipulation library in order for a robot to execute different tasks [168]. Similarly, Wörgötter et al. [169] abstracted actions, and generalized manipulations according to their topological effects, instead of the objects. They examined the contact between objects during manipulations and derived a small subset of contact situations, which reliably describe almost all possible single hand manipulations. In this work, we make use of their manipulation abstraction, in order to make our interaction model task invariant.

4.2.3 Psychology and neuroscience

Once the task related motion prediction is taken into account, the perception aspect of interaction has to be considered as well. Therefore, it is reasonable to leverage insights from psychology and neuroscience as they investigate the factors involved in interaction. Among a multitude of topics, entrainment or synchronization [149], [170] is arguably one of the broadly studied interaction component. Various studies show that entrainment emerges, even if humans are not instructed to synchronize their motions for dyadic tasks [149], [171].

Another important set of properties for HHI is related to the theories on attention. During HHI, humans have to split their attention between their task and the interacting partner, resulting in an attention distribution. Corbetta et al. [28] explain those mechanisms and show that attention is drawn to the biggest stimuli. However, this trend changes during task executions. Prior information about the task helps the attention system to filter out unimportant stimuli, while paying special attention to task relevant stimuli [172]–[174]. For HHI in a shared workspace, the dominant percept is vision and the dominant stimulus are fast motions that may lead to collisions [175], [176]. For our study, these behavioral properties on synchronicity and attention distribution are also considered for constructing our ontology and interaction models.

This review has shown relevant work in the field of HRI, followed by the two components of HHI in a shared workspace, task related motions and effects, as well as relevant work on human interaction behavior. In the next section, these components are merged into a model for dyadic interactions.

4.3 Modeling Interactions for HHI

For a systematic analysis of HHI in a shared workspace scenario, it is first necessary to determine the components that influence those interactions. Sebanz et al. [149] identifies three key questions that describe these influences: “What happens? Where does it happen? And, when?" These questions are linked to the inference of actions and their goals (*what*), the locations at which the motion or action takes place (*where*), and the timing, in other words, synchronization of actions (*when*). In this section, we use these three questions as a guideline to create an interaction model for HHI. We first explain our insights into the posed questions, followed by a derivation of an interaction model. In order to help with following our explanation, we use the interaction case shown in Fig. 4.1 as a recurring example. This example consists of a pick and place task, combined with a setup that forces the dyad to cross arms during the execution.

What. The question of “what happens" can be answered from two different levels of abstraction. The higher level of abstraction is the intention of the interacting agents, i.e. what they want to do with the task related object(s). There have been several approaches on how to predict human intention, such as affordances [25], [162], or library based methods [168]. In our formulation, we assume that the intention is either already known, or can be correctly inferred at the start of interaction. This is a reasonable assumption, especially for repetitive close proximity interaction tasks such as collaborative manufacturing scenarios. Once the intention is presumed, the next level of abstraction, i.e. how the actors are going to realize their intention by their motion, becomes relevant. Thus, in a next step, the task is split into relevant subtasks, resulting in a sequence of actions. For the given pick and place example, this is the sequence of picking up an object, bringing it to the target position, and then placing it. The partners then uses this information to estimate the influence of each other’s task sequence on their task. This sequential aspect can be further abstracted if the subtasks are generalized as it has been done in [169]. Using the pick and place example, whether the object is picked up or placed down at one of the predetermined locations is insignificant. It just has to be known that a task related event takes place at that location. Since less information about the actual manipulation action needs to be processed, the attention on the interaction, i.e. the relevant task sequences that influence each other’s tasks, becomes critical. This allocation of attention directly leads to the question of where the important interaction sequences happen.

Where. The question of “where does it happen" is again analyzed by two levels of abstraction. The first is the allocation of the shared workspace. Given our pick-and-place example, this is described by which person grasps over the other’s arm, i.e. the rough location each interacting agent can occupy during their next transport motion. In addition to the allocation of space, joint attention is also relevant to the allocation of tasks [149]. For close proximity and repetitive interactions such as collaborative manufacturing scenarios, we assume that the tasks are pre-allocated. The joint attention is mostly on the distribution of space. During preliminary experiments, we observed that this allocation is relatively stable after a short period of adjustment in the early phases of interaction. Hence, we assume that space allocation is negotiated initially and stays mostly the same thereafter. Consequently, attention needs to be paid only to interaction related stimuli [28], [172], i.e. task relevant motions. Discordant movement behaviors, e.g. that are faster than expected or that might lead to collision, increases the partners’ attention to the motion. In essence, timing is tightly coupled to these conditions, and thus, plays a crucial role. Taking our example case (Fig. 4.1), if both actors move into the workspace at the same time, a collision may occur, and more attention has to be paid on partner’s movement. However, if they move into the shared workspace alternately, then clearly the dyad does not obstruct their motion reciprocally.

When. Considering the arguments on *what* happens and *where*, we can argue that, in general, attention is increased towards motions only if they are timed such that they can result in a collision. If the confined workspace is restrictive for joint occupation, it is a reasonable strategy to time the motions such that the space is shared in turns. This approach aligns with the concept of entrainment or synchronization, which expresses the tendency of humans to fully synchronize their motions [170]. Although asynchronous motions within the shared space reduce the mental effort through attention, this strategy may no longer be valid if time efficiency is critical, as in most manufacturing scenarios. Especially, if one partner’s task takes longer, then it becomes infeasible for the other to wait. Taking our pick-and-place example, this would be the case if assembling the brick requires a more complex action such as screwing rather than just plugging in. In essence, for an interaction involving individual tasks, timing is not one of the main determinants but more of a feature of interaction.

4.3.1 Interaction model

In this section, based on the insights given for the posed questions, we propose an interaction model with three assumptions. The model is constrained to a shared and spatially constricted workspace ($A1$). Furthermore, we assume that the interaction is situated in a manufacturing environment ($A2$). This second assumption includes that tasks are known and distributed beforehand, and that tasks are executed time efficiently and repeatedly. The repetition of the tasks ensures that the workspace is usually allocated, and that we can observe an interaction pattern over time, while still capturing the variance of similar motions. Movement patterns are determined w.r.t. the synchronization of the motions, and ensure comparable interaction behaviors. Based on these assumptions, we formulate the task sequence for follow up task executions as a sequence of approaching and withdrawing motions, and an action phase in between, similar to [169]. With the insights from attention theory, we can merge the withdraw and approach phases into a motion phase, since it does not matter what motion type is performed as long as it interferes with another motion. This reduces the task procedure to a sequence of actions and motions.

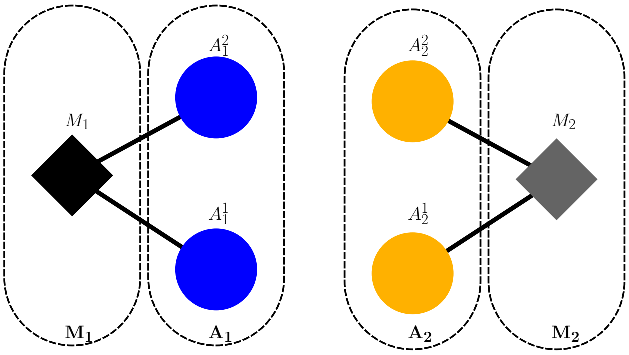


Figure 4.2: Graphical representation of the proposed interaction model. Action $\mathbf{A_i}$ (round) and motion $\mathbf{M_i}$ (squared) spaces are colored separately. Each partite vertex set is framed.

Lastly, we assume that a task is defined by two distinct locations, in which actions take place ($A3$). For a structured analysis, this restriction is imposed on both agents, i.e. the pick-up and placement locations are spatially different in our example pick-and-place task. Furthermore, alongside the reasoning of Wörgötter [169], we model interactions such that the model is task independent, e.g. for a crossing arms movement behavior, the crucial part is the transport motion, and whether it is a pick-and-place task or fixing a stripped screw is irrelevant for the model structure. Therefore, we assume that the micro motions of each action, i.e. placing the brick or screwing, take place in a black-box, which we call *action spaces*. Note that these micro motions are restricted spatially around the manipulated object, and thus most likely to be ignored by the partner's attention system. These action spaces are connected to each other by a *motion space*, for which the trajectory in between both action spaces is abstracted, i.e. the shape of the trajectory is not taken into account. This allows including the variance of individual behaviors, as well as the motion adjustments due to environmental influences, e.g. obstacle avoidance, into our model.

Using the above reasoning, we create a graph based model of interactions. Two fully connected bipartite graphs $K(2,1)$ are combined into a 4-partite graph with the following vertices and edges:

$$\mathbf{G} = \{\mathbf{A_i}, \mathbf{M_i}, \mathbf{E}\},$$
$$\mathbf{A_i} = \left\{A_i^1, A_i^2\right\}, \quad \mathbf{M_i} = \{M_i\},$$
$$\mathbf{E} = \left\{\left\{A_i^1 M_i\right\}, \left\{A_i^2 M_i\right\}\right\}, i \in \{1,2\}$$

Here, $\mathbf{G}$ represents the interaction model for two actors, $\mathbf{A_i}, \mathbf{M_i}$ the disjoint sets of vertices that represent the model, and $\mathbf{E}$ the edges connecting them. The vertices of the set $\mathbf{A_i}$ are called action (manipulation) spaces and represent the blackbox in which actions take place. The vertex of the set $\mathbf{M_i}$ represents the motion trajectory connecting the two action spaces. Hence, a sequence of tasks can be represented by a sequence of vertices of a graph. Furthermore, this graph contains all locations at which an interaction can take place, be it motion or action spaces. A graphical representation of this model can be found in Fig. 4.2. Corresponding to our example of a pick-and-place task, the picking up of the object would be represented by A_i^1, while the placement of the object by A_i^2, and the transport motion by M_i. Based on this model, we will derive an ontology for HHI in the next section, by exploring all possible edge combinations of the interaction model.

4.4 Human-Human Interaction Ontology

The proposed 4-partite interaction model provides us with a tool to systematically analyze HHI. In a first step, all possible edge combinations for the interaction model are generated and analyzed. For this analysis, an interpretation of the individual edges with respect to human interactions is essential to reduce the number of feasible cases. Each edge has to represent a specific interaction between the dyad members during the execution of an arbitrary task. Three types of interaction are specified as follows:

- $E_1(k,l) = \left\{A_1^k A_2^l\right\} \forall k,l \in \{1,2\}$: An interaction between the action spaces of the dyad. We interpret this as a shared action space. Examples for such an interaction would be a hand over task or taking objects from the same stash.
- $E_2(k,j) = \left\{A_i^k M_j\right\} \vee E_2(l,i) = \left\{A_j^l M_i\right\} \forall k,l \in \{1,2\}$: An interaction between an action space of one agent and the trajectory of the other. We interpret this as one actor grasping over the action space of the other. As a result, the respective agent has to avoid the other's action space for improved cooperation.
- $E_3 = \{M_i M_j\}$: An interaction between the trajectories of agent i and agent j. This can be interpreted as cross over motion, where agents' trajectories intersect. Therefore, the agents have to avoid collision with their partners (Fig. 4.1).

The combination of these edges results, i.e. three possible edges per node, makes a total of 9 possible edges, resulting in a total number of $N = 2 + \sum_{p=1}^{8} \frac{9!}{p!(9-p)!} = 513$ interaction scenarios. The total number N is computed by either successively adding edges to the unconnected graph, or by fully connecting all vertices, and then eliminating individual edges. In order to reveal the reasonable cases, the following simplifications and reductions are applied in the form of edge equivalences:

- First, all edges violating model assumption *A*3 (see Sec. 4.3.1) are eliminated. All graphs containing the edge set $E = \{E_1(1,l), E_1(2,l)\} \vee \{E_1(k,1), E_1(k,2)\}$ violate this assumption and are therefore omitted. According to the edge interpretation, two action spaces from the same partition are merged, and thus eliminate the necessity of a motion.
- A similar problem arises, when two action spaces are merged, i.e. when an edge E_1 exists. If there is an additional incident edge E_2, an agent would grasp over their own action space. This can as well be seen as part of the grasping process, and thus merged into the action space. Therefore, all graphs containing the edge set $E = \{E_1(k,l), E_2(k,j)\} \vee \{E_1(k,l), E_2(l,i)\}$ are omitted.
- Besides edge reductions due to the model definition, symmetry can also be used to reduce the number of cases. Since initial and final points are insignificant for the model, and the positions of the action spaces can be set arbitrarily as long as they fulfill the edge interpretations, symmetry of an agent's action spaces can be assumed. Therefore, based on the assumption that $A_i^1 \simeq A_i^2$, and $A_j^1 \simeq A_j^2$, a graph $\mathbf{G} = \{\mathbf{A}, \mathbf{M}, \mathbf{E_x}, E_2(1,j)\}$ is isomorph to a graph $\mathbf{H} = \{\mathbf{A}, \mathbf{M}, \mathbf{E_x}, E_2(2,j)\}$, if they share the other set of arbitrary edges $\mathbf{E_x}$. As a result $\mathbf{G} \simeq \mathbf{H}$ holds.
- Since the relationship between dyad motions is investigated, models with symmetry in mo-

tion spaces can be discarded, i.e. whether agent i or agent j grasps over the other's action space is not critical. In combination with the previous assumption, $E_2(k, j) \simeq E(l, i)$ holds true for any indices k, l.

Based on these simplifications, and symmetry assumptions, the number of feasible cases could be reduced to 20 interaction cases (Fig. 4.3a), which represent the *full ontology*. Exemplary setups for each of these 20 interaction cases are illustrated in Fig. 4.3b. Obstacles are introduced for some cases that enforce interaction variations in order to extend the analyzed dyadic movement behavior set. If those obstacles are removed, the interaction would change (as can be seen for case 18, which becomes very similar to case 16, if the object is removed), since the blue agent is no longer forced to grasp over one of the yellow action spaces.

4.4.1 Ontology reduction

As it can be seen in the examples, some interaction cases of the *full ontology* still look similar. Based on the theoretical and heuristic deductions as well as an experimental verification, the full ontology of 20 cases could be reduced to 7 cases, which are called the *reduced ontology* for the rest of the chapter. For the theoretical deduction, the degree-of-freedom of action and manipulation spaces are observed. An action or manipulation space is assumed to have a degree-of-freedom, if the corresponding vertex has no edges from the set $E = \{E_1, E_2, E_3\}$. For the reduction, the following cases were merged, and a proto-case is highlighted in bold for each reduced case:

- Zero degree-of-freedom cases: **10**, 12, 17, and 18. These are the ontology cases which are constrained the most. Due to the high number of E_2 interactions, the motions of each agent are thoroughly restricted. Both agents have to grasp over two action spaces, or have a shared action space, which limits the possible trajectories to the one seen in the examples in Fig. 4.3b. Such setups are necessary to enforce E_2 interactions. Due to the constraints of those setups and the elbow kinematics of a human, there is barely any motion redundancy left, making these cases some of the easiest to predict. Furthermore, the kinematic restriction of the elbow on one side of the workspace enforces a strict timing between agents to work efficiently. Consequently, this case restricts the motion of the agents the most, due to its high number of action space interactions. This stands in contrast to cases 19 and 20, which are are also restricted in their action spaces, but have less restricted trajectories due to the absence of E_2 interactions.

- One degree-of-freedom cases: **7**, 11. These cases are similar to case **10**. The only difference is that one agent has more options for one of their action spaces. This results in a relaxed trajectory planning for one agent, and thus a broader range of interactions. This can also be seen in the possible variability of the experimental setup of these cases.

- Two degree-of-freedom cases: **5**, 6, 9, 13, and 19. These cases have a similar spatial configuration to the extent that either one of the action spaces of an agent is in close proximity to one of the other agent's, or that they share an action space. For cases 5 and 9, the close proximity of two action spaces is necessary in order to ensure the two action and motion space interactions. Additionally, the merged cases require asynchronous motions. For cases 13 and 19, this is necessary due to the shared action space, which limits the space for executing the action, and as such should be used in sequence. For cases 5 and 9, an in-line placement of the

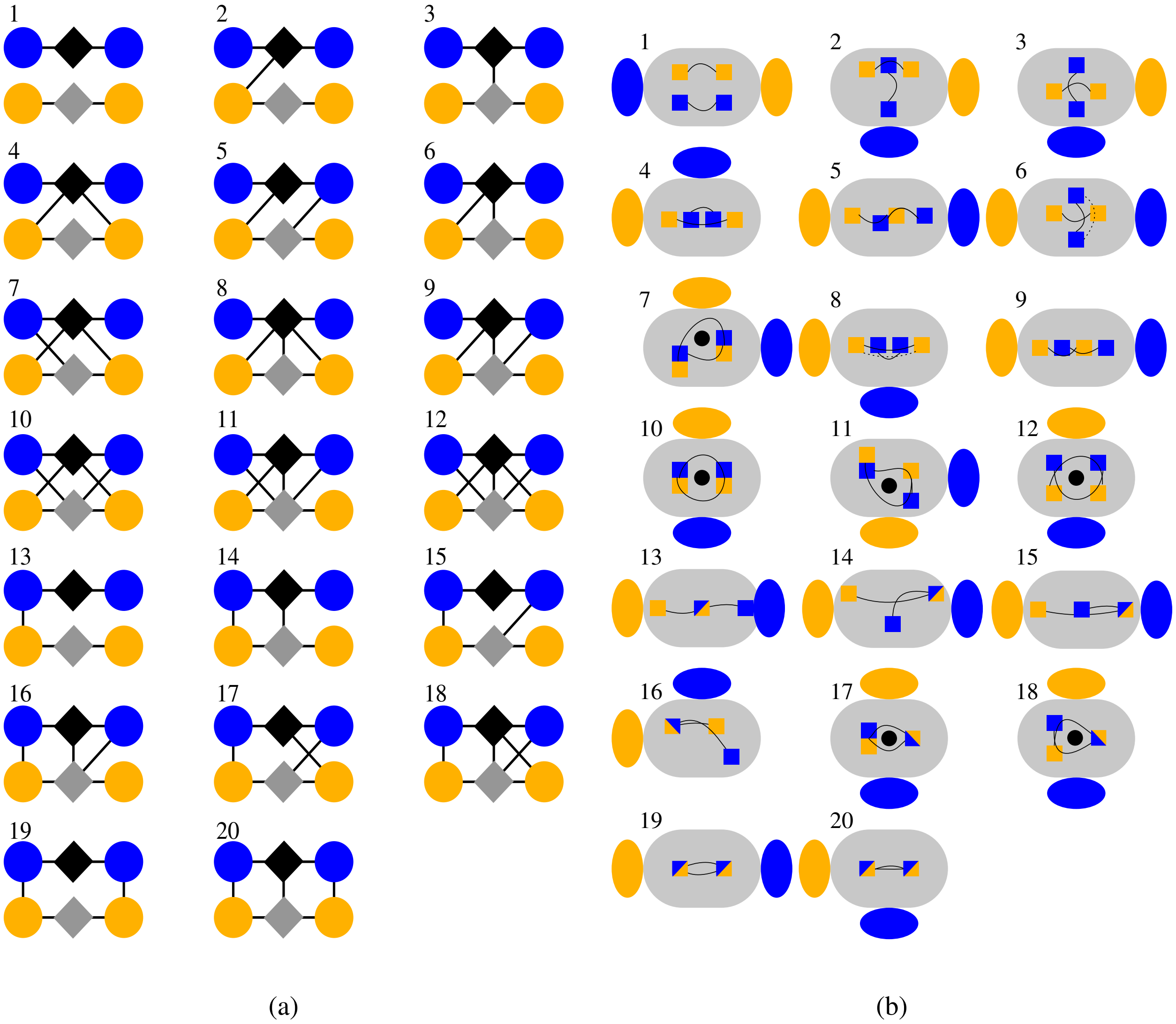


Figure 4.3: (a) Graph representation, and (b) exemplary setups for each case in the *full ontology*. (a) Action (round) and motion (squared) spaces are colored separately. (b) Gray area depicts the shared workspace, while the black circle represents an obstacle blocking the path, and dashed lines show the motion of the elbow, where necessary.

action spaces is required to realize these scenarios, if no obstacles are introduced (Fig. 4.3b). Otherwise, they will be similar to cases 7 and 11. This in-line placement leads to one person partially obstructing the other's action space. In essence, asynchronous motions are favorable, which is similar to shared workspace cases such as case **2**. Lastly, there is a second shared workspace in case 19, however, it does not impose any additional restrictions, since asynchronous behavior is already a favorable option.

- Two degree-of-freedom cases: **4**, 8, 14, 15, 16, and 20. These cases are similar due to their spatial configuration. All of them have one person grasping over the other's both action spaces, while having no interaction on their own action spaces. Their dissimilarity lies in the crossing of motion spaces in case 8. However, since the motion space has to lie in between the action spaces, there will be no influence of this interaction possibility on the overall interaction. The first person always grasps over both action spaces, and therefore, the inclusion of an additional motion space does not change the overall behavior, while the second person is mostly working in the other's motion space, which stays the same in case 4 and 8. Considering case 15, one of the action spaces is merged, making it a shared workspace equivalent of case 4.

4.5 Human Interaction Behavior Modeling

In this work, the idea is to reconstruct a cost function, by learning from human-human interaction movement data. This motion data is available in the form of a time parametrized trajectory ξ in a configuration space, which can be defined as follows:

$$\xi = [q_1 \dots q_N]$$

where q_i are the vectors of configuration, and N denotes the number of waypoints. In this work, we consider a 14 dimensional kinematic model of the human arm. It is assumed that human motions are optimal with respect to an unknown cost function [235], [59], [82]. This hidden cost is approximated as a linear combination of user defined features $C(\xi) = w^T \Phi(\xi)$. In order to solve the inverse problem of finding the optimal weighting for this cost function, normal inverse optimal control methods are intractable due to the high dimensionality. We use the sampling based approach of path integral inverse reinforcement learning (PIIRL) [95]. PIIRL assumes that the cumulative cost $C(\xi)$ function is composed of a general cost term G, and a control cost term A, which enforces smoothness, i.e. $\Phi(\xi) = [G(\xi), A(\xi)]^T$. The general cost term has the following form:

$$G(\xi) = \int_{t=0}^{T} \phi(q_t)\, dt \simeq \sum_{i=1}^{N} \phi(q_i)\, \delta t$$

PIIRL samples trajectories with low smoothness cost around each demonstration in order to reconstruct the weights of the cost function. The sampling distribution is defined using Multivariate Gaussians, $\mathcal{N}(\xi_d, \sigma \mathbf{R}^{-1})$, centered at each demonstration ξ_d, where $\mathbf{R}^{-1} = K^T K$, and K is a matrix of finite differences that computes time derivatives. In this work, we used the finite difference matrix of the original PIIRL algorithm [95]. The weights of the cost function can be found by solving the following minimization problem:

$$w^* = \underset{w}{\operatorname{argmin}} = -\sum_{i=1}^{D} \log \frac{e^{-w^T \Phi_i}}{\sum_{s=1}^{S} e^{-w^T \Phi_{i,s}}}$$

where D is the number of demonstrations, S the number of samples around each demonstration, and Φ_i, $\Phi_{i,s}$ the respective feature counts. These weights can be used within a stochastic trajectory optimizer to generate feasible and predictive human motion trajectories [27].

With this formulation, control policies are found by learning human movement behaviors from HHI demonstrations. The proposed ontology categorizes expected interaction behaviors, which in turn allows learning a distinct policy per case. Analysis of the acquired policies informs us about the possible physiological and behavioral strategies used in dyadic interaction. Next, the details of the experiments, during which the interaction behavior data was collected, is explained.

4.6 Human-Human Collaboration Experiments

In this section, the experimental setup used to verify our ontology representation is explained. We conducted two separate experiments for verification. The first experiment used the ontology cases with the most degrees of freedom of motion, in order to verify whether the ontology is feasible, and also invariant to the relative positioning of the dyad. The second experiment was designed to test the rest of the ontology cases, while relaxing some of the controlled parameters of the first experiment. In this section, we first introduce the task selection for both experiments, followed by an explanation of the positioning selection of the participants. Lastly, we describe both experiments.

4.6.1 Task and collaboration case selection

The goal of both experiments was to analyze interaction processes between humans sharing a confined workspace, and therefore, an adequate task has to be chosen. The task should be simple, while at the same time generalizable. The simplicity of the task ensures that the main focus of participants lies on the interaction instead of the task itself. The generalization capacity of the task extends the application of the results to a broad range of use cases. As a result of those considerations, we chose a pick-and-place task due to the fact that it is a routine activity and also part of almost all manufacturing tasks [169]. With this task selection, the collaboration focus is only restricted to the interaction trajectories.

4.6.2 Relative positioning

Once the task is set, the next step is to define the relative positioning of the participants. In most interaction experiments, the participants are placed opposite to each other such that each participant can fully view the other participant's motion. However, in real-life scenarios, this may not always be the most feasible setup. An exploration of viable relative positions is necessary in order to verify its presumed influence on the interaction.

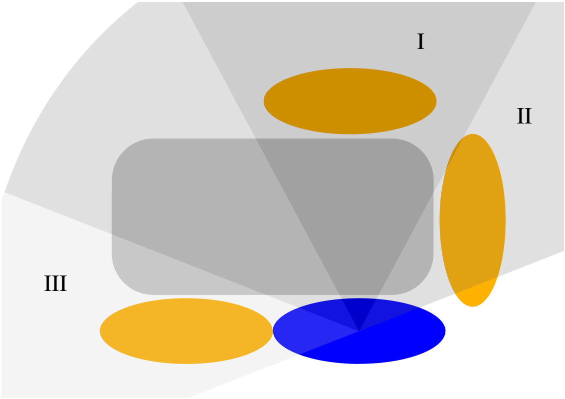


Figure 4.4: The figure shows the analyzed relative dyad positioning, frontal (I), orthogonal (II), same side (III). The circular sectors represent the respective sections of the field of view.

Positioning the interacting partners opposite to each other ensures that each of them are in the central field of view of the other. Hence, visual distinction of motions, and objects is maximized, which in turn, improves the reaction times. However, this capability of the human declines with each degree of eccentricity in the human field of view [177], [178]. For studying the effects of this change on human interaction behavior, we decided to evaluate three different sections of the field of view for the experiments: (i) central, (ii) peripheral, and (iii) extreme peripheral field of view. In the central field of view, the recognition of objects and motions is at its best, whereas the effectiveness of recognition declines in the peripheral field and vanishes in the extreme peripheral region [178]. Unlike object recognition, motion directions can still be recognized even in the extreme peripheral vision, which is argued to be due to the threat detection instincts [179]. In order to test this hypothesis within our experiment, we decided to position participants in three different configurations (Fig. 4.4).

4.6.3 First experiment

The first experiment was designed to test two hypotheses:

- $H1$: Ontology cases are distinguishable with respect to the features extracted.
- $H2$: Ontology cases are invariant, in terms of the features, to the relative positioning of the dyad.

We used the ontology cases with the most degrees of freedom in setups, which are cases 1, 2, and 3. These cases can be setup for any relative positioning, since they impose no constraints on the relative positions of the action spaces with respect to each other. The positioning invariance was tested for the relative positions I, II, and III (Fig. 4.4), resulting in a total of nine different setups for the experiment. The initial and final positions of each task were fixed in order to ensure that a different placement does not skew the results. One experimenter was also designated as one of the dyad for all the trials for providing steady and fluid behavior. This ensured comparable and consistent interaction behaviors by the subjects as well as the possibility to interrupt synchronous behavior -intentionally by the experimenter, if required- that broadens the spectrum of the timing behavior.

The experiment was conducted as follows: Experimental procedure as well as the task of the participant were explained to the subject, and the participant signed a form of consent. Next, the tracking markers were fit onto their body. At the start, a habituation phase of one minute was provided, where the experimenter and the participant executed their tasks for random setups, and the experimenter forced some interactions, in order to familiarize the participant with the task as well as the experimenter. Afterwards, the experiment was conducted in three sections, with three interaction cases each. Each section contained the three selected ontology cases from a different relative position. These sections were randomly selected. The cases in the section were arranged such that the free motion case 1 was always selected first, followed by a random order of cases 2 and 3. Free motion case (case 1) was set first as it does not enforce any interaction explicitly, and thus helps participants to get familiar with each relative position. Each of the interactions took place for exactly one minute, in which the participant and the experimenter should move lego bricks that formed a tower-like structure from one position to another and vice versa as often as possible. An effective movement speed was encouraged, in order to guarantee interactions, and to avoid waiting of participants for each other. Between each interaction case, there was a short break of half a minute. The experiment has been conducted with ten healthy participants, of which nine were male and one female of ages 25.3 ± 2.5.

4.6.4 Second experiment

In the second experiment, we tested the rest of the missing ontology cases, while relaxing the fixed setup constraint. Position invariance was not investigated as it is verified with the first experiment (Sec. 4.7) and also some of the more complex cases are only viable for a single setup. We used the relative positions I and II, and only tested the first hypothesis ($H1$) on classifying ontology cases between 4 and 20 in this secondary experiment. Those interaction cases were split into two groups, which were tested with a break in between. The first group consisted of cases without shared action spaces, i.e. cases 4-12, while the second consisted of the remaining cases, i.e. cases 13-20. The order of the cases within the group was randomized for each participant, while the order of the groups was kept fixed. This was preferred since the shared manipulation cases require some degree of synchronization, which gets easier once the participant is more familiar with the task and the partner. The second participant was again an experimenter in order to ensure a consistent natural interaction. The setups for each recording can be seen in Fig. 4.3b. The procedure of the experiment was similar to the first experiment. This second set of experiments has been conducted with 8 healthy participants, of which seven were male and one female of ages 25 ± 2.4.

4.6.5 Recording and preprocessing

The recording was done by using the optical Qualisys tracking system with eight cameras. The participant and the experimenter were equipped with optical trackers positioned at the thorax, back, shoulder, elbow, wrist and hand. It was attempted to place the markers at the following anatomical landmarks: one at the xiphoid process, one at the Sternum, two randomly along the vertebra, two at the front and back of the glenohumeral rotation centre, one at the elbow, one at the radial and ulnar styloid each, and one at the back of the hand. In the second experiment, two trackers were attached to the elbow for improving the tracking accuracy. Data was first labeled and corrected of erroneous trackings, before being exported to a Matlab file. Further processing of the data has been conducted in Matlab.

The data was recorded with a frequency of 250HZ (∼15000 frames/recording). In a first step, the recording was segmented automatically, by detecting points at which the absolute velocity of the wrist was close to zero. Resulting segmentation errors were corrected manually. Subsequently, a 14 DOF arm model was used to reconstruct the joint angles of the model. Missing data points were interpolated linearly, and the resulting angles were filtered with a first order Savitzky-Golay filter. This filtering also helped to compensate for noise and tracking errors. In a next step, the data was downsampled to 50Hz. A total of 8589 valid movement segments is extracted for both experiments.

4.7 Results

In this section, we present the results for the previously explained HHI experiments. The details of the preprocessing steps, the features used for analysis, and the used machine learning method are provided. Next, we show that the proposed ontology is invariant to the positioning of the dyad for a subset of the ontology cases ($H2$ in Sec. 4.6.3). This is followed by case classification results for the *full* (20-case), and the *reduced ontology* (7-case). Subsequently, we briefly introduce results on the intrapersonal level. Lastly, by modeling ontology cases distinctly as a combination of cost functions as a proof-of-concept representation, an analysis on how such cost distributions change w.r.t. the ontology case is presented.

4.7.1 Features

It is first necessary to find a measure of the interactivity in order to classify interactions. Referring back to section 4.3, the most important factors for the interaction are location and timing awareness, if the task is known. The location property during the interaction can be captured by the pairwise distances between kinematic features of agents. For a similar problem, Mainprice et al. [27] used the pairwise distances between the recorded joint positions. However, this measure is not representative enough, since it does not measure the shortest distance between the limbs, which could also have contact even the distance between the relevant joints is not close to a critical value, e.g. if forearms of agents crosses each other orthogonally. Hence, we also compute the pairwise shortest distances between the limbs as additional features.

The total feature vector has a dimension of 25. For the classification of each case, we used the mean feature values for each segment. The classification was conducted using a support vector machine (SVM) with a quadratic kernel. For training purposes, holdout validation was used.

