Me and You Together: Collaboration Between a Robot and a Human in a Manipulation Game

Miguel Afonso Tomás Faria

Thesis to obtain the Master of Science Degree in Information Systems and Computer Engineering

Supervisors: Prof. Francisco António Chaves Saraiva de Melo
Prof. Ana Maria Severino de Almeida e Paiva

Examination Committee

Chairperson: Prof. Daniel Jorge Viegas Gonçalves
Supervisor: Prof. Francisco António Chaves Saraiva de Melo
Member of the Committee: Prof. Manuel Fernando Cabido Peres Lopes

November 2016
I would like to start by thanking to both my supervisors, Prof. Francisco S. Melo and Prof. Ana Paiva, for their support during this thesis. They were always available to help and advise me whenever I had doubts regarding how to solve a problem or if ran into a trouble spot. They were always there to support my writing, by steering me in the best practices.

I also would like to thank my colleagues Rui Silva and Hang Yin for the help they gave in the early stages of this work and when I needed help with technical problems.

I would like to thank to Ruben Rebelo and Vânia Mendoça, two colleagues that were always available to test changes I had done to the scenario and give me their valuable feedback.

I would also like to acknowledge Patrícia Alves-Oliveira for her help in elaborating the questionnaires and analyze the results of the studies.

Also, I would like to thank the people that took time out of their day to be part of the user studies, without them this work could not have been finished.

Finally, I must express my very profound gratitude to my family that in the last year has done nothing else than support me and always guarantee that I had the necessary conditions and space to work on this thesis. This, besides the continuous support and encouragement throughout my years of study, makes them the biggest support I have.

I cannot finish these acknowledgments without thanking to my wonderful friends that were always there to support me when I had doubts if I was really doing things right or if I had to discuss some idea. They were also very important to keep my spirits up, even through the times of most stress, when they would persuade me to go out and have a little of fun.

To all of you, my sincere greatest thanks

Miguel Faria
Abstract

In this work we investigate how a robot leading a collaborative task can be expressive and efficient. To that purpose we present a work on collaborative manipulation task between a robot and multiple humans, which focuses on understanding how a robot’s movement impacts the human partners’ perception of its intentions during the collaboration. We propose a system architecture for the robot based on the framework of Collaborative Probabilistic Movement Primitives (CoPMP), for the generation of the robot’s motion, combined with the notions of legibility and predictability of motions to express the robot’s intentions. The CoPMP are useful since they allow the use of learning from demonstration to teach movements to the robot and create models to combine those demonstrated movements in new ones adapted to collaborative purposes. Besides legible and predictable motions, we developed a third approach - hybrid motion - that gives the robot the capability of deciding between executing a predictable motion or a legible motion. This approach decides which motion to execute depending on what the robot perceives as better for the collaboration, given the fact that it interacts with more than one user. To test the impact that these three approaches have on people, we designed an user study with a Baxter robot, where it fills cups of water to three people that are asking it simultaneously, while trying to be as expressive as possible. The results obtained show that the hybrid motion performs better than the other motions in this particular scenario.

Keywords

Artificial Intelligence; Robot; Human-Robot Collaboration; Robot Manipulation; Motion Decision; Intention Transmission; Collaborative Manipulation
Resumo

Neste trabalho investigamos como um robot, líder numa tarefa colaborativa, pode ser expressivo e eficiente. Para esse efeito, apresentamos um trabalho na tarefa de manipulação colaborativa entre um robot e vários humanos, que se concentra na compreensão de como os movimentos de um robot têm impacto na percepção dos parceiros humanos sobre as suas intenções durante a colaboração. Propomos uma arquitetura de sistema para o robot com base em Collaborative Probabilistic Movement Primitives (CoPMP), para a geração de movimento do robot, combinado com as noções de legibilidade e previsibilidade de movimentos para melhor expressar as intenções do robot. As CoPMP são úteis, uma vez que permitem o uso de aprendizagem a partir de demonstração para ensinar os movimentos ao robot e criar modelos para combinar os movimentos demonstrados em novos, adaptados para fins de colaboração. Além movimentos legíveis e previsíveis, desenvolvemos uma terceira abordagem - movimento híbrido - que dá ao robot a capacidade de decidir entre a execução de um movimento previsível ou um movimento legível. Esta abordagem decide qual o movimento para executar dependendo do que o robot percebe como melhor para a colaboração, dado o facto de que ele interage com mais do que um utilizador. Para testar o impacto que essas três abordagens têm nas pessoas, nós planeámos um estudo com utilizadores usando um robot Baxter, onde ele enche copos de água a três pessoas que estão pedindo-o simultaneamente, ao tentar ser tão expressivo quanto possível. Os resultados obtidos mostram que o movimento híbrido tem um desempenho melhor do que as outras propostas neste cenário particular.

Palavras Chave

Inteligência artificial; Robot; Colaboração Pesso-Robot; Manipulação com Robot; Decisão de Movimento; Transmissão de Intenção; Manipulação Colaborativa
Contents

1 Introduction 1
  1.1 Motivation .................................................. 3
  1.2 Problem ..................................................... 3
  1.3 Hypotheses .................................................. 4
  1.4 Scientific Contribution ....................................... 5
  1.5 Document Outline ........................................... 5

2 Related Work 7
  2.1 Background .................................................. 9
    2.1.1 Human-Robot Collaboration ............................... 9
    2.1.2 Machine Learning ........................................ 11
    2.1.3 Motion Control using Planning ............................ 12
    2.1.4 Motion Control using Learning ............................ 12
    2.1.5 Predictability and Legibility ............................. 14
  2.2 State of the Art ............................................. 15
    2.2.1 Motion Decision Techniques ............................. 15
      2.2.1.A Covariant Hamiltonian Optimization Motion Planning 17
      2.2.1.B STOMP: Stochastic Trajectory Optimization Motion Planning 17
      2.2.1.C Interaction Primitives ................................. 18
      2.2.1.D Probabilistic Movement Primitives ....................... 19
      2.2.1.E Influences in Motion Decision .......................... 20
    2.2.2 Safe Human-Robot Collaboration Interactions ............... 21
      2.2.2.A Influences on safety in Human-Robot Collaboration (HRC) interactions 22
    2.2.3 Transmission of Intention Through Movement ................. 23
      2.2.3.A Influences in Intention Transmission ................... 23

3 Multi-User Cup Filling System 25
  3.1 Vision Module ............................................. 28
    3.1.1 Kinect V2 .............................................. 28
List of Figures

2.1 Some examples of current applications of robots in HRC. In 2.1(a) we see HOBBIT, a robot applied to elderly care. In 2.1(b) we see Robonaut, a robot used to help astronauts. In 2.1(c) we see a robot helping a surgeon. In 2.1(d) we see BEAR, a robot used to help the military. .................................................. 10

2.2 Policy derivation using the generalization approach of determining (a) an approximation to the state to action mapping function, (b) a dynamics model of the system model and (c) a plan of sequenced actions. [3] .................................................. 13

2.3 Predictable motion and Legible motion results, from [13]. On top comparative results on handwriting using predictable motions (on left) and legible motions (on right). On the bottom part, the comparative results when grasping an object, using predictable motion (on left) and legible motion (on right), with the bars showing the person’s level of confidence regarding the robot’s objective. .................................................. 15

3.1 Robot’s system architecture .................................................. 27

3.2 Kinect camera with the color and infrared (IR) cameras marked. .................................................. 29

3.3 Comparison of images recorded by the color and IR cameras. On the left a frame from the color image. On the right a frame of the depth image for the same scene. As possible to observe, each camera’s image does not capture the same exact same objects: the color image captures more on the width dimension, while the IR image captures more of the height dimension. .................................................. 30

3.4 Result from segmenting a possible setting for the cups to be filled. On top the original color image. On the bottom, on the left the result of segmenting to find orange objects, on the middle segmenting to find green objects and on the right color segmentation to find blue objects. .................................................. 32

3.5 Image showing the result of identifying the centres of mass in the original color frame recorded, the centres of mass for each object are marked as black circles. .................................................. 35
3.6 When moving to the left cup, executing a legible motion causes more confusion, by moving almost to the blue before closing in on the orange one to serve it.  

4.1 Baxter robot, developed by “Rethink Robotics”.  

4.2 The scenario layout. The participants extend the cups as a sign to ask for water, as seen on the left, and the robot, on the right, proceeds to serve them one at a time. The participants are told, when given the correspondent cup, to place themselves in a specific mark on the floor. There is no pre-defined ordering for the cups to be filled and after the robot starts to move the participants have to understand who the robot is moving towards and try to facilitate the robot’s movement.  

4.3 The robot starts to reach for one of the cups. When one person understands that it is reaching for him/her, he/she reaches for robot. If it understands it is not moving towards him/her, the person helps the robot by moving the cup away.  

5.1 Average time, in seconds, each participant took to understand the robot was going to serve him, organized per movement type.  

5.2 Average time, in seconds, each participant took to understand the robot was not going to serve him, organized per movement type.  

5.3 Results for the Hoffman’s questionnaire for evaluating fluency in Human-Robot Collaboration metrics - perceived fluency, robot contribution, safety and capability - and for the perceived predictability and legibility measures. The results are in a 6-point likert scale.  

5.4 Results for perceived intelligence and perceived animacy of the robot. The results are in a 5-point likert scale.  

5.5 Results for forced questions, show the total frequency each condition was chosen in the questions.
List of Tables

5.1 Cronbach’s alpha results for fluency, trust, safety, capability and robot contribution measures. ................................................................. 61
# Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HRC</td>
<td>Human-Robot Collaboration</td>
</tr>
<tr>
<td>HRI</td>
<td>Human-Robot Interaction</td>
</tr>
<tr>
<td>DOF</td>
<td>Degrees of Freedom</td>
</tr>
<tr>
<td>DMP</td>
<td>Dynamic Motion Primitives</td>
</tr>
<tr>
<td>MP</td>
<td>Movement Primitives</td>
</tr>
<tr>
<td>DTW</td>
<td>Dynamic Time Warping</td>
</tr>
<tr>
<td>IP</td>
<td>Interaction Primitives</td>
</tr>
<tr>
<td>ProMP</td>
<td>Probabilistic Movement Primitives</td>
</tr>
<tr>
<td>IPMP</td>
<td>Interaction Probabilistic Movement Primitives</td>
</tr>
<tr>
<td>GMM</td>
<td>Gaussian Mixture Model</td>
</tr>
<tr>
<td>GMR</td>
<td>Gaussian Mixture Regression</td>
</tr>
<tr>
<td>STOMP</td>
<td>Stochastic Trajectory Optimization Motion Planning</td>
</tr>
<tr>
<td>CHOMP</td>
<td>Covariant Hamiltonian Optimization Motion Planning</td>
</tr>
<tr>
<td>HTN</td>
<td>Hierarchical Task Networks</td>
</tr>
<tr>
<td>HMC</td>
<td>Hamiltonian Monte Carlo</td>
</tr>
<tr>
<td>HMM</td>
<td>Hidden Markov Models</td>
</tr>
<tr>
<td>RL</td>
<td>Reinforcement Learning</td>
</tr>
<tr>
<td>LfD</td>
<td>Learning from Demonstration</td>
</tr>
<tr>
<td>KLfD</td>
<td>Keyframe Learning from Demonstration</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>ROS</td>
<td>Robot Operating System</td>
</tr>
<tr>
<td>IR</td>
<td>infrared</td>
</tr>
<tr>
<td>FOV</td>
<td>Field of View</td>
</tr>
<tr>
<td>HD</td>
<td>High Definition</td>
</tr>
<tr>
<td>SD</td>
<td>Standard Definition</td>
</tr>
<tr>
<td>CoPMP</td>
<td>Collaborative Probabilistic Movement Primitives</td>
</tr>
<tr>
<td>JTAS</td>
<td>Joint Trajectory Action Server</td>
</tr>
</tbody>
</table>
# Introduction

## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Motivation</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Problem</td>
<td>3</td>
</tr>
<tr>
<td>1.3 Hypotheses</td>
<td>4</td>
</tr>
<tr>
<td>1.4 Scientific Contribution</td>
<td>5</td>
</tr>
<tr>
<td>1.5 Document Outline</td>
<td>5</td>
</tr>
</tbody>
</table>
1.1 Motivation

In today's society interaction between robots and humans is increasing, either in a work capacity or to keep humans company. They are used in assembly lines, elderly care, to take care, entertain or teach children, amongst other applications. This creates a need to understand how to better integrate robots in these tasks and how to make this interaction more seamless and safe for the humans that share the same space as the robots.

Being used in the most diversified environments from homes, offices, hospitals and other more challenging environments such as outer space, the area of Human-Robot Collaboration (HRC) is not only very important but has a high economic impact. [5] Research in the area is responsible for several breakthroughs on how to improve the interaction between humans and robots, by making these interactions safer and less expensive.

However, in these kinds of tasks, humans and robots do not only share the same space but also need to work together to achieve a specific goal or objective. To that end, they combine both teamwork, communication and intent expression methods to reach the goal in an effective and safe manner. This shows how important it is that the interaction goes smoothly and without problems, otherwise the task can be seriously impaired and even impossible to complete. [9,26,27]

This work integrates the area of HRC, more precisely the study of how a robot can learn different motions to perform a certain task and how different motion types interfere in the collaboration and cooperation between the different players of the task.

Specifically, this work aims to study how different motions, performed by the robot during a collaborative manipulation task, influence the way people perceive the robot's objective.

The system developed performs a collaborative task, with multiple people, in a fast, efficient and clear way. Showing how giving a robot the ability to select the motion type to perform improves collaboration efforts.

1.2 Problem

As presented in the previous section, this work aims to study the influence of a robot's movement on people's perception about the robot's target, during a collaboration task. This is important since implicit communication, like correctly inferring a robot's target from its movements, plays a crucial role in the efficiency of a collaborative task's execution. [5, 7, 54]

To study this problem, we develop a robot capable of serving cups of water to several people. At first glance this may appear as a minor task, since we all are capable of reaching for the cabinet where our glasses are, take one out and pour water in it.
However, there are situations where we do not serve drinks only to ourselves or to a small number of people. There are settings such as public cafeterias or some restaurants where one person, normally a bartender or employee, has to serve lots of people in a short period of time. In these settings people usually adopt one of two ways to ask for the drink: either form an orderly queue and follow a first in first out organization or they all swarm the employee responsible and ask to be served. It is up to the responsible to choose who he serves, the order in which he serves and also the area in which he/she accepts the requests.

