TÉCNICO LISBOA

UNIVERSIDADE DE LISBOA

INSTITUTO SUPERIOR TÉCNICO

Using Legibility to Improve Interactions with Agents during Multi-User Interactions

Miguel Afonso Tomás Faria

Supervisor: Doctor Francisco António Chaves Saraiva de Melo

Co-Supervisor: Doctor Ana Maria Severino de Almeida e Paiva

Thesis approved in public session to obtain the *PhD degree* in

Computer Science and Engineering

Jury final classification: Pass with Distinction

2023

UNIVERSIDADE DE LISBOA

INSTITUTO SUPERIOR TÉCNICO

Using Legibility to Improve Interactions with Agents during Multi-User Interactions

Miguel Afonso Tomás Faria

Supervisor: Doctor Francisco António Chaves Saraiva de Melo

Co-Supervisor: Doctor Ana Maria Severino de Almeida e Paiva

Thesis approved in public session to obtain the *PhD degree* in

Computer Science and Engineering

Jury final classification: Pass with Distinction

Jury:

Chairperson: Doctor Maria Inês Camarate de Campos Lynce de Faria, Instituto Superior Técnico, Universidade de Lisboa

Members of the Committee:

Doctor Estela Guerreiro Bicho Erlhagen, Escola de Engenharia, Universidade do Minho

Doctor Francisco António Chaves Saraiva de Melo, Instituto Superior Técnico, Universidade de Lisboa

Doctor Anca Dragan, Electrical Engineering and Computer Sciences Faculty, Berkeley, University of California, EUA

Doctor Rodrigo Martins de Matos Ventura, Instituto Superior Técnico, Universidade de Lisboa

Funding Institution - Fundação para a Ciência e a Tecnologia

2023

Abstract

Communication between humans and agents is of great importance for interactions between humans and agents to occur smoothly, namely in interactions that require collaboration. In this thesis investigate the influence of legible motion in multi-party interactions, particularly multi-party collaborative tasks. Namely, we are interested in investigate if the use of legibility improves the expressiveness and clearness of a robot. We then consider several sequential decision-making tasks and multi-party interactions in this class of tasks.

The first aspect we consider important to investigate is the impact, in multi-party interactions, of different users' points-of-view (PoVs) over the workspace. By having different PoVs over the same movement, each user has its own perception of it, possible leading to different interpretations of the robot's intention. Thus, we propose that the definition of legibility for multi-party interactions jointly considers the different users' PoVs, defining the legibility of a movement as the average of the legibility for each user. Using this definition, a legible movement focuses on improving the group's understanding of the robot's intention, instead of improving the individual user's legibility.

The second aspect we explore in this work is the impact of legibility in sequential decision-making. Standard decision making processes focus on optimizing an underlying cost function, thus improving a robot's efficiency in solving a task. However, the optimal solution for a task is not always the more explicit one; *e.g.* consider a robot fetching an specific item from a list of items in a warehouse; the optimal solution may lead it through areas that contain other items on the list, causing an observing human partner confusion as to why the robot did not pick the other items on the path. To solve this problem we propose policy legible Markov decision problem (PoLMDP), a framework for legible decision making, where the robot's actions optimize both the underlying cost function and the robot's expressiveness. This way, the robot's actions are clear about its intentions, while the robot is solving it's own tasks and objectives.

Finally, we show the effectiveness of our framework for legible decision-making in transmitting the robot's intentions to partners, both when the partner is just an observer and when the partner is teamed with the robot to solve a collaborative task. Our results show that using PoLMDP improves both human inference of a robot's intentions from its actions and the efficiency of collaboration in collaborative tasks.

Keywords: legibility, multi-party interactions, decision making, agent planning, XAI

Resumo

A comunicação clara é crucial para interações entre humanos e robots, ou outros agentes artificiais. Neste trabalho aborda-se o problema de melhorar a clareza de um robot usando legibilidade, durante interações de multi-utilizadores. Legibilidade mede quão bom é um movimento em desambiguar o objetivo de um robot, de outros objetivos possíveis. Para explorar o impacto da legibilidade em interações com vários humanos; primeiro exploramos como definir legibilidade no contexto de interações de multi-utilizadores e, em segundo lugar, exploramos a noção de legibilidade na seleção de ações e tomada de decisões de robots.

Na primeira parte desta tese, propomos uma definição de legibilidade que considera conjuntamente as diferentes perspetivas humanas sobre o mesmo movimento robótico. Assim, a legibilidade é definida em função de todo o grupo em vez de cada humano em particular, permitindo aplicações tanto em interações com um só utilizador como com múltiplos utilizadores. Também mostramos, por meio de um estudo com humanos, que a nossa definição de legibilidade para interações de multi-utilizadores melhora a compreensão das intenções de um robot, quando comparado com a formulação para um único utilizador.

Na segunda parte da tese, damos um passo além dos movimentos do robot e exploramos a aplicação da legibilidade na tomada de decisão de robots e outros agentes artificiais. Exploramos a aplicação da legibilidade no contexto da tomada de decisão em ambientes estocásticos, desenvolvendo uma abordagem para tomada de decisão legível denominada *policy legible MDP (PoLMDP)*. O PoLMDP é capaz de lidar com a incerteza enquanto permanece computacionalmente tratável, como mostramos num estudo comparativo com outras abordagens de última geração para tomada de decisão legível. Também mostramos, em dois estudos diferentes, que nossa abordagem PoLMDP torna os objetivos de um robot mais fáceis de entender, tanto em cenários de agente único quanto multi-agente, quando comparado aos MDPs tradicionais.

Palavras-chave: legibilidade, interações multi-utilizador, tomada de decisão, planeamento com agentes, XAI

Acknowledgments

This thesis encloses the work of my PhD, a labour of love for the fields of human-robot and more generally human-agent interaction and artificial intelligence. However none of what I have achieved in this PhD would have been possible without some very special people that supported me along the way.

The first two people on this long list are none other than my two advisors: Francisco Melo and Ana Paiva. I honestly could not have asked for a better duo than you two to guide me through this journey. Each one of you brought a set of tools and knowledge that complemented each other and always allowed me to find the answers. Thanks to your guidance, I found my area of interest merging human-robot interaction with machine learning. Francisco, you brought the expertise in machine learning and the patience to go over, with me, all the technical aspects of the most diverse problems that appeared. I've learned so much with you and I feel that, no matter how long I work on the field of machine learning, I will always have something new to learn from you. Thank you for always having your door open when I needed to talk, be it about my research or some other topic, you always made feel more like a peer than a student of yours! Ana, you brought all the expertise regarding interaction with robots, user studies and all that wonderful world of human-robot interaction. Ever since that first meeting, all the way back in 2015, where you gave me Anca's paper and asked me to read it and tell you if I was interested in doing my Master's Thesis on it that I've been hooked in the field. So, thank you for having put me on this path! Thanks to both of you, I discovered how much I love teaching and see the spark in the eyes of students when they connect with what I'm explaining. Thanks for having come to me one afternoon, towards the end of my Master Thesis, and asked if I'd like to continue at GAIPS as a researcher. Both of you were instrumental for the researcher I am today!

I am very grateful to my CAT committee – professor Rodrigo Ventura, professor Estela Bicho and professor Manuel Lopes – for bringing unique perspectives that helped shape the later half of my PhD. Your comments on the state of my work, namely how scattered my focus was, trying to go for everything, were crucial in narrowing down the focus on combining legibility and decision-making. For that and all the other comments you gave me I have to thank you.

I also want to thank professors Alberto Sardinha and Inês Lynce, whom I had the pleasure of teaching and have given me the chance of teaching subjects that I usually don't teach. Teaching with both of you widened my view of the world and the applications of machine learning.

For the last 8 years I have been part of the GAIPS lab at Inesc-ID and it has been an amazing experience. It's almost impossible to name every single member that has impacted me, so I want to extend a hear felt thanks to everyone at the lab from the ones that already left the lab like Hang, Filipa and Rui, to the ones that are now leaving like Ali, Miguel, Manuel and Ricardo, to the ones that are now starting or have already started their journey there like Ana, Fábio, Bernardo, Henrique, the two Ineses... You will always be cherished! I have to issue a special thanks to my room mates Ali, Miguel, Fábio, Ana (and João and Raquel, i know you are no longer part of room 2N7.3, but to me you'll always be part), you guys were always there when I needed to blow off steam, had some sort of mental block or when there were any demos, so thank you for having made the last years so enjoyable!

To my friends, the list is way too big to enumerate every single one of you, but I'll never forget all the nights together, the trips together, the dinners, themed parties,... Those were the things that helped me pushing threw all the difficulties this PhD posed.

To my parents and grandmother, you were always there by my side and I can never repay all the support you gave, since ever. I honestly have no words to thank you for all you have done. I can only say how fortunate of having you all in my life and I promise that I'll try my hardest to always make you proud.

Finally, there's one person without whom nothing of this would have been possible. To my other half, to my best friend, companion and best girlfriend I could have asked. Thank you Ana!! You have always been there to support me even when I didn't know I needed you. Through the course of this PhD, you have had nights robbed of my company, plans laid to the ground, weekends shared with my research... you name it, you have endured and never complained. I feel so lucky to have you by my side and to able to celebrate with you my accomplishments, even when you don't get why they are so important. Thank you and I love you so much!

To all of you, my sincere thanks

Miguel Faria

Contents

List of Tables

List of Figures

Acronyms

AI	artificial intelligence
AIA	artificial intelligent agent
ASD	autism spectrum disorders
DMP	dynamical movement primitive
DoF	degree of freedom
FoV	field-of-view
HRC	human-robot collaboration
HRI	human-robot interaction
I-POMDP	interactive POMDP
IRL	inverse reinforcement learning
L-MDP	legible Markov decision problem
M-Turk	amazon's mechanical turk
MDP	Markov decision problem
MP	movement primitive
MMDP	multi-agent Markov decision problem
MUL	multi-user legibility
o.d.e.	ordinary differential equation
PoLMDP	policy legible Markov decision problem
POMDP	partially observable Markov decision problem
PoV	point-of-view
ProMP	probabilistic movement primitive
RMRC	resolved motion rate control
SUL	single-user legibility
XAI	explainable artificial intelligence

1

Introduction

Robots are becoming a common sight in society, no longer restricted to just working in factories. So far, they can be found in the most varied roles: assisting humans to overcome disabilities in everyday tasks [20, 52]; helping children through educational applications [17, 57, 98]; assist the elderly to maintain their independence [8]; assisting in healthcare [48, 54, 71]; or in entertainment of humans [24].

The integration of robots in society requires robots to be able to interact smoothly with humans, and to correctly communicate their internal state. Correct communication is essential for interactions between humans and robots to be *safe* and prevent injuries to humans [72]; *efficient*, to complete personal goals [42, 77]; and *natural*, so humans do not feel uneasy interacting with robots [91].

The importance of correct communication rises even more in scenarios where one robot interacts with various humans simultaneously. In these scenarios, the efficiency and safety of the task comes from the group's understanding of what each participant's current goal is. Thus, a robot that is not capable of being simultaneously clear to all parties can be a serious liability for those interacting with it. This failure in communicating can cause different parties to interpret differently the robot's intentions and act in a way that leads them to a collision, be it physical or just of personal goals. Let us consider Example 1.

> **Example 1** *A robot, working in a cafeteria, is tasked with filling customers' glasses with water when prompted to. John and Mary ask the robot to fill their glasses and remain close to each other while talking and waiting, with Mary closer to the right side of the robot and John to the left side. The robot decides to first fill John's glass and then Mary's, planning the movement accordingly. The robot's movement planner concludes that the best course of action is to execute a steady straight movement, with the pitcher on the right arm, towards John's cup to fill it. However, given their proximity, the movement comes close to both John and Mary and both interpret that their respect cup is going to be filled. The wrong interpretation causes both to move their glasses, in a collaboration effort with the robot, and unintentionally collide with each other.*

Communication can be achieved in a myriad of ways, divided in two main types of communication: explicit and implicit communication. Explicit communication is a type of communication where the actor explicitly communicates intent. Implicit communication is a type of communication where the recipient must interpret the action of the actor to infer the intention [7]. The combination of explicit and implicit communication allows humans to easily interact with each other, even in situations where no explicit signals are traded [40]. Humans naturally interpret signals from each other during interactions, trying to predict the intentions of others and adapt accordingly. An example is when humans follow the gaze of their partners to understand and preempt the outcomes of their actions [33]. Therefore, robots involved in human-robot interaction must be capable of leveraging the use of explicit and implicit methods of communication for interactions to feel normal to humans.

1.1 Problem statement

Body movements and the way we choose and sequence our actions when performing a task are extremely useful and essential tools in communicating intents and emotions. Body movements convey the same intentions independently of culture, upbringing or education, and humans comprehend those intentions innately [34]. Human decision-making and acting allows for humans to establish a "procedural common ground", leading to better task coordination [22, 85]. The use of these types of implicit communication by a robot also leads to people consider robots more socially engaging, warm, friendly and empathetic; as well as more expressive when conveying different emotions. This attitude towards robots improves task performance and causes less errors [90].

Legibility measures the degree of distinctiveness with which an observer can categorize and decode its surroundings [55].Dragan, Lee, and Srinivasa [31] applied the concept of legibility to robotic movements, defining *legible movements*. These movements have been designed to augment a robot's expressiveness by reducing the ambiguity regarding the robot's objective. Thus making the robot's movements clearer and easier "to read" by human users.

The use of legible movements by a robot is a possible solution to communication problems in multi-party interactions, as in Example 1. A robot capable of correctly perform legible movements and leverage those movements together with other communication methods is capable of minimizing situations where humans misunderstand a robot's intentions. The concept of legibility can be applied more generally than robotic movements, we postulate that legibility can be applied to a robot's decision making process to make more complex tasks easier to understand when the robot is part of a team. Considering again Example 1, in the example the robot's decision making process considered that it was optimal to serve first John and secondly Mary; however, looking at the positioning of Mary and John, serving Mary first would have been more legible since she was closer to the robot's serving arm and so the action would have been easier to understand.

Thus, in this thesis we address the question:

Research Question *Can an autonomous agent (such as a robot) better convey its intentions in multi-party interactions by using legible actions?*

To investigate this problem we break it down into sub-problems:

RQ 1 In multi-party interactions, will the joint consideration of multiple users' PoVs improve legibility of a robot's movement?

RQ 2 Can the notion of legibility be extended to problems of sequential decision-making under uncertainty, to improve a human's understanding of an agent's actions?

RQ 3 Can legible decision-making improve a team's efficiency in multi-agent tasks?

We now elaborate on the different sub-problems above.

1.1.1 Perspective influence in legibility during multi-party interactions

Legible movements aim at improving user's understanding of a robot's objective, by creating movements that simultaneously approach the intended objective and get away from other possible objectives. Thus, the resulting movements reduce the ambiguity of what the robot's intention is.

In multi-party interactions, a robot interacts simultaneously with multiple users, each with its own point-of-view (PoV) over the workspace and the robot's movements. Thus during these interactions, there are multiple perspectives of a robot's movement that may lead to multiple interpretations of it.

However, the existence of multiple different perspectives over the same movement, can have an impact in the movement's legibility, causing some movements to be interpreted differently when observed from different perspectives. Example 1, provides a scenario where different perspectives can impact the perception of a robot's motion: the robot executes a legible movement to fill John's cup, but does not account for the various perspectives. The movement may be optimized to improve John's perception but given the fact Mary and John have different perspectives, Mary may perceive the movement differently and believe the robot is filling her cup, moving towards the robot.

In problem **RQ 1** we investigate a way to mitigate the negative impacts of multiple perspectives in multi-party interactions. We propose an extension to the legibility metric that combines the legibility of the movement according to each human's PoV. Thus, the perspectives of all human partners shape the trajectory, making the legibility of the group increase instead of the legibility of a subset of the human partners.

1.1.2 Legibility in sequential decision-making tasks

Robotic and other artificial intelligent agents (AIAs)'s interact with humans, so they need to make their decisions clear for humans. However, autonomous agents' decision-making is not always clear to those interacting with them. One of the biggest hurdles to understand the behaviour of intelligent agents is inferring the reasoning behind the agents' actions and behaviours [16].

An intelligent agent's decision-making can be described in terms of optimizing a specific reward or cost function. However, humans don't always act "optimally" because humans, despite having their own set of internal preferences, do not actively try to optimize some function; they instead behave in a manner that seems logic for the situation at hand. This difference between human and artificial agents decision making processes is crucial to understand why humans may have difficulties in understanding the reasoning of AIAs.

We consider the possibility to make robotic and AIAs' decision-making easier to understand by humans by extending the notion of legibility – originally proposed for robot motion – to more general decision making processes, making the decision processes more transparent and self-explanatory for the humans observing and interacting with the agent.

Question **RQ 2** explores exactly how we can extend the notion of legibility to decision making problems, and whether this extension improves the overall interaction between humans and agents. We propose a new framework for decision-making capable of choosing actions and behaviours for agents that allow a human interacting with the agent to better understand its intentions.

1.1.3 Impact of legible decision-making in teamwork efficiency

Teamwork efficiency is directly influenced by the communication between team members, namely by how the team members coordinate with each other to achieve the team's goals. However, for the team's members to correctly coordinate with each other they must be able to easily infer each others' intentions from observed actions and behaviours. This way, each member can adapt its goals, actions and behaviours to better support the team and help in the team's efforts.

Considering this fact, robots and artificial agents in general must decide on the best actions to perform and behaviours to display that allow a fast task completion and yield more information about the agent's current objectives. Otherwise, if an agent focuses only on solving the task, it may indicate that it is trying to achieve a goal different than the one it is truly going for. Consider Example 2, showcasing a scenario where the agent's action may lead to an incorrect understanding.

> **Example 2** *A robot and a human are playing a collaborative game of Word Blocks, where they have to build words with the letters available in wooden blocks. The only letters available at the time are one "b" block, two "a" blocks, one "o" block, one "r" block and one "t" block. The robot decides, without communicating with the human, to start making the word "boar" and starts by picking the block "b". Then the human trying to help picks the block with the letter "a", being the most common letter available, and places it next to the "b" block. The robot then, trying to be optimal, picks the block with the letter "o" and places it between the "b" and "a" blocks. Finally, with the sequence of blocks "b","o,"a", the human believes the robot is trying to write "boat" and places the "t" block to complete the sentence, creating the wrong word. However, if the robot had changed one action and it had picked and placed the "r" block at the end instead of the "o" block, the human partner would have had more information about the word being built and would had been more difficult to build the wrong word.*

Question **RQ 3** explores the problem of applying legible decision-making in team scenarios, understanding if legible actions improve the overall team efficiency in solving collaborative tasks.

1.2 Summary of contributions

Summarizing, in the thesis we contribute:

1. A novel definition of legible movements focused in multi-party interactions that takes in consideration the PoV of the human partners in the legibility metric.

2. PoLMDP, a novel framework for legible decision-making, that allows the efficient com-

putation of legible plans in both stochastic and deterministic scenarios improving human understanding of an agent's underlying goals.

3. A study on the impact of legible decision-making in collaborative tasks.

Our novel definition of legible movements in multi-party interactions has been published in [36]. The two remaining contributions are currently under review: our framework for legible decision-making, PoLMDP, is under reviewing process for the Elsevier's Artificial Intelligence Journal; our final contribution is under review for the IJCAI conference.

1.3 Thesis outline

The rest of the thesis is structured as follows:

- Section 2 provides an overview of background material and notation.
- Section 3 reviews the literature on legibility and general communication in robotics and human-robot interaction (HRI).
- Section 4 introduces our first contribution and explores the impact of multiple perspectives over a movement and how we can incorporate those perspectives to improve a movement's legibility.
- Section 5 introduces the policy legible Markov decision problem framework – a framework based on the formalism of MDPs for legible decision-making – and the evaluation of its performance in making a robot's intentions clear to humans.
- Section 6 describes the study conducted to explore the impact of legible decision-making in team scenarios, namely in tasks that require strict collaboration between team members to succeed.
- Section 7 revisits the major contributions of the thesis, explores its main conclusions and points some directions for future work.

2

Background

In this chapter we go over several core concepts that support our work and formalize the notation used in this document. We start with trajectory representation, clarifying the notation used. Afterwards we define viewport projection and provide an overview of what are legible and predictable movements, two properties associated to movements that improve the communication of intent of robots. Finally, we provide an overview of what is an MDP and the notation used in this thesis.

2.1 Trajectory representation

A robot trajectory, which we denote as $\boldsymbol{\xi}$, is a sequence of robot configurations. Trajectories may describe the physical displacement of the robot or the movement of a robot's limb. Thus, a trajectory is a mapping from a time step t to the corresponding robot configuration, $\boldsymbol{\xi}(t)$. Each robot configuration is a vector that can be described in:

- 3D cartesian space, in which case a configuration is a vector $\boldsymbol{\xi}_t = [x_1(t), x_2(t), x_3(t)]^T$ — describing solely the 3D position of the robot — or a vector $\boldsymbol{\xi}_t = [x_1(t), x_2(t), x_3(t), \alpha_1(t), \alpha_2(t), \alpha_3(t)]^T$ — describing both the 3D position and orientation of the robot;
- joint space, in which case a configuration is a vector $\boldsymbol{\xi}_t = [\theta_1(t), \theta_2(t), \ldots, \theta_N(t)]^T$ with N the number of joints in the robot and each $\theta_n(t)$ is the angle joint of the robot.

There are many frameworks to generate robotic trajectories, in this work we adopt *movement primitives*.

2.1.1 Movement primitives

Movement primitives (MPs) are used to concisely represent robot control policies. The aim of MPs is to represent simple movements, such as hitting and batting movements [50, 59], and compose these movements into complex robot skills. MPs have been successfully used to model tasks like 'Ball-in-the-Cup' [58], ball throwing [27] and pancake flipping [62].

In this work we use two MP formulations: *dynamical movement primitives (DMPs)* [49] and *probabilistic movement primitives (ProMPs)* [83].

Dynamical movement primitives are a type of MPs that represent trajectories as dynamical systems. Specifically, the trajectory of the robot along each dimension corresponds to the solution of a damped spring system, *i.e.*, the solution to the ordinary differential equation (o.d.e.)

$$\tau \ddot{x} = \alpha_x \left(\beta_x (g - x) - \dot{x}\right) + f, \tag{2.1}$$

where τ is a time factor, α_x and β_x are positive constants, g is the goal state and x, $\dot{x}$ and $\ddot{x}$ correspond to the system's position, velocity and acceleration. The f term represents a non-linear forcing function that modifies the trajectory's shape towards to the goal g. Equation 2.1, known as *transformation system*, resembles a point attractor dynamical system on goal g with the addition of term f.

The forcing term is defined as

$$f(z) = \frac{\sum_{i=1}^{N} \Psi_i(z)\, w_i}{\sum_{i=1}^{N} \Psi_i(z)} z\,(g - x_0)\,, \tag{2.2}$$

where each Ψ_i is a Gaussian basis function,

$$\Psi_i(z) = \exp\left(-\frac{1}{2h_i^2}(z - c_i)^2\right),$$

N is the number of basis functions, h_i and c_i represent the width and center of each basis function, x_0 is the initial state, w_i are adjustable weights that shape the trajectory and z is a phase variable that behaves as a replacement for time, and is governed by the o.d.e.

$$\tau \dot{z} = -\alpha_z z, \tag{2.3}$$

with α_z a constant. Thus, the forcing function f is a set of Gaussians that are "activated" as the canonical system, given by (2.3), converges to zero.

Given their dynamical formulation, DMPs generate trajectories that are goal-oriented, robust to perturbations and noise and arbitrarily shapeable to avoid obstacles or to follow demonstrated trajectories. Furthermore, the τ parameter in Equations 2.1 and 2.3 allows for temporal scaling, while the $(g - x_0)$ factor in Equation 2.2 allows for spatial scaling. These two properties make DMPs capable of generalizing trajectories to different execution speeds, or to targets closer or farther away, giving DMPs topological invariance. Topological invariance means that trajectories that use the same DMP model have the same shape, even if the execution speed and movement amplitude are different, allowing for interesting applications such as movement recognition [61].

The forcing function in (2.2) defines a discrete or stroke based movement, such as swinging a baseball bat. DMPs can also execute cyclic or rhythmic based movements, such as bouncing

Property	Implementation
Temporal Scaling	Modulate Phase
Rhythmic or Stroke Movements	Basis Function
Optimal Behavior	Encode Variance
Trajectory Modulation	Conditioning
Coupling of DoF	Mean and Covariance
Co-Activation (Combination) of MP	Product of Distributions
Learning from Demonstration	Maximum Likelihood

Table 2.1: Movement Primitives properties and implementation in Probabilistic Movement Primitives

a ball on a table tennis paddle, by instead using a forcing function of the form

$$f\left(r,\phi\right)=\frac{\sum_{i=1}^{N}\Psi_i\left(\phi\right)w_i}{\sum_{i=1}^{N}\Psi_i\left(\phi\right)}r,$$

with,

$$\Psi_i\left(\phi\right)=\exp\left(\frac{\cos\left(x-c_i\right)}{2h_i^2}-1\right),$$

and the canonical system

$$\tau\dot{\phi}=1,$$

where r is the amplitude signal and ϕ the phase signal.

The forcing function can be designed by humans or learned through reinforcement learning techniques. For human-designed forcing functions, human experts define the w_i parameters in Equation 2.2. Having forcing functions manually designed allows for well defined movements to be easily implemented as robot control policies. In the case of learned forcing functions, the system learns the w_i parameters. Learning the w_i parameters allows DMPs to imitate observed movements as well as improve learned forcing functions with new human interactions and demonstrations.

DMPs can be extended to multiple degrees of freedom by temporally coupling the different degrees of freedom (DoFs). The coupling is performed using the same canonical system to all DoF, while each one has different transformation system with its own forcing function.

Probabilistic movement primitives are another type of MPs that represent trajectories as a probabilistic distribution over possible movements. The idea behind ProMPs is to use operations from probability theory to support a set of desirable properties for MPs, summarized in Table 2.1.

In the ProMP framework, a MP corresponds to a probability distribution over possible trajectories. Each trajectory $\boldsymbol{\xi}$ is a sequence of T different configurations for a 1-DoF robot. ProMPs encode the control policy using a hierarchical Bayesian model.

In order to capture the variance information of the different trajectories, ProMPs use a weight vector $\boldsymbol{w}$ to compactly represent a single trajectory and introduce two distributions, $p(\boldsymbol{\xi}|\boldsymbol{w})$ and $p(\boldsymbol{w};\boldsymbol{\theta})$. $p(\boldsymbol{\xi}|\boldsymbol{w})$ defines the probability of observing trajectory $\boldsymbol{\xi} = \{\xi(0), \xi(1), \ldots, \xi(t)\}$ given the weight vector $\boldsymbol{w}$ and is given as

$$p\left(\boldsymbol{\xi}|\boldsymbol{w}\right) = \prod_t \mathcal{N}\left(\boldsymbol{\xi}(t)|\,\boldsymbol{\Psi}(t)\boldsymbol{w}, \boldsymbol{\Sigma}\right),$$

with

$$\boldsymbol{\xi}(t) = \boldsymbol{\Psi}(t)\boldsymbol{w} + \boldsymbol{\epsilon}(t), \tag{2.4}$$

and

$$\boldsymbol{\Psi}_t = [\psi_t, \dot{\psi}_t]^T, \tag{2.5}$$

a $2{\times}n$ dimensional time-dependent basis function matrix for the joint positions $\psi(t)$ and velocities $\dot{\psi}(t)$, where n defines the number of basis functions. $\boldsymbol{\Sigma}$ is the observation noise variance and $\boldsymbol{\epsilon}(t) \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ represents a zero-mean i.i.d. Gaussian noise. The distribution $p\left(\boldsymbol{w};\boldsymbol{\theta}\right)$ captures the variance of trajectories with parameters $\boldsymbol{\theta} = \{\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}}\}$ that specify the mean and variance of $\boldsymbol{w}$. Combining the two distributions as follows

$$p\left(\boldsymbol{\xi}|\boldsymbol{\theta}\right) = \int p\left(\boldsymbol{\xi}|\boldsymbol{w}\right) p\left(\boldsymbol{w}|\boldsymbol{\theta}\right) d\boldsymbol{w}, \tag{2.6}$$

defines a distribution over trajectories with parameters $\boldsymbol{\theta}$. Equation 2.6 defines the hierarchical Bayesian model used by ProMPs, given by the observation noise variance $\boldsymbol{\Sigma}$ and the parameters $\boldsymbol{\theta}$. The observation noise variance $\boldsymbol{\Sigma}$ is obtained from the demonstration data and the parameters $\{\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}}\}$ can be learned from multiple demonstrations using maximum likelihood estimation.