4.7.2 Location invariance

We used the data from the first experiment to validate the hypothesis ($H2$) on positioning invariance. A SVM classifier is trained for the three ontology cases with a 50% holdout validation,

	1	2	3	4	8	14	15	16	20	5	6	9	13	19	7	11	10	12	17	18
1	89.6	7.2	1.9	0.1	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.1	0.1	0.5	0.1	0.1	0.0	0.0	0.0	0.0
2	8.7	80.9	6.4	0.6	0.2	0.0	0.9	0.0	0.1	1.0	0.1	0.3	0.2	0.7	0.0	0.0	0.0	0.0	0.0	0.0
3	2.6	12.7	77.1	0.0	1.2	1.2	0.7	0.6	0.4	0.5	0.4	0.2	0.1	1.0	0.2	0.1	0.2	0.2	0.3	0.1
4	1.0	12.6	0.2	70.0	6.4	0.3	8.4	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
8	0.2	5.5	1.4	8.6	69.9	4.6	6.1	2.2	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14	0.1	0.5	7.7	0.0	3.8	63.9	3.2	11.5	8.7	0.0	0.0	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.0
15	2.0	11.9	1.0	5.5	5.7	2.2	67.1	0.4	3.0	0.0	0.0	0.0	0.0	0.0	0.9	0.2	0.0	0.0	0.0	0.0
16	0.0	1.8	6.6	0.0	2.5	14.4	2.6	61.1	10.9	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
20	0.7	2.1	4.1	3.5	3.1	5.8	6.1	9.0	64.4	0.0	0.0	0.0	0.0	0.0	0.9	0.3	0.0	0.0	0.0	0.0
5	0.4	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	56.2	3.6	15.9	18.2	2.1	0.0	0.0	0.4	0.1	0.6	0.5
6	0.1	0.7	0.2	0.0	0.0	0.0	0.0	0.0	0.0	2.9	84.9	5.2	2.4	1.6	0.0	0.0	0.6	0.2	0.7	0.4
9	0.7	1.2	1.6	0.0	0.0	0.0	0.0	0.0	0.0	19.4	4.5	51.9	14.5	2.5	0.0	0.0	1.0	0.9	1.4	0.4
13	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	14.4	0.4	5.5	79.5	0.0	0.0	0.0	0.0	0.0	0.1	0.1
19	4.3	13.0	1.0	0.0	0.0	0.0	0.0	0.0	0.1	2.5	1.7	2.3	1.2	73.4	0.0	0.0	0.1	0.0	0.3	0.0
7	6.1	3.3	3.8	1.5	0.4	0.4	1.5	0.1	0.6	0.0	0.0	0.0	0.0	0.0	66.5	15.8	0.0	0.0	0.0	0.0
11	1.5	3.6	4.5	1.3	0.5	0.5	0.8	0.7	0.9	0.0	0.0	0.0	0.0	0.0	18.0	67.7	0.0	0.0	0.0	0.0
10	2.4	0.3	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.9	2.7	1.1	0.3	0.6	0.0	0.0	61.3	9.7	18.7	1.9
12	1.0	0.7	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.1	3.2	1.0	0.0	0.0	0.0	0.0	22.5	51.6	11.1	8.3
17	1.2	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	2.3	0.9	0.2	0.3	0.0	0.0	12.4	3.4	68.7	9.1
18	0.2	0.3	0.1	0.0	0.0	0.0	0.0	0.0	0.0	1.4	5.2	1.2	1.6	0.0	0.0	0.0	6.3	7.9	32.5	43.4

Table 4.1: Normalized confusion matrix for the interpersonal analysis with the *full ontology*. Order of the ontology cases according to *reduced ontology*.

each of which consisted of recordings from three different dyad positioning. We then computed the confusion matrix with respect to the prediction of each case, and the relative position used in the recording (Table 4.2). The recognition rates for each case are above 83%. Furthermore, the misclassified segments are evenly distributed among the relative positions, which indicates that no relative position has a strong influence on the misclassification rate. Consequently, those ontology cases are location invariant w.r.t. the features used, supporting the first hypothesis ($H1$ in Sec. 4.6.3) we investigate.

4.7.3 Ontology classification

Classification accuracy for all ontology cases were assessed by using the data of all participants, and randomly splitting them into two holds of 50% data size. In a next step, we trained a quadraric SVM for 100 times and computed the normalized confusion matrix both for the *full ontology* (20-case, Table 4.1), as well as the *reduced ontology* (7-case, Table 4.3). The confusion matrix of the *full ontology* is sorted by the respective merged cases, for which the overall recognition rate of 74% is achieved. Considering the block diagonal structure of the correct and incorrect recognition rates, the merged cases cover most of the misclassifications. This is further confirmed in the confusion matrix of the *reduced ontology*, for which the overall recognition rate increases to 88%. Furthermore, the remaining highest misclassifications mostly result from cases 2 and 3, with the rest having a recognition rate above 85%.

		Actual Case								
		1			2			3		
		F	**O**	**S**	**F**	**O**	**S**	**F**	**O**	**S**
Predicted	**1**	-	91	-	3	3	2	1	1	1
	2	3	3	4	-	83	-	2	2	3
	3	1	1	2	4	5	5	-	83	-

Table 4.2: Normalized confusion matrix for the location invariance. Predictions are further split into the tested dyad positions, frontal (F), orthogonal (O), same side (S).

		Predicted class						
	%	**1**	**2**	**3**	**4**	**5**	**7**	**10**
Correct class	**1**	89.6	6.9	1.8	0.2	1.2	0.2	0.1
	2	8.2	78.2	6.3	3.5	3.8	0.0	0.0
	3	2.6	13.0	76.7	4.3	2.3	0.3	0.8
	4	0.7	3.1	2.1	93.6	0.0	0.5	0.0
	5	0.8	1.5	0.5	0.0	95.9	0.0	1.2
	7	2.8	2.6	2.5	6.9	0.0	85.2	0.0
	10	0.9	0.0	0.0	0.0	6.7	0.0	92.4

Table 4.3: Normalized confusion matrix for the interpersonal test with the *reduced ontology*.

4.7.4 Intrapersonal classification

For the intrapersonal classification, we could use the data of six participants who performed both experiments. All of them were male of age 25 ± 2 years. The classification was conducted using the *reduced ontology* with seven cases. The results of each person vary from 89.2 to 93.5% (Table 4.4 first row), which is comparable to the results from the complete dataset. By testing the classifiers from each individual person against the rest of the data, we evaluated the generalization of individual behaviors to the overall behavioral patterns. These recognition rates range from 38.3 to 67.4% (Table 4.4 second row), which are significantly less than the overall and intrapersonal classification results. On the one hand, individual behavior generalizes up to some degree, especially in cases around 65% recognition rates, which are above chance. However, on the other hand, each individual person still has distinct behavioral patterns during interaction. This argument is further supported by tests, in which the individual data is classified with the model trained on the rest of the data (Table 4.4 third row), for which the recognition rates range from 66.0 to 85.7%. This improvement compared to the individual rates indicates that a broader range of behaviors is able to generalize quite well, and that most behaviors fall into this generalization.

Subject#	**1**	**2**	**3**	**4**	**5**	**6**
Intra-Subject	91.0	89.2	91.7	92.4	92.9	93.5
Subj vs Group	67.4	38.3	46.8	65.3	63.8	61.6
Group vs Subj	66.0	67.7	69.6	83.5	85.7	76.4

Table 4.4: Results of intrapersonal movement behavior classification tests. First row: testing intra-subject, second row: subject versus rest, third row, rest versus subject.

4.7.5 Cost comparison

In order to check whether the ontology could represent different interaction behaviors, control models for each case were learned from data. Those models are found as a combination of cost functions by the previously introduced path integral inverse reinforcement learning (PIIRL) approach. By sampling trajectories according to the covariance matrix used as in the work of Mainprice et al. [27], it could be guaranteed that the produced samples have minimal jerk. Furthermore, the samples were restricted to the range of motion shown during the experiments. For the composite cost function, the shortest distance features as well as the distances between joints were used. The resulting weighting vectors for ontology cases one to three reveal differences on the control policies (Fig. 4.5). The feature weights correspond to the pairwise distances between the humerus, forearm, and hand (features 1-9), as well as the distances between the shoulder, elbow, wrist joint and the furthest measured point of the hand (features 10-25). Although most feature weights are zero, i.e. they are negligible in the control policy, the rest distinguishes the influence of features between all three cases.

4.8 Discussion

In this section, a further analysis and discussion of our results is presented. For the ontology cases 1, 2, and 3, we have shown that those cases are invariant to the dyad positioning. Despite only a subset of the ontology cases were tested, we assume that this either holds true for the rest as well, or for some cases is infeasible to evaluate. This assumption is based on the fact that the tested cases are the least restricted of the whole ontology. There is no shared workspace (case 13), or necessary specific setup (case 10), or restrictions of the motion of one participant (case 8). Hence, we assume that a dependence on the positioning of the dyad, if it exists, would be visible in those cases with high degrees of freedom. Since such a difference was found to be small, the variance due to the positioning of the dyad w.r.t. the features used for representing them is assumed to be negligible for other cases as well.

The results from the classification analysis have provided multiple insights into the dyadic interaction behavior control. First, the used features can adequately describe and distinguish different interaction scenarios. Second, we could show that the construction of the *full ontology* is reasonable, since it recognizes each case with a rate of at least 45%, which is 40% points above chance, and in general above 60%. Furthermore, the analysis on the *full ontology* indicates that it is feasible to merge some cases due to their similarity. The resulting *reduced ontology* has also proven

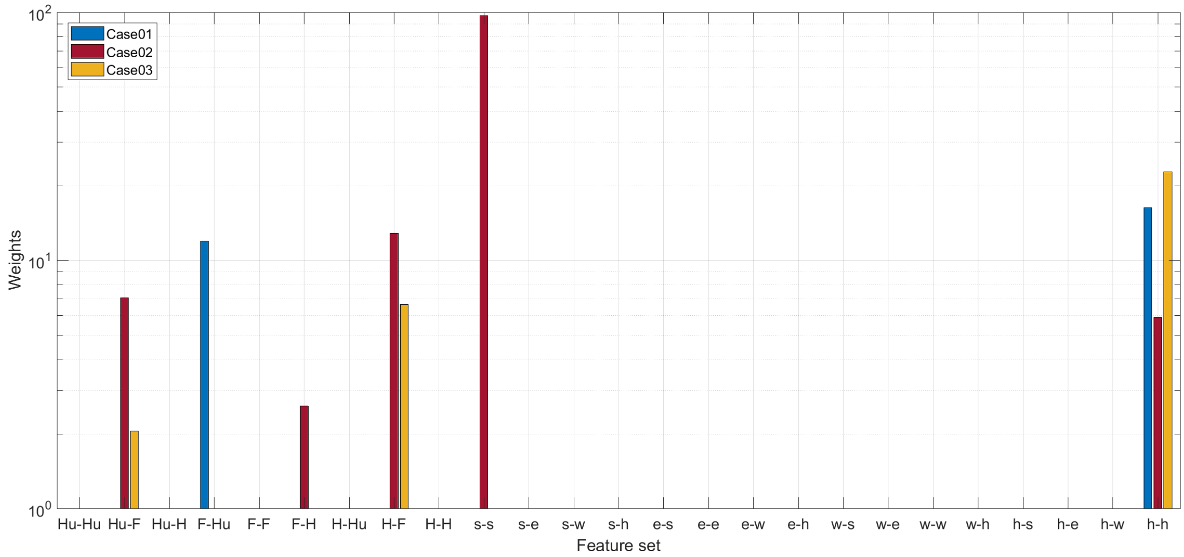


Figure 4.5: The feature weights for three different ontology cases. Feature sets are the minimal distances between segments (humerus, forearm, hand) in capital letters, and between joints (shoulder, elbow, wrist, hand) in lower case letters. Distances are in the form Person1-Person2.

to be effective, increasing the overall recognition rate up to 88%. It has to be noted that there also exist different feasible mergers, as it can be seen in the confusion matrix of the *full ontology*. Nevertheless, most of the merged cases would still be similar to the merger we proposed in this work, for which we picked the most effective combinations based on a systematic deduction and data analysis.

The results from the intrapersonal analysis also support the feasibility of the reduced ontology. The high average intrapersonal recognition rates of 91.8% confirm that there exist distinct interaction scenarios which demand different dyadic movement behaviors. Still, these interaction scenarios also depend strongly on individual behavior, as it can be seen in the group comparison results. However, this individual behavior is generalizable for each ontology case, based on the generalization results, as well as the interpersonal results.

In addition to the location invariance for the first three cases, with the second experiment, we have shown that the ontology is also invariant to small setup changes, as slight variations on the action spaces were intentionally enforced. This is further supported by the better classification results of the merged ontology. Additionally, the proposed ontology is claimed to be task invariant, and as such usable in a multitude of scenarios. This is presumed valid due to the fact that most manipulation tasks can be split into pick and place tasks [169]. The only restriction on this assumption should be the actions on objects that dynamically influence the motion of the actor. The ontology framework can also generalize to joint tasks, such as hand over tasks, or joint assembly tasks. This would be realized by using cases with a shared workspace, while enforcing the synchronicity of the dyad behavior.

Besides classifying human motion behavior, we also tried to verify if humans tend to synchronize in different setups. We could observe the tendency to synchronization during the experiments. However, we were unable to show this effect in our synchronization estimate. We attribute this to the difficulty to measure synchronization for tasks, in which the motion trajectories are intersecting, as well as for tasks, which allow a variance of end effector poses at each action space.

Lastly, the cost functions of the three presented cases distinctly encode the control policies according to the human behavior. For the first ontology case, which has the highest degree-of-freedom, only the distances between the hand points (feature 25) as well as between the forearm of agent 1 and the humerus of agent 2 (feature 4) are constrained to ensure that no collisions take place in this case. In comparison to case one, case two and three have distinctively stronger restrictions on the motion. Case two has larger and more active weights, due to the closer interaction between both hands during the grasping action, whereas case three is less restrictive due to the broader options for crossing arms. Overall, it can be seen that different ontology cases can result in different cost functions, and thus unique interaction behavior control policies. Especially, if dyadic movements are close to the action spaces of the users, the control strategy gets more restricted.

4.9 Conclusion

In this work, we presented an ontology for HHI for a shared workspace scenario. We derived an interaction model based on insights from psychology and neuroscience, and used that model to derive the ontology. This ontology was analyzed in depth, and a reduction of the ontology cases was proposed. It has been shown that the reduced ontology can be recognized with high accuracy. Moreover, we have shown that the ontology generalizes well to different behaviors, and that it can partially handle unknown behaviors. Invariance to dyad positioning for a subset of ontology cases has also been demonstrated. Lastly, unique interaction policies, that a robotic agent can use to generate interaction motions, have been learned for each case. In essence, the proposed ontology is a first step towards understanding close proximity collaboration behaviors in detail, and transferring humanlike dyadic movement control policies to robotic agents for achieving natural, safe and effective human-robot collaboration.

This study not only encourages but also provides a generic procedure for systematically exploring non-physical, close proximity interaction movement behaviors and building control policies for dyadic interaction in subsequent studies. Such a thorough coverage and analysis of HHI cases will serve as a base and reference for later studies to propose and benchmark new learning algorithms, and evaluate close proximity HRI behaviors for a wide range of scenarios.

In future work, the control policies learned can be used both to predict human motions, and also to control the robot movement during close interaction. Such an experiment would show if the ontology can directly be transferred to HRI, or whether human partners exhibit a different behavior during HRI. Furthermore, it would be useful to test the ontology with different tasks, in order to verify the proposed task invariance further.

5

Learning of Interaction Behavior Control Using RNNs: From HHI to HRI[1]

A key problem in robotics is enabling an autonomous agent to perform human-like arm movements in close proximity to another human. However, modeling the human decision and control process of movement during dyadic interaction presents a challenge. Although most prior approaches rely on multi-component robot motion planning architectures, we use data of two humans performing interfering arm reaching movements (Ch. 4) to extract and transfer interaction behavior control skill to a robotic agent. In addition to learning control policies by inverse reinforcement learning as described in the previous chapter, I proposed a novel framework to learn interaction behavior control for a robotic agent directly from data by using recurrent neural networks (RNNs) and a novel activation function. However, instead of learning a policy as a combination of predefined features, here the focus is on learning robot end effector control from human-human interaction (HHI) demonstrations without making any modeling assumptions on the dyadic movement behavior. A recurrent neural network based framework is constructed to learn a policy that computes control signals for a robot end effector in order to replace one human. The learned policy is benchmarked against unseen interaction data and a state-of-the-art learning from demonstration framework in simulated scenarios. We compare several architectures and investigate a new activation function of three stacked *tanh()*. The results show that the proposed framework successfully learns a policy to imitate human movement behavior control during dyadic interaction. The policy is transferred to a real robot and its feasibility for close-proximity human-robot interaction is shown.

5.1 Introduction

Working in close proximity with interfering arm reaching movements is one of the challenges that humans master very well. They manage to account for a large variety of different behavioral rules, e.g. collision free reaching actions, achieving complex task goals and making it easy for other humans to predict their actions. As robots become ubiquitous in our daily lives, they are expected to interact with humans in close proximity in such a natural, effective, and yet safe way.

[1] This work has been submitted to and under review in the following publication: [241]

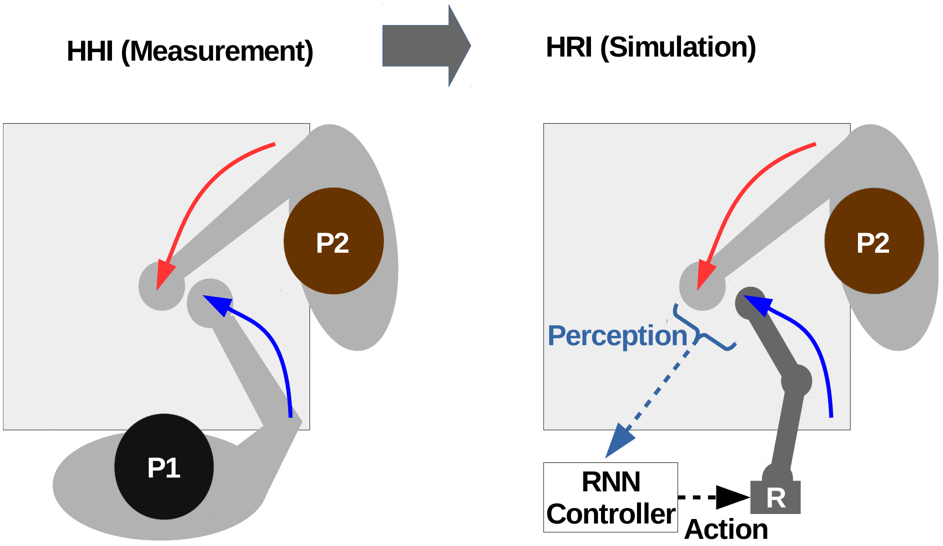


Figure 5.1: The interaction skill is transferred from HHI data to the HRI scenario. P1, P2: person 1 and 2 in the dyad, R: robotic agent, continuous line: hand movement, dashed line: inputs (*Perception*) and outputs (*Action*) to RNN controller.

One major challenge is the wide variety of relevant spatial and temporal behavioral features that the human movement exhibits during an interaction scenario. A second challenge is that there is no ground truth of how to behave in a Human-Human Interaction (HHI) or Human-Robot Interaction (HRI) scenario, as humans display inter- and intrapersonal movement variance. These require either a comprehensive framework with many components each dedicated to such subtle features, or a very capable learning structure. Consequently, this problem has been typically solved by equipping the robot with several task-specific behavioral models to achieve human-like interaction skills, including mechanisms for collision-avoidance, path re-planning, movement prediction [27], [242]. These aggregated approaches fall short of effectively generalizing to new task conditions, and taking into account movement variations of both partners.

Instead of building designated components and combining them, the interaction skill learning can be taught with learning from demonstration (LfD), as we follow in this work. However, it is still a challenge to learn a policy that both enables the robot to perform a human-like interaction behavior in close proximity to a partner, and also generalizes to unseen situations. Unlike an isolated skill training, interaction behavior learning has to incorporate dyadic interaction and task related features by using appropriate demonstrations. In essence, interaction skill enforces adaptation of agents to each other's motion behavior, while executing goal-directed movements.

This work proposes a framework to learn interaction behavior control for a robotic agent. Dyadic interaction trajectories of two humans working in a confined work space are used as demonstrations to learn a policy (Fig. 5.1). The focus is on learning robot end effector control for goal-reaching arm movements. We simultaneously use perception and actions of a reference HHI scenario in order to learn this mapping. Perception, in our case, refers to the relative positioning information of two partners' hands. Acceleration values of human hand are used as the action signals. For the training, we particularly investigate recurrent neural networks (RNNs) as they have demonstrated their capabilities on sequence learning tasks.

The aforementioned challenges are addressed by using recurrent neural networks as they are capable of learning a variety of subtle features together with remarkable generalization capabilities. As a starting point, we use a couple of RNN-cell types, Long Short-Term Memory (LSTM) [180] and Gated Recurrent Unit (GRU) [181], and then explore potential improvements specifically for HRI. The contributions of this work are:

- Development of a framework for cross-domain sequence learning of control signals from positional data.
- Imitation of human interaction behaviors with a varying level of detail depending on the network complexity.
- Thorough comparison of different RNN architectures, including a newly introduced activation function, that demonstrates the effectiveness of learning from HHI to control an autonomous agent during HRI.

While this approach is mostly validated in a simulated scenario, the learned policies were successfully applied to control a redundant robotic arm (KUKA LWR4+), and tested for close proximity human-robot interaction online.

5.2 Related Work

Here, we categorize and review different lines of research due to the multidisciplinary nature of our work.

Interaction Modeling Approaches Here, we focus on studies where human factors are considered for robot action and motion planning. One group of studies focus on specific collaboration scenarios. One approach uses interaction meshes in order to facilitate a human-robot LEGO rocket assembly task [182]. Additionally, Lee and Ryoo make use of image-based interaction information to implement an HRI placing scenario [183]. For both of these approaches the demonstrated capabilities are very scenario-oriented and it is unclear how well they can generalize in a broader sense.

A second important line of research are planning based approaches. The key idea is to re-plan robot actions and motions by adapting to the human movement behaviors [237], [238], [27], [184], however, the multiplicity of components in those architectures enforces tight coupling between functionalities and makes them harder to maintain. In addition, the generalization capability of these approaches were seldom tested extensively.

Several studies investigate motion synchronization between partners, where the robotic agent adapts to the motion of the human partner [125], [185]. Lee et al. formulates an interaction model learning for a high five-like interaction task [185] . Agent learns how to synchronize with the human by recognizing its partner's reaction and then by encoding the bidirectional movement patterns with an HMM. The HMM is also used to generate similar interactions. Instead of learning motion and interaction separately, Amor et al. formulates an interaction primitive to directly generate synchronous movement by responding to an observed human motion [125]. While the focus is on motion synchronization for both studies, pick-and-place tasks may also involve asynchronous dyadic movement patterns, which increases the complexity of modeling such interaction movements.

A recent study follows a similar path as we do by learning interaction behaviors with RNNs, and extracting HRI information from image data [186]. Although the authors manage to learn joint position controllers, unlike ours, their approach is not based on HHI but only on HRI data. Moreover, their focus is on discrete action selection and adaptation w.r.t. the action taken by the human, and thus interfering motion trajectories were not considered. Nguyen et al. also follow a similar approach using advanced LSTM architectures [187], however, again with a focus on high-level action sequences rather than control of the movement. Pellegrinelli et al. extend prior work on POMDPs, and considers not only high-level action selection but also motion control of the robotic partner [188]. The focus is on autonomous action-selection given predicted human goal to provide assistance. The interaction is assessed quantitatively and also by subjective evaluation. However, the control output and the movement patterns of the robot has not been investigated in terms of similarity to human close proximity interaction behavior.

NN-Based Skill Learning Besides approaches that explicitly model the interaction aspect, there is also a related area of robot skill learning that in general gives a good foundation on how a robot can learn to act and interact with its environment. The latter is the key differentiating factor from our work. Tanneberg et al. use spiking RNNs to teach a robot obstacle avoidance with dedicated neurons to capture the scene [189]. Unlike our approach they do not use generic HHI data but solely kinesthetic teaching that does not capture the interaction explicitly. Another recent approach tackles obstacle avoidance by using neural networks to learn coupling terms for dynamic movement primitives [190]. However, the focus is on rolling out trajectories that avoid static obstacles. In contrast, during dyadic interaction both parties affect each other's movement constantly, hence the control policy has to be able to tackle such dynamic changes in the environment.

RNN in Control of Dynamical Systems RNNs have also been subject to pure control theoretic investigation. Particularly studies from the late 1990s had a large impact and in general supported the feasibility of RNN-based controllers. One fundamental study investigated stability parameters for control systems with RNNs [191], however without direct application in complex skill control. For robotic arm controllers an early approach demonstrated that RNNs can be used to follow reference trajectories [192]. In addition, there exists general work on control and identification [193] using diagonal RNNs and using recurrent fuzzy neural networks [194]. However none of those explicitly focuses on interaction scenarios.

Learning-Agnostic HRI Methods There is also a variety of HRI methods without neural networks, which is one of the key differentiators regarding our work. We highlight some important work to show what is adjacent to our problem. Gabler et al. [237] model the interaction scenario with a game theoretical approach to choose robotic actions. However, unlike our approach the actual trajectories are based on a set of predefined Dynamic Movement Primitives and based on a predefined trajectory model. Hence, it is not clear how such an approach could be used to learn generic non-model-based interaction. Mainprice et al. developed an advanced re-planning algorithm together with inverse optimal control for close-collaboration scenarios [27]. They pursue a similar path as we do, re-planning trajectories on the fly, but it is unclear how this approach could generalize to a larger variety of new scenarios. Lasota and Shah focus on *human-aware motion planning* which also combines perception of the human with adaptive planning [184]. Their approach explicitly predicts the interacting human's trajectory and thus represents an approach that composes the HRI functionality from different modules, which is not necessary in our model as we immanently capture all relevant relationships with the neural network. Besides these behavior-oriented work, there exists classical control theory in HRI. De Luca and Flacco [195] developed a dedicated control framework including collision avoidance suitable to a collaboration scenario. Haddadin et al. investigated how dangerous injuries can be prohibited when a robotic arm handles tools in the proximity of a human [196]. Both approaches focus on the challenge of minimizing collisions and their harm respectively, not basing control on HHI trajectory data in an end-to-end framework. This mainly distinguishes their work from ours, as we implicitly assume that all safety relevant features can be captured by the neural network.

5.3 Methods

In this section, we outline our methodology to construct a framework for learning robot end effector control from human-human interaction demonstrations. Two Recurrent Neural Network (RNN) architectures along with a newly proposed activation function are described, followed by the formulation of sequence learning by RNNs. Next, we introduce the dataset used and the features extracted. Lastly, we explain the training and testing procedures we followed.

5.3.1 Recurrent neural networks

RNNs are function approximators and suitable to sequence learning. Like feedforward networks, RNNs are typically organized in layers. Input signals are propagated through the network starting with the network inputs to the first layer. In this study, we focus on two well-established RNN structures, the Long Short-Term Memory (LSTM) [180] as well as the Gated Recurrent Unit (GRU) [181]. Let $\mathbf{x}_t$ be inputs and $\mathbf{y}_t$ be outputs of a layer j at time step t. Furthermore, let $\odot$ denote the element-wise multiplication for two vectors, and σ as well as ϕ denote different nonlinear activation functions. Then, we obtain the following equations for the LSTM network

$$\mathbf{i}_t^j = \sigma\left(\mathbf{W}_{i,x}^j \mathbf{x}_t^j + \mathbf{W}_{i,y}^j \mathbf{y}_{t-1}^j + \mathbf{b}_i^j\right) \tag{5.1}$$

$$\mathbf{f}_t^j = \sigma\left(\mathbf{W}_{f,x}^j \mathbf{x}_t^j + \mathbf{W}_{f,y}^j \mathbf{y}_{t-1}^j + \mathbf{b}_f^j\right) \tag{5.2}$$

$$\mathbf{o}_t^j = \sigma\left(\mathbf{W}_{o,x}^j \mathbf{x}_t^j + \mathbf{W}_{o,y}^j \mathbf{y}_{t-1}^j + \mathbf{b}_o^j\right) \tag{5.3}$$

$$\mathbf{g}_t^j = \phi\left(\mathbf{W}_{g,x}^j \mathbf{x}_t^j + \mathbf{W}_{g,y}^j \mathbf{y}_{t-1}^j + \mathbf{b}_g^j\right) \tag{5.4}$$

$$\mathbf{c}_t^j = \mathbf{f}_t^j \odot \mathbf{c}_{t-1}^j + \mathbf{i}_t^j \odot \mathbf{g}_t^j \tag{5.5}$$

$$\mathbf{y}_t^j = \mathbf{o}_t^j \odot \phi\left(\mathbf{c}_t^j\right) \tag{5.6}$$

For the GRU network, we obtain similar equations. However, here the concept of memorizing and forgetting information is captured with less computational steps and parameters.

$$\mathbf{r}_t^j = \sigma\left(\mathbf{W}_{r,x}^j\mathbf{x}_t^j + \mathbf{W}_{r,y}^j\mathbf{y}_{t-1}^j + \mathbf{b}_r^j\right) \tag{5.7}$$

$$\mathbf{z}_t^j = \sigma\left(\mathbf{W}_{z,x}^j\mathbf{x}_t^j + \mathbf{W}_{z,y}^j\mathbf{y}_{t-1}^j + \mathbf{b}_z^j\right) \tag{5.8}$$

$$\tilde{\mathbf{y}}_t^j = \phi\left(\mathbf{W}_{\tilde{y},x}^j\mathbf{x}_t^j + \mathbf{W}_{\tilde{y},y}^j\left(\mathbf{y}_{t-1}^j \odot \mathbf{r}_t^j\right) + \mathbf{b}_{\tilde{y}}^j\right) \tag{5.9}$$

$$\mathbf{y}_t^j = \mathbf{z}_t^j \odot \mathbf{y}_{t-1}^j + \left(\mathbf{1} - \mathbf{z}_t^j\right) \odot \tilde{\mathbf{y}}_t^j \tag{5.10}$$

Each multilayer network of LSTM or GRU units is followed by a linear output layer $\mathbf{y}_{\text{out}} = \mathbf{W}_{\text{out}}\mathbf{x}_{\text{out}} + \mathbf{b}_{\text{out}}$. Those widely-used cell types use the concept of memorizing information in several stages: first, by reusing outputs from previous time steps $\mathbf{y}_{t-1}$, second, by computing $\mathbf{c}_t$ from $\mathbf{c}_{t-1}$ in LSTM and $\mathbf{y}_t$ from $\mathbf{y}_{t-1}$ in GRU architectures. From now on, the variable θ of an LSTM-based or GRU-based network summarizes the matrices $\mathbf{W}^j, \mathbf{W}_{\text{out}}$ and vectors $\mathbf{b}^j, \mathbf{b}_{\text{out}}$ for all layers and activation functions including the linear output layer. Next, we introduce a novel activation function to improve the network performance.

5.3.2 Activation functions

For LSTM and GRU networks, in general, $\sigma = \text{sigmoid}$ and $\phi = \tanh$ have been used as the activation functions. While we use and analyze them in this work, we introduce another function that is dedicated to the properties of HHI. In particular, we replace ϕ for the GRU cell with a composition of several scaled $\tanh()$ functions:

$$\phi\left(x\right) = \frac{1}{3}\sum_{p=1}^{3}\tanh\left(s\left(x - x_p\right)\right) \tag{5.11}$$

where x_1, x_2, x_3 are constant offsets and s is a scaling factor. We refer to it as triple-tanh. While the composition of several activation functions is a well known idea, e.g. with a piecewise linear function [197], we choose the combination parameters to suit the interaction learning process. The parameters are chosen as $x_1 = -2, x_2 = 0, x_3 = 2, s = 4$, with inflection points: $(-1.00, -0.33)$ and $(1.00, 0.33)$ (Fig. 5.2). This results in similar saturation behavior as for the tanh()-function at the borders. Moreover, the inflection points support convergence of the GRU's $\tilde{\mathbf{y}}$, i.e. how much of the previous state needs to be carried over, as a distinct range of input values is mapped to an almost fixed output value through the activation function. Because $\mathbf{y}_t$ is a convex combination of $\mathbf{y}_{t-1}$ and $\tilde{\mathbf{y}}_\mathbf{t}$ (eq. 5.10), a convergence of $\mathbf{y}_t$ and thus $\mathbf{y}_{t-1}$ will typically coincide with a convergence of $\tilde{\mathbf{y}}_\mathbf{t}$. This activation design choice can be used by the network to learn a more complex state convergence. Moreover, as the activation function is applied element wise on multiple coordinates, a state saturation of one coordinate may trigger a different dynamic behavior of another state that can be beneficial to generate outputs for a natural interaction behavior. In essence, for close proximity interaction trajectories, we assume three phases, where the dyadic movements approach, are in close proximity to, and move away from each other. This motivates defining a multistage activation function to be able to differentiate a range of input values and to map them to corresponding output control signals distinctively. The inputs capture relative positioning information and thus the triple-tanh function indirectly captures a temporal together with a spatial behavior. Note that, for the LSTM it is not intuitive to draw similar connections as the architecture is different than

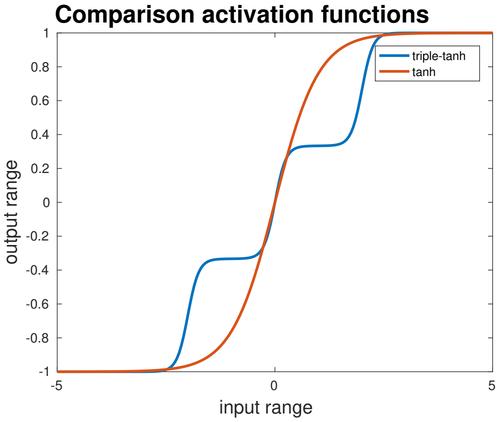


Figure 5.2: The classical *tanh()* and the introduced triple-*tanh()* activation function for $x_1 = -2, x_2 = 0, x_3 = 2, s = 4$

GRUs. Thus, we do not use the triple-tanh activation function for the LSTM cells in our analysis.

5.3.3 Learning sequences with RNNs

In general, the motivation for sequence learning is to predict future outputs within the same domain as the input space. However, in this work, supervised learning with RNNs is used to learn sequence of control signals from spatial interaction data. Therefore, spatial input sequences, $\mathbf{X}_k = [\mathbf{x}_{0,k}, \mathbf{x}_{1,k}, ...\mathbf{x}_{n_k-1,k}]$, and acceleration output sequences, $\mathbf{Y}_k = [\mathbf{y}_{0,k}, \mathbf{y}_{1,k}, ...\mathbf{y}_{n_k-1,k}]$, of length n_k are presented to the neural network, where k is the batch index. Then, a mapping is learned to estimate the necessary acceleration signal, $\hat{\mathbf{Y}}_k$, given a spatial input feature sequence. The resulting prediction error can be measured with a cost or distance metric. This error typically accounts for multiple sequences either as part of a batch or regarding the full data set. In our case, we use a mean squared error that is normalized w.r.t. each single sequence pair's length. In order to optimize the prediction, we have the following minimization problem

$$\min_{\theta} \mathbf{D} = \frac{\sum_{k\in K} \|\mathbf{Y}_k - \hat{\mathbf{Y}}_k\|_2^2}{\sum_{k\in K} n_k} \tag{5.12}$$

where K is the index set of sequences under investigation in one batch, and with all variable network parameters summarized as θ. The optimization is approximately solved with numerical methods, for which we use the BFGS algorithm [198], and the backpropagation algorithm is used for updating the weights [199]. It is important to highlight that training is performed on full sequences that were already recorded. Recall, in this work we refer to this learning type as *off-line learning*. Having introduced the computational components of our implementation, we can now focus on the interaction scenario itself.