It is in settings of one person to serve many that a robot is useful, since they are good in repetitive tasks. Using these scenarios as the basis for this work, we focus on the case where the people swarm the robot and the robot not only has to serve them, but to choose who to serve and move accordingly without giving a false idea of who its target is.

Having this in mind, the question that will guide this thesis is:

- In a multiple user collaborative manipulation task, where the humans do not previously know the robot’s objectives, what is the best type of trajectory to perform, in order to minimize confusion between users regarding the robot’s target?

### 1.3 Hypotheses

In line with prior results of Dragan et al. [15], we expect to verify that the participants notice a difference between legible and predictable movements and that they will prefer legible movements to predictable movements.

Given that there are multiple people interacting with the robot at the same time, we expect that when there are multiple possible objectives for the robot, legible movements and hybrid movements will result in quicker response times, with hybrid movements being better than legible movements.

Regarding hybrid movements, we expect that these will result in shorter reaction times and to be preferred over the other movements, although not necessarily less confusing than legible movements. Also, we expect they will be perceived as more legible and predictable than the other movements.

Finally, we expect that the participants will perceive the collaboration between the robot and the humans differently depending on the type of movement being executed, but their perception of intelligence of the robot will not be influenced.

To summarize:

- **H1** - The perception of collaboration will be impacted by the type of movement performed.

- **H2** - Participants will prefer and consider easier to work with hybrid movements to legible movements and legible movements to predictable movements.
• H3 - Hybrid motions will result in better efficiency of the task by decreasing time took to understand the robot’s intentions and also will make people think of it as more legible and predictable.

• H4 - There will not be a difference in perceived confusion between hybrid and legible motions.

• H5 - The perceived intelligence of the robot will not be impacted by the type of movement performed.

1.4 Scientific Contribution

With this thesis, we contribute to the area of HRC and Human-Robot Interaction (HRI) by providing more in-depth insights on collaboration between robots and people. We also hope to contribute to better the integration of robots in society and the relation between people and them.

To that end, we are going to study the interaction between people and a robot, during a collaboration, trying to find the best ways for the robot to fulfill its task efficiently and without causing harm to the people involved. With this study, we contribute with insights about the effects on people’s understanding of a robot’s intentions of different movement types during collaborative multi-user manipulation tasks.

This work also contributes with a novel motion type that combines legibility and predictability, in order to improve movement expressiveness, by allowing the robot to choose which motion type to perform. We also contribute with a study on how this motion type impacts the collaboration effort and the movements capability of transmitting intention.

1.5 Document Outline

The rest of this document will present the research conducted during this work, the system developed to perform this study, the user study carried out and its results. We end with the main conclusions and findings to take out of this work.

In the next section, the Related Work, we will explore some of the most recent breakthroughs and developments in HRC and motion teaching and how they influence this work.

In the third section we will present the system’s architecture.

The fourth section will explain how we designed the user study and the task performed and in the fifth we will analyze the results from the study conducted. In the final section we will present the main conclusions of this work and what we have planned for this work in the future.
Related Work

Contents

2.1 Background ......................................................... 9
2.2 State of the Art ..................................................... 15
2.1 Background

Before starting to explore the works that constitute the most recent advancements in the area of HRC and in robot motion decision, there are some concepts that are useful for this work and as such must be explained.

The concepts that will be explored are Human-Robot Collaboration, machine learning, motion decision with planning, motion decision with learning and the concepts of predictability and legibility.

2.1.1 Human-Robot Collaboration

Human-Robot Collaboration is a part of HRI that focuses mostly on the study of how to make people and robots interact in a safe and efficient way during tasks that require both parties to collaborate, either in a cooperative manner or just working side-by-side sharing the same physical space, to achieve a goal.

HRC began as a result of the robots being used in more than just industrial settings and also because people started to work in more proximity of robots. This changes created a need to make the interaction safer when humans and robots had to share the same workspace or had to work in team or in collaboration with robots. The created need also fostered a big interest in these studies. [9,25,26]

The area has accompanied the evolutions of Robotics and has also allowed robots to be used in new contexts, mostly only imagined in Science Fiction stories, such as being surgeons or personal assistants. With the studies of HRC tasks and settings, robots have been introduced in activities that range from accompanying the elderly to operating in operation rooms side-by-side with surgeons. [9] This way, robots are starting to be more than simple tools to become companions, assistants and even peers in work environments and everyday life.

Currently robots can be found, amongst other applications, working with people in kindergartens helping children learn notions of geometry and helping with storytelling, playing some roles in theatre plays. Robots have also been helping in military operations extracting injured soldiers or carrying heavy objects during rescue missions [9] and in space exploration assisting astronauts in logistic and maintenance related tasks. [25] In Figure 2.1 we can see some examples of robots applied in those areas.

Aside from these applications, robots have been heavily used in factories and industrial tasks in assembly lines side-by-side with other human workers. Some examples include the work of robots in some factories of BMW, Audi and Volkswagen, three major German automobile manufacturers, where robots are being used to perform tasks that could cause repetitive strain injuries and other tasks that are not so ergonomically safe. [42] There are already plans to introduce robots that perform even more assistance by giving workers the tools and parts needed in certain assembly processes, relieving the humans from the work load of having to go get those tools and parts or even remember that they are not there, allowing the worker to be more focused in the task at hand. [42]
Figure 2.1: Some examples of current applications of robots in HRC. In 2.1(a) we see HOBBIT, a robot applied to elderly care. In 2.1(b) we see Robonaut, a robot used to help astronauts. In 2.1(c) we see a robot helping a surgeon. In 2.1(d) we see BEAR, a robot used to help the military.

The applications just described were possible due to a combination of works in the various areas of investigation in HRC that go from the study of motion to the study of how to communicate or understanding intentions subjacent to certain movements and how communication can be done in collaboration tasks without hindering the task’s performance.

This way the main areas of investigation that help improving the area of human-robot collaboration are robot motion learning and planning, communication, robot design and machine learning.

Motion learning and planning are two ways to have a robot decide the correct movement to make and are important for HRC for two main reasons:

1. The correct decision about the movement allows a robot to collaborate with humans without hitting them or other obstacles in the workspace;
2. correctly performed movements allow the robot to better convey its intentions, which enables the other participants to quickly understand who the target is and adapt.

These reasons are important because they lead to a better performance of the task and a bigger sense of security by the human participant.

Communication is responsible for studying how humans communicate during tasks, where different teams have to work together to fulfill their objectives and then trying to replicate those behaviours in robots. It is really important since in this type of tasks there is always a need to communicate intention, even in indirect ways such as with subtle head nods or through facial expressions, and the same needs to happen with robots so the task performance is natural and efficient. [5, 7, 31, 54] This communication can be done using two types of communication [5]:

- explicit communication cues such as speech, gaze, gestures, etc.
- implicit communication cues such as facial expressions, emotional cues, haptic communications, etc.

Both types are very important and must be addressed equally in HRC, otherwise can impair the task performance in terms of time consumed and naturalness of the task execution.

Robot design is also a big part of HRC. Robot design is very important for the robots that are created and used for these tasks because they have to be safe to be around of and for people to be comfortable and confident when having to perform tasks in close proximity to a robot or a group of robots. [5, 27, 42] Because of that there are a set of principles that must be followed so the interaction is safe and without hindering the task performance. Some are collision detection, stopping functions for the robot, limitation of the robot’s motion and movement speed, minding the collaborative requirements of the tasks to perform and technological and ergonomic requirements. [42]

Finally, machine learning is an important part of HRC because it is impossible to program a robot to expect every possible case and occurrence in performing a collaborative task. So we need to give the robot the ability to adapt to changes in the task variables and not fail in performing the task. This is achieved by allowing robots to learn how to respond to certain stimuli without impairing the task execution and the interaction, by extracting patterns and making predictions based on the data used to train. [5, 44]

2.1.2 Machine Learning

Machine Learning is a specific area of Computer Science focused on developing algorithms capable of learning and creating correct models of knowledge from training datasets, allowing a machine to make correct predictions about new data. Machine learning has been successfully used in many areas
of application like speech recognition, computer vision, bio-surveillance, robot control and accelerating empirical sciences. [44].

There are various learning strategies used, in machine learning, to enable a machine to learn and create these models of knowledge. Examples of learning strategies include learning from instruction, learning from analogy and learning from examples. [43]

In the area of robotics one of the most used type of machine learning approaches is Reinforcement Learning (RL) [11, 34], mostly in learning correct sequences of actions, since it is a method that allows a robot to discover an optimal behaviour through trial-and-error. This is useful since it is impossible for the robot’s developer to know every possible scenario and through trial-and-error the robot can more easily maximize its gain than if for example the focus was figuring out what are the rules behind the task. Consider the example of training a robot to return a table tennis ball over the net, as in [47]. In this example the robot might make observations about the dynamic variables involved and with RL the robot infers a policy to correctly move its arm and return the ball. The use of RL can lead to better results since it is difficult to teach a robot every possible occurrence in a table tennis game. [34]

2.1.3 Motion Control using Planning

Using planning as an approach to decide the best movement to perform is a technique much used in mobile robotics and robot motion literature [13–17, 22, 38–41]. The idea in this type of approach is for the robot to plan a trajectory from a starting configuration of the joint space to an objective configuration. There are various methods used to create plans that minimize the energy spent and jerk of the robot’s joints to perform the movement and, optionally, to make it more natural and smooth. Planners are much used since they are one of the oldest ways of having a machine perform a complex task that involves multiple different actions. Some approaches using planning are explained later as CHOMP [52] and STOMP [32].

2.1.4 Motion Control using Learning

Motion decision can be taught to a robot using learning. This type of approach is fairly recent but has shown very good results, as is the case with Stanley [58] the robotic car than won the 2005 DARPA Grand Challenge. The use of this approach gives robots more adaptability to changes in the environment, by using methods like reinforcement learning.

The use of learning for motion decision is the main building block in Learning from Demonstration (LfD), where the robot learns to move given human demonstration and uses those demonstrations to infer the correct movement. [1, 3]

The decision process is the part where the “robot intelligence” is present and where the learning
occurs. In order for the robot be capable of deciding the best movement, it must be able to derive a policy from the demonstrated data. The policy derivation can be done in one of three techniques [3], as summarized in Figure 2.2:

1. using a mapping function, in which the robot learns a policy that directly approximates the observed state - \( Z \) - to the corresponding action - \( A \) \((f() : Z \rightarrow A)\).

2. using a system model, in which the robot determines a state transition model of the world \((T(s'|s,a))\), for each tuple \((s - state, a - action, s' - next state)\), and a reward function \((R(s))\) from the demonstration data and uses them to derive a policy.

3. using plans, this technique borrows a little from decision with planning and uses the demonstration data and some additional user intention information to learn a set of rules \((L)\) that associate pre- and post-conditions \((preC \text{ and } postC)\) with each action \((L(preC, postC|a))\) and an optional state model \((T(s'|s,a))\). The rules and model are then used by a planner to generate a sequence of actions.

The mapping function approach tries to reproduce the underlying teacher policy in the training data and generalize it for similar situations that may not be part of the training data. This mapping function is generally done either by using classification techniques, e.g. Gaussian Mixture Model (GMM) and Hidden Markov Models (HMM), when the resulting function is discrete or regression techniques, e.g. Gaussian Mixture Regression (GMR) and Neural Networks, when the resulting function is continuous. In the case of classification techniques, the training data is grouped by similarity of the input data, resulting into discrete categorization of the inputs that correspond to an expected action. Regression techniques
also work by dividing the training data by inputs, but instead of using discrete classifications, these
techniques try to group similar training instances in continuous domains instead of discrete categories
and when new instances occur, these technologies instead of assigning a category that corresponds to
an action, they select the action that better suits the inputs on a continuum range of values.

The system model approach tries to derive the correct action policy by using a RL structure. In
this approach, the system uses the training data to learn a transition function and then the policy is
obtained using a rewards function. The reward function that is used can either be learned, using RL
techniques, or it can be defined by the user. Thus the techniques based on this approach can be
divided in two categories: when the reward function is defined by the user they are called engineered
reward techniques and when the reward function is learned the techniques are called learned reward
techniques. [3]

The plans approach derive the action policy through desired robot behaviors presented as plans.
Each of these presented plans represents the policy as a sequence of actions that lead the robot from
the initial state to the final goal state, the actions are defined in terms of the pre-conditions the state must
meet and post-conditions of that action. Besides training data, methods that use this approach also rely
on annotations done by the teacher for the system to learn the set of rules of pre- and post-conditions
for each action and the state transition model of the world. With these set of rules and transition model,
a planner then selects the best sequence of actions for the robot to go from the initial state to the goal
state.

2.1.5 Predictability and Legibility

Predictability and legibility are terms used in everyday life to describe movements and behaviours
which we are expecting to result in a certain outcome. However in robot motion they are different
concepts that represent different and sometimes contradictory properties. [13]

A predictable motion matches the movement we expect to see in order to achieve a certain objective.
A legible motion, on the other hand, allows a person to understand the robot's objective quickly even
when that objective is unknown to the person. [17] Figure 2.3 shows the differences in results of using
predictable motions and legible motions in handwriting and grasping objects. As it is possible to observe,
in the case of predictable motions the inference goes from the goal to the action and in the case of legible
the inference goes from the action to the goal.

So a legible motion is a more "readable" or "understandable" motion because it allows a person to
quickly and with confidence affirm what the robot's objective is. While a predictable motion is a more
"expected" or "unsurprising" motion, since it matches a person's expectation about how a robot will move
to achieve a certain objective. [13]
Figure 2.3: Predictable motion and Legible motion results, from [13]. On top comparative results on handwriting using predictable motions (on left) and legible motions (on right). On the bottom part, the comparative results when grasping an object, using predictable motion (on left) and legible motion (on right), with the bars showing the person’s level of confidence regarding the robot’s objective.