The basis function matrix $\boldsymbol{\Psi}$ in (2.4) is time dependent, which prevents temporal scaling and executing trajectories faster or slower than demonstrated. Thus, to decouple the movement from the time signal, a phase variable z_t is introduced, as with DMPs, modifying the basis function as follows

$$\psi(t) = \psi(z(t)),$$
$$\dot{\psi}(t) = \frac{d\psi(z(t))}{dz(t)}\dot{z}_t,$$

with t varying from 0 to T, where $z(0) = 0$ defines the beginning of the movement and $z(T) = 1$ the end, $\dot{z}(t) = 1/T$ and $\psi(z(t)) = [\psi_0(z(t)), \ldots, \psi_n(z(t))]$ where

$$\psi_i(z) = \frac{b_i(z)}{\sum_{j=1}^{n} b_j(z)}. \tag{2.7}$$

The choice of basis functions used depends on the type of movement the ProMP must model. For discrete or stroke-based movements, the basis functions are

$$b_i(z) = \exp\left(-\frac{(z - c_i)^2}{2h}\right),$$

while for cyclic or rhythmic based movements

$$b_i(z) = \exp\left(\frac{\cos(2\pi(z - c_i))}{h}\right),$$

with h the width of the basis functions and c_i the center of the ith basis function.

ProMPs are extended to multiple DoFs by coupling the different DoFs, so that all DoFs are controlled together. To achieve this coupling, both the $\boldsymbol{w}$ weight vector and the $\boldsymbol{\Psi}$ basis function matrix are extended to cover all the DoFs of a robot, while sharing the same phase variable.

The use a Bayesian model allows for ProMPs to modulate trajectories to reach different final positions or passing certain via-points. To modulate trajectories to reach different final positions or to pass through specified via-points, the new intended observation $\{\boldsymbol{y}^*, \boldsymbol{\Sigma}^*\}$ is added to the probabilistic model at the desired time step t. In the new observation, $\boldsymbol{y}^*$ represents the desired position and $\boldsymbol{\Sigma}^*$ describes the accuracy of the observation. With the addition of this observation, a new trajectory $\boldsymbol{\xi}^*$ is defined that can the be solved using (2.6) to find the parameters that better model the new trajectory.

As previously explained, in ProMPs each MP creates a probability distribution that represents multiple ways to execute a movement. If a ProMP model maintains a set of i different primitives, then these different primitives can be co-activated by taking the product of their distributions creating a new probability distribution,

$$p_{new}(\boldsymbol{\xi}) \propto \prod_i p_i(\boldsymbol{\xi})^{\gamma^{[i]}},$$

where $\gamma_t^{[i]}$ are factors comprised between 0 and 1 that denote the activation of each primitive. The primitive p_{new} allows the ProMP model to execute movements that use the parts of the MP that best resemble the desired trajectory, allowing to reach zones of the trajectory space that are well covered by different MPs.

Another operation ProMPs can perform is to blend MPs. Blending MPs is an operation that allows ProMPs to define movements that sequentially follow different MPs, allowing to compose complex movements from different simpler movements. To achieve this blending behaviour, the ProMP model uses a reasoning similar to the co-activation of different primitives, where the trajectory is decomposed into single time steps before taking the product of MPs,

$$p_{new}(\xi) \propto \prod_t \prod_i p_i(\boldsymbol{y}_t)^{\gamma_t^{[i]}},$$
$$p_i(\boldsymbol{y}_t) = \int p_i\left(\boldsymbol{y}_t|\boldsymbol{w}^{[i]}\right) p_i\left(\boldsymbol{w}^{[i]}\right) d\boldsymbol{w}.$$

By decomposing each MP in single time steps, allows for finer grain control of the executing MP and to compose MPs to sequentially execute one after the other instead of executing a combination of all the MPs.

Dynamical movement primitives vs. Probabilistic movement primitives

DMPs and ProMPs are two similar frameworks of MPs that can learn different robot control policies. Both frameworks allow to scale the trajectories generated to varied execution speeds, to represent both rhythmic and stroke based movements, to shape the trajectories around obstacles and to different goals or to create movements that control multiple coupled DoFs.

However, the two frameworks have differences that make them ideal for different scenarios. DMPs use the forcing function f to modulate the generated trajectory and this function can be either learned or defined by humans. Thus, DMPs can create movements that do not need human demonstration. In comparison, ProMPs have the limitation that they require a set of example trajectories to create the probabilistic model used to create new movements. Thus, ProMPs are not capable of create new movements without previous examples like the DMPs.

On the other hand, the probability model used in ProMPs allows the framework to co-activate different MPs, generating movements capable of adapting to different task restrictions and demands. By co-activating different MPs, ProMP created movements can cover parts of the trajectory space that are well covered by some but not all of the model's MPs. Besides co-activation, ProMPs are able to create movements composed by different simpler movements by sequentially activating different MPs. By sequentially executing different MPs, the composed movement is capable of executing actions covered by the different MPs. DMPs, on the contrary of ProMPs, can not perform these types of combinations because each DMP model generates movements by manipulating a set of w_i parameters to shape the trajectory, thus modeling only one way of executing a movement.

2.2 Viewport projection

Viewport projection or perspective projection is a technique widely used in robotics, vision and games to project a 3D scene, visible by a camera or a human eye or any other visual sensor, to a 2D plane. The conversion from 3D to 2D is useful because it allows to know what objects are occluded – out of the field-of-view of the sensor and thus out of view – and how each object is being perceived given their proximity to the sensor.

Viewport projection is done as follows: assuming that the points that define each object to project are defined in the sensor's coordinate system[1] project the points from 3D to 2D and clip out those with a position outside the currently visible area of the sensor.

The transformation between two coordinate systems uses the translation between the systems' origins and rotation between corresponding axes on each system, to transform points defined in one coordinate system into points defined in the other. Consider we have a scenario like Figure 2.1, where point P is defined in a coordinate system A – P_A – and we want obtain the same point defined in system B – P_B. We obtain P_B from P_A by applying transformation ${}_BT^A$ as in (2.8).

$$P_B = {}_BT^A \cdot P_A, \tag{2.8}$$

with P_A defined in homogeneous coordinates in coordinate system A, P_B the same point de-

[1] If not transform the points into the sensor's coordinate system

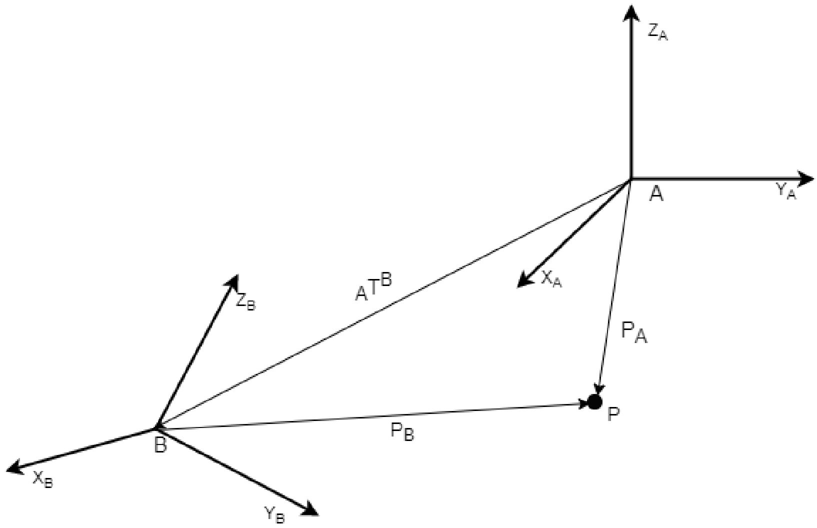

Figure 2.1: Coordinate transformation. Point P_A is defined in the coordinate system A, but we need to have it defined in coordinate system B. To achieve so, we apply transformation $_BT^A$, as in (2.9), to point P_A to obtain the point's position in relation to the origin of coordinate system B.

scribed in the system B and $_BT^A$ defined as

$$_BT^A = \begin{bmatrix} _BR^A & _BTr^A \\ 0 & 1 \end{bmatrix}, \tag{2.9}$$

where $_BR^A$ is a 3×3 matrix where each column defines, respectively, the rotation between the x, y and z axes of each system and $_BTr^A = [_BTr_x^A, _BTr_y^A, _BTr_z^A]$ defines the translation vector between the origin of each coordinate system.

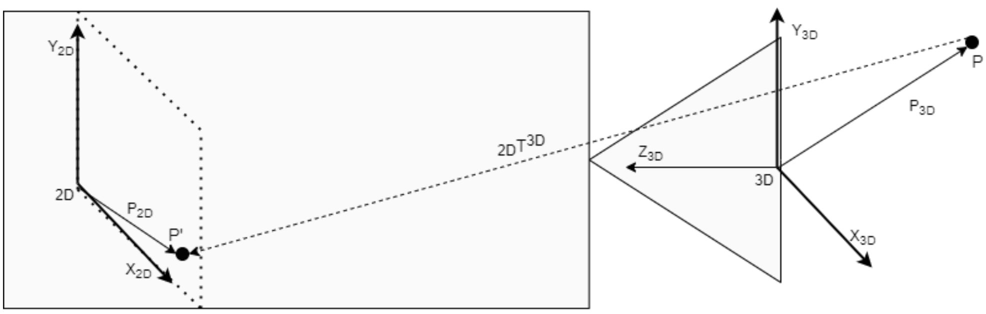

Figure 2.2: Perspective transformation of a point P in a sensor's 3D coordinate system — right — defined by P_{3D} to P' in the sensor's 2D viewport — left — defined by P_{2D}. A point defined in the 3D coordinate system of a sensor is transformed to the sensor's viewport 2D coordinates by applying transformation $_{2D}T^{3D}$, defined in (2.11).

With each object to be projected defined in the sensor's coordinate system, the next step is to project the points that define each object to the sensor's viewport. A sensor's viewport is the 2D area visible to the sensor, limited by the sensor's field-of-view (FoV) and the depth clipping planes that define the depth space in which the camera perceives objects. Thus, the objective of the viewport projection is to determine the tuple $P_{2D} = (x_{2D}, y_{2D})$ from the 3D point definition $P_{3D} = (x_{3D}, y_{3D}, z_{3D})$. Consider Figure 2.2, where we have a point P defined in the sensor's 3D coordinate system, P_{3D}, and we want to determine the corresponding point in the sensor's viewport, defined as P_{2D}. Equation 2.10 shows how P_{2D} can be obtained from P_{3D} defined in homogeneous coordinates.

$$P_{2D} = {}_{2D}T^{3D} \cdot P_{3D}, \tag{2.10}$$

with ${}_{2D}T^{3D}$ defined as follows:

$${}_{2D}T^{3D} = \begin{bmatrix} S & 0 & 0 & 0 \\ 0 & S & 0 & 0 \\ 0 & 0 & N & 1 \\ 0 & 0 & F & 0 \end{bmatrix} \tag{2.11}$$

Where:

$$S = \frac{1}{\tan(\frac{fov}{2} \times \frac{\pi}{180})}, \quad N = -\frac{fp}{fp - np}, \quad F = \frac{fp \times np}{fp - np},$$

with fov, fp and np positive constants that give, respectively, the human's field-of-view and the defined closer and farthest planes between which the sensor perceives the scene.

The point P_{2D} obtained from (2.10) is defined as a tuple with $P_{2D} = (x_{2D}, y_{2D}, z_{2D}, w_{2D})$. The x_{2D}, y_{2D} pair gives the point's 2D position. The z_{2D} coordinate remaps the 3D space into the 2D space, causing objects closer to the 3D system's origin to appear bigger, while objects farther appear smaller. The w_{2D} coordinate maintains the homogeneity of the transformation. Usually, both z_{2D} and w_{2D} coordinates are 1, defining unit vectors, and thus we can extract the $P_{2D} = (x_{2D}, y_{2D})$ coordinates directly from (2.10). However, in special cases, one or both of the (z_{2D}, w_{2D}) coordinates do not define a unit vector and in those cases, before obtaining the 2D coordinates, x_{2D} and y_{2D} must be normalized as follows:

$$x_{2D} = \frac{x_{2D}}{w_{2D} \times z_{2D}}, \quad y_{2D} = \frac{y_{2D}}{w_{2D} \times z_{2D}}.$$

Finally, the viewport defines a square area that only captures the 2D points where x_{2D} and

y_{2D} are comprised between -1 and 1. Thus after obtaining the tuple $P_{2D} = (x_{2D}, y_{2D})$, we must check if the sensor captures the point or not, by checking if the point falls within the -1 and 1 boundaries.

2.3 Markov Decision Processes

A *Markov decision problem* (MDP) is a model for sequential decision problems in stochastic environments. A MDP M is defined as a tuple $\langle X, A, P, R, \gamma \rangle$, with:

- X defining the state space;
- A the action space;
- P the state transition probabilities, where $P(y \mid x, a)$ indicates the probability of moving from state x to state y upon executing action a;
- R is the reward function, defined as $R : X \times A \to \mathbb{R}$, where $R(x,a) = \mathbb{E}\left[R_t \mid X_t = x, A_t = a\right]$, and X_t, A_t and R_t respectively denote the state, action and reward at time t;
- $\gamma \in [0, 1)$ is a discount factor, indicating the relative importance of future rewards against present rewards.

Solving a MDP amounts to computing an *optimal policy* π^*. A policy is a mapping from states to actions describing which action the agent should take in each state, and we can define the *value* associated with a policy as

$$v^\pi(x) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_t \mid X_t = x\right],$$

where X_t and R_t are the state and reward at time step t, respectively. The optimal policy is such that $v^{\pi^*}(x) \geq v^\pi(x)$ for all $x \in X$ and all policies π. The value associated with the optimal policy is denoted v^*, and we define the *optimal Q-function* as

$$q^*(x,a) = r(x,a) + \gamma \sum_{y \in X} P(y \mid x, a) v^*(y).$$

The optimal Q-function can be computed in polynomial time using dynamic programming [80], and the optimal policy can be computed from q^* simply as $\pi^*(x) = \operatorname{argmax}_{a \in A} q^*(x,a)$.

3

Related work

In this chapter we discuss published research relevant to this thesis. We discuss works that investigate the impact of different methods of communication in human-robot interaction. We introduce the discussion incrementally, starting by exploring the importance of communication in HRI and different ways robots have communicated with humans. After the initial discussion, we go over approaches that use explicit and implicit communication. We start with approaches where robots explicitly communicate intent and, therefore, are easier to deploy successfully. We then discuss implicit communication approaches, more complex to deploy because the intent robots communicate must be interpreted by humans. Legibility, a core concept in this thesis, falls in this category. Afterwards, we discuss the impact of different approaches that use legible and predictable movements in HRI, particularly the impact on a robot's expressiveness. Finally, we move to more general human-agent interaction, discussing how implicit communication approaches – and more specifically legibility – have been deployed to improve agents' transparency and interpretability.

3.1 Robotic communication in human-robot interaction

Communication is a crucial part of HRI because it allows humans and robots to interact and understand each other. Communication between humans and robots in HRI has been studied in settings like the use of speech or text to clarify the robot's objectives and intentions [66, 88] or to persuade the humans interacting with the robot to a specific course of action [82]; using facial expressions to convey emotions as a reaction of the human's actions [82]; combining gaze and body movements to augment the teaching and persuasive capabilities of a robot [18, 46]; or use lights to capture a human's attention or signal to a human the robot needs assistance [6, 35].

Correct communication is essential from safety and efficiency perspectives in HRI and in shared human-robot workspaces in general. Clear communication allows humans that share a space with robots, to understand the robot's intentions and its current internal state, improving robot's transparency and thus allowing humans to adapt their behaviours and actions accordingly. Understanding the robot's intentions helps preventing injuries in humans or damages to the robot or surrounding environment due to the emergency changes to the robot's actions [4, 47]. Correct communication impacts the efficiency of the interaction in numerous ways: improving reaction times and human comprehension of the robot's objectives thus improving collaboration efforts [11, 32] or progressing in interaction tasks that require information to be communicated [10, 35].

Communication methods can be divided in two types: explicit communication where the robot deliberately conveys information to a human by being explicit about what it is trying to convey, such as speech or haptic communication; and implicit communication where the robot communicates through its actions and behaviours and not in a deliberate way, requiring that the human partner interprets what is being communicated, such as in the case of body motion or facial expressions. Explicit and implicit communication can occur independently from each other. Explicit communication imposes less cognitive load on the human because the robot explicitly conveys its intentions to the human. However, may cause some awkwardness in the interaction if overused, causing the robot to become extremely verbose contrary to normal human-to-human interactions. In contrast, implicit communication makes interactions feel more natural, given the human tendency to try and interpret social cues from the being humans interact with, even causing humans to trust more the robots and feel more comfortable around them [90].

Although both types of communication can occur separately and independently of each other, humans combine both in order to improve communication and facilitate complex tasks such as those that require coordination and/or collaboration. In such tasks humans might issue a verbal, nodding or pointing instruction and reinforce the instruction with force applied in the direction of the instruction [40].

3.1.1 Explicit communication in human-robot interaction

Given the simplicity of explicit communication in allowing for humans to easily understand the robot's intentions, the use of explicit communication methods has seen widespread usage in HRI [7]. As Bauer et al. presented [7], explicit communication can occur under different communication channels with some of them being speech, explicit gaze, facial expressions and body gestures — such as sign language and hand signals — and haptic communication — communication through the application of force to orient the receiver towards a certain course of action.

Admoni et al. [2] discussed in their literature review that explicit gaze — eye movements that are intentionally expressive, like gaze aversion to communicate thoughtfulness — has been used is several HRI applications to augment speech communication. In the review, the authors explored the usefulness of gaze in human-robot communication analysing the impact of gaze in three categories: how humans perceive robot gaze and how it affects human behaviour; how robot gaze can be used to improve human engagement and task participation; and, finally, what tools exist to generate robot gaze focusing in the mathematical and technological contributions to HRI. The works covered in these three categories offer a vast overview over the use of gaze, specially the first two categories, which focus more on the impacts and perceptions of gaze on humans, offer interesting insights to applications of gaze. Regarding the perception by humans of robot gaze, Admoni et al. found a common trend in the literature regarding gaze in multi-party HRI: people notice more the robot's gaze when it looks at them or close to them. Also, the use of mutual gaze allows the robot to obtain human attention more frequently and direct that attention to objects relevant for the interaction, causing humans to think of the robot as a social entity and the interactions to feel more human-like. In learning from demonstration interactions, the use of mutual gaze makes the human teacher feel that the robot is more interested and invested in the task, leading to the human spending more time teaching the robot and paying more attention to the learning robot [51]. Another interesting conclusion from the work of Admoni et al. is related to robots used in the therapy of children with autism spectrum disorders (ASD), where the use of gaze has shown a positive impact in the therapy leading to some children displaying spontaneous social gaze behaviours in response to robots.

In the works of Correia et al. [24, 25], the authors developed a social robot to play the Sueca card game, a team card game that places two teams of two playing against each other. The robot used a combination of speech and facial expressions with gaze to communicate with its partner and adversaries. The system would use encouraging sentences when addressing its human partner, motivating the human to play better; and competitive utterances and facial expressions when addressing its adversaries or when the adversaries would get the upper hand. During these interactions, the robot gaze would focus on the person being addressed, implicitly indicating to the humans who the robot was addressing. Correia et al. [24] showed that the behaviours used in conjunction with the ability to play well lead positive interactions and to trust in the robot by the humans interacting. Correia et al. [25], followed the previous work with the Sueca playing robot and researched how differences in the social behaviours of a robot impacted humans' choice of teams. Again the robot combined speech, gaze and facial expressions to interact with the humans playing the game. In this study, the authors designed two different robot characters, one that focused more on winning the game and the other focused on fostering team spirit. To build these two significantly different characters, the authors designed the win focused character using more competitive utterances, showing joy when its team was winning and anger when losing; and, the team focused character assuming a more supportive role, displaying sadness instead of anger when loosing. The study conducted, placed the robots playing in opposite teams and each human participant would partner with each robot once and at the end of each interaction the participants would rate each robot. With this work, the team of Correia et al. studied the factors driving humans to choose teams and how the social behaviour exhibited by a robot can impact the choice, concluding that team performance and personal characteristics were the principal aspects that impact the choice. One interesting insight from the work is how the use of explicit communication, in the form of speech and facial expressions, allowed for the design of significantly different robotic characters to which humans ascribed different personalities but also personal preferences.

The work of Lackey et al. [64] shows the use of hand gestures to coordinate a team composed of human military personnel and robots during deployment in different types of military operations. In this work, the authors explore the possibility of using hand gestures by a soldier to order a robotic partner to follow a person of interest and inform about changes, while the soldier moves to a better positioning for the mission's purpose. Hand gestures were also the focus in the work of Kattoju et al. [56], with the exploration of the usage of hand gestures in conjunction with speech commands to again direct a robot during a military operation. In this latter work, the soldier-robot team had to gather intelligence about two buildings surrounded by obstacles, with the human using gestures and simple speech commands to orient the robot's actions. The results of the study show that the use of hand gestures allowed for the robot to better understand the human's orders both when used in alternative to speech and when used simultaneously with speech.

Explicit communication is not restricted solely to the aforementioned channels of communication, although these are the most commonly used. In some of our own work [35], we use lights combined with body movements to allow a robot to communicate with humans. In this work, a non-anthropomorphic spherical robot — Sphero — capable only of emitting light in different colors and rolling around in the floor had to guide human participants from the room the study began to a stand with candies in another room. The scenario in this study consisted of a room arranged to look and feel like a living room where the participant began and a teleoperated Sphero would come in after. The interaction was composed by a brief introduction phase where the human participant could interact freely with the robot and a second phase where the robot would try to guide the participant from the inicial room to another room where the bowl with candies awaited. During the entire interaction the Sphero robot used light signaling combined with body movements to capture the participant's attention, display emotions and finally trying to convince the participant to follow the robot from the initial room to the second one. The results of the interaction show how the combination of explicit communication using light signaling with implicit communication through body movements allowed a simple non-anthropomorphic robot, with few communication capabilities, to display complex behaviours such as convincing a human partner to follow it.

These works, highlight the power in the simplicity of explicit communication methods: allowing to create multiple robotic characters, communicating emotions, give orders in a team and create interactions that feel natural. Although explicit communication channels are the main methods of communication in the works explored, several of them used implicit communication methods to support the communication. Thus, implicit communication, like the body movements in [35], are important in fluid and natural interactions between humans and robots.

3.1.2 Implicit communication in human-robot interaction

While explicit communication has been widely used in robotics to improve robots' communication skills, a common trend in works of HRI is to combine it with some sort of implicit communication channel. In Faria et al. [35] the movements of the robot were combined with the lights to get the participant to follow it and in Lackey et al. [64] the use of gestures was accompanied with eye contact and intensity of contact to convey the urgency of the order. The combination of explicit and implicit methods is important because the use of explicit communication on its own can lead to interactions feeling unnatural, since humans are used to read communication cues from other humans (and animals) to better understand their intentions [40].

Implicit communication is the type of communication that deals with these "communication cues" and uses them to enrich the interaction requiring that the recipient interprets the received signals [7]. Examples of such communications are body posture and movements, implicit gaze such as looking to a person when addressing it, proactive task execution and action planning.

Another aspect that makes implicit communication important is evidenced in the work of Wang et al. [94], where high context cultures — cultures where people have close connections, where most behaviours are not explicit and communication is mostly implicit [45] — like Korean and Chinese cultures, people expect and accept better the use of implicit communication methods over the use of explicit communication.

Implicit communication is extremely powerful, allowing for humans to complete tasks with minimal or even no verbal communication. As in the case of Calisgan et al. [12], where a study showed humans' techniques to complete a collaborative tangram puzzle experiment without using verbal communication. During this study the participants took turns placing or orienting tangram pieces to completely fill a rectangular frame. The authors identified a set of implicit body posture and gaze communication participants used, signaling that their turn was over: resting the hands on the table, crossing the arms, joining hands in front of the body, stepping back from the puzzle or looking up to the partner. All of these implicit cues in combination with other more explicit ones such as pointing to the partner and indicate they could go, allowed the various teams of participants to communicate with each other and complete the puzzles assigned.

Li et al. [67] have also researched the question of non-verbal implicit communication using implicit gaze instead of body movements and posture. The work focus on developing a system to be used by assistive robots in activities of daily living, that uses implicit gaze — gaze cues that allow to interpret the person's intentions and desires — by tracking the user's gaze and detecting what objects the user focus the most. The resulting system was showed to be able to identify the intentions of the participants when looking at a virtual kitchen, showing that implicit gaze is a good tool for robots that have to interact with humans.

Breazeal et al. [10] showed that combining implicit and explicit communication would lead to more effective collaborations than using only explicit communication. In this study, a human would have to guide a robot (Leo) through a task. First the human would teach the names of the buttons to Leo and then ask Leo to turn them on. The task is simple, however possible errors in communication can lead to a cascade effect that if undetected would lead to the task failing. The results show that when implicit and explicit communication were used, the human was able to understand quickly, from the robot's implicit reactions, situations where there was a communication error between himself and the robot. On the contrary, in the case of only explicit communication where the robot only provided information regarding the buttons when prompted, humans took longer to recognize errors in communication. The ability to understand early communications errors created better human mental models of the robot's behaviour and knowledge, leading to shorter task execution times.

In Baraka et al. [6], an autonomous robot — CoBot — navigates a University department. The CoBot is tasked with guiding visitors to their destinations, communicating with humans using speech while using light signaling to reveal its internal state. The transparency of a robot's internal state is an important aspect in human-robot interaction, however at any given time the internal state can include a number of different parameters that humans may not understand. As such, in [6], the authors focused on studying how to reveal relevant information for the interaction about the CoBot's internal state using light signaling. Three scenarios were selected to study the approach: waiting for a human input, blocked by a human obstacle and showing task progress to a user. The three scenarios were chosen because they represent situations the CoBots find themselves in frequently. The results of this work show that using light signaling in these scenarios improved the CoBots' communication capabilities, supporting the speech communication with humans.

Another work that shows the impact and importance of combining implicit and explicit communication methods is the work in the INSIDE project[1], a project whose focus was to develop a robotic system to help in the therapy of children with ASD. The system developed in the INSIDE project [71], used a combination of explicit communication through speech, sounds and facial expressions and implicit communication through body movements, implicit gaze and lights. The system was tested in a hospital during real therapy sessions, where the therapist was assisted by the robot in therapeutic tasks. During these sessions, the therapist and the robot tried to elicit from the child social behaviours like responding to requests, answering to questions, demonstrating empathy, among others.

In one of these activities the robot would find its path blocked by an obstacle and the child would have to remove the obstacle from the way. Before explicitly asking for help, the robot would try to move around the obstacle implicitly communicating to the child that it wanted to progress its movement that way. On another task, to elicit emphatic behaviour in helping the therapist, the therapist would knock over a tower made of blocks and the robot would focus its gaze on the blocks on the floor as an implicit indication the child should help. Besides these specific occasions, where implicit behaviour was the main communication channel, other implicit behaviours where used to enrich the interaction, such as the use of lights to signal if the robot was happy or sad with how the session was progressing. The results from these therapy sessions [71] show how the robot was able to engage in social interaction with children with ASD, showing the potential of robots in aiding the therapy of these disorders.

As we have discussed, implicit communication is essential for a robot that needs to interact with humans, not only in supporting the use of other explicit communication methods, as with [10, 35, 71], but also as a standalone method of communication when other methods of communication are unavailable, as with [12, 67]. Among implicit communication methods, one of the most commonly used is body movements.

[1] http://gaips.inesc-id.pt/inside/

Body movements as implicit communication

Body movements are commonly used given their usefulness as most are cross-cultural and humans are capable of comprehending them without being previously taught their meaning [34]. As seen in Wang et al. [94], cultural upbringing has a significative impact in human perception of robots, robotic communication and impacts human-robot collaboration efforts. Thus, using a method of communication like body movement is necessary for universal integration in societies, because it is of easy comprehension across different cultures. An example of body movements as a useful communication tool is the work of Rodrigues et al. [86], regarding the use of body movements to support the oral discussion of different subjects in human-to-human interactions. In this work, the team observed that body movements were used to get the attention of other humans, show that current speaker has finished speaking and support in conveying the emotions regarding a personal experience.

Body movement is also useful for humans to gather information of people they interact with. As observed by Moral et al. [76], doctors that are trained in using communication cues from body movements, among other implicit communication cues, are able to give better treatment to the patient, who regards them as more respectful, thus improving the physician-patient relation.

Body movements have also showed usefulness in HRI communication, as a primary communication channel. Lee et al. [65] conducted a study where they placed participants in a virtual reality setting of a pedestrian crossing a crosswalk with an autonomous vehicle approaching the crosswalk. The study aimed at testing the impact of the yielding behavior of the vehicle — slowing down before the crosswalk — on the participant's intent to cross and trust the vehicle. The study's results lead to the conclusion that the vehicle's yielding behavior positively impacted the pedestrian's trust in the vehicle stopping, leading to the pedestrian to cross without feeling the need to double check the car. Body movements have shown promise in improving simple non-anthropomorphic robots with few communication capabilities, as in the case of the work previously explored of Faria et al. [35]. In this work, body movements combined with light signaling made possible for a Sphero robot to elicit free form interaction with humans, culminating in the humans following the robot from the room the study started to another room.

As these works have shown, body movements are extremely useful to implicitly communicate intent. However, some movements are easier to understand than others. For example in [35], although people followed the robot, when asked what the robot meant to convey with a movement designed to convey the intention "Follow Me!", only 32% of the participants correctly associated the movement with the intention. Thus, body movements in robots need to be more expressive, making the association between the movement and the intention clearer. This is the idea behind two movements designed by Dragan et al. [29, 31] — *legible* and *predictable* movements — that intend to maximize the information conveyed regarding the robot's intention.