5.3.4 Data

We use human-human interaction motion-capture data of close-proximity reaching movements that was recorded for the ontology construction presented in Chapter 4. The same dataset from the

Name	Num. train.	Max. len.	Num. test
D1	31 212	169	23
D2	12 664	124	38
D3-1/2/3	15 312	109	200

Table 5.1: Data sets. *Num. train.*: number of training samples, *Max. len.*: Number of time steps for the longest sequence in the data set, *Num. test.*: number of test samples. The sets D3-1, D3-2, D3-3 have the same size but are created through random shuffling of superset they are taken from.

human-human interaction (HHI) experiment conducted (Chapter 4.6) is used for this work. The hand positions $s_{\mathrm{H,P1}}$ and $s_{\mathrm{H,P2}}$ of the persons P1 and P2, respectively, are extracted. From now on, we assume that the robotic agent is supposed to learn the movement of P1 as part of the interaction scenarios. In order to eliminate a directional bias, we augment the original data by rotating it with 90, 180 and 270 degrees [200].

After these pre-processing steps, the input and output sequences are computed. Two features are used to capture the interaction movement behavior of the hand movements. First, $\mathfrak{f}_{\mathrm{H2G},t}$ stores the distance between P1's current and goal hand position, $\mathbf{s}_{\mathrm{H,P1},t}$ and $\mathbf{s}_{g,\mathrm{P1},t}$, respectively (Eq. 5.13). Second, $\mathfrak{f}_{\mathrm{H2H},t}$ describes the relation between two persons' hands, $s_{\mathrm{H,P1}}$ and $s_{\mathrm{H,P2}}$, and is computed by a slightly modified formula of the potential fields approach (Eq. 5.14).

$$\mathfrak{f}_{\mathrm{H2G},t} = \mathbf{s}_{g,\mathrm{P1},t} - \mathbf{s}_{\mathrm{H,P1},t} \tag{5.13}$$

$$\mathfrak{f}_{\mathrm{H2H},t} = \begin{cases} S(d^*, d)\left(\mathbf{s}_{\mathrm{H,P2},t} - \mathbf{s}_{\mathrm{H,P1},t}\right) & \text{, if } d_t > d^* \\ \mathbf{0} & \text{, otherwise} \end{cases} \tag{5.14}$$

where $S(d^*, d) = \left(\frac{1}{d^*} - \frac{1}{d_t}\right)\frac{1}{d_t}$, $d_t = \|\mathbf{s}_{\mathrm{H,P2},t} - \mathbf{s}_{\mathrm{H,P1},t}\|$ and $d^* = 400\mathrm{mm}$ is a heuristically chosen constant. These metrics are then used to define the network input sequences. The output sequences, i.e. the control signals to be learned, are the P1's hand acceleration values and hence, we compute

$$\mathbf{a}_{H,\mathrm{P1}} = \nabla^2 \mathbf{s}_{\mathrm{H,P1}} \tag{5.15}$$

with $(\nabla\zeta)_t = \frac{\zeta_{t+1} - \zeta_{t-1}}{2}$ for intermediate, $(\nabla\zeta)_0 = \zeta_1 - \zeta_0$ and $(\nabla\zeta)_{n_k-1} = \zeta_{n_k-1} - \zeta_{n_k-2}$ for initial and terminal time steps. Then, the input and output sequences are formed as

$$\mathbf{X}_k = \begin{bmatrix} \mathfrak{f}^k_{\mathrm{H2G},0} & \mathfrak{f}^k_{\mathrm{H2G},1} & \cdots & \mathfrak{f}^k_{\mathrm{H2G},n_k-1} \\ \mathfrak{f}^k_{\mathrm{H2H},0} & \mathfrak{f}^k_{\mathrm{H2H},1} & \cdots & \mathfrak{f}^k_{\mathrm{H2H},n_k-1} \end{bmatrix} \in \mathbb{R}^{6\times n_k} \tag{5.16}$$

$$\mathbf{Y}_k = \begin{bmatrix} \mathbf{a}_{H,\mathrm{P1},0} & \mathbf{a}_{H,\mathrm{P1},1} & \dots & \mathbf{a}_{H,\mathrm{P1},n_k-1} \end{bmatrix} \in \mathbb{R}^{3\times n_k} \tag{5.17}$$

For further computations, the raw data, $\mathbf{X}_k$ and $\mathbf{Y}_k$, that is presented to the neural networks is normalized to $[-1, 1]$.

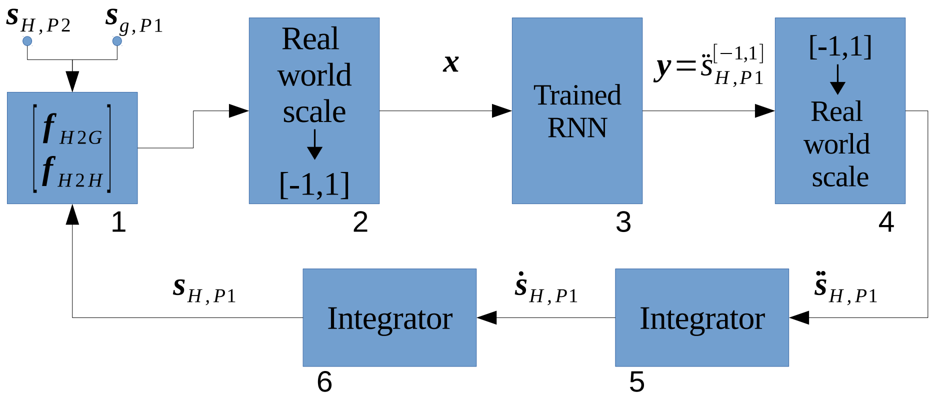

$s_{H,P2}$
$s_{g,P1}$
$\begin{bmatrix} f_{H2G} \\ f_{H2H} \end{bmatrix}$
1
Real world scale ↓ [-1,1]
2
x
Trained RNN
3
$y = \ddot{s}^{[-1,1]}_{H,P1}$
[-1,1] ↓ Real world scale
4
$\ddot{s}_{H,P1}$
Integrator
5
$\dot{s}_{H,P1}$
Integrator
6
$s_{H,P1}$

Figure 5.3: On-line simulation signal workflow: 1) $\mathrm{s}_{g,\mathrm{P1}}$, $\mathrm{s}_{\mathrm{H,P2}}$, $\mathrm{s}_{\mathrm{H,P1}}$ are captured in $\mathfrak{f}_{H2H}$, $\mathfrak{f}_{H2G}$, 2) The results are mapped from mm and $1/mm$ to $[-1,1]$, 3) Trained network computes outputs, 4) Features are mapped from $[-1,1]$ to accelerations in mm/s^2, 5/6) double numerical integration to compute simulated hand position $\mathrm{s}_{\mathrm{H,P1}}$.

Based on this formulation, we extract several sub data sets from the recordings. The different data sets are chosen in order to validate the influence of different design metrics (Table 5.1). As the recordings comprise different scenarios due to variations in positioning, trajectories belonging to similar setups are labeled accordingly. Hence, we are able to test performance on *similar* as well as *different* interaction scenarios. For the training sets D1 and D2, a small set of recordings are left out to test the learning performance on setups that are different from the ones used during training. For the training sets D3-n, we use a subset of scenarios consisting of three different interaction setups, and for each, 5% of the data is left out for testing.

5.3.5 Training

We train several architectures (Table 5.2) in order to identify relevant parameters that impact the learning performance for interaction data. The open source framework *TensorFlow*™ is used. For LSTM, we use a conventional implementation with a forget bias of 1 that is added to $\mathrm{b}^{\mathrm{j}}_{\mathrm{f}}$ (Eq. 5.2) as in [201]. The GRU networks are directly based on the original study [181]. The network states are reset to 0 before processing each sequence. The variables are initialized using an initializer, that draws uniformly distributed samples from $\left[-\frac{\sqrt{6}}{\sqrt{n_{in}+n_{out}}}, \frac{\sqrt{6}}{\sqrt{n_{in}+n_{out}}}\right]$ with the respective matrix dimensions n_{in} and n_{out} [202]. The exceptions are the RNN cell biases which are initialized to 0, and the biases in GRU that contribute to $\mathbf{r}$ and $\mathbf{z}$ (Eq. 5.7, 5.8), which are initialized to 1.

5.3.6 Simulation

While we trained on full input and output sequences, our aim is to learn a policy from this *off-line* process to control *on-line* behavior. During testing, the network is only presented with the initial inputs and then at each time instance computes a new output control signal which again influences the simulated robot end effector position. As we predict acceleration signals, we need to integrate the re-scaled output twice, which is done with the trapezoid rule (Fig. 5.3). The simulation runs for 200 time steps, each 32ms. We also investigate how robust the learned behavior is in the presence of uncertainty. Hence, we also test on a case where a normally distributed noise $\mathcal{N}\left(\mu = 0\mathrm{mm}, \sigma^2 = 9\mathrm{mm}^2\right)$ is added as independent noise sources to P2's hand position $\mathrm{s}_{\mathrm{H,P2}}$, as well as to P1's target hand location $\mathrm{s}_{g,\mathrm{P1}}$. Note that, this corresponds to equally adding noise to $\mathrm{s}_{\mathrm{H,P1}}$ as we constructed our features, $\mathfrak{f}_{\mathrm{H2G}}$, $\mathfrak{f}_{\mathrm{H2H}}$, to contain these variables as differences (Eq. 5.13, 5.14).

5.4 Results

In order to investigate how well the simulated robotic hand compares to the observed pattern, we conduct a variety of tests. One major challenge is to measure the performance because there is not any standard similarity metric for gauging dyadic interaction behavior quality. Thus, we introduce several metrics to evaluate the imitated interaction behavior quantitatively. Recall that we always simulate the interaction scenario with the learned policy, and compare the resulting behavior with the trajectories of the person's hand that the robot is supposed to mimic in the interaction. In that regard, we maintain two different view points to analyze the results: first, demonstrating the success of our approach by showing that for a wider variety of architectures we successfully learn the interaction pattern; second, comparing different *architecture clusters*, i.e. combinations of comparable units per layer and data sets in order to examine the effectiveness of different cell types. Each cluster comprises 3 configurations, i.e. C1-3, 4-6, . . ., 22-24. Here, we investigate the cell-type performance in a winner-take-all manner, highlighting the order of performance under the corresponding test metric (Table 5.2). For C16-24, we use averaged values of each performance criterion, using the mean over the models that were trained based on three differently shuffled data sets. In terms of reaching criterion, we focus on a region of 30mm around the goal hand position and consider the target reached, if the simulated, or the recorded, hand permanently stays within this region for the remaining of the simulation after a time step t_0.

5.4.1 Human-likeness

While there is a variety of possible metrics to measure human-likeness, we define it as spatial and temporal hand trajectory similarity between the recorded and the simulated data, in the segment until t_0. The spatial similarity is measured using Probabilistic Movement Primitives (ProMP) [127] which is a model to capture the statistical properties of sequential movement data. Note that, we use both the forward and the backward movements as separate interaction cases. In that regard, we fit a ProMP model to each of the forward and backward trajectory test subsets with an interpolated fixed number of time steps. These distributions are then compared by KL divergence against the simulated end effector trajectories as part of the interaction scenario. The results are benchmarked based on architecture details (Table 5.2).

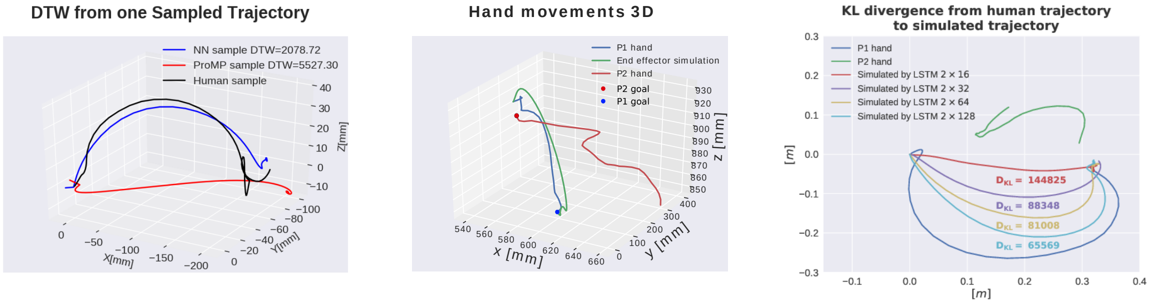
Overall, we see that different architectures can create similar results. First, the comparison in *architecture clusters* slightly favors LSTM-based networks for test cases which are spatially *different* than the data used to train the networks, i.e. C1-15. Second, for the *similar* train/test case *architecture clusters* LSTM based networks perform slightly better as well, i.e. C16-24. In general, KL divergence metric is not intuitive to interpret and we observe similar values related to a dataset. Additionally, we empirically show that, if we vary the network sizes in a broader range, the simi-

C.	**Net**	**Units**	**Data**	**KL div. bw**	**KL div. fw**	**Reach Time MSE**	**Terminal Dist.**	**Num. reached**
				(no n. / noise)	(no n. / noise)	(no n. / noise)	(no n. / noise)	(no n. / noise)
				ProMP-based	ProMP-based	[s^2]	[mm]	[%]
1	GRU3t	2 x 5	D1	6e+03 / 9.8e+03	4.5e+03 / 7.4e+03	29 / 29	9 / 17	4 / 4
2	GRU	2 x 5	D1	3.8e+02 / 3.6e+02	4.8e+02 / 6.6e+02	0.26 / 1.5	13 / 13	100 / 100
3	LSTM	2 x 5	D1	3.6e+02 / 4e+02	4.5e+02 / 5.5e+02	0.091 / 0.089	6.6 / 7.3	100 / 100
4	GRU3t	2 x 8	D1	4.1e+02 / 4.3e+02	5.4e+02 / 5.7e+02	0.1 / 0.099	5 / 6.5	100 / 100
5	GRU	2 x 8	D1	3.5e+02 / 3.8e+02	1e+03 / 1.7e+03	2.7 / 2.8	8.7 / 11	91 / 91
6	LSTM	2 x 8	D1	4.1e+02 / 4.1e+02	3.9e+02 / 5.1e+02	1.6 / 1.6	9.7 / 10	96 / 96
7	GRU3t	2 x 9	D1	3.8e+02 / 3.8e+02	6.5e+02 / 6.7e+02	0.075 / 0.075	6.9 / 7.3	100 / 100
8	GRU	2 x 9	D1	3.1e+02 / 3.3e+02	6.8e+02 / 1.5e+03	3.2 / 6.9	15 / 16	91 / 87
9	LSTM	2 x 9	D1	3.7e+02 / 4.2e+02	6.2e+02 / 5.8e+02	0.08 / 0.079	7.4 / 8.1	100 / 100
10	GRU3t	2 x 7	D2	1.6e+03 / 1.3e+04	1.2e+03 / 2.6e+04	0.29 / 27	7.9 / 18	100 / 32
11	GRU	2 x 7	D2	1.7e+03 / 8.7e+02	1.2e+03 / 1.5e+03	0.15 / 14	5.4 / 16	100 / 95
12	LSTM	2 x 7	D2	1.4e+03 / 1.3e+03	1.2e+03 / 1.1e+03	0.28 / 0.22	7.3 / 6.5	100 / 100
13	GRU3t	2 x 9	D2	1.7e+03 / 1.3e+04	1.2e+03 / 1.3e+04	0.14 / 27	4.1 / 21	100 / 50
14	GRU	2 x 9	D2	1.7e+03 / 3.7e+03	1.2e+03 / 3.9e+03	0.17 / 18	3 / 13	100 / 89
15	LSTM	2 x 9	D2	1.7e+03 / 1.1e+03	1.1e+03 / 9.2e+02	0.15 / 0.83	3.5 / 6.6	100 / 97
16	GRU3t	2 x 9	D3(1)	1.9e+02 / 1.7e+02	1.3e+02 / 1.3e+02	0.055 / 0.056	2.3 / 4.4	100 / 100
17	GRU	2 x 9	D3(1)	2.3e+02 / 2e+02	1.4e+02 / 1.3e+02	0.058 / 0.06	1.6 / 4.7	100 / 100
18	LSTM	2 x 9	D3(1)	2.2e+02 / 1.9e+02	1.3e+02 / 1.2e+02	0.057 / 0.058	1.6 / 4.5	100 / 100
19	GRU3t	2 x 9	D3(2)	1.3e+02 / 1.2e+02	3.6e+02 / 3.1e+02	0.049 / 0.049	2.5 / 4.3	100 / 100
20	GRU	2 x 9	D3(2)	1.5e+02 / 1.2e+02	3.2e+02 / 2.8e+02	0.053 / 0.053	1.9 / 4.9	100 / 100
21	LSTM	2 x 9	D3(2)	1.2e+02 / 1.2e+02	3.4e+02 / 2.9e+02	0.049 / 0.05	1.8 / 4.7	100 / 100
22	GRU3t	2 x 9	D3(3)	7.8e+02 / 7.3e+02	1e+03 / 8.8e+02	0.13 / 0.13	3 / 4.7	100 / 100
23	GRU	2 x 9	D3(3)	7.7e+02 / 6.6e+02	1e+03 / 8.7e+02	0.13 / 0.14	2 / 4.8	100 / 100
24	LSTM	2 x 9	D3(3)	7.7e+02 / 7e+02	1e+03 / 8.1e+02	0.3 / 0.19	1.6 / 4.4	99 / 100

Table 5.2: Compared network architectures, C.: configuration number, Net: cell types (GRU3t is GRU with triple-tanh activation function), Units: number of units in the layers, Data: data sets (Table 5.1) where a number in parentheses indicates a cluster of spatially *similar test cases*.

larity to human-like trajectories become more significant, i.e. larger networks tend to have smaller KL divergence to human trajectories (Fig. 5.4c).

The temporal similarity is measured based on the Mean Squared Error (MSE) of reaching time t_0, again comparing simulated and observed trajectories (Table 5.2). We can observe two phenomena: first, reasonable reaching time deviations below $0.1\mathrm{s}^2$, and second, some scenarios that do not perform well with large deviations up to $29\mathrm{s}^2$. This is, in general, related to trajectories that do not reach the target with these architectures. There is no clear distinction between the MSE performance values within *architecture clusters*. However, for *different* test cases, i.e. C1-15, the triple-tanh activation function slightly improves the performance of GRU based architectures, whereas LSTM networks perform the best in the respective *architecture clusters*. For the *similar* test cases, i.e. C16-24, the GRU with triple-tanh activation outperforms the other architectures,



(a) (b) (c)

Figure 5.4: (a) Comparison of generated trajectory from ProMP and our approach, and their closest human trajectory in terms of DTW metric.(b) Comparison of hand positions. Recorded interaction scenario: P1's (blue), and P2's (red) hand positions, and RNN-controlled end effector position (green). (c) Benchmark of the end effector trajectories generated by larger LSTM architectures on a recorded representative 2D hand trajectory.

achieving an averaged MSE below $0.05\mathrm{s}^2$.

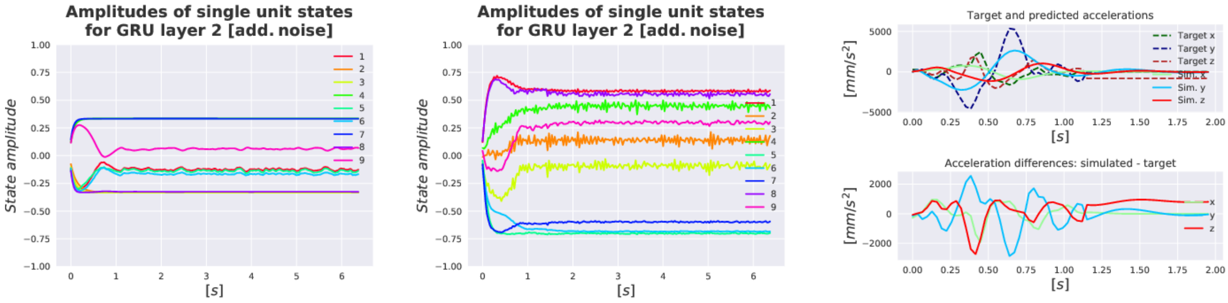
5.4.2 Convergence of hand position

Next, we compare the convergence of the simulated end effector position to a given target location. This analysis is concerned only with gaining a comprehensive insight on the imitation capability of our proposed framework from HHI to HRI, but not with traditional controller stability. We intend to establish a metric, that addresses similar concerns as the latter one. Hence, we measure the number of cases in which the target location is permanently reached as part of the simulated interaction scenario based on the $30\mathrm{mm}$ reaching criterion as in the previous subsection (Table 5.2). For the *different* test cases, i.e. C1-15, LSTM based architectures are the most capable regarding this performance criterion, each benchmarked to similar architectures. However, for *similar* test cases. i.e. C16-24, almost all architectures achieve a 100% rate of reaching the target. Regarding the overall feasibility we need to note, that the majority of tests has a goal reaching ratio above 90%.

The second criterion in the convergence analysis is the average terminal hand-to-goal distance for the successfully reached cases. Here, we observe good performance ranging from a few mm to a few cm of terminal distance.

5.4.3 Qualitative analysis

Hand-related trajectories Finally, we investigate how well the shapes of the simulated hand trajectories are suited compared to the observed scenario. For illustrative purposes we choose a particular case together with a successful architecture. We compare hand acceleration and position with the ground truth. There is a notable difference in each of the simulated accelerations by coordinate. However, the overall tendency on when to accelerate is captured by the network (Fig. 5.5c). It is important to note, that this interaction behavior emerges although the input data is different between ground truth and test data as the network outputs influence the network inputs



(a) (b) (c)

Figure 5.5: States of the second GRU layer with noise perturbation for a) C7 (GRU+triple-tanh) and b) C8 (GRU+tanh). c) Comparison of original and simulated accelerations.

during testing. Next, we focus on the hand movement. The simulated hand position follows a similar shape as in the observed data as part of the interaction scenario. Although it slightly overshoots, it still reaches the target sufficiently under the defined conditions (Fig. 5.4b). Lastly, for both acceleration and position, we note that the generated trajectories are smooth in general.

Influence of noise on layer states In our comparison it becomes clear that noise has a negative impact on the measured performance criteria. But we still observe that several architectures successfully perform the task even under noise. One factor that is of interest to our comparison is the impact of noise on the layer states for the activation function we introduced compared to vanilla GRU architectures. Recall, that the triple-tanh activation function leads to saturations for values around its inflection points. For the layer states we observe, that for an exemplar architecture of seventh and eigth configuration, the GRU cell states with triple-tanh activation function are less sensitive to the noise perturbation which is beneficial (Fig. 5.5a,b and Table 5.2).

Trade-off between expressiveness and complexity So far, we investigated the feasibility of our approach on a variety of small-sized architectures with BFGS as a capable optimization algorithm, corresponding to applications with limited computing and data resources. Now we look into how much the performance can be increased by using a larger LSTM network structure on a training set of 32 632 samples with rotation. The BFGS algorithm is computationally too expensive for such larger networks, thus we use the ADAM optimizer. This learning framework is capable to imitate more fine-grained details with increasing architecture and data complexity (Fig. 5.4c). Hence, the more complex the architecture is, the closer the generated end effector movement behavior resembles the original observation. This illustrates the capability of the proposed learning framework to imitate human interaction behaviors on multiple levels of detail depending on the network architecture and data complexity. For the analysis presented in Sec. 5.4.4 & 5.4.6, we use these LSTM policies.

5.4.4 ProMP comparison

We already used ProMPs in order to establish a KL-divergence similarity metric. Now, we compare our approach with ProMPs with respect to the trajectories generated for interaction cases. In this

	ProMP from test set		ProMP trained separately	
	mean	sample	mean	sample
ProMP forward	**118**	5238 ± 2203	2195	4198 ± 1939
Ours forward	1101	$\mathbf{1833 \pm 782}$	**1302**	$\mathbf{2028 \pm 832}$
ProMP backward	**152**	5269 ± 2958	2010	5998 ± 2728
Ours backward	878	$\mathbf{2244 \pm 770}$	**858**	$\mathbf{2266 \pm 800}$

Table 5.3: The dynamic time warping (DTW) values between trajectories generated by different approaches and the human trajectory, for forward and backward reaching separately. For *sample* DTW, we record the mean value and standard deviation.

experiment, both ProMP and our method are tested on the same dataset, and compared using the dynamic time warping (DTW) metric for quantifying the similarity between two trajectories.

We consider two evaluation measurements, namely *mean DTW* and *sample DTW*. For *mean DTW*, we calculate the mean trajectories of generated samples from the RNN-based policy and the ProMP model, and compare them with the mean human trajectory. However, human trajectories often exhibit more complex patterns that are not well described by a single mean, e.g., some trajectories may also lie afar from the mean. These trajectories can also be equally interesting to generate. Therefore, we did a thorough evaluation, by looking for the closest sample in the human trajectory set, and computed the metric as the *sample DTW*.

In the first experiment, we compare our approach with a ProMP model that is directly learned from the test set. Unlike ProMP, our RNN is trained on a different training set, and never exposed to these test samples before. In our second experiment, we divide the test set into two parts. The first part is only for training the ProMP model, which contains 15 samples (model parameters converged). While the second part has 21 samples for testing both ProMP and our approach.

ProMP performs well in mimicking the mean human trajectory, when it is trained on the test set directly (Table 5.3). Nevertheless, it fails to capture the complex patterns of human trajectory, as suggested by an inferior *sample DTW* performance, for which our approach performs better. Example outputs of both approaches and the closest human trajectory can be seen in Figure 5.4a.

5.4.5 Correlation Analysis

Till now, the original trajectories have been compared to the simulated ones. Now, we would like to also compare the network state trajectories to some of the quantities that are computed directly or as a side product, namely: wrist position, wrist velocity, wrist acceleration and input features. We analyze the relationship and dependency between the network states and the quantities extracted from the recorded data. A normalized cross correlation between single states and the quantities under investigation is computed. At this point, we narrow down our results to a representative case with good KL-divergence for new cases without noise, i.e. a configuration with 2 GRU layers, 9 units each and our triple-*tanh* activation function.

In order to measure similarities we use the following common normalized cross correlation between one of the metrics and one of the states

$$\frac{\langle state, metric\rangle}{\|state\|_2\|metric\|_2} \tag{5.18}$$

with $\langle\cdot,\cdot\rangle$ the inner product, $\|\cdot\|_2$ the 2-norm where $state$ corresponds to a one-dimensional state time series of a single unit's network state. The $metric$ corresponds to a time series representing a single coordinate from either input features, wrist position, wrist velocity or wrist acceleration, e.g. time-series for x-coordinate of wrist acceleration. The normalization accounts for different amplitudes and maps the results into the interval of $[-1, 1]$. In order to capture the results for multiple training trajectories, we stack the values for each sample horizontally per unit of a state, sequences are truncated to the time of goal-reaching convergence (Fig. 5.6).

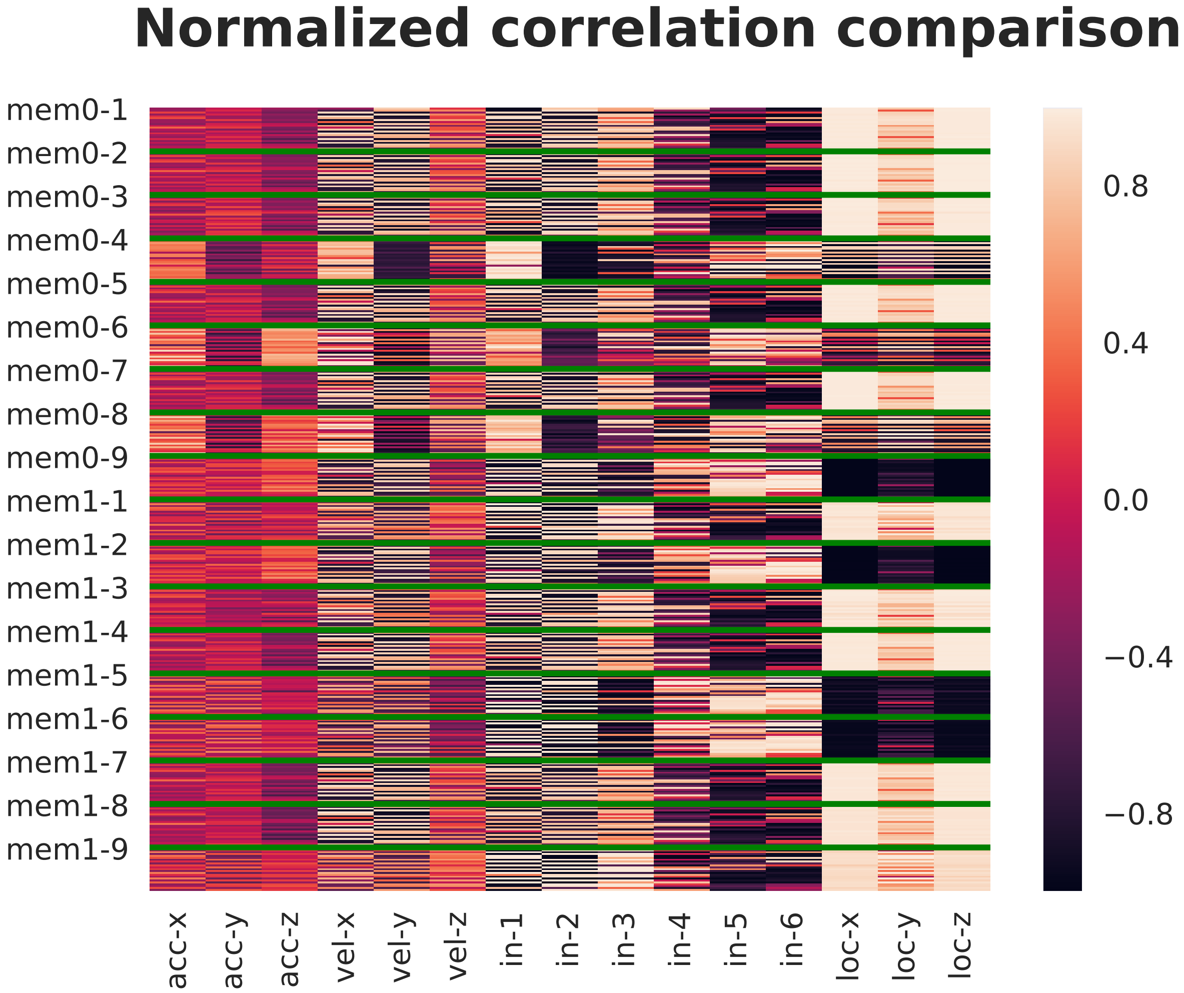


Figure 5.6: Normalized cross correlation of states and metrics (acc: acceleration, vel: velocity, loc: location all in x,y,z direction, in: input features in its six dimension, i.e. in-1,2,3: wrist-to-goal metric, in-4,5,6: wrist-to-wrist metric, c.f. previous chapter). For each state we stack the values of all the samples in the test set horizontally, which gives either a uniform color for similar behavior or stripes for varying behavior for different samples.

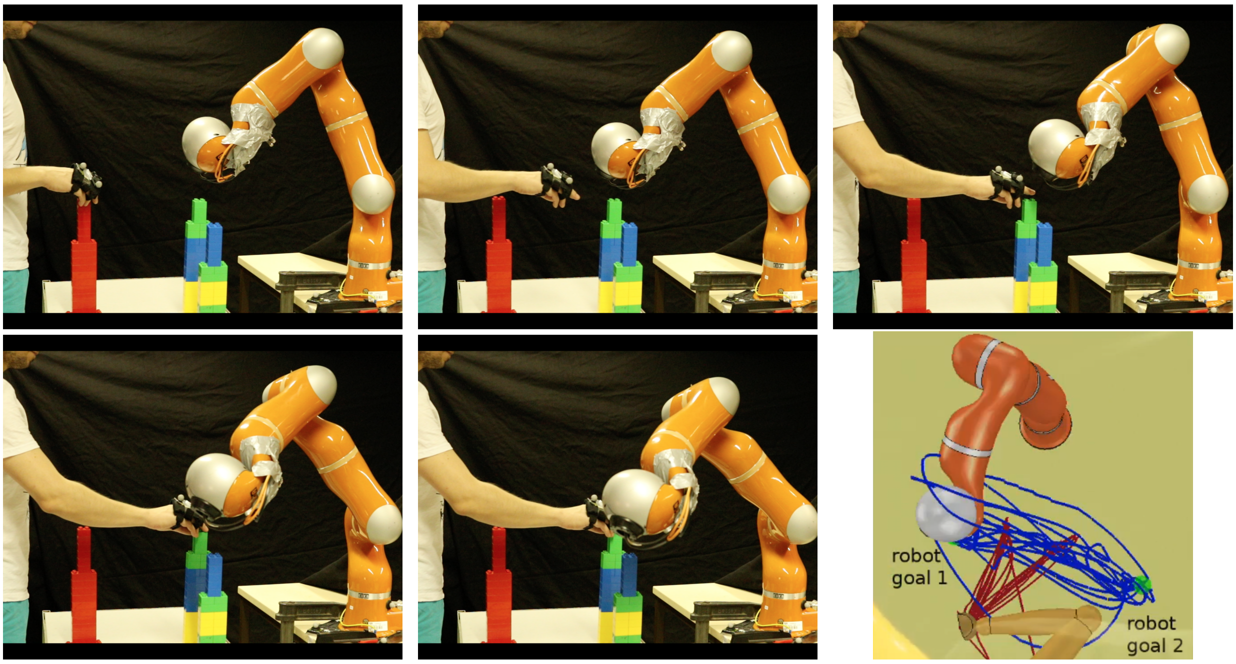


Figure 5.7: The scenes from a close proximity HRI with a Kuka robot, and the top-down view of the trajectories of the human hand and the robot end effector (lower-right).