2.2 State of the Art

The area of collaborative object manipulation between humans and robots has some considerable works done, despite the fact that it is recent. However the focus has been mostly on making the robot follow the human’s lead and the focus of this work is the opposite. So our focus will be on works where the robot takes a more leading role.

We start with the used technologies in motion decision, presenting the most recent and most used technologies; proceeding to review some of the most recent considerations about safe interactions between people and robots and finally explore the most recent breakthroughs in the expression of intention through movement with robots leading the task.

2.2.1 Motion Decision Techniques

Motion decision to robots is, currently, mostly done in one of two ways, either using motion planning or using motion learning through demonstration.

Motion planning is based on the idea of a robot planning and optimizing the sequence of actions in order to move to achieve a certain goal. Various approaches to motion planning exist, but there are two
types that have shown to be extremely successful in manipulation problems. One type is sampling-based motion planning algorithms and the other are optimization based planners. [32]

Sampling-based planners are planning algorithms that construct graphs between sampled points randomly and answer the queries with the shortest collision-free trajectory that connects the initial state and the objective state. [33, 63] Optimization-based planners are sampling planners that instead of just finding the shortest trajectory also use extra knowledge from heuristics to find an optimal path according to defined cost functional. [33, 63]

Learning from demonstration, on the other hand, works by using a set of demonstration trajectories performed by a human teacher to teach the robot how to move. [3] Traditional LfD techniques work with demonstrations in the form of continuous trajectories, where the start and end of the trajectory are explicitly demarcated by the human teacher. Recently another type of LfD has developed, called "Keyframe Learning from Demonstration (KLfD)" where the learning process instead of using continuous trajectories, uses discrete series of points that describe the trajectory, allowing for a more precise control of the robot’s pose than when executing the task from start to finish. [49]

In LfD the demonstrations can be done either by using motion recording systems that capture the teacher’s movements or by executing the trajectories with the robot and commanding it to record them. [3, 49]

From the methods and approaches mentioned in the previous paragraphs, there are some that are especially important for our work, since they show very good results when applied to collaborative manipulation tasks and scenarios. These are the Stochastic Trajectory Optimization Motion Planning (STOMP) [32] and Covariant Hamiltonian Optimization Motion Planning (CHOMP) [52, 63] algorithms and the Interaction Primitives (IP) [2] and Probabilistic Movement Primitives (ProMP) [36] models used in LfD.

CHOMP and STOMP are similar approaches in the sense that both allow to generate a trajectory by optimizing trajectories obtained in previous iterations that are valid for the given problem. However these two approaches differ in the way used to derive and optimize the trajectory from the space constraints.

IP and ProMP are two alternative models for representing robot movement used in LfD. They differ in the fact that IPs are based on Dynamic Motion Primitives (DMP)¹ and ProMP are based on Movement Primitives (MP)².

We now discuss these approaches in greater detail.

¹A DMP is a parameterized representation of a trajectory in the form of a dynamical system. The parameters of the system correspond to parameters of the trajectory; by changing these, the trajectory can be easily modulated to different endpoints, velocities, etc. while essentially retaining its shape. [30]
²Compact parameterizations of the robot’s control policy, allowing imitation and reinforcement learning. [50]
2.2.1.A Covariant Hamiltonian Optimization Motion Planning

CHOMP, [52], is intended to produce high quality trajectories for robots with many Degrees of Freedom (DOF), these trajectories being both smooth and collision free. CHOMP is based on an iterative optimization algorithm that picks an initial workspace configuration and tries to obtain the intended final workspace configuration. In each iteration it tries to obtain a trajectory with the minimal energetic cost for the robot’s joints and that minimizes collisions, by using an objective functional that considers both the workspace obstacles and the movement smoothness, penalizing both movements that pass too close to an obstacle and movements that have high changes in velocities and accelerations. With this it is possible to reduce jerkiness in the robot’s movement and also create slower and smoother movements, safe from obstacle collision.

The optimization procedure uses a steepest descent algorithm with a gradient descent to find the minimum value of dispended energy. The gradient descent uses the functional gradient for the update rule, which tries to make small adjustments to the acceleration of the movement instead of simply making small changes in certain parameters that define the movement, allowing the update rule to be a function of the trajectory itself and not of the representation used for the movement.

In [63], Zucker et. al. introduced the Hamiltonian Monte Carlo (HMC) algorithm to perturb the trajectory when it reaches a minimum or another value where it gets stuck. This “push”, instead of a random restart, allows the algorithm to explore other more promising configurations instead of just randomly start over, which would not be certain to result in better configurations.

2.2.1.B STOMP: Stochastic Trajectory Optimization Motion Planning

STOMP, [32] by Kalakrishnan et al., is an algorithm for motion trajectory planning, similar to CHOMP, where, in consecutive iterations, the obtained trajectory is optimized to create a smooth trajectory without collisions with possible obstacles in the workspace. Like CHOMP this can be applied to robot movement around a workspace or just to the movement of limbs of a robot to perform specific tasks.

STOMP optimizes the trajectory generated using derivative-free stochastic optimization methods. This optimization methods allow to optimize trajectories with arbitrary costs, which were not possible using gradient optimization when the costs were non-smooth or non-differentiable. This problem is resolved by using a gradient estimation, based on [12] and [57], that does not depend on the trajectory rollout.

STOMP employs stochastic behaviour in the trajectory generation to explore possible trajectory configurations, controlling this exploration in order to prevent the system from making unexpected jumps to unexplored parts of the state space. With this behaviour the optimization procedure does not get stuck, always following similar configurations decreasing motion jerk.

In each iteration, the planned trajectory is updated trying to minimizing its cost. At the beginning of the
iteration, a set of new noisy trajectories is created and for each trajectory the algorithm computes both the cost and probability associated with each trajectory. With these values, the trajectory parameters are updated, using the probability-weighted convex combination of the noisy parameters for that iteration, calculated using the probability of generated trajectory for that iteration. By combining the sampling of new trajectories with the updating of the trajectory parameters, STOMP generates a smooth trajectory and optimized according to the selected performance criterion.

The algorithm stops when there is a convergence of trajectory costs, the costs stop to show big changes and the resulting trajectory is a combination of all the random noisy trajectories generated and evaluated. This algorithm has another advantage, the fact that the only open parameter is the magnitude of the expectation noise which reduces the complexity of the calculations needed in the optimization.

2.2.1.C Interaction Primitives

IPs applied for HRC first appeared in [2] and have since been used in the teaching of movement through demonstration or imitation.

IPs are a model to represent a robot's movement from recoded trajectories, using DMPs that can be chained to create more complex movements.

A DMP is a representation of a human movement, encoded using a dynamical system that describes it. This representation can afterwards be used to generate different variations of the original movement, allowing a robot to generalize that movement to new scenarios. Formally, a DMP is written as:

\[ \ddot{y} = (\alpha_y (\beta_y (g - y) - (\dot{y})/\tau)) + f(x)\tau^2 \]

where \( y \) is the joint angle controlled, \( g \) is the goal state, \( \tau \) is a time scaling factor, \( \alpha_y \) and \( \beta_y \) are constant coefficients that make this system critically-damped and \( f(x) \) is the forcing function \( f(x) = \phi(x)^T w \), with \( \phi(x) \) Gaussian basis functions and \( w \) the corresponding weight vectors.

The difference for normal DMPs is that IPs originate a generalized model that can apply DMP to human-robot interaction scenarios, allowing the robots to engage in interactions similar to the trained ones with a human partner. An IP can be formally regarded as a DMP which represents a joint activity of two interaction partners [2], correlating the robot's actions with the human's instead of what normal DMP does that is generating robot's actions that consider only the robot's movement.

In order to properly react to the movement of the observed agent, the IP performs three steps:

1. **Phase Estimation**, the system matches the observed movement with the recorded demonstrations using Dynamic Time Warping (DTW) - a dynamic programming algorithm that matches time series, finding the optimal correspondence between data points.

2. **Predictive DMP distributions**, the system tries to predict the other agent's behaviour by creating distribution models over the parameters the DMPs and uses those models to extract the most
likely parameters of movement to complete the observed movement.

3. *Correlating both agents*, the system correlates the movement of both the robot and the observed agent, obtaining the movement parameters that best correlate both agents.

In order for the prediction distributions to also estimate the target position of movement, aspects as the shape parameters and the goal attractors for all DOFs are included in the distribution model for each DMP.

The correct correlation of both the robot and the observed agent allows the collaboration task to be successful and also to predict how the observed agent will move during the rest of the task.

### 2.2.1.D Probabilistic Movement Primitives

ProMPs applied to HRC tasks were first described in [36] and were intended as an alternative to IP. The application of ProMP to collaboration tasks aims at making the robot’s movements spatially and temporally correlated with the human’s movements, instead of just focusing on the robot’s movements like in [2].

ProMPs, were presented in [50] by Paraschos et al., as a probabilistic formulation of movement primitives that allowed properties like co-activation, adaptability, optimality and modulation of the movement. With this formulation, the ProMP allow to model the time-varying variance of the trajectories to capture multiple demonstrations with high-variability. Considering a DOF as one joint or Cartesian state of a human or a robot, at each time step $t$ each DOF is represented by its position $q_t$ and velocity $\dot{q}_t$. So, for each DOF, we denote $y_t = [q_t, \dot{q}_t]^T$ and a trajectory as a sequence $\tau = y_{1:T}$. A compact representation of the trajectory is accomplished using a weight vector $w$ and Gaussian basis functions $\psi$, resulting in:

$$ y_t = \begin{bmatrix} \psi_t \\ \dot{\psi}_t \end{bmatrix} w + \epsilon_y = \Psi_t^T w + \epsilon_y $$

and the probability of observing trajectory $\tau$

$$ p(\tau|w) = \prod_t \mathcal{N}(y_t|\Psi_t^T w, \sigma_y) $$

where $\Psi_t = [\psi_t, \dot{\psi}_t]$ defines the $n \times 2$ dimensional time-dependent basis matrix for the joint positions $q_t$ and velocities $\dot{q}_t$, $n$ defines the number of basis functions and $\epsilon_y \sim \mathcal{N}(0, \sigma_y)$ is zero-mean i.i.d. Gaussian noise. The speed of execution of the movement is decoupled from the original movement by adding an artificial "clock", known as phase variable $z$. The phase variable replaces the time in order to control the basis functions $\psi_t$, allowing to synchronize multiple DOF of the same robot or of multiple agents (robot or humans), by defining that $z_0 = 0$ and that $z_T = 1$. Although most works have used
\(z(t) = t\) \([18,19,36,37]\), any monotonically increasing function can be used as long as it respects those constraints regarding \(z\). \([50]\)

In general, ProMPs are learned from multiple demonstrations, which reveal the variance of the task, the uncertainty of the execution, as well as to introduce exploration noise when required. So, the model captures these trajectory differences by defining a distribution over the trajectories’ weights, \(p(w|\theta)\), where \(\theta\) is the learning parameter that captures the correlation among the weights within the trajectory and between demonstrations of the same DOF. With this, each trajectory’s probability is given by

\[p(\tau|\theta) = \int p(\tau|w)p(w|\theta)dw\]

The application of ProMPs to collaboration tasks is done by extending the weight vector to consider both the observed agent - task partner - and the controlled agent - robot - resulting in a vector like:

\[w_d = \{[w^t_t, ... w^p_t], [w^t_q, ... w^q_t]\}\]

where \(w_d\) is the augmented weight vector for the \(d\)-th demonstration, with \(w^p_t\) being the \(n\)-dimensional column vector of weights of the \(p\)-th DOF of the observed agent and \(w^q_t\) the vector of weights of the \(q\)-th DOF of the controlled agent. With this augmented weight vector, the properties of a ProMP like movement modulation or co-activation can be used for collaboration tasks.

### 2.2.1.E Influences in Motion Decision

In motion decision this work will build on several works, mainly from Marco Ewerton \([18,19]\) and Heni Ben Amor \([2]\).

Ewerton’s and Amor’s works, \([2,18,19]\), have introduced multiple ways of integrating DMPs and MPs in motion learning directed towards HRC tasks. These works describe applications of models that show great promise in the field of LfD, IP and ProMP, both described previously (see sections 2.2.1.C and 2.2.1.D). Both have also contributed with improvements to those models, as well as with frameworks and studies on how to use them in collaborative tasks.

In \([18]\), Ewerton and Amor et al. present an improvement to the basic IP framework that only considered a single interaction pattern, instead of the usual occurrence on interaction between humans and robots, where the interaction consists on many different patterns combined to perform the task at hand. This way they propose a combination of IP and ProMP, called Interaction Probabilistic Movement Primitives (IPMP). This new framework combines IPs using probabilistic theory, like in the case of ProMP, and learns a model of how to mix different IPs in order to perform a full collaboration task with different parts and patterns of interaction. Thus presenting a framework that is capable of predicting human action and of adapting to changes, both on the task executed and on the workspace configuration.
In [19], they show how robots can learn models of spatio-temporal variability to adapt to human changes in space occupancy and speed, changes in the environment, etc. They use IPMP complemented with a probabilistic model of how humans occupy and move in the shared space to avoid collisions with humans. This work shows a useful framework for collaboration tasks, because with such a probabilistic model a robot is capable of not only adapting its movements to the humans, but also predict how they will move and decide its movements accordingly.

Maeda et al. presented in [37] a new framework based in the ProMP, the Collaborative Probabilistic Movement Primitives (CoPMP). This framework allows both action recognition and human-robot coordination, which is extremely important in collaborative tasks because it allows a robot to better decide the movements to execute. Also, it allows the robot to identify that the task has changed, which gives it more flexibility and adaptability.

The advantages given by the CoPMPs make them a powerful tool to be used in movement decision by learning, where collaboration in close-quarters is required. The fact that this framework allows for a new trajectory to be defined using a small set of samples and for the movement to be conditioned to multiple positions or only to a specific point, makes the CoPMP a framework well suited for this work. Also, given that CoPMPs use information regarding the human partner's pose and allow the robot to quickly adapt to a human blocking its movement, the collaboration efficiency increases.

