Recognizing Collaboration Intent to Control Physical Human-Robot Interaction

BY

STEFANO CASTAGNERI
B.S., Politecnico di Torino, Turin, Italy, 2016

THESIS

Submitted as partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Chicago, 2018

Chicago, Illinois

Defense Committee:

Miloš Žefran, Chair and Advisor
Barbara Di Eugenio, Computer Science
Marina Indri, Politecnico di Torino

To my parents,

Duilio and Antonella,

they always believed in me.

ACKNOWLEDGMENTS

I would like to thank my UIC adviser, professor Miloš Žefran, who always helped and guided me in this research. I would like to thank my Politecnico adviser, prof. Marina Indri, for the support during these months. I appreciate the questions asked by the committee during the defense, in particular prof. Barbara Di Eugenio. They helped me in better explaining the contents of the research.

I would like to thanks all my Italian friends and the TOP-UIC group, they supported and helped me during these months.

I am also thankful to Sanket, Bahareh, Zhanibek and all my colleagues in the Robotics Lab for helping me with ROS and Baxter Robot and always made me feel comfortable in the lab.

Last, but not least, a special thanks to my family for the great opportunity they gave me and for always supporting me.

SC

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

CHAPTER **PAGE**

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (Continued)

FIGURE **PAGE**

LIST OF FIGURES (Continued)

FIGURE **PAGE**

LIST OF ABBREVIATIONS

pHHI	Physical Human-Human Interaction
pHRI	Physical Human-Robot Interaction
ADLs	Activities of Daily Living
UIC	University of Illinois at Chicago
VL	Virtual Linkage
ME	Minimum Energy
PM	Polynomial Model
PM-ON	On-line Polynomial model
PM-OFF	Off-line Polynomial Model
SPB	Single Person Bi-manual
SYNC	Synchronized
L/F	Leader/Follower
F/L	Follower/Leader
COOP	Cooperation (scenario)
ROS	Robotic Operating System
API	Application Programming Interface
JCB_s	Joint Control Boards

LIST OF ABBREVIATIONS (Continued)

PID	Proportional Integral Derivative
IK	Inverse Kinematics

SUMMARY

In recent years, there has been intense interest in collaborative robots, both for industry and household applications. While significant progress has been made, physical human-robot interaction is still presenting a challenging problem that has not been satisfactorily solved. When a human is interacting with another human, the forces they exchange represent a communication channel and a continuous stream of information flows between them. When a human is interacting with a robot, the forces applied by the robot are interpreted by the human that in turn reacts to them; obviously, people are expecting the robot to also react to the forces they are applying. In this research, we identify different types of collaboration during collaborative manipulation and use this information to better control human-robot interaction. We propose a new metric for the identification of the cooperation intent and study how to best compute the interaction force, on which our metric is based, in a real time application. We also propose a control framework that uses a set of robot controllers that are selected using the identified collaboration intent to control the robot during collaborative tasks. Finally, we present our preliminary experiments with the Baxter robot. The experiments have been performed in order to understand the precision, repeatability and safety of the robot using different control approaches. These experiments informed the proposed controllers and are the key for their future implementation of the Baxter robot.

CHAPTER 1

INTRODUCTION

Many applications require humans to physically interact with robots in order to perform a variety of tasks. Examples are rehabilitation, robotic assistance for the elderly or disabled, entertainment, education and assistance with manipulation in warehouses or with assembly in factories. In all these applications, the quality of the cooperation is what researchers are trying to maximize in order to make the collaboration feel natural to humans. Clearly, different applications require different cooperation strategies.

1.1 Motivation and Research Questions

The focus of this research is on robotic caregivers for the elderly. When an elderly person needs help with activities of daily living (ADLs), the help needs to be provided by a caregiver, either in the home or institutional setting. But there is an acute lack of trained caregivers, and living at home has far better outcomes. Elderly assistive robots are thus a promising alternative that could both reduce cost, increase access, and provide better options for the elderly. However, in order to prevent both physical and mental deterioration [1] [2], the elderly needs to be actively engaged in everyday tasks. The robot thus needs to work with rather than in place of, the human.

Understanding what constitutes successful collaboration in physical Human-Human Interaction (pHHI) and in physical Human-Robot Interaction (pHRI) is the key to making the

interaction feel natural. In particular, it is necessary to characterize what makes the collaboration cooperative. Once a measure of the cooperation is available, it is possible to regulate the contribution of the robot to the task in order to increase the quality of the collaboration. The cooperation quality is related to the forces applied during the collaboration; in particular, the interaction forces represent a communication channel during a collaboration. Different approaches have been proposed to compute the interaction forces [3] [4] [5]. However, not all of them can be used to effectively identify different cooperation scenarios.

Once a measure of the cooperation is available, it is possible to predict the human's intent and control the robot in order to match this intent. This leads to the following list of questions that guided our research:

(i) How to define the cooperation in a collaborative task?

(ii) How to compute the interaction forces? Which model is meaningful, which is not?

(iii) How to link the cooperation scenario to the human's collaboration intent?

(iv) How to control a manipulator in order to make the collaboration natural and adaptable to human intent?

1.2 Related Work

Human motion has been studied in detail by many researchers and different models exist to describe the average trajectory followed by humans in single arm reaching movements. A variety of models are based on optimal control, in which the key idea is the minimization of a specific cost function while performing the task. The cost functions adopted to study

the average behavior involve energy, smoothness, accuracy and other multi-attribute costs [6]. Flash and Hogan studied a minimum-jerk trajectory model to describe the average motion of the human arm [7]. In particular, this method generates a bell-shaped velocity profile and it is what we have adopted in our research. Uno, Kawato and Suzuki developed a model that minimizes the torque change during the motion in order to study the average trajectory [8].

For what concerns bi-manual and dyadic motions, the minimum jerk model has been adapted in order to also take into account the rotational jerk [9]. It has also been shown that the trajectory of an object in bi-manual or dyadic motions is highly correlated with the minimum-jerk trajectory [10].