Darker areas correspond to a strong negative correlation, where lighter colors correspond to a strong positive correlation in a large number of individual samples. Horizontal stripes with alternating color represent different correlations for different samples from the test set. Strong correlations between both of the two layers' states and the actual wrist position in x,y and z direction as well as for the z-direction of our feature $\mathfrak{f}_{H2G}$, which captures the wrist-to-goal distance, are observed. Besides, we also observe some non-uniform but strong correlation over several cases for the second features set $\mathfrak{f}_{H2H}$ particularly in x and y direction. Finally, there is some relevant but less strong correlation for velocities and accelerations in x and y directions for the units 4, 6, 8 in the layer 0.

5.4.6 Capabilities of the policy and robot implementation

While the presented extensive analysis was conducted in a simulated environment, we also constructed a proof of concept real world HRI setup using the LSTM architectures introduced in Sec. 5.4.3.c. A KUKA LWR4+ together with a damped least-squares IK solver was used in a close proximity HRI setup (Fig. 5.7). In this setup, the task requirements, i.e., the position of the initial and final position of the dyad, was different than the dataset used for training the policy. Still, the robot was successful at executing point-to-point reaching movements closer to a human partner.

The policy has also been tested for various other tasks in order to understand its capabilities further (Fig. 5.8). It has been found out that the policy enables the robot to execute a sequence of tasks smoothly - note that the training did not involve any explicit guidance on stopping and then restarting movement for another task. More notably, the policy was tested for goal switching in mid-flight, in an attempt to simulate a task selection due to some interruption, e.g. human partner blocking the path. The robot manages to change the task and successfully reaches the new target

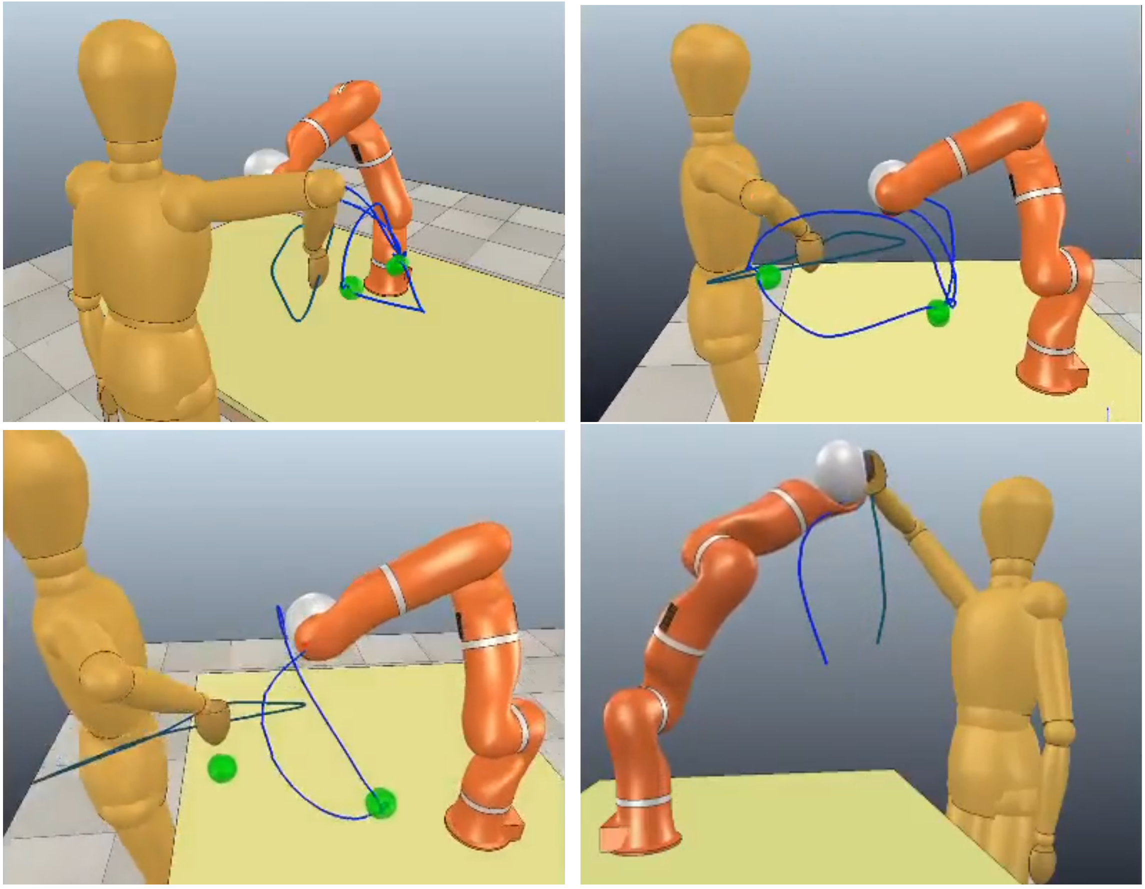


Figure 5.8: The scenes from a close proximity HRI with a Kuka robot (the real robot was controlled simultaneously, but for better explaining the interactions with visuals, simulated versions are used). The sequential task execution (top row, different perspectives), task change due to possible collision with human (bottom-left), and handover/high-5-like scenario where the robot follows a dynamic target (bottom-right) with human hand trajectories (dark green), robot end effector trajectories (blue) and robot target positions (green).

location without interruption in motion. Lastly, we tested whether the robot could follow a moving target to simulate a hand-over or high-5 scenario, and again, the robot was able to reach a moving target swiftly.

5.5 Discussion

Here, we provide our insights on the results and some limitations of this work. Regarding the design questions, GRUs together with triple-tanh activation perform slightly better than standard GRUs in terms of reaching time, whereas for goal position convergence and trajectory similarity LSTM based architectures outperform the others. Although the differences are not always notable, GRU architectures benefit from the triple-tanh activation function by being able to capture richer motion characteristics in a timing-sense, potentially due to the availability of multiple saturation regions around the inflection points. One critical reason for the good performance of LSTM architectures might be their increased number of tunable parameters for a fixed number of units in the comparison group (Eq. 5.1-5.6 vs. Eq. 5.7-5.10)

The proof-of-concept testing of the learned policies on a redundant KUKA robotic arm verified two things: first, learning from HHI data is suitable for HRI scenarios, and second, the proposed approach is feasible to learn and control close proximity interaction behavior online. Having said that, we also observed some critical points. As the policies are trained only from HHI data, some discrepancies on human movement occurs during interacting with a robotic partner. We argue that, human-like movement generation is a good starting point for the robot to imitate, and in a future work, this imitated control can be used as an initial policy to be improved with a policy improvement algorithm during interaction to adapt to the human preferences and task effectiveness.

The results, however, have some limitations. While the large number of successful test cases illustrate the effectiveness of the proposed learning framework, there is no formal guarantee for a collision-free and converging performance. Besides performance issues, we also need to consider the expressiveness of hand trajectories. Current robot implementation relies on a damped least-square inverse kinematics solver without any link collision considerations. Even though modeling the hand position characteristics are essential for natural dyadic interaction, the outcome of controlling multiple joints with such an approach is yet to be shown. Particularly distance-keeping between more joints becomes a bigger concern, but the acquired policy would enable tackling more complex cases.

5.6 Conclusion

In this work, we proposed a supervised learning framework to imitate close proximity dyadic interaction movement behavior. Control of a robot end effector is learned from human-human interaction (HHI) demonstrations to provide an effective and intuitive human-robot interaction (HRI). Generalized policies were acquired effectively by the proposed learning approach using recurrent neural networks (RNNs). The learned policies were successfully applied on a robot that interacts with a human partner in close proximity.

The capabilities to learn on-line control through off-line training are significant considering that there is no explicit modeling of the interaction scenario. Additionally, the achieved cross-domain mapping, from spatial data to acceleration control, encourages to use RNNs as continuous control policies for robotic interaction and manipulation skills.

For future work, there are several directions to consider. Besides an extension to joint position control, hierarchical control is another promising path to follow. Especially, considering the capability of our proposed approach to represent a variety of movement behaviors, from generalized and overt to subtle features, a high-level controller may be developed to provide an intuitive interaction by switching between multiple behavioral models depending on the task.

Part III

Close Proximity Human-Robot Interaction

6

Progressive Stochastic Motion Planning for Human-Robot Interaction[1]

This work introduces a new approach to optimal online motion planning for human-robot interaction scenarios. For a safe, comfortable, and efficient interaction between human and robot working in close proximity, robot motion has to be agile and perceived as natural by the human partner. The robot has to be aware of its environment, including human motions, in order to proactively take actions while ensuring safety, and task fulfillment. Human motion prediction constitutes the fundamental perception input for the motion planner. The prediction system, which is based on probabilistic movement primitives, generates a prediction of human motion as a trajectory distribution learned in an offline phase. The proposed stochastic optimization-based planning algorithm then progressively finds feasible optimization parameters to re-plan the motion online that ensures collision avoidance while minimizing the task-related trajectory cost. Our simulation results show that the proposed approach produces collision-free trajectories while still reaching the goal successfully. We also highlight the performance of our planner in comparison to previous methods in stochastic motion planning.

6.1 Introduction

Robots are envisioned to be present in everyday human life as well as in manufacturing. Independent from application scenarios, robots are expected to interact with human partners in a natural way, i.e. as similar as possible to the humans interacting with each other. There are three main requirements to achieve such a seamless interaction: *(R1)* Both, human and robot, have to achieve the goal defined by the task to be solved, *(R2)* robot has to be aware of the changes in the environment, including human motions, to adapt accordingly, and *(R3)* the robot behavior has to be reactive, yet still understandable by humans.

In human-robot close interaction scenarios, the robot trajectory generation is a challenging problem as it has to not only fulfill the aforementioned requirements but also consider the robot's

[1] This work has previously appeared in the following publication: [242]

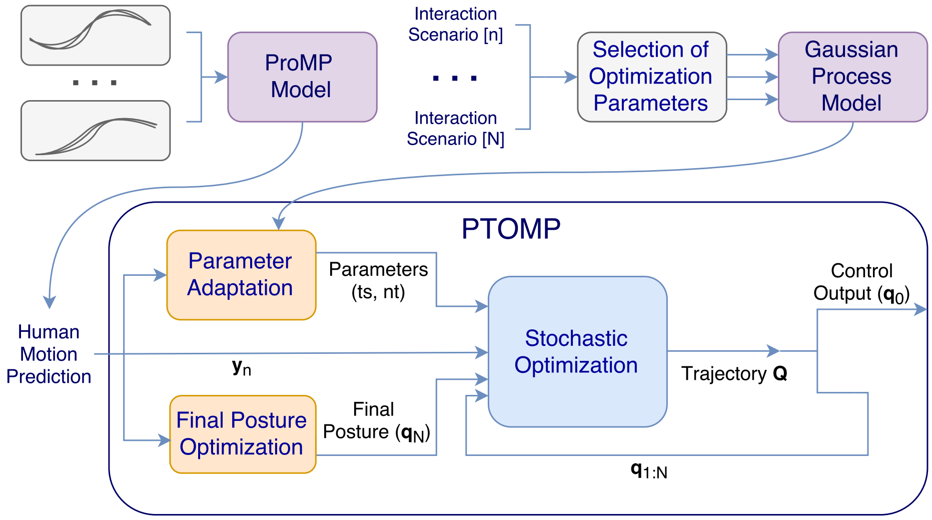


Figure 6.1: Overview of the framework: The Probabilistic Movement Primitive (ProMP) model that enables the motion prediction and the GP model that selects the optimization parameters are trained in the offline phase. The online motion planning is an adaptive process as the stochastic optimization is supported by updated prameterization and final posture as well as a human motion prediction.

capabilities (e.g. joint limitations) and the variations in human motions. The three requirements (R1-3) are highly interlinked. In order for the robot to react to ongoing human actions (R3), the robot has to be capable of predicting these human actions (R2). For achieving the required goals by both partners (R1), the robot has to plan a goal-directed motion trajectory while allowing the human partner to easily reach their target location (R2-3).

Recently, there has been an increased interest in close HRI. On the one hand, there are studies focusing on the safety aspect of such interactions [29], [123], where the human is treated as a dynamic obstacle which needs to be avoided while being indifferent to how humans perceive the interaction. On the other hand, few studies investigate motion models for the robot that can be perceived as legible [7], predictable [8], and understandable [30] by the human partner. However, all those aspects are interdependent on each other, which demands an integrated approach to motion planning for the robot.

In this work, we propose a motion planning framework that addresses the challenges in human-robot collaboration in a unified structure. Given that each partner has a task to fulfill, we seek to formulate a stochastic optimization model to generate robot motions that 1) take into account human motion prediction while 2) minimizing the trajectory cost of the robot, and yet 3) avoid collisions without obstructing the human partner's task execution. This establishes a shared workspace in which a human-robot team can work safely and naturally in close proximity.

An overview of the framework is presented in Figure 6.1. Our approach comprises a human motion prediction module along with a stochastic trajectory optimization method. Human motions are recorded for the set of tasks we are interested in, and then utilized for training probabilistic Movement Primitives (ProMPs) in an offline phase. During interaction, learned ProMPs generate predictive trajectory distributions for the human's motion given observed data. Based on this prediction, a final posture is optimized for the robot to avoid collision with the human as well as the feasible parameters for the motion optimization method is acquired w.r.t. the task conditions. As a last step, the motion trajectory of the robot is optimized efficiently with the identified parameters by taking into account the prediction and the optimal final posture to re-plan the motion online accordingly. The whole process is repeated at each time interval during task execution. This adaptive optimization process enables agile and flexible robot motion generation.

This work concentrates on the formulation of the stochastic motion planner which progressively optimizes the trajectory of the robot during interaction. We implement, test, and then analyze our approach both in simulation and in a preliminary human trial. Our analysis demonstrate the trade-off between safety and interaction related costs. Results show that the proposed method, by acknowledging this trade-off, enables a cautious interaction while providing locally optimal solutions for task achievement. Comparison to some state-of-the-art motion planners reveals the benefits of our approach both in terms of optimality and safety.

The novel approach we propose is a fundamental step toward safe and natural HRI in close proximity. The main contributions of this work are:

- We propose an integrated approach where human motion prediction is incorporated into a trajectory optimization method efficiently.
- The trajectory optimization is achieved online by a novel parameter adaptation method and progressively recomputed depending on the interaction dynamics.
- The proposed method solves for locally optimal trajectories given time constraints while still providing seamless interaction.

6.2 Related Work

Existing methods in motion planning for robotics arms are generally classified in two main branches that have received most significant attention over the last two decades, which are sampling-based and optimization-based approaches. Sampling-based approaches randomly generate sampled points in the configuration space at each iteration and check the collision of these points with obstacles by using a collision-checking component. A graph (roadmap) is then constructed by a set of collision-free points and is extended at every iteration until a solution from the start to the final position is found. As a result, these approaches become very effective in high-dimensional spaces since they do not require a full representation of the environment explicitly.

The most influential sampling-based motion planning algorithms to date include probabilistic roadmaps (PRMs) [203] and rapidly-exploring random trees (RRTs) [204]. Both algorithms rely on the idea of connecting points sampled randomly from the state space, although they differ in the way of constructing a graph connecting these points. One major advantage of sampling-based approaches is fast computation and therefore they can be easily applied online. However most of these methods treat human co-workers as constraints or as obstacles that the robot needs fulfill or avoid [205]. To the best of our knowledge, there is no existing sampling-based motion planning framework that considers joint collaboration between human and robot while planning.

Unlike sampling-based approaches, optimization-based methods exploit the whole configuration space and seek for the best possible solution which minimizes (maximizes) a performance metric.

In robot motion planning, optimization-based approaches help to save energy and imitate human motion by defining proper cost functions and consider different type of constraints in robot motions. With the broad validity of the cost functions, these planners can be easily adapted in different human-robot collaboration scenarios.

Optimization-based approaches for motion planning problem can be classified into two different branches: gradient-based optimization and stochastic optimization. A well-known gradient-based method is the CHOMP algorithm [206] which utilizes covariant gradient techniques for optimization calculations. The sampled trajectory is optimized within consecutive iterations where the covariant gradient update rule ensures convergence of CHOMP to a locally optimal trajectory. However, the main drawback of this method is that, the cost and constraint representations need to be differentiable, which is impractical for many real-life cases, especially in HRI, where uncertainties and disturbances exist in most of the cases.

An approach to bypass this problem is stochastic optimization, where STOMP [207] is a typical candidate. In STOMP, the numerical optimization is considered as a stochastic direct optimization. Here, instead of determining the gradient to update the trajectory numerically, STOMP uses a stochastic gradient estimation. This is basically done by generating noisy trajectories around a feasible initial trajectory and evaluating their performance in terms of the cost function to determine the gradient update. However, this prohibits the flexibility and adaptability. As the whole trajectory is computed prior to execution, it may become invalid when the environment changes.

To enable flexibility and adaptability in robot motion, planning while executing is considered. The ITOMP algorithm takes STOMP as starting point and introduces the interleaving of motion generation and execution [208]. By doing this, ITOMP gains the ability to be used for real-time motion generation.

Even though optimization-based methods are widely used in robotics, the application in human-robot interaction, or especially joint collaboration is still very limited. One of the reasons is that there is no proper way to define human-related cost functions which are varied depending on different kinds of tasks and human partners. A very recent research [27] uses inverse optimal control to learn the cost functions from sampled trajectories and then applies these cost functions to a STOMP-based optimization process to predict human motions and therefore provides better reactive motions for the robot. However, the method is only applied for a very specific, simple pick-and-place task.

A major drawback of optimization-based approaches is the heavy computation time, which makes it inapplicable in online scenarios if the environment or tasks are complex. There are some approaches that try to overcome this drawback i.e. combing optimization with machine learning techniques [209] or dynamic movement primitives [210]. However, they are still limited in the static cases or they neglect the optimality aspect in the online phase as a trade-off.

6.3 Stochastic Motion Planning

Here, we explain briefly the general structure of a stochastic trajectory optimization formulation [207] as the foundation of our motion planning framework for human-robot interaction in close proximity.

Stochastic optimization for motion planning relies on generating noisy trajectories around an initial (possibly infeasible) trajectory. Then, those trajectories are evaluated based on their costs. Calculating the cost of a trajectory requires a discretized representation of consecutive states. For a D-DOF robotic arm, a feasible state representation is defined in the joint space for the robot's configuration $\mathbf{q}_n \in \mathbb{R}^D$ at each time-step n. Concatenating the sequential postures of N time-steps, the trajectory representation $\mathbf{Q} \in \mathbb{R}^{N \times D}$ is obtained as:

$$\mathbf{Q} = [\mathbf{q}_1^T, \mathbf{q}_2^T, ..., \mathbf{q}_N^T]^T. \tag{6.1}$$

The d^{th} column of $\mathbf{Q}$ is indexed with $\mathbf{q}^d \in \mathbb{R}^N$ to represent the trajectory of the d^{th} joint as well. The generalized formulation of the cost function is given as follows:

$$\min_{\mathbf{Q}} \Big[J(\mathbf{Q}) = \sum_{n=1}^{N} C(\mathbf{q}_n, \dot{\mathbf{q}}_n, ...) + \sum_{d=1}^{D} \frac{1}{2} \|\mathbf{A}\mathbf{q}^d\|^2 \Big]$$
$$\text{s.t. } \ \mathbf{q}_0 = \mathbf{q}_{\text{init}}, \ \mathbf{q}_N = \mathbf{q}_{\text{final}} \quad \mathbf{q}_{\text{init}}, \mathbf{q}_{\text{final}} \text{ are const.}$$

where $C(\mathbf{q}_n, \dot{\mathbf{q}}_n, ...)$ is the sum of arbitrary state-dependent cost functions, $\mathbf{A}$ is a finite differencing matrix and $\frac{1}{2}\|\mathbf{A}\mathbf{q}^T\|^2$ represents the control cost as the norm of the accelerations in quadratic form.

NT number of sample trajectories, each having N number of time-steps, are generated by adding a noise covariance that ensures smoothness. As explained in [207], at every iteration the initial trajectory is updated with weights that are inversely proportional to the costs computed for the noisy trajectories. Running consecutive iterations by following the same procedure over the updated trajectory allows the result to approach to an optimal solution. Note that, the trajectory is planned as a whole, and not modified during the motion [207], thus, preventing any kind of reaction from the robot to the environmental dynamics.

We extend this standard formulation by introducing interleaving of motion planning and execution as in [208]. Interleaving in this context means the same optimization procedures are computed after the robot executes the control signal at each time-step. The motion is divided into $N-1$ intervals and the robot moves from the posture $\mathbf{q}_n$ to $\mathbf{q}_{n+1}$ during which the optimization continues for the trajectory $[\mathbf{q}_{n+1}^T, \mathbf{q}_{n+2}^T, ..., \mathbf{q}_N^T]^T$ similar to *receding horizon control*. The optimal trajectory varies at each interval depending on the updated cost function, and we employ this functionality to provide responsive behavior for the robot.

6.4 PTOMP

6.4.1 Overview

In this section, we introduce the **P**rogressive **T**rajectory **O**ptimization-based **M**otion **P**lanning algorithm for Human-Robot Interaction Scenarios (PTOMP). This algorithm forms the basis of a novel approach to adapt the previous work on stochastic motion planning to human-robot interaction by integrating human motion prediction, and caring for human partner's perception in the planned motion. The main contribution of our framework is concentrated on (*i*) choosing feasible parameters, i.e. number of noisy trajectories (NT) and number of time-steps (TS), for the optimization method to achieve online capability, (*ii*) incorporating the interaction related cost functions to provide a natural interaction, and (*iii*) enabling safe interaction by taking proactive actions based on an efficient human motion prediction.

Our framework (Figure 6.1) comprises offline and online stages. In the offline phase, a human motion library is constructed for each task as Probabilistic Movement Primitives (ProMPs). This library enables predicting the human motion online (Section 6.4.2). The prediction is fed into to trajectory optimization to solve for a collision-free optimal motion. In another offline phase, a comprehensive set of different conditions are simulated with varying optimization parameters, NT, TS. This allows learning the space of trajectory cost values for a multitude of tasks with different set of optimization parameters (Section 6.4.5).

In the online phase, trajectory optimization is run with a combination of cost functions (Section 6.4.4) by choosing the feasible parameters from the learned model and integrating human motion prediction, and final posture optimization (Section 6.4.3) to ensure natural and safe interaction.

6.4.2 Prediction

The occupancy estimation of a dynamic obstacle in the workspace is utilized for the collision cost calculation of future time-steps. Even if it is possible to make this estimation from the velocity and acceleration of the obstacle as in [208], its accuracy is likely to remain reliable only for a short period of time. In other words, the trajectory optimization will define a collision cost based on an unreliable estimation, thus hindering the safety of the avoidance behavior.

To avoid collision in the workspace, the human arm can also be considered as a dynamic object to be avoided. In general, human motion estimation requires a specialized prediction method as it depends on arm dynamics and its control results in variation in motion [211]. To imitate such behavior online, we use Probabilistic Movement Primitives (ProMPs) and learn a distribution of a motion behavior by training with multiple trajectories performed for a specific task [127]. ProMPs represent a discrete trajectory $X = \{x_n\}$, $n = 0 \dots N$ defined by states x_n over time N with the formulation

$$\boldsymbol{y}_n = [x_n, \dot{x}_n]^\top = \boldsymbol{\Phi}_n^\top \boldsymbol{\omega} + \boldsymbol{\epsilon}_y, \tag{6.2}$$

where ω is the weighting vector over the $k \times 2$ dimensional time-dependent basis matrix $\boldsymbol{\Phi}_n = [\phi_n, \dot{\phi}_n]$ with k being the number of basis functions and $\boldsymbol{\epsilon}_y \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_y)$ is zero-mean independent Gaussian noise, while $\boldsymbol{\Phi}_n^\top \boldsymbol{\omega}$ gives the mean of the trajectory. Introducing a Gaussian distribution to also represent variance $p(\boldsymbol{\omega}; \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\omega} | \boldsymbol{\mu}_\omega, \boldsymbol{\Sigma}_\omega)$ over the weighting vector ω results in the following distribution for the trajectory:

$$\begin{aligned} p(\boldsymbol{y}_n; \boldsymbol{\theta}) &= \int \mathcal{N}(\boldsymbol{y}_n | \boldsymbol{\Phi}_n^\top \boldsymbol{\omega}, \boldsymbol{\Sigma}_y) \mathcal{N}(\boldsymbol{\omega} | \boldsymbol{\mu}_\omega, \boldsymbol{\Sigma}_\omega) d\boldsymbol{\omega} \\ &= \mathcal{N}(\boldsymbol{y}_n | \boldsymbol{\Phi}_n^\top \boldsymbol{\mu}_\omega, \boldsymbol{\Phi}_n^\top \boldsymbol{\Sigma}_\omega \boldsymbol{\Phi}_n + \boldsymbol{\Sigma}_y). \end{aligned} \tag{6.3}$$

Using a set of motion observations, the parameters $\boldsymbol{\mu}_\omega$, $\boldsymbol{\Sigma}_\omega$ can be estimated with maximum likelihood estimation [136].

By this formulation, the ProMPs enable an online human motion prediction, where a trajectory along with the variance for each time point is generated. This variance information is useful for human-robot interaction scenarios where the robot should also consider the uncertainties of human behaviors.

6.4.3 Final posture optimization

While interacting with a human, the robot should also be expected to reach a final posture that conforms to a legible behavior. As the standard formulation of stochastic trajectory optimization uses a fixed final state, we also incorporate a final posture optimization step which precedes the motion planning phase. Moreover, in close interaction scenarios where both the robot and the human reach nearby targets, the final posture of the robot should be optimized so that it avoids collision. The algorithm searches for a final posture of $\mathbf{q}_N^*$ that satisfies the condition

$$\mathbf{q}_N^* = \arg\min_{\mathbf{q}_N} \left[J(\mathbf{Q}_{1\ldots N}^*) \,\forall\, \mathbf{q}_N \in \mathbb{R}^D \right] \tag{6.4}$$

and thereby enables the most optimal trajectory while still assuring the task of reaching the target position for the end-effector. As the optimization result could not be known beforehand we target an estimation that satisfies the equation 6.4. Here we assume that the optimal final posture that minimizes the cost of the current trajectory would also minimize the cost of the optimized trajectory. Therefore, during the run of motion planning we make the final posture optimization over the cost of the instant trajectory.

6.4.4 Cost functions

Interaction cost An efficient human-robot collaboration highly depends on the fluency of interaction between both sides. Just as the necessity of human motion prediction by the robot, the legibility of robot motion for the human subject is also crucial. Following the results of recent studies claiming that the legibility of a robot's motion comes from its closeness to human motion behavior [212], [213], we look into models to generate human-like motion. In motor control field, human motion generation has been studied as an optimization problem. Considering the total energy [53], torque change [4], standard deviation [33] of the motion has been suggested to be used as cost functions that describe human motion. Recent studies extended this claim by showing that human motion planning tends to reduce the effort while still ensuring a smooth motion that reaches the target [9]. To represent the effort we introduce the state dependent energy to the cost function.

$$\begin{gathered} J_{\text{effort}} = \sum_{i \in C} (m_i h_i + \frac{1}{2} m_i V_i^2), \\ C = \{\text{Upper arm}, \text{Lower arm}, \text{Hand}\} \end{gathered} \tag{6.5}$$

which is the sum of kinetic and potential energy during the motion. Alongside the total energy, the minimization of the hand jerk in cartesian space is also used [32]:

$$J_{\text{jerk}} = \int_0^T \dddot{x}^2 + \dddot{y}^2 + \dddot{z}^2 dt \tag{6.6}$$

Parameters	Number of Iterations	Efficiency of a single Iteration
Duration of the Motion	Linearly Proportional	No Relation
Number of Time-steps	Quadratically Inverse Proportional	Proportional
Number of Noisy Trajectories	Linearly Inverse Proportional	Proportional

Table 6.1: Summary of how the increase on the parameters influences the optimization.

with $[x, y, z]$ giving the cartesian space coordinates derived from joint configurations of the robot. The combination of those costs is referred as the *interaction cost*

$$J_{\text{interaction}} = J_{\text{effort}} + J_{\text{jerk}} \tag{6.7}$$

Control cost As humans can perform smooth and consistent motions in interaction scenarios, they also favor robot partners that can smoothly move in a shared workspace [212]. As the optimization is run on the kinematic level of the robot, we take the acceleration of the motion as the input signal to be minimized. The control cost, i.e. the smoothness cost, is defined as the sum of norms of acceleration in quadratic form at each time-step, and is represented by the term:

$$J_{\text{control}} = \sum_{d=1}^{D} \frac{1}{2} \|\mathbf{A}\mathbf{q}^d\|^2 \tag{6.8}$$

Collision cost One of the main task in human-robot interaction is to ensure collision avoidance in order to satisfy the safety restrictions and provide efficient collaboration. As collision avoidance is a part of the natural interaction, it also helps the legibility of the robot's motion.

For the collision cost calculation, the algorithm needs to determine the collision point at first. To reduce the calculation time, we define the lower and upper arms of both the robot and the human subject as cylindrical shapes. Through a trigonometric calculation, we determine the distance and the location of collision as:

$$[dist_n, loc_n] = \text{collision}(\mathbf{q}_n, \mathbf{y}_n) \tag{6.9}$$

where the location is the distance of collision to the robot's end-effector. As the collision cost minimization should be driving the robot away from the human arm, we define a continuous cost function that decreases towards the end-effector of the robot:

$$J_{\text{collision}} = \begin{cases} K * loc_n, & \text{if } dist_n \leq dist\_limit \\ 0, & \text{otherwise} \end{cases} \tag{6.10}$$

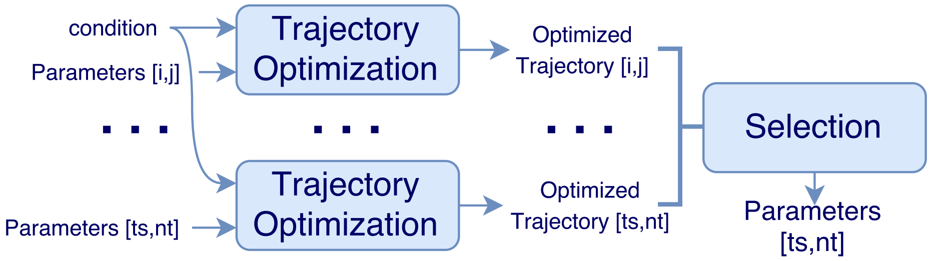


Figure 6.2: Parameter selection from simulations under a constant condition.

with $dist\_limit$ indicating the collision limit.

Joint limitations As the servo motors of the robot can perform limited rotations, the motion planning requires a limitation for the joint rotation to comply with the physical limitations of the robot. Moreover, to satisfy a legible motion, the robot needs to be restricted to a similar rotation limitation as human joints. Such limitations can be provided by setting a threshold limit. Instead of using such threshold limits as constraints in the optimization, we incorporate the joint limitation as a another cost function which remains continuous and dramatically increases at the limitations. For example, to keep the d^{th} joint rotation between $-L$ and $+L$ degrees, we set the joint limitations function as:

$$J_{\text{limit}} = K_1 * \mathrm{e}^{K_2\left(|\mathbf{q}^d|-L\right)} \tag{6.11}$$

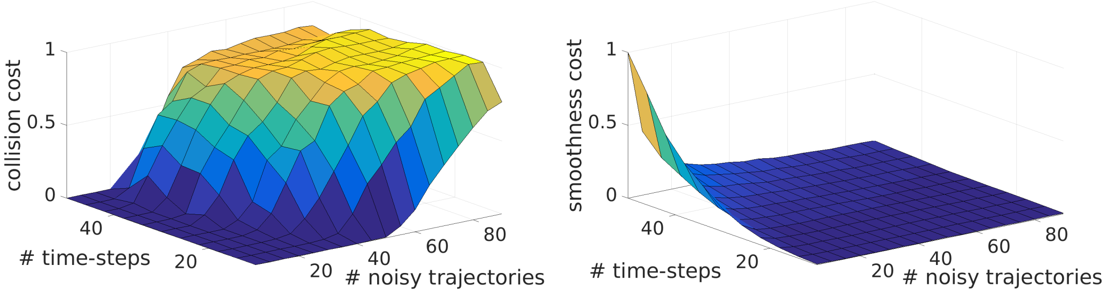
where the higher values of the coefficient K_2 provides a sharper increase of cost at the limitation.

6.4.5 Parameter adaptation

As real-time motion planning requires the optimization to run simultaneously with the execution, there is a certain time allowed for the calculations at each interval. In stochastic optimization, the calculation consists of consecutive iterations, where the trajectory converges to optimality. If the time limitation does not allow to run enough number of iterations to converge to the optimality, then the resulting trajectory is considered as sub-optimal. This sub-optimal result might still satisfy the requirements defined by the cost functions. However, the collision must not be tolerated as it concerns the safety criteria and hinders the task execution.