The approaches presented in the last paragraphs - IPMP and CoPMP - give diverse alternatives for a robot's motion decision using learning approaches. They were developed specially for collaboration purposes and show great promise for a work like ours, where conditions may change unexpectedly. Another aspect that makes IPMP and CoPMP good frameworks for our work is that they try to predict and/or recognize the actions of the other members of the collaborative task, allowing a better adaption to the humans.

Approaches based on ProMP and IP have been tested in multi-task and in multi-agent scenarios. However, the results are not significant enough to ensure that these frameworks will work in all of these scenarios, without extra considerations about time of response and quality of the solution given. The fact that in our study we will need keep track of multiple agents moving in unison, adds extra difficulties because the system has to consider more perceptions in the decision process and the solution may have some noise.

### 2.2.2 Safe Human-Robot Collaboration Interactions

Safety during interactions in HRC tasks is one of the biggest concerns in HRC research, mostly because robots can cause harm to humans during interactions, both physically as psychologically.

One aspect that has always been much researched is how to design a robot, not only at the hardware
level but also at the software control level. [27, 42] These design considerations have contributed to the creation of design patterns and guidelines based on results from studies, as much as directives and laws about robot utilization and operation, which advise about how to design both the hardware and the software for robots involved in collaborative tasks. [42]

In order to make people feel safe during an HRC, it is important not only to teach the robot how to move safely and without causing harm to humans, but also there must be a clear understanding of who shall act and when and the roles of each participant. That is what usually happens during a collaboration task, one of the participants does one task and the other does another task, when the tasks require a division of labours; or if the task requires both to perform the same action, there is a coordination of efforts to perform it. [48]

Usually in collaboration tasks there are two popular paradigms: “leader-follower” and “equal partners”. In leader-follower one agent drives the progression of the task and the others follow the instructions and in equal partners the leading role is shared by the agents intervening, switching the leadership between them. [27]

This knowledge of each one’s roles is extremely important for the robots engaged in such tasks to know what to do and also to predict the co-worker’s intention. Maintaining situation awareness relative to the task progression can lead to safer interactions during an HRC task.

As Hinds, Roberts and Jones concluded in [28], people tend to rely more in robots when they are more similar to a human than to a machine, giving fewer justifications about their actions and trusting more on them to fulfil their part of the task without such careful supervision. This trust and reliance shown by the participants towards the robot, during the task execution, improve the safety because it shows that the human participant feels safe working with the robot and sharing the space. This feeling of safety makes the interaction safer, since there is less hesitation by the human and the robot can better predict the intention and movements of its partner.

### 2.2.2.A Influences on safety in HRC interactions

In this section, we analyzed several works on how to make an HRC interaction safe without affecting its efficiency and fluidity. This work will then build on Hinds, Roberts and Jones [28] regarding their conclusions on how more natural interactions and reactions by a robot makes people trust more on the robot doing its part of the work, which in turn influences the task’s performance.

The works of Hayes [27] and Mutlu [48], also have a high impact on our work given that they present important notions for collaboration tasks that sometimes are overlooked in Human-Human collaboration tasks. Those notions of roles and who leads the task are important for this work. Because not only the task progression is more fluid, but the robot knowing when to act and when to stop and wait for its turn, gives the task a more natural feel. This fosters, in the human participants, more trust in the robot and a
feel of more at ease with it, allowing them to better understand the robot's intentions.

### 2.2.3 Transmission of Intention Through Movement

One important aspect in any collaborative task, either among humans or between humans and robots, is the easy understanding of intention without the need to ask the other what they are doing. [54] This aspect lead to the study of how to convey intention through the way we move during the execution of a task, the way we approach an object or our pose while we perform a task.

There have been various works about how to make robots express the intent of a movement more clearly while they perform a task. [7,23,31,54] Most of the results in these studies are based on animation and creating human-like movements based on animation principles, such as [23]; and on the study of how people physically communicate intention prior to giving an object to another one or to a physical interaction [54].

The most recent studies have shown that different ways to create a trajectory yield different results in the perception of motion intention by a human partner. They show that a movement that is purely efficient, where efficiency is concerned with energy usage and collision avoidance, is not always the best movement to convey intent and can sometimes scare people.

A better way to transmit intention was shown to be movements that are not purely efficient, but that encompass a certain smoothness and also an earlier deviation towards the motion objective, allowing this way for the human partner to understand more easily what the robot is doing and why, and react more accordingly.

These conclusions are derived from works such as [13,15,17,24], that show the benefits of designing trajectories of movement more focused on the target instead on just the efficiency of the movement. In [15], Dragan et al. show the effects of predictable and legible motions in collaborative tasks and how they impact the perception of people about the objective of the robot's movement. They show that when the human partner does not have previous knowledge about the robot's target, the legible motion is better at conveying the movement's intent; but that when the human has an idea of what the robot's objective is, the more predictable motions are better.

### 2.2.3.A Influences in Intention Transmission

The works of Anca Dragan et al., [13,15,17], will serve as basis for this work. The motive for this impact is the fact that only now the roles in collaborative tasks have switched, having the robots gone from followers to leaders and few works explore the impacts in collaboration of this paradigm.

In these works, Dragan has presented how the concepts of *legibility* and *predictability* can be used to make the robot's movements more expressive and how to create movement trajectories with these characteristics. [13,17] Apart from presenting these notions and how to create such motions, in [15],
Dragan et al. show that these motions are very good at conveying intentions in situations where users have little prior knowledge about the robot’s intentions. These findings provide a good basis for work on intention transmission in collaboration tasks, where people have little knowledge about the robot’s intentions and have to infer those same intentions. So they prove to be adequate for our research on the impacts that movement has on multi-user human-robot collaborative manipulation tasks.

The findings of Gielniak et al. [23, 24] and Takayama et al. [56] about generating human-like motion using animation principles and creating anticipation about the objective are important for the generation of movement in a collaboration, because they allow the movements to be more expressive. The combination of these findings with the definitions of legibility and predictability of [13], are strong basis for the resulting system to be expressive in its movements and increasing the task performance.
3

Multi-User Cup Filling System

Contents

3.1 Vision Module ................................................................. 28
3.2 Movement Decision Module .............................................. 35
3.3 Social Interaction Module ................................................. 43
This work focuses on a robot filling cups to multiple people and study the impact of the robot's motion on people's understanding of the robot's objective. To reach that end, we developed a system that integrates the main conclusions of chapter 2, combining the flexibility of motion decision using learning with intention transmission using the notions of legibility and predictability to create safe interactions. Also, given that naturalness in the interaction improves people's trust in the robot, we developed a social component so that the robot was capable of a more natural interaction. The system developed is comprised of three modules, described in Figure 3.1:

- vision module - receives the data from the kinect camera and processes the information in order to identify the targets, the cups in our case;
- movement decision module - decides the next target and how to move towards it in an expressive manner;
- social interaction module - interacts socially with the humans making the task feel more natural, and safe, and the interaction more fluid.

By observing Figure 3.1 it is possible to understand that the Movement Decision module is the central module in our architecture. This happens because all the robot's behavior is dependent on cup to be filled and the movement the robot performs. Because of this, the movement decision module is central to the architecture and as we will show next, it is also the module where all reasoning is done.

The communications between different modules are done by trading messages using a publish and subscribe approach, allowing the modules to work in parallel. However, communication inside one module is done mostly using services that force the module to follow a given sequence of actions and
so guarantee the correct module operation. Both the publish and subscribe and the service systems use the Robot Operating System (ROS) framework\(^1\) - a framework composed by a collection of tools, libraries, and conventions that allows for people to create complex and robust robotic behavior across several platforms and programming languages.

### 3.1 Vision Module

The vision module in our architecture uses data from a Kinect V2 camera mounted on top of the robot’s head to identify the positions of the cups in the workspace.

The vision system used in this work is very simple:

1. a Kinect V2 mounted on the robot’s head sends color images and IR images using a collection of tools and libraries for a ROS interface, called IAI Kinect2 [62];
2. the images received are processed to identify the various cups present in the scene;
3. finally the information regarding the cups’ positions is published with the position of each cup.

#### 3.1.1 Kinect V2

The Kinect V2 was used in this work to get information about the workspace, namely about the position of the cups for the robot to fill. In our work we used a set of tools and libraries available for ROS that bridge the Kinect camera with the ROS system, called IAI Kinect2 [62].

The IAI Kinect2 serves as a bridge between the data recorded by the camera and the ROS system running by receiving the color image and IR image data and publishing it in different ROS topics. The images published in the topics are both the raw images recorded and the images after being processed to correct distortions provoked by the camera. Besides this division in corrected and not corrected images, the images are also published under different topics for different resolutions of image, the resolutions supported are Standard Definition (SD) (512x424), quarter High Definition (HD) (960x540) and full HD (1920x1080). Since the IR and color cameras have different native resolutions - the color camera records in full HD and the IR camera records in SD - the bridging application has to convert between resolutions to be able to publish in the available resolutions, which creates some errors in the topics that do not correspond to the native images.

In order for the information of IR depth images and color images to be used correctly and also combined, the cameras need to be calibrated. The IAI Kinect2 also offers a set of tools to do this calibration, allowing to calibrate only the color camera, only the IR camera or to calibrate the transformations from

\(^1\)www.ros.org
one camera to the other in order to combine the data of both. This processes of calibration of one camera generate one file per camera with the constants to correct the distortion provoked by the calibrated camera; the process to calibrate both, generates the same files as the individual calibrations but creates an extra file with the constants for the projection of one pixel in the depth camera to the corresponding pixel in the color camera.

As showed previously, the calibration process is important because the cameras are placed in different points of the Kinect and also do not have the exact same alignment, as it is seen in Figure 3.2. This mismatch of position and alignment prevents the directly conversion of a pixel in a color frame to a pixel in an IR frame. This way to obtain the same point in the other frame the point in one frame needs to be projected in the other frame.

Besides the projection that has to be done to get a color point in the IR camera, or the inverse, there also are differences in the Field of View (FOV) of each camera, as presented in Figure 3.3. This way, after projecting a point from one camera frame to the other, the resulting point has to be verified to check if the camera can see that point, sometimes this is straightforward because one or both the point’s coordinates are negative or with values that exceed the maximum valid for that image. However, when we use the images that already rectify distortions created by the camera, more complex checks need to be performed to guarantee that the same point is visible in both cameras.

### 3.1.2 Color Segmentation

Color segmentation is a type of image segmentation to extract information about objects from an image. Color segmentation gives better results than monochromatic image segmentation because it allows to use more information about the objects than just the light intensity. Color segmentation is very useful for applications in pattern recognition and computer vision. [10]
There are various methods to do color segmentation and to process the results of segmenting images in order to enhance the results obtained. Currently one of the more used free libraries to do image processing is the OpenCV\(^2\) library, a library released under BSD for both academic and commercial uses. This is a very simple to use library, with support for C/C++, Python and Java applications and runs on most operating systems. We use this library in this work to perform the color segmentation of the images, because it offers a simple interface, has seen a widespread use in computer vision applications and because it is a library that gives good results very quickly.

There are various methods to identify objects in a scene and color segmentation is one of them. Other methods that can be used are: simple image segmentation to find an object by its contour or using machine learning to train a classifier to recognize a specific object like in [51].

In this case we chose to use color segmentation because it is a method simple to implement and that gives good results and that does not need large banks of data to train like the case of approaches based on machine learning or that can fail the object recognition because the contour was not correctly identified. Besides this, color segmentation also gives advantages that other methods do not. By using color the system can associate with each object detected more specific information than if the identification was based on the shape of the object, information that can later be used to check if a specific object was already dealt with, or in our case if a cup has been filled.

However the use of color segmentation also restricts a little the system’s flexibility, because if the target does have a colour that the system expects then it is not identified and is ignored. This restriction does not cause a great problem for our work, because in this work we will only use a small number of cups, for which the expect colors can be easily loaded into the system.

The color segmentation process can be done by using one the various color spaces existent, since

---

\(^2\)http://opencv.org/
OpenCV library we use has support for all the color spaces. In this work the color segmentation is performed using the HSV color space. HSV stands for Hue, Saturation and Value and in this space a color is defined by its hue that is the pure color to which a color refers to, by its saturation that describes how white a color is an by its value that describes how dark a color is. [53]

The HSV space is very useful to use in color segmentation because each color is defined by a specific hue and the saturation and value only affect how bright or dark that color is perceived, as such one needs only to find a correct hue interval for the color segmentation to work, since the saturation and value of a color are influenced by the light falling on the objects and not by the intrinsic color of the object. This way by giving a saturation and value intervals that exclude shades close to white or close to black, one needs only to focus on the color hue and find the smaller interval to identify an object.

For the system we developed in this work the color segmentation is performed in three steps:

1. conversion from RGB color space to HSV color space;
2. elimination of image objects that are outside the intervals of hue, saturation and value defined;
3. elimination of noise remaining after the segmentation.

The first two steps of the segmentation process are done recurring to functions available in the OpenCV library that convert images between color spaces and that remove elements of an image that are not within given intervals in color space. The intervals for the segmentation were determined empirically for each of the cups that we use in the experiment in order to eliminate as much of the other objects in the scene as possible.

The last step in the segmentation is done by applying two transformations to the resulting segmented image in order to remove small objects and other artefacts that remain from the segmentation. This removal makes the final image easier to process in order to find the intended object’s center of mass, because the image after this removal rarely has more objects remaining than the object we intend to extract from the image and this way the center of mass is easier to determine. Figure 3.4 shows the results of the segmentation of a scene with the three cups that have to be recognized, as it can be observed the segmented images - presented on the bottom - only have the cups and the rest of the image is in black.

In order for the removal of noise to work properly, it is needed to define the minimum size an object can have to be considered not noise and the shape of that object. The shape used was that of a rectangle, because of the shapes available by the OpenCV library that was the one that better fitted our target - the cups - and because the process to remove noise is based on the idea of finding patterns in the image that have a different shape and smaller size than those defined and remove them, then using a rectangular shape was the best guarantee that the cup was not discarded as noise for not matching the expected shape.
3.1.3 Cup position identification

The identification of the position of each cup is the most important task that the vision module has. In order to correctly identify the position of each cup after the color segmentation, the information in the color image must match the information in the depth image. This, so that when the center of mass of each cup is determined, it is easy to obtain the 3D position in meters of the cup in relation to the robot. With the correspondence of both images and the centres of mass determined, all that is left is to compute the 3D positions based on the depth information and publish those positions.