Many researchers also studied the forces that are applied by humans during a collaborative task. In particular, the interaction force, the portion of the force that is not responsible for the motion of the object, represents a communication channel between humans during a collaboration. Different models have been proposed to compute the interaction forces. One of them is based on the assumption that the interaction forces follow the mechanical internal forces [3]: we call this the virtual-linkage (VL) model. The minimum-energy (ME) model instead assumes that the energy is minimized during the collaboration [4]. Noohi, Žefran and Patton proposed a polynomial model (PM) to compute the interaction force [5]. In particular, this method computes the interaction force exploiting the trajectory of the force vectors. These three methods will be explained in details in chapter 2 since they have been used extensively for this research.

Human intent has been also studied in detail and its crucial role in HRI has been analyzed in many researches. Nehaniv *et al* [11] proposed a classification of human gestures into five main

classes. They also showed the importance of taking both the kinematics of the gesture and the interactional context into consideration. Kulic and Croft [12] proposed a method for predicting human intent based on physiological signals (blood volume pressure, skin conductance, chest cavity expansion/contraction, corrugator muscle activity) and used this information to improve the safety of HRI. Breazeal and Aryananda [13] studied how to recognize human intent in robot-directed speech without knowing the linguistic content. They presented a method to identify four patterns that communicate praise, prohibition, attention and comfort and how to use this information to instruct robots and facilitate the learning process in HRI. A different approach, based on stacked denoising auto-encoders is presented in [14]. This research studied human intent with respect to inanimate objects exploiting the environment information obtained by a Kinect camera and using deep learning techniques. Kelley *et al* [15] proposed a similar approach that models the robot's acquired experience using Hidden Markov Models.

While recognizing human intent has been studied extensively and many different approaches have been proposed, little work exists on pHRI. Erden and Tomiyama [16] developed a physically interactive control scheme for a robotic manipulator. Human intent, the desired position, is estimated observing the human applied forces at the end effector. A switching scheme is also proposed to alternate two different controllers. Kazerooni [17] studied pHRI when the capability of the human to perform the task is limited by the physical strength and then an extender (a robot used to increase human mechanical strength) is used. He developed general models for the human, the extender and the interaction, analyzing stability and performances. Duchaine and Gosselin [18] proposed a control scheme to increase the level of safety in pHRI guaranteeing

that the robot will never be unstable. They also proposed a new variable admittance control that provides an intuitively human interaction. Mielke *et al* [19] studied the forces and torques data in leader-follower human-human dyadic tasks. They showed how the interaction force is necessary for a co-manipulation and how lateral movements follow a minimum-jerk trajectory. They also proposed different metrics to identify good lateral movements but they did not show how to use this information to control pHRI.

In this research, we propose a new metric to identify cooperation intent and a model that exploits this information to improve pHRI. Some of the existing models detect human intent using physical quantities that are not easily available when performing ADLs. Other methods require information from the entire task and can not be implemented in case of real-time applications. In contrast, our approach detects human intent exploiting easily measurable physical quantities that can be obtained in real-time. Our model is able to detect multiple collaboration scenarios and presents an intuitive description of the interaction. Moreover, it is able to recognize sudden changes in the collaboration scenario. The metric we propose requires the computation of the interaction force and the robot must have complete knowledge of the task. The accuracy of the method is strictly related to the accuracy of the computed interaction force and the approximation of the motion trajectory.

1.3 Thesis Organization

This thesis is organized in five chapters.

Chapter 1: This chapter provides an introduction to the research and states the research questions. It also presents results of other research related to the same topic.

Chapter 2: This chapter provides the background on various topics used for the research. In particular, it discusses human motion trajectory, forces during human motion, interaction force models and clustering methods.

Chapter 3: This chapter presents the cooperation study. We identify different cooperation scenarios using different clustering methods. Finally, we compare different models for the computation of the interaction forces and investigate their performance.

Chapter 4: In this chapter, the cooperation strategy is presented to control the manipulator during the collaborative manipulation. The idea of the switching controllers is explained. Details on the manipulator and the software are also provided.

Chapter 5: This chapter discusses the results and provides the conclusions. The chapter also presents possible extensions of the work.

CHAPTER 2

BACKGROUND

This chapter provides the background material that will be used in the rest of the thesis. In the first section (2.1) the minimum-jerk motion model is introduced and the solution in case of zero boundary conditions is reported. The human forces in reaching movements are analyzed in section 2.2. Section 2.3 introduces the concept of interaction force and three different methods to compute it. Finally, in section 2.4, an overview of different clustering methods that will be used in the analysis of data collected during human studies is provided.

2.1 Human Motion Trajectory in Reaching Movement

Human motion has been studied extensively. It has been observed that human reaching motion between two points, A and B, follows a straight line with a distinct velocity profile. It has been shown that the motion follows a minimum-jerk trajectory, generating the bell-shaped velocity profile [7]. In particular, the following cost function is minimized:

$$L(t_i, t_f, x) = \frac{1}{2} \int_{t_i}^{t_f} \left|\left| \frac{d^3 x(t)}{dt^3} \right|\right| dt \tag{2.1}$$

$$x^*(t) = \arg\min_{x(t)} (L(t_i, t_f, x)) \tag{2.2}$$

where $x(t) \in \mathbb{R}^3$ represents the motion trajectory, while t_i and t_f are the initial and final time respectively. The cost function is the integral of the norm of the third derivative of position,

which is called jerk; this is why the optimal solution (Equation 2.2) is called minimum-jerk trajectory. The optimization problem can be easily solved if the zero boundary conditions are assumed (the initial and final velocity are equal to zero). In this case, the trajectory can be rewritten as follow:

$$x^*(\tau) = x(t_i) + D(6\tau^5 - 15\tau^4 + 10\tau^3) \tag{2.3}$$

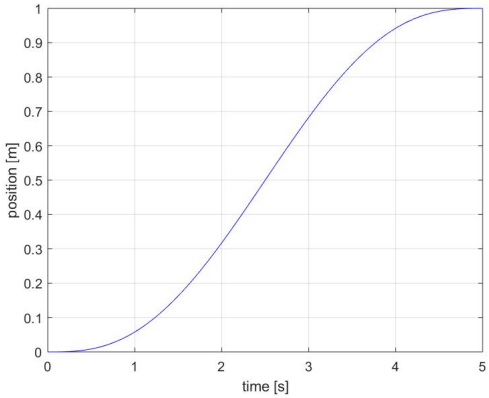
$$\tau = \frac{t - t_i}{t_f - t_i} \tag{2.4}$$