In our analysis we have observed that even under the same scenario, different trajectories are generated when the optimization is run with different parameterizations. These results are obtained as the optimization parameters influence the performance of optimization. Reducing the calculation time for each optimization iteration proportionally increases the number of performed iterations which is expected to support optimality, however, the quality (efficiency) of iterations also play a role. The three main parameters that influence both the calculation time and the quality of each iteration are (1) the number of time-steps discretizing the trajectory in time, (2) the number of noisy trajectories generated at each iteration and (3) the duration of the robot motion.

The discretization (resolution) of the trajectory changes the calculation time of a single optimization interval as more time-steps for the trajectory means higher load of cost calculation. However, choosing a very low discretization reduces the smoothness of the trajectory. It also hinders the robot's reaction to the environmental dynamics due to the reduced cost function update frequency.



(a) Plot of collision cost

(b) Plot of control cost

Figure 6.3: Example figures for a collision avoidance scenario simulated under the same environmental conditions, but with different parameterizations.

Similarly, having higher number of generated noisy trajectories increases the calculation time at each iteration, but at the same time provides smoother trajectories. Lastly, an increase on the motion duration clearly allows more time for the optimization, but results in a slower motion for the robot which can prevent efficient interaction. A summary of how these three parameters influence the number of iteration run in a single interval and the efficiency of a single iteration is given in Table 6.1. This analysis indicates that, the most feasible optimization parameters can not be easily selected as there is a trade-off in terms of their effect on optimization performance.

6.4.6 Criteria of parameter selection

To determine the feasible parameters we evaluate their performance in terms of the costs over the optimized trajectories. We run simulations with varying parameterization over a grid under a constant condition as shown in Figure 6.2. For this evaluation, we consider the collision and the control cost of the optimized trajectories. The visualizations of how these costs are effected by the parameterization under an example condition is given in Fig 6.3. An acceptable performance requires to be capable of ensuring obstacle avoidance. Therefore, any parameter set that has a non-zero collision cost is considered as poorly performing and filtered out for the parameter selection. A higher discretization of trajectory increases the response frequency to environmental dynamics since the optimization is recomputed for each time-step. Hence, the parameter selection algorithm opts for choosing as many time-steps as possible after the initial filtering. However, this causes an increase in the control cost (Figure 6.3b). Therefore, a strategy for choosing a feasible parameter considering the smoothness cost is necessary as well. From our preliminary analysis, we found setting a limitation of 10% of maximum smoothness cost as a criteria for the secondary filtering provides good results consistently. Thus, the maximum values are chosen from the filtered subset of parameters. The procedure of this parameter selection is also detailed out in Algorithm 1 as a pseudocode.

6.4.7 Formulation of parameter selection

For the online selection of the optimization parameters, we need to provide a formulation that returns the parameter set when the conditions defining the dynamics of the scene are given. As this

Algorithm 1 Parameter Selection Algorithm

```
For a given condition cond, max number of time-steps TS, and noisy trajectories NT:
  for $i = 0, i \leq$ TS, $i$++ do
    for $j = 0, j \leq$ NT, $j$++ do
      if $J_{\text{collision}}(traj([i,j] \mid \text{cond})) = 0$ and
      $J_{\text{control}}(traj([i,j] \mid \text{cond})) \leq 0.1 * max(J_{\text{control}})$ then
        Store $\mathbb{A} \leftarrow [i,j]$
      end if
    end for
  end for
  ts = $max(k \mid [k,l] \in \mathbb{A})$
  nt = $max(l \mid [ts,l] \in \mathbb{A})$
```

formulation cannot be achieved simply by a heuristic method online, we rely on building a generalized model that can map interaction conditions as the input to the optimization parameters as the output. This necessitates defining the conditions of the environment that influence the optimization the most. The *prediction uncertainty* and the *prediction error* are two metrics that require the optimization to be more reactive against a possible collision. The *distance between the robot and human arm* and the *remaining time until the collision* for the sub-optimized trajectory also determine how quickly the optimization requires to reach a solution such that it ensures a collision-free motion. Finally, the *immediate collision cost* indicates how close the current trajectory is to optimality. Combination of these five metrics define the state of the condition and serve as the inputs of the parameter selection function.

The parameter selection function is trained with Gaussian Process Regression Models [135]. Here, the training requires a set of real input and output data to fit a function estimation. To train for this formulation we run a set of simulations, by varying environmental conditions and recording a list of the feasible parameters selected. The metrics that define the condition of the scenario are taken as the input data while the selected optimization parameters are the outputs. As the function estimation of the Gaussian Process returns a single output, we need to train different models for different output parameters separately.

6.5 Results

Here, we present the results of our approach, PTOMP, on collision avoidance performance in online HRI scenarios and compare to a state-of-the-art motion planner, ITOMP. The efficiency of the GP-model based parameter selection, and its effect on the performance of PTOMP are also analyzed. Lastly, the results for the final posture optimization is provided. In these analysis we use the KUKA LWR4™ robot with 6 degrees of freedom.

In order to emphasize the influence of optimization parameters (NT, TS) on the resulting robot motion trajectory, an example HRI scenario is simulated where both agents execute their reaching motions, and the robot utilizes two different strategies (Figure 6.4). One subject's recorded data is used for simulation synchronously with robot's motion execution. The human motion is predicted

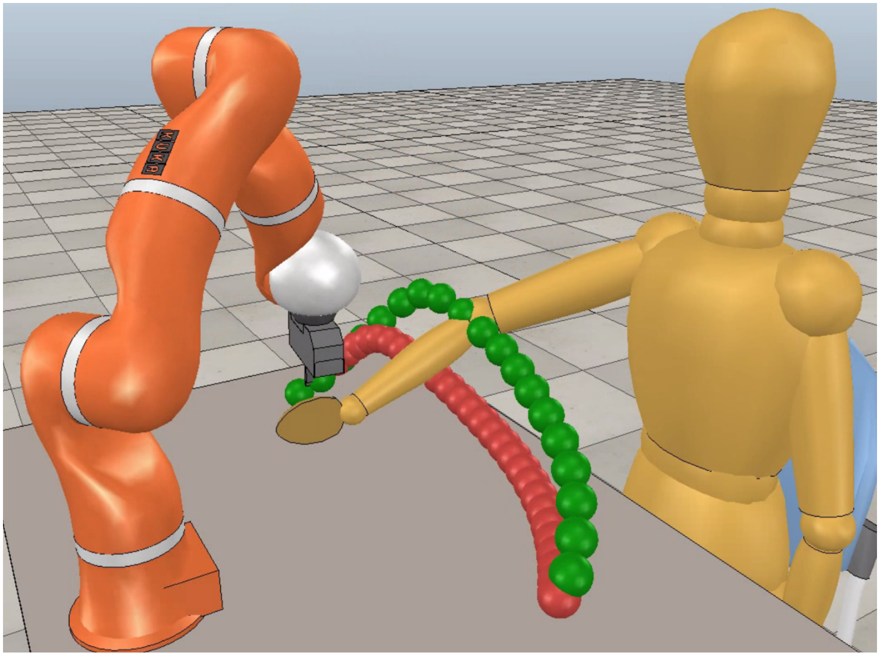
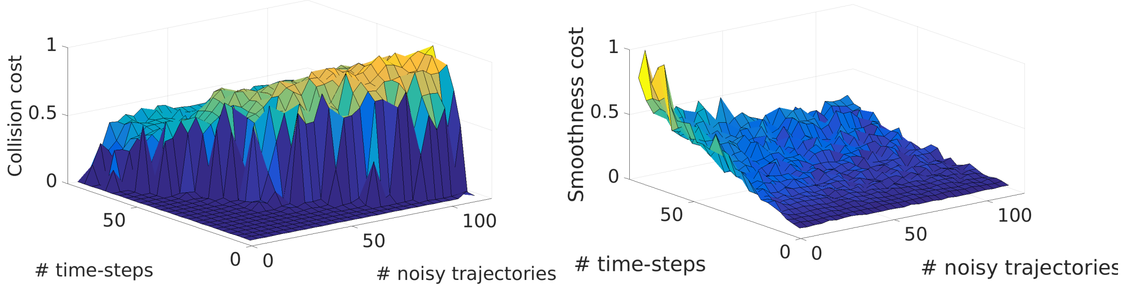


Figure 6.4: Interaction scene in an example HRI scenario. The green trajectory represents the robot's motion optimized by PTOMP. A second motion optimized with infeasible parameters is also shown with the red trajectory.

based on the ProMP model learned offline for similar reaching tasks. PTOMP uses this prediction to generate a collision free trajectory early on by using GP-model based parameter selection (green in Figure 6.4). In our example, we choose the duration of robot's reaching motion as 1.5 seconds. The results of the parameter selection function under the given conditions were 42.87 time-step discretization and 30.23 noisy trajectories per iteration, which are rounded down to (42, 30) (note that the most feasible parameters are (45, 32) if the simulation analysis is directly used rather than GP-model output). To visualize a comparison we run a second simulation with 55 time-steps and 50 noisy trajectories, which are close to the first case but arbitrarily chosen. However, these parameters turn out to be infeasible for this scenario (Figure 6.4 red trajectory). This shows that, without a proper parameter identification, such a motion planner is prone to fail in avoiding the collision.

For evaluating the effectiveness and reliability of parameter selection method, we run similar tests under 450 different conditions. Here, we train the GP model with 400 conditions and spare the other 50 for the evaluation which is based on comparing the results of the parameter selection model to the results of the simulation analysis. The error between both outputs shows a standard deviation of 4.17 for the number of time-steps, and a standard deviation of 5.48 for the number of noisy trajectories. Hence, the GP-model based parameter selection remains in a close range to the actual values used for training. In addition, with parameter adaptation, the collision avoidance is achieved with a success rate of 89 percent while it reaches 98 percent when we apply a heuristic by reducing both the number of time-steps and noisy trajectories by the standard deviation found in the analysis phase.

We evaluate the contribution of human motion prediction to the optimization problem. Here, we make the comparison with ITOMP's velocity-based occupancy estimation approach. We run these



(a) Plot of collision cost

(b) Plot of control cost

Figure 6.5: The plots show that the obstacle avoidance is provided under lower number of time-steps and noisy trajectories. Here the obstacle cost appears to be higher with less time-steps when collision is not avoided. This is due to the uncertainty of the human motion which was not included in Figure 6.3.

tests under the same scenario used for Figure 6.4. The trajectories followed by both algorithms are given in Figure 6.6. We also run these comparison tests for other 450 scenarios that we used for parameter selection evaluation. Running these tests with the same parameter selection output both for ITOMP and PTOMP, ITOMP succeeded to provide a collision-free trajectory only at 67 cases which correspond to a success rate of 14.88 percent in comparison to PTOMP's success rate of 89.34 percent. With reducing the number of time-steps and noisy trajectories again with their corresponding standard deviations in parameter selection function, ITOMP's success rate increases to 27.11 percent.

Finally, we evaluate the performance of the final posture optimization in two scenarios. In the first case (Figure 6.7a) the robot executes a pick and place task as the human motion does not interrupt its trajectory. Here, the final posture satisfies the legible motion as the robot leans to the side that it is approaching from. This behavior is due to the total energy minimization cost over the trajectory. In the second scenario we introduce a human motion that occupies a space where it causes a collision at the initial final posture as shown in Figure 6.7b. To avoid the collision, final posture optimization drives the robot to approach its target from the other side while trajectory optimization plans a collision-free trajectory that complies with the new final posture.

6.6 Conclusion

We have presented PTOMP, an optimization-based online motion planning algorithm for human-robot interaction scenarios. Our approach concentrates on providing the interaction efficiency to robot motion planning in close human-collaboration by introducing legibility criteria while increasing the optimization performance of the online computation. Our algorithm is powered by a high-precision human motion prediction method, ProMP, that both allows the algorithm to run the optimization for future states early on as well as enabling the parameter adaptation of the optimization algorithm for a performance boost. We have highlighted the optimization performance of the motion planning under the human motion prediction and also shown the task execution success increased by the online parameter adaptation. The test results support that our algorithm exceeds the performance of the previous work in terms of robot motion legibility and trajectory optimality.

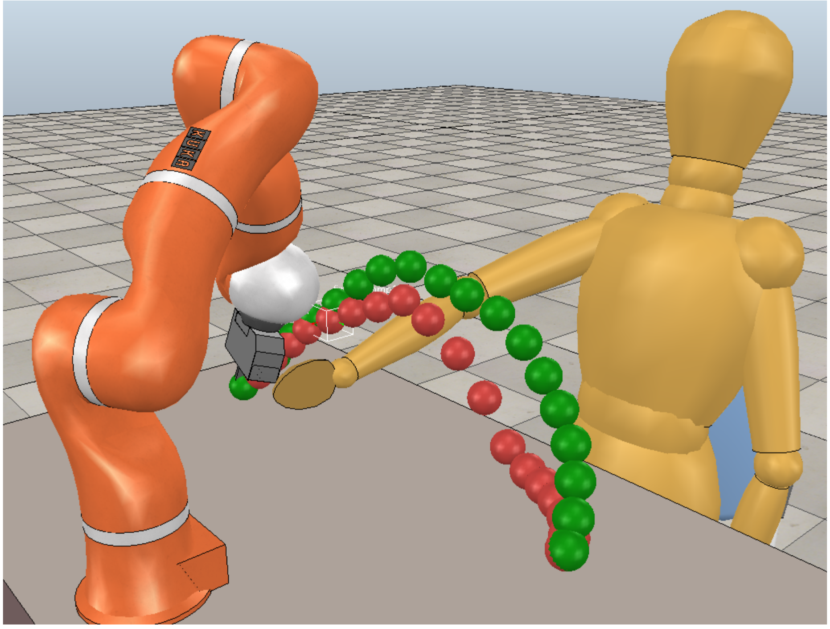
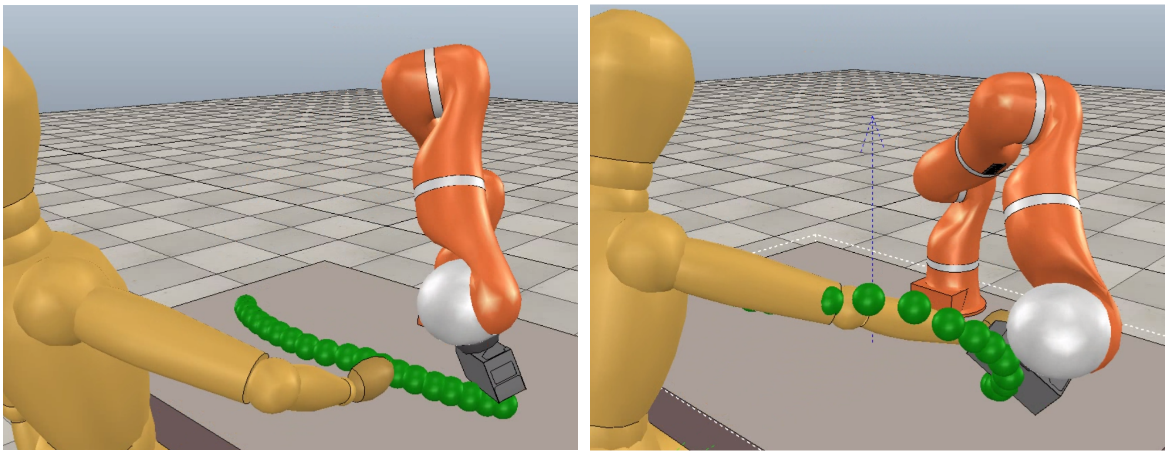


Figure 6.6: Comparison of PTOMP (green) and ITOMP (red) generated trajectories. PTOMP relies on ProMP-based human motion prediction whereas ITOMP uses its default short-sighted prediction. The motion planning of ITOMP fails to react until the human hand almost interrupts its trajectory, while PTOMP enables taking a proactive action and thereby plan a legible optimal motion.



(a) Final posture without the human interruption

(b) Final posture in close human collaboration

Figure 6.7: (a) shows that the final posture optimization drives the robot to lean right in order to comply with the interaction costs. However, when the human hand occupies the right side of the target as seen on (b) the optimization orients the posture to avoid collision.

In our algorithm the legibility of the robot motion depends on the human motion planning criteria in obstacle-free space. The interaction is then provided by considering the human arm as a dynamic obstacle to be avoided. This combination does not take into account the human response to robot's motion. The observations on real human-human interaction scenarios can be utilized in order to replicate a similar interaction. In that regard, a possibility for future work is to adapt the PTOMP algorithm to a humanoid robot.

7

Human Guided Policy Improvement for Predictable and Safe Robot Motion Generation

In this work, we explore a method to learn safe robot motions that adapt to the human preferences to facilitate close human robot interaction. Safety, legibility and predictability of robot motion have always been investigated independently. Several methods were proposed that produce real-time obstacle avoiding trajectories, while others developed optimization based algorithms for legible robot motions. Optimization of a fixed policy enables effective and safe close proximity HRI (Sec. 6.4), yet humans might have different expectations and interpretations of their robotic partner's movements. The research question in this line of work is whether control policies can be learned through close interaction with humans that provide safe and *readable* movement for human partners. By *readable*, we mean not only a predictable goal (task) position for the robot, but also human *understandable* motion patterns so that a natural interaction can be achieved.

7.1 Introduction

Nowadays, robots are no longer only industrial machines behind fences. Instead, they are being integrated more in our daily lives as well as in collaborative manufacturing scenarios. The motivation is to combine the experience and flexibility of humans with the power and preciseness of robots. In such scenarios, robots are expected to move in a natural way, similar to human-human interaction. When two humans perform a task together, they can anticipate each other's movements and perform a complementary action without the need of verbal communication. This facilitates teamwork and increases the efficiency of joint tasks [214]. To achieve such an interaction between humans and robots, the robot's motion has to be readable to the human [215], and be aware of its surroundings to provide a safe environment, while still being efficient in performing its task.

In order for humans to feel comfortable working with robots in close proximity, they have to understand the robots' behavior and be able to infer their actions. Identifying the factors that contribute to these natural robot movements is not trivial. In a study conducted in [216], participants wanted

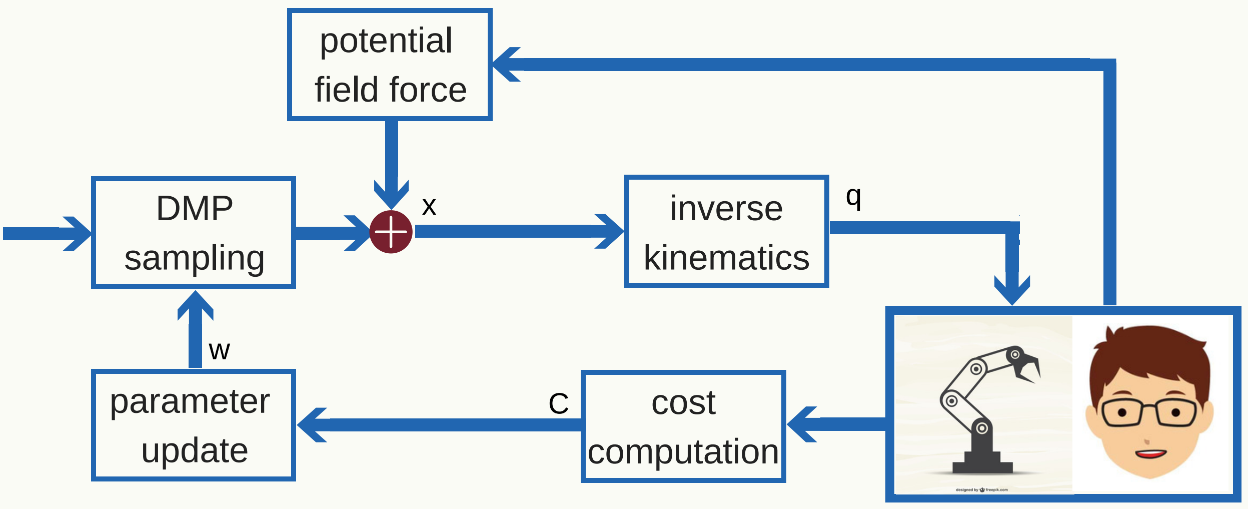


Figure 7.1: Pipeline of the algorithm: The DMP is the policy for the learning algorithm. We sample in parameter (DMP weight) space, convert the trajectories into joint positions using inverse kinematics and perform the roll-out online while adding the force resulting from the potential field. After the roll-out and an interaction iteration are complete, we compute the cost resulting from the interaction and update the policy for the next sampling phase.

robots assisting at home to be predictable, controllable and have human-like communication. Another study that investigated the subjective effects of direction of approach and distance of robots when handing an object over to humans, came to the conclusion that the frontal approach which is the most predictable is subjectively preferred most by the participants [217]. In addition, Bortot et al. [218] discovered that understanding and predicting the behavior of the robot increases the well-being of humans. However, predictability alone is not sufficient if we want robots and humans to work in close proximity. In order to ensure the safety of humans in such scenarios, the robot has to modify its trajectories in real-time to reliably avoid collision with the human. Combining this safe behavior with predictability maximizes human comfort.

In this thesis, the interdependency between readability, safety and efficiency is tackled for achieving natural human-robot interaction. A reinforcement learning method and inverse kinematics are combined to produce robot joint trajectories that are both *safe* and *readable*. The dynamic movement primitives (DMPs) formulation is used with policy improvement through black-box optimization (PIBBO) algorithm to learn readable motion generation during online close proximity interaction with human partners (Fig. 7.1). Both end effector position and joint position controllers are developed and tested for their effectiveness. In addition, we extend DMPs with a dynamic potential field approach, and define null-space constraints to ensure avoidance for the robot links. Finally, we test the transferability of the learned policy to another task in a human study.

7.2 Related Work

Legible and predictable robot motion was first defined by Dragan et al. [7]. In their work, the authors differentiate legibility and predictability and provides a mathematical model to produce and evaluate such motions. They assume that humans expect robots to be efficient in their movements and compare all possible goals in the scene to determine the most probable one. This probability is formulated in an equation and is being maximized for the targeted goal. This approach has some limitations. The algorithm was tested only with two goals for the robot, which the human had to predict by pausing a video which showed the robot moving to one of the two. This setup was very simple as the probability between two goals is already 50%. Another limitation is that the subjective evaluation of robot efficiency differs from one individual to another and the robot did not adapt its movements to the preferences of each participant.

In the work of Stulp et al. [8] the team generates robot motions that learn from interaction which motions are most predictable to the current participant. Here, dynamic movement primitives (DMPs) are used for motion planning and policy improvement through black box optimization, which was also formulated by Stulp et al. [219], is applied to improve the robot's legibility to the human iteratively. This is done by only optimizing human guessing time about the action of the robot and the correctness of the prediction without having formal criteria about legibility. In a recent study [220], the authors showed that transferring the learned policy to other individuals leads to better prediction in the beginning and can thus lead to shorter adaptation times for new subjects. However, in this approach no close interaction scenarios were considered as no necessary collision avoidance methods were integrated.

In [242], a stochastic motion planning algorithm is introduced that predicts human motions and adjusts the robot's trajectories on-line to avoid the predicted region. This allows close interaction between humans and robots, but does not examine predictable or legible motion. Another method for obstacle avoidance was presented by Park et al. [221]. The goal is to adapt trajectories while avoiding obstacles in mid-flight. The DMPs for movement reproduction are combined with dynamic potential fields for obstacle avoidance. Computing inverse kinematics with null-space constraints further ensured link collision avoidance. This method is successful in avoiding dynamic obstacles in the way of the robot. However, the aim of the work was not about enabling the robot to interact with humans, but rather to perform desired movements in the presence of obstacles.

7.3 Methods

7.3.1 Dynamic movement primitives

In order to generate robot trajectory for the policy we use dynamic movement primitives (DMPs) [222]. The DMPs combine two systems. First one is a closed loop critically damped spring-damper system that reaches the defined attractor state, which in our case is the goal position, y_g, for each target.

$$\tau\ddot{y} = \alpha(\beta(y_g - y) - \dot{y}) \tag{7.1}$$

τ is the time constant, while α and β are positive constants. By setting β to $\alpha/4$ we get a critically damped system. The variables y, $\dot{y}$ and $\ddot{y}$ are the position, velocity and acceleration, respectively.

The second system in the DMP is the forcing term. In order to deform the trajectories and achieve any desired shape, the spring-damper system is modulated with this non-linear forcing term $f(x)$.

$$\tau \ddot{y} = \alpha(\beta(y_g - y) - \dot{y}) + f(x) \tag{7.2}$$

The forcing term $f(x)$ is the weighted summation of Gaussian basis functions multiplied by a canonical dynamical system, x, and is obtained by the following equation

$$f(x) = \frac{\sum_{i=1}^{N} \psi_i(x)\omega_i}{\sum_{i=1}^{N} \psi_i(x)} x, \tag{7.3}$$

with

$$\psi_i(x) = exp(-\frac{1}{2\sigma_i^2}(x - c_i)^2) \tag{7.4}$$

defines the Gaussian basis functions with centers c_i and variances σ_i, N is the number of basis functions and ω_i are adjustable weights, which will be optimized by the policy improvement method explained in 2.3. The canonical system x replaces the time dependency of the forcing term. It goes from 1 to 0 during a movement ($x_{\text{init}} = 1, x_{\text{final}} = 0$) and is obtained by the equation

$$\dot{x} = -\alpha x, \tag{7.5}$$

where α is a predefined constant. This ensures convergence to the goal while keeping the forcing term not directly dependent on time.

Since the spring-damper system leads to high initial accelerations, which is not desirable for robots, we use a goal system, which moves the attractor state of the system from the initial state y_0 to the goal state y_g during the movement. This delayed goal attractor y_{gd} itself is represented as an exponential dynamical system that starts at y_0 and converges to y_g.

$$\dot{y_{gd}} = -\alpha_g(y_g - y_{gd}) \tag{7.6}$$

Thus the DMP equation becomes:

$$\tau \ddot{y} = \alpha(\beta(y_{gd} - y) - \dot{y}) + f(x) \tag{7.7}$$

The DMP has several advantages, which make it very suitable for our application:

- It is guaranteed to converge to the goal, since the canonical system is 0 at the end of every movement.
- the weights ω_i can be adapted to generate any desired trajectory. In our case this is especially relevant, since we want to learn the optimal trajectory and adjust the weights online with each interaction.
- As there is no time-dependency, the duration of the movement can be altered just by changing τ.

For our purposes we use a modified DMP to ensure obstacle avoidance.

$$\tau \ddot{y} = \alpha(\beta(y_{gd} - y) - \dot{y}) + f(x) + \varphi(x, v) \tag{7.8}$$

This added factor $\varphi(x,v)$ results from the potential field, which will be introduced in the following chapter.

Implementation We use one DMP with three goal systems for the three cartesian positions of the end-effector. Previous work with similar setups has used the DMP directly in joint space [8]. This works well for predictability adaptation. However, as we want to incorporate obstacle avoidance, we chose to run the DMP in end-effector space as opposed to joint space. This has two main advantages:

- It allows us to add the repellent forces resulting from the obstacles directly to the differential equation of the DMP without transforming.
- It enables setting the start and attractor state of the roll-out directly in the task space of the robot, and thus giving the robot freedom to position the rest of the joints in the null-space to avoid obstacles.

We use 20 equally spaced Gaussian radial basis functions for the DMP and have 4 samples per update for each goal for the exploration with 10 parameter updates.

In the sampling phase, we add small perturbations to the DMP parameters and run the policy for each sample. With each iteration we let the variance factor for the perturbations decay, so that the trajectory eventually converges.

7.3.2 Artificial dynamic potential field

With artificial potential fields, operational space of the robot is regarded as a field of forces [223], [224]. The goal that has to be reached has an attractive force, while the obstacles that have to be avoided exert a repulsive force on the manipulator. In our case we will only use repulsive forces, as the attraction to the goal is guaranteed by the DMP. The additional term $\varphi(x)$ in eq. 7.8 is then the force resulting from the potential field

$$\varphi(x) = -\nabla U(x), \tag{7.9}$$

where the dynamic potential field is defined as in [221] (eq. 7.10), with the following properties:

- The magnitude of the potential field decreases with the distance of the end-effector x to the obstacle (similar to a static potential field).
- The magnitude of the potential field increases with the speed of x and is zero when the speed of x is zero.
- The magnitude of the potential field decreases with the angle between the current velocity direction of x relative to the obstacle and the direction towards the obstacle (Fig. 7.2). In addition, the magnitude is zero, if θ is over 90° (i.e., the end effector moves away from the obstacle).

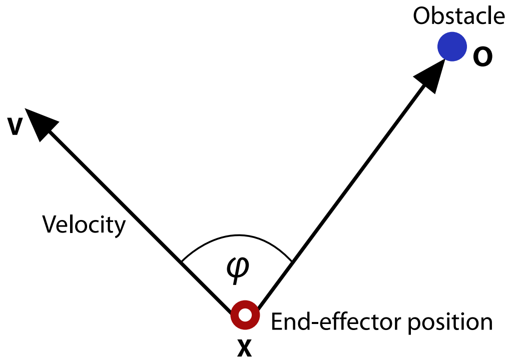


Figure 7.2: The steering angle φ between the velocity vector, $\boldsymbol{v}$, and the vector from the end-effector position to the obstacle position, $p(x)$.

The dynamic potential field is defined as

$$U_{dyn}(x,v) = \begin{cases} \lambda(-cos\theta)^{\beta}\frac{\|v\|}{p(x)}, & 0 \leq \theta \leq \frac{\pi}{2} \\ 0, & \frac{\pi}{2} < \theta \leq \pi \end{cases} \tag{7.10}$$

where v is the velocity, λ is a constant for the strength of the entire field and β is a constant. θ can be obtained by

$$cos(\theta) = \frac{v^T x}{\|v\|p(x)} \tag{7.11}$$

and is limited to a range between 0 and π. For moving obstacles, x and v can be defined as the relative position and velocity to the obstacle respectively. The force $\varphi(x)$ is then computed as

$$\varphi(x,v) = -\nabla_x U_{dyn}(x,v) = \lambda(-cos\theta)^{\beta-1}\frac{\|v\|}{p(x)}(\beta\nabla_x cos\theta - \frac{cos\theta}{p(x)}\nabla_x p(x)) \tag{7.12}$$

for $0 \leq \theta \leq \frac{\pi}{2}$ and can be used to modify the DMP as in eq. 7.8.

It is worth noting that the DMP loses its goal convergence property due to this additive force, if the obstacles are positioned very closely to the goal. In our experiments we avoid such scenarios.

Implementation We use a motion capture system to detect the position of the human (and the velocity), which are then used to compute the repulsive force. This is done at every time step of the rolled out trajectory, before computing the next joint positions.

7.3.3 Policy improvement through black-box optimization

We are using a policy improvement method to iteratively update the weights of the DMP to achieve the desired trajectory. All policy improvement methods have two basic steps:

1. Exploration by perturbation: The exploration noise ϵ_t can either be added to the actions, i.e. the output of the policy ($\pi_\theta(x) + \epsilon_t$), or directly to the input parameters of the policy ($\pi_{\theta+\epsilon_t}(x)$).

2. Policy update: Here, the parameters of the policy get updated in order to minimize the total cost. Gradient descend follows the negative gradient of the cost function to find the local minimum. Another - in our application more favorable - method is the reward-weighted averaging. There are other policy update methods, which will not be covered in this thesis.

Reward-weighted averaging does not require differentiability of the cost function, which makes it more stable than gradient descend, if the cost function is not continuous.

Specifically for this project we chose Policy Improvement through Black-box Optimization (denoted PI^{BB}) as our policy improvement method. This learning method has been proposed by Stulp et al. in [219]. As PI^{BB} is a black-box optimization method, it treats the whole roll-out as a black-box and does not have knowledge about the individual actions. Thus, it is a parameter perturbing approach. The output u_t of the policy is computed as:

$$u_k = \pi_{\theta+\epsilon_k}(x), \quad \text{with} \quad \epsilon_t \sim \mathcal{N}(0, \Sigma) \tag{7.13}$$

In our case the policy π_θ, is the DMP and θ are the corresponding weights for the Gaussians.

The parameter update is done using reward-weighted averaging. First, the cost C_k for each roll-out is computed. Then we assign higher probabilities P_k to trajectories with a lower cost and vice versa.

$$P_k = \frac{e^{-1/\lambda C_k}}{\sum_{k=1}^{K} e^{-1/\lambda C_k}} \tag{7.14}$$

K is the number of roll-outs and λ is a constant between 0 and 1.

The parameter update is then as follows:

$$\delta\theta = \sum_{k=1}^{K} P_k \epsilon_k \tag{7.15}$$

$$\theta \leftarrow \theta + \delta\theta \tag{7.16}$$

After taking the weighted average of all roll-outs, the new trajectory is closer to the trajectories with less cost. This process of perturbing and updating is repeated until the desired cost value is achieved or the maximum number of updates is reached.

Implementation In our application, the DMP is the policy π_θ, and the weights of the Gaussian functions are the parameters θ. The exploration is done by rolling out different trajectories and evaluating them using the cost values resulting from the interaction with the human. The cost function is detailed in section 7.3.5.

7.3.4 Inverse kinematics with null-space constraints

As the output of the DMP is in cartesian space, forward and inverse kinematics model of the KUKA LWR IV are used for joint-space control.

Link	a_i	α_i	d_i	θ_i
1	0	$\pi/2$	0	θ_1
2	0	$-\pi/2$	0	θ_2
3	0	$-\pi/2$	0.4	θ_3
4	0	$\pi/2$	0	θ_4
5	0	$\pi/2$	0.39	θ_5
6	0	$-\pi/2$	0	θ_6
7	0	0	0	θ_7

Table 7.1: Denavit-Hartenberg parameters. θ_i are the joint variables that have to be determined.