3.1.3.A Depth to Color Mapping

The depth frame and color frame correspondence is then the first step in the cup identification process. In order to quickly make the correspondence between the center of mass of a cup in the color image and the corresponding position in the real world in relation to the robot, we need to know at which distance the point is from the robot. So, in our system we decided that it is wiser to map the depth frame into the color frame and use that mapped data to get the depth of a cup’s center of mass and consequently the 3D position of the cup. This since both images do not show exactly the same data and the conversion is easier to do from the depth image to the color image.
In order to do that mapping, we go through the entire depth frame and for each pixel: first the point is converted into its 3D real world position; then the 3D point in the depth camera is transformed into the same point in the color camera and finally the 3D color point is converted in camera pixels. The expressions used for this conversion were adapted from [8].

\[
\begin{align*}
P^d_x &= (x^d - c^d_x) \times \text{depth}(x^d, y^d)/f^d_x \\
P^d_y &= (y^d - c^d_y) \times \text{depth}(x^d, y^d)/f^d_y \\
P^d_z &= \text{depth}(x^d, y^d) \\
P^c &= R^c_d \times P^d + T^c \\
p^c_x &= (P^c_x \times f^c_x/P^c_z) + c^c_x \\
p^c_y &= (P^c_y \times f^c_y/P^c_z) + c^c_y \\
p^c_z &= P^c_z
\end{align*}
\]

where \(x^d\) and \(y^d\) correspond to the pixels in the IR image, \(P^d\) corresponds to the 3D world position in the IR camera space corresponding to the pixels. \(P^c\) is the 3D world position in the color camera space and \(p^c\) is the corresponding point in pixels in the color image. The \(c^d_x, c^d_y, f^d_x, f^d_y\) constants are the intrinsic constants for the depth camera determined during the Kinect calibration; the \(c^c_x, c^c_y\) are the x and y coordinate for the camera center in a frame from the IR camera and the \(f^c_x, f^c_y\) are the focal distance constants for the IR camera.

The first three expressions are used to project a point from the IR camera pixel space to the camera 3D metric world space coordinate. This transformation is a needed step, since the calibration constants are all for transformations between points in 3D metric real world coordinates.

The forth expression is the projection of the 3D point from the depth camera space into the color camera space. This is a simple matrix transformation between different coordinate systems, with the rotation matrix (the \(R^c_d\) in the expression) giving the rotation of one coordinate system in relation to the other and the translation vector (the \(T^c\) in the expression) giving the relative translation of the center of one coordinate system to the other. Both the \(R^c_d\) and the \(T^c\) constants are obtained by doing the extrinsic calibration of the Kinect system and are for transformations from the depth 3D metric coordinate system to the color 3D metric coordinate system.

The final three expressions mirror the first three, but in this case the projection is from the color 3D metric real world coordinate system to the color image. As previously the \(c^d_x, c^d_y, f^d_x, f^d_y\) are the intrinsic constants for the camera center and focal distance and are determined during the intrinsic calibration of the Kinect’s color camera. The last expression normally would not be important if the objective is to find the color pixel that corresponds to a specific depth frame pixel. However, in our case, we are interested in storing a depth mapping so that after the center of mass of each cup is calculated it is easier to obtain
the real world position of that cup.

The mapping of the points in the depth camera to the color camera has one last consideration besides projecting the point from the depth camera coordinate system to the color camera coordinate system. This consideration is that the two cameras do not have the same FOV, in the case of the depth camera the FOV is $70.6 \times 60$ and in the case of the color camera the FOV is $84.1 \times 53.8$. Due to those differences after the projection of a point between cameras, a check about whether or not the point is in view in the destination camera is performed. In the system developed, the check is performed only by checking if the resulting $x$ and $y$ coordinates for the point in the color image are within the image size limits - between 0 and 1980 for the $x$ coordinate and between 0 and 1080 for the $y$ coordinate. This happens because we use rectified images, which already correct distortion problems caused by the camera. This check could in fact be simplified and only be performed over the $y$ coordinate, since this is the only dimension where the depth camera records more information than the color image, however as a safety measure against possible errors we chose to also verify the $x$ coordinate and guarantee that the point is valid in both axes.

3.1.3.B Object Center of Mass Determination

With both the image segmented by colors and the depth information mapped into the scene visible by the color camera, it is possible to identify the cups and their respective centres of mass. This process is done in two steps, the first is to identify the object's shape and its center of mass and the second is to calculate the 3D coordinates in meters for each center of mass.

In the first step we first determine the cups contours by using a function available in the OpenCV library that finds the contours of every object present in an image. When the segmentation for each color returns an image with only the cup, the contours function returns only the contours for the cup. However, when the segmentation returns the cup and some small objects, resulting of noise in the image processing, first we have to identify the correct contours. That identification is done by calculating the area of each object identified and the object with bigger area is considered to be the cup. Since the segmentation was optimized for each cup it is not probable that another object with the same exact combination of HSV values is presented in the image.

After the contours of a cup are obtained, then the center of mass for the cup is found by using the concept of image moments. Using a function available in the OpenCV library, the moments of a specific object in the segmented image are determined. With those moments we determine the center of mass, which the center weighted point of the object limited by the contours. In Figure 3.5 it is possible to observe the result of determining the centres of mass using this method.

---

3. An image moment is a particular weighted average of pixels' intensities, useful to determine image properties such as an object's area or centroid. [59]
Finally, having the centres of mass for each of the cups, the 3D position in the real world is determined by getting the depth corresponding to the point in the mapped depth frame and re-projecting the $x$ and $y$ coordinates of the center of mass using:

$$p^c_x = ((x - c^c_x) \times P^c_x) + f^c_x$$
$$p^c_y = ((x - c^c_y) \times P^c_y) + f^c_y$$
$$p^c_z = P^c_z$$

The $z$ in the previous expressions is the depth value stored during the mapping process.

After the positions in meters are determined, their values are published with each $x, y, z$ being associated to the identification of the corresponding cup.

### 3.2 Movement Decision Module

The Movement Decision module is responsible for deciding the next target of the robot’s movement - in our case the next cup to fill; the movement trajectory performed to reach the target and coordinate the robot’s movement with the social interactions.

The module is composed by sub-modules: one is responsible for deciding the next target and communicating with the social interaction module and the second one is responsible for determining the best trajectory of movement to perform in order to achieve the target and executing it.

The managing sub-module, as responsible for deciding the cup to fill and for coordinating the social
interactions, communicates with the other modules in our system: it receives the information from the vision module regarding the cups’ positions and communicates with the social module. The communication with both modules is performed through ROS topics. The information received is processed to find new cups in the workspace. If the ones already there are no new cups, the information if updated if the positions changed significantly. Regarding the social interaction module, when changes to the facial expressions need to be done a message is sent with the new expression to be displayed.

The trajectory execution sub-module only interacts with the managing sub-module, since it does not need any information from the other modules nor to communicate any information to the other modules. The communication between the sub-modules is done using a simple ROS service, that the managing sub-module calls when there is a new target or the previous one has changed position significantly.

As one can conclude from [15], both predictable and legible trajectories have great advantages: legible trajectories are very good at transmitting intention in situations of ambiguity regarding the movement’s objective and predictable trajectories use less energy and time to execute a motion and, in cases where there is not great ambiguity between possible objectives, perform as well than legible motions.

However, each type of trajectory has some shortcomings: legible trajectories tend to take a lot of time, that sometimes is not needed, to better transmit intentions; predictable trajectories, by taking a more direct approach to move towards an objective, can lead to movements that may not allow other participants to adapt as quickly and make the collaboration less effective.

In order to minimize some of these shortcomings, while maintaining an efficient collaboration, we devised an approach that combines both predictable and legible trajectories and, given the layout of the objectives in the workspace, chooses which to perform. With this approach we expect to maintain the advantages of using legible trajectories in a collaboration task, but in occasions where the possible objectives do not cause ambiguity we use predictable trajectories in order to make the task more efficient both in time consumed and in energy expended.

3.2.1 Collaborative Probabilistic Movement Primitives

CoPMP are type of ProMP, a method to learn interactive movement primitives and extrapolate new movement trajectories. They are adapted to be used in collaborative and interaction tasks and are capable of adapting to task changes like the case of the IPMP [18]. CoMP! (CoMP!) combine the capabilities of the ProMP to mix different movement primitives to determine the best movement, with the capabilities of IP to coordinate the robot’s movements with the movements of the other task’s participants and the disposition of the workspace.

The CoPMP are presented in [37] and being based in the ProMP use the same reasoning process to learn the movement model. However, instead of focusing solely on correlating the different exemplified trajectories spatially and temporally, the CoPMP also focus on deducing a model that correlates the DOF
of the trajectories recorded for the robot and the human participants. Like with IPMP, in order to adapt to changes in the task being performed, CoPMP mix the different models generated for each task and then choose the movement to perform as the most probable across the different models. [37]

So, the CoPMP present an approach that combines the adaptability and capability of trajectory extrapolation conditioned to a single target of the ProMP with the interaction capabilities and trajectory extrapolation of the IPMP.

The CoPMP are thus capable of both predicting a human partner’s movements allowing the robot to better adapt to the humans’ movements. Also, the CoPMP allow the generated movements to correlate with the humans’ actions, which is important not only because it boosts collaboration like with ProMP but also by recognizing the task stage the robot’s system has more information and can generate movements that are more correct and that allow the participants to better interact with the robot.

In this work the CoPMP are used by creating the model based on the demonstrations given for each type of trajectory - a model is created for the predictable motions and one model is created for the legible motions - and then depending on the type of trajectory to be performed one of the models is used to extrapolate the trajectory. These models are created in the trajectory execution node and when needed the managing node calls a ROS service that based on the model and with a target passed by the managing module, generates a new trajectory and passes it to the robot for execution.

With the model created is easy to come up with the new trajectory by conditioning the movement to end in a specific target. If the movement to perform is towards a new target then the movement is determined by copying the created model and then conditioning it to the new target. The model copying is important because the conditioning process changes the original model and so the model used in subsequent movements would be different. However, if the robot is not trying to move towards a new target, but instead is correcting the movement, then the model used is the one already conditioned by the previous movement. The idea is not to find a new movement that will achieve a certain target but given the already performed movement adapt it to a new target.

For each motion type - legible and predictable - twenty trajectories were exemplified. For each motion type, people were asked to give a set of positions where they would hold the cup normally and we recorded the trajectories to move to that position.

The trajectories demonstrated for each type of movement follow the principles described by Dragan et al. in [13,17], but in our case we used kinesthetic teaching to teach the robot the movements, instead of using a planner that optimized the robot’s movements. The predictable motions were designed to be direct and non surprising: the example trajectories would mimic the movements one would expect to see when expecting to get his cup filled - as direct as possible and as fluid as possible. The legible motions, being movements “action-to-goal”, were designed to try to explicitly communicate the movement’s target. This was achieved by performing a more wide movement that would move away from the other targets
as much as possible and exclude them. This goes in line with the definition of a legible motion in [17], in which a legible motion is a movement that causes small confusion regarding what the robot is moving towards. This way, legible movement tended to be less directed, instead they described more of an arc that would go the opposite way of the other cups, while being natural - the right arm would tend to arc more to the right with rightmost targets than with leftmost targets and the left arm vice-versa.

To validate the demonstrated trajectories, we asked some people to come to the lab. These people were presented with a scenario similar with the one on the final experiment. In this scenario, instead of the people holding the cups, these were placed on top of a table. We then would execute a sequence of trajectories for both the predictable and legible trajectories and ask people to try to predict to which cup the robot was moving towards. At the end we also asked them to evaluate qualitatively which trajectories were easier to understand and which ones were more according to their expectations. With this validation we were able to conclude that the trajectories were in line with the design principles.

### 3.2.2 Target selection

Before the model can extrapolate the movement to execute the target must be determined. As explained previously, the CoPMPs accept targets that are either multiple points of reference or only a single point. In our work, we are only interested in moving to a cup, represented by the 3D position of its center of mass, so CoPMPs are a good choice for this work.

The decision process is composed by three parts:

1. keep the information regarding the cups’ positions updated;
2. identify when the robot has to select a new target, either because the last movement finished or because the current target has become unreachable;
3. decide from the remaining reachable cups the next one to be filled.

The first part of the decision process occurs in parallel to the other, the system is always listening for new positions of the cups. When new information is available the system checks if the position of any cup has changed significantly: more than 30cm in the X or Z axis, which is the dimension of the cups’ diameter, so if a cup moves more that 30cm to the left or right or 30cm closer or away from the robot that cup’s information is updated.

The second part of the decision process regards when to select a new target for the movement. The system selects a new target either if there is no target selected - this because the last movement has finished or an error has occurred during the movement - or if the last target has become unreachable and so a new target needs to be selected.

The identification of whether the movement towards a target has finished successfully or in an error is done by monitoring if the Joint Trajectory Action Server, explained later, returns an error code or
a success code during, or after, the movement has been completed. If an error occurs during the movement execution, then the system orders the robot to stop and move to a neutral stance before deciding on the next target.

Regarding the second condition that dictates if a new target needs to be selected, this happens when new data regarding the cups’ position is available. At the beginning of the decision process the system verifies if there is a current target and if it is still reachable or not. If the target is no longer reachable, then, before determining a new one, the system orders the robot to stop and return to a neutral stance and only afterwards decides the next target to move towards (which can be the last one if it has become reachable again).

The third part of the decision process is the target identification, this identification and decision of the target can be performed using a myriad of parameters. In our case, since the robot cannot move around a room to place itself closer to a participant, we chose to use as criteria for choosing the next target the distance from the robot towards each of the targets.

With the distance towards all the remaining targets computed, the system selects the next objective as the one closest to the robot. If there are two or more of these the system selects randomly between them.

After deciding the next target, the system communicates the new target to the CoPMP module that generates the trajectory and sends the trajectory to the robot to be executed.