3.2 Legible and predictable movements

Dragan et al. [31] defined *predictable* and *legible* movements, two types of motions that tackle the problem of communicating intentions. Predictable movements aim at making movements that are stereotypical and match expected behavior to achieve an objective; while legible movements aim at making movements that are easy to understand and enable quicker inference of the robot's intent.

The concept of *predictable movement or motion* was defined by Dragan et al in [31] as follows

> **Definition 1** *Predictable movement is a movement that matches what an observer would expect, given the goal G_R.*

While the concept of *legible movement or motion* was defined as

> **Definition 2** *Legible movement is a movement that, when observed by a human, allows the human to infer the robot's goal G_R, with elevated confidence and as quickly as possible.*

We leave the formalization of both definitions to Section 4.1.

The definition of these two types of movements makes them *"fundamentally different and often contradictory"* [31], given how the inference between the action executed and the action's goal is performed.

In the case of predictable movements, the action's goal influences how a human infers the action executed, as such the goal of the robot influences human's expectation regarding its movement. Consider the scenario described in Example 1 where a robot has to fill the cups of water held by John and Mary. If the robot already filled John's cup, both humans would expect the robot's to fill Mary's next and the movement to be as straight as possible towards Mary's cup. So, the goal of filling Mary's cup influences the movement the humans expect to see.

On the other hand, in the case of legible movements the movement of the robot — the action executed — leads to the human inferring the action's goal. As such the robot's action influences human expectation of what is the robot's goal. Considering again the scenario in Example 1. When the robot still has to fill both Mary and John's cups and starts to move towards John's cup, while maximizing the distance towards Mary's, both humans expect the robot to fill John's cup. Thus, the movement influences the humans' expectation about the robot's goal.

3.2.1 Impact of legibility and predictability in human-robot interactions

The works of Dragan et al. on legibility [29, 31] have driven the community to research the impact of legible movements in HRI. Dragan et al. [32] presents one of the first studies that researches the impact of legible and predictable movements in conveying information to humans. In this study the authors compared legible, predictable and movements generated with a general obstacle avoidance motion planner, in a collaborative task where a robot helped a human in fulfilling orders in a coffeeshop scenario. With the study, the team concluded that legible movements lead to more fluid collaborations between the robot and the human, as well as making the human partner feel more comfortable while interacting with the robot.

However, the omniscient view in the initial definition of Dragan et al. did not correctly represent how a human perceives a robot's movement, because the human is limited by the current point-of-view over the task. In order to create a more correct model of the human perspective, Nikolaidis et al. [79] proposed an extension to the legibility model that accounts for human perspective. In the extension, Nikolaidis et al. propose that instead of taking the original trajectory described in the world space, the trajectory be projected into the human viewport before applying the optimization process. Thus, the optimization procedure can produce changes that have a real impact in the human's perception of the task and improve the understanding of the robot's intent.

Besides these initial works, the impact of legibility in human-robot interaction has been researched under numerous premises and assumptions and explored in different scenarios. In a systematic review by Ajomshoae et al. [4], legible motions were considered among the main methods to improve trust and transparency of a robotic system, thus increasing human confidence in robots and team efficiency in human-robot collaboration tasks. The link between legibility and system transparency comes from legibility reducing the ambiguity between possible objectives and making clearer the robot's intentions. As concluded by Alonso et al. [3], improving the system's transparency and corresponding increase of trust in the system makes humans delegate more tasks to robots and decreases errors caused by incorrectly perceive robot's internal state.

Legible and predictable motion applications have been various, exploring different scenarios of human-robot interaction, specially scenarios of human-robot collaboration. These works have focused on single-user tasks to gauge the impact of legible and predictable motions in HRI and given insights in: improving task efficiency through combining legibility and human intention prediction [19, 99]; improving the efficiency of maritime ship maneuvers in human-robot teams [97]; or in developing new communication avenues between humans and multi-robot systems [13].

In the field of HRI, human-multi-robot communication has shown a low development of communication methods for proximal interactions. To tackle this lack of research, Capelli et al. [13] proposed the application of the concept of legibility to the group movements of the robots that compose the system. Thus making easier for the human interacting with the multi-robot system to understand each group's objective. The authors validated the approach using a study conducted in virtual reality, where a human user interacted with multiple multi-robot teams. The results of the study show that using legible movements allowed the users to be more correct and take less time to correctly predict each team's objectives.

Zhu et al. [99] and Chang et al. [19] explore the impact of legibility and prediction of human intention in human-robot collaboration (HRC) tasks. Zhu et al. [99] researched how the application of legibility to the domain of sequential decision-making impacted efficiency in HRC. In this work, the authors designed a collaboration assembly task where the robot had to guide the human towards the correct assembly. The system had to predict, from a pool of possible solutions, what solution the human currently thought was the correct and act accordingly. The results showed that the combination of prediction of the human's objective with a legible choice of what piece to assemble next, lead to less time to complete the task and less errors than when combining prediction with a simple efficiency based approach. The work of Chang et al. [19] focus on the direct impact of legibility and intention prediction in collaborative manipulation tasks. The authors compare the impact of intention recognition against no intention recognition and legible versus predictable movements in a task where the user and the robot have to empty two cups into corresponding bins. The results of this study suggest that the combination of intention recognition with legible motions may lead to improved collaboration efforts in tasks with concurrent objectives.

Another field where legibility has showed positive impacts is in the field of driving autonomous vehicles. In [97], Woerner et al. explored the use of legibility in autonomous vessels to improve communication with human driven vessels in maneuvers of overtaking and head-on or crossing encounters. Given the restrictions imposed by the COLREGS rules — set of internationally agreed rules for collision regulation — the only possible communication through movement for vessels navigating in open waters is a change in heading — either turning slightly to the left or right — or change in velocity. The system was tested by placing one human controlled vessel interacting with five autonomous vessels, with each vessel having to reach a point in a straight line across from the starting position. The placement of each vessel was planned to cause simultaneous overtaking, head-on and crossing encounters. Using legible movements, the authors designed a system capable of interacting with human driven vessels in open water and flag vessels that do not conform with COLREGS rules.

The effects of legible and predictable motions in HRI have been widely researched, with a clear tendency for legible motions to improve the conveyance of intent by robots. The literature spans various areas, from the impact in collaboration tasks to the driving of autonomous land and maritime vehicles. However, most works focus on researching the impact of legible motions in interactions between one human and one or multiple robots. The impact of legible motions in the interaction between one robot and multiple human users is largely unexplored, and needs to be investigated, because these interactions are common in society and correct communication in such occurrences plays a crucial point in the integration of robots in society. More broadly, legibility can play a crucial role in rendering the decision process of artificial agents more interpretable by human users. The extension of legibility beyond robot to more general decision processes is, therefore, an important and unexplored avenue for research.

3.3 Legibility in decision-making

Explainable artificial intelligence (XAI) is a line of research focused on build intelligent systems that are capable of explaining to humans the reasoning behind their behaviours, decisions, actions and recommendations. However, each XAI system is defined for a specific context because explanations are context dependant and also depend on the expectations of the system's user [43].

The importance of XAI systems is a fact known since when artificial intelligence (AI) systems were mostly developed as expert systems that supported human decision making, through the use of a knowledge base. However, XAI has received new attention from research and industry in recent years [28]; motivated by systems with the ability to provide explanations, about their internal decision processes, being preferred by users, developers and regulators over systems without these abilities [23].

Most approaches to improve AIAs' transparency have focused on making more transparent technical aspects of AIAs such as the reason behind application failures [5, 14, 28], or the inference processes of complex decision process like deep learning approaches [1, 89, 95]. However, most of this approaches focus on the interpretability of AIAs and in making agents explainable from a user perspective, *i.e.* the agent justifies its actions by presenting its reasoning after deciding, instead of making the reasoning clear while deciding. The growing intertwining of AIAs with society, has extended the use of these agents beyond simple tools and applications we use, to become peers and collaborators with whom humans interact. In this sense, AIAs need to be transparent during the entire decision process instead of just at the end of the interaction [4]. This avenue of research has been explored in the work of Chakraborti, Sreedharan, and Kambhampati [15], where the authors propose a planning technique named *MEGA* (Multi-model Explanation Generation Algorithm) that balances generated plans with the explanations needed for that plan to be optimal for an external observer. MEGA offers a planning approach that does not require the external observer to ask the agent to explain its internal state, but offers it as a complement to the generated plan thus making the agent more transparent. However, during an interaction a human or other external observer, an agent cannot simply explain its reasoning in an explicit manner: not only because it makes the interaction feel unnatural [10, 40], but also because the agent's partner may be occupied with another task and cannot give the agent time to read its explanation. So, for an interaction feel natural an autonomous agent must be naturally transparent and its actions easily explainable by the agent's partner's model with little need for the agent to provide explicit explanations [87].

The ability of legible motions to shape motions to become more expressive has made this notion a good candidate to create more transparent AIA systems [4]. Alonso and De La Puente [3] present a review of literature that gathers several works on transparency in shared autonomy workspace and explore different methods of increasing transparency. In this review, the authors conclude that legibility is one of the most common methods to increase transparency by modelling the system's behaviours to become easily interpretable and understood by external observers. Chakraborti et al. [16] provide a taxonomy of the concepts of legibility, explicability and predictability, which have had some overlapping applications. The authors define plan explicability measures how close a plan is to the expectations of an observer regarding a specific goal or planning problem, and plan predictability as a special case of plan explicability where the plan is explicable and unambiguous about possible plans. In contrast, plan legibility focus on reducing the ambiguity over possible goals without prior knowledge of the agent's goal, so plan legibility is closely related to plan transparency.

The usefulness of legibility in making more transparent robots has led to legibility being used beyond robotic motions. Habibian and Losey [44] designed a framework to divide the subtasks of a more overarching task between humans and robots, using legible and fair allocations. The designed framework does a bilevel optimization: the first level is an optimization for the allocation of the subtasks, such that the human clearly understands what the robots' are doing and what subtasks are left for the human to do; then the second level of optimization focuses on optimizing the robots' motions to create motions that are in line with the decided allocation and thus, when observed by the human, allow him to understand what are the robots' intentions. With this work, we can observe how the notion of legibility can be applied beyond robotic motions to create allocations that are easily understood by humans, making easier for humans to understand their role in the task and how they can collaborate.

Legibility has been successfully applied to decision-making in scenarios where action outcome is deterministic. MacNally et al. [70] formalize the problem of legible decision-making as a *Goal partially observable Markov decision problem (POMDP)* [53], where the agent's goal is to choose the actions that transform an initial state belief b_0 into the goal belief state b_G. Using this formalization, the authors design a method for action selection, in deterministic scenarios, that chooses the sequence of actions that constitutes the plan that achieves b_G and best disambiguates the intended goal state from other possible goal states. Kulkarni, Srivastava, and Kambhampati [63] also explored the use of legibility for decision-making in deterministic scenarios, focusing on its application in adversarial and cooperative environments. The authors propose a framework that uses legibility to create more transparent plans in cooperative tasks and to obfuscate plans when the agent is in an adversarial environment. In the proposed framework, the authors use a classical planning formulation where the heuristic function – that helps in deciding the next action to add to the plan – uses the concept of legibility to measure the ambiguity of each possible action and choose the less ambiguous action (in cooperative environments) or the most ambiguous actions (in adversarial environments). Following the work of Chakraborti et al. [16], Sreedharan et al. [92] propose a general framework for the generation of explicable, legible or predictable plans in deterministic scenarios. The users define *Generalized Human-Aware Planning Problem (G-HAP)* as a planning problem where the underlying cost function defines the type of plan the agent will generate.

The notion of legibility has also been applied to scenarios of planning under uncertainty.

Miura, Cohen, and Zilberstein [75] present a formulation of legibility for MDPs, named legible Markov decision problem (L-MDP). In L-MDP, the agent focus on choosing, at each time step, the most optimal action that also maximizes the information transmitted to an observer about the agent's goal. To accomplish such optimal and legible behaviour, the agent reasons about the observer's belief of the agent's objective given the history of the observed agent actions. The observer's belief regarding the agent's intentions is modelled using the multiagent framework of interactive POMDPs (I-POMDPs) [41]. By reasoning about the observer's belief – using this reasoning to drive the planning algorithm – the agent can derive a *legible policy* that best disambiguates the agent's goal [74]. The legible policy is obtained by iteratively updating the assumed observer's belief and with the updated belief simulate the possible actions – using a method like UCT [60] – to find the one that best disambiguates the agent's goal. The results of a user study, conducted by the authors, showed that the resulting legible policies are capable of better transmitting the agent's intentions than using standard optimal policies. However, the nature of legible Markov decision problem (L-MDP) makes its planning complexity similar to that of POMDP, limiting its applications to small scale state spaces as the planning can become intractable in large scale state spaces.

4

Legibile movements in multi-party scenarios

In this chapter we introduce our first contribution: an extension of the notion of legibility to scenarios involving the interaction with multiple users. We start in Section 4.1 with the formal definition of predictable and legibile motions, according to Dragan et al. [31], whose impact in HRI we discuss in Section 3.2.1. Section 4.2 presents an exploratory study on the impact of legible motions in multi-party interactions. In Section 4.3 we contribute *multi-user legibility (MUL)*, an extension of the notion of legibility for multi-party interactions that takes into consideration that each user interacting with the robot perceives the movement differently. In the same section we present the results of a study that addresses Question **RQ 1** and shows the positive impact of MUL on human-robot interactions. Finally, in Section 4.4, we tackle the problem of generating legible movements for multi-party interactions in joint space. This is a problem motivated by the formulation of legibility considering trajectories defined in cartesian space, which may lead to movements unfeasible by the robot and require the movement to be adapted, losing some of the legibility.

4.1 Definition of predictable and legible motions

Both *predictable* and *legible* motions, as defined by Dragan et al. in [31], are based on humans applying the principle of rational action to robots [26, 38]. The principle of rational action, as formulated by Karl Popper [84], states that a rational agent will act efficiently and justifiably to achieve its goal. As such, by applying such principle to robots, humans expect robots to move as rationally as possible to achieve their goals, with a reason behind every action.

A predictable motion is an "efficient" movement and tries to minimize an arbitrary cost function C. It is designed to be not surprising and to match what a human expects the agent to optimize, making use of the principle of rational action. The inference of the goal of a predictable motion can be modeled as

$$\mathcal{I}_P(G_R) = \underset{\xi \in \Xi_{S \to G_R}}{\operatorname{argmin}}\ C(\xi),$$

where G_R denotes the robot's goal and Ξ the set of trajectories towards G_R. This expression allows to design a metric that evaluates how predictable a given trajectory ξ as follows

$$\mathrm{Pred}(\xi) = \exp(-C(\xi)) \tag{4.1}$$

A legible motion uses the human application of rational action over robots, by shaping a robot's movement to emphasize the movement's objective — if a robot does not move "efficiently" to the goal there must be a reason behind such behavior. To shape the trajectory, legible motions leverage animation principles to create trajectories that are readable [93] and that encourage "anticipatory" movements [39]. Using these principles, a legible movement conveys the most relevant information for goal prediction in the beginning of the movement — by bringing the robot as close as possible to its objective, while moving as far away as possible from other possible objectives. Thus, allowing the user to infer the goal from a snippet of the movement observed. In legible movements, goal inference can be modeled as

$$\mathcal{I}_L(\xi_{S\to\xi(t)}) = \operatorname*{argmax}_{G\in\mathcal{G}} P(G|\xi_{S\to\xi(t)}), \tag{4.2}$$

where $\xi_{S\to\xi(t)}$ denotes the observed snippet of movement, $\mathcal{G}$ the set of possible goals and $P(G|\xi_{S\to\xi(t)})$ models the probability of goal G being the objective of the snippet observed. $P(G|\xi_{S\to\xi(t)})$ is computed using Bayes' Rule,

$$P(G|\xi_{S\to\xi(t)}) \propto P(\xi_{S\to\xi(t)}|G)P(G), \tag{4.3}$$

with $P(G)$ the prior on the goals and $P(\xi_{S\to\xi(t)}|G)$ modeling the probability of the observed trajectory snippet being observed when the robot is moving towards goal G as

$$P(\xi_{S\to\xi(t)}|G) = \frac{\int_{\xi_{\xi(t)\to G}} P(\xi_{S\to\xi(t)\to G})}{\int_{\xi_{S\to G}} P(\xi_{S\to G})} = \frac{P(\xi_{S\to\xi(t)}) \int_{\xi_{\xi(t)\to G}} P(\xi_{\xi(t)\to G})}{\int_{\xi_{S\to G}} P(\xi_{S\to G})}. \tag{4.4}$$

Assuming the use of the principle of rational action, the user will expect the robot to minimize a cost function C. However, it is unlikely that all other possible trajectories will surprise a human observer. Thus, $P(\xi_{A\to B})$ can be approximated by $\exp(-C(\xi_{A\to B}))$ and with this substitution, (4.4) becomes

$$P(\xi_{S\to\xi(t)}|G) \propto \frac{\exp(-C(\xi_{S\to\xi(t)})) \int_{\xi_{\xi(t)\to G}} \exp(-C(\xi_{\xi(t)\to G}))}{\int_{\xi_{S\to G}} \exp(-C(\xi_{S\to G}))}. \tag{4.5}$$

As shown by Dragan et al. in [30], to avoid computing the integrals in (4.5) the probabilities can be approximated using Laplace's method. By approximating C as a quadratic, its Hessian is constant, and according to Laplace's method,

$$\int_{\xi_{A\to B}} P(\xi_{A\to B}) \approx k\ \exp(-C(\xi^*_{A\to B})),$$

with k a constant and $\xi^*_{A\to B}$ the optimal trajectory starting in A with goal state B. Applying this change into (4.5) and using it in (4.3) we obtain:

$$P(G|\xi_{S\to\xi(t)}) \propto \frac{\exp(-C(\xi_{S\to\xi(t)}) - C(\xi^*_{\xi(t)\to G}))}{\exp(-C(\xi^*_{S\to G}))} P(G). \tag{4.6}$$

The expression in (4.2) allows the design of a metric that evaluates the legibility of a movement and choose the most legible movement. A legible movement allows the human observer to predict the robot's goal correctly, confidently and as quickly as possible. So, a legibility metric must give high values to trajectories that have high $P(G|\xi_{S\to\xi(t)})$ and achieve these high values as early as possible. The legibility metric can then defined as

$$\mathrm{Leg}(\xi) = \frac{\int P(G_R|\xi_{S\to\xi(t)})h(t)dt}{\int h(t)dt}, \tag{4.7}$$

where $P(G_R|\xi_{S\to\xi(t)})$ is as defined in (4.6) normalized across all goals in $\mathcal{G}$ and the function h is a weighting function, that gives more weight to earlier parts of the trajectory.

A following work by Dragan et al. [29], proposed a method that uses (4.1) and (4.7) to generate either predictable or legible motions using a gradient ascent approach. To generate legible motions, in each iteration, the process improves the legibility score of the trajectory ξ following

$$\xi_{i+1} = \xi_i + \frac{1}{\eta} M^{-1} \nabla \mathrm{Leg}(\xi),$$

where M is used to measure the norm of a trajectory such that $\|\xi\|^2_M = \xi^T M \xi$ and $\nabla\mathrm{Leg}(\xi)$ is the gradient of (4.7). To generate predictable movements the equation uses the gradient of (4.1).

In the same work, the authors define a possible cost function C as the sum of squared velocities given by

$$C(\xi) = \frac{1}{2}\int \|\xi'(t)\|^2 dt$$

that encourages smooth trajectories that go straight to the goal.

4.2 Legibility in multi-party interactions

Not much work has been done in exploring if the results obtained in single user interactions also apply to multi-party interactions. In multi-party scenarios the human partners are not solely focused on the robot to read cues regarding their intentions, but also have to account for the

Figure 4.1: Scenario layout. The robot on the right serves each of the users on the left. The cups are filled by no particular order and the participants should respond to the robot's movement.

other human (or robot) partners and deduce their respective objectives.

Thus, we conducted an exploratory study where we compared the performance of legible motions with predictable motions. We chose predictable motions as the control group because, according to [32], this is one of the preferential types of motion for collaboration tasks and also because this type of motions aims at being efficient at the eyes of a human. In this study, we used a cafeteria like scenario where an autonomous robot, serving as a "bartender", has to pour a cup of water to one of several users. In our scenario, a Baxter robot interacts simultaneously with multiple human users, successively pouring water in the cups held by the users (see Fig. 4.1). As the human users tend to approach the bartender when they believe that they will be served next, interaction is more effective if the motion of the robot can be easily interpreted by the different users.

In order to address this scenario, we developed an interaction control system for the Baxter that comprises three modules: a *vision module* that identifies the position of each cup using a Kinect camera; a *decision module* that selects which cup to fill next and generates the corresponding movement; and a *social interaction module*, responsible for making the interaction feel more natural by using facial expressions and/or speech.

The decision module uses ProMPs to generate the robot's serving motion. Serving motions are learned from demonstration and designed to take into account principles of predictability and legibility. In particular, at each moment the decision module decides the cup to serve and generates a *predictable motion* or a *legible motion*.

Besides the two models of legible and predictable motions, we designed and compared a third type of motion, dubbed the *hybrid motion*. This third motion, described in Section 4.2.1, uses an empirically designed rule base system to decide to execute either a predictable or a legible motion, depending of factors such as the number of cups remaining to serve and the proximity between the intended cup and other cups to serve. This hybrid motion tries to maximize the robot's expressiveness, by taking advantage of the changing configuration of the workspace during interaction.

Section 4.2.2 presents the main results and insights from this first exploratory study.

4.2.1 Hybrid movement

During an interaction, the robot should be able to select, *in runtime*, whether a legible motion or predictable motion is more adequate, depending on the current context of task. We thus developed an approach that, given a target cup and the task context (e.g., the state of the other cups and the position of the different users), decides whether to perform a legible or a predictable motion. The selection is done by checking whether a legible movement towards the target cup would be more expressive than executing a predictable movement. This verification relies on the following criteria:

- When the intended target is closely surrounded by other possible targets, a more direct (predictable) movement is preferable to a wider (legible) movement;
- When the intended target has other possible targets on one side alone, a legible movement from the side with no other targets is preferred to a more predictable movement;
- When the target has other possible targets nearby, on the side that the robot will approach — e.g., when the robot is reaching the leftmost cup with the right arm and there are other cups on the right of the objective cup — a predictable movement is preferred to a legible movement;
- Finally, if there is only one target remaining or there is no ambiguity regarding intended target, a predictable movement is preferred.

To determine when a target is sufficiently close to affect the expressiveness of a legible movement, we tested different configurations and found that, for distances smaller than 50cm between possible targets, people find legible movements to be more confusing than predictable

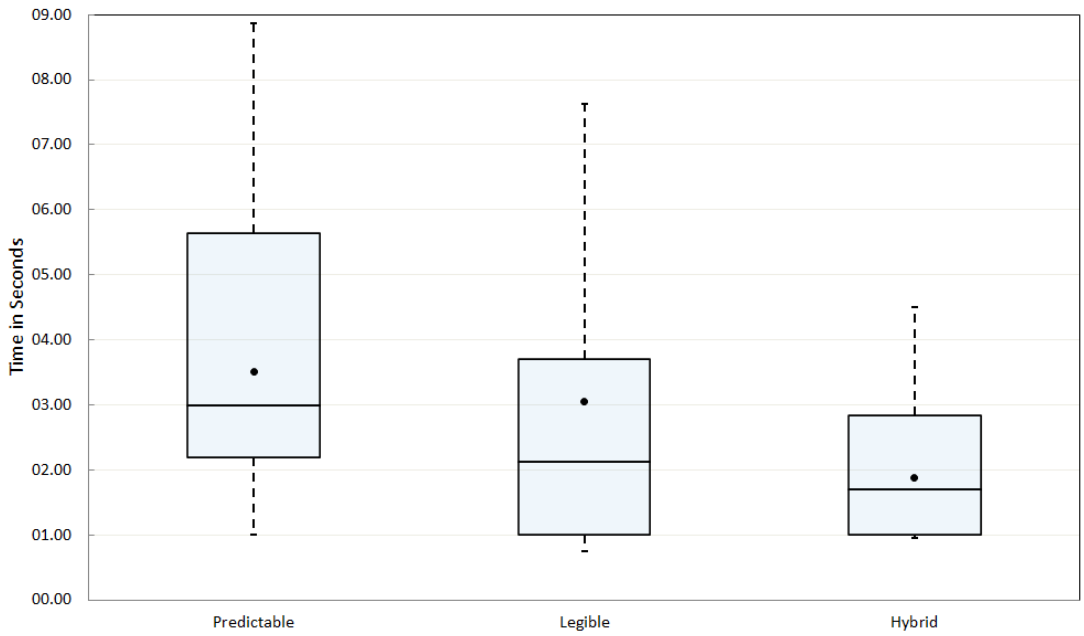

Figure 4.2: Average time that each participant took to understand that the robot was serving him/her, organized per movement type.

movements. We used this value as the minimal distance that a cup should be from other cups on the side of the executing robot arm for the robot to execute a legible motion. We also concluded that when the distance between targets is close to the diameter of a cup, a predictable motion is preferred to a legible one.

4.2.2 Discussion of the study results

The exploratory study had a total of 33 participants, recruited from the Lisbon area, out of which 22 were male and 11 were female. Their ages ranged from 19 to 33 years old, with a mean age of 23 years old. The participants were randomly matched in groups of three, according to the availability of each participant. Each group of participants interacted with the robot three times, one for each motion type (predictable, legible and hybrid). The order of the different motion types was randomly selected, to prevent influence across groups of users. And each interaction consisted of three movements of the robot, one for each cup. For the purpose of the study, and to prevent a serving pattern that the participants could exploit, the robot randomly selected the next cup among those that were reachable and not yet served. Such random selection also forced the participants to be focused on the motion of the robot.

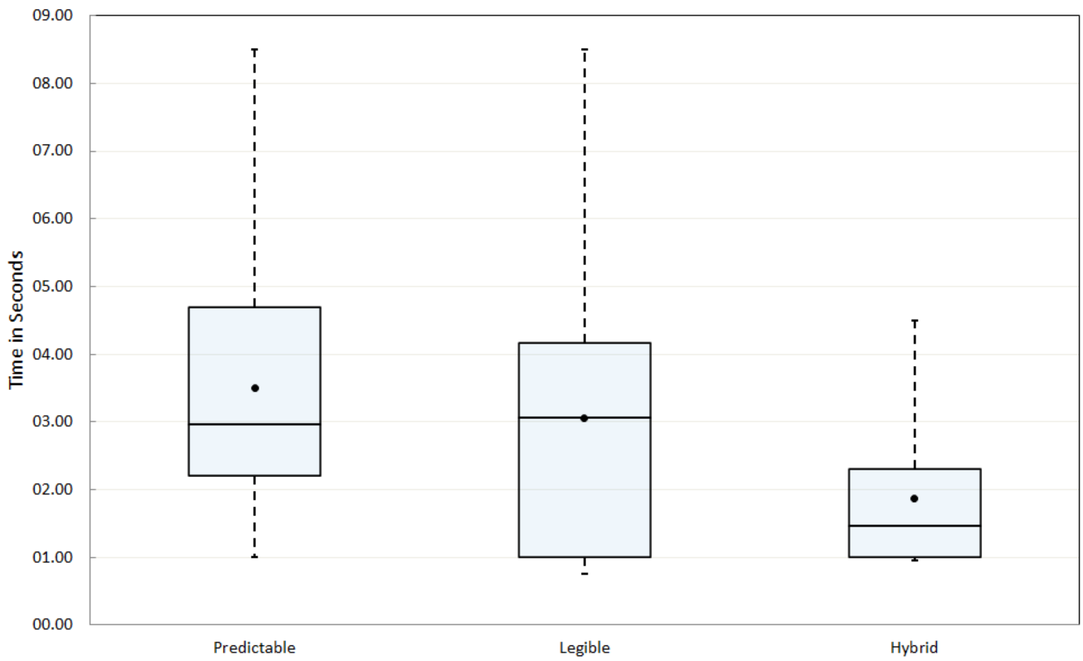

Figure 4.3: Average time that each participant took to understand that the robot was *not* serving him/her, organized per movement type.

To evaluate the performance of legible motions in multi-party scenarios, we analyzed objective measures such as time taken to recognize either if the robot was moving to fill the cup held by the user or held by another user (see Figures 4.2 and 4.3) and the number of wrong predictions for each user. Through the administering of a subjective questionnaire after each interaction, we also evaluated the perceived animacy, intelligence and capability of the robot as well as the perceived fluency of the task.

The results of this first exploratory study show agreement with previous results in the literature, namely the conclusions of Dragan et al. [32] regarding how legible motions allow human users to have a better grasp of the robot's intentions. However, some results give unexpected insights regarding the use of legible motions in multiple user interactions.

The first result that we find important, and also unexpected, is that there was no significant difference, in terms of wrong predictions and of time that took between the robot starting to move and people understood the robot's objective, between predictable and legible motions. On the contrary, the hybrid motion allowed users to take significantly less time to react when compared to both legible and predictable motion. This is interesting and unexpected because, following prior work, we expected legible motions to contribute for less reaction times and total task times than the predictable motions. However, this lack of difference leads us to conclude that 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.