$$D = x(t_f) - x(t_i) \tag{2.5}$$

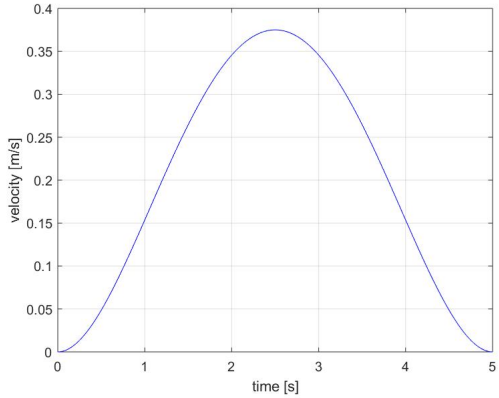
Figure 1 shows the x-coordinate of position, velocity and acceleration of a minimum-jerk trajectory with $x(t_i)$ and $x(t_f)$ equal to $(0, 0, 0)$ and $(1, 0, 0)$, while t_i and t_f equal to 0 and 5 respectively.

2.2 Forces during Human Reaching Movement

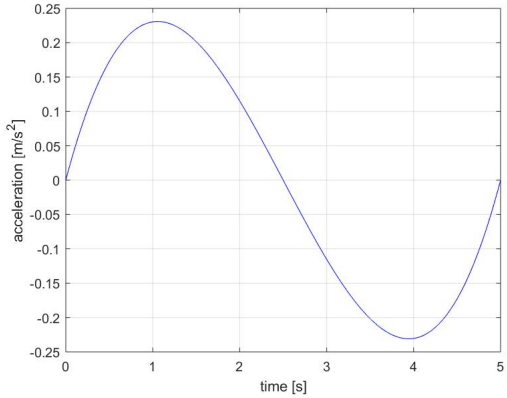
Much research has showed that when an external disturbance force is applied during a human reaching movement, the human can adapt and compensate the disturbance. In other words, humans are able to compensate the external force applying the same amount of force in the opposite direction, achieving again the optimal trajectory x^*, after a sufficient number of learning trials. In particular, it has been shown [5] that the applied forces follow a minimum-jerk trajectory and can be written as



(a) position



(b) velocity



(c) acceleration

Figure 1: Minimum-jerk trajectory

$$F^*(t) = \arg\min_{F} \left(\frac{1}{2} \int_{t_i}^{t_f} \left|\left| \frac{d^3 F(t)}{dt^3} \right|\right|^2 dt \right) \tag{2.6}$$

Shadmehr and Wise showed [20] that using the calculus of variations, the force can be rewritten as a fifth-order polynomial (Equation 2.7), since the sixth derivative of $F^*(t)$ is equal to zero.

$$F^*(t) = \sum_{k=0}^{5} c_k t^k \tag{2.7}$$

Finally, if the zero boundary conditions are respected, the initial and final forces will be zero too. If instead other boundary conditions have to be satisfied, the calculus of variations can be used to find the new solution (c_k coefficients).

2.3 Interaction Force

The concept of interaction force is crucial for collaborative manipulation. In a dyadic cooperation (see Figure 2), the sum (F_{sum}) of the two applied forces (f_1 and f_2) is responsible for the motion of the manipulated object. The applied forces could be decomposed into the effective forces (f_1^* and f_2^*) and the interaction force (F^i):

$$f_1 = f_1^* + F^i \tag{2.8}$$

$$f_2 = f_2^* - F^i \tag{2.9}$$

As discussed in Chapter 1, the interaction force represents a communication channel between humans during a dyadic task. Since the interaction forces of the two hands cancel each other, they do not contribute to the object's motion:

$$F_{sum} = f_1 + f_2 = f_1^* + f_2^* \tag{2.10}$$

It is possible to introduce a new parameter α standing for the contribution of each person in performing the task. The effective forces and the interaction force could be rewritten as function of α as follow:

$$f_1^* = \alpha F_{sum} \tag{2.11}$$

$$f_2^* = (1-\alpha) F_{sum} \tag{2.12}$$

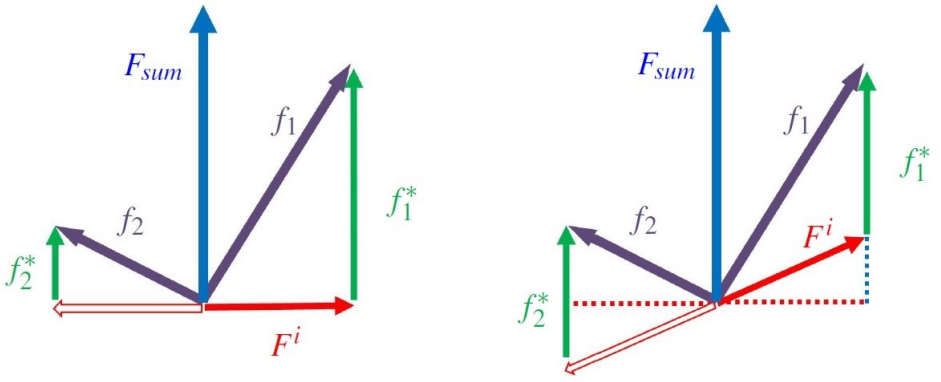
$$F^i = f_1 - f_1^* = (1-\alpha) f_1 - \alpha f_2 \tag{2.13}$$

These equations constitute an under-determined system and are satisfied for any value of α. Only if F_{sum} is equal to zero a unique solution exists, with $F^i = f_1 = -f_2$ and $f_1^* = f_2^* = 0$. Figure 2 shows two possible decompositions of the applied forces, one where the interaction force is orthogonal to F_{sum}, and one where the interaction force has both orthogonal and parallel components.

Researchers proposed models to uniquely determine the interaction force. In the following sections, three of these methods are presented in more detail since they will be used in this research.