The target selection, however is not as simple as just deciding when to select a new target and from the remaining targets choose the one closest to the robot. Some of the possible targets can be obscured by participants, the participants may have moved the targets away from the robot’s reach or the readings from the vision system may come with errors, even the objectives already accomplished must be taken into account so not to re-engage them.

These details must be addressed and dealt with when deciding on a new target. So, for every target in the list of objectives remaining to address before selecting one for analyzing the distance, the system checks if the target is visible and reachable by checking if the target’s distance is less that the robot’s arm reach. If the target passes both checks then it is checked if the target is the closest one. After the process is completed for one target it repeats for all the other targets in the remaining objectives list.

### 3.2.3 Connection with Baxter

The connection sub-module is composed by two parts: one which creates the CoPMP model used to generate new trajectories and the other that connects with Baxter’s Joint Trajectory Action Server and executes the trajectory generated using the CoPMP model.

This sub-module communicates with the movement managing sub-module through the usage of services. The services available allow for the movement managing sub-module to:
• ask for a new trajectory and execute it;
• execute a new trajectory by sending a trajectory to the connection sub-module;
• ask for feedback regarding the state in which a trajectory is;
• stop and/or restart the execution of a given trajectory.

The connection sub-module works as an interface between the movement managing sub-module and the robot itself. Thus, allowing for the manager sub-module to focus on selecting the targets and managing when social interactions. This way, the workload of the managing sub-module is less than it would be if the connection sub-module would not exist. As an interface, this node does not do much processing or computations besides building the CoPMPs models when it starts working.

As explained previously, in section 3.2.1, in our work the CoPMP are built depending on the type of trajectory being performed: if only legible trajectories are used then the model created is only for legible trajectories; if only predictable trajectories are used the model only uses predictable trajectories as base and if the hybrid approach is being used then a model for each type of trajectories is built. With the models built, every time the managing sub-module decides that the robot has to move towards a new objective, it calls the service in the connection sub-module that generates a new trajectory and executes it. When the movement finishes or an error occurs, the services returns an appropriate message to the managing sub-module besides returning control to it.

All of these services communicate with the robot using the Joint Trajectory Action Server (JTAS), this is a set of tools available in the Baxter SDK that eases the process of executing a trajectory with the Baxter Robot. The JTAS allows for trajectories to be sent to the robot as a sequence of positions and a desired time frame for the execution and then the Baxter’s CPU translates those points into joint values that are executed.

We use the JTAS because it simplifies the process of executing a trajectory using the Baxter’s arms, by only having to send the sequence of positions the joints need to have and the time at which those positions must be reached. The tools available by the JTAS also allow the system to track the movement’s progression as well as any errors or information regarding whether the movement is pending to be executed or had a problem during the execution.

The JTAS, as explained in the last paragraphs, sends messages regarding the state of the movement and also regarding the state of the end of the movement. The messages regarding the state of the action are sent when the JTAS is asked for feedback regarding the trajectory execution. These are used by our system in order to understand if the trajectory asked to be performed is executing well, if it had any problem starting or if it is waiting for the JTAS to finish another task before executing it.

Regarding the messages about how the trajectory execution has finished, these are sent the program that called the service either when a trajectory has finished or if an error occurred. The messages sent
may signify that the trajectory ended in a success or has been aborted or if another error has occurred. In our case we only consider the difference between success and an error, so all the messages that are not a success are treated as an error and our system considers that the movement failed. We adopted this simplification because the other results like the movement being recalled - cancelled by the user prior to start executing - or rejected - cancelled by the JTAS prior to start executing by the objective being unreachable - do not happen after the trajectory starts executing and as such do not matter in order to ascertain if the movement was successful or unsuccessful. This simplification also allows the response back to the manager to be binary, which makes the entire process of distinguishing successful movements from unsuccessful ones easier and faster.

3.2.4 Hybrid Approach

In this work we developed an approach allowing the robot to select when to perform a legible motion or when to perform a predictable motion. This choice is based on a set of rules created by observing human-human interactions in collaborative manipulation tasks, like filling cups with drinks, and also through the conclusions of Strabala et al. in [54] and [55] regarding how people approach situations of physical handover and manipulation of objects.

Allowing the robot to choose between performing a predictable motion or a legible motion gives an advantage to the robot in terms of performance against using purist approaches. Using this hybrid approach the robot is able to choose either to perform a legible motion or a predictable motion depending on the gains obtained for performing a more readable movement instead of a more natural movement, like when there is ambiguity between possible objectives and the robot performs a legible motion to overstate its objective or if the objectives are well spaced and using a more direct movement is better for the task. Besides these improvements, by analyzing the configuration of possible objectives in the workspace, the robot can also verify if performing a legible motion does indeed decrease possible confusion for the human partner, like in Figure 3.6 where the robot going for the leftmost cup, before the ones on the right, using a legible movement causes more confusion than if using a direct and predictable movement.

With these improvements we expect that the collaboration between the humans and the robot improves. This is motivated by the fact that this choice makes the robot's movements easier to understand by the humans, which makes the interaction more fluid since they will better understand the robot's objective. This faster understanding of the robot's intention, makes the humans’ reaction time shorter and the collaboration more efficient.

Finally in terms of safety, we also expect that it also increases because with a better understanding of the robot's intentions, people involved in the task can adapt to a robot's change in position, stance or movement and as such both the task's safety as well as the safety felt by the people involved are
expected to increase.

The decision between a legible and a predictable motion is done by analyzing the distribution of the objects in the workplace and the position of each objective relative to the others. This way, if the distribution is such that performing a legible motion towards the objective will lead to a better understanding of the robot’s objective, then the robot performs it, otherwise the robot executes a predictable motion.

For the task at hand, we designed a set of rules that guided the robot’s decision between the motion types. These rules were based on observations of human-human interactions and on the results from Dragan et al. [15]:

- if the selected objective is closely surrounded by two or more objectives, a more direct movement (predictable movement) is preferable than a more wide movement (legible movement);

- if the selected objective only has objectives on one side then a movement that approaches the objective from the side with no objectives is better than a more direct one, a legible movement is preferred to a predictable one;

- if the selected objective has other possible objectives close by, on the side the arm will approach from - example the robot is reaching for the leftmost cup with the right arm and three cups on the right of the objective - than a predictable movement is preferred to a legible one;

- if there is only one remaining objective or if there is no ambiguity regarding possible objectives then a predictable movement is preferred to legible movements.

To decide if an objective has other objects close by, in such a way that a legible motion would not be justified, we tested legible motions and found that for distances closer than 50cm between possible objectives, people would find a legible motion to be more confusing than a predictable motion. This way, we used this value as the minimal distance a cup could be from another on the arm’s side for the robot to execute a legible motion. Likewise, we tested the same legible motions with a target with other possible
targets for the robot on either side and concluded that on distances to the other objects less than 30cm (approximately the diameter of the cups used as objectives), a predictable motion would be preferred to a legible motion.

Even though this hybrid approach tries to combine both the advantages of predictable motions and those of legible motions it increases the time the robot’s system takes to decide the best movement to perform. This because it needs to analyze the environment in order to choose the best type of movement to execute before deciding what is the best movement to perform.

Currently the hybrid motion approach is based on a rule system tailored for the task of filling cups with drinks. However, this could be generalized to other tasks by either creating a set of rules that are common to multiple tasks - like preferring legible movements to predictable ones when reaching for targets from a side that has no objects - or by using machine learning techniques to teach the robot example scenarios where legible motions are preferred to predictable ones and the other way around. Although, the choice between executing a predictable or a legible motion, sometimes has to take in consideration aspects specific for the task the robot is going to perform. In those cases, the decision has to be more task-oriented and the system needs to have more strict rules to resolve those cases, reducing the flexibility of the system.

3.3 Social Interaction Module

In order for the interaction to be fluid and natural, we created a module that is responsible for interacting with the humans and display some emotions, like we do in human-human interactions. This way the robot’s “internal state” is more explicit and we maintain an “illusion of life”, keeping the participants engaged even when the robot is not moving.

So this module performs two main tasks: displaying emotions and state of mind through facial expressions displayed on a LCD on the robot’s head and asking for the humans to perform certain actions to better the interaction.

The display of facial expressions contributes to the naturalness of the task and for the people to trust more in the robot by emulating a human interacting with them. The facial expressions change in key moments of the task: like when the robot is deciding the next cup to fill and displays a face with the eyes moving and looking at the various cups; or when the robot correctly serves a cup and it displays a very happy face, like someone that excited for being helpful. Also, to contribute to the task’s naturalness, the module simulates the robot following its own movement by changing the eyes to always point to the end-effector that has the water bottle.

The interaction with humans through voice contributes both to the naturalness of the task by filling gaps where the robot is stopped with small talk and to the fluency of the task by asking the participants
to bring the cups closer or to raise them higher when the cups are out of reach.
User Study

Contents

4.1 Setup ................................................................. 47
4.2 Experience Design .................................................. 49
4.3 Procedure ............................................................ 51
4.4 Sample ............................................................... 52
4.5 Measures ............................................................ 52
This section describes the experimental evaluation of our system, in the scenario previously described, where the robot serves a drink to a set of users holding colored cups.

4.1 Setup

In our user study we used a Baxter robot for the collaboration task, with a Kinect camera mounted on its head to perceive the workspace and recognize the position of each cup.

4.1.1 Baxter Robot

Baxter is a robot developed by "Rethink Robotics" to serve in industrial settings or in research and education. It is designed to be safe to work around and with, since it has built-in security features to prevent it from continue moving once it hits an obstacle or if it is exerting too much force on an object.

Baxter, as can be seen in Figure 4.1, has two arms, each with 7 DOF, and a screen on the head. This screen can be used to display informations or to display faces or facial expressions that are used to make Baxter more human-like.

Baxter has a wide range of sensors:

- a 360° sonar and front camera on the head;

- each wrist has an integrated camera;
• each joint has sensors to determine torque, position and velocity.

Also it can be fitted with electrical parallel grippers or a vacuum cup gripper.

Besides the hardware specifications, Baxter utilizes an open source Software Development Kit built on a standard ROS framework, which allows for researchers to build a wide array of custom applications. These applications can be run from a connected workstation or locally through access to the on-board CPU.

4.1.2 Kinect camera

Kinect is a depth and color camera developed by Microsoft. The first version was intended to be used with its gaming console the Xbox, but since then Microsoft realized that the camera had potential to more than just gaming and so launched the second version that already has more image definition, both in depth as in color, has better precision in depth measuring and has a bigger field of view than the first version.

The Kinect camera is a widely used camera for robotic applications, because of its cost advantage relative to the results that it gives.

Although it is a fact that a motion tracking system is capable of more precise measurements of both position and on object tracking, the Kinect camera is a solution that is more affordable and also gives more mobility and flexibility of usage when compared to other systems.

Kinect also offers as advantage the fact that it has had a widespread use in robotics and as such there are lots of functionalities, on the form of libraries and tools, developed for ROS that allow for new applications to use those functionalities instead of having to create them again.

For our work we decided to use the Kinect version two (V2) instead of the first version. This choice is derived from the hardware improvements present in Kinect V2:

• Kinect V2 supports color images with 1920x1080 resolution and depth images with 512x424 resolution instead of the resolutions of 640x480 for color images and 320x240 for depth images supported by the first version [60];

• In the newer version the depth is measure using the time of flight that takes an IR beam to hit an object instead of using a structured light\(^1\) algorithm licensed by Microsoft and not publicly available. With this change it is easier to calculate the real world coordinates of a point without having to use proprietary software and libraries [35];

• The use of time of flight instead of structured light also allow the newer versions of the Kinect camera to be more precise in the measuring of an objects depth, which was another of the reasons why we chose to use this version.

\(^1\)Process where a known pattern is projected onto the scene and the depth is inferred from the deformation of that pattern.
For the experience we performed, the Kinect camera was mounted on Baxter’s head because the camera needs to be able to capture the cups held by the human participants and also because placing the camera this way allows the system and the training data to be used between different Baxter robots, since the camera is placed in the same place allowing for the data used in one robot to be used in another with few changes having to be done.

In order to mount the kinect on top of Baxter’s head we printed a support in a 3D printer. This support is not official from Rethink Robotics, it was developed by HumaRobotics\(^2\), a team from Generation Robots’, and aims at making collaborative tasks more natural and efficient by allowing the vision system to observe its environment without any obstacles hindering Baxter’s movements. This support is available for download and print at Generation Robots’ blog [6].

### 4.2 Experience Design

The user study designed simulated a cafeteria or restaurant scenario. In this scenario there is a waiter - the Baxter robot - which has the task of serving drinks to the people that approach it looking for something for a drink.

We chose this scenario because:

1. it is a collaborative task that can easily be focused on the robot’s movement and how that movement communicates intentions, which is the main objective of this work;

2. is a task where the humans engage the robot at the same time and as such there is a need to collaborate with the robot and with each other to finish the work quickly;

3. is a collaborative task which progression does not depend on the human team, but on the robot’s action and that makes the human team give more attention to the robot and to its movements;

4. is a repeatable task, where each participant is exposed to the same situation across the three testing conditions;

5. is a task that simulates a real world scenario where the participants would have to collaborate with the robot.

Figure 4.2 shows the positions of both the participants and the robot. As told previously, this scenario simulates a cafeteria or restaurant setting, where the robot waits for people to approach it and extend the cups as a sign of asking to fill them.

The participants’ positions were fixed and were decided considering:

\(^2\)http://www.humarobotics.com/
The scenario layout. The participants extend the cups as a sign to ask for water, as seen on the left, and the robot, on the right, proceeds to serve them one at a time. The participants are told, when given the correspondent cup, to place themselves in a specific mark on the floor. There is no pre-defined ordering for the cups to be filled and after the robot starts to move the participants have to understand who the robot is moving towards and try to facilitate the robot's movement.

- the limits of the Kinect camera regarding where it could perceive the participants and the cups, which resulted from testing the space where the Kinect would have less errors;

- the distance from the robot to the participants, given that the arms had a maximum reach and that the closer to that maximum distance the worse the robot's movements are and may even result in incorrectly generated movements. This way the robot was capable of executing safe and correct trajectories;

- the relative position from one participant to another, so that they were equally spaced and so there would be ambiguity between them regarding who the robot was going for.