Further analysis of the differences in performance between legible and predictable motions, shows that legible motions have a tendency to lead to faster reaction times than predictable motions for users to recognize they are the target of the movement, see Figure 4.2. However the same tendency almost disappears when users have to recognize they are not the target of the movement, see Figure 4.3. These results point to the users' perception influencing how legible movements convey intention, since movements that are not focused on the user are not as informative as the ones focused on the user.

Another result that is important is that giving the robot the freedom to choose between executing a predictable and a legible motion, depending on the workspace configuration, leads to better collaborations, both in terms of the time it takes for the participants to understand the robot's intentions and in terms of them to "read" the movement and adapt better to it.

The fact that both the perceived capability and intelligence of the robot did not show a significant difference is interesting, since it proves that even if the robot does a less natural or less rational movement - like executing a more wide movement - as long as it behaves as supposed (it fulfills its collaboration role) people think he is capable and intelligent.

Overall, the results of this study suggest that legible motions have a positive impact on multi-party interactions, with hybrid motions leading to consistently faster reaction times and to standard legible motions showing a tendency to lead to faster reaction times. Another aspect that supports the positive impact of legible motions is that predictable motions lead to more task errors than legible and hybrid motions combined. Although the difference in errors was not statistically different, suggests at predictable motions being more confusing.

4.3 Multi-party legibility

As discussed in Section 3.2 and concluded in our initial exploratory study on legibility in multi-party scenarios, the use of legible motions allows for users to better understand the robot's intentions.

However, in some multi-party interactions the robot interacts simultaneously with all the users, each with a different PoV over the task. For example, in the scenario of our initial study a robot sequentially serves cups of water held by different human users standing around the robot; in [24] a robot plays a game of cards, simultaneously interacting with three human users sitting in different sides of the same table; [54] describe a scenario in which a robot is deployed as part of a surgical team to support the staff. In all of these scenarios, for a robot to use legible motions it must be able to generate movements that are simultaneously legible for all partners involved; otherwise, it could optimize the legibility for one partner but reduce the legibility for another one, causing deception regarding its intentions to the other partners.

Having different PoVs over the same movement causes a movement to be perceived differently from each and for the movement to be legible for all the legibility metric should be influenced by the legibility for each PoV. Thus, we propose that the standard single-user legibility (SUL) model should be extended to a multi-user legibility model, where the legibility metric is a combination of the perceived legibilities for each point-of-view. Under this MUL model, the resulting movements would not maximize individual legibility, as happens with SUL, but would maximize the combined group legibility, creating movements that make all the users better understand the robot's movement and not just part of them.

In order to improve group legibility, MUL needs to incorporate information regarding the perceived legibility of each of the task's users. However, the integration of information regarding the different perceived legibilities must be such that no user is favored over the other users. Otherwise, we could fall back in the situation of single-user legibility where the robot's movement gives more information to part of the users. Thus, the MUL model averages the perceived legibilities of the users giving equal weight to all users. We decided to give equal weight to all the users because different PoVs give different perspectives over the workspace and may contribute to better or worse information. Thus, without previous knowledge regarding which perspectives are better, attributing different weights to each PoV could give more importance to perspectives that offer worse information and create trajectories that would decrease legible. Also, by giving the same weight to all users' perspectives we can guarantee that the movement would always be kept in the FoV of each user, thus keeping the resulting movement always visible for all users.

4.3.1 Definition of multi-party legibility

We now formalize the notion of *multi-user legibility*. This notion is based on the single-user legibility notion from Nikolaidis et al. [79], an extension to the definition by Dragan et al., that projects the trajectory to the user's PoV. This extension allows for the resulting trajectory to be more expressive for the user's perspective than the omniscient view used by Dragan et al.

Consider a trajectory ξ_W, defined in Cartesian space, as

$$\xi_W = [[x_1, y_1, z_1], [x_2, y_2, z_2], \ldots, [x_T, y_T, z_T]] \tag{4.8}$$

where T is the number of time points in the trajectory. In MUL, the legibility of trajectory ξ_W in a setting comprising N users is defined as

$$\mathrm{Leg}_{MUL}(\xi) = \frac{1}{N}\sum_{n=1}^{N} \mathrm{Leg}_n(\xi_W),$$

where $\mathrm{Leg}_n(\xi)$ is the single-user legibility of trajectory ξ as perceived by user n, which we refer to as Leg_{SUL_n}. According to Nikolaidis et al., for Leg_{SUL_n} to create legible movements we must take into account how the movement is viewed from the PoV of the user. Thus, Nikolaidis proposes, in (4.6), projecting the robot's trajectory from the world space to the user's viewport before computing the cost function, thus capturing the observing user's expectation in the user's viewplane instead of in the world space. This defines a new cost function $\bar{C}(\xi)$ as follows

$$\bar{C}(\xi_W) = C(\xi_{2D}) = C({}_{2D}T^H(\xi_W)) \tag{4.9}$$

$${}_{2D}T^W(\xi_W)) = {}_{2D}T^H \cdot {}_H T^W \cdot \xi_W \tag{4.10}$$

where ${}_{2D}T^H$ defines a viewport projection between from the 3D human coordinate system and ${}_H T^W$ defines the transformation between the 3D world coordinate system and the 3D human coordinate system, following Section 2.2.

Plugging the legibility expression of MUL, the update step for the optimization in Equation 4.1 becomes

$$\xi_{i+1} = \xi_i + \frac{1}{\eta} M^{-1} \nabla \mathrm{Leg}_{MUL}(\xi), \tag{4.11}$$

with

$$\nabla \mathrm{Leg}_{MUL}(\xi) = \frac{1}{N}\sum_{n=1}^{N} R_n^{-1} \cdot \nabla \mathrm{Leg}_{SUL_n}(\xi), \tag{4.12}$$

R_n^{-1} is the inverse of the rotation between the world coordinate space where ξ was defined and the nth human's coordinate space in homogeneous coordinates and

$$\nabla \mathrm{Leg}_{SUL_n}(\xi) = \nabla \mathrm{Leg}_n(\xi_{2D}(t)) \cdot \nabla P_n(\xi(t)),$$

where $\nabla P_n(\xi(t))$ is the gradient for the $3D$ to $2D$ perspective transformation of the trajectory ξ to for human partner n, accounting for how the trajectory varies in the user's perspective during the task, defined as

$$\nabla P_n(\xi(t)) = \begin{bmatrix} \frac{S}{(FNz_t^2 - Fz_t)} & 0 & \frac{-Sx_t(2Nz_t-1)}{N^2Fz_t^4-2NFz_t^3+Fz_t^2} \\ \\ 0 & \frac{S}{(FNz_t^2 - Fz_t)} & \frac{-Sy_t(2Nz_t-1)}{N^2Fz_t^4-2NFz_t^3+Fz_t^2} \end{bmatrix},$$

having S, N and F as defined in Section 2.2 and $\nabla \mathrm{Leg}_n(\xi_{2D}(t))$ the gradient of the legibility in (4.7), as described by Dragan et al. in [29], for the trajectory defined in the nth human user viewport.

Applying R_n^{-1} converts $\nabla \mathrm{Leg}_{SUL_n}(\xi)$, defined in the 3D human coordinate system, back to the original world orientation, allowing to combine the gradients from the different perspectives.

4.3.2 Simulations

Having our model defined, the next step is to guarantee that our model works properly. Thus we compared the performance of the model versus the SUL model by simulating various scenarios of collaborative interactions. In these simulations we compared the legibility metrics of each trajectory after being optimized either with the MUL model or with the SUL model. We chose a scenario with three different objects organized on top of a table and three human observers. Considering that in this scenario we had three observers, we optimized the movement once using the MUL model and three times using the SUL model — each time taking into account the PoV of a different human observer.

To test the resilience of the model, we optimized the movements for each of the three objects on the table to account for changes caused from reaching for different objects in the same configuration of objects. Another important aspect of the resilience is to changes in the workspace, so we also simulated for different configurations of the objects on the table, to account for different relative placements of the objects and their impact of the resulting movements.

In total, we performed 48 optimizations, 12 using MUL and 36 using SUL — 12 for each human PoV. We then computed the average group legibility for each of the 48 optimized trajectories.

We decided to compare each MUL optimization with the SUL optimizations for the same configuration and target, resulting in 36 comparisons, because the model's objective is to optimize the overall legibility for the group. The comparisons show that MUL generated trajectories achieved a better group legibility in 69% of the optimizations (25 out of 36). Also, several optimizations using SUL created trajectories that would go outside one or multiple of the other partners' field-of-view and with MUL that did not occur, thus resulting in trajectories that are safer to interact with, since they are always in view.

Finally, regarding the 11 cases where a SUL optimization resulted in a better group legibility than MUL, this might be due to a human having a privileged perspective on the task — the human partner's FoV gives a very good perspective over the workspace — allowing SUL to take better advantage of the PoV than MUL. The reason for this is that MUL equally considers the contributions of each user's PoV and as such, if there is the special case of a PoV offering a particularly advantageous perspective of the task, that PoV is weighted down by the others less advantageous PoVs.

4.3.3 Experimental evaluation

To explore whether considering the combination of the different users' points-of-view positively impacted the perceived legibility of a movement, we conducted a study through Amazon's M-Turk[1]. In this study, each participant watched 3 sets of videos of trajectories executed by a robot. The trajectories observed were optimized using MUL or SUL, testing how the trajectories generated with the two approaches differed when observed by humans.

Setup

In the study conducted, we focus on the impact of robot motion. So, we designed a task setup that minimizes distractions from the movements. We replicate a scenario where the robot has to execute a collaboration task with 3 other humans, thus creating an experience similar to what would happen in real life with multiple users.

The scenario resembles a room in a house, where the robot and the three users are placed around a table with three objects on top of it - a soda can, a rubber duck and a telephone. The three objects were placed in a single line, evenly spaced between them and with the humans equally distant from the table. User 1 is placed across from the robot, looking in its direction and Users 2 and 3 are placed on each side of the robot, facing each other. Figure 4.4 shows a diagram of the setup.

Given the focus in developing expressive movements, the scenario was reduced to a section

[1] https://www.mturk.com/

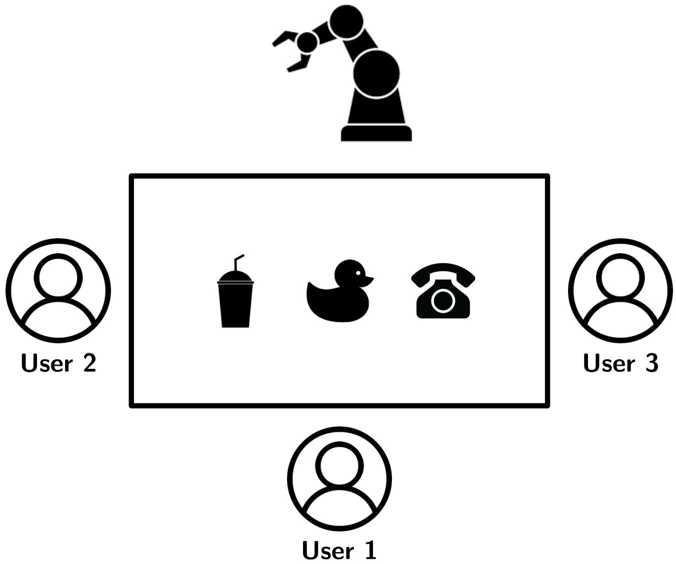

Figure 4.4: Setup of the user study. In the center a table with three objects on it – a soda can, a rubber duck and a telephone – and around the table three users looking towards the objects and the robot. Each of this users has a different point-of-view allowing to evaluate the perception of each user regarding the movements of the robot.

where the robot has to pick an object on top of a table, with the human partners observing the movement from different PoVs. By focusing on just the movement, we could remove distractions unrelated with the movements, giving more focus on the impact of the movement on the users' perception. With the three users in different sides of the table, the movements are perceived with significant differences because of the difference between the PoVs. Thus, we can study the impact different optimizations have on how each user perceives the resulting movements, specially we can compare how optimizations that focus on one PoV fare against optimizations that consider all the PoVs available. Also, by reducing the scope of the interaction to just the movement of picking one object, we reduce the amount of perceptual noise focusing on the movements and the participants have less contextual information to help them in predicting the robot's objective, giving more focus to the movements and the information conveyed by them.

Design and hypotheses

As discussed previously, we developed this model to explore the research question:

In interactions with multiple users simultaneously, does combining their perceived views of a robot's movement improve the movement's legibility?

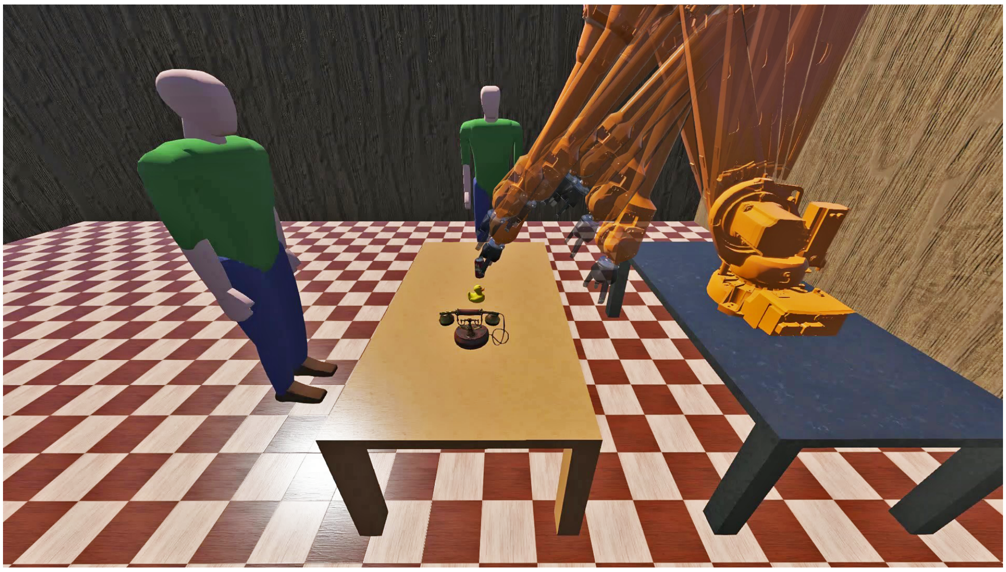

Figure 4.5: Example of a movement towards the soda can, as seen from the point-of-view of User 3, starting with the robot arm in a rest position, with the gripper hovering between the robot and the table.

Hypothesizing that:

H1 *Humans will consider a movement generated using multi-user legibility clearer than when generated using single-user legibility*.

H2 *Humans will understand quicker and more confidently the robot's target, when faced with a movement generated using multi-user legibility than with one generated with single-user legibility*.

The study followed a between-subjects, with each group being a different optimization approach — optimized with MUL or optimized with SUL. The choice for a between-subjects design aimed to prevent a lengthy and tiring questionnaire given that to ensure an equal distribution of participants across the different conditions and points-of-view each participant would have to watch several different videos, which would cause a risk of decreased performance by the end of the questionnaire. Finally, because there were three different PoVs, there were three different SUL movements for each of the objects stemming from optimizing the movement for each PoV.

So, each participant watched 3 sets of 3 videos, each video with different length — 6, 12 and 18 seconds, all using either MUL or SUL optimized movements. After each video the participant was asked to predict what object the robot was going to grab and rate how confident they were in their prediction. At the end of each set, the participant would watch a 20 seconds video with the full movement and was asked if the movement matched its prediction and if not, why did the participant predict another object. After watching the 3 sets of videos the participant would have to rate how clear the three movements were.

To prevent participant bias towards choosing the same target between sets of videos, we randomized the target of the robot between sets and reminded the participant that it was a different movement and as such the target might change.

All videos were recorded using the WeBots simulator, an open source simulator developed by Cyberbotics [73]. Figure 4.5 show a movement of the simulated robot towards the soda can on the table, as viewed from the perspective of User 3, as recorded in the WeBots software.

Results and analysis

We recruited 315 participants using amazon's mechanical turk (M-Turk), after removing those that failed the control question, with approximately 98% from the USA and the remaining 2% distributed across Canada, Australia and the UK. We restricted the participation on the questionnaire to these four countries to reduce language barrier problems. The participants' age varied from 23 to 76 years old and an average of 42 years old. An analysis of the education level of the participants shows that 60% have a higher education degree and 90% have finished at least high school. Regarding the occupation of the participants, 94% are employed and 2% are students.

Before analyzing the performance of the MUL model against the SUL model, we aggregated the results of the SUL models into one single metric, otherwise we would not be comparing the performance of the MUL model against the SUL model but the MUL model against specific instances of the SUL. A direct comparison between MUL and specific instances of the SUL model would not be correct, because the SUL model is dependent on the PoV of the user and if a PoV offers a specially advantageous or prejudicial view of the movement it will cause comparisons that do not reflect a normal interaction.

With the questionnaires we measured the average perceived clearness of the movements, the time taken to correctly predict the robot's target and the average confidence in the prediction. We considered that a wrong prediction of the target was considered as taking the full length of the video to make a decision. For correct predictions, we considered that the time taken to predict was the earliest a participant answered correctly without wrongly predicting afterwards, *i.e.* if a participant correctly predicted the target in the 6 seconds video, but predicted wrong in the 12 seconds video and in the 18 seconds video went back to the correct prediction, the

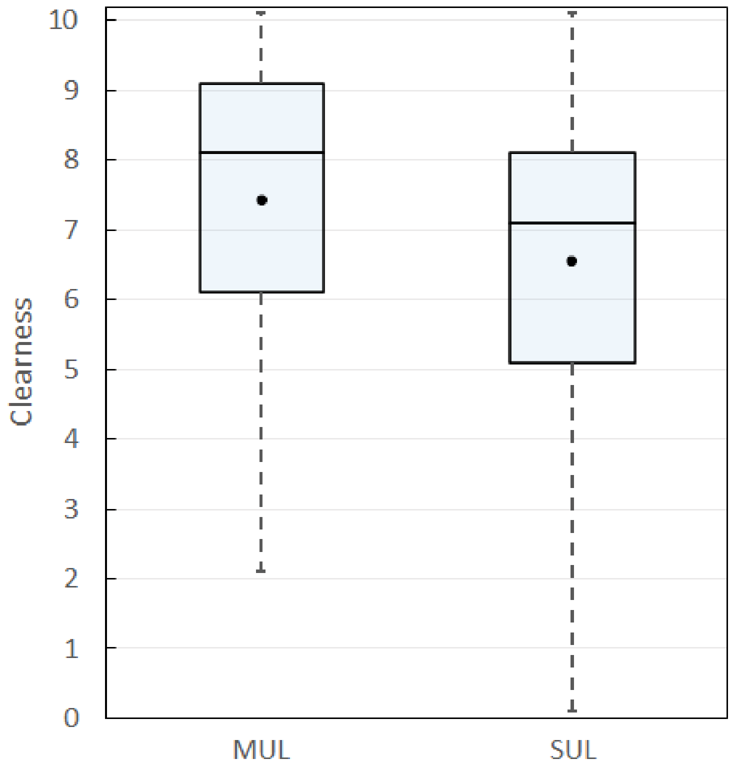

Figure 4.6: Boxplot comparing the results for the perceived clearness of the MUL model and the aggregated SUL models. The average for each model is marked with a dot.

participant took 18 seconds to predict.

To analyse the confidence in the prediction we combined the scores for each 6, 12 and 18 second videos as in [79]. For the clearness of the movements, as each participant only rated the clearness of the movements after observing the three sets of videos, we did not perform any type of pre-processing on the data.

Finally, we conducted a normality test that showed us that all three measures — perceived clearness, time taken and confidence in prediction — did not follow a normal distribution, so all the analyses of the three measures used non-parametric tests.

An analysis of the perceived clearness allows to answer hypothesis **H1**, if using the MUL model generates clearer movements than using the SUL model. We conducted a Mann-Whitney test that showed MUL was considered significantly clearer by users than SUL, $U = 7277.5, p = 0.006$, with MUL achieving an average clearness of 7.338 and SUL achieving 6.4580, thus supporting hypothesis **H1**. Figure 4.6 shows a boxplot comparison between both models.

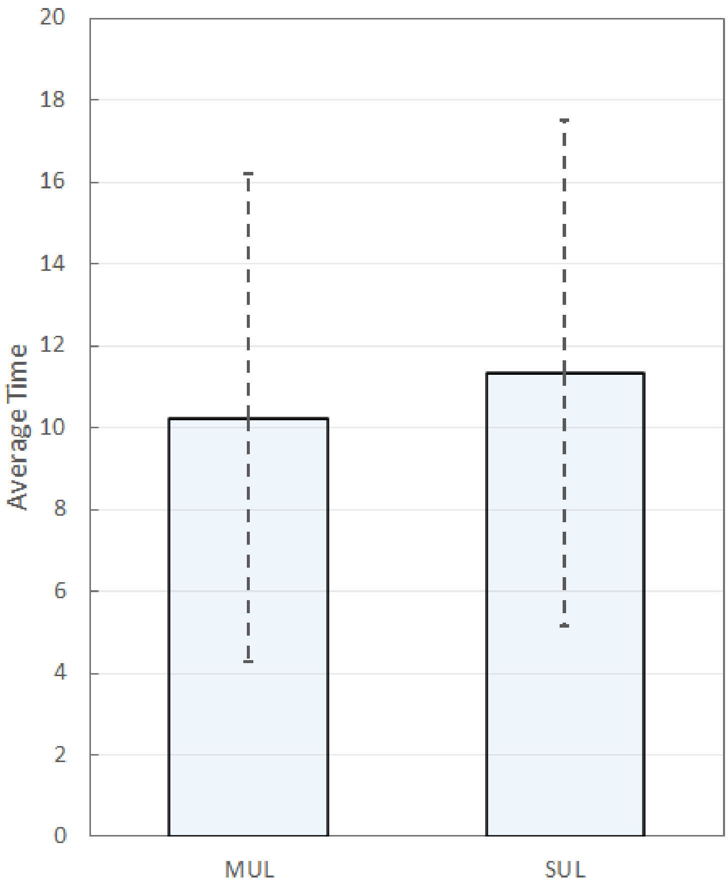

Figure 4.7: Average time to correctly predict the target according to the model used, with standard deviation.

To answer hypothesis **H2**, we analysed both the time taken to predict what target the robot was going to grab and the confidence associated with said prediction. Again, we conducted a Mann-Whitney test to compare the two models for both the time and confidence measures. Applied to the average time taken, the Mann-Whitney test showed people took significantly less time to correctly predict the robot's target with MUL optimization than with SUL optimization, $U = 75722, p = 0.037$. Figure 4.7 shows that on average participants paired with the MUL model took 10.234 seconds to correctly predict the robot's target, while participants paired with the SUL model took 11.331 seconds.

A follow-up analysis of the results show a significant difference between both models in the number of participants that needed 6 and 18 seconds to predict the correct target. In this analysis we ignored answers that wrongly predicted the target, since we were interested in

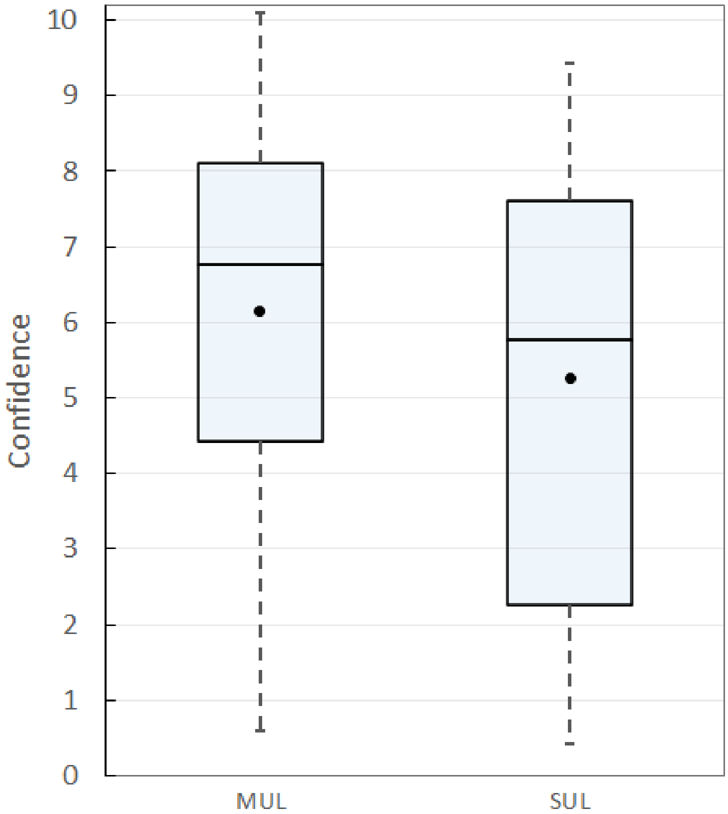

Figure 4.8: Boxplot comparing the confidence in the predictions between the MUL model and the aggregated SUL models. The average for each model is marked with a dot.

understanding how early a participant could correctly predict, when paired with each model. A Mann-Whitney test showed that participants paired with MUL model where more prone to need only 6 seconds to correctly predict the target than when paired with SUL, $U = 47324, p = 0.004$, with 79.6% of the participants paired with MUL needing only 6 seconds to correctly predict against 68.5% when paired with SUL. The inverse tendency was verified, with a Mann-Whitney test showing a bigger tendency for participants paired with SUL to need 18 seconds to correctly predict the target than when paired with MUL, $U = 47477, p = 0.001$, the results show that 21.5% of the participants paired with SUL took the full 18 seconds when only 10.8% of the participants paired with MUL took as much time.

Regarding confidence in prediction, the Mann-Whitney test conducted showed participants are significantly more confident in their predictions with MUL than with SUL, $U = 43971.5, p < 0.001$. An analysis of Figure 4.8 shows that when paired with MUL, on average, participants rated their confidence in the prediction as 6.046 out of 10 and when paired with SUL the value dropped to 5.163. The results of these two tests support hypothesis **H2**.

Discussion

Overall, the results in Section 4.3.3 support our premise that, in a multiple user interaction, combining the perceived legibilities of the movement, from the users' perspectives, increases the legibility of the movement for the group. By combining the different perspectives, the movement's optimization is influenced by the perspectives all the users' have of the movement thus creating movements that, although not the most legible for a single user, improve the legibility for all the users involved, which in turn improve the users' deductions of the robot's objectives independent of a particular user's perspective. Improving the overall group legibility, instead of a single user, insures a more fluid and safer interaction since it gives all users a better understanding of the robot's current objective.

One specific finding that shows the usefulness of the MUL model over the SUL model is that, when paired with MUL, participants showed an increase of about 10% in correctly predicting the robot's target in the first 6 seconds of observed movement. This is an especially positive aspect because legible movements are supposed to allow users to understand the robot's intentions as quick as possible to give them more time to adapt safely and efficiently.

Finally, when reading the reasons the participants gave when the prediction was not correct, the most common justifications for both models were that, from their perspective, the movement would pass too close to another object leading to incorrect predictions. However, we also observed that, in the participants paired with the SUL model, there were others that justified the wrong prediction because from their point-of-view the movement seemed to pass by the correct object and moved towards another object or seemed to have sudden changes in directions. These aspects caused some participants to think that some videos in the same set were from different movements. These results for the SUL model are in line with some of the results we observed in the exploratory study in Section 4.2, where some users complained that given their perspective they thought the robot was moving towards them and moved towards the robot.

4.4 Multi-party legibilty in joint space

Trajectories defined in joint space are advantageous for robot control because the trajectory is always feasible by the robot, since the trajectory is described as a sequence of joint configurations that respect the physical limitations of the robot. Also, defining trajectories in joint space allows for the optimization of the trajectory to always respect the robot's limitations and the resulting optimized trajectory to be feasible.

Legibility as defined in [31] uses cartesian space trajectories to capture the human's expectations towards the robot's movement. This approach is useful because it offers a simple formulation to model human expectation of how the robot should move rationally. The cartesian representation also allows the approach to be robot free, thus applicable to a myriad of robots. However, the cartesian representation has two main drawbacks stemming from not considering the physical limitations of the robot. The first drawback is that by considering a robot free approach, the optimization process may result in a trajectory unfeasible by the physical constraints of robot used. In turn, forcing to find the points closest to the optimized trajectory that the robot can reach, which will affect the robot's legibility.

The second drawback is the need to perform inverse kinematics on the resulting trajectory to get the sequence of joint configurations for the trajectory. Inverse kinematics is a computationally heavy procedure that does not guarantee a single solution for the joint configuration of a specific point, thus leading to possible sequences of joint configurations that represent trajectories with significantly different joint configurations. These differences force the robot to move with high accelerations to follow the configuration sequence, causing increased strain of the robot's mechanical system.