2.3.1 Virtual Linkage Model

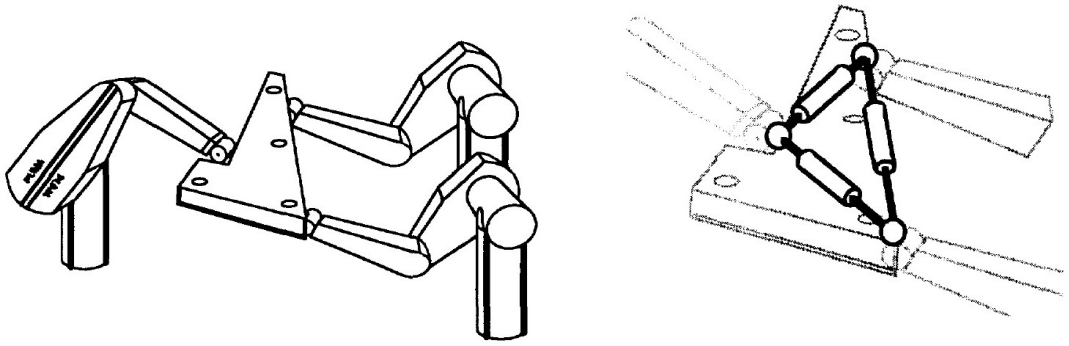
The virtual linkage (VL) model was proposed by Khatib and Williams [3]. They proposed a model to compute the internal forces during a multi-grasp manipulation. In particular, the concept of *virtual linkage* is introduced: "A virtual linkage associated with an n-grasp



(a) orthogonal components only (b) orthogonal and parallel components

Figure 2: Two possible decompositions of the applied forces into effective forces and interaction force. Reprinted from "A Model for Human-Human Collaborative Object Manipulation and Its Application to Human-Robot Interaction," by E. Noohi, M. Žefran and J. L. Patton, 2016, IEEE Transactions on Robotics, 32, 4. Copyright 2016 by IEEE.

manipulation task is a $6(n-1)$-degree-of-freedom mechanism whose actuated joints characterize the object's internal forces and moments" (Khatib & Williams, 1993, p. 1025). The forces and the moments, that are applied to the virtual linkage at the grasp points, are transmitted to the virtual linkage's actuators. The virtual linkage system will be in equilibrium when the same amount of forces and torques, but in opposite direction, will be applied. These forces and torques correspond to the internal forces and torques [3]. Figure 3 shows an application of the virtual linkage model to a three-arm manipulation task. The grasp points are represented as spheres while the links connecting the grasp points represent the actuators of the virtual linkage.



(a) (b)

Figure 3: Virtual linkage model: a) a three-arm manipulation task; b) a virtual linkage model corresponding to three-arm manipulation task. Reprinted from "The Virtual Linkage Model: A Model for Internal Forces in Multi-Grasp Manipulation" by D. William and O. Khatib, 1993, IEEE International Conference on Robotics and Automation, 1026. Copyright 1993 by IEEE.

In case of a two-grasp manipulation, there is only one component of the internal force and it lies on the line connecting the two grasp points [3]. The internal force will be given by the difference of the applied forces at the grasp points. In case of a dyadic cooperation, the effective forces and the interaction force become:

$$f_1^* = f_2^* = \frac{1}{2} F_{sum} \tag{2.14}$$

$$F^i = \frac{1}{2}(f_1 - f_2) \tag{2.15}$$

which is equivalent to setting α equal to 0.5.

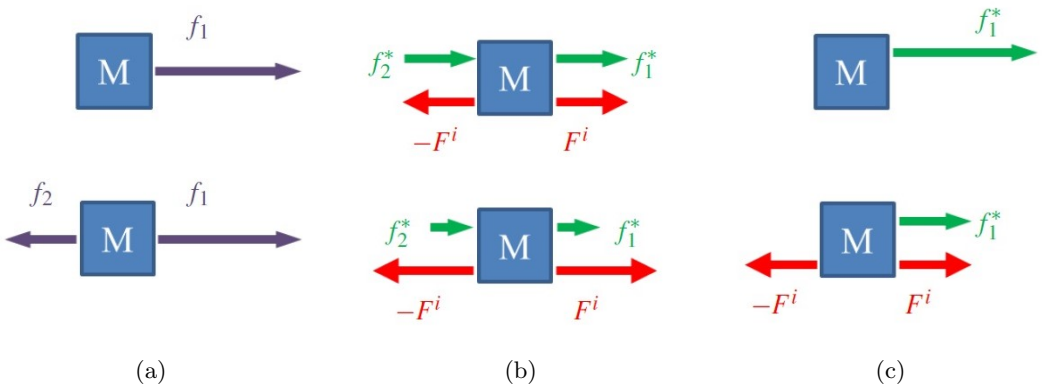
2.3.2 Minimum-Energy Model

The minimum energy model is motivated by the study on dominance measures and distribution in haptic interaction conducted by Groten *et al* [4].They investigated the dominance distribution between partners and implicitly developed the minimum-energy model to compute the interaction force. The key idea is that the interaction force is present only if the two partners apply forces in opposite directions, both pulling or pushing the object. Again, the interaction forces are not responsible for the motion of the object and can be determined as follows:

$$F_1^i = \begin{cases} 0, & \text{if } sgn(f_1) = sgn(f_2) \\ f_1, & \text{if } sgn(f_1) \neq sgn(f_2) \wedge |f_1| \leq |f_2| \\ -f_2, & \text{if } sgn(f_1) \neq sgn(f_2) \wedge |f_1| > |f_2| \end{cases} \tag{2.16}$$

The interaction force for the second partner would be $F_2^i = -F_1^i$. Using the relations between the applied forces (f_1 and f_2) and the interaction forces (F_1^i and F_2^i) it is possible to compute the effective forces (f_1^* and f_2^*).

Figure 4 shows the computation of interaction forces in a dyadic task applying both VL model and ME model. Note that in the case of the VL model, the two effective forces have always the same magnitude, independently on the number of applied forces. In the case of the ME model instead, when only one force is applied, the interaction force is zero and the applied force equals the effective force.



(a) (b) (c)

Figure 4: Interaction force models comparison: a) applied forces; b) virtual-linkage model; c) minimum-energy model. Reprinted from "A Model for Human-Human Collaborative Object Manipulation and Its Application to Human-Robot Interaction," by E. Noohi, M. Žefran and J. L. Patton, 2016, IEEE Transactions on Robotics, 32, 5. Copyright 2016 by IEEE.