Each experiment was done with three participants simultaneously and was composed of three interactions between the humans and the robot. In each interaction the participants were exposed to one of our conditions - legible, predictable or hybrid motions. Each interaction was composed of three movements, since the robot would fill the cups of the three participants.

The order in which the robot would serve the participants was not pre-determined, so as too keep them engaged in the task and also to prevent that more observing participants would not notice patterns and exploit them for lesser reaction times. So, each time the robot would choose a new cup to serve, it would randomly select from the remaining cups that were within reach the next one. Figure 4.3 shows one of this movements to serve, as it is observable at the start the participants do not know who the robot is moving towards. However, as they figure that the robot is moving towards them or to one of the other participants, they react accordingly. When the robot is moving to serve another participant, the participants would move their cup away from the robot to help it move more easily. When they
understood the robot was moving for them, the participants would move the cup accompanying the robot’s movement.

Figure 4.3: The robot starts to reach for one of the cups. When one person understands that it is reaching for him/her, he/she reaches for robot. If it understands it is not moving towards him/her, the person helps the robot by moving the cup away.

Besides randomizing the order by which the robot fills the cups, the sequence of conditions that each participant is exposed to is also randomized. This way, in each experience the different motion types are executed in different orders. And to avoid correlation between the results of different approaches.

Each experience was composed by three testing interactions, however, to reduce the novelty effect and surprise felt by participants, every group would perform a practice round with the robot. This way, the participants would get used to the robot’s movements and how the task would progress. In the practice round, the robot would always perform predictable motions, since these are more natural and usual movements.

4.3 Procedure

The experiment took part in one of our lab rooms. As participants arrived they were asked to wait outside until the entire group of three for the experience was there.

After the entire group would arrive to the lab, they were taken to the room where they would perform the experiment. After they entered the room, they were explained that they would interact with Baxter, which had been trained to serve drinks to people and that they would interact with it three times, all three being served each time.

We explained that we were interested in ascertaining the best way for Baxter to serve them and so we needed them to evaluate each interaction, which was done using a questionnaire filled after each interaction.

Finally, the researcher would explain that the robot was safe to interact with and that he would remain in the room to monitor the robot and guarantee that there would be no problems. Also, the participants were told that at no time they were to interact with the researcher unless told to.
After explaining the experience and the safety measures, the researcher would give each participant one of the cups, which he would use in the entire experience and were told to move to the corresponding mark on the floor. When everyone was in their place and ready, the researcher would start the first interaction. After that interaction, the participants would each fill the first part of the questionnaire, where they evaluated the last interaction according with the parameters discussed later in section 4.5. This process would repeat itself for the next two interactions.

At the end of the last part of the questionnaire, each participant would sign the consent form and was given a movie voucher.

4.4 Sample

A total of 33 participants performed our experiences. Of these 22 were male and 11 females, with ages between 19 and 33 years old, an average of 23.15 years old and a standard deviation of 2.72 years.

The participants were recruited from the local community. All of them had a technical background in an engineering area, 25 of which with a background in computer science.

As explained previously, the experiment used a within-subjects design as to allow the participants to compare each of the movements they were exposed to.

4.5 Measures

To evaluate the collaboration between the robot and the humans we used both objective and subjective measures.

In terms of objective measures we analyzed the reaction time and number of errors for each participant across every condition. With this, we could objectively compare the motion types in terms of intention expression and in terms of confusion. As an error would occur when two or more participants would get confused about the robot’s target and the wrong one would be served, this could serve to measure confusion.

The reaction time was measured for each participant, in each condition, in two ways: the time it took, from the beginning of the robot’s movement until the participant understood the robot was moving to him, when the robot was moving to that participant; the average time it took, from the beginning of the robot’s movement until the participant understood the movement was not for him, when the robot was moving to another participant. These two analysis are motivated by the fact that perceiving something is moving towards us or towards another person are different cognitive processes. [20, 21, 61] Also, the division in the two ways of testing, allowed us to understand another aspect in intention transmission.
through movement, if there is a difference between motions when people perceive if they are or are not the robot's objective and which are better at creating each perception.

Regarding the number of errors, we considered an error when a participant was wrongly served. In these cases both the participant supposed to have the cup filled and the one that got the cup filled were considered as having failed and their reaction time was considered 8.5 seconds (the time it took for the robot to complete a full movement). This way we had a metric to evaluate the confusion caused by each motion type.

The collaboration between humans and robots is not evaluated solely based on objective measures like reaction time, total time to complete a task or number of errors by one or both parts. The level of collaboration is also evaluated by subjective aspects like how people perceived the fluency of the collaboration or how they saw the robot's contribution to the collaboration in relation to their own contribution. In order to evaluate the collaboration in terms of these measures we elaborated a questionnaire that intended to evaluate:

- the perceived collaboration and fluency of the task;
- the perceived legibility and predictability of the movements;
- the perceived animacy and intelligence of the robot.

All of these subjective measures were applied one time to each participant for each condition.

The perceived collaboration and fluency were evaluated using the Hoffman's questionnaire for evaluating fluency in Human-Robot Collaboration. [29] This is a questionnaire, elaborated by Hoffman et. al. in 2013, that evaluates a human's participant perception of how good was the collaboration. The evaluation is done using measures of perceived fluency, trust, contribution of the robot and perceived teamwork. From this questionnaire we used the fluency measure to analyze the perceived fluency of the task and the measures of robot contribution, trust in robot and working alliance for human-robot teams, as well as the fluency measure, to analyze the perceived collaboration of the task.

The analysis of perceived predictability and legibility of each movement used the predictability and the legibility dimensions explained by Dragan et al. in [15]. The predictability measure explored how surprising and how close or distant from the users’ expectations the movements were. The legibility measure explored the participants’ perception about the robot's expressiveness regarding its intentions and objectives.

The animacy and perceived intelligence dimensions of the Godspeed questionnaire [4] were used to evaluate how each participant rated the robot's animacy and intelligence for each type of movement. The Godspeed questionnaire was developed by Bartneck et. al. in 2009 and focus on analyzing human perception of a robot in terms of human like features. The features analyzed by this questionnaire are
how a persons perceives the robot’s anthropomorphism, animacy, likeability, intelligence and felt safety during interaction.

Finally, at the end of the third questionnaire the participants answered four forced-choice questions about which of the movements was their favorite, which was less confusing, which was easier to work with and which were they faster with. With these, we intended to evaluate the participants’ preferences of motion type and which motion type they found less confusing.
Results

Contents

5.1 Objective Measures .......................................................... 57
5.2 Subjective Measures ......................................................... 60
5.3 Forced-choice Questions .................................................... 62
The 12 sessions performed resulted in 297 movement trials. Of these, 99 regarding the time each participant took to understand the robot was moving towards him and 198 regarding the time each participant took to understand the robot was moving towards another participant. Also, the questionnaires resulted in 33 instances for each condition.

5.1 Objective Measures

The analysis in terms of the objective measures considered the following occurrences for each condition:

- reaction time to understand the robot is moving towards the participant;
- reaction time to understand the robot is moving to another one of the participants;
- number of times people wrongly understood the robot’s movement.

Figures 5.1 and 5.2 show the averages and standard deviations for each of the first two objective measures listed. Figure 5.1 shows the results for when people understood the robot was moving to serve them, while in Figure 5.2 shows the results for the time took to understand the robot was moving for another participant.

![Figure 5.1](image)

**Figure 5.1:** Average time, in seconds, each participant took to understand the robot was going to serve him, organized per movement type.

Starting with the reaction times when the robot is moving towards the participant, the data regarding these times does not follow a normal distribution because of two occurrences: participants wrongly
understood that the robot was moving towards another participant and participants that were served last, sometimes already knew they were next and started collaborating with the robot earlier than usual.

Given the non-normality of the data distribution, we performed the Friedman Test to test if there exist differences between the means of the time it took for each participant to understand that the robot was moving towards him, for each condition. We chose the Friedman Test because it is the alternative to the one-way ANOVA with repeated measures for non-parametric data, like the one we have in our case and because our data respects the assumptions: one group is measured on three or more different occasions, our dependent variable is ordinal or continuous level and that the group is a random sample of the population.

The test showed a significant difference in the time participants took to understand the robot was moving towards him depending on the motion type, with the result $\chi^2(2) = 11.546, p = 0.003$.

A post hoc analysis with a Wilcoxon signed-rank tests was conducted to determine exactly between which conditions the difference was verified. Since this is a post hoc analysis the significance level had to be adjusted to prevent false significant results. The analysis used a Bonferroni correction resulting in a significance level set at $p < 0.017$ for this post hoc analysis. Median (IQR) reaction time from the beginning of the robot’s movement until a participant understanding that the robot was moving towards him for the predictable, legible and hybrid motions was 02.63s (01.77s to 05.63s), 03.00s (01.63s to 04.79s), 01.70s (01.00s to 02.85s), respectively. There were no significant differences between predictable and legible motions ($Z = -0.78, p = 0.946$). However, between the conditions hybrid and predictable and conditions hybrid and legible motions existed differences.

The results show that hybrid motions’ reaction times were significantly lower than on the other cases.
For the case of hybrid vs predictable, the post hoc analysis reported a result of \( Z = -2.695, p = 0.006 \), which shows a predominance of results where the reaction time with hybrid motions is inferior to those with predictable motions. For hybrid vs legible, the post hoc analysis reported a result of \( Z = -3.940, p = 0.000013 \), which shows a predominance of results where the reaction time with hybrid motions is inferior to those with legible motions.

These results are in line with H3 that states that hybrid motions will result in better collaboration efficiency in the task.

Like with the case of the time it took for each participant to understand the robot was moving to him, the distribution of the data has a non-normal distribution. Likewise, we performed a Friedman Test and found out that there is a significant difference between the different motion types, with \( \chi^2(2) = 16.464, p = 0.000266 \).

To understand exactly between which conditions the difference of times was so significant, we administered a post-hoc test, the Wilcoxon signed-rank tests with a Bonferroni correction, resulting in a significance level set at \( p < 0.017 \). With this test we concluded that between predictable and legible motions there is no significant difference \((Z = -1.778, p = 0.77)\), but between predictable motions and hybrid motions \((Z = -4.076, p = 0.000008)\) and legible motions and hybrid motions \((Z = -2.790, p = 0.004)\) there is a difference. By looking at the median (IQR), the reaction times for predictable, legible and hybrid motions were 2.66s (1.54s to 4.44s), 1.59s (1.00s to 3.08s) and 1.11s (1.00s to 2.08s), respectively. This allows us to conclude that the hybrid motions allowed the participants to infer that the robot was not moving to them quicker than with the other motion types, again this goes in line with our H3 hypothesis that says that the hybrid motion will enable better efficiency in the collaboration.

The results for these two measures - time that took to understand the robot was moving for the person or for another person - show that hybrid motion performed significantly better than legible and predictable motions in communicating intentions. Also, another interesting result that stems from these results is that, contrary to prior works, legible and predictable motions did not have differences in reaction times.

Regarding the analysis of the number of wrongly perceived movements, of the 33 predictable movements 6 (18.182%) of them ended with two people getting confused about who the robot was directing the movement, of the 33 legible movements 1 (3.03%) of them ended with people getting confused about the robot’s target and on the 33 hybrid movements there were none that ended with participants getting confused about the robot’s target.

These results show, although not with great significance, a tendency for the predictable motions to be more confusing than the other motion types in this kind of tasks. This can be motivated by the fact that in tasks like this, people are not focused solely on the robot’s movement and when the targets are close to each other predictable motions may not be expressive enough to make people understand that the robot is not moving for them.
5.2 Subjective Measures

Figure 5.3 shows the results in terms of means and standard deviation for each subjective measure analyzed using Hoffman's questionnaire for evaluating fluency in Human-Robot Collaboration and for the results of the perceived legibility and predictability. The scales are in a 6-point likert scale.

![Figure 5.3](image)

**Figure 5.3:** Results for the Hoffman's questionnaire for evaluating fluency in Human-Robot Collaboration metrics - perceived fluency, robot contribution, safety and capability - and for the perceived predictability and legibility measures. The results are in a 6-point likert scale.

The subjective measures of perceived fluency and perceived collaboration were analyzed in terms of perceived fluency, perceived robot contribution, perceived trust in the robot, perceived safety and perceived robot capabilities to fulfill the task. Of these measures, only the perceived robot contribution had a non-normal distribution of the data. This way a repeated measures One-Way Anova was performed over the combined scores of each item in the fluency, trust, safety and capability metrics and a Friedman test over the combined scores of each item in the robot contribution metric.
Before performing the Anova or Friedman tests, we tested each of the subjective measures regarding each scale’s internal consistency with the Cronbach’s alpha test, the results are presented in Table 5.1 and show that the internal consistency of the scales is rated acceptable or above.

<table>
<thead>
<tr>
<th>Measure</th>
<th>$\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fluency</td>
<td>0.855</td>
</tr>
<tr>
<td>Trust</td>
<td>0.907</td>
</tr>
<tr>
<td>Safety</td>
<td>0.708</td>
</tr>
<tr>
<td>Capability</td>
<td>0.851</td>
</tr>
<tr>
<td>Robot Contribution</td>
<td>0.753</td>
</tr>
</tbody>
</table>

Table 5.1: Cronbach’s alpha results for fluency, trust, safety, capability and robot contribution measures.

The repeated measures One-way Anova tests showed that there were only significant differences in the perceived fluency ($F(2, 64) = 3.143, p = 0.050$). A post-hoc test was administered to understand where specifically these differences were, but no significance between motion types pairs was found, which leads to the conclusion that the perceived fluency is impacted by the motion type, but no specific one is especially perceived as more fluent than the other motion types. The robot contribution metric was analyzed using a Friedman test, which returned that there are no significant differences in terms of the robot contribution perceived by each participant in the different motion types.

These results in part contradict our H1 hypothesis, because apart from the perceived fluency, the other measures to evaluate the perceived collaboration show that the participants did not notice any differences between robot motion type. However, if we look to Figure ?? it is clear that, although there are no significant results, the participants rated the motion types consistently positive, what indicates that they perceived the collaboration between the humans and the robot correct and positive, even when the movements caused some confusion.