Given the drawbacks inherent with performing the legibility optimization solely in cartesian space, we consider important for the trajectory to be defined in joint space. Thus, considering now trajectory ξ_θ, defined in joint space as

$$\xi_\theta = [\boldsymbol{\theta}_1^T, \boldsymbol{\theta}_2^T, \dots, \boldsymbol{\theta}_N^T],$$

where N is the number of timesteps in the trajectory and $\boldsymbol{q}$ the robot configuration in joint space as defined in Section 2.1. Equation (4.10) that expects the trajectory defined in Cartesian space becomes

$$_{2D}T^W(\xi_\theta) = {}_{2D}T^H \cdot {}_HT^W \cdot FK(\xi_\theta), \tag{4.13}$$

where $FK(\xi_\theta)$ is the forward kinematics transformation for the used robot. Forward kinematics is a method that uses the kinematics equations of a robot to compute the end-effector position and orientation given a specific joint configuration. With (4.13) we can use a trajectory defined in joint space and easily compute the legibility of said trajectory. However, to correctly optimize the trajectory and prevent trajectories that do not respect the physical limitations of the robot we must be able to convert back to joint space before the update step of the optimization in (4.11). To perform this reverse conversion, Equation 4.11 becomes

$$\xi_{i+1} = \xi_i + \frac{1}{\eta} FK^{-1} \left(M^{-1} \nabla Legibility_{MUL}(\xi_i) \right) \tag{4.14}$$

where FK^{-1} defines the inverse of the forward kinematics transformation. Usually to revert a forwards kinematics transformation, a inverse kinematics operation is performed to obtained the joint values from the end-effector position and orientation. However, as previously discussed the problem of inverse kinematics does not have a singular solution and also it is not designed to convert gradients back, making impossible to use inverse kinematics in this case.

In this work, in order to solve the problem of converting from Cartesian space to Joint space without using inverse kinematics, we use the principles of resolved motion rate control (RMRC) [96]. RMRC is a technique to move a robot's end-effector from a starting position to a target position without solving the inverse kinematics problem. Instead using the rate of change of the robot's joint angles for a given rate of change of the robot's end-effector. The process is rooted in the following linear transformation that gives the end-effector position as a function of the rate of change of the joint angles

$$\dot{\boldsymbol{x}} = J \cdot \dot{\boldsymbol{\theta}}, \tag{4.15}$$

where $\dot{\boldsymbol{q}}$ defines the rate of change of joint angles for the robot, $\dot{x}$ defines the velocity of the end-effector and J is defined as the partial derivatives in order to $\boldsymbol{\theta}$ in the form

$$J = \begin{bmatrix} \frac{\delta x}{\delta \theta_1} & \frac{\delta x}{\delta \theta_2} & \dots & \frac{\delta x}{\delta \theta_N}, \end{bmatrix}$$

being N the number of joints in the robot. Thus, by inverting the relation in(4.15) we obtain a function for the rate of change of the joint angles given the end-effector's velocity

$$\dot{\boldsymbol{\theta}} = J^{-1} \dot{x} \tag{4.16}$$

However computing the inverse of J is not trivial, because J is not guaranteed to be invertible. The solution for this problem requires the computation of the pseudoinverse of the Jacobian, $J^{\#}$, changing Equation (4.16) to

$$\dot{\boldsymbol{\theta}} = J^{\#} \dot{\boldsymbol{x}} \tag{4.17}$$

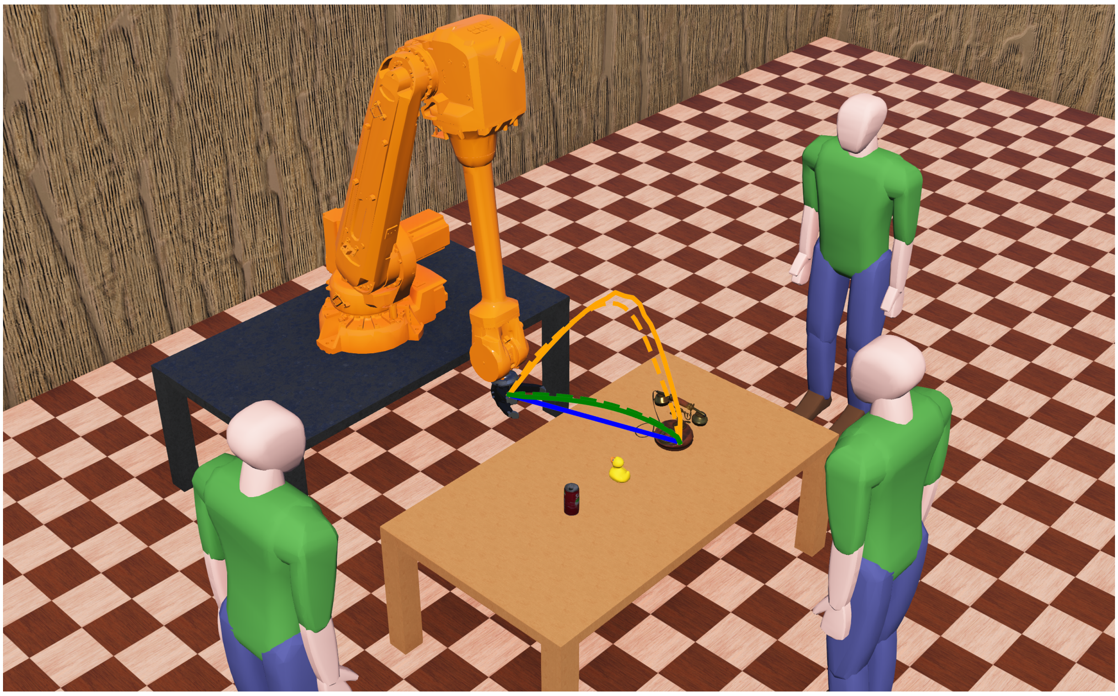

Figure 4.9: Example trajectory of a robot executing a legible trajectory optimized in joint space – green – and in cartesian space – orange. The dashed lines are the trajectories resulting from both optimization approaches and in continuous line are the same trajectories executed in the WeBots simulator using the IRB4600-40 model. As can be observed, the green trajectory – optimized in joint space – presents little difference from the theoretical to the real movements, the orange line however in the later half of the movement can not replicate the theoretical movement given the robot's constraints. In blue we can see the original trajectory before being optimized.

Applying (4.17) to (4.14) yields

$$\xi_{i+1} = \xi_i + J^{\#} \left(\frac{1}{\eta} M^{-1} \nabla Legibility_{MUL}(\xi_i) \right), \tag{4.18}$$

the updated expression for the gradient of legibility of MUL, allowing for the update of the trajectory to be performed in joint space instead of Cartesian space. In Figure 4.9, we can observe the differences between optimizing a trajectory – trajectory in blue – in joint space and in Cartesian space. The robot used in this simulation was the IRB4600-40 from ABB[2]. In orange we present a trajectory optimized in joint space and in green optimized in Cartesian space, for both movements we present a dashed line showing the theoretical movement resulting from the optimizations and in a continuous line the movement executed by the robot in the WeBots simulator. The most noticeable difference is that in joint space the trajectory does not present

[2]https://new.abb.com/products/robotics/industrial-robots/irb-4600/

as steep a curve as when optimized in Cartesian space, this is justified by the IRB4600-40's physical limitations that do not allow him to achieve the same points as in the Cartesian space optimization that is not limited by robot's physical constraints. However, for the trajectory optimized in Cartesian space, the robot only could match the expected movement for the beginning of the trajectory ending up missing the later half of the movement completely, contrary to the joint optimized trajectory that closely matched the expected movement, thus showing the advantage of having a trajectory optimized in joint space instead of Cartesian space.

4.5 Chapter summary

In this chapter, after an initial exploratory study about the impact of legibility in multi-party interactions, we explored how to define the notion of legibility for robotic movements in multi-party interactions. The results from the exploratory study guided us in understanding that, in multi-party interactions, a legible motion may have different interpretations depending on the human's perspective of the robot's movement.

So, in our definition of legibility for multi-party interactions, we combine each human's movement legibility into a single movement legibility for the human party. We also showed, in an user study, that our notion allows for robot to be more transparent regarding its intentions than when using the formulation from Dragan et al. [31].

Finally, we explored how to create multi-party legible movements directly in joint space instead of in cartesian space, thus avoiding having to transform the obtained trajectory from cartesian space to joint space.

5

Legible Decision Making

Legibility is a concept that can have positive impact beyond the scope of robotic movement. The need for expressiveness is not essential only for robot's movements. During an interaction with other agents, an artificial intelligent agent (AIA) may need to be expressive regarding its decisions and actions, namely in collaborative tasks or tasks with conflicting goals. In these types of interactions, a legible decision making process would empower a robot with the ability of making its intentions clearer and give other agents interacting with the robotic agent more time to understand and adapt to the robot's actions. More broadly, autonomous systems that are transparent regarding its intentions are preferred by human users, since humans find easier to interpret these systems' intentions and goals, even when the human is simply an observer and does not interact directly with the agent.

The idea behind legible decision-making is to shift the focus from plain optimal decision towards optimal decision making that conveys information regarding the agent's intentions. To achieve such behaviour, the agent must balance optimizing the underlying reward function, with maximizing the information conveyed by the agent's actions.

In this chapter we present the work done in exploring question **RQ 2**, focused on the application of legibility to decision making. We start in Section 5.1 by discussing in more detail the main advantages of legible decision-making. In Section 5.2, we describe our proposed framework for legible decision making using MDPs and in Section 5.3, we present the evaluation of our proposed framework.

5.1 Legible decision-making

An autonomous agent's behaviour is governed by its underlying reward function, a function that values the agent's actions in terms of its preferences and objectives. So, decision-making systems focus on optimizing the long term reward of an agent, thus maximizing the agent's preferences. However, focusing solely on being optimal may lead to a loss of clarity regarding what are the agent's current objectives, specially in scenarios with multiple objectives. The field of explainable artificial intelligence (XAI) focuses on solving this problem of building systems that maintain optimal behaviours and are also clear about their reasoning, thus improving agents' transparency and interpretability [23, 43]. Legible decision-making focus on creating decision making processes that are optimal and transparent about the agent's current goals [3, 4].

Having transparency in AIAs is extremely important in tasks that require interaction with other agents, because the coordination between agents relies on understanding each other's goals, thus avoiding get in each other's way. Moreover, in interactions that require collaboration, agent coordination is essential to task completion. Consider again Example 1 from Section 1, where a robotic agent has to collaborate with two humans to fill their cups with water. In that example, the robot starts by filling the cup of the person on his left, despite the fact the movement might be more difficult to interpret because the robot is holding the water pitcher on the right hand. In this example, it would more clear to fill the person on the right of the robot, since the robot was holding the pitcher with the hand on that side and so the robot's movement would be easier to interpret, leading to better coordination between the robot and the humans.

Transparency in AIAs' intentions is useful beyond interaction with other agents. Consider the problem of machine teaching, where a system is trying to learn the model of how to behave from a teacher (either human or AIA), who is providing examples of the correct behaviour [100]. The system trying to learn the correct behaviour policy needs to correctly infer the teacher's goal in order to better learn the underlying reward function. So, the teacher's examples need to be transparent, regarding the teacher's goal, so the student system needs less examples to learn the best behaviour policy. Otherwise, the student system will require more examples to correctly infer the teacher's underlying objective function, specially if the task being taught has similarities with other possible tasks. Consider we have an autonomous car learning how to drive a human from home to a new work site, the human may supply examples of shorter trajectories that pass by other points of interest, such as the human's dentist or the human's parents house; these trajectories represent optimal behaviours, however can cause the autonomous car difficulty in inferring the human's goal, since the trajectories share examples with other possible goals for the human. So, it would be better to give the car less ambiguous examples, so it could learn the end goal faster.

As discussed in Section 3.3, legible decision-making has already been successfully applied to deterministic scenarios. Deterministic scenarios assume that the outcome of an action is guaranteed to always be the same, *i.e.*, an agent that tries to move to another room always succeeds. This kind of scenarios are useful for planning tasks such as task scheduling or path planning, where the decision-making process deals with more macro level decision points. However, deterministic scenarios do not correctly represent real world interactions because in the real world an agent's action may not always lead to the same outcome, *i.e.*, an agent that tries to move to another room may find the room's door closed and so remains in the same room. So, the real world is better modelled as a stochastic environment, where an agent's action has a certain chance of leading to the intended outcome but also has a chance of leading to a different outcome. For stochastic scenarios, legible decision-making has been successfully used in the framework of legible Markov decision problem (L-MDP) from Miura, Cohen, and Zilberstein [75].

The framework of L-MDP applies legible decision-making to stochastic scenarios as a special case of interactive POMDP (I-POMDP) [41]. I-POMDP is a multi-agent framework defined for an observing agent (i) and on observed agent (j) and an extension of POMDP to multi-agent settings. A L-MDP is described as a tuple

$$\mathrm{L-MDP} = \langle S, A, P, \gamma, \Theta, B, R \rangle\,, \tag{5.1}$$

where:

- S, A, P, and γ are as defined in Section 2.3;
- Θ is a set representing possible intentions/goals/capabilities of the agent;
- $B : H^* \rightarrow \Delta^{|\Theta|}$ represents the assumed belief of the agent given a history H^* and $\Delta^{|\Theta|}$ is a simplex on Θ;
- $R : S \times A \times \Delta^{|\Theta|} \rightarrow \mathbb{R}$ is the reward function describing how desirable it is to take an action give a state and a belief $b \in \Delta(\Theta)$.

The value function of a L-MDP is defined as:

$$V_H^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{H} \gamma R(B(h_t))|S_0 = s, \pi\right], \tag{5.2}$$

with H the finite planning horizon.

Given the dependence on the interaction history, L-MDP does not verify the Markov property and so cannot be solve with traditional MDP methods like Value Iteration. Actually, the authors in [74] propose to use UCT to simulate stochastic sequences that make the history needed to solve the L-MDP. This method has a major drawback, which the authors discuss, of making L-MDP intractable for large state spaces and planning horizons.

5.2 Policy legible MDPs

An optimal policy π^* describes the most efficient way an agent can solve an MDP. However, the optimal policy does not guarantee the decisions to be clear to an observer and may cause doubts regarding what the agent is trying to accomplish. In Fig 5.1 we can observe, in blue, a

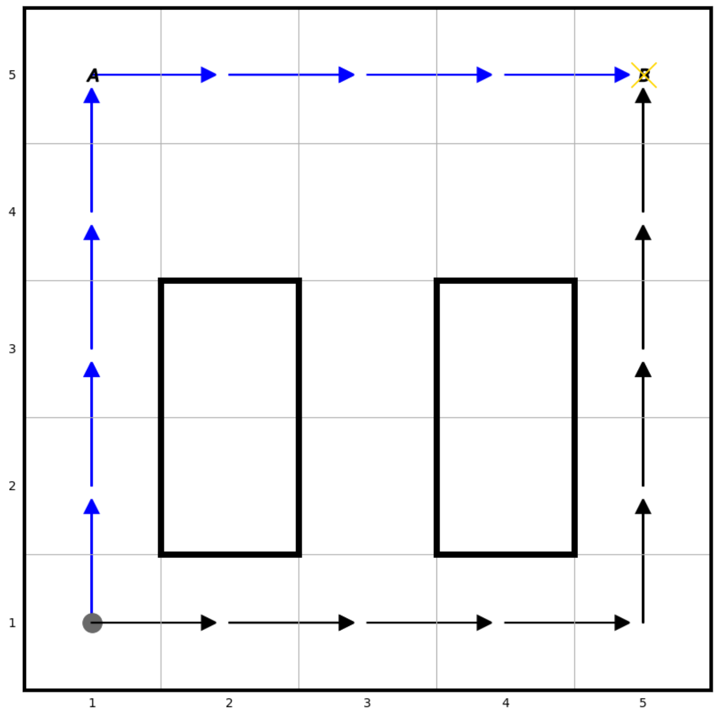

Figure 5.1: Example of maze-like environment with two goals, A and B. The blue arrows indicate a possible action sequence following an optimal policy, while black arrows indicate an action sequence following a legible policy (which is also optimal).

possible sequence of actions prescribed by an optimal policy, for an agent moving in a maze world scenario, towards the objective B in that world. We can observe that the optimal policy leads the agent to objective B while going through A. An observer that did not know the robot's intentions could, upon observing the initial actions of the agent, confuse the robot's goal to be A rather than B.

We propose to adapt the notion of legibility to MDPs to yield policies that offer both a solution with high expected reward and make clear the agent's current objectives. To achieve such policies we introduce *PoLMDP*. A PoLMDP is defined in the context of an environment with N different objectives, each represented by a different reward function $r_n, n = 1, \ldots, N$, and thus defining a different MDP – $\text{MDP}_1, \text{MDP}_2, \ldots, \text{MDP}_N$. A PoLMDP is described as a tuple $\langle X, A, P, r_{\text{leg}}, \gamma, \beta \rangle$, with X, A, P and γ are as in Section 2.3, r_{leg} defines the legible reward function and measures each action's legibility in each state, and β is a non-negative constant that defines how close the legible function follows the optimal expected reward.

Following the original definition of Dragan and Srinivasa [29], we define r_{reg} to measure the legibility of an action a in state x by evaluating how likely it is that a is executed given that the current goal is defined by reward r_n, in opposition to performing the same action when trying to achieve another possible objective. In Figure 5.1 we can observe, in black, the decision sequence prescribed by a PoLMDP policy, for an agent moving in a maze-like scenario towards objective B. The action sequence prescribed by PoLMDP focus on moving the agent away from objective A as soon as possible, while getting closer to B.

We define our legible reward function r_{leg} for a target reward r_n as

$$r_{\text{leg}}(x, a) = P(r_n \mid (x, a)), \tag{5.3}$$

where

$$P(r_n \mid (x, a)) \propto P((x, a) \mid r_n)P(r_n).$$

This is similar to Equation 4.3. However, instead of observing a movement snippet, we observe the current state and action performed as indicators of the robot's intentions. We choose this representation because, in a MDP, each decision step is an independent event and as such the probability of executing an action a in state x is not influenced by possible states and actions that preceded the current state. This way, we need only observe the current state-action pair to infer the robot's intentions. Considering a uniform distribution as the prior on the probability of observing each goal, we can simplify the previous expression as

$$r_{\text{leg}}(x, a) = P((x, a) \mid r_n). \tag{5.4}$$

This probability reflects how probable is executing an action in a world state when trying to achieve one specific objective against when it tries to achieve another possible objective. We follow the same maximum-entropy principle adopted by Dragan and Srinivasa [29] and define $P((x, a) \mid r_n)$ as

$$P((x, a) \mid r_n) = \frac{\exp(\beta Q_n^*(x, a))}{\sum_{m=1}^{N} \exp(\beta Q_m^*(x, a))}, \tag{5.5}$$

with β the parameter in the PoLMDP description tuple and $Q_{r_n}^*$ the optimal Q-function for MDP$_n$. From (5.5), an action a is more legible the larger the gain of executing it in MDP$_n$ in comparison to the gain of performing a for other possible goals. The policy obtained using the reward in (5.4) promotes actions that guide the agent towards its intended goal, while increasing the agent's expressiveness.

5.2.1 Relation with other approaches

The approach taken in our PoLMDP is closest with the L-MDP of Miura, Cohen, and Zilberstein [75]. Both approaches offer formulations of legibility for MDPs, differing on the approach taken to compute the legible policy. In the formulation of L-MDP, the MDP builds a reward function from a belief b_t that, at each time step t, translates the observer's inferred goal from the actions of the agent observed up to time step t. The authors propose update the belief b_t as follows

$$b_t\left(\theta|h_t\right) = P\left(\theta|s_{t-1}, a_{t-1}, s_t, b_t\right)$$
$$= \frac{\hat{T}\left(s_{t-1}, a_{t-1}, s_t|\theta\right)\hat{\pi}\left(s_{t-1}, a_{t-1}|\theta\right)b_{t-1}\left(\theta\right)}{\sum_{\theta'}\hat{T}\left(s_{t-1}, a_{t-1}, s_t|\theta'\right)\hat{\pi}\left(s_{t-1}, a_{t-1}|\theta'\right)b_{t-1}\left(\theta'\right)},$$

with θ the agent's goal, $\hat{T}$ is the assumed transition for a given θ and $\hat{\pi}$ the agent's policy for a given θ. The reward is computed from either the Euclidean distance or KL-divergence between the observer's (estimated) belief, b_t, and the belief b^* translating the correct goal

$$dist(b_t, b^*) = D_{KL}(b_t \,||\, b^*) = \sum_{\theta\in\Theta} b_t(\theta)\log\frac{b_t(\theta)}{b^*(\theta)}.$$

The dependence of the reward on b_t introduces a dependence on the history of the process, rendering L-MDPs not amenable to the use of standard MDP (or POMDP) solution techniques.

Our approach circumvents the dependence on the history by pre-computing $b(\theta)$ using (5.5) and can be obtained from that of Miura, Cohen, and Zilberstein [75] by instead considering the distance between beliefs given by the *Total Variation* (TV) distance

$$dist(b_t, b^*) = \frac{1}{2}D_{TV}(b_t \,||\, b^*) = \frac{1}{2}\sum_{\theta\in\Theta}|b_t(\theta) - b^*(\theta)|,$$

where b^* is an indicator function for the agent's goal. The KL-divergence, used by Miura et al., is a special case of f-divergence with $f(x) = x\log x$, while the TV-distance we use in PoLMDP is a special case of f-divergence with $f(x) = |x-1|$.

5.3 Experimental evaluation

In this section we present the evaluation of our framework. Our evaluation aimed at answering three main questions. The first question was

How does how framework of legibility compare with that of L-MDP from Miura et al. [75] in terms of (i) legibility of the computed policies and (ii) computational efficiency?

To answer this question we compared the performance of the two frameworks, in a simulated Maze World scenario, in terms of the average legibility of the solutions obtained and the average time to obtain the solutions.

The second question of this evaluation was

Do PoLMDPs lead to policies that more explicitly indicate the goal of the agent's actions?

To answer this question we compared the number of examples required for an agent to infer goal of a PoLMDP agent against the examples required when observing an optimal agent.

The final question of this evaluation was

In an interaction with human users, does a robot using a legible policy generated by PoLMDP convey its intentions faster than using a standard optimal policy?

To answer this question we conducted an online user study, where each participant played a guessing game with either a PoLMDP agent or an optimal agent.

We will now explain the scenario used in the three evaluation steps, detailing the motivation for using the chosen scenario, and afterwards we present the three evaluations performed.

5.3.1 Evaluation scenario

In our evaluations we used the *maze world* scenario, where a robot had to navigate and reach one of various coloured areas scattered in the maze. These areas would serve as the different goals the robot could be aiming to accomplish. The robot had available five different actions: moving up, moving down, moving left, moving right and no operation. When moving in the maze, the robot had a 15% chance of failing the action and staying in the same location.

The maze world scenario chosen is a scenario widely used to compare agents and robotics frameworks and is interesting because it serves as the base to model different real world scenarios, *e.g.* the plan of a building during a search and rescue mission or the floor plan of a warehouse with robots retrieving different items to deliver. Besides being a common scenario for comparisons in simulated environments, the maze world scenario is an environment that could naturally occur in an interaction between humans and autonomous agent systems. Thus, by using a maze world we can reuse the same environments across the different evaluations, allowing for the obtained results to applied across evaluations. The 15% chance of failing and staying in the same place introduces stochasticity and does not impact the legibility of the robot in the case of a failure. Other options would be for the robot to veer off course in a failure, but by veering off course we would be introducing artifacts that could impact the legibility in unforeseen ways and thus impact the measured levels of legibility. Also, the option of having a failure of movement instead of veering off course was motivated by the fact that, in the user study, a participant could find weird the robot moving in a diagonal when indicated that the robot could only move up, down, left or right.

5.3.2 Comparison with legible MDP (L-MDP)

The comparison of our framework with Miura's L-MDP allows to understand what are the advantages our framework offers over other state-of-art frameworks. We only compared with the L-MDP because it is the only framework that proposes to solve the problem of legible decision making using stochastic environments.

Setup

Our comparison used a maze world scenario and explored how each framework performed when we scaled:

- the number of possible goals, keeping constant the number of states;
- the number of states in the world, keeping the number of goals constant.

The measures we used to compare the frameworks were: the average time taken, by each framework, to find a sequence of decisions from the initial state to the given goal; the average legibility value for each sequence using the legible function of PoLMDP and using the legible function of Miura's L-MDP.

To compare the performance of the two frameworks with varying number of goals, we tested both frameworks on a 25x25 maze world with the number of possible goals varying between 3 and 10 possible goals, making up 7 different world configurations.

Regarding the states scalability test, we designed multiple maze world configurations where we varied the number of states. All mazes had 6 possible goals, except for the smallest maze which size could only accommodate 3 goals. The smallest maze was taken from the paper where Miura presents the L-MDP framework and has 40 states, a 5 rows by 8 columns maze. The rest of the mazes had the following dimensions: 100 states (10x10), 625 states (25x25), 1600 states (40x40), 2500 states (50x50), 3600 states (60x60) and 5625 (75x75). Again totalling 7 different world configurations.

For each test we sampled 250 pairs of initial state and goal for each world configuration, with the only requirement being that the initial state was different than the goal state.

With these tests we aim at comparing the performance of the two frameworks under different scalability conditions, namely the impact on the average time taken to obtain a solution and the average quality of the solutions.

After sampling the 250 pairs for each test and each world configuration, we ran those pairs through each framework, limiting their execution time to a maximum of 2 hours for each test pair. If in those 2 hours, a framework could not give a solution, we would mark that test pair as a failure.

Results

With all the testing pairs ran, we had to do a pre-processing of the results to guarantee a balance dataset of the results. This step was needed because of the disparity of failed tests between the two frameworks, as shown in dotted lines in Figures 5.4 and 5.7, which caused to have an unbalanced number of test results for our PoLMDP and for Miura's L-MDP. Thus, to balance the datasets, we decided to use only 100 samples for each testing condition, which was the highest common number of samples between all testing conditions, *i.e.*we used 100 samples of each of the 7 world configurations for each of the testing conditions.

Given that we used a time constraint to determine test failure, we used the same time criteria to balance the dataset. Thus, for each world configuration in both the state scaling and goal scaling conditions, we ordered the results by time the time taken to obtain a solution and, for each world configuration, we removed the result entries that presented higher execution time. This way we reduced the amount of bias introduced by using the same criteria as before.

After balancing the results for each configuration, we extracted 100 test results for each configuration. For the configurations that had more than 100 samples, we randomly selected 100 samples for each configuration, considering a sample the results corresponding to same pair of initial state and goal for both frameworks. For the configurations with only 100 samples, we did no random sampling and used the available 100 samples.

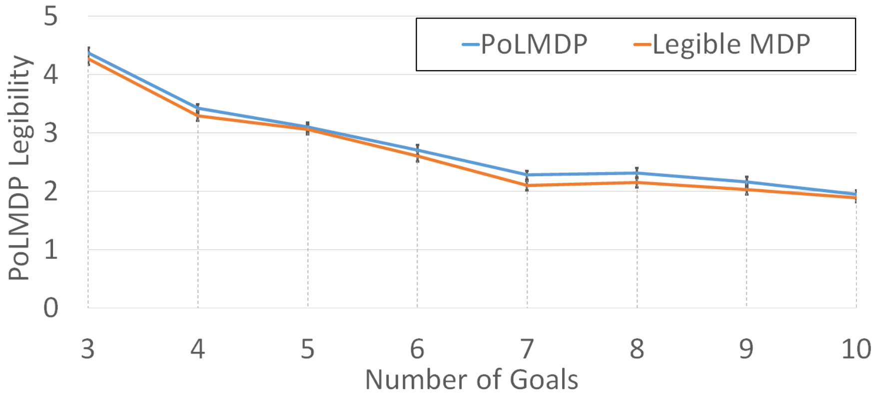

Figure 5.2: Results for the PoLMDP legibility metric performance comparison between the PoLMDP framework against Miura's Legible MDP, when we vary the number of possible goals in a mazeworld like scenario.

For the goals scalability test, the percentage of failures is shown in Figure 5.4 in dashed lines, while Figure 5.7, in dashed lines, shows the percentages of failures for the states scalability test.

Regarding the evaluation for the average time taken to give a solution and the legibility performance according to each of the two legibility metrics used, Figures 5.2, 5.3 and 5.4 show the results for the goals scalability and Figures 5.5, 5.6 and 5.7 show the evaluation results for the state scalability test. Figures 5.2 and Figure 5.5 show the average legibility, according to the PoLMDP's legibility function, of the obtained solutions for each framework; Figures 5.3 and Figure 5.6 show the average legibility, according to the L-MDP's legibility function, of the obtained solutions for each framework; and Figures 5.4 and Figure 5.7 show the average time each framework needed to obtain a possible reward.