2.3.3 Polynomial Model

Noohi and Žefran proposed a polynomial model for the computation of the interaction force [5]. They first computed the interaction force in a mass-spring system and then generalized the analysis to a rigid body. They showed how in a dyadic task the interaction force follows a minimum-jerk trajectory and can be computed as follows:

$$F^i(t) = \arg\min_F \left(\frac{1}{2} \int_{t_i}^{t_f} \left|\left| \frac{d^3F(t)}{dt^3} \right|\right|^2 dt \right) \tag{2.17}$$

or equivalently

$$F^i(t) = \sum_{k=0}^{5} c_k t^k \tag{2.18}$$

The coefficients c_k can be computed introducing the following five constraints [5]:

$$F^i(t_i) = f_1(t_i) \tag{2.19}$$

$$F^i(t_m) = f_1(t_m) \tag{2.20}$$

$$F^i(t_f) = f_1(t_f) \tag{2.21}$$

$$\dot{F}^i(t_i) = \dot{f}_1(t_i) \tag{2.22}$$

$$\dot{F}^i(t_f) = \dot{f}_1(t_f) \tag{2.23}$$

The five constraints introduce five coefficients, the last one is found solving the optimization problem (Equation 2.17).

The polynomial model assumes that the task is well known and it is based on the motion model: the precision of the polynomial model is strictly related to the precision of the motion model [5]. One important advantage with respect to VL and ME models is that PM can compute the interaction force based only on one of the two applied forces (f_1 or f_2). Obviously, this will turn out to be important in case of pHRI since only the human applied force is directly available. On the other hand, in order to compute the interaction force at the current time, PM requires the value of one of the two applied forces in the future: this model is not directly

applicable for the real-time control of the manipulator. For this reason we will refer to this model as off-line polynomial model (PM-OFF).

An extension of the PM-OFF has been proposed [5] to compute the interaction force in real-time. We will refer to this model as on-line polynomial model (PM-ON). PM-OFF requires to know the value of the applied force at t_i, t_m and t_f. PM-ON needs to predict the value of the applied force in the future, in order to compute the actual value of the interaction force. The suggested prediction for the human applied force ($\tilde{F}_H$) is:

$$\tilde{F}_H(\tau, t) = \begin{cases} F_H(\tau), & \text{if } \tau \leq t \\ F_H(t), & \text{if } \tau > t \end{cases} \tag{2.24}$$

where τ is the time variable and t is the current time. This prediction is simply setting the human applied forces to remain constant in the future and equal to last available force sample. Figure 5 shows the interaction force computed using PM-OFF and PM-ON.

2.4 Overview of Used Clustering Methods

In this research, three clustering methods were primarily used to find clusters of different kinds of data. The following paragraphs explain the differences between these methods, presenting advantages and disadvantages of each.

K-means Algorithm

K-means is a clustering method that minimizes the total intra-cluster variance. It is based on the distance (it is possible to define the distance in a variety of different ways)

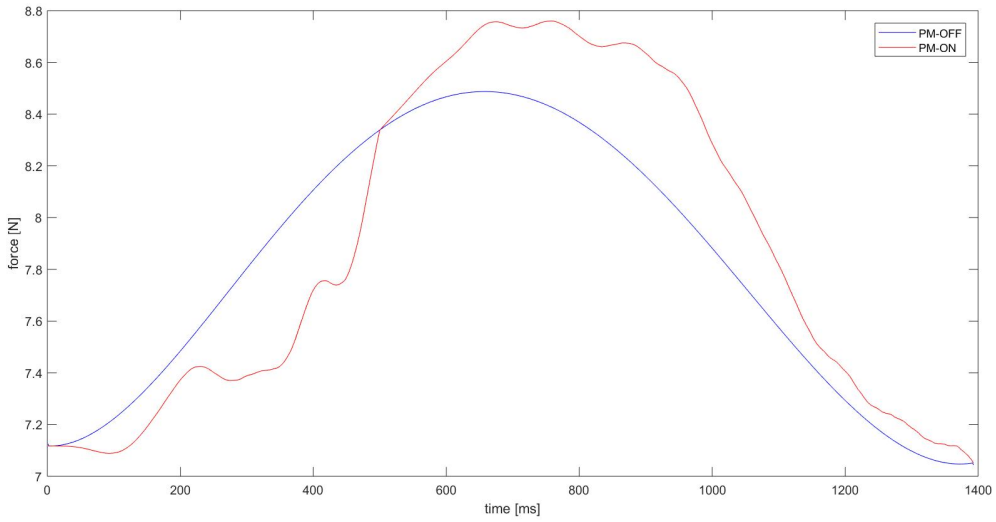


Figure 5: Interaction force computed using PM-OFF and PM-ON

between points and centroids (centers of the clusters) and it is an iterative procedure, until convergence. One of the main advantages of this method is the speed of convergence. On the other hand, the user must specify the number of clusters and the initial centroids (otherwise taken randomly or guessed). The algorithm works pretty well with spherical clusters, but completely fails with dataset of random shapes.

DBSCAN Algorithm

The Density Based Spatial Clustering of Application with Noise (DBSCAN) is an algorithm connecting points with a sufficiently high density. It does not require to specify in advance the number of clusters, but requires two other parameters: ϵ that stands for

the point's neighborhood and $MinPts$ that represents the minimum number of points to construct a cluster. The DBSCAN method is able to recognize clusters of various shapes and it can identify and distinguish noise from clusters. On the other hand, choosing parameters (ϵ and $MinPts$) is not easy and the algorithm fails to classify clusters with large density difference.

Agglomerative Clustering Algorithm

The agglomerative clustering method is a strategy belonging to the more general hierarchical clustering. In this method, the initial number of clusters is equal to the number of points; then pairs of clusters are grouped to construct bigger clusters. The decision whether to group two clusters into a new one is based on the distance between the two. The main advantage is that it is possible to use any measure of the distance. This method is always applicable, it does not require specifying the number of clusters in advance and it is possible to specify the maximum number of clusters ($MaxCl$).