Forward model The forward kinematics are based on the Denavit-Hartenberg transformation matrices (Table 7.1). Using this model, the end-effector position relative to the joint positions θ_i is calculated. This is then used to compute the Jacobian needed for the inverse kinematics.

Inverse model For controlling the robot, the cartesian trajectory produced by the DMP is transformed into joint positions by the inverse kinematics

$$\dot{\theta} = J^{+}\dot{x} \tag{7.17}$$

with J^{+} being the pseudo inverse of the Jacobian J, i.e., the least-squares approximation to the real inverse, of the end-effector. In our case, only the pseudoinverse is applicable due to the redundant configuration of the KUKA robot. We make use of this redundancy and extend the equation 7.17 to add null-space constraints to prevent link collisions with obstacles without affecting the final trajectory [225].

$$\dot{\theta} = J^{+}\dot{x} + [J_0(I - J^{+}J)]^{+}(\dot{x_0} - J_0 J^{+}\dot{x}) \tag{7.18}$$

where x_0 is the point on the robot closest to the obstacle and J_0 is the corresponding Jacobian. This added term induces a movement $\dot{x_0}$ away from the obstacle.

7.3.5 Cost computation

Instead of trying to find the latent cost functions that might affect constructing a natural close proximity human-robot interaction, as explicitly provided in some recent studies for legible and predictable motion [7], we formulate a composite cost function (slightly different variations for each experiment (Sec. 7.4)) that contains the main components that is informative for such interaction.

Several cost functions are identified and mixed depending on the experimental setup. Here, we list all the costs used. and specify later which one of those are used when explaining the individual experiments in Sec. 7.4:

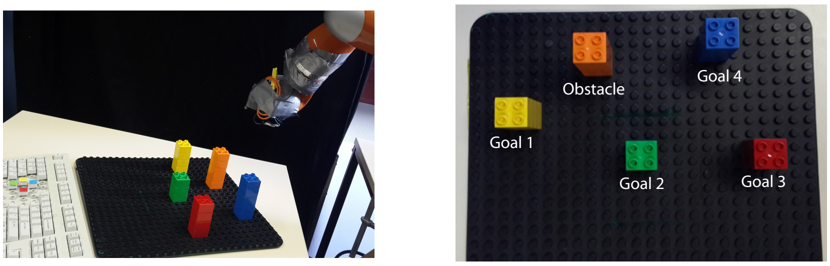
Experiment	λ_{ej}	λ_{θ}	λ_{pred}	λ_{task}	λ_{obs}	λ_{dur}	λ_{δ}
End effector control	1	2	4	20	2	-	3
Joint control + Transfer	1	1	10	2	10	1	-

Table 7.2: Weights for the cost components for each experiment.

- End-effector jerk J_{ej}: the sum of the third derivative of the end-effector position of the robot at each time step in the trajectory.
- Angular jerk J_{θ}: the sum of the third derivative of the angular positions of the controlled joints in the robot at each time step in the trajectory.
- Human time J_{pred}: the time taken by the human to make a prediction about the robot's target. It starts when the robot starts moving and ends when the human reaches one of the targets or presses a button.
- Accuracy J_{task}: whether the human prediction was correct, translating to 0 cost ($J_{task} = 0$), or if the prediction was wrong which results to a cost of 1 ($J_{task} = 1$).
- DMP accuracy J_{obs}: the difference between the direct output of the DMP and the actual trajectory after modification for obstacle avoidance. This is motivated for teaching the DMP the avoidance behavior.
- Human duration J_{dur}: the duration of the human movement between when the human starts moving and reaches the goal. It is a measurement of human's confidence in the robot's presence.
- The weighted distance between the trajectories, J_{δ}, which measures how distinct the trajectory to the targeted goal is in comparison to the trajectories to the other goals. This cost is calculated using the following equation:

$$J_{\delta} = \left(\sum_{g=1}^{G} \sum_{t=0}^{T} \frac{1}{t} \, \mathrm{d}(\boldsymbol{p}_t, \boldsymbol{q}_t) \right)^{-1} \tag{7.19}$$

where G is the number of the goals excluding the targeted goal, g is the other goal whose trajectory is compared to the targeted goal trajectory, t is the time step at which we calculate the distance, T is the total time of the trajectory, $\boldsymbol{p}_t$ is the position at t in the trajectory to the targeted goal, $\boldsymbol{q}_t$ is the position at t in the trajectory to the goal g and $\mathrm{d}(\boldsymbol{p}_t, \boldsymbol{q}_t)$ is a function that calculates the Euclidean distance between $\boldsymbol{p}_t$ and $\boldsymbol{q}_t$.



(a) Full view with Kuka

(b) Goals and obstacle position

Figure 7.3: The setup of the experiment.

7.4 Experiments

7.4.1 Preliminary experiments

Initial experiments have been conducted to investigate the effects of directly controlling the robot joint positions versus only the end effector position by the optimized control policy during human-robot interaction (HRI). First, the dynamic movement primitive (DMP) policy is used for end effector control of the robot where the avoidance term (potential field generated force) is directly applied on the DMP formulation. Note that, here there is only one static obstacle (Fig. 7.3). The joint position control is achieved by the IK-solver without constraints.

The experiment was conducted among six subjects 2 females, and 4 males, with an average age of 24 ± 6,. The experiment starts with a habituation phase in which the robot performs the initial trajectory to each one of the four goals without a specific order. This habituation phase allows the human subject to get prepared for the actual experiment. The initial trajectories are straight line movements in x, y, and z dimensions from the starting position to the goal, with the exception of $G2$ and $G3$ for which z dimension is pre-trained to avoid the static obstacle barely.

After the habituation phase the actual experiment starts. The experiment consists of 10 updates, and each update has 6 different trajectories for each goal. The robot chooses the goals randomly and performs a trajectory for each goal with the new mean updated parameters, as well as sampled basis function weights, $\boldsymbol{\omega}_k$.

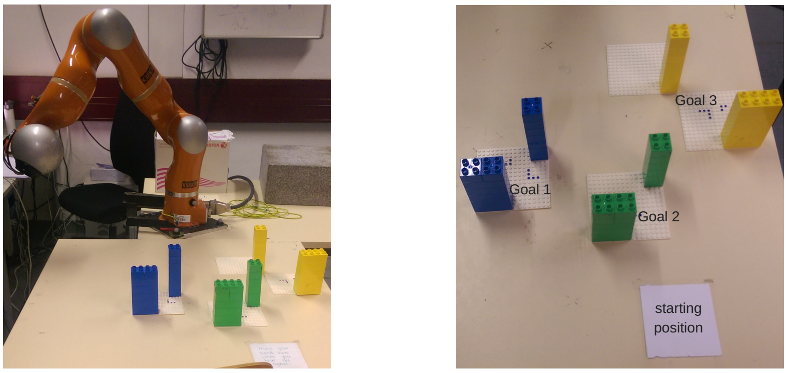
A composite cost function is constructed as

$$J_c = \lambda_{\text{ej}} J_{\text{ej}} + \lambda_\theta J_\theta + \lambda_{\text{pred}} J_{\text{pred}} + \lambda_{\text{task}} J_{\text{task}} + \lambda_{\text{obs}} J_{\text{obs}} + \lambda_\delta J_\delta. \tag{7.20}$$

Each term in the cost function is normalized from zero to one then multiplied by its scaling factor (Table 7.2), which are tuned by trial and error.

7.4.2 Transferability experiment

In this last experiment, the focus is on the transferability of a pre-learned policy to another task and subjects. The experimental setup consists of three goals, identified with different colors. Each goal consists of one thick block (robot's target) and one thin block (human's target) built with



(a) Full view with Kuka

(b) Goals and start position

Figure 7.4: The setup of the second experiment.

LEGO bricks (Fig. 7.4a and 7.4b). The starting position of the robot was chosen to give the robot maximum manipulability with respect to the setup, while the final robot position is given by the target locations, and only constrains the end-effector position. The trajectory from initial to final position is generated by the DMP for the end effector (eq. 7.8), and the inverse kinematics for the joint positions (eq. 7.18).

During the experiment, the robot chooses one of the three goals randomly and reaches it. During the movement of the robot the human has to predict the goal the robot is aiming for and move as fast as possible to the corresponding thinner goal. If the paths of the robot and the human intersect, the robot modifies its trajectory to avoid collision. The impedance controller of the robot further ensures the safety of the human at all times. There are 15 updates and each update consists of 3 samples for each goal.

To investigate the transferability of the learned robot policy to other tasks, we have two different experiments with the same setup. They just differ in the policy parameters.

- Experiment A is the control group. Here, the robot learns the policy "from scratch" during the experiment.
- In Experiment B, the policy is trained with a similar setup (the difference being the initial position of the robot), and the policy parameters are saved to be used for the same setup as Experiment A.

The cost function is similar to the preliminary experiment with joint control. The experiment was conducted on 13 human subjects with an average age of 23±7, 7 females and 6 males. 6 participants were involved in Experiment A, and another 6 in B and one participant interacted with the robot to train the policy for Experiment B. The cost values and the robot and human trajectories were saved and processed for each human subject for both experiments.

7.5 Results and Discussion

Here, the results from both experiments are provided.

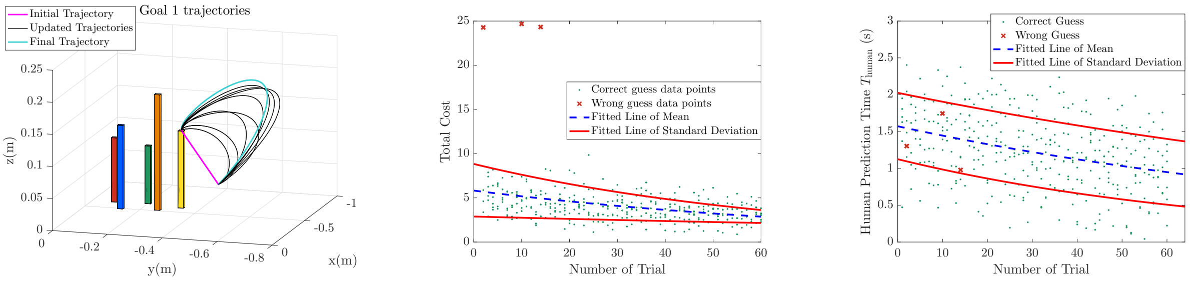


Figure 7.5: The trajectories to $G1$ for one of the representative human subjects (left), along with the total cost (center), and the human reaction time (right) for predicting $G1$ for all the human subjects.

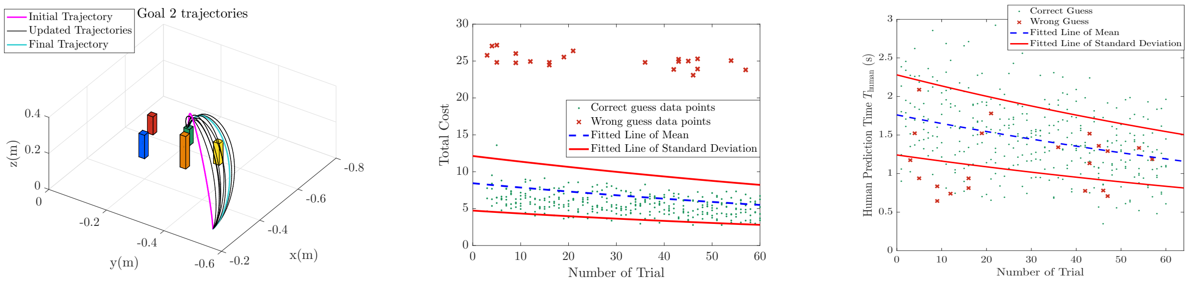


Figure 7.6: The trajectories to $G2$ for one of the representative human subjects (left), along with the total cost (center), and the human reaction time (right) for predicting $G2$ for all the human subjects.

7.5.1 Results for the preliminary experiment

The curvature of the trajectory away from the other goals and obstacle is distinctive for this goal (Fig. 7.5). $G1$ was the easiest goal for prediction given the fact it does not have any other goals or obstacles placed in front of it and it was the closest goal to the starting position of the robot. The human prediction time decreases during the trial over all six human subjects without any wrong predictions at the end. There are only 3 wrong guesses at the early updates from the 360 total $G1$ trajectory predictions by the six human subjects. The total costs also decreases and converges towards the last updates with a noticeable decrease in standard deviation between the six subjects.

Figure 7.6 shows the initial, updated and final trajectories towards $G2$ for one of the human subjects. The initial trajectory is a straight line in xy plane but with an elevation in z dimension to avoid colliding with the orange obstacle. The final trajectory gets only a slight curvature in the xy plane towards $G1$ and a high curvature in z dimension.This trajectory prevents the end effector to collide with the obstacle and also makes the trajectory to $G2$ distinguishable than the trajectory to $G1$ that has a much larger curvature in xy plane, and than the trajectory to $G3$ that has a curvature in xy plane in the opposite direction towards $G4$ as shown in Figure 7.7.

The human prediction time for the human subjects decrease towards the last updates. The amount of wrong guesses for $G2$ is bigger that $G1$ because the subjects would sometimes confuse the trajectory to $G2$ as a trajectory to $G3$ and occasionally to $G1$. Almost half of these wrong predictions is caused by an early prediction of the goal. The total costs of the trajectories to $G2$ show a convergence towards the last updates. Note that the costs with values more than 20 is due to a wrong

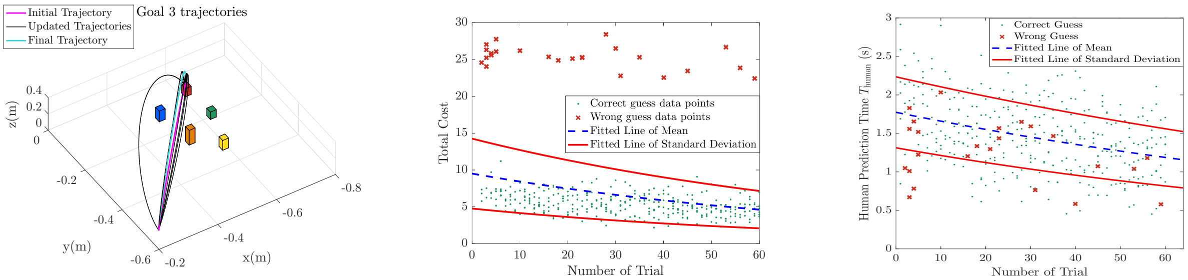


Figure 7.7: The trajectories to $G3$ for one of the representative human subjects (left), along with the total cost (center), and the human reaction time (right) for predicting $G3$ for all the human subjects.

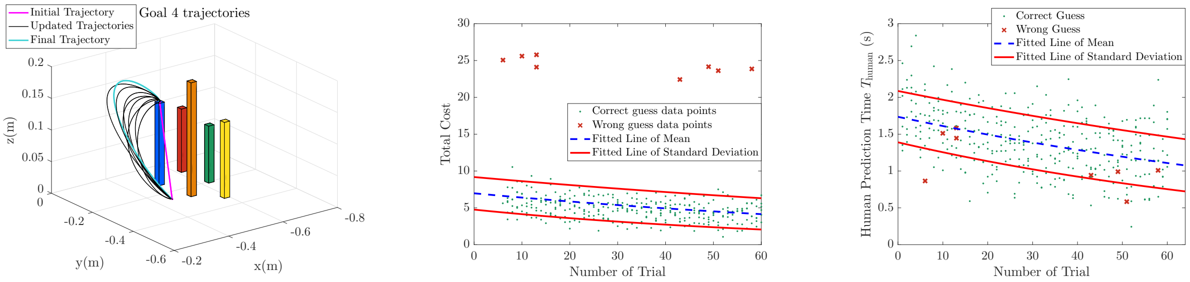


Figure 7.8: The trajectories to $G4$ for one of the representative human subjects (left), along with the total cost (center), and the human reaction time (right) for predicting $G4$ for all the human subjects.

prediction of the goal because the accuracy cost is multiplied by a scaling factor 20.

Overall, for the other two goals, $G3$ and $G4$, similar trends are observed (Fig. 7.7 and 7.8). We can notice that the final trajectory to a certain goal is not necessarily the one with the most exaggerated motion away from the other goals. This is due to the combination of the cost terms and their scaling factors. An exaggerated motion in the wrong direction might also lead to a longer human reaction time especially in the early updates, as the human subject might not be expecting such exaggerated motion so they wait until the end of the trajectory to make a prediction.

One interesting outcome is the final trajectories for each subject (Fig. 7.9). The policy improvement process leads to different final trajectories for each subject. Such difference is consistent for all the goals. This is an indication that our approach generates predictable trajectories that adapt to

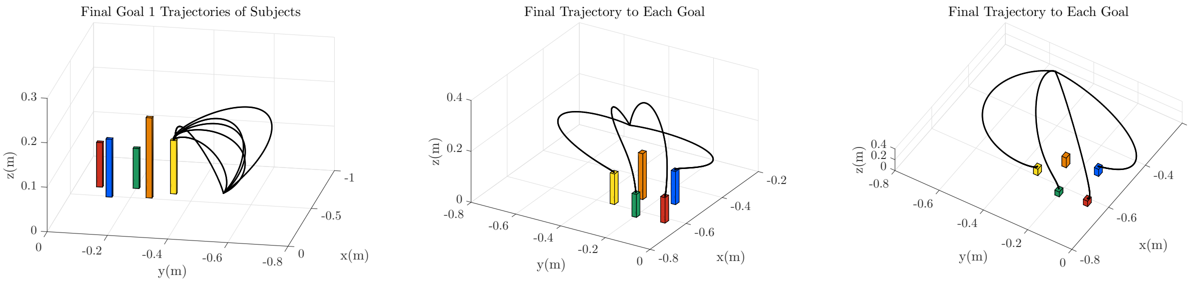


Figure 7.9: The final trajectories to $G1$ for all the subjects (left), along with the final trajectory per goal for a representative subject from two perspectives (center and right).

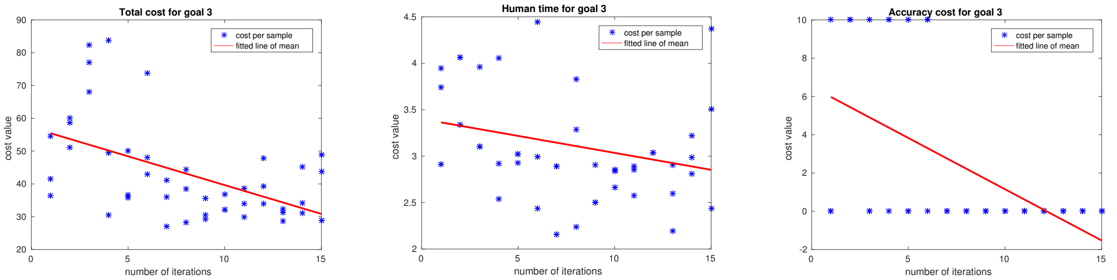


Figure 7.10: The total cost (left), human response time (center), and accuracy (right) over trials of a representative subject in Experiment A.

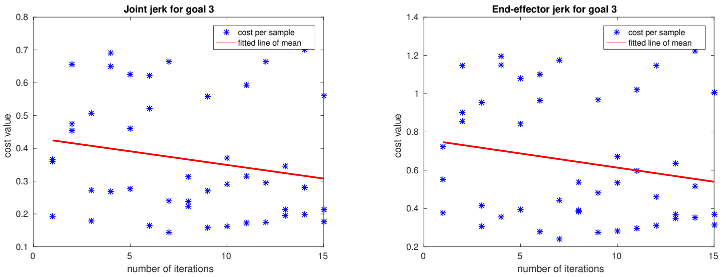


Figure 7.11: The joint jerk (left), and end effector jerk (right) of the robot during a trial in Experiment A.

the human preferences. Figure 7.9 also shows the final trajectory to each goal for the same subject. The trajectories are distinctive to allow lower prediction time for each goal.

7.5.2 Results for the transferability experiment

Experiment A First, we will present the results for the Experiment A. The robot learned its policy during the interaction starting with a simple initial controller following a straight line. The cost components of one of the representative subjects in this experiment show that both the human time and the accuracy cost are decreasing with each update, i.e. the subject is getting faster and more accurate over time (Fig. 7.10).

We noticed two trends in participants. Some of them wait a long time to be sure about their prediction and then move their hand, while some have very fast response times, but bad accuracy in the beginning. This is due to the personal preferences of the humans and might be related to their experience and confidence around robots. Even though the degree of improvement changes, in general the human response time and the accuracy cost decrease during this close proximity interaction.

The jerk costs (both for the end effector and the joints) are scattered throughout the trials (Fig. 7.11). Such high jerk profiles arise when the robot has to avoid collision with the human and does an exaggerated maneuver. However, on average the jerk costs still decrease over time.

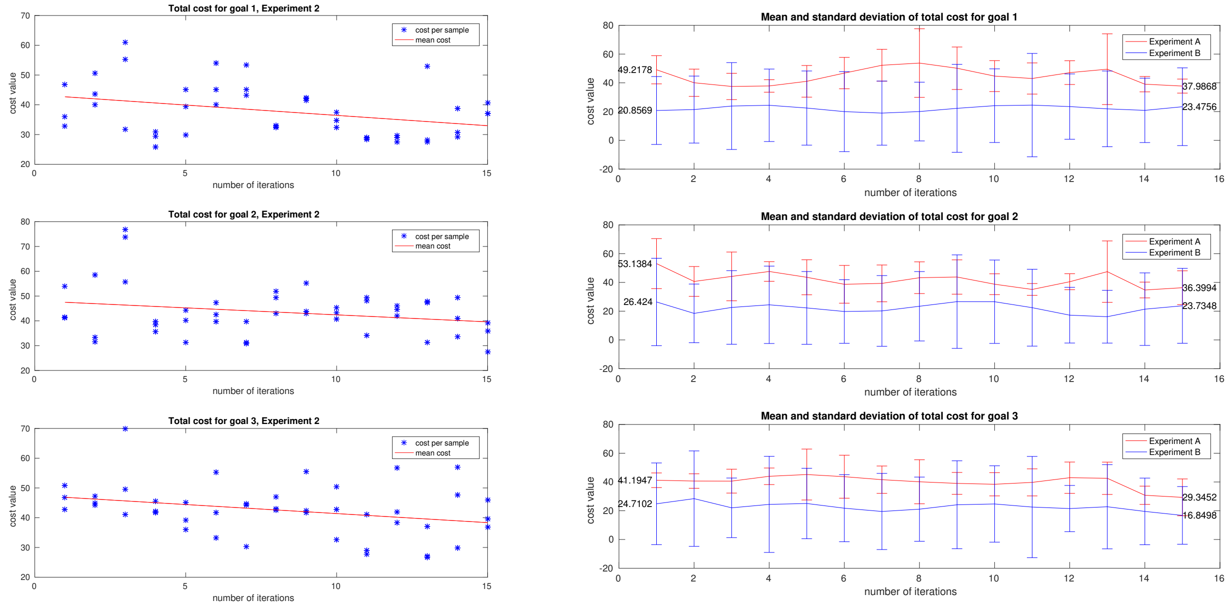


Figure 7.12: The total cost for the case where the robot is trained with only 1 subject for Experiment B (left), and the mean and standard deviation of the total cost for both Experiment A and B for all subjects per goal (right).

Experiment B In Experiment B, the subjects interacted with the robot that was trained with the policy of another subject for a different task. This participant did the experiment with the robot with a different initial position, and thus configuration. After the experiment was complete, the policy parameters were saved and then used for the same setup as Experiment A. In essence, the transferability of the policy to another task and for other subjects is investigated. First, we show the cost values for the one participant during the training of the policy which was used for other 6 subjects (Fig. 7.12 (left)). The cost values decreased for all the goals. When we used the policy for the different task, however, the cost stayed relatively the same (Fig. 7.12 (right, blue)). This might be that the pre-learned policy generates discernible trajectories at the start of the trials. Another reason might be due to the fact that the covariance of the exploration was already small, so the pre-learned policy was stuck in a local minimum and did not improve further.

Comparison between experiments A and B The analysis on mean and standard deviation of the cost values shows that Experiment A has higher cost values than Experiment B (Fig. 7.12 (right)). In general, the overall cost for Experiment A decreases over time. However, it does not reach the values observed in Experiment B. Because of the trained policy in Experiment B, participants had shorter response times from the very first iteration (Fig. 7.13).

In all the experiments the position of the blocks forced the participants to come in close proximity of the robot. The obstacle avoidance algorithm was successful in avoiding collision in these scenarios.

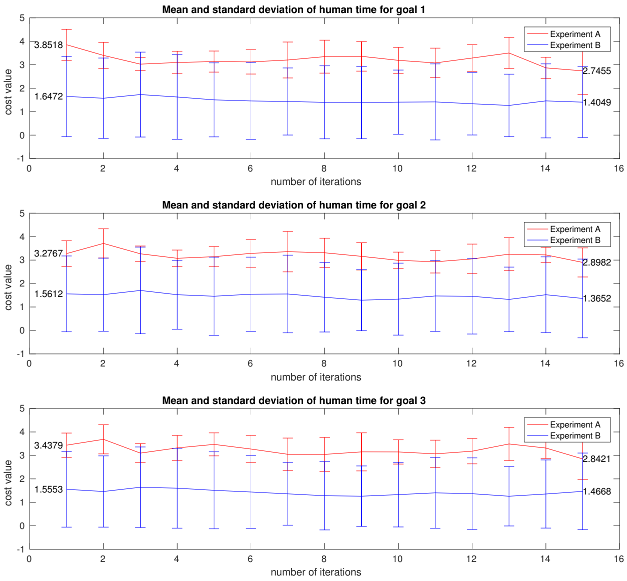


Figure 7.13: The mean and standard deviation of the human response time for all subjects, (red - Experiment A, and blue - Experiment B).

7.5.3 Conclusion

In this work, a framework is developed to generate predictable robot motion that can adapt to human preferences and can avoid dynamic obstacles, which in our case is the human hand during interaction). The experiments that were conducted show that robots are able to adapt their behavior to human preferences. They can learn to become more predictable while still giving humans the freedom to move safely in the same work space. The humans became faster and more confident in their predictions. Task transferability between subjects is tested. In our experiment, the learned policy produced better results in the new task than the control group without a pre-learned policy. This confirms our hypothesis that the policy learned by this framework is indeed transferable to other tasks and also to other humans.

Having said that, there are some limitations for our work. Due to the relatively small size of our study, a definitive claim on the applicability of this framework with other subjects from different backgrounds or with different tasks cannot be made. Another missing investigation is how much of these observed improvements are due to human learning and adaptation to the robot movement, or robot changing its policy, and thus the movement behavior. Experiments with a few days break in between trials per subject would help reducing the learning effects, and may provide a better perspective to understand the impact of robot policy change.

8 Conclusions

In this thesis, I presented my interdisciplinary approach to achieve safe, natural, and effective close proximity human-robot interaction (HRI). All of the algorithms and frameworks in this thesis focus on human-in-the-loop robot control and learning. My dissertation addresses two main research questions: First, how can we model human movement behavior, both in isolation and also during interaction? Second, how can human level interaction capability be achieved for robotic agents to provide natural, safe, and effective HRI? The dissertation proposes an interdisciplinary approach to enable such close proximity HRI by combining (*i*) human sensorimotor control models, (*ii*) human-human interaction (HHI) dynamics, and (*iii*) novel learning and optimization algorithms for robot interaction behavior generation.

In my thesis, I focused on comprehensively covering different aspects of close-proximity HRI. One such aspect is inevitably the human factors. As different scientific communities tackle those problems in isolation (i.e., focus on cognitive and motor skills in neuroscience vs. fast prediction of human motions in visual computing), there is a gap between accurate sensorimotor control models and matching human level effectiveness for motion generation and control. In Chapter 2, a hierarchical control architecture is proposed for explaining human arm reaching control by the central nervous system (CNS). The inverse optimal control (IOC) results indicate that humans rely on a composite of controllers, each of which can be described by an optimal controller. More notably, there is a trade-off between kinematics and dynamics related controllers, and humans combine them effortlessly according to the task. These findings suggest a process of sensorimotor learning of multiple forward and inverse internal models within the CNS simultaneously and then controlling them in a hierarchical manner effectively. Next, in Chapter 3, I showed how this composite controller can be approximately represented by Gaussian process regression (GPR) models, and combined with data-driven approaches for modeling and predicting human reaching motions online. Here, the focus was both to represent person specific control models, and also to effectively use them for online computations. The combination of model-based and data-driven approaches are presented that complement each other to fulfill this representational accuracy and computational efficiency requirements.

The symbiosis of model-based and data-driven approaches was a running theme of this thesis. The connection between Chapters 4 and 5 is another example of this necessary co-existence. In Chapter 4, I presented a systematic approach to investigate close proximity human-human interaction scenarios by using both human kinematic models, and also an extensive set of recorded dyadic interaction cases. This analysis resulted in an ontology that provides a unified and generalized graph-based representation that is used to model, classify, and predict close proximity human interaction behaviors. Thanks to our understanding of composite controllers for human reaching motions, similar combined models were assumed and learned by inverse reinforcement learning (IRL) to understand the strategies used by humans in such situations. These learned policies enable generating the appropriate robot trajectories similar to humans' during interaction. Such a thorough coverage and systematic analysis of HHI cases serve as a base and reference for future studies to propose and benchmark new learning algorithms, and to evaluate close proximity HRI behaviors for different tasks. The learning framework developed (Chapter 5) uses the same dataset of HHI scenarios, and learns policies that imitates human interaction behavior control. This formulation is similar to end-to-end learning approaches by only relying the distance features that can be obtained by visual and proximity sensors. I demonstrated that the recurrent neural network (RNN) based policy can efficiently be used by a robot to interact with human partners in close proximity. The policy is also generalized in the sense that it allows reaching any target in 3D space, and even when the target is dynamic, for which it was not explicitly trained. Different levels of behavioral cloning is achieved depending on the network complexity. Similar to my proposed hierarchical control structure for human motor control, an ensemble of policies can be combined to model complex behaviors for autonomous agents, and to the best of my knowledge, the demonstrated policies are the first neural network controllers that can operate on real robotic arms in close proximity to human partners without additional avoidance planning and control components.

The simulation and real world evaluations in Chapters 5, 6, and 7 show that the proposed models and algorithms enable robots to work with humans. Both the imitation learning and stochastic optimization methods are able to provide safe close proximity HRI that has been missing in prior studies, and the policy improvement method allows adapting to human movements and preferences, which can be utilized for longterm learning architectures. These two approaches verify the significance of human-in-the-center planning, and the online policy improvement approach offers a long-term learning framework for personalized assistance. Both methods also rely on model-based and data-driven formulations, similar to the prior studies presented in the thesis. In essence, human involvement necessitates constructing solutions that exploit best of both worlds in terms of technical approaches.

In the next section, I will discuss some high level limitations of the proposed methods and future work that could address them, as well as broader challenges in the areas of safe learning and constrained policy optimization. I will also discuss the possible future directions that can be followed in interaction and motor skill learning.

8.1 Challenges and Limitations

Model-based formulations is a common theme in all subcomponents of my thesis research, and the human kinematics and especially the dynamics models are also the cause of an underlying limitation of such approaches. Due to the computational complexity of solving optimal control problems (OCPs), and much harder inverse optimal control (IOC) formulations, only a simplified system model has been feasible in my work. Even though general behavioral trends of human movement can still be analyzed with such simplified models, they might also obscure subtle but important motor control properties. In addition, sensory feedback for movement control is missing in the current form of the proposed (I) OCP formulation, which is known to be critical for any human motor control task.

Another crucial challenge in all my work is the availability of human movement data, whether it is free-space arm motions or close proximity dyadic interaction. The data provided with this work will help further research in these areas. However, for a more sustainable solution, simulation based learning approaches is a promising direction for future work. Using the developed methods to simulate both human movements and robot interaction control, a reinforcement learning framework can be applied for acquiring a broader and richer range of interaction skills.

If we look at the research trends in related fields, the focus of robotics and control problems has been on robust and precise robotic movement. The human-robot interaction (HRI) can surely benefit from those formalisms, as some novel approaches presented in this thesis has accomplished. On the other hand, social robotics research, in general, focuses on subjective human evaluations of robotic performances. These studies also guide the critical directions that theoretical and algorithmic investigation in robotics should explore further. Nevertheless, one key prerequisite of natural and safe close proximity collaboration, which has been neglected at large in both fields, is understanding the control and decision making processes of humans. I believe, only after positioning the human factors at the center, autonomous agents can reliably learn and interact with human partners.

Having said that, I think the current robotic hardware is still not suitable for fast and safe close proximity HRI due to their operational space, manipulability constraints, lack of sensors, and bulky structure. In addition to constructing policies controlling the end effector or joint position, additional layers of controllers have to be implemented due to safety and real-time control frequency requirements. Naturally, both passive and active compliant behavior are expected for safe and natural HRI, which series-parallel elastic actuators might provide in the future. On the other hand, virtual and augmented reality setups have a great potential to simplify and expedite the process of conducting experiments and testing new methods efficiently to drive the research in this field faster.

Another key challenge is efficiency of numerical computations involved in controlling robotic agents in real world scenarios when working near humans. Safe interactions can be provided by model- and optimization-based approaches, however, performance and efficiency of numerical methods used to solve such problems have mostly been an afterthought. Fast trajectory computation methods [226], [227] as well as fast avoidance approaches exist [29], however, solely constructing reactive controllers and motion generators limits the collaborative opportunities between human and robot teams. In effect, such efficiency is a necessary but not sufficient condition for any purposeful close proximity HRI. Collaborative agents need richer interaction skills, and adaptation capabilities, i.e., they should not just *react* but also deliberately *respond* to their partners' behaviors and the dynamic environment.