Discussion

The results of the first evaluation show interesting results. The first result that deserves analysis is the percentage of tests that failed. The L-MDPs had an average of 40-60% fail rate in most of the tests in both scalability tests. The only cases where such did not occur where on the tests with world configurations with small state spaces (≤ 100 states), where the fail rate was between 4% and 13%. On the other hand, looking to the results of PoLMDP, we can observe that the fail rate was always at 0%, that means PoLMDP was capable of always finding a solution in under 2 hours, no matter how big the number of states or the number of goals presented. This is an extremely interesting fact, because when we use sequential decision systems in robots, we need them to give a solution quickly and the PoLMDP offers that capacity. These results

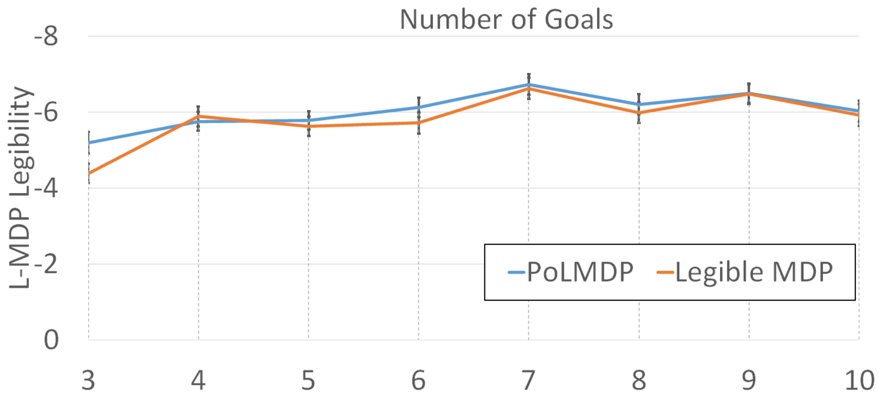

Figure 5.3: Results for the Miura's Legible MDP legibility metric performance comparison between the PoLMDP framework against Miura's Legible MDP, when we vary the number of possible goals in a mazeworld like scenario.

are closely related with the results obtained for the average time to find a solution, where again PoLMDP outperformed L-MDP; the analysis of the average time also yields another interesting fact, with an increasing number of states the PoLMDP's average time to find a solution increases at much lower rate than that of L-MDP, hinting at the PoLMDP being more resistant to changes in the number of states.

Regarding the results of the legibility metrics, the two frameworks did not show a significant difference on the results obtained for the scalability of goals. Both frameworks performed better according to their respective metrics, but without significant differences.

5.3.3 Evaluation of intention transmission

Inverse reinforcement learning (IRL) agents use examples given by an expert to try and learn the expert's underlying reward function to solve a given sequential decision problem. To that extent, the example samples given to an IRL agent must clearly show the expert's preferences and best practices to solve the problem. In a single objective scenario, simply presenting examples of the best practices allows an IRL agent to learn the reward function being used; however, in a scenario with multiple possible objectives that may not be the case, because the best practices to achieve one of the goals may overlap the best practices to reach a set of other possible goals. So, by supplying an IRL samples given by our PoLMDP or by a standard optimal MDP approach and analyse which approach leads to the IRL agent to learn the correct reward function faster, we can evaluate if using our framework is better to convey an agent's goals and intentions than use a simple optimal MDP agent, in a multiple objectives scenario.

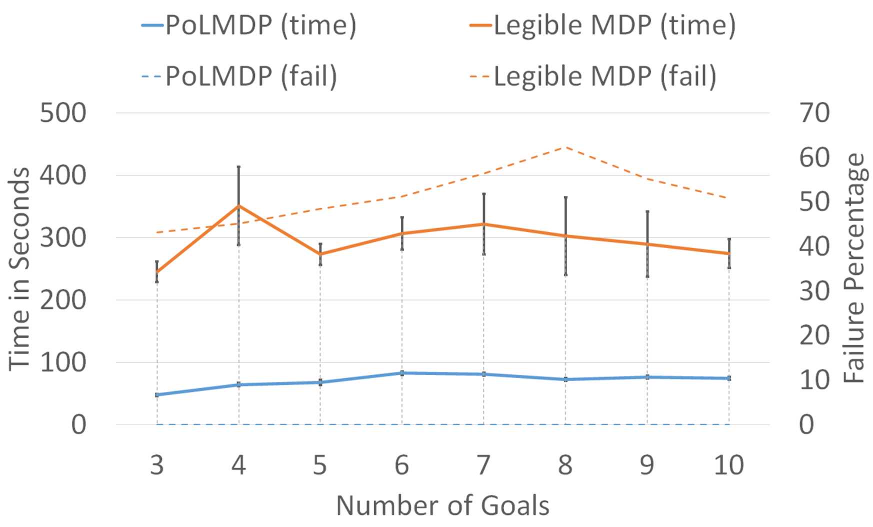

Figure 5.4: Results for the time performance comparison between the PoLMDP framework against Miura's Legible MDP, when we vary the number of possible goals in a mazeworld like scenario. In continuous lines we show the average times, and, in dashed lines, the percentage of failed tests.

Setup

To understand the best approach to teach an IRL agent, we did two different tests. The first test aimed at understanding how the two approaches fared when teaching an IRL agent a complete sequence of decisions from beginning to end, resembling how a human expert teaches the complete process of solving a problem. This test explores how the two approaches teach an IRL agent with examples that are logically expected to follow previous ones. The second test aimed at exploring how well the two frameworks teach an IRL agent the best actions in situations without a necessary connection between them. This test resembles how an expert would teach fringe situations in solving a problem where, instead of showing the entire process, the expert would focus on showing how to act in specific situations. With this test, we explore how well the two approaches teach an IRL agent using examples that do not necessarily follow previous ones.

In this evaluation we used 4 different configurations of 10x10 mazes. The 4 configurations varied both on the position of the possible goal locations and the configuration of the walls in the mazes, however in all the configurations there were 6 possible goal locations the robot could reach.

For the comparison of sequential decisions, we sampled 250 pairs of initial positions in the

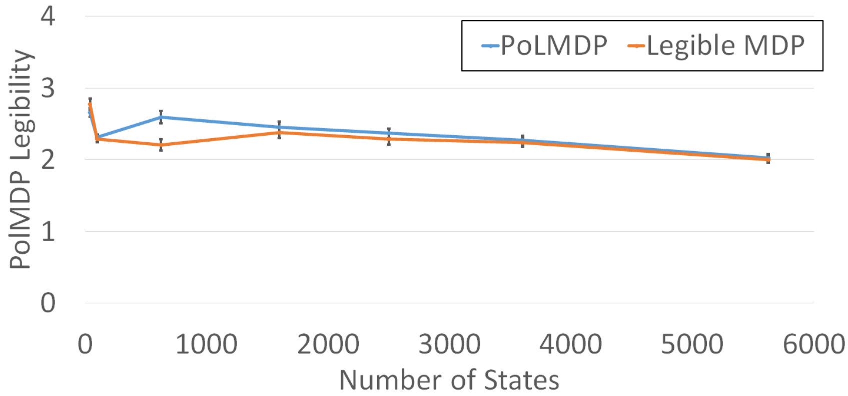

Figure 5.5: Results for the PoLMDP legibility metric performance comparison between the PoLMDP framework against Miura's Legible MDP, when we vary the number states in the mazeworld scenario.

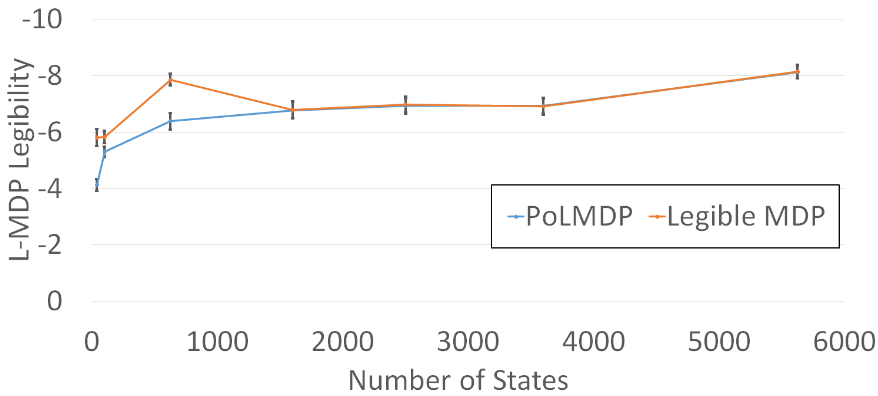

Figure 5.6: Results for the Miura's Legible MDP legibility metric performance comparison between the PoLMDP framework against Miura's Legible MDP, when we vary the number states in the mazeworld scenario.

mazes, for each of the 4 world configurations. Then, using either a policy generated by PoLMDP or the optimal policy, we obtained, for each initial position, ten trajectories of 20 steps each between that initial position and each of the possible goals. With all the trajectories generated, we then took each trajectory and sequentially gave more examples of the same trajectory to an IRL agent. Each time we supplied a new example, we asked the IRL agent to predict what was the robot's goal, using the samples of the trajectory seen so far, registering if the prediction was correct or incorrect. We used a different instantiation of an IRL agent for trajectories generated with a PoLMDP policy and trajectories generated with an optimal policy.

For the comparison of non-logically connected examples, we sampled 250 sets with 20 random states each, for each of the 4 world configurations. Then we obtained the action prescribed,

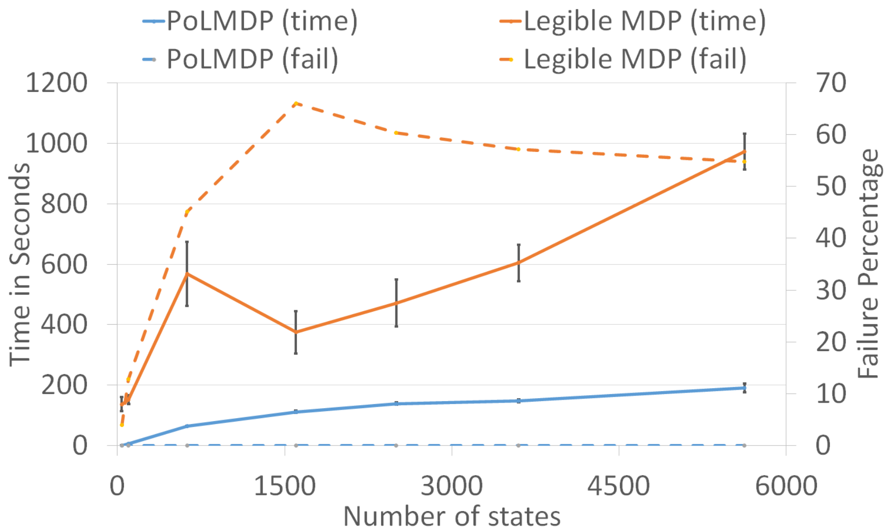

Figure 5.7: Results for the time performance comparison between the PoLMDP framework against Miura's Legible MDP, when we vary the number states in the mazeworld scenario. In continuous lines we show the average times, and, in dashed lines, the percentage of failed tests.

using either the PoLMDP policy or the optimal policy, for each random state to move towards each one of the possible goals. After finishing this process, we took each set of 20 state-action pairs and sequentially gave each pair to an IRL agent, asking the agent to predict, with each new pair, the goal most probable for the robot. Each time we got a prediction, we compared it to the expected prediction and marked either as correct or incorrect. We used a different instantiation of an IRL agent for trajectories generated with a PoLMDP policy and trajectories generated with an optimal policy.

Results

After running the evaluation on both testing conditions, we aggregated the results and obtained the average ratio of correct predictions, depending on the approach used and on the number of samples given to the IRL agent.

The results for the test with complete decision sequences can be observed in Figure 5.8. The results show that both approaches allowed the agent to learn at a similar rate, although the samples from a PoLMDP policy appear to allow the IRL agent to correctly predict the teacher's goal more frequently when less samples were shown. However these results do not show a significant difference between the two approaches when the examples used are related to each other.

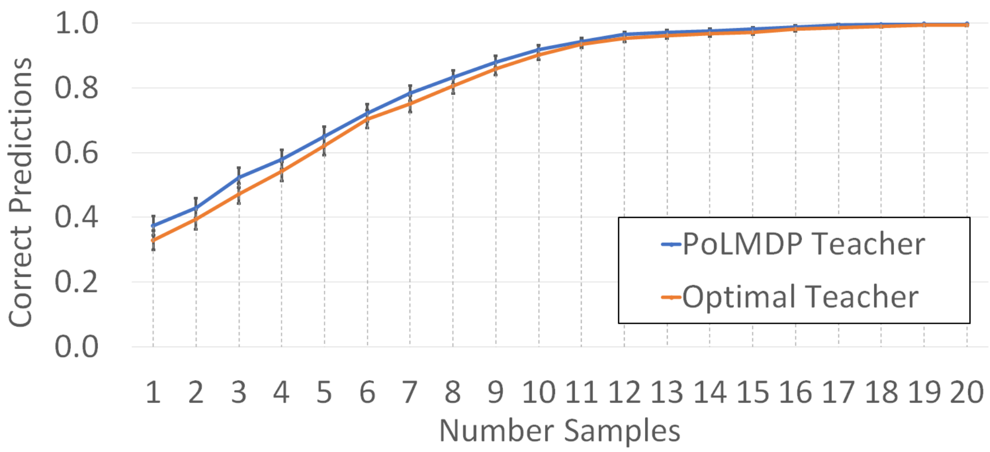

Figure 5.8: Ratio of correct predictions depending on the number of examples shown to an IRL agent, with error bars. These results pertain to the condition where the samples formed a complete sequence of decisions.

The results for the test with non-logically connected examples can be observed in Figure 5.9. The results show that both IRL agents learned the teacher's intentions after 20 examples, but the two approaches have shown differences in performance. When using examples from a PoLMDP policy, the IRL agent learned the teacher's intentions at a significantly faster pace than when the teacher used examples from an optimal policy. When analysing the results, we observe that, when using a PoLMDP policy, with only 5 examples the IRL agent correctly predicted the learner's intentions in 80% of the trials. The same performance was only achieved by an IRL agent using an optimal policy after observing 7 examples. Only after 12 samples the performance of an IRL agent becomes similar using either approaches, with an agent using a PoLMDP policy showing better learning before then.

Discussion

The results of both tests show that the use of PoLMDP policies allows an IRL agent to learn the underlying reward function and the teacher's intentions faster than using a standard optimal policy. This difference is more distinct in the test with examples without a necessary relation between them. In this test we observed that the IRL learner paired with the PoLMDP teacher continually had a higher correct prediction ratio than the learner paired with the optimal teacher.

The results for the test with a complete sequence of decisions do not show a significant difference between both approaches. This was not surprising, because in this test both approaches had to give a sequence of decisions logically connected with each other and thus the

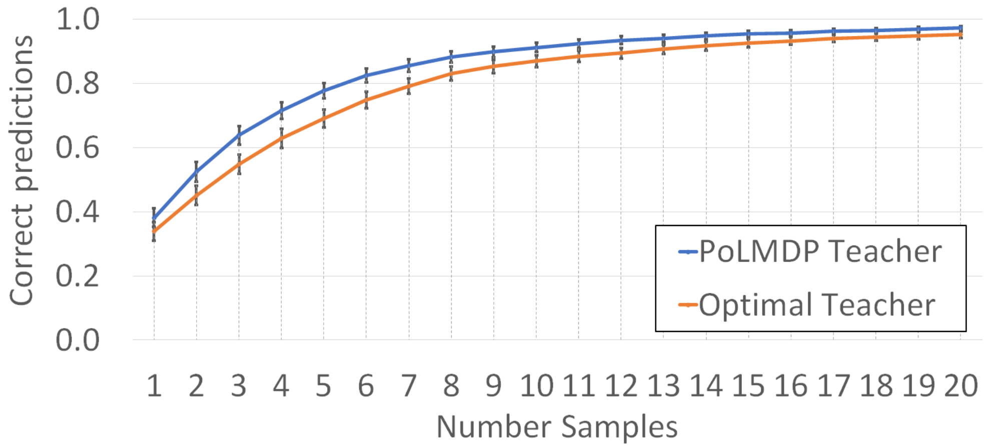

Figure 5.9: Ratio of correct predictions depending on the number of examples shown to an IRL agent, with error bars. These results pertain to the condition where the samples had no specific correlation between each other.

examples sequentially drove the learner towards the correct goal. However, the results show a slight trend for the performance of a learner paired with a PoLMDP teacher to be better, which is in line with the principles of legibility that aim at making actions easier to read and understand the intentions behind them.

Overall, the results of this evaluation point to the PoLMDP framework allow for an IRL agent to learn the teacher's underlying reward function faster.

5.3.4 User Study

The user study is an important aspect of our evaluation because this framework is meant to improve the interaction between robots or other autonomous agents and humans. Thus, to correctly infer the impact PoLMDP has on possible human users, we need to evaluate if a robot using a PoLMDP policy is better at conveying intentions than a robot using a policy that maximizes the underlying reward function.

Setup

We conducted a user study on the Prolific[1] platform, an online platform to conduct academic research and data collection. Our study aimed at comparing the performance of our PoLMDP framework in conveying a robot's internal goals and explored the question:

[1] https://prolific.co/

“Does the use o PoLMDP generated policies lead a more informative robot decision making?”

To support our exploration of the problem and answering the research question, we postulate the following working hypotheses:

H1 *Participants will understand better the robot’s intentions, when paired with a PoLMDP policy than when paired with an optimal policy*.

H2 *Participants will understand quicker and more confidently the robot’s intentions, when paired with a PoLMDP policy than when paired with an optimal policy*.

Thus we designed an online study, disguised as a guessing game, where the participants had to correctly predict where the robot was moving towards. Our study followed a between-subjects design, with each group being a different type o policy – PoLMDP policy or optimal policy. Each participant would observe 10 small videos of a robot moving, in a maze world scenario, to one of 6 differently coloured areas. Figure 5.10 shows a still from one of the possible videos the participants would watch. The participants had to correctly predict which of the coloured areas was the robot’s objective and as fast as possible. For each video, the participants had a play and stop button to control how much of the video to watch and when they felt they knew the objective they would stop the video, choose what coloured circle was the robot’s objective and rate how confident they were in the prediction.

At the end of the study, the participants were presented with a score on how well they performed depending on how fast they correctly predicted the robot’s objectives.

Before beginning the study, the participants had to read a consent form and choose if they consented to the study. After that they answered some demographic questions about their age, gender, nationality, degree of education, occupation and familiarity with robots.

Results

We recruited 150 participants through the Prolific platform, with 66% from Europe, 19% from Africa, 11% from Northern America and Mexico and the remaining 3% from South America and the Middle East. The participants average age was between 18 and 29 years old, but their ages varied from 18 to 70 years old. Their gender distribution was 59% female, 38% male and less than 3% identified as non-binary. Finally, 88% reported to have had little or no interaction with robots in their lives, 9% reported to interact occasionally with robots and less than 3% interact frequently with robots.

Figure 5.10: Still of a possible video the participants would watch. Each of the coloured circles is a possible location the robot could moving towards and the participants had to predict the correct one as soon as possible.

Besides the demographic data, for each participant we measured the average time taken to predict the robot's objective and the percentage of correct predictions. We also measured the self-disclosed rating of confidence in the predictions. Since each participant answered to 10 different videos and the videos were randomly sampled from a pool of 37 videos for each condition, we measured the participants' answers for each video presented.

To measure the average time taken to predict the robot's objective, we recorded the timestamp on the video when the participant stopped the video before making a prediction and the full length of the video. Then, if a participant correctly predicted the robot's objective, the participant's time would be the recorded timestamp, if the prediction was wrong the prediction time would be the full length of the video. Regarding the rating in confidence, each participant was asked to rate their confidence in a 7 point Likert scale after issuing a prediction in each video.

Before we started the analysis of the results, we conducted a normality test on the three measures used. This test reported that the answers obtained did not follow a normal distribution, so all analyses use non-parametric tests or tests that do not assume normality of answers.

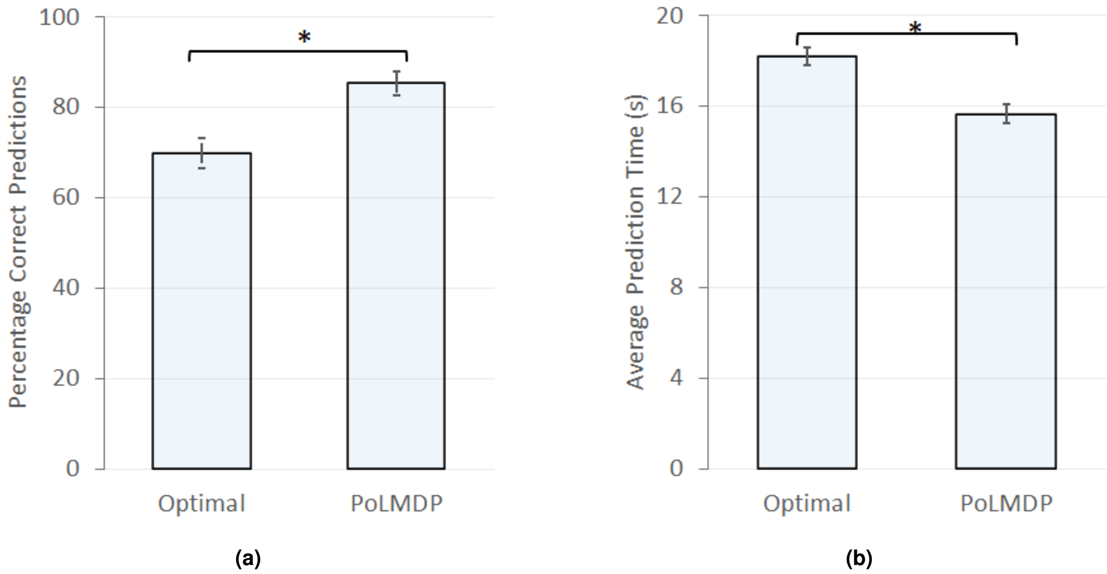

Figure 5.11: Bar plots, with 95% confidence interval error bars, for the percentage of correct predictions and average time. Figure 5.11a shows the percentage of correct predictions, according to the type of policy used. (*$p < 0.05$). Figure 5.11b shows the average time to correctly predict the robot's objective, according to the type of policy used. (*$p < 0.05$).

The analysis of the number of correct predictions yielded that participants paired with PoLMDP correctly predict the robot's objective in 85% of the predictions, while those paired with the optimal condition correctly predicted the objective in 70% of the predictions. We conducted a Pearson's Chi-Square test, $\chi^2(1) = 48.864, p < 0.001$, showed that the difference in percentages was significant, supporting our **H1** hypothesis. Figure 5.11a shows the results for the percentage of correct answers according to the type of policy used.

The analysis of the average time to correctly predict the objective, showed that on average participants paired with the PoLMDP condition took 15.67 seconds to correctly predict the robot's objective, while participants paired with the optimal condition took 18.22 seconds. We conducted a Mann-Whitney test that showed the difference was significant, $U = 177976, p < 0.001$, thus supporting our hypothesis **H2**. Figure 5.11b shows the results for the average time to correctly predict the robot's objective, with the standard error bars.

Finally, regarding the self-rated confidence in the predictions, Figure 5.12 shows a boxplot comparing the two conditions. Participants paired with the PoLMDP condition rated their confidence, on average, as 6.08 out of 7 while participants paired with the optimal condition rated as 5.78 out of 7. A Mann-Whitney test conducted compared the two conditions averages, $U = 231203, p = 0.001$, showing that participants paired with the PoLMDP condition were sta-

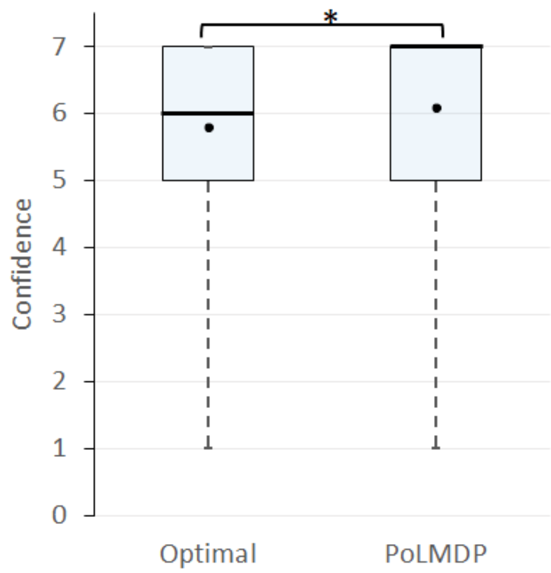

Figure 5.12: Boxplot comparing the self-rated confidences in the predictions, according to the type of policy used. The average for each policy is marked with a dot and thicker black line marks the median. (*$p < 0.05$)

tistically more confident in their predictions than those paired with the optimal condition. These results support hypothesis **H2**.

Discussion

The results of the user study support both of our working hypotheses, thus showing the positive impact of using our PoLMDP policy. Our approach allowed the robot to be more expressive regarding its internal goals, making clearer for the users interacting with the robot what the robot was trying to achieve.

Participants paired with our legible policy saw an increase of 15% in correct predictions, taking on average 3 seconds less to predict the robot's objective. These two aspects support the usefulness of this type of policy in interaction scenarios between humans and robots: by causing humans to have better predictions about a robot's intentions and with less time needed to predict, humans have more time to analyse the workspace and decide on the best action to perform according to their own internal objectives and intentions. Thus, humans can think about what they are doing, instead of just reacting to a robot, giving humans more power in the interaction.

Another aspect to highlight from this study is the higher confidence human participants felt in their predictions. The confidence was not only higher, but consistently higher in humans paired with PoLMDP as seen by the boxplot in Figure 5.12 where the median line is at the highest possible value. This consistency of high confidence is important, because when humans feel confident in their decisions they can focus better in their personal objectives and tasks and, in collaborative scenarios, this can lead to increased task performance. Thus, an increase in confidence has a great impact in the success of an interaction and the overall interaction experience.

5.4 Chapter summary

In this Chapter we presented the problem of legible sequential decision-making in stochastic scenarios. We proposed a framework for legible decision-making dubbed *policy legible Markov decision problem (PoLMDP)*, following the formalism of MDP. PoLMDP creates legible behaviours by evaluating the actions available to an action, at each decision point, choosing the one that is more salient for the goal the agent is trying to reach. We showed in a comparative study that our proposed framework is capable of outperform other approaches for legible sequential decision-making in stochastic scenarios. We also showed in two other evaluations that PoLMDP is better at conveying an agent's intentions, to both humans and autonomous agents, than simpler optimal approaches. Thus, showing that not only PoLMDP is capable of producing tractable legible behaviours, but these behaviours are capable of better displaying an agent's intentions.

6

Legible Decision Making in Teams

In Chapter 5 we explored the application of *legibility* in stochastic decision making and its impact in interactions between humans and robots; providing insights into question **RQ 2**. We presented and described PoLMDP, a framework for legible decision making, which we showed improves human-agent interactions, making the robot's intentions clearer to the human, and outperforms other approaches that aim at legible decision-making.

Our approach in Chapter 5 followed related research on legible decision-making, focusing on scenarios where the human's interaction with the robot is reduced to simply predicting the robot's intentions, taking the role of an observer. However, human interaction with robots and other AIAs extends beyond such a passive role: humans can share the same workspace as robotic agents, requiring adaptation to the agent's movements; humans can work in teams or other collaborative organizations to achieve a common goal; among other kinds of interaction. Thus, it is important to understand the impact of legible decision-making in scenarios where the human or other artificial agent has a more active role in the interaction and does not solely have to predict intentions and observe behaviours.

In this chapter we explore question **RQ 3**, exploring how the efficiency of a collaborative task is affected when agents use legible decision-making. We start this Chapter by presenting, in Section 6.1, the formalism of multi-agent Markov decision problem (MMDP), a formalism for multi-agent scenarios we are going to use throughout this Chapter. In Section 6.2 we motivate the problem that drives the work of this Chapter and in Section 6.3 we describe how we adapted PoLMDP to multi-agent scenarios, where the environment is not solely affected by the agent's actions. Finally, in Section 6.4 we present the study we conducted to analyse the impact of legible decision making in the efficiency of a team in a collaborative task.

6.1 Multi-agent Markov decision problem

Multi-agent Markov decision problems (MMDPs) generalize MDPs to multi-agent cooperative scenarios. MMDPs describe sequential decision tasks in which multiple agents must choose an individual action at each time-step that jointly maximize the underlying reward function. A MMDP is described as a tuple $M = \langle N, X, (A_k), P, R, \gamma \rangle$, with:

- X and γ are as defined in Section 2.3;
- A_k is agent's k individual action space;
- N is the number of agents in the scenario;

- P is the transition probability from a state x to y when the joint action $a = (a_1, \ldots, a_N)$ is taken;
- R is the reward signal all agents receive when execute joint action a in state x.

In a MMDP, a joint action is a tuple $a = (a_1, \ldots, a_N)$ and consequently $A = \times_{k=1}^{N} A_k$ denotes the joint action space. For $k = 1, \ldots, N$ we denote $A_{-k} = A_1 \times \ldots \times A_{k-1} \times A_{k+1} \times \ldots \times A_N$ the joint action space of all agents except agent k, leading to a joint action a also being denoted as $a = (a_k, a_{-k})$.

In MMDPs, a joint policy is a mapping from states to joint actions, $\pi \colon X \times A \to [0, 1]$, describing the action each agent should take in each state. A joint policy is a combination of N individual policies, *i.e.*,

$$\pi(x, a) = \prod_{k=1}^{N} \pi_k(x, a_k) \tag{6.1}$$

where $a = (a_1, \ldots, a_N)$. Like with the joint action, for $k = 1, \ldots, N$ we can denote the joint policy as $\pi = (\pi_k, \pi_{-k})$, where π_k denotes the individual policy of agent k and π_{-k} denotes the joint policy of all agents except agent k. Similarly to a MDP, solving a MMDP amounts to computing an *optimal joint policy* π and process to compute the optimal joint policy is indistinguishable from an ordinary MDP.

6.2 Teamwork and legible decisions

Teaming is a typical human approach in order to execute more difficult/complex tasks or to increase efficiency in solving a simple task. In team configurations is extremely important that team members clearly understand each other's intentions, so that they do not get in each other's way and the collaboration is efficient. Legibility can potentially play a crucial role in this type of interaction because it can function solely as a communication mechanism for humans and other agents to understand intentions; but also can contribute to the coordination of team members, by allowing other team members to better understand a specific member's intentions and react accordingly.