CHAPTER 3

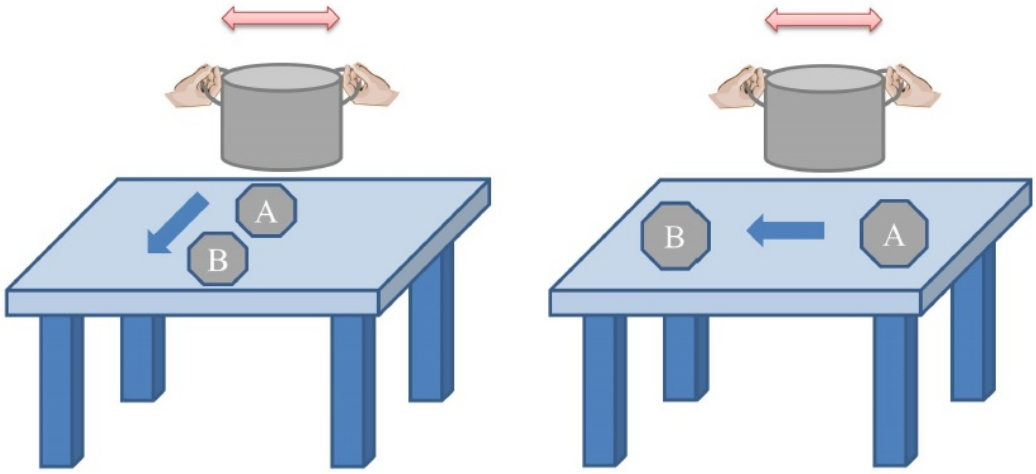
COOPERATION STUDY

In this chapter we describe the human study used to collect the data on which we base our research. The data collection experiment was designed and performed by Dr. Ehsan Noohi at the Robotics Lab at the University of Illinois at Chicago (UIC) [5]. The human study was conducted with 22 participants; they were asked to move an aluminum pot ($w < 22N$) from a point A to a point B horizontally. Two force sensors (SI-65-5 ATI Gamma) were placed in between the handles and the pot to collect the applied forces. The pot's orientation and acceleration were measured using a 9DOF-Sensor-Stick SparkFun IMU. Based on the direction of movement with respect to the line connecting the handles, two type of motions are specified (Figure 6):

Type 1 : perpendicular movement

Type 2 : parallel movement

The task was performed in different cooperation situations. In single-person-bimanual (SPB) mode, only one person performs the task holding the pot with both hands. In dyadic mode instead, two people perform the motion, each holding one handle of the pot. In particular in synchronized (SYNC) scenario, the two participants are asked to perform the task at the same time without assigning a cooperation role, while in a leader/follower (L/F) scenario, a role



(a) (b)

Figure 6: Type 1 (a) and type 2 (b) motions. Reprinted from "A Model for Human-Human Collaborative Object Manipulation and Its Application to Human-Robot Interaction," by E. Noohi, M. Žefran and J. L. Patton, 2016, IEEE Transactions on Robotics, 32, 8. Copyright 2016 by IEEE.

(leader or follower) is assigned to each participant: the leader directs the motion of the object while the follower simply follows the leader.

This chapter is organized as follow: in the first sections, it is shown how the cooperation is analyzed in [5] and why this approach is difficult to implement for real-time control. In the second part of the chapter, a new metric is proposed that allows us to characterize the type of cooperation in real-time. A comparison between virtual linkage (VL), minimum energy (ME), off-line polynomial model (PM-OFF) and on-line polynomial model (PM-ON) is also reported.

3.1 Cooperation metrics

Noohi and Žefran [5] proposed five quantitative metrics to identify different features of the cooperation. Each metric tries to characterize a different feature of a force signal, such as the energy or the variation rate. The five metrics are as follows:

Similarity index: this index measures how much the two effective forces differ at every time instant. It is defined as

$$I_s = \frac{1}{t_f - t_i} \int_{t_i}^{t_f} M_s(t) dt \tag{3.1}$$

where

$$M_s(t) = 1 - \left| \frac{||f_1^*(t)|| - ||f_2^*(t)||}{||F_{sum}(t)||} \right| \tag{3.2}$$

Efficiency index: it measures the disagreement between the two effective forces during the task. It is computed as

$$I_e = \frac{1}{t_f - t_i} \int_{t_i}^{t_f} M_e(t) dt \tag{3.3}$$

where

$$M_e(t) = \frac{||F_{sum}(t)||}{||f_1^*(t)|| + ||f_2^*(t)||} \tag{3.4}$$

Fairness index: this index measures the difference in energy of the effective forces. It is defined as follows:

$$I_f = 1 - \left| \frac{N_1 - N_2}{N_{sum}} \right| \tag{3.5}$$

where

$$\begin{cases} N_1 = \int_{t_i}^{t_f} ||f_1^*(t)||dt \\ N_2 = \int_{t_i}^{t_f} ||f_2^*(t)||dt \\ N_{sum} = \int_{t_i}^{t_f} ||F_{sum}(t)||dt \end{cases} \tag{3.6}$$

Cooperation index: it measures the average value of the cooperation during the task. A new parameter $\delta = \frac{1}{2} - \alpha$ is introduced and the effective forces are expressed as a function of δ:

$$f_1^*(t) = \left(\frac{1}{2} - \delta\right) F_{sum} \tag{3.7}$$

$$f_2^*(t) = \left(\frac{1}{2} + \delta\right) F_{sum} \tag{3.8}$$

Then, the cooperation index is defined as

$$I_c = 1 - \frac{I_\delta}{max(I_\delta)} \tag{3.9}$$

where

$$I_\delta = \frac{1}{t_f - t_i} \int_{t_i}^{t_f} \delta(t)dt \tag{3.10}$$

Comfort index: this index measures the difficulty of the task taking into account the average rate of change of the parameter δ. It is defined as