Last but not least, from artificial intelligence (AI) research perspective, there is a lack of focus on joint manipulation and close interaction skills for autonomous systems that can work with humans. Although significant improvements achieved for continuous control tasks with deep reinforcement learning (DRL) techniques, thanks to the newly developed algorithms and efficient use of neural network based architectures, the focus has been purely on single agent skills [228]. Some recent studies have started to look into safe learning methods, but they are still towards learning low dimensional dynamic system models or agent skills [147], [229], [230]. The learning from demonstration (LfD) and programming by demonstration (PbD) approaches are effective for skill learning, and recently used for close interaction cases [231]. They tend to work within the demonstrated task space well, as in our own work [241], however, their generalization capability and safety issues still remain as challenges and their limitations.

8.2 Future Directions and Applications

Sensorimotor control The first contribution of my dissertation research is the identification of multiple internal models (MIMs) by an inverse optimal control (IOC) formulation to explain human arm reaching control. In addition, a trade-off between kinematics and dynamics related control models is identified w.r.t. the task parameters, and a metric is proposed to define a hierarchical control structure within the central nervous system (CNS) [235]. This formulation, together with similar metrics, enables investigation of a wide range of motor tasks to understand and model human sensorimotor control better in future studies. I think human motor control flexibility emerges from a hierarchical, yet easily deformable, functional mappings from sensory information to motor commands. In that regard, on a bigger scale, building new bridges to fill the the gap between neuroscientific findings, e.g. plasticity of neurons in brains [232], and learning algorithms for skill acquisition, e.g. hierarchical reinforcement learning (RL) [233], or decision making [234] is a challenging but very crucial research direction.

Going beyond imitating human-human interaction for HRI The ontology proposed in this thesis is a first step towards a systematic analysis of close proximity human-human interaction (HHI). The results show that the ontology can be used to classify, and predict human motions accurately, and the policies learned from our extensive user studies demonstrate the feasibility of generating human-like interaction behavior control by imitating humans. Nevertheless, imitation alone limits the generalization capabilities and cannot provide safety guarantees. In order to improve the learning process of autonomous systems, control theoretic reachability or invariance set analysis can be integrated into the policy optimization procedure. This integration will enable a safe learning procedure by relying on the system dynamics and checking the feasibility of each action before considering it in policy evaluation phase. However, a simulation environment, where the agent can learn through exploration, is still necessary. In such an environment, human-like movements can be simulated for one agent by the imitated human control policies learned from demonstrations, while a second agent learns safe and natural dyadic behavior by a constrained policy optimizer. This is also critical for any autonomous system that is expected to act near humans in our daily lives. The agent will be able to take safe actions and still learn through interaction and observation for long-term learning, adaptation, and personalization.

A comprehensive human-robot interaction architecture Currently, the research studies involving human-robot interaction (HRI) covers a small subset of tasks and interaction scenarios. Following a top-down approach by constructing a comprehensive architecture, comprising three main building blocks, inspired by the cognitive perception-action-learning (PAL) loop hypothesis for human sensorimotor control, would enable a generalized formulation for interaction situations.

In order to improve the awareness capabilities of the agent, the *perception* module considers three main properties, which are multimodal sensory information from the environment, e.g. including objects being jointly manipulated, the motion behavior characteristics, and spatio-temporal features related to person- and situation-dependent social signals. For proactive *action* execution, novel social movement primitives (SMPs), consisting of a control law, e.g. an optimal control problem with social constraints, can be formulated to generate socially acceptable motions. As each motion primitive executed by the robot results in a different reaction of the human, the corresponding action-perception dependency needs to be accounted for to build a behavioral memory for the autonomous agent. In a next step, SMPs are augmented by the perception signals to form the associative social skill memories (AS^{2}Ms) of the robot.

An AS^{2}M still is an individual behavior model with limited generalization. As a last step, a suitable generalized behavior model has to be constructed from the repertoire of behaviors that exist as SMPs along with AS^{2}Ms. This requires a supervised deep sensorimotor learning framework due to the high-dimensionality of the sensory data, in which AS^{2}Ms along with the observations from perception module guide the learning of high-level controllers.

In sum, these multidisciplinary approaches would endow robotic agents with generalized interaction skills. In essence, they will be able to learn and control complex motor skills to effectively interact with humans and in dynamic environments autonomously.

Bibliography

[1] E. Todorov, W. Li, and X. Pan, “From task parameters to motor synergies: A hierarchical framework for approximately optimal control of redundant manipulators”, *J. Field Robotics*, vol. 22, no. 11, pp. 691–710, 2005. DOI: 10.1002/rob.20093. [Online]. Available: https://doi.org/10.1002/rob.20093. 1, 4, 43

[2] W. L. Nelson, “Physical principles for economies of skilled movements”, *Biological cybernetics*, vol. 46, no. 2, pp. 135–147, 1983. 1, 16, 22

[3] T. Flash and N. Hogan, “The coordination of arm movements: an experimentally confirmed mathematical model”, *J. Neurosci.*, vol. 5, no. 7, pp. 1688–1703, 1985. 2, 4

[4] M. Kawato, Y. Maeda, Y. Uno, and R. Suzuki, “Trajectory formation of arm movement by cascade neural network model based on minimum torque-change criterion”, *Biological cybernetics*, vol. 62, no. 4, pp. 275–288, 1990. 2, 121

[5] E. Todorov and M. I. Jordan, “Optimal feedback control as a theory of motor coordination”, *Nature neuroscience*, vol. 5, no. 11, p. 1226, 2002. 2, 6, 16, 43, 47,

[6] P. A. Lasota, T. Fong, and J. A. Shah, “A survey of methods for safe human-robot interaction”, *Foundations and Trends in Robotics*, vol. 5, no. 4, pp. 261–349, 2017. DOI: 10.1561/2300000052. [Online]. Available: https://doi.org/10.1561/2300000052. 2, 8

[7] A. D. Dragan, K. C. T. Lee, and S. S. Srinivasa, “Legibility and predictability of robot motion”, in *Int. Conf. on HRI, Tokyo, Japan*, IEEE/ACM, 2013, pp. 301–308. 2, 8, 116, 130, 1

[8] F. Stulp, J. Grizou, B. Busch, and M. Lopes, “Facilitating intention prediction for humans by optimizing robot motions”, in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, IEEE, 2015, pp. 1249–1255. 2, 8, 116, 131, 1

[9] S. Albrecht, K. Ramirez-Amaro, F. Ruiz-Ugalde, D. Weikersdorfer, M. Leibold, M. Ulbrich, and M. Beetz, “Imitating human reaching motions using physically inspired optimization principles”, in *Humanoids*, IEEE, 2011, pp. 602–607. 2, 49, 68, 121

[10] K. D. Piatkevich, E. E. Jung, C. Straub, C. Linghu, D. Park, H.-J. Suk, D. R. Hochbaum, D. Goodwin, E. Pnevmatikakis, N. Pak, *et al.*, “A robotic multidimensional directed evolution approach applied to fluorescent voltage reporters”, *Nature chemical biology*, p. 1, 2018. 3

[11] Y. Uno, M. Kawato, and R. Suzuki, “Formation and control of optimal trajectory in human multijoint arm movement”, *Biological cybernetics*, vol. 61, no. 2, pp. 89–101, 1989. 4, 16, 21, 22, 39

[12] D. Liu and E. Todorov, "Evidence for the flexible sensorimotor strategies predicted by optimal feedback control", *The Journal of Neuroscience*, vol. 27, no. 35, pp. 9354–9368, 2007. 4, 41, 43, 50

[13] N. Hogan, "The mechanics of multi-joint posture and movement control", *Biological cybernetics*, vol. 52, no. 5, pp. 315–331, 1985. 4

[14] R. L. Sainburg, "Should the equilibrium point hypothesis (EPH) be considered a scientific theory?", *Motor control*, vol. 19, no. 2, pp. 142–148, 2015. 4

[15] M. Manto, J. M. Bower, A. B. Conforto, J. M. Delgado-García, S. N. F. da Guarda, M. Gerwig, C. Habas, N. Hagura, R. B. Ivry, P. Mariën, *et al.*, "Consensus paper: Roles of the cerebellum in motor control—the diversity of ideas on cerebellar involvement in movement", *The Cerebellum*, vol. 11, no. 2, pp. 457–487, 2012. 4, 46

[16] R. Shadmehr, M. A. Smith, and J. W. Krakauer, "Error correction, sensory prediction, and adaptation in motor control", *Annual review of neuroscience*, vol. 33, pp. 89–108, 2010. 4, 16, 39, 46

[17] D. M. Wolpert, R. C. Miall, and M. Kawato, "Internal models in the cerebellum", *Trends Cogn. Sci.*, vol. 2, no. 9, pp. 338–347, 1998. 4, 5, 15, 17, 40–42

[18] M. Haruno, D. M. Wolpert, and M. Kawato, "MOSAIC model for sensorimotor control and learning", *Neural Comput.*, vol. 13, pp. 2201–2220, 2001. 5, 16, 39–41

[19] Y. Demiris and B. Khadhouri, "Hierarchical attentive multiple models for execution and recognition of actions", *Robotics and Autonomous Systems*, vol. 54, no. 5, pp. 361–369, 2006. DOI: 10.1016/j.robot.2006.02.003. [Online]. Available: https://doi.org/10.1016/j.robot.2006.02.003. 5

[20] N. Sugimoto, M. Haruno, K. Doya, and M. Kawato, "MOSAIC for multiple-reward environments", *Neural computation*, vol. 24, no. 3, pp. 577–606, 2012. 5, 40

[21] K. O. Ellefsen and J. Tørresen, "Evolving neural networks with multiple internal models", in *Proceedings of the Fourteenth European Conference Artificial Life, ECAL 2017, Lyon, France, September 4-8, 2017*, MIT Press, 2017, pp. 138–145. DOI: 10.7551/ecal_a_025. [Online]. Available: https://doi.org/10.7551/ecal_a_025. 5

[22] P. Evrard and A. Kheddar, "Homotopy-based controller for physical human-robot interaction", in *RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication*, Sep. 2009, pp. 1–6. DOI: 10.1109/ROMAN.2009.5326065. 6

[23] N. Jarrassé, V. Sanguineti, and E. Burdet, "Slaves no longer: Review on role assignment for human–robot joint motor action", *Adaptive Behavior*, vol. 22, no. 1, pp. 70–82, 2014. DOI: 10.1177/1059712313481044. eprint: https://doi.org/10.1177/1059712313481044. [Online]. Available: https://doi.org/10.1177/1059712313481044. 6

[24] B. Liu, T. Chu, L. Wang, and G. Xie, "Controllability of a leader-follower dynamic network with switching topology", *IEEE Trans. Automat. Contr.*, vol. 53, no. 4, pp. 1009–1013, 2008. DOI: 10.1109/TAC.2008.919548. [Online]. Available: https://doi.org/10.1109/TAC.2008.919548. 6

[25] H. S. Koppula and A. Saxena, "Physically grounded spatio-temporal object affordances", in *European Conference on Computer Vision*, Springer, 2014, pp. 831–847. 7, 77, 78

[26] J. Mainprice, R. Hayne, and D. Berenson, “Predicting human reaching motion in collaborative tasks using inverse optimal control and iterative re-planning”, in *International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 885–892. 7, 76

[27] J. Mainprice, R. Hayne, and D. Berenson, “Goal set inverse optimal control and iterative replanning for predicting human reaching motions in shared workspaces”, *IEEE Trans. Robotics*, vol. 32, no. 4, pp. 897–908, 2016. DOI: 10.1109/TRO.2016.2581216. [Online]. Available: https://doi.org/10.1109/TRO.2016.2581216. 7, 8, 42, 50, 74,

[28] M. Corbetta and G. L. Shulman, “Control of goal-directed and stimulus-driven attention in the brain”, *Nature reviews neuroscience*, vol. 3, no. 3, pp. 201–215, 2002. 7, 74, 77, 79

[29] F. Flacco, T. Kroeger, A. D. Luca, and O. Khatib, “A depth space approach for evaluating distance to objects-with application to human-robot collision avoidance”, *J. Intell. Rob. Syst.*, vol. 80, pp. 7–22, 2015. 8, 116, 147

[30] D. Carton, W. Olszowy, and D. Wollherr, “Measuring the effectiveness of readability for mobile robot locomotion”, *Int. J. of Social Robotics*, vol. 8, no. 5, pp. 721–741, Nov. 2016. 8, 116

[31] D. M. Wolpert, Z. Ghahramani, and M. I. Jordan, “An internal model for sensorimotor integration”, *Science*, vol. 269, no. 5232, p. 1880, 1995. 15

[32] T. Flash and N. Hogan, “The coordination of arm movements: An experimentally confirmed mathematical model”, *The journal of Neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985. 16, 22, 39, 41, 4

[33] C. M. Harris and D. M. Wolpert, “Signal-dependent noise determines motor planning”, *Nature*, vol. 394, no. 6695, pp. 780–784, 1998. 16, 39, 43, 47, 4

[34] M. Sağlam, N. Lehnen, and S. Glasauer, “Optimal control of natural eye-head movements minimizes the impact of noise”, *J. Neurosci.*, vol. 31, no. 45, pp. 16 185–16 193, 2011. 16

[35] M. Sağlam, S. Glasauer, and N. Lehnen, “Vestibular and cerebellar contribution to gaze optimality”, *Brain*, vol. 137, no. 4, pp. 1080–1094, 2014. DOI: 10.1093/brain/awu006. 16

[36] C. D. Takahashi, R. A. Scheidt, and D. J. Reinkensmeyer, “Impedance control and internal model formation when reaching in a randomly varying dynamical environment”, *J. Neurophysiol.*, vol. 86, no. 2, pp. 1047–1051, 2001. 16

[37] P. N. Sabes, “The planning and control of reaching movements”, *Curr. Opin. Neurol.*, vol. 10, no. 6, pp. 740–746, 2000. 16

[38] I. L. Kurtzer, J. A. Pruszynski, and S. H. Scott, “Long-latency reflexes of the human arm reflect an internal model of limb dynamics”, *Curr. Biol.*, vol. 18, no. 6, pp. 449–453, 2008. 16

[39] J. W. Krakauer, M.-F. Ghilardi, and C. Ghez, “Independent learning of internal models for kinematic and dynamic control of reaching”, *Nat. Neurosci.*, vol. 2, no. 11, pp. 1026–1031, 1999. 16

[40] H. Imamizu, T. Kuroda, S. Miyauchi, T. Yoshioka, and M. Kawato, “Modular organization of internal models of tools in the human cerebellum”, *Proc. Natl. Acad. Sci.*, vol. 100, no. 9, pp. 5461–5466, 2003. 16

[41] M. A. Smith and R. Shadmehr, “Intact ability to learn internal models of arm dynamics in Huntington’s disease but not cerebellar degeneration”, *J. Neurophysiol.*, vol. 93, no. 5, pp. 2809–2821, 2005. 16

[42] J. Doyon and H. Benali, “Reorganization and plasticity in the adult brain during learning of motor skills”, *Curr. Opin. Neurol.*, vol. 15, no. 2, pp. 161–167, 2005. 16

[43] A. J. Bastian, “Learning to predict the future: The cerebellum adapts feedforward movement control”, *Curr. Opin. Neurol.*, vol. 16, no. 6, pp. 645–649, 2006. 16

[44] E. Todorov, “Optimality principles in sensorimotor control”, *Nature neuroscience*, vol. 7, no. 9, pp. 907–915, 2004. 16, 17, 39–41, 43, 47

[45] J. R. Flanagan, E. Nakano, H. Imamizu, R. Osu, T. Yoshioka, and M. Kawato, “Composition and decomposition of internal models in motor learning under altered kinematic and dynamic environments”, *J. Neurosci.*, vol. 19, RC34–1, 1999. 16, 40

[46] A. Karniel and F. A. Mussa-Ivaldi, “Does the motor control system use multiple models and context switching to cope with a variable environment?”, *Exp. Brain Res.*, vol. 143, no. 4, pp. 520–524, 2002. 16, 40

[47] J. Kluzik, J. Diedrichsen, R. Shadmehr, and A. J. Bastian, “Reach adaptation: What determines whether we learn an internal model of the tool or adapt the model of our arm?”, *J. Neurophysiol.*, vol. 100, no. 3, pp. 1455–1464, 2008, ISSN: 0022-3077. DOI: 10.1152/jn.90334.2008. 16, 40

[48] P. R. Davidson and D. M. Wolpert, “Internal models underlying grasp can be additively combined”, *Exp. Brain Res.*, vol. 155, no. 3, pp. 334–340, 2004. 16, 39, 40

[49] F. C. Anderson and M. G. Pandy, “Dynamic optimization of human walking”, *J. Biomech. Eng.*, vol. 123, no. 5, pp. 381–390, 2001. 16

[50] C. K. Chow and D. H. Jacobson, “Studies of human locomotion via optimal programming”, *Math. Biosci.*, vol. 10, no. 3, pp. 239–306, 1971. 16

[51] H. Hatze and J. D. Buys, “Energy-optimal controls in the mammalian neuromuscular system”, *Biol. Cybern.*, vol. 27, no. 1, pp. 9–20, 1977. 16

[52] E. Burdet and T. E. Milner, “Quantization of human motions and learning of accurate movements”, *Biol. Cybern.*, vol. 78, no. 4, pp. 307–318, 1998. 16

[53] R. M. Alexander, “A minimum energy cost hypothesis for human arm trajectories”, *Biol. Cybern.*, vol. 76, no. 2, pp. 97–105, 1997. 16, 121

[54] J. L. Emken, R. Benitez, A. Sideris, J. E. Bobrow, and D. J. Reinkensmeyer, “Motor adaptation as a greedy optimization of error and effort”, *J. Neurophysiol.*, vol. 97, no. 6, pp. 3997–4006, 2007. 16

[55] D. W. Franklin, E. Burdet, K. P. Tee, R. Osu, C.-M. Chew, T. E. Milner, and M. Kawato, “CNS learns stable, accurate, and efficient movements using a simple algorithm”, *J. Neurosci.*, vol. 28, no. 44, pp. 11 165–11 173, 2008. 16

[56] I. O’Sullivan, E. Burdet, and J. Diedrichsen, “Dissociating variability and effort as determinants of coordination”, *PLOS Comput. Biol.*, vol. 5, no. 4, e1000345, 2009. 16

[57] K. Mombaur, A. Truong, and J.-P. Laumond, “From human to humanoid locomotion—an inverse optimal control approach”, *Auton. Robots*, vol. 28, no. 3, pp. 369–383, Apr. 2010, ISSN: 0929-5593. DOI: 10.1007/s10514-009-9170-7. 16, 25, 40, 41, 50, 68

[58] B. Berret, E. Chiovetto, F. Nori, and T. Pozzo, “Evidence for composite cost functions in arm movement planning: An inverse optimal control approach”, *PLOS Comput. Biol.*, vol. 7, no. 10, J. Diedrichsen, Ed., e1002183, Oct. 2011, ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1002183. 16, 20, 40, 41, 4

[59] S. Albrecht, M. Ulbrich, and M. Leibold, “A bilevel optimization approach to obtain optimal cost functions for human arm movements”, *Numer. Algebra Contr. Optim.*, vol. 2, no. 1, pp. 105–127, Mar. 2012, ISSN: 2155-3289. DOI: 10.3934/naco.2012.2.105. 16, 25, 37, 42, 4

[60] K. Friston, “What is optimal about motor control?”, *Neuron*, vol. 72, no. 3, pp. 488–498, 2011. 17, 39, 40

[61] E. Bizzi, F. A. Mussa-Ivaldi, and S. F. Giszter, “Computations underlying the execution of movement: a biological perspective.”, *Science*, vol. 253, no. 5017, pp. 287–91, 1991, ISSN: 0036-8075. DOI: 10.1126/science.1857964. 17

[62] F. A. Mussa-Ivaldi, S. F. Gisztert, and E. Bizzi, “Linear combinations of primitives in vertebrate motor control”, *Neurobiology*, vol. 91, pp. 7534–7538, 1994. 17

[63] C. B. Hart and S. F. Giszter, “A neural basis for motor primitives in the spinal cord.”, *J. Neurosci.*, vol. 30, no. 4, pp. 1322–1336, 2010, ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.5894-08.2010. 17

[64] F. J. Valero-Cuevas, H. Hoffmann, M. U. Kurse, J. J. Kutch, and E. A. Theodorou, “Computational models for neuromuscular function”, *IEEE reviews in biomedical engineering*, vol. 2, pp. 110–135, 2009. 18

[65] S. Albrecht, “Modeling and numerical solution of inverse optimal control problems for the analysis of human motions”, PhD thesis, University Library of TU Munich, 2013. 18, 40, 41

[66] D. A. Levinson and T. R. Kane, “AUTOLEV—a new approach to multibody dynamics”, in *Multibody systems handbook*, Springer, 1990, pp. 81–102. 18

[67] M. Diehl, *Numerical Optimal Control (DRAFT)*. Leuven, Belgium, 2016. 21, 22, 53

[68] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995. 21, 49

[69] B. Houska, H. J. Ferreau, and M. Diehl, “Acado toolkit—an open-source framework for automatic control and dynamic optimization”, *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, 2011. 22

[70] S. Ben-Itzhak and A. Karniel, “Minimum acceleration criterion with constraints implies bang-bang control as an underlying principle for optimal trajectories of arm reaching movements”, *Neural Computation*, vol. 20, no. 3, pp. 779–812, 2008. 22

[71] Y. Wada, Y. Kaneko, E. Nakano, R. Osu, and M. Kawato, “Quantitative examinations for multi joint arm trajectory planning—using a robust calculation algorithm of the minimum commanded torque change trajectory”, *Neural Networks*, vol. 14, no. 4, pp. 381–393, 2001. 22, 41, 51

[72] J. Nishii and T. Murakami, “Energetic optimality of arm trajectory”, in *Proc of Int Conf on Biomechanics of Man*, 2002, pp. 30–33. 22

[73] B. Berret, C. Darlot, F. Jean, T. Pozzo, C. Papaxanthis, and J. P. Gauthier, “The inactivation principle: Mathematical solutions minimizing the absolute work and biological implications for the planning of arm movements”, *PLoS Comput Biol*, vol. 4, no. 10, e1000194, 2008. 22, 41

[74] A. Biess, D. G. Liebermann, and T. Flash, “A computational model for redundant human three-dimensional pointing movements: Integration of independent spatial and temporal motor plans simplifies movement dynamics”, *The Journal of Neuroscience*, vol. 27, no. 48, pp. 13 045–13 064, 2007. 22

[75] H. P. Nguyen and J. B. Dingwell, “Proximal versus distal control of two-joint planar reaching movements in the presence of neuromuscular noise”, *Journal of biomechanical engineering*, vol. 134, no. 6, p. 061 007, 2012. 25, 26

[76] T. K. Vintsyuk, “Speech discrimination by dynamic programming”, *Cybernetics and Systems Analysis*, vol. 4, no. 1, pp. 52–57, 1968. 25

[77] R. Eberhart and J. Kennedy, “A new optimizer using particle swarm theory”, in *Micro Machine and Human Science, 1995. MHS ’95., Proceedings of the Sixth International Symposium on*, Oct. 1995, pp. 39–43. DOI: 10.1109/MHS.1995.494215. 25

[78] F. V. Berghen and H. Bersini, “Condor, a new parallel, constrained extension of powell’s uobyqa algorithm: Experimental results and comparison with the dfo algorithm”, *Journal of computational and applied mathematics*, vol. 181, no. 1, pp. 157–175, 2005. 25

[79] M. J. Powell, “Least Frobenius norm updating of quadratic models that satisfy interpolation conditions”, *Mathematical Programming*, vol. 100, no. 1, pp. 183–215, 2004. 25

[80] D. C. Sorensen, “Newton’s method with a model trust region modification”, *SIAM Journal on Numerical Analysis*, vol. 19, no. 2, pp. 409–426, 1982. 25

[81] Qualisys, “Qualisys motion capture systems”, *URL: http://www. qualisys. se/, accessed on*, pp. 04–04, 2008. 27

[82] V. H. Vu, B. Isableu, and B. Berret, “Adaptive use of interaction torque during arm reaching movement from the optimal control viewpoint”, *Scientific reports*, vol. 6, p. 38 845, Dec. 2016, ISSN: 2045-2322. DOI: 10.1038/srep38845. [Online]. Available: http://europepmc.org/articles/PMC5151091. 37, 41, 42, 49, 59, 84

[83] R. Zenk, M. Franz, H. Bubb, and P. Vink, “Technical note: Spine loading in automotive seating”, *Applied ergonomics*, vol. 43, no. 2, pp. 290–295, 2012. 37, 41, 63

[84] D. Kee and I. Lee, “Relationships between subjective and objective measures in assessing postural stresses”, *Applied Ergonomics*, vol. 43, no. 2, pp. 277–282, 2012. 37, 41, 63

[85] J. Fox, *An R and S-Plus companion to applied regression*. Sage, 2002. 37

[86] M. Kawato, “Internal models for motor control and trajectory planning”, *Curr. Opin. Neurol.*, vol. 9, no. 6, pp. 718–727, 1999, ISSN: 09594388. DOI: 10.1016/S0959-4388(99)00028-8. 39–41

[87] M. Desmurget and S. Grafton, “Forward modeling allows feedback control for fast reaching movements”, *Trends Cogn. Sci.*, vol. 4, no. 11, pp. 423–431, 2000. 39, 43

[88] D. M. Wolpert, Z. Ghahramani, and M. I. Jordan, “Are arm trajectories planned in kinematic or dynamic coordinates? An adaptation study”, *Exp. Brain Res.*, vol. 103, no. 3, pp. 460–470, 1995. 39

[89] C. Tin and C.-S. Poon, “Internal models in sensorimotor integration: Perspectives from adaptive control theory”, *J. Neural Eng.*, vol. 2, no. 3, S147, 2005. 39

[90] D. M. Wolpert and J. R. Flanagan, “Computations underlying sensorimotor learning”, *Curr. Opin. Neurol.*, vol. 37, no. Supplement C, pp. 7–11, 2016, ISSN: 0959-4388. DOI: https://doi.org/10.1016/j.conb.2015.12.003. 40

[91] M. Descoins, F. Danion, and R. J. Bootsma, “Predictive control of grip force when moving object with an elastic load applied on the arm”, *Exp. Brain Res.*, vol. 172, no. 3, pp. 331–342, Jul. 2006, ISSN: 1432-1106. DOI: 10.1007/s00221-005-0340-3. 40

[92] F. Crevecoeur, J. Thonnard, and P. Lefèvre, “Forward models of inertial loads in weightlessness”, *Neuroscience*, vol. 161, no. 2, pp. 589–598, 2009, ISSN: 0306-4522. DOI: https://doi.org/10.1016/j.neuroscience.2009.03.025. 40

[93] O. White and J. Diedrichsen, “Flexible switching of feedback control mechanisms allows for learning of different task dynamics”, *PLOS ONE*, vol. 8, no. 2, pp. 1–8, Feb. 2013. DOI: 10.1371/journal.pone.0054771. 40

[94] M. Barbiero, C. Rousseau, C. Papaxanthis, and O. White, “Coherent multimodal sensory information allows switching between gravitoinertial contexts”, *Front. Physiol.*, vol. 8, p. 290, 2017, ISSN: 1664-042X. DOI: 10.3389/fphys.2017.00290. 40

[95] M. Kalakrishnan, P. Pastor, L. Righetti, and S. Schaal, “Learning objective functions for manipulation”, in *IEEE Int. Conf. Robot. Autom. (ICRA)*, IEEE, 2013, pp. 1331–1336. 42, 84

[96] S. Ewart, S. M. Hynes, W. G. Darling, and C. Capaday, “A Donders’ like law for arm movements: The signal not the noise”, *Front. Hum. Neurosci.*, vol. 10, p. 136, 2016, ISSN: 1662-5161. DOI: 10.3389/fnhum.2016.00136. 42

[97] J. L. Vercher and G. M. Gauthier, “Oculo-manual coordination control: Ocular and manual tracking of visual targets with delayed visual feedback of the hand motion”, *Exp. Brain Res.*, vol. 90, no. 3, pp. 599–609, Sep. 1992, ISSN: 1432-1106. DOI: 10.1007/BF00230944. 43

[98] R. Miall and D. Wolpert, “Forward models for physiological motor control”, *Neural Netw.*, vol. 9, no. 8, pp. 1265–1279, 1996, Four Major Hypotheses in Neuroscience, ISSN: 0893-6080. DOI: https://doi.org/10.1016/S0893-6080(96)00035-4. 43

[99] D. W. Franklin and D. M. Wolpert, “Computational mechanisms of sensorimotor control”, *Neuron*, vol. 72, no. 3, pp. 425–442, 2011. 43

[100] R. Seidler, D. Noll, and G. Thiers, “Feedforward and feedback processes in motor control”, *NeuroImage*, vol. 22, no. 4, pp. 1775–1783, 2004, ISSN: 1053-8119. DOI: https://doi.org/10.1016/j.neuroimage.2004.05.003. 43

[101] J. S. Perkell, “Movement goals and feedback and feedforward control mechanisms in speech production”, *J. Neurolinguistics*, vol. 25, no. 5, pp. 382–407, 2012, ISSN: 0911-6044. DOI: https://doi.org/10.1016/j.jneuroling.2010.02.011. 43

[102] B. Berret, E. Chiovetto, F. Nori, and T. Pozzo, “Manifold reaching paradigm: How do we handle target redundancy?”, *J. Neurophysiol.*, vol. 106, no. 4, pp. 2086–2102, 2011, ISSN: 0022-3077. DOI: 10.1152/jn.01063.2010. 43

[103] W. Li, E. Todorov, and D. Liu, “Inverse optimality design for biological movement systems”, *Proc. IFAC*, vol. 44, no. 1, pp. 9662–9667, 2011. 43

[104] T. Rajpurohit, W. M. Haddad, and E. A. Theodorou, “Nonlinear-nonquadratic optimal and inverse optimal control for stochastic dynamical systems”, in *Proc. Am. Control Conf. (ACC)*, IEEE, 2016, pp. 6568–6573. 43

[105] K. Yarrow, P. Brown, and J. W. Krakauer, “Inside the brain of an elite athlete: The neural processes that support high achievement in sports”, *Nature Reviews Neuroscience*, vol. 10, no. 8, p. 585, 2009. 46

[106] D. A. Rosenbaum, *Human motor control*. Academic press, 2009. 47

[107] S. E. Engelbrecht, “Minimum principles in motor control”, *Journal of Mathematical Psychology*, vol. 45, no. 3, pp. 497–542, 2001. 47, 49

[108] J. Soechting and F. Lacquaniti, “Invariant characteristics of a pointing movement in man”, *The Journal of Neuroscience*, vol. 1, no. 7, pp. 710–720, 1981. 49

[109] F. Lacquaniti and J. F. Soechting, “Coordination of arm and wrist motion during a reaching task”, *The Journal of Neuroscience*, vol. 2, no. 4, pp. 399–408, 1982. 49

[110] C. Papaxanthis, T. Pozzo, and M. Schieppati, “Trajectories of arm pointing movements on the sagittal plane vary with both direction and speed”, *Experimental brain research*, vol. 148, no. 4, pp. 498–503, 2003. 49

[111] J. F. Soechting and M. Flanders, “11 arm movements in three-dimensional space: Computation, theory, and observation.”, *Exercise and sport sciences reviews*, vol. 19, no. 1, pp. 389–418, 1991. 49

[112] D. M. Wolpert and M. Kawato, “Multiple paired forward and inverse models for motor control”, *Neural networks*, vol. 11, no. 7, pp. 1317–1329, 1998. 50

[113] M. Desmurget, D. Pélisson, Y. Rossetti, and C. Prablanc, “From eye to hand: Planning goal-directed movements”, *Neuroscience & Biobehavioral Reviews*, vol. 22, no. 6, pp. 761–788, 1998. 50

[114] H. Cruse and M. Brüwer, “The human arm as a redundant manipulator: The control of path and joint angles”, *Biological cybernetics*, vol. 57, no. 1-2, pp. 137–144, 1987. 50

[115] D. A. Rosenbaum, L. D. Loukopoulos, R. G. Meulenbroek, J. Vaughan, and S. E. Engelbrecht, “Planning reaches by evaluating stored postures.”, *Psychological review*, vol. 102, no. 1, p. 28, 1995. 50

[116] S. Gielen, “Review of models for the generation of multi-joint movements in 3-d”, in *Progress in Motor Control*, Springer, 2009, pp. 523–550. 50

[117] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, “Maximum entropy inverse reinforcement learning”, in *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008*, D. Fox and C. P. Gomes, Eds., AAAI Press, 2008, pp. 1433–1438. [Online]. Available: http://www.aaai.org/Library/AAAI/2008/aaai08-227.php. 50

[118] N. D. Ratliff, D. Silver, and J. A. Bagnell, “Learning to search: Functional gradient techniques for imitation learning”, *Auton. Robots*, vol. 27, no. 1, pp. 25–53, 2009. DOI: 10.1007/s10514-009-9121-3. [Online]. Available: https://doi.org/10.1007/s10514-009-9121-3. 50

[119] E. Theodorou, J. Buchli, and S. Schaal, “A generalized path integral control approach to reinforcement learning”, *Journal of Machine Learning Research*, vol. 11, pp. 3137–3181, 2010. [Online]. Available: http://portal.acm.org/citation.cfm?id=1953033. 50

[120] S. Levine, Z. Popovic, and V. Koltun, "Nonlinear inverse reinforcement learning with gaussian processes", in *Advances in Neural Information Processing Systems 24: 25th Annual Conf.on NIPS*, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. C. N. Pereira, and K. Q. Weinberger, Eds., 2011, pp. 19–27. [Online]. Available: http://papers.nips.cc/paper/4420-nonlinear-inverse-reinforcement-learning-with-gaussian-processes. 50