Consider a collaborative task of trash collection, where a team of agents has to collect trash spread over a large area. An agent, acting legibly, can improve the efficiency of the team in two aspects: first, by being clear about what items its going to pick, frees the other team members to pick other trash items; second, by performing legible actions, if it moves to pick a bigger trash item or an item harder to collect, the other agents are faster at understanding that agent's intentions and may be quicker in providing assistance. Another example of the possible influence of legibility in collaboration tasks, is Example 2 from Section 1.1.3 where a robot and a human play a collaborative of Word Blocks. In the example, the robot decides to make the word "boar" with the blocks available. However, by following optimal behaviour, conveys the intention of making the word "boat" to the human by first picking and placing the "b" block and then, after the human places the "a" block, places the "o" block between the "b" and "a" blocks. This sequence of actions leaves the human with two choices: place the "r" block at the end of the word, forming the word "boar"; or place the "t" block at the end, forming the word "boat". The human chooses to place the "t" block, thus forming the word "boat" and not the intended word "boar". If, instead, the robot had used a legible approach it could have verified that, after placing the "b" and "a" blocks, placing the "r" block after the "a" block would convey more information about the word the robot intended to make.

These two examples show how important it is for an autonomous agent to be clear when it needs to collaborate with other agents. Since it is a part of a team, the task execution no longer depends solely on the agent's actions but on the joint actions of all the members of the team. Thus, if our autonomous agent is not clear about its objectives it can hinder the task execution and the overall team performance; ranging from just a minor impact in team performance – because other team members need to pay more attention to the agent's actions to figure what the agent is trying to achieve – to possibly leading to a task failure – because the agent conveyed the wrong goals to the other team members, causing their responses to cause a failure in the task.

As explored in Section 5.2, PoLMDPs make use of legible decision-making to generate behaviours that are better at conveying intentions and personal goals to other agents. Thus, this framework shows potential in improving a team's efficiency, by making easier for the team mates to interpret our agent's actions and understanding its actions. However, PoLMDPs have been defined in the context of single-agent scenarios and so have to be adapted to use in multi-agent scenarios.

6.3 Multi-agent PoLMDP

In multi-agent scenarios the world state is not influenced by a single acting agent, but by the combination of actions of all the agents acting in the environment. Thus, an agent's decision making process has to take the influence of other agents into consideration when deciding on the best course of action. This makes the decision process more complex than in single-agent scenarios, because each agent's individual optimal policy needs to account for other agents executing varying actions. Thus, an agent's individual optimal policy becomes a best response to the other agents' actions, creating a *Nash equilibrium* [78] with the other agents – no single action can improve its payoff by unilaterally switching its action policy.

Multi-agent systems fall under one of two modalities when considering how to reach a Nash equilibrium: adversarial games, where the Nash equilibrium represents the best response in maximizing the agent's reward, while minimizing the other agents' rewards; or cooperation games, where the agents compose a team and the Nash equilibrium represents the best response that maximizes all agents' rewards [68]. In the case of teamwork we use the modality of cooperation games because the agents share the same goal. The formalism of MMDPs offers a simple approach to model cooperation games: the optimal joint policy decides each agent's action in a centralized manner that maximizes the team's payoff, almost as if each agent was an extension of a single entity.

The joint behaviour of a MMDP allows for a very good coordination between team members, however it does not correctly model how a real interaction occurs. In real world interactions, each agent acts independently from the other team members, without a central decision process deciding how each agent acts. Thus, each agent executes its own action, observes how other team members react and the changes their joint actions have on the world. The application of PoLMDPs is thus useful, because it gives this independently acting agents more insights into the decision process of each other. Allowing for the reaction to each other's actions to be more informed, which in turn leads to better responses from the team members because they have access to more information. However, to apply PoLMDP to coordination games we need to adapt the framework, defined for single-agent scenarios, to multi-agent cooperation games scenarios, while retaining each agent's expressive action independence.

The central aspect of the PoLMDP framework is the definition of a legible reward that an agent tries to maximize the expected long-term return of. Equation 5.4 defines the PoLMDP's legible reward as being the agent's action that more clearly shows the agent's goal, among all actions possible in state x. This way, the legible reward is indexed to the current world state x and to the agent's individual action a. In multi-agent scenarios, the way the world progresses is tied the joint actions of all agents, instead of tied to each individual agent's actions. So, to apply PoLMDPs to these scenarios we need to approximate how the world is impacted by the actions of a specific agent. We obtain this approximation via the use a MMDP.

Solving a MMDP constitutes computing a joint optimal policy π^*, which is a combination of the agents' individual policies $\pi_1^* \times \ldots \times \pi_N^*$. So, from π^* we can denote two sets of policies: π_{ca}^* that denotes the individual optimal policy of our controlled agent ca, and π_{-ca}^* that denotes the joint optimal policies of all other agents except our agent. If we fixate all other agents to follow the actions prescribed by π_{-ca}^*, we can define a standard MDP with the same state space and discount factor as the MMDP; the action space is the action space for agent ca, A_{ca}, in the MMDP; transition probabilities for each action $a \in A_{ca}$ is given as follows

$$P_a(y \mid x, a) = \sum_{a_{-ca} \in A_{-ca}} P^{\text{MMDP}}(x_{t+1} = y \mid x_t = x, a_t = (a, a_{-ca})) * \pi_{-ca}^*(x, (a, a_{-ca})),$$

with $x, y \in X$, A_{-ca} is the joint action space of MMDP without agent ca and P^{MMDP} are the transition probabilities of the MMDP; and reward function given as follows

$$R(x, a) = \sum_{a_{-ca} \in A_{-ca}} R^{\text{MMDP}}(x_t = x, a_t = (a, a_{-ca})) * \pi_{-ca}^*(x, (a, a_{-ca})),$$

with $x, y \in X$ and R^{MMDP} is the reward function of MMDP.

Then, to apply PoLMDP in multi-user scenarios, we first solve a MMDP for each of the possible objectives in the environment. Secondly, with all the MMDPs, we do the process just described to approximate each MMDP to a single-agent MDP and solve each MDP obtaining the optimal Q-function. And finally, having the optimal Q-functions for each MDP, we can define a PoLMDP, for our agent's goal, with the same state space, action space, transition probabilities and γ as the MDP for the same goal, and legible reward function as defined by Equation 5.4.

6.4 Legible Teamwork Experiments

Legible decision-making allows for an agent to compute the actions that are both more efficient and more expressive regarding its own personal intentions. This is a useful skill for tasks that require teamwork, since for correct teamwork the team members need to correctly communicate their intentions and be clear about their goals.

In this section, we present a study conducted that explored the impact of legible decision making in a team's efficiency while executing a task of collaborative food foraging. We used the LB-Foraging benchmark scenario [81], where a team of agents had to pick all the food items in the environment. Our team of agents was composed by two agents: a leader agent, who knew the sequence to pick each item, and a follower agent, who did not know the food sequence and had to predict the next item to pick. We chose this scenario because there are no explicit communication means, forcing the follower to deduce the leader's intentions from its actions, mimicking situations in the real world where communication devices fail or there are interference that makes explicit communications impractical. Also, by making both agents have to work together to pick each food item, we place the collaborative effort as an essential aspect of the task and so we can better measure the influence of legible actions in the team performance. Finally, by enforcing a strict hierarchy of leader-follower with only the leader knowing the sequence to pick the food items, we place more weight on the clear communication of intentions between the two team members.

6.4.1 Scenario Setup

In this work we used the scenario of Level-based Foraging from Christianos, Schäfer, and Albrecht [21]. This scenario is a benchmark for multi-agent reinforcement learning that allows for an arbitrary number of agents to fulfill a task of foraging in the scenario for food in both pure collaborative or competitive-collaborative capacities. Figure 6.1 shows an example of the level-based foraging scenario.

The scenario of level based foraging we used is composed by a grid of NxM squares – with N the number of rows and M the number of columns in the grid – and each agent has available the actions of moving one square up, down, left or right, pickup a food item in an adjacent position and not doing anything. Each agent in the environment has a level and so has each food item. Each agent can only pick food items that have a level equal or less than the agent's level. In case of the food item having a level greater than the agent's, the agent has to cooperate with other agents so that the sum of their levels is greater or equal than the food's level. This level restriction allows to force pure cooperative tasks, by making the level of the agents always less than the food items, or competitive-collaborative by making the foods' levels not all equal, with some food items able to be picked without collaboration and others requiring collaboration. Each agent receives a reward for the items it participates in picking, with the rewards being equally divided in pure collaborative scenarios and divided according to the contribution – agents with higher level get higher rewards – in competitive-collaborative scenarios.

In our study we used a pure collaborative scenario since this placed more focus on the team

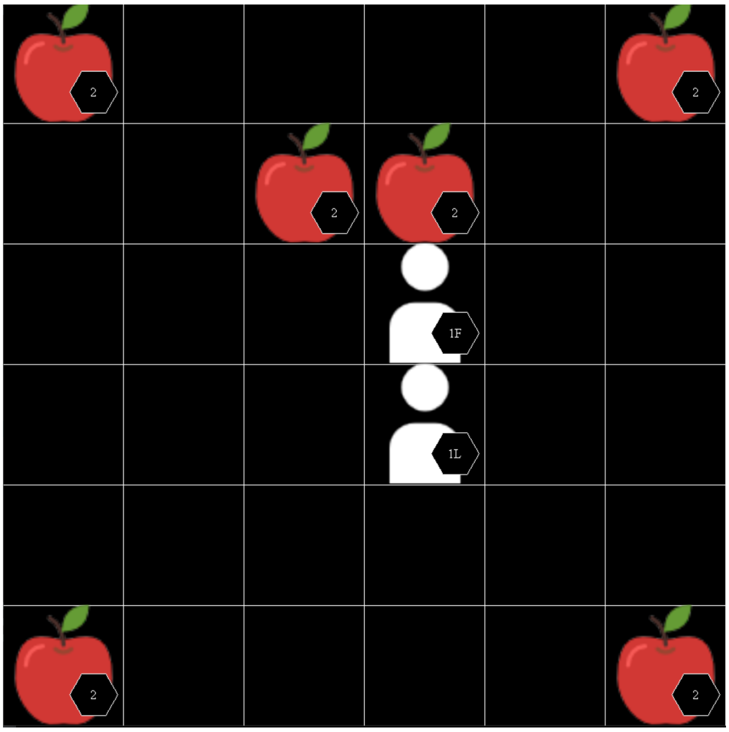

Figure 6.1: Level-based foraging benchmark example. In the image we can se the two agents and six food items spread in the environment. The food items are all level two foods – marked by the "2" in the corner of the food item – and each agent is at level one – marked by the "1" in the agentś sprite corner. Also each agent either has a "L" or a "F" next to the level, marking if the agent is the leader or the follower.

aspect and on the impact of legibility. Thus, each food item's level was equal to the sum of the levels of the two agents in the team. We set up 3 different size grids – 5×5, 6×6 and 7×7 squares – and for each grid we chose eight possible locations for the food items, for which six were randomly sampled in each run. We had to choose specific locations because the scenario is not built for the food items to be picked in a specific order and, in some configurations, the environment registered the pickup action on the wrong item. Thus, by pre-screening possible positions we avoided this limitation of the environment.

6.4.2 Team Composition

The study used a pure collaborative setting, so to pick a food item both agents had to be in different positions, adjacent to the item either horizontally or vertically, and execute the action "Pick". The agents in the team had different roles: one was the leader, knowing the sequence to pick the items and responsible with relaying that information to the other team member; and one was the follower, which had to infer the correct food to pick from the leader's actions. Also, with the agent role was associated a social convention [9] defining where each agent would position itself to pick the food item – the leader would pick from a vertically adjacent location and the follower from an horizontally adjacent location. The social convention used offered a simple mechanism for coordination by simply shaping the reward signal received by each agent depending on the agent's role; this avoided the agents trying to move to the same adjacent locations when picking the items.

Agent Implementation

The scenario of LB-Foraging used is a multi-agent scenario, however to test the impact of legibility in team performance we decided that a non-centralized approach was better at simulating a real world scenario. As such, we first defined a MMDP as follows:

- N was 2;
- the state space X was a set of tuples of the form ((x_l, y_l), (x_f, y_f), p), where (x_1, y_1) described the position of the leader agent, (x_f, y_f) described the position of the follower agent and p described if the food item had been picked or not;
- the action space A was defined by the environment and allowed each agent to execute the actions "Up", "Down", "Left", "Right", "Pick" and "Noop";
- the transition probabilities P was given by the environment and always resulted in the success of the executed joint action;
- the reward signal R was given by the environment as 1 when the agents picked the food item and 0 otherwise;
- the discount factor γ was 0.9.

After solving the MMDP we approximated a MDP for each agent using the process described in the previous Section, thus decoupling each agent's actions and having a decentralized decision process. The different roles required that each agent had different capabilities, so the exact implementations of each agent differed slightly depending on the agent's role in the task.

The leader agent knew the sequence of food items to pick and had to implicitly communicate it to the follower agent. Thus, the leader used one of two types of MDPs: for the optimal leader, we used directly the approximated MDP that optimized the underlying reward function, thus resulting in an optimal policy π^*; for the legible leader, we used a PoLMDP that optimized a legible reward r_{leg}, thus resulting in an optimal legible policy π^*_{leg}. At each decision point, the leader would take its observation and use either π^* or π^*_{leg} to determine the best action to reach and pick the food item.

The follower always implemented the approximated MDP and optimized the underlying reward function. However, it did not know the correct food picking sequence and had to infer it from the leader's actions. Thus it used a Bayesian inference process as in GIRL [69] to infer the next food item to pick. To that end, the follower used the history of the observed leader's states and actions, since they picked the last food item, and matched the observed states and actions to the optimal Q-functions of each remaining food item, finding the one with highest probability of yielding the observed sequence of state-action pairs.

The implementation of the agents' MDPs also differed on the reward function. Since the agents acted in a decentralized manner, we used a coordination mechanism of social conventions, encoded in the reward function, to help with coordinating the agent's actions and avoid the agents getting stuck trying to reach the same position. As such, we established that the leader agent could only pick a food item when if was vertically adjacent to the item and the follower agent only when was horizontally adjacent to the item.

6.4.3 Performance Evaluation

This work focused on exploring the question:

"Does the use of legible decisions improve the efficiency of a team in collaborative tasks?"

To support the exploration of our problem we postulate the following working hypotheses:

H1 *Team performance will increase when agents use legible behaviours*.

H2 *Agents observing legible behaviours will be able to understand other agents' intentions faster*.

The study conditions were reflected on the team composition: one with both agents using an optimal policy and the second composition where the leader used a PoLMDP policy and the follower agent used an optimal policy. In the rest of the text we will refer to the condition of both agents using an optimal policy as the *optimal condition* and for the condition with the legible leader and optimal follower as the *legible condition*.

We measured the impact of legibility in terms of team performance by recording the average number of steps needed for each team to finish the task; and the impact in terms of intention conveyance by recording the average number of steps for the follower correctly infer the leader's objective, without a posterior wrong inference. We collected these metrics for three different LB-Foraging grid sizes – 5×5, 6×6 and 7×7 squares. By varying the grid sizes, we were able to understand the impact of legibility when the agents have more freedom of movement and the decision processes are longer to reach the goal.

For each condition and grid size we ran 250 simulations, where the agent team had to pick 6 food items uniformly sampled from 8 possible initial positions. To guarantee that both conditions were exposed to the same food configurations, we pre-sampled all 250 simulation scenarios, for each grid, before running the simulations. For each simulation we sampled the initial 6 food positions, the initial agent positions and the sequence to pick the food items. Pre-sampling the sequence to pick the food items was important to ensure that one agent composition did not benefit from having sequences that led to easier predictions. Also, pre-sampling the sequence guaranteed that when trying to pick an item, that item had at least one horizontal and one vertical adjacent position free.

Finally, each simulation would end either when the team picked all 6 food items or if the agents needed more than 200 steps to pick all the food items. Adding a maximum number of steps established a time limit to solve the task, better representing the real world where the agents would have a maximum amount of time to solve the task. If team would go over the 200 step limit, that attempt was marked as a failure to finish the task.

6.4.4 Evaluation Results

After running all simulations we got 500 samples for each grid size. We then proceed to validate the results obtained and ended up removing one sample from both the 5×5 and 6×6 grids and 5 from the 7×7 grid. We removed these results, because all of them reported taking longer than 200 steps, but when analysing the history of states and actions, we observed that the agents got stuck on a cycle of trying to pick a food item and the environment not recognizing that action. Thus, we removed the corresponding samples for both team compositions because this constituted an environment error. With the error samples removed we had 249 samples for each agent composition in both the 5×5 and 6×6 grids and 245 samples for each composition in the 7×7 grid.

With the final samples we logged, for each team composition and each test run, the number of steps taken to pick all the food items and the average number of steps to correctly infer the leader's objective. With the logged measures, we computed their averages for each condition and grid size. We observed that besides the runs that failed due to an environment error, no more runs ended in a failure, so we did not need to differ metric calculation to account for failures. We also conducted a normality test on the results, which showed that their distribution significantly deviated from the normal distribution. Since the data did no follow a normal distribution, we only conducted non-parametric tests.

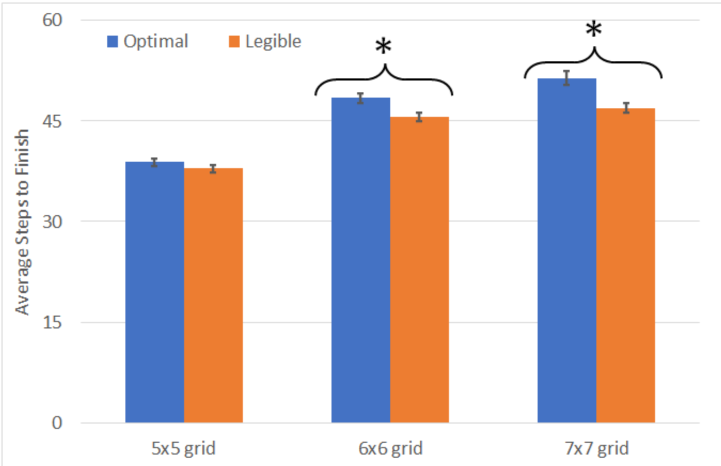

Figure 6.2: Results for the average number of steps to pick all the food items in the world, in black bars are the standard error bars. In blue we have the results for the optimal condition, while in orange for the legible condition. (*$p < 0.05$)

Figures 6.2 and 6.3 show the plots with the results. Figure 6.2 shows the average number of steps needed for each team composition to pick all six food items for the different grid sizes. Figure 6.3 shows the average number of steps needed for the follower agent to correctly infer the next food item to be picked, for the different grid sizes. In both plots the blue bars show the results for the optimal condition and the orange bars show the results for the legible condition.

The results in Figure 6.2 show that the agents in the legible condition achieved a better or equal performance than the agents in the optimal condition. In fact, for both the 6×6 and 7×7 grids the team in the legible condition took $\simeq 46$ steps to pick all the food items, while the agents in the optimal condition took $\simeq 48$ and $\simeq 51$ steps on average to pick all the food items. Only on the 5×5 grids did the performance of both team compositions was similar with the agents in the optimal condition taking on average 39 steps to pick all the food items and the agents in the legible condition taking on average 38 steps. These conclusions were validated by Mann-Whitney U test that showed that there are significant differences between the average number

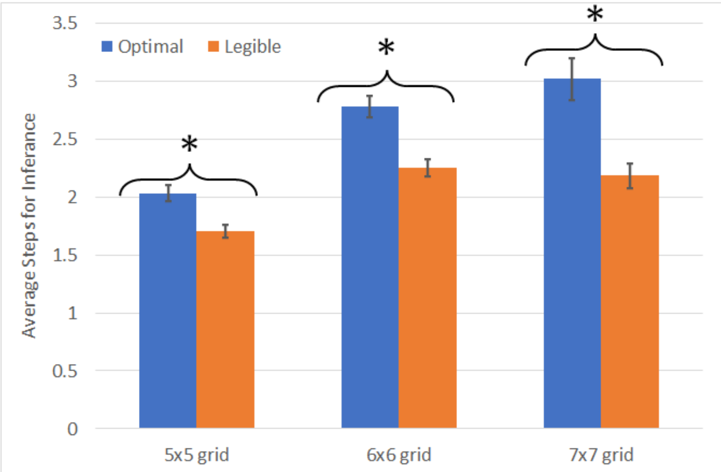

Figure 6.3: Results for the average number of steps to correctly infer the next food item to be picked, in black bars are the standard error bars. In blue we have the results for the optimal condition, while in orange for the legible condition. (*$p < 0.05$)

of steps in the 6×6 and 7×7 grid results. Table 6.1 shows the results of the conducted Mann-Whitney U test, with higher mean ranks representing conditions with higher average number of steps to pick all the foods in the corresponding grid size.

Grid size	Optimal condition Mean rank	Legible condition Mean rank	*p* value
5×5	256.39	242.61	0.285
6×6	268.76	230.24	0.003
7×7	269.59	221.41	< 0.001

Table 6.1: Results of the conducted Mann-Whitney U test for the average number of steps to pick all food items, between the optimal and legible conditions for the different grid sizes. The condition with higher mean rank has the highest average number of steps.

The results in Figure 6.3 for the average number of steps to correctly predict the next food item to be picked, show that for all the three grid sizes the legible condition allowed for the follower to infer the next food to be picked quicker than in the optimal condition. In the case of the 5×5 and 6×6 grids, the legible condition team got an average improvement of around half a step in inferring the next food item. In the case of 5×5 grids the average reduced from $\simeq 2$ steps in the optimal condition to $\simeq 1.7$ steps in the legible condition; and in 6×6 grids the average number of steps reduced from $\simeq 2.8$ steps in the optimal condition to $\simeq 2.3$ steps in the legible condition. In the case of the 7×7 grids, the legible team showed an average improvement of almost a full step less in inferring the next food item, decreasing from $\simeq 3$

steps in the optimal condition to $\simeq 2$ steps in the legible condition. Again, to understand if the differences between the conditions had statistical significance we conducted a Mann-Whitney U test. Table 6.2 shows the results for the Mann-Whitney U test, where we can observe that for all three grid sizes the differences between the teams were significant with the legible leader capable of being clearer about the next food item to pick.

Grid size	Optimal condition	Legible condition	*p* value
	Mean rank	Mean rank	
5×5	266.55	232.45	0.08
6×6	276.43	222.57	< 0.001
7×7	273.89	217.11	< 0.001

Table 6.2: Results of the conducted Mann-Whitney U test for the average number of steps to infer the next food item, between the optimal and legible conditions for the different grid sizes. The condition with higher mean rank has the highest average number of steps.

6.4.5 Discussion

The results in Section 6.4.4 allow us to conclude that overall the performance of the team with the legible leader was higher than the performance of the team with the optimal leader. The results of the average number of steps to pick all the food items show statistical differences between the two team compositions in the bigger grids, with the performance of the two teams being equal in the case of the 5×5 grids. The performance in the 5×5 grid being similar can be explained by the fact that the environment was heavily constrained in terms of free cells, and the distance between the agents and the food items was never so high, that the extra time it took to predict the food items would translate in a higher loss in performance. However, the overall results are in line with our hypothesis **H1** that postulated that using legible behaviours would increase team performance.

The second major conclusion from the results is: using legible behaviours led to the follower agent to infer the next food item and consequently the food sequence faster. This result is in line with our hypothesis **H2** that postulated that agents observing legible behaviours would understand the intentions of other agents faster. Besides validated **H2**, the result from the average number of steps needed to infer the food item yields an indirect impact of hypothesis **H1**. This comes from allowing a faster understanding of the teammate's intentions, the follower was able to collaborate more efficiently, leading to a better team performance.

Finally, there is a third conclusion from the results that arises when analysing the plots: there is a trend for the legible team's performance to stabilize while the optimal team tends to take

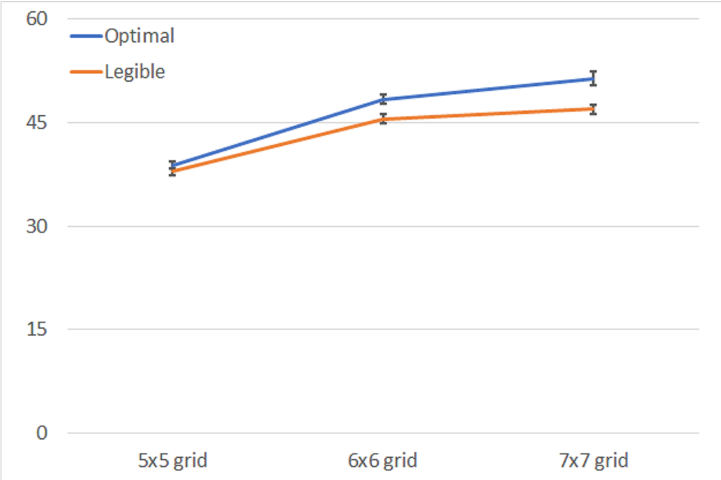

Figure 6.4: Trend line between the averages to pick all the food items, in black bars are the standard error bars. In blue the line for the optimal condition, while in orange for the legible condition.

linearly more time to complete the task. If we plot a trend line between the averages to pick all food items for each condition, we observe that the optimal condition increases at a faster rate than the legible condition. Figure 6.4 shows a plot with the trend lines for the average number of steps to pick all food items. If we perform the same analysis to the results of the average number of steps to infer the next food to pick, we observe that again the growth rate of average number of steps required starts to stabilize – and even decreases slightly – in the bigger grids in the legible condition. Figure 6.5 shows a plot with the trend lines for the average number of steps to correctly infer the next food item.

This conclusion is of interest because it hints at the fact that in bigger grids – and consequently bigger state spaces – a legible agent can make better use of the less constrained workspace to find paths that lead to better intention communication and better team efficiency. This conclusion is further accentuated by the trend line in Figure 6.5 that shows that in bigger grids the follower agent starts to need less steps to infer the correct food item. A legible agent being able of making better use of more unconstrained environments is extremely interesting, because it shows the agent can find decision sequences that take advantage of the options a more unconstrained environment offers to increase the team's performance by increasing the information conveyed to the teammates.

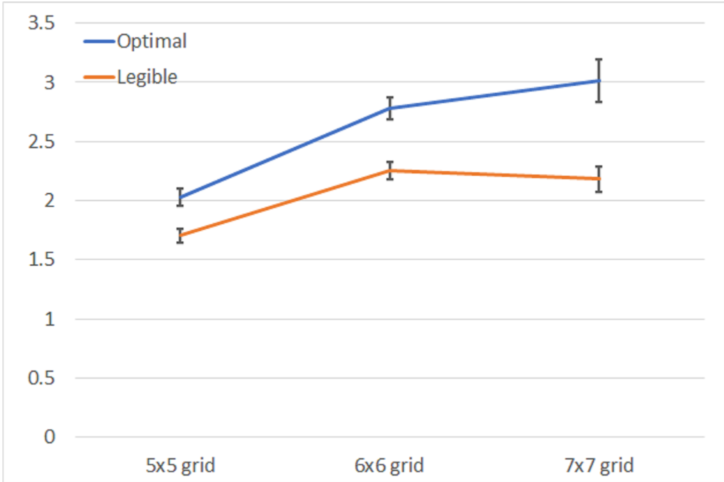

Figure 6.5: Trend line between the averages to infer the next food item, in black bars are the standard error bars. In blue the line for the optimal condition, while in orange for the legible condition.

6.5 Chapter summary

This Chapter explored the impact of legible decision-making in team scenarios, specially in team scenarios that require strict collaboration to complete the task. We first explained how we can apply PoLMDP (previously defined to single-agent scenarios) to multi-agent scenarios: we start with a MMDP and then approximate the obtained model to a MDP, which can be used to extract the legible reward function needed for PoLMDP. Finally, we applied PoLMDP in a collaborative task of level-based foraging with an explicit hierarchy of leader-follower and showed that a team with a legible agent achieves better performance than teams that have only optimal agents.

7

Conclusion and Future Work

In this work we focused on applying the notion of legibility, a notion previously used in robotics to generate movements that improve a robot's communication capabilities, to multi-user interactions and to decision-making. We addressed the question:

Research Question *Can a robot, in interactions with multiple human users, be more expressive and clear regarding its goals using legibility?*

We explored this questions under two avenues that are crucial for improving robots and other autonomous agents' expressiveness: (i) the way a robot moves in interactions with multiple humans; and (ii) the way an agent decides on a sequence of actions when interacting with humans.

7.1 Work contributions

We explored how a robot can improve its expressiveness using legible movement, in multiple user interactions, by focusing on generating movements that do not improve the legibility for a single human, but instead to all the human partners interacting. This rationale comes from the fact that, during a multi-party interaction, it does not matter improving a specific user's understanding of a robot's movements; if the others do not understand the robot's intention, they might interfere with the movement, hindering the interaction and even hurt themselves or damage the robot.