$$I_o = 1 - \frac{I_p}{max(I_p)} \tag{3.11}$$

where

$$I_p = \frac{1}{t_f - t_i} \int_{t_i}^{t_f} ||\dot{\delta}(t)||dt \tag{3.12}$$

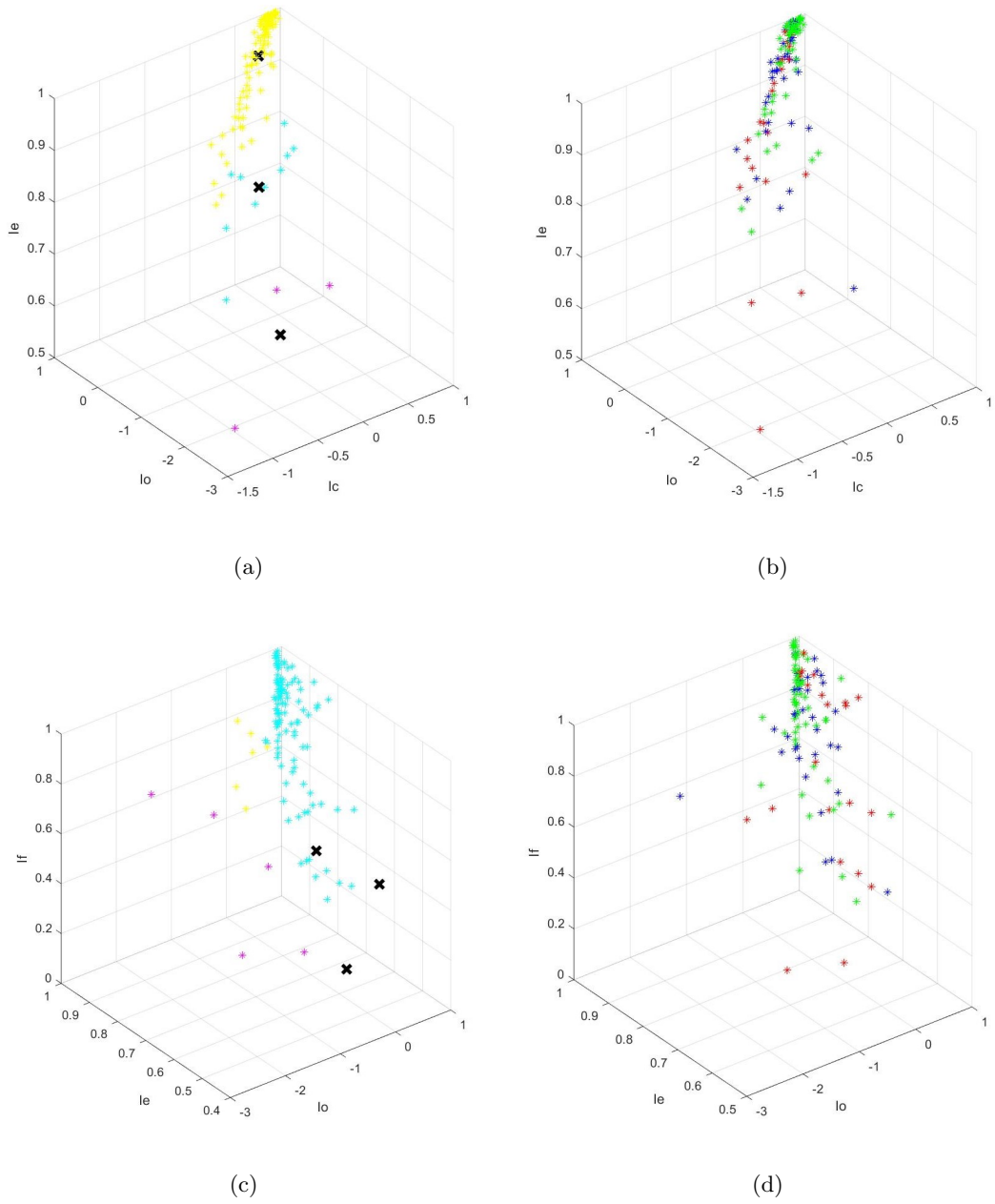
The five metrics presented above are all bounded between 0 and 1. A value of 1 represents an ideal (perfect) cooperation, while 0 stands for the worst case. These five indexes were computed plugging in the data from the human study and applying PM-OFF [5]. We tried to find clusters from the obtained metrics and understand if these indexes could be used to identify different cooperation situations in real-time. The results of our analysis are reported in the next section.

3.2 Clustering of Cooperation Metrics

Starting from the five metrics defined in [5], we tried to determine if the metrics computed from the experimental data cluster into well-defined classes. We thus performed clustering on the computed metrics. Different clustering methods were applied (section 2.4). Our goal was to find a clustering method that would be able to at least distinguish the three main cooperation situations investigated in the study: SPB, SYNC and L/F.

3.2.1 K-means Clustering

We implemented a k-means algorithm in MATLAB to cluster the human study data. Since the k-means algorithm requires the number of clusters as input, we ran the code by specifying 3, 5 and 7 clusters. Furthermore, different ways to compute the distance metric were implemented. In particular: Euclidean distance (Equation 3.13), cosine distance (Equation 3.14), correlation distance (Equation 3.15) and cityblock distance (Equation 3.16).



(a) (b)

(c) (d)

Figure 7: K-means algorithm results comparison: a) 3 centroids, euclidean distance metric, black crosses for centroids; b) dataset points: SPB in green, SYNC in blue, L/F in red; c) 3 centroids, cosine distance metric, black crosses for centroids; d) dataset points: SPB in green, SYNC in blue, L/F in red