Another aspect that is interesting is that the hybrid motion was, in average, rated as the most collaborative motion and the predictable motion was rated as the least collaborative one. Although these results are not statistically significant, they show a tendency in the rating of the motion types and maybe with a bigger sample these changes could become more demarcated.

However, looking at the dimensions used to test the perceived collaboration, we notice that three of those dimensions - trust, safety and capability - are not directly impacted by the change in motion types. The difference in motion types affects the expressiveness of the movements and not directly the trust and safety felt or the perceived capability of the robot in fulfilling its task.

In terms of the perceived predictability and perceived legibility, the results obtained are present in Figure ?? and their distribution was non-normal. As for the metrics for evaluate the perceived collaboration, we did a Cronbach’s alpha test on the two metrics to check their internal consistency. The predictability scale has an acceptable internal consistency, with an $\alpha = 0.763$ and the legibility scale has an excellent internal consistency, with an $\alpha = 0.911$. 

61
The analysis of the results for these two measures was done using the Friedman test, since both scales have a non-normal distribution. The results for the predictability measure show that there was no significant difference in perceived predictability of the different motions ($\chi^2(2) = 3.429, p = 0.180$), although the median (IRQ) for the predictable, legible and hybrid conditions were of 4.33 (3.00 to 4.83), 4.00 (2.50 to 4.67) and 4.00 (3.33 to 5.00), respectively, which are high values given that a 6-point scale was used. These results, although not significantly different show that all the motion types were considered very predictable and not surprising, with both the predictable and hybrid motion having very similar results.

In terms of the perceived legibility, the Friedman test shows a significant difference between different motions ($\chi^2(2) = 7.431, p = 0.024$). A post-hoc analysis with Wilcoxon signed-rank tests was conducted with a Bonferroni correction applied, resulting in a significance level set at $p < 0.017$ and resulted in a significant difference between perceived legibility in hybrid and predictable motions ($Z = -3.075, p = 0.001$) and hybrid and legible motions ($Z = -2.446, p = 0.013$). The median (IQR) of perceived legibility for the predictable, legible and hybrid motions were 4.25 (3.25 to 5.00), 4.00 (3.00 to 5.00) and 5.00 (3.500 to 5.250), respectively. So, we can conclude that in terms of perceived legibility the hybrid motion was perceived as more legible than both the legible and the predictable motions, which is aligned with our $H3$ hypothesis.

The animacy and perceived intelligence metrics, presented in Figure ??, were both taken from the Godspeed questionnaire, using a 5-point likert scale, and analyzed using a repeated measures One-Way Anova. Both tests returned that there were no significant differences among the motion types, with the animacy dimension result being $F(1.697, 54.306) = 0.904, p = 0.396$ and the perceived intelligence dimension result being $F(2, 64) = 0.768, p = 0.468$. The result for the perceived intelligence dimension goes in line with our $H5$ hypothesis.

In terms of perceived intelligence, the results show that people perceived the robot as intelligent, rating it higher than 3 in 5 across every conditions, which shows a clear tendency of thinking of the robot as responsible, competent and wise. The perceived animacy, was not so well rated, with an overall rate barely reaching 3 in a 5 point scale, showing that although people did not see the robot as completely artificial or mechanical, they also did not see it as a living or natural entity. This can be influenced by the robot having an artificial look, see Figure 4.1 in section ??.

### 5.3 Forced-choice Questions

The analysis of the forced-choice questions was done by analyzing the frequency that each motion type was selected in each one of the questions. Since we were dealing with nominal frequencies, we performed the Chi-Square Goodness of Fit and obtained that there are no significant differences between
the motion types regarding which is the preferred one and which is the one that is less confusing.

Figure 5.5: Results for forced questions, show the total frequency each condition was chosen in the questions.

In terms of which is the less confusing motion, the frequencies were very similar with 9 people saying that predictable motion was less confusing, 13 saying legible and 11 saying hybrid. This result is interesting, even more when contrasting with number of wrong inferences of the robot’s objective that were presented previously: as shown previously 18% of the predictable movements caused doubts on the participants and the wrong one was served and in 3% of the legible movements the same occurred.

The fact that hybrid and legible motions had a similar frequency of choice, as the least confusing movement, with no significant difference goes in line with H4. This hypothesis stated that hybrid motion would not be less confusing than legible motion.

Regarding the preferred motion type, although there are no significant differences, the analysis of the results shows that, when directly asked which motion type was preferred, only 7 participants chose predictable motions, while 14 chose legible and 12 chose hybrid motions. So, there are strong indications that, albeit there are no clear difference in preferences between hybrid and legible motions, the predictable motions are by far the least preferred, which goes in line with H2 when we expect that hybrid and legible motions would be preferred over predictable motions.
Conclusion

Contents

6.1 Future Work .......................................................... 68
Human-Robot Collaboration is an area in constant expansion and that is going to have a big impact on society, because robots will have to be capable of collaborating with people.

In this work we focused on one aspect of HRC, collaborative manipulation of objects, specially the impact of movement in communication during collaborative manipulation tasks. This is a relevant topic because robot's movement, when expressive, allows the robot to implicitly communicate intentions to the humans and improve both the task's fluency and the collaboration effort.

After reviewing work in the area, we concluded that multi-user collaborative manipulation task scenario were interesting to investigate such impact. The choice was mainly because in such a scenario people's focus is not only on the robot's movement but also on the actions of the other peers and this increases the need for the robot's movements to be expressive. Thus, we designed a user study where a robot would fill cups of water to three people in which we analyzed how different robot motion impacted the collaboration.

For the study, we created a system based on the CoPMP framework to generate the robot's movements, combined with the notions of legibility and predictability to improve the movements communication of intention.

Besides using legible and predictable motions, we designed a third approach. This approach - hybrid movement - combined legible and predictable motions by allowing the robot to choose which one to execute. With hybrid movement we expected that the collaboration process would improve, by executing legible motions or predictable motions when each would be more beneficial for the collaboration effort.

The study's results showed some interesting results regarding how the perception of people was affected by the robot's movements. The first result important to note, is that the predictable and legible motions did not have significant differences, both for the time participants took to understand the robot was moving towards them and for when the robot moved towards another participant. Also, in terms of perceived legibility and predictability, there were no notable differences between motions. These results are interesting because they contradict prior works in the area however, given the fact that this task required people to collaborate in a small space, if the movement caused the smallest doubt, either because the robot approached the cups with the wrong movement or because another participant wrongly understood the robot, people would loose confidence about the robot's objective and would take longer to act. Thus, we can conclude that although legible motions are more expressive, the workspace configuration and the existence of other users play important roles on people's interpretation of the robot's objective based on its movements.

Regarding the hybrid motion, the time people took to interpret the robot's movements was significantly lower than with the other motion types. This decrease in time shows that hybrid motion was better in exposing the robot's objectives and also indicates that, in this task, the hybrid motion improves the collaborative effort. So, in terms of time that take for people to understand the robot's intentions and
react, hybrid motion allowed participants to understand if the robot was moving for them or for another of the participants faster than with any of the other approaches. Also, the hybrid motion was perceived as more legible than both legible and predictable motions. This result shows that the participants felt that the hybrid motion was easier to read and understand than the others. Also, this difference shows that it allowed for a better collaboration by giving the participants more time to adapt to the robot’s movements.

Another result that is interesting is that the participants did not show a clear individual preference between the three motions types, but hybrid and legible motions were preferred to predictable motions more frequently. This shows a tendency for the participants, in this kind of tasks, to prefer motions that are less direct and that give them more time to adapt to the robot’s movement, instead of the more direct and expected movements.

Finally, there is one last result that is interesting: the fact that the perceived intelligence and animacy metrics were not impacted by the motion type. The fact that motion type does not impact the perceived intelligence and animacy is interesting since it shows that, although the robot does not always perform the more rational, natural or expected movement, this does not impact the impression of intelligence and animation in people. So, as long as the robot fulfills its duties correctly and in a safer manner, people will perceive it as intelligent. And, as long as people can correctly understand the robot’s intention and the movements are not erratic or much different a human’s, they still see the robot as something that is alive to a certain extent.

With this work we showed that during a collaboration task, where multiple people have to manipulate common objects with the robot, the way the robot moves can have an impact on how each one perceives the robot’s intentions and perceives the collaboration effort. Also, we showed that enabling a robot to decide when to execute a more direct and expected motion or a more wide and readable motion, allows the collaboration to improve by decreasing the reaction time of the humans when the robot is moving.

6.1 Future Work

Future work will be focused on improving the robot decision of when to executing legible or predictable motions using machine learning, so to generalize this decision process to other configurations and other tasks than the ones seen here.

We are also interested in testing these findings in more complex scenarios, where the humans are also focused in performing other tasks and so the robot’s movements must be clear.

Also, we are interested in discovering the impact that other means of implicit communication would have in tasks like these and if using those would affect the need for expressiveness of the robot’s movements.

Finally, there are some works that, although not integrated in this work since they were not aligned
with our focus, present some interesting ideas for improvements on the current system.

First are the works of Mohseni-kabir et al. in [45, 46], which showed how to utilize Hierarchical Task Networks (HTN)\(^1\) in teaching robot complex tasks and how to use these HTN in collaborative natured tasks. These works give useful frameworks and ideas that can be used in abstracting different interactions that occur in more complex scenarios and setups.

Second are the works of Mainprice et al. [39–41] that present notions for safe human-robot interactions and object handovers, like the use of costmaps to decide the best movements or use models of human behavior to predict reaching patterns, that are can prove useful in systems and scenarios where the workspace in extremely crowded.

\(^1\)Method of planning the execution of robotic motion in complex tasks by subdividing each complex task into more simple tasks, creating levels of abstraction in the planning.
Bibliography


User Study Questionnaire

Following we present part of the questionnaire that was given to the participants. As this thesis was done with Portuguese speakers, the questionnaire is written in Portuguese. Since the questionnaire was composed by three equal parts we only present the last part. This last part has the particularity of having the four forced-choice questions that do not exist in the previous parts.
Parte 3

1. A equipa humanos-robot trabalhou fluentemente.
   Discordo  1  2  3  4  5  6 Concordo
   Totalmente

2. A fluência da equipa humanos-robot melhorou com o tempo.
   Discordo  1  2  3  4  5  6 Concordo
   Totalmente

3. O robot contribuiu para a tarefa ser fluída.
   Discordo  1  2  3  4  5  6 Concordo
   Totalmente

4. O robot contribuiu igualmente para o trabalho de equipa.
   Discordo  1  2  3  4  5  6 Concordo
   Totalmente

5. O robot foi o membro mais importante da equipa.
   Discordo  1  2  3  4  5  6 Concordo
   Totalmente

6. O robot trabalhou bem como parte da equipa.
   Discordo  1  2  3  4  5  6 Concordo
   Totalmente

7. Eu confiei no robot para fazer a sua parte na tarefa.
   Discordo  1  2  3  4  5  6 Concordo
   Totalmente
8. O robot era de confiança.
   Discordo 1 2 3 4 5 6 Concordo Totalmente

9. Eu e o robot confiámos um no outro.
   Discordo 1 2 3 4 5 6 Concordo Totalmente

10. O robot gosta de mim.
    Discordo 1 2 3 4 5 6 Concordo Totalmente

11. Senti-me desconfortável com o robot.
    Discordo 1 2 3 4 5 6 Concordo Totalmente

12. Sinto-me seguro a trabalhar próximo ao robot.
    Discordo 1 2 3 4 5 6 Concordo Totalmente

13. Sinto que o robot não me vai acertar enquanto se está a mexer.
    Discordo 1 2 3 4 5 6 Concordo Totalmente

    Discordo 1 2 3 4 5 6 Concordo Totalmente
15. Estou confiante na capacidade do robot para me ajudar.  
Discordo 1 2 3 4 5 6 Concordo Totalmente

16. Se eu já soubesse a que copo o robot ia dirigir-se, eu iria antecipar correctamente o movimento do robot.  
Discordo 1 2 3 4 5 6 Concordo Totalmente

17. Sabendo a que copo o robot se moveu, esse movimento coincidiu com o que eu esperava.  
Discordo 1 2 3 4 5 6 Concordo Totalmente

18. O movimento do robot não me surpreendeu.  
Discordo 1 2 3 4 5 6 Concordo Totalmente

19. O robot consegue perceber como se mexer para eu facilmente prever o seu objectivo.  
Discordo 1 2 3 4 5 6 Concordo Totalmente

20. Foi fácil prever o objectivo do robot.  
Discordo 1 2 3 4 5 6 Concordo Totalmente

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Concordo Totalmente</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discordo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Totalmente</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

22. O robot mexia-se de forma a facilitar eu perceber a quem se dirigia. 

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Concordo Totalmente</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discordo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Totalmente</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

23. Qual dos tipos de movimentos permitia-te ser mais rápido? 

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Concordo Totalmente</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discordo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Totalmente</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

24. Qual dos tipos de movimentos era mais fácil de trabalhar com? 

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Concordo Totalmente</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discordo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Totalmente</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

25. Qual dos tipos de movimentos preferes? 

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Concordo Totalmente</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discordo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Totalmente</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

26. Qual dos tipos de movimentos era menos confuso de perceber? 

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Concordo Totalmente</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discordo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Totalmente</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Para cada linha, *seleciona o número* que melhor representa a tua opinião do Baxter:

- Morto
- Estagnado
- Mecânico
- Artificial
- Vivo
- Vivacidade
- Orgânico
- Natural
<table>
<thead>
<tr>
<th>Atributo</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>Atributo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inerte</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Interativo</td>
</tr>
<tr>
<td>Apático</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Responsivo</td>
</tr>
<tr>
<td>Incompetente</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Competente</td>
</tr>
<tr>
<td>Ignorante</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Conhecedor</td>
</tr>
<tr>
<td>Irresponsável</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Responsável</td>
</tr>
<tr>
<td>Pouco inteligente</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Inteligente</td>
</tr>
<tr>
<td>Insensato</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Sensato</td>
</tr>
</tbody>
</table>

**Obrigada pela tua colaboração!**

**Por favor, assina o consentimento informado.**