[121] C. Finn, S. Levine, and P. Abbeel, "Guided cost learning: Deep inverse optimal control via policy optimization", in *Proceedings of the 33nd International Conference on Machine Learning, ICML*, ser. Workshop and Conference Proceedings, vol. 48, JMLR, 2016, pp. 49–58. [Online]. Available: http://jmlr.org/proceedings/papers/v48/finn16.html. 50

[122] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response", *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 14–29, 2016. 50

[123] J. Mainprice and D. Berenson, "Human-robot collaborative manipulation planning using early prediction of human motion", in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2013, pp. 299–306. 50, 51, 116

[124] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: Learning attractor models for motor behaviors", *Neural computation*, vol. 25, no. 2, pp. 328–373, 2013. 50, 51

[125] H. B. Amor, G. Neumann, S. Kamthe, O. Kroemer, and J. Peters, "Interaction primitives for human-robot cooperation tasks", in *2014 IEEE International Conference on Robotics and Automation, ICRA 2014, Hong Kong, China, May 31 - June 7, 2014*, IEEE, 2014, pp. 2831–2837. DOI: 10.1109/ICRA.2014.6907265. [Online]. Available: https://doi.org/10.1109/ICRA.2014.6907265. 50, 68, 97

[126] Y. Cui, J. Poon, T. Matsubara, J. V. Miró, K. Sugimoto, and K. Yamazaki, "Environment-adaptive interaction primitives for human-robot motor skill learning", in *16th IEEE-RAS International Conference on Humanoid Robots, Humanoids 2016, Cancun, Mexico, November 15-17, 2016*, IEEE, 2016, pp. 711–717. DOI: 10.1109/HUMANOIDS.2016.7803352. [Online]. Available: https://doi.org/10.1109/HUMANOIDS.2016.7803352. 50, 68

[127] A. Paraschos, C. Daniel, J. R. Peters, and G. Neumann, "Probabilistic movement primitives", in *Advances in neural information processing systems (NIPS)*, 2013, pp. 2616–2624. 51–53, 104, 120

[128] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert, "Learning movement primitives", in *Robotics Research. The Eleventh International Symposium*, Springer, 2005, pp. 561–572. 51

[129] Z. Wang, K. Mülling, M. P. Deisenroth, H. B. Amor, D. Vogt, B. Schölkopf, and J. Peters, "Probabilistic movement modeling for intention inference in human–robot interaction", *The International Journal of Robotics Research*, vol. 32, no. 7, pp. 841–858, 2013. 51

[130] G. Maeda, A. Maloo, M. Ewerton, R. Lioutikov, and J. Peters, "Anticipative interaction primitives for human-robot collaboration", in *2016 AAAI Fall Symposium Series*, 2016, pp. 325–330. 51

[131] G. J. Maeda, G. Neumann, M. Ewerton, R. Lioutikov, O. Kroemer, and J. Peters, "Probabilistic movement primitives for coordination of multiple human–robot collaborative tasks", *Autonomous Robots*, pp. 1–20, 2016. 51

[132] G. McLachlan and K. Basford, *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, New York, 1988. 51

[133] S. Calinon, F. D'halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and reproduction of gestures by imitation", *IEEE Robotics & Automation Magazine*, vol. 17, no. 2, pp. 44–54, 2010. 51

[134] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval", *Intelligent Service Robotics*, vol. 9, no. 1, pp. 1–29, 2016. 51

[135] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005, ISBN: 026218253X. 52, 125

[136] A. Lazaric and M. Ghavamzadeh, "Bayesian multi-task reinforcement learning", in *ICML-27th International Conference on Machine Learning*, Omnipress, 2010, pp. 599–606. 52, 120

[137] D. A. Winter, "Coordination of motor tasks in human gait", *Perspectives on the Coordination of Movement*, vol. 61, pp. 329–363, 1989. 55

[138] N. Bernstein, *The co-ordination and regulation of movements*. Pergamon Press, 1967. 55

[139] Y. Tseng, J. P. Scholz, and G. Schöner, "Goal-equivalent joint coordination in pointing: Affect of vision and arm dominance", *Motor control*, vol. 6, no. 2, pp. 183–207, 2002. 55

[140] P. Haggard, K. Hutchinson, and J. Stein, "Patterns of coordinated multi-joint movement", *Experimental Brain Research*, vol. 107, no. 2, pp. 254–266, 1995. 55

[141] K. J. Cole and J. H. Abbs, "Coordination of three-joint digit movements for rapid finger-thumb grasp", *Journal of neurophysiology*, vol. 55, no. 6, pp. 1407–1423, 1986. 55

[142] D. C. Knill and A. Pouget, "The Bayesian brain: The role of uncertainty in neural coding and computation", *TRENDS in Neurosciences*, vol. 27, no. 12, pp. 712–719, 2004. 55

[143] H. S. Koppula and A. Saxena, "Anticipating human activities for reactive robotic response.", in *IROS*, 2013, p. 2071. 55

[144] V. H. Vu, B. Isableu, and B. Berret, "On the nature of motor planning variables during arm pointing movement: Compositeness and speed dependence", *Neuroscience*, vol. 328, pp. 127–146, 2016, ISSN: 0306-4522. DOI: https://doi.org/10.1016/j.neuroscience.2016.04.027. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306452216300926. 55

[145] S. M. Ross, *Introduction to probability models*. Academic press, 2014. 63

[146] A. K. Akametalu, S. Kaynama, J. F. Fisac, M. N. Zeilinger, J. H. Gillula, and C. J. Tomlin, "Reachability-based safe learning with gaussian processes", in *53rd IEEE Conference on Decision and Control, CDC 2014, Los Angeles, CA, USA, December 15-17, 2014*, IEEE, 2014, pp. 1424–1431. DOI: 10.1109/CDC.2014.7039601. [Online]. Available: https://doi.org/10.1109/CDC.2014.7039601. 69

[147] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. H. Gillula, and C. J. Tomlin, "A general safety framework for learning-based control in uncertain robotic systems", *CoRR*, vol. abs/1705.01292, 2017. arXiv: 1705.01292. [Online]. Available: http://arxiv.org/abs/1705.01292. 69, 148

[148] S. Kraus, S. Albrecht, M. Sobotka, B. Heißing, and M. Ulbrich, “Optimisation-based identification of situation determined cost functions for the implementation of a human-like driving style in an autonomous car”, in *International Symposium on Advanced Vehicle Control (AVEC)*, 2010, pp. 412–417. 69

[149] N. Sebanz and G. Knoblich, “Prediction in joint action: What, when, and where”, *Topics in Cognitive Science*, vol. 1, no. 2, pp. 353–367, 2009. 74, 77–79

[150] G. Knoblich and R. Flach, “Predicting the effects of actions: Interactions of perception and action”, *Psychological Science*, vol. 12, no. 6, pp. 467–472, 2001. 74

[151] R. Hayne, R. Luo, and D. Berenson, “Considering avoidance and consistency in motion planning for human-robot manipulation in a shared workspace”, in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, IEEE, 2016, pp. 3948–3954. 74, 76

[152] O. Brock and O. Khatib, “Elastic strips: A framework for motion generation in human environments”, *The International Journal of Robotics Research*, vol. 21, no. 12, pp. 1031–1052, 2002. 76

[153] C. Park, J. Pan, and D. Manocha, “Real-time optimization-based planning in dynamic environments using gpus”, in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, IEEE, 2013, pp. 4090–4097. 76

[154] L. Rozo Castañeda, S. Calinon, D. Caldwell, P. Jimenez Schlegl, and C. Torras, “Learning collaborative impedance-based robot behaviors”, in *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013, pp. 1422–1428. 76

[155] S. Calinon, P. Evrard, E. Gribovskaya, A. Billard, and A. Kheddar, “Learning collaborative manipulation tasks by demonstration using a haptic interface”, in *Advanced Robotics, 2009. ICAR 2009. International Conference on*, IEEE, 2009, pp. 1–6. 76

[156] Y. Li, K. P. Tee, W. L. Chan, R. Yan, Y. Chua, and D. K. Limbu, “Role adaptation of human and robot in collaborative tasks”, in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, IEEE, 2015, pp. 5602–5607. 76

[157] P. Donner and M. Buss, “Cooperative swinging of complex pendulum-like objects: Experimental evaluation”, *IEEE Trans. Robotics*, vol. 32, no. 3, pp. 744–753, 2016. 76

[158] A. Edsinger and C. C. Kemp, “Human-robot interaction for cooperative manipulation: Handing objects to one another”, in *RO-MAN 2007-The 16th IEEE Int.Symposium on Robot and Human Interactive Communication*, IEEE, 2007, pp. 1167–1172. 76

[159] M. Cakmak, S. S. Srinivasa, M. K. Lee, S. Kiesler, and J. Forlizzi, “Using spatial and temporal contrast for fluent robot-human hand-overs”, in *Proceedings of the 6th international conference on Human-robot interaction*, ACM, 2011, pp. 489–496. 76

[160] K. W. Strabala, M. K. Lee, A. D. Dragan, J. L. Forlizzi, S. Srinivasa, M. Cakmak, and V. Micelli, “Towards seamless human-robot handovers”, *Journal of Human-Robot Interaction*, vol. 2, no. 1, pp. 112–132, 2013. 76

[161] H. Admoni, A. D. Dragan, S. S. Srinivasa, and B. Scassellati, “Deliberate delays during robot-to-human handovers improve compliance with gaze communication”, in *ACM/IEEE International Conference on Human-Robot Interaction, HRI’14, Bielefeld, Germany, March 3-6, 2014*, ACM, 2014, pp. 49–56. DOI: 10.1145/2559636.2559682. [Online]. Available: http://doi.acm.org/10.1145/2559636.2559682. 76

[162] E. Ugur, Y. Nagai, E. Sahin, and E. Oztop, “Staged development of robot skills: Behavior formation, affordance learning and imitation with motionese”, *IEEE Transactions on Autonomous Mental Development*, vol. 7, no. 2, pp. 119–139, 2015. 77, 78

[163] E. Ugur and J. H. Piater, “Bottom-up learning of object categories, action effects and logical rules: From continuous manipulative exploration to symbolic planning”, in *IEEE International Conference on Robotics and Automation, ICRA 2015, Seattle, WA, USA, 26-30 May, 2015*, IEEE, 2015, pp. 2627–2633. DOI: 10.1109/ICRA.2015.7139553. [Online]. Available: https://doi.org/10.1109/ICRA.2015.7139553. 77

[164] H. S. Koppula, R. Gupta, and A. Saxena, “Learning human activities and object affordances from rgb-d videos”, *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 951–970, 2013. 77

[165] H. Koppula and A. Saxena, “Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation”, in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, 2013, pp. 792–800. 77

[166] E. E. Aksoy, A. Abramov, J. Dörr, K. Ning, B. Dellen, and F. Wörgötter, “Learning the semantics of object–action relations by observation”, *The International Journal of Robotics Research*, p. 0 278 364 911 410 459, 2011. 77

[167] E. E. Aksoy, M. Tamosiunaite, R. Vuga, A. Ude, C. Geib, M. Steedman, and F. Worgotter, “Structural bootstrapping at the sensorimotor level for the fast acquisition of action knowledge for cognitive robots”, in *Development and Learning and Epigenetic Robotics (ICDL), 2013 IEEE Third Joint International Conference on*, IEEE, 2013, pp. 1–8. 77

[168] M. J. Aein, E. E. Aksoy, M. Tamosiunaite, J. Papon, A. Ude, and F. Wörgötter, “Toward a library of manipulation actions based on semantic object-action relations”, in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2013, pp. 4555–4562. 77, 78

[169] F. Wörgötter, E. E. Aksoy, N. Krüger, J. Piater, A. Ude, and M. Tamosiunaite, “A simple ontology of manipulation actions based on hand-object relations”, *IEEE transactions on autonomous mental development*, vol. 5, no. 2, pp. 117–134, 2013. 77–80, 85, 92

[170] T. Lorenz, B. N. Vlaskamp, A.-M. Kasparbauer, A. Mörtl, and S. Hirche, “Dyadic movement synchronization while performing incongruent trajectories requires mutual adaptation”, *Frontiers in human neuroscience*, vol. 8, 2014. 77, 79

[171] K. Watanabe, “Behavioral speed contagion: Automatic modulation of movement timing by observation of body movements”, *Cognition*, vol. 106, no. 3, pp. 1514–1524, 2008. 77

[172] R. Desimone and J. Duncan, “Neural mechanisms of selective visual attention”, *Annual review of neuroscience*, vol. 18, no. 1, pp. 193–222, 1995. 78, 79

[173] H. E. Egeth and S. Yantis, “Visual attention: Control, representation, and time course”, *Annual review of psychology*, vol. 48, no. 1, pp. 269–297, 1997. 78

[174] C. L. Folk, R. W. Remington, and J. C. Johnston, “Involuntary covert orienting is contingent on attentional control settings”, *Journal of Experimental Psychology Human Perception and Performance*, vol. 18, pp. 1030–1030, 1992. 78

[175] D. Gao, V. Mahadevan, and N. Vasconcelos, “On the plausibility of the discriminant center-surround hypothesis for visual saliency”, *Journal of vision*, vol. 8, no. 7, p. 13, 2008. 78

[176] L. Itti and C. Koch, "Computational modelling of visual attention", *Nature reviews neuroscience*, vol. 2, no. 3, pp. 194–203, 2001. 78

[177] P. D. Tynan and R. Sekuler, "Motion processing in peripheral vision: Reaction time and perceived velocity", *Vision research*, vol. 22, no. 1, pp. 61–68, 1982. 86

[178] J. Sardegna, S. Shelly, and S. Steidl, *The encyclopedia of blindness and vision impairment*. Infobase Publishing, 2002. 86

[179] M. To, B. Regan, D. Wood, and J. Mollon, "Vision out of the corner of the eye", *Vision research*, vol. 51, no. 1, pp. 203–214, 2011. 86

[180] S. Hochreiter and J. Schmidhuber, "Long short-term memory", *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735. 97, 99

[181] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation", in *Proc. of Conf. on Emp. Meth. in Nat. Lang. Proc. (EMNLP)*, Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. 97, 99, 103

[182] D. Vogt, S. Stepputtis, S. Grehl, B. Jung, and H. B. Amor, "A system for learning continuous human-robot interactions from human-human demonstrations", in *IEEE ICRA*, 2017, pp. 2882–2889. DOI: 10.1109/ICRA.2017.7989334. 97

[183] J. Lee and M. S. Ryoo, "Learning robot activities from first-person human videos using convolutional future regression", pp. 1497–1504, 2017. DOI: 10.1109/IROS.2017.8205953. [Online]. Available: https://doi.org/10.1109/IROS.2017.8205953. 97

[184] P. A. Lasota and J. A. Shah, "Analyzing the effects of human-aware motion planning on close-proximity human-robot collaboration", *Human Factors*, vol. 57, no. 1, pp. 21–33, 2015, PMID: 25790568. DOI: 10.1177/0018720814565188. eprint: https://doi.org/10.1177/0018720814565188. 97, 99

[185] D. Lee, C. Ott, and Y. Nakamura, "Mimetic communication model with compliant physical contact in human - humanoid interaction", *I. J. Robotics Res.*, vol. 29, no. 13, pp. 1684–1704, 2010. DOI: 10.1177/0278364910364164. [Online]. Available: https://doi.org/10.1177/0278364910364164. 97

[186] S. Murata, Y. Li, H. Arie, T. Ogata, and S. Sugano, "Learning to achieve different levels of adaptability for human-robot collaboration utilizing a neuro-dynamical system", *IEEE Trans. on Cog. and Dev. Syst.*, vol. PP, no. 99, pp. 1–1, 2018, ISSN: 2379-8920. DOI: 10.1109/TCDS.2018.2797260. 98

[187] D.-C. Nguyen, G. Bailly, and F. Elisei, "Learning off-line vs. on-line models of interactive multimodal behaviors with recurrent neural networks", *Pattern Rec. Letters*, vol. 100, no. Supplement C, pp. 29–36, 2017, ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2017.09.033. 98

[188] S. Pellegrinelli, H. Admoni, S. Javdani, and S. S. Srinivasa, "Human-robot shared workspace collaboration via hindsight optimization", in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2016, Daejeon, South Korea, October 9-14, 2016*, IEEE, 2016, pp. 831–838. DOI: 10.1109/IROS.2016.7759147. [Online]. Available: https://doi.org/10.1109/IROS.2016.7759147. 98

[189] D. Tanneberg, A. Paraschos, J. Peters, and E. Rueckert, “Deep spiking networks for model-based planning in humanoids”, in *IEEE-RAS 16th Int. Conf. on Humanoid Robots (Humanoids)*, Nov. 2016, pp. 656–661. DOI: 10.1109/HUMANOIDS.2016.7803344. 98

[190] A. Rai, G. Sutanto, S. Schaal, and F. Meier, “Learning feedback terms for reactive planning and control”, in *IEEE Int. Conf. on Robot. and Autom. (ICRA)*, May 2017, pp. 2184–2191. DOI: 10.1109/ICRA.2017.7989252. 98

[191] Y.-C. Chu and K. Glover, “Stabilization and performance synthesis for systems with repeated scalar nonlinearities”, *IEEE Trans. Automat. Contr.*, vol. 44, no. 3, pp. 484–496, Mar. 1999, ISSN: 0018-9286. DOI: 10.1109/9.751343. 98

[192] A. Karakaşoğlu and M. K. Sundareshan, “A recurrent neural network-based adaptive variable structure model-following control of robotic manipulators”, *Automatica*, vol. 31, no. 10, pp. 1495–1507, 1995, ISSN: 0005-1098. DOI: https://doi.org/10.1016/0005-1098(95)00057-4. 98

[193] C.-C. Ku and K. Y. Lee, “Diagonal recurrent neural networks for dynamic systems control”, *IEEE Trans. Neural Netw.*, vol. 6, no. 1, pp. 144–156, Jan. 1995, ISSN: 1045-9227. DOI: 10.1109/72.363441. 98

[194] C.-H. Lee and C.-C. Teng, “Identification and control of dynamic systems using recurrent fuzzy neural networks”, *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 4, pp. 349–366, Aug. 2000, ISSN: 1063-6706. DOI: 10.1109/91.868943. 98

[195] A. D. Luca and F. Flacco, “Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration”, in *4th IEEE RAS EMBS Int. Conf. on Biomed. Robot. and Biomech. (BioRob)*, Jun. 2012, pp. 288–295. DOI: 10.1109/BioRob.2012.6290917. 99

[196] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, “Soft-tissue injury in robotics”, in *IEEE Int. Conf. on Robot. and Autom.*, May 2010, pp. 3426–3433. DOI: 10.1109/ROBOT.2010.5509854. 99

[197] F. Agostinelli, M. D. Hoffman, P. J. Sadowski, and P. Baldi, “Learning activation functions to improve deep neural networks”, *CoRR*, vol. abs/1412.6830, 2014. arXiv: 1412.6830. 100

[198] P. E. Gill, W. Murray, and M. H. Wright, *Practical optimization*. Academic Press, 1981, ISBN: 978-0-12-283952-8. 101

[199] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors”, *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986. DOI: 10.1038/323533a0. 101

[200] T. Amaral, L. M. Silva, L. A. Alexandre, C. Kandaswamy, J. M. de Sá, and J. M. Santos, “Transfer learning using rotated image data to improve deep neural network performance”, in *Image Analysis and Recognition - 11th Int. Conf., ICIAR, Portugal, October 22-24, 2014, Proc., Part I*, A. J. C. Campilho and M. S. Kamel, Eds., ser. Lecture Notes in Computer Science, vol. 8814, Springer, 2014, pp. 290–300. DOI: 10.1007/978-3-319-11758-4_32. 102

[201] F. A. Gers, J. A. Schmidhuber, and F. A. Cummins, “Learning to forget: Continual prediction with LSTM”, *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000, ISSN: 0899-7667. DOI: 10.1162/089976600300015015. 103

[202] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks”, in *Proc. of the 13th Int. Conf. on Artificial Intel. and Stat.*, Y. W. Teh and M. Titterington, Eds., ser. Proc. of Mach. L. Res. Vol. 9, Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256. 103

[203] L. E. Kavraki, P. Svestka, J. C. Latombe, and M. H. Overmars, “Probabilistic roadmaps for path planning in high-dimensional configuration spaces”, *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 566–580, Aug. 1996. 117

[204] J. J. Kuffner and S. M. LaValle, “RRT-connect: An efficient approach to single-query path planning”, in *Robotics and Automation (ICRA), IEEE Int. Conf. on*, vol. 2, 2000, 995–1001 vol.2. 117

[205] E. A. Sisbot, L. F. Marin-Urias, R. Alami, and T. Simeon, “A human aware mobile robot motion planner”, *IEEE Trans. Robot.*, vol. 23, no. 5, pp. 874–883, Oct. 2007. 117

[206] M. Zucker, N. Ratliff, A. Dragan, M. Pivtoraiko, M. Klingensmith, C. Dellin, J. A. (Bagnell, and S. Srinivasa, “CHOMP: Covariant Hamiltonian optimization for motion planning”, *Int. J. Rob. Res.*, May 2013. 118

[207] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal, “Stomp: Stochastic trajectory optimization for motion planning”, in *Robotics and Automation (ICRA), IEEE Int. Conf. on*, May 2011, pp. 4569–4574. 118, 119

[208] C. Park, J. Pan, and D. Manocha, “ITOMP: Incremental trajectory optimization for real-time replanning in dynamic environments”, in *ICAPS*, AAAI, 2012. 118–120

[209] R. Lampariello, D. Nguyen-Tuong, C. Castellini, G. Hirzinger, and J. Peters, “Trajectory planning for optimal robot catching in real-time”, in *Robotics and Automation (ICRA), IEEE Int. Conf. on*, May 2011, pp. 3719–3726. 118

[210] R. Weitschat, S. Haddadin, F. Huber, and A. Albu-Schäffer, “Dynamic optimality in real-time: A learning framework for near-optimal robot motions”, in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, November 3-7, 2013*, IEEE, 2013, pp. 5636–5643. DOI: 10.1109/IROS.2013.6697173. [Online]. Available: https://doi.org/10.1109/IROS.2013.6697173. 118

[211] W. Li and E. Todorov, “Iterative linear quadratic regulator design for nonlinear biological movement systems”, in *ICINCO (1)*, INSTICC Press, 2004, pp. 222–229. 120

[212] Á. C. González, H. Admoni, and B. Scassellati, “Effects of form and motion on judgments of social robots’ animacy, likability, trustworthiness and unpleasantness”, *Int. J. Hum.-Comput. Stud.*, vol. 90, pp. 27–38, 2016. 121, 122

[213] D. Matsui, T. Minato, K. F. MacDorman, and H. Ishiguro, “Generating natural motion in an android by mapping human motion”, in *IROS*, IEEE, 2005, pp. 3301–3308. 121

[214] W. Erlhagen, A. Mukovskiy, F. Chersi, and E. Bicho, “On the development of intention understanding for joint action tasks”, in *IEEE 6th International Conference on Development and Learning*, Jul. 2007, pp. 140–145. DOI: 10.1109/DEVLRN.2007.4354022. 129

[215] A. Kirsch, K. Thibault, A. Sisbot, R. Alami, M. Lawitzky, S. Hirche, P. Basili, and S. Glasauer, “Plan-based control of joint human-robot activities”, *KI - Künstliche Intelligenz*, 2010. 129

[216] K. Dautenhahn, S. Woods, C. Kaouri, M. Walters, K. Koay, and I. Werry, "What is a robot companion - friend, assistant or butler?", *International Conference on Intelligent Robots and Systems*, 2005. 129

[217] K. Koay, E. Sisbot, D. Syrdal, M. Walters, K. Dautenhahn, and R. Alami, "Exploratory study of a robot approaching a person in the context of handing over an object", *AAAI spring symposium: multidisciplinary collaboration for socially assistive robotics, pages 18–24*, 2007. 130

[218] D. Bortot, M. Born, and K. Bengler, "Directly or on detours? How should industrial robots approximate humans?", *International Conference on Human-Robot Interaction*, 2013. 130

[219] F. Stulp and O. Sigaud, "Policy improvement methods: Between black-box optimization and episodic reinforcement learning", 34 pages, Oct. 2012, [Online]. Available: https://hal.archives-ouvertes.fr/hal-00738463. 131, 135

[220] B. Busch, J. Grizou, M. Lopes, and F. Stulp, "Learning legible motion from human-robot interactions", *International Journal of Social Robotics*, 2017. 131

[221] D. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields", *IEEE-RAS International Conference on Humanoid Robots*, Dec. 2008. 131, 133

[222] A. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots", *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on, volume 2, pages 1398–1403*, 2002. 131

[223] B. H. Krogh, "A generalized potential field approach to obstacle avoidance control", *International Robotics Research Conference*, 1984. 133

[224] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots", *International Conference in Robotics and Automation*, 1985. 133

[225] A. Maciejewski and C. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments", *The International Journal of Robotics Research*, 1985. 136

[226] T. Kröger, "Opening the door to new sensor-based robot applications—the Reflexxes motion libraries", in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, IEEE, 2011, pp. 1–4. 147

[227] G. Huber, V. Gabler, and D. Wollherr, "An online trajectory generator on SE(3) with magnitude constraints", en, in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Vancouver, Canada, Sep. 2017, pp. 6171–6177. DOI: 10.1109/IROS.2017.8206518. 147

[228] S. Levine, "Reinforcement learning and control as probabilistic inference: Tutorial and review", *ArXiv e-prints*, May 2018. arXiv: 1805.00909 [cs.LG]. 147

[229] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees", in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 908–919. [Online]. Available: http://papers.nips.cc/paper/6692-safe-model-based-reinforcement-learning-with-stability-guarantees. 148

[230] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, “Learning-based model predictive control for safe exploration and reinforcement learning”, *ArXiv e-prints*, Mar. 2018. arXiv: 1803.08287. 148

[231] E. Pignat and S. Calinon, “Learning adaptive dressing assistance from human demonstration”, *Robotics and Autonomous Systems*, vol. 93, pp. 61–75, Jul. 2017. DOI: 10.1016/j.robot.2017.03.017. 148

[232] S. El-Boustani, J. P. K. Ip, V. Breton-Provencher, G. W. Knott, H. Okuno, H. Bito, and M. Sur, “Locally coordinated synaptic plasticity of visual cortex neurons in vivo”, *Science*, vol. 360, no. 6395, pp. 1349–1354, 2018, ISSN: 0036-8075. DOI: 10.1126/science.aao0862. eprint: http://science.sciencemag.org/content/360/6395/1349.full.pdf. [Online]. Available: http://science.sciencemag.org/content/360/6395/1349. 148

[233] F. Yang, D. Lyu, B. Liu, and S. Gustafson, “PEORL: integrating symbolic planning and hierarchical reinforcement learning for robust decision-making”, *CoRR*, vol. abs/1804.07779, 2018. arXiv: 1804.07779. [Online]. Available: http://arxiv.org/abs/1804.07779. 148

[234] D. M. Wolpert and M. S. Landy, “Motor control is decision-making”, *Current opinion in neurobiology*, vol. 22, no. 6, pp. 996–1003, 2012. 148

Own Publications and Supervised Student Projects

[235] O. S. Oguz, Z. Zhou, S. Glasauer, and D. Wollherr, "An inverse optimal control approach to explain human arm reaching control based on multiple internal models", *Scientific Reports*, vol. 8, no. 1, p. 5583, 2018, ISSN: 2045-2322. DOI: 10.1038/s41598-018-23792-7. [Online]. Available: https://doi.org/10.1038/s41598-018-23792-7. 4, 15, 84, 148

[236] O. S. Oguz, Z. Zhou, and D. Wollherr, "A hybrid framework for understanding and predicting human reaching motions", *Frontiers in Robotics and AI*, vol. 5, p. 27, 2018, ISSN: 2296-9144. DOI: 10.3389/frobt.2018.00027. [Online]. Available: https://www.frontiersin.org/article/10.3389/frobt.2018.00027. 5, 45, 74

[237] V. Gabler, T. Stahl, G. Huber, O. S. Oguz, and D. Wollherr, "A game-theoretic approach for adaptive action selection in close distance human-robot-collaboration", in *2017 IEEE International Conference on Robotics and Automation (ICRA) Singapore, May 29 - June 3, 2017*, IEEE, Singapore: IEEE, 2017, pp. 2897–2903, ISBN: 9781509046324. 5, 47, 49, 97, 98

[238] O. S. Oguz, V. Gabler, G. Huber, Z. Zhou, and D. Wollherr, "Hybrid human motion prediction for action selection within human-robot collaboration", in *International Symposium on Experimental Robotics, ISER 2016, Tokyo, Japan, October 3-6, 2016.*, D. Kulic, Y. Nakamura, O. Khatib, and G. Venture, Eds., ser. Springer Proceedings in Advanced Robotics, vol. 1, Springer, 2016, pp. 289–298. DOI: 10.1007/978-3-319-50115-4_26. [Online]. Available: https://doi.org/10.1007/978-3-319-50115-4_26. 5, 49, 50, 97

[239] K. H. Dinh, O. S. Oguz, G. Huber, V. Gabler, and D. Wollherr, "An approach to integrate human motion prediction into local obstacle avoidance in close human-robot collaboration", in *Proceedings of IEEE Workshop on Advanced Robotics and its Social Impacts, ARSO*, 2015. DOI: 10.1109/ARSO.2015.7428221. 5, 47

[240] O. S. Oguz, W. Rampeltshammer, and D. Wollherr, "An ontology for human-human interaction and learning interaction behavior models", *ACM Transactions on Human-Robot Interaction (under review)*, 2018. 7, 73

[241] O. S. Oguz, B. M. Pfirrmann, M. Guo, and D. Wollherr, "Learning hand movement interaction control using rnns: From hhi to hri", *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4100–4107, Oct. 2018c, ISSN: 2377-3766. DOI: 10.1109/LRA.2018.2862923. 7, 95, 148

[242] O. S. Oguz, O. C. Sari, K. H. Dinh, and D. Wollherr, "Progressive stochastic motion planning for human-robot interaction", in *26th International Symposium on Robot and Human Interactive Communication, RO-MAN 2017, Lisbon, Portugal, August 28 - Sept. 1, 2017*, IEEE, 2017, pp. 1194–1201. DOI: 10.1109/ROMAN.2017.8172456. [Online]. Available: https://doi.org/10.1109/ROMAN.2017.8172456. 9, 47, 49, 51, 68, 76,

[243] O. S. Oguz, A. Kucukyilmaz, T. Sezgin, and C. Basdogan, "Improving human-computer cooperation through haptic role exchange and negotiation", in *Immersive Multimodal Interactive Presence*, Springer London, 2012, pp. 229–254. 76

[244] O. S. Oguz, A. Kucukyilmaz, T. Metin Sezgin, and C. Basdogan, "Supporting negotiation behavior with haptics-enabled human-computer interfaces", *IEEE Transactions on Haptics*, vol. 5, no. 3, pp. 274–284, 2012.

[245] O. S. Oguz, A. Kucukyilmaz, T. M. Sezgin, and C. Basdogan, "Haptic negotiation and role exchange for collaboration in virtual environments", in *2010 IEEE Haptics Symposium, HAPTICS 2010*, 2010, pp. 371–378.

[246] W. Rampeltshammer, "Ontology of human-human interaction", Master's thesis, Technical University Munich, Dec. 2016.

[247] Z. Zhou, "Understanding and predicting human reaching motions by using inverse optimal control", Master's thesis, Technical University Munich, Jan. 2017.

[248] O. C. Sari, "Online motion planning framework for human robot interaction", Master's thesis, Technical University Munich, Mar. 2017.

[249] B. Pfirrmann, "Interaction learning with recurrent neural networks", Master's thesis, Technical University Munich, Jan. 2018.

[250] H. Zhu, "Associating sensor traces with dynamic movement primitives", Master's thesis, Technical University Munich, Apr. 2018.

[251] S. Paillan, "Safe deep reinforcement learning for human-robot interaction", Master's thesis, Technical University Munich, Oct. 2018.

[252] S. Riemann, "Dynamic movement primitives for human-robot interaction", Bachelor's Thesis, Technical University Munich, Aug. 2015.

[253] S. Gasse, "Learning human-human collaborative behavior in pick and place tasks", Bachelor's Thesis, Technical University Munich, Apr. 2016.

[254] N. Donia, "Predictable robot motions using policy improvement", Bachelor's Thesis, Technical University Munich, Aug. 2016.

[255] F. Klein, "Human posture prediction for upper body reaching tasks", Bachelor's Thesis, Technical University Munich, Jul. 2016.

[256] B. Cetinkaya, "Guided policy search towards building interaction skills", Bachelor's Thesis, Technical University Munich, Feb. 2017.

[257] O. C. Oguz, "Stochastic optimal feedback control for modeling motor control of human reaching tasks", Bachelor's Thesis, Technical University Munich, Mar. 2017.

[258] O. Shalaby, "Natural human-robot interaction using policy improvement", Bachelor's Thesis, Technical University Munich, Aug. 2017.

[259] E. Mattioli, “Correlation analysis between robot motion and human perception”, Bachelor’s Thesis, Technical University Munich, Oct. 2017.

[260] M. Elsayed, “Predictable robot motion generation with dynamic obstacle avoidance for human-robot interaction”, Bachelor’s Thesis, Technical University Munich, Jun. 2018.

[261] A. Magdy, “Constrained hybrid reinforcement learning”, Bachelor’s Thesis, Technical University Munich, Jul. 2018.

[262] M. Shalaby, “Human motor control based on stochastic optimal sliding mode”, Bachelor’s Thesis, Technical University Munich, Jul. 2018.