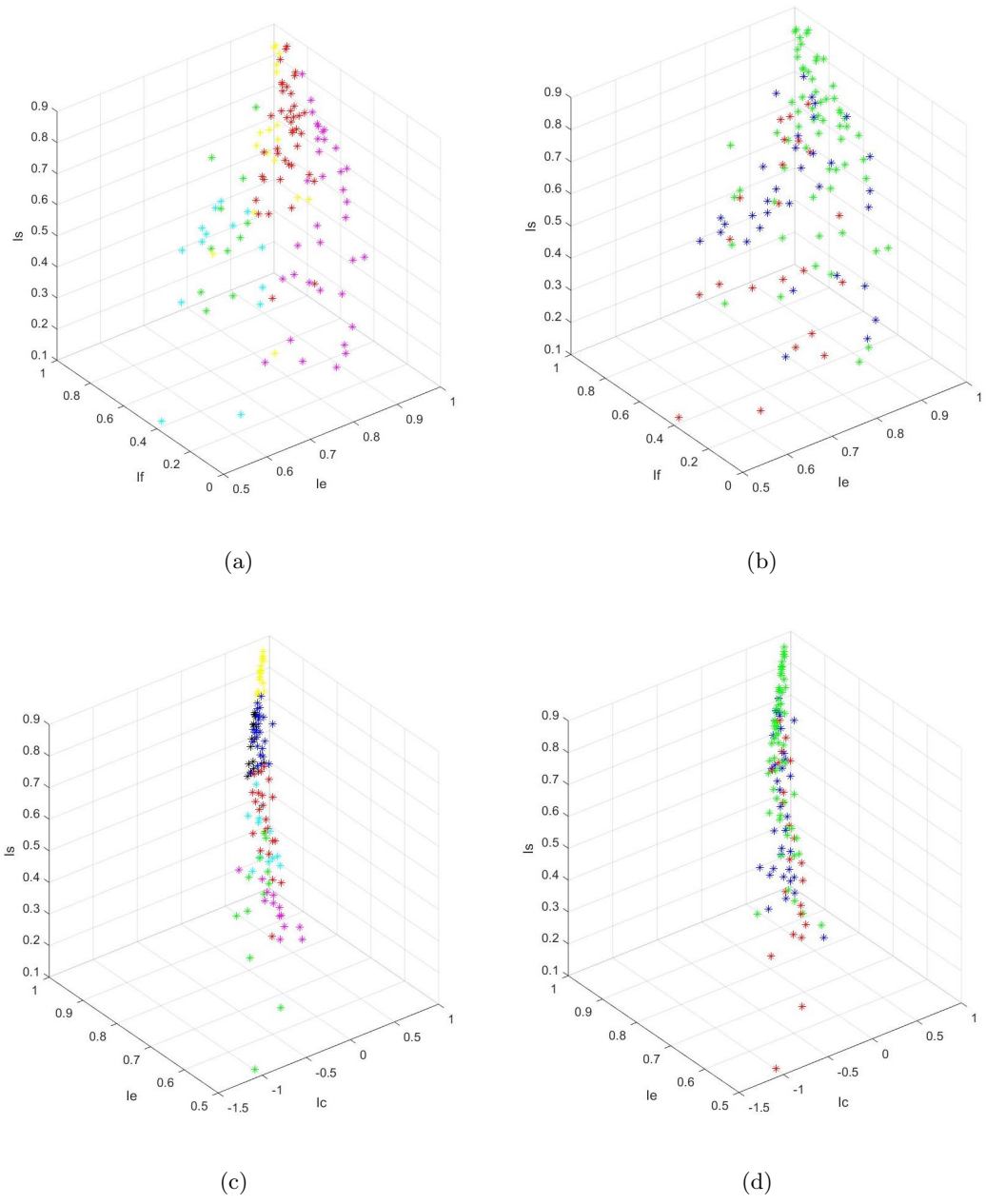


Figure 8: K-means algorithm results comparison: a) 5 centroids, correlation distance metric, centroids not plotted for a better readability; b) dataset points: SPB in green, SYNC in blue, L/F in red; c) 7 centroids, cityblock distance metric, centroids not plotted for a better readability; d) dataset points: SPB in green, SYNC in blue, L/F in red

$$d(x,c) = \sqrt{(x-c)(x-c)'} \tag{3.13}$$

$$d(x,c) = 1 - \frac{xc'}{\sqrt{(xx')(cc')}} \tag{3.14}$$

$$d(x,c) = 1 - \frac{(x-\bar{x})(c-\bar{c})'}{\sqrt{(x-\bar{x})(x-\bar{x})'}\sqrt{(c-\bar{c})(c-\bar{c})'}} \tag{3.15}$$

$$d(x,c) = \sum_{j=1}^{p} |x_j - c_j| \tag{3.16}$$

where

$$\bar{x} = \frac{1}{p}\left(\sum_{j=1}^{p} x_j\right)\bar{1}_p \tag{3.17}$$

$$\bar{c} = \frac{1}{p}\left(\sum_{j=1}^{p} c_j\right)\bar{1}_p \tag{3.18}$$

and x, c and p are the observation, the centroid and the dimension of the observation respectively. $\bar{1}_p$ is a raw vector of p ones.

Moreover, since the initial centroids are randomly chosen, the algorithm was run many times in order to find the best result. The classification was always done feeding the algorithm with all the five metrics.

In order to show the results of the clustering procedure, we decided to construct a 3D plot of the points, choosing 3 indexes out of 5. Examples of these plots are reported in Figure 7 and Figure 8. In all the figures, the identified clusters are reported on the left with different colors, while the original metrics are shown on the right. Different colors were also used to separate different cooperation situations: green for SPB, blue for SYNC and red for L/F.

The 3D graphical results reported and the 5D (5 metrics) classification that we observed as output of the k-means algorithm clearly showed that it is not possible to meaningfully classify the computed metrics dataset using this method. In fact, in all tests, k-means method always failed and found clusters that do not overlap with the three investigated cooperation situations (SPB, SYNC, L/F). This was in a certain sense predictable since the shape of the clusters was found not to be spherical.

3.2.2 DBSCAN Clustering

We next implemented a DBSCAN algorithm. The software used was again MATLAB. The input to the algorithm were the five computed metrics and the results were analyzed both in the 5D space and in the 3D space, as was done for k-means algorithm (sub-section 3.2.1). Since the DBSCAN algorithm requires as input the parameters ϵ and $MinPts$, as explained in section 2.4, a loop was built inside the code in order to test a large range of ϵ and $MinPts$. With an ϵ too small, the algorithm is not able to identify any cluster, classifying all the points as noise. Increasing this parameter too much instead led to one cluster and some isolated noise elements. The algorithm resulted to be less sensitive to the $MinPts$ parameter, but a very small value, 1 for instance, led to a situation in which lots of clusters were identified. When this parameter is increased, between 5 and 50, the results were quite independent from $MinPts$. For very large values, larger than 50, all points were classified as noise in case of small ϵ.

Figure 9 shows the graphical representation of the results along with three of the five metrics with $MinPts$ equal to 5 and ϵ equal to 0.15 (a) and 0.30 (c). Note how by increasing the ϵ parameter, the number of points classified as noise is decreasing, but also the number of clusters