To avoid such scenarios, in Chapter 4, we introduced the notion of *multi-user legibility* (MUL). This notion allows to create legible movements that improve an overall group legibility, instead of an individual user's legibility, while considering that each human has its own point-of-view over the movement. Thus, the resulting motion is kept in view of all the humans and improves the understanding of all the partners of the robot's intentions. We achieve this behaviour by merging each individual partner's legibility, according to its perspective, into a single legibility metric which then is iteratively optimized.

On a set of simulated scenarios of a robotic arm reaching to grasp one of three items on top of a table, MUL generated movements that achieved better group legibility, than approaches that focused on improving the legibility for a specific user. On a user study, that again compared MUL with approaches that focused on specific users, we found that humans that observed MUL generated movements, were able of correctly understanding the robot's objectives quicker and with more confidence across different perspectives, than when observing movements generated with approaches that focused on improving the understanding of human partners in specific perspectives. The results of both the simulated scenarios and the user study, suggest that using MUL significantly improves human understanding of a robot's intentions, in multi-party interactions.

Regarding legible decision making, in Chapter 5, we introduced *policy legible Markov decision problem* (PoLMDP), a framework for legible decision making in stochastic decision making based on the formalism of MDP. PoLMDP was developed with the idea that when a human is interacting with a robot, or other type of artificial intelligent agent, the human only observes the outcome of the actions executed by the robot. Thus, the robot's decision making process must select actions that at each decision point, give more information about the robot's internal state, without requiring the robot to make explicit which action is going to execute.

PoLMDP generates legible actions using a two-step process: first, it solves a MDP for each possible goal of the agent or robot and then, with all the MDPs solved, the framework computes a legible reward function that promotes actions that are more indicative of the robot's goal. Thus, the actions generated by PoLMDP are a subset of optimal actions, for each underlying MDP, that are most representative of the objective trying to be reached.

Using a simulated scenario of a robot solving a maze, trying to reach a specific coloured objective, we showed that PoLMDP is capable of performances better than the current state-of-art approaches for legible decision making, solving the maze faster while maintaining similar levels of legibility. Using the same simulated scenario, PoLMDP was able to teach an agent how to solve the maze faster than with a MDP agent that simply found the optimal solution. We replicated the simulated scenario in an online user study, where the participants played a guessing game with the robot, trying to guess the robot's objective as quickly as possible. We compared solutions given by PoLMDP with solutions given by standard MDPs and observed that the solutions of PoLMDP allowed humans to predict the robot's objectives faster, more consistently and with more confidence. Thus, using legible decisions improves the interactions between humans and robots, giving humans better insights to robots' internal states and intentions.

Finally, we explored legible decision making in the context of teamwork and teamwork efficiency. In Chapter 6, we conducted a set of simulations that compared a team composed by two optimal agents with a team composed by a legible agent and an optimal agent. We used a leader-follower paradigm to enforce the impact of legibility on the task. The agent team had to collaborate to pick food items spread in a grid environment and only the leader agent knew the next food item to be picked. The teams that had a legible PoLMDP agent were, on average, capable of picking all the food items faster, requiring less time for the follower agent to infer the next food item. The difference in team performances increased with the size of the grid, suggesting that for bigger grid sizes having a legible leader agent significantly increases team performance.

Overall, with this thesis we contributed with:

- a novel notion of legibility for multi-party robotic movements that takes into consideration the different partner's perspective and showed through a user study that this novel notion improves human understanding of a robot's movements; this contribution has been published in [36].
- a novel framework for legible decision-making in both stochastic and deterministic scenarios, dubbed PoLMDP, that selects, at each decision step, the optimal action that best represents of the agent's underlying goals; this contribution can be found in ArXiv [37] and is currently under reviewing process for the Elsevier's Artificial Intelligence Journal.
- a study on the impact of legible decision-making in strict collaborative tasks where the team members need to communicate personal goals to each other, we showed that teams with agents capable of legible decision-making can achieve better performance than teams only with optimal agents; this contribution is currently under review for the IJCAI conference.

7.2 Impact on human-robot interaction and explainable AI

The work in this thesis shows that legibility can improve an artificial agent's expressiveness and clearness in interactions with multiple users. Our work, by branching into both robotic movement and more general agent decision-making yields interesting impacts on both *human-robot interaction* (HRI) and *explainable artificial intelligence* (XAI).

The notion of MUL, we contributed in this work, is capable of making a robot's movement clearer in interactions with multiple users, which is extremely important in making robots easy to interact and collaborate with and thus improving human-robot interactions. Making a robot's movements easier to understand by all the human partners has the direct impact of making interactions more fluid and efficient, since humans will require less time to understand the robot's intentions and have more time to react accordingly. This improved team efficiency is important to integrate robots in applications such as in healthcare areas like surgery teams where a robot must communicate correctly and clearly for the rest of the medical staff to perform adequately. Other application where correct movement communication has a direct impact is in entertainment areas such as theatre where movement is an important communication tool and where people may need to observe the movements from different perspectives.

Moreover, on a more subjective aspect, by having clearer movements, the humans interacting with the robot will be able to rely more on its capabilities and will need to divert less time to control a robot's actions. In turn, this will lead to increased human trust more in robots not only as tools, but also as peers and social entities when sharing workspaces and physical locations. Thus, reducing human scepticism towards robots and help robots to be more welcome in domestic settings, where humans have a tendency to be more sceptical robotic presence.

The combination of improved efficiency in interacting with humans and increased human trust in robots' abilities, can lead to robots becoming more integrated in society; with humans finding it easier to understand how to interact and collaborate with robots, thus finding more uses to the advantages robot bring to society.

The exploration of legible decision-making, in the form of the novel framework PoLMDP and analysis of its impact in both single and multi-agent scenarios impacts both HRI and XAI. The development of legible decision processes gives robots more transparency during interactions with humans, because it gives humans more insight and understanding about the reasoning behind a robot's actions. This increased understanding directly impacts human-robot interactions twofold: primarily, helps humans in quickly understanding a robot's intentions and what are its underlying goals, which allows for interactions to occur more naturally without humans requiring extra time learning how to interpret a robot's behaviours to extract meaning from them; secondly, when a robot fails during an interaction with a human or does an action that is not expected by the human user, the human user can trace the reasoning behind this unexpected behaviour and more easily understand if the cause was a robot's malfunction or something in the way the human behaved that forced the robot to react that way.

Besides these more direct impacts, interactions between humans and robots can be indirectly impacted by the increased understanding of a robot's reasoning in terms of the trust in the robots. Human trust in someone or system increases more quickly when the human can understand the reasoning processes and so can more easily predict that person or system future behaviours. This is especially important when a robot's exhibits unexpected behaviour, humans have shown to recover more quickly trust in the robot if they can understand what caused that behaviour [88].

Legible decision-making has similar impacts on XAI, when we consider XAI systems in the context of human-agent collaborative interactions. Humans that have to collaborate with non-robotic autonomous agents, show more confidence in interacting when they understand how an autonomous agent works and even trust those agents more. For example, using legible sequential decisions, we can have virtual agents that function as advisors in search and rescue operations, counselling team leaders by generating plans that are easier for humans to understand. This way, team leaders can possibly trust more on delegating to autonomous agents operational planning and focus more on managing the personnel during operations.

But the most interesting impacts of our work in XAI come when we consider XAI that do not have to work side-by-side with humans. Having an autonomous agent that is more transparent and better at conveying its goals, opens a whole new range of applications to autonomous agents. For example, we can have autonomous agents being used to teach other agents, thus making easier to share acquired knowledge across different agents, possibly making agents more flexible in their applications and easier to deploy in new tasks. We can also use such agents to teach humans new skills, with frameworks capable of recognizing what are the most salient features and behaviours to display, we can develop systems to support teaching of new skills to humans.

So, frameworks like PoLMDP can pave the way for more wider acceptance of autonomous agents – physical, virtual or simply as applications – as peers and trust on their capabilities without strict human oversight.

7.3 Future work

With this thesis, we have contributed with a novel type of expressive motion for human-robot interactions, a novel formalism for decision making applicable in both human-robot and human-agent interactions and novel insights on the impacts of legibility in interactions between humans and robots. However our contributions are not without their limitations.

On the avenue of legible movements, we introduced a model that allows for the generation of expressive movements in multi-party interactions, but our approach relies on correctly define the perspectives of the human users. As we found during this work, correctly defining the perspectives of human users over a task is not always easy and sometimes a slight error in the definition of the perspective may lead to harder to understand movements. So, having a robot learn how to be simultaneously legible to multiple humans from repeated interactions, is a possible alternative to the current method of defining and optimizing a legible cost function. With such an approach, we hope to fix a limitation on our contribution by giving the robot more flexibility in interactions, using more information of the workspace more than just the human perspective to support the generation of legible motions. Also, having the robot learn how to be legible can lead to more interesting models, capable of generalizing the concept of legibility to understand when to execute a legible movement or a more simple and straight movement, better resembling human reasoning is similar scenarios.

Another approach we leave for future work is to understand how legible movements combine with other communication means – either explicit or implicit – to improve robot transparency in both single and multi-party interactions. Humans do not use communication methods, like body movements, isolated; usually they use combinations of multiple communication methods to better convey intentions and overcome limitations that each communication method has. For robots to better integrate in society and live in the middle of humans, they also have to be capable of combining multiple communication means to better convey intentions. For example, legible movements are extremely good at conveying intentions through movements, however their usefulness declines when the view of the movement becomes obstructed or if the robot's intention cannot be conveyed solely with movements. Thus, by exploring how to combine legible movements with other communication means, we expect to reduce the limitations of communicating through legible movements by empowering robots with other communication tools.

On the avenue of legible decision making, we explored the impact of legibility in teams in simulation scenarios, however in physical human-robot interactions, humans can react differently than the agents in the simulations, *i.e.*, humans do not follow an optimal policy or focus solely on being efficient. Human reasoning does happen over long time horizons, humans observe the world around them and act as they believe is better given the current situation. Conversely, human goal recognition is also not based on optimal actions, but in personal history and bias. So, in a real world interaction with humans, both human actions and goal recognitions may sometimes deviate from what is observed in simulations. Thus, we are interested in replicating our simulation studies from Chapter 6 with real humans and investigate if the results in simulated environments translate to physical environments.

Collaborative tasks are not restricted to strict collaboration scenarios, where team members must always work together to finish the task; in some tasks, the sub-tasks can be executed independently. Since all team members have to deduce what other members are trying to achieve, the inference process is no longer simply one-way but is between all team members simultaneously. In these kinds of tasks, legibility may play a bigger role in team coordination than in the type of tasks we explored. Thus, we are interested in explore how legible decision-making impacts collaboration and coordination in this type of tasks.

Finally, our proposed PoLMDP framework was shown to allow efficient computation of legible plans in both stochastic scenarios. However, by using a standard MDP formulation, it requires to keep a model of the environment to compute a legible policy and in high-dimension state spaces this requirement may cause problems of memory. Thus, we are interested in investigating other approaches that may lighten the requirement for memory. Namely, currently we are interested in investigating approaches in two directions: hierarchical MDPs and model-free reinforcement learning. With hierarchical MDPs, we are interested in understand if we can decompose the problem into a hierarchy of sub-tasks and still generate legible behaviours, because the decomposition into smaller sub-tasks may create artefacts that may influence legibility, such as the creation of new intermediate goals. With model-free reinforcement learning, we expect to investigate two aspects: the first is if we can learn the notion of legible action with reinforcement learning, thus solving the limitation of needing to maintain a full model of the environment; and second is investigating if we can generalize the notion of legible actions to similar tasks, thus making legible agents easier to apply to new tasks.

7.4 Final Words

Robots and other autonomous agents are becoming a common sight in society, no longer restricted to be seen solely as tools. They can be found in the most varied roles in close proximity and interacting with humans. For their correct integration, correct communication of intentions and internal state is crucial for mutual understanding.

In this work we showed how legibility can be applied in different scenarios, using different both movements and decision making processes, to improve interactions in both human-robot and human-agent interactions. Through this work, we showed how agents using legibility are consistently rated as clearer and easier to interact with.

However the work on legibility is not over, as many open challenges remain to be addressed. Thus, this thesis is one more step towards creating autonomous systems that are more transparent and explainable about their intentions; contributing to the greater goal of integrating autonomous agents in society, in a manner that everyone can easily interact with without the need for advanced knowledge in robotics or artificial intelligence.

Bibliography

[1] Behnoush Abdollahi et al. “Transparency in fair machine learning: the case of explainable recommender systems”. In: *Human and machine learning*. Springer, 2018, pp. 21–35.

[2] Henny Admoni et al. “Social eye gaze in human-robot interaction: a review”. In: *Journal of Human-Robot Interaction* 6.1 (2017), pp. 25–63.

[3] Victoria Alonso et al. “System transparency in shared autonomy: A mini review”. In: *Frontiers in Neurorobotics* 12.November (2018), pp. 1–11. ISSN: 16625218. DOI: 10.3389/fnbot.2018.00083.

[4] S Anjomshoae et al. “Explainable Agents and Robots: Results from a Systematic Literature Review”. In: *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. Aamas. 2019. URL: http://www.diva-portal.orghttp//urn.kb.se/resolve?urn=urn:nbn:se:umu:diva-158024.

[5] Alejandro Barredo Arrieta et al. “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI”. In: *Information fusion* 58 (2020), pp. 82–115.

[6] Kim Baraka et al. “Expressive lights for revealing mobile service robot state”. In: *Iberian Robotics Conference (ROBOT)*. Springer. 2016, pp. 107–119.

[7] Andrea Bauer et al. “Human–robot collaboration: a survey”. In: *International Journal of Humanoid Robotics* 5.01 (2008), pp. 47–66.

[8] Roger Bemelmans et al. “Socially assistive robots in elderly care: A systematic review into effects and effectiveness”. In: *Journal of the American Medical Directors Association* 13.2 (2012), pp. 114–120.

[9] Craig Boutilier. “Planning, learning and coordination in multiagent decision processes”. In: *TARK*. Vol. 96. 1996, pp. 195–210.

[10] Cynthia Breazeal et al. “Effects of nonverbal communication on efficiency and robustness in human-robot teamwork”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2005, pp. 708–713.

[11] Baptiste Busch et al. “Learning Legible Motion from Human-Robot Interactions”. In: *International Journal of Social Robotics* (2017), pp. 1–15. ISSN: 1875-4791. DOI: 10.1007/s12369-017-0400-4. URL: http://link.springer.com/10.1007/s12369-017-0400-4.

[12] Ergun Calisgan et al. “Identifying nonverbal cues for automated human-robot turn-taking”. In: *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2012, pp. 418–423.

[13] Beatrice Capelli et al. “Understanding multi-robot systems: on the concept of legibility”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2019, pp. 7355–7361.

[14] Diogo V Carvalho et al. “Machine learning interpretability: A survey on methods and metrics”. In: *Electronics* 8.8 (2019), p. 832.

[15] Tathagata Chakraborti et al. “Balancing explicability and explanation in human-aware planning”. In: *arXiv preprint arXiv:1708.00543* (2017).

[16] Tathagata Chakraborti et al. “Explicability? legibility? predictability? transparency? privacy? security? The emerging landscape of interpretable agent behavior”. In: *International Conference on Automated Planning and Scheduling (ICAPS)* Icaps (2019), pp. 86–96. ISSN: 23340843. arXiv: 1811.09722.

[17] Shruti Chandra et al. “Children’s peer assessment and self-disclosure in the presence of an educational robot”. In: *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2016, pp. 539–544.

[18] Shruti Chandra et al. “Do Children Perceive Whether a Robotic Peer is Learning or Not?” In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. 2018, pp. 41–49.

[19] Mai Lee Chang et al. “Effects of integrated intent recognition and communication on human-robot collaboration”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2018, pp. 3381–3386.

[20] Tiffany L Chen et al. “Robots for humanity: using assistive robotics to empower people with disabilities”. In: *IEEE Robotics& Automation Magazine* 20.1 (2013), pp. 30–39.

[21] Filippos Christianos et al. “Shared Experience Actor-Critic for Multi-Agent Reinforcement Learning”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2020.

[22] Herbert H Clark. “Coordinating with each other in a material world”. In: *Discourse studies* 7.4-5 (2005), pp. 507–525.

[23] Roberto Confalonieri et al. “A historical perspective of explainable Artificial Intelligence”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 11.1 (2021), e1391.

[24] Filipa Correia et al. “A social robot as a card game player”. In: *Conference: Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*. 2017.

[25] Filipa Correia et al. ““I Choose... YOU!” Membership preferences in human–robot teams”. In: *Autonomous Robots* 43.2 (2019), pp. 359–373.

[26] Gergely Csibra et al. “The teleological origins of mentalistic action explanations: A developmental hypothesis”. In: *Developmental science* 1.2 (1998), pp. 255–259.

[27] Bruno Da Silva et al. “Learning parameterized skills”. In: *International Conference on Machine Learning (ICML)* (2012).

[28] Finale Doshi-Velez et al. “Towards a rigorous science of interpretable machine learning”. In: *arXiv preprint arXiv:1702.08608* (2017).

[29] Anca Dragan et al. “Generating legible motion”. In: *Robotics: Science and Systems (RSS)* (2013).

[30] Anca D Dragan et al. “Formalizing assistive teleoperation”. In: *Robotics: Science and Systems (RSS)* (2012).

[31] Anca D Dragan et al. “Legibility and predictability of robot motion”. In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2013, pp. 301–308.

[32] Anca D Dragan et al. “Effects of robot motion on human-robot collaboration”. In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2015, pp. 51–58.

[33] Jon Driver IV et al. “Gaze perception triggers reflexive visuospatial orienting”. In: *Visual cognition* 6.5 (1999), pp. 509–540.

[34] Paul Ekman et al. “The repertoire of nonverbal behavior: Categories, origins, usage, and coding”. In: *Nonverbal Communication, Interaction, and Gesture* 1.1 (1969), pp. 49–98.

[35] Miguel Faria et al. “Follow me: Communicating intentions with a spherical robot”. In: *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2016, pp. 664–669.

[36] Miguel Faria et al. “Understanding Robots: Making Robots More Legible in Multi-Party Interactions”. In: *2021 30th IEEE International Conference on Robot Human Interactive Communication (RO-MAN)*. 2021, pp. 1031–1036. DOI: 10.1109/RO-MAN50785.2021.9515485.

[37] Miguel Faria et al. “" Guess what I’m doing": Extending legibility to sequential decision tasks”. In: *arXiv preprint arXiv:2209.09141* (2022).

[38] György Gergely et al. “Taking the intentional stance at 12 months of age”. In: *Cognition* 56.2 (1995), pp. 165–193.

[39] Michael J Gielniak et al. “Generating human-like motion for robots”. In: *International Journal of Robotics Research* 32.11 (2013), pp. 1275–1301.

[40] Naomi Gildert et al. “The need for combining implicit and explicit communication in cooperative robotic systems”. In: *Frontiers in Robotics and AI* 5 (2018), p. 65.

[41] Piotr J Gmytrasiewicz et al. “A framework for sequential planning in multi-agent settings”. In: *Journal of Artificial Intelligence Research* 24 (2005), pp. 49–79.

[42] Michael A Goodrich et al. “Seven principles of efficient human robot interaction”. In: *IEEE International Conference on Systems, Man and Cybernetics (SMC)*. Vol. 4. IEEE. 2003, pp. 3942–3948.

[43] David Gunning et al. “XAI—Explainable artificial intelligence”. In: *Science robotics* 4.37 (2019), eaay7120.

[44] Soheil Habibian et al. “Encouraging Human Interaction With Robot Teams: Legible and Fair Subtask Allocations”. In: *IEEE Robotics and Automation Letters* 7.3 (2022), pp. 6685–6692. DOI: 10.1109/LRA.2022.3174264.

[45] Edward Twitchell Hall. *Beyond culture*. Anchor, 1989.

[46] Jaap Ham et al. “Making robots persuasive: the influence of combining persuasive strategies (gazing and gestures) by a storytelling robot on its persuasive power”. In: *International Conference on Social Robotics*. Springer. 2011, pp. 71–83.

[47] Bradley Hayes et al. “Challenges in shared-environment human-robot collaboration”. In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)* 8.9 (2013).

[48] Claire AGJ Huijnen et al. “Mapping robots to therapy and educational objectives for children with autism spectrum disorder”. In: *Journal of autism and developmental disorders* 46.6 (2016), pp. 2100–2114.

[49] A. Ijspeert et al. “Dynamical movement primitives: Learning attractor models for motor behaviors”. In: *Neural Computation* 25.2 (2013), pp. 328–373.

[50] Auke J Ijspeert et al. “Learning attractor landscapes for learning motor primitives”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2003, pp. 1547–1554.

[51] Akira Ito et al. “Why robots need body for mind communication-an attempt of eye-contact between human and robot”. In: *IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2004, pp. 473–478.

[52] David O Johnson et al. “Socially assistive robots: a comprehensive approach to extending independent living”. In: *International journal of social robotics* 6.2 (2014), pp. 195–211.

[53] Leslie Pack Kaelbling et al. “Planning and acting in partially observable stochastic domains”. In: *Artificial intelligence* 101.1-2 (1998), pp. 99–134.

[54] Kirsten E Kaplan et al. “Toward human-robot collaboration in surgery: performance assessment of human and robotic agents in an inclusion segmentation task”. In: *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2016, pp. 723–729.

[55] Stephen Kaplan. “A model of person-environment compatibility”. In: *Environment and behavior* 15.3 (1983), pp. 311–332.

[56] Ravi Kiran Kattoju et al. “Technological evaluation of gesture and speech interfaces for enabling dismounted soldier-robot dialogue”. In: *Unmanned Systems Technology XVIII*. Vol. 9837. International Society for Optics and Photonics. 2016, 98370N.

[57] Guy Keren et al. “Kindergarten assistive robotics (KAR) as a tool for spatial cognition development in pre-school education”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2012, pp. 1084–1089.

[58] Jens Kober et al. “Policy search for motor primitives in robotics”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2009, pp. 849–856.

[59] Jens Kober et al. “Movement templates for learning of hitting and batting”. In: *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2010, pp. 853–858.

[60] Levente Kocsis et al. “Bandit based monte-carlo planning”. In: *European conference on machine learning*. Springer. 2006, pp. 282–293.

[61] Ali H Kordia et al. “Movement recognition and prediction using DMPs”. In: *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 8544–8550.

[62] Petar Kormushev et al. “Robot motor skill coordination with EM-based reinforcement learning”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2010, pp. 3232–3237.

[63] Anagha Kulkarni et al. “A unified framework for planning in adversarial and cooperative environments”. In: *Proc. of the AAAI Conf. on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 2479–2487.

[64] Stephanie Lackey et al. “Defining next-generation multi-modal communication in human robot interaction”. In: *Proceedings of the Human Factors and Ergonomics Society*. Vol. 55. 1. SAGE Publications Sage CA: Los Angeles, CA. 2011, pp. 461–464.

[65] Yoon Kyung Lee et al. “Gentlemen on the Road: Effect of Yielding Behavior of Autonomous Vehicle on Pedestrian Head Orientation”. In: *arXiv* (2020).

[66] Benjamin Lettl et al. “Self-explanation capability for cognitive agents on-board of UCAVs to improve cooperation in a manned-unmanned fighter team”. In: *AIAA Infotech@ Aerospace (I@ A) Conference*. 2013, p. 4898.

[67] Songpo Li et al. “Implicit Intention Communication in Human-Robot Interaction Through Visual Behavior Studies”. In: *IEEE Transactions on Human-Machine Systems* 47.4 (2017), pp. 437–448. ISSN: 21682291. DOI: 10.1109/THMS.2017.2647882.

[68] Michael L Littman. “Value-function reinforcement learning in Markov games”. In: *Cognitive systems research* 2.1 (2001), pp. 55–66.

[69] M. Lopes et al. “Active learning for reward estimation in inverse reinforcement learning”. In: *Proc. Eur. Conf. Machine Learning and Knowledge Discovery in Databases*. 2009, pp. 31–46.

[70] Aleck M MacNally et al. “Action Selection for Transparent Planning.” In: *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. 2018, pp. 1327–1335.

[71] Francisco S Melo et al. “Project INSIDE: towards autonomous semi-unstructured human–robot social interaction in autism therapy”. In: *Artificial intelligence in medicine* 96 (2019), pp. 198–216.

[72] George Michalos et al. “Design considerations for safe human-robot collaborative workplaces”. In: *Procedia CIrP* 37 (2015), pp. 248–253.

[73] Olivier Michel. “Cyberbotics Ltd. Webots™: professional mobile robot simulation”. In: *International Journal of Advanced Robotic Systems* 1.1 (2004), p. 5.

[74] Shuwa Miura et al. “A unifying framework for observer-aware planning and its complexity”. In: *Uncertainty in Artificial Intelligence*. PMLR. 2021, pp. 610–620.

[75] Shuwa Miura et al. “Maximizing legibility in stochastic environments”. In: *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*. IEEE. 2021, pp. 1053–1059.

[76] Roger Ruiz Moral et al. “Effectiveness of a Communication Skills training program for medical students to identify patients communicative clues”. In: *Patient Education and Counseling* (2020).

[77] Bilge Mutlu et al. “Coordination mechanisms in human-robot collaboration”. In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. Citeseer. 2013, pp. 1–6.

[78] John F Nash Jr. “Equilibrium points in n-person games”. In: *Proceedings of the national academy of sciences* 36.1 (1950), pp. 48–49.

[79] Stefanos Nikolaidis et al. “Viewpoint-Based Legibility Optimization”. In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2016, pp. 271–278.

[80] Christos H Papadimitriou et al. “The complexity of Markov decision processes”. In: *Mathematics of operations research* 12.3 (1987), pp. 441–450.

[81] Georgios Papoudakis et al. “Benchmarking Multi-Agent Deep Reinforcement Learning Algorithms in Cooperative Tasks”. In: *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS)*. 2021.

[82] Raul Paradeda et al. “The Role of Assertiveness in a Storytelling Game with Persuasive Robotic Non-Player Characters”. In: *Annual Symposium on Computer-Human Interaction in Play*. 2019, pp. 453–465.

[83] Alexandros Paraschos et al. “Probabilistic movement primitives”. In: *Advances in Neural Information Processing Systems (NIPS)* (2013), pp. 2616–2624.

[84] Karl Popper. “The myth of the framework”. In: *Rational changes in science*. Springer, 1976, pp. 35–62.

[85] Giacomo Rizzolatti et al. “The mirror-neuron system”. In: *Annual Reviews of Neuroscience* 27 (2004), pp. 169–192.

[86] Isabel Galhano Rodrigues. “Fala e movimentos do corpo na interacção face a face: uma proposta de análise de meios de contextualização e estruturação de sequências narrativas”. In: *Revista da Faculdade de Letras-Línguas e Literaturas* 22 (2020).

[87] Tatsuya Sakai et al. “Explainable autonomous robots: a survey and perspective”. In: *Advanced Robotics* 36.5-6 (2022), pp. 219–238.

[88] Maha Salem et al. “Would You Trust a (Faulty) Robot?: Effects of Error, Task Type and Personality on Human-Robot Cooperation and Trust”. In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. 2015, pp. 141–148. ISBN: 9781450328838. DOI: 10.1145/2696454.2696497. URL: http://dl.acm.org/citation.cfm?doid=2696454.2696497.

[89] Wojciech Samek et al. “Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications”. In: *Proceedings of the IEEE* 109.3 (2021), pp. 247–278. DOI: 10.1109/JPROC.2021.3060483.

[90] Shane Saunderson et al. “How robots influence humans: A survey of nonverbal communication in social human–robot interaction”. In: *International Journal of Social Robotics* 11.4 (2019), pp. 575–608.

[91] Allison Sauppé et al. “Design patterns for exploring and prototyping human-robot interactions”. In: *SIGCHI Conference on Human Factors in Computing Systems*. 2014, pp. 1439–1448.

[92] Sarath Sreedharan et al. “A Unifying Bayesian Formulation of Measures of Interpretability in Human-AI Interaction.” In: *Proc. Int. Joint Conference on Artificial Intelligence (IJCAI)*. 2021, pp. 4602–4610.

[93] Leila Takayama et al. “Expressing thought: improving robot readability with animation principles”. In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. 2011, pp. 69–76.

[94] Lin Wang et al. “When in Rome: the role of culture & context in adherence to robot recommendations”. In: *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2010, pp. 359–366.

[95] Richard Warner et al. “Making Artificial Intelligence Transparent: Fairness and the Problem of Proxy Variables”. In: *Criminal Justice Ethics* 40.1 (2021), pp. 23–39.

[96] Daniel E Whitney. “Resolved motion rate control of manipulators and human prostheses”. In: *IEEE Transactions on man-machine systems* 10.2 (1969), pp. 47–53.

[97] K Woerner et al. “Legibility and predictability of protocol-constrained motion: Evaluating human-robot ship interactions under COLREGS collision avoidance requirements”. In: *Robotics Science and Systems*. 2017.

[98] Elmira Yadollahi et al. “When deictic gestures in a robot can harm child-robot collaboration”. In: *ACM Conference on Interaction Design and Children*. 2018, pp. 195–206.

[99] Huaijiang Zhu et al. “Legible action selection in human-robot collaboration”. In: *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2017, pp. 354–359.

[100] Xiaojin Zhu. “Machine teaching: An inverse problem to machine learning and an approach toward optimal education”. In: *Proc. of the AAAI Conference on Artificial Intelligence*. Vol. 29. 1. 2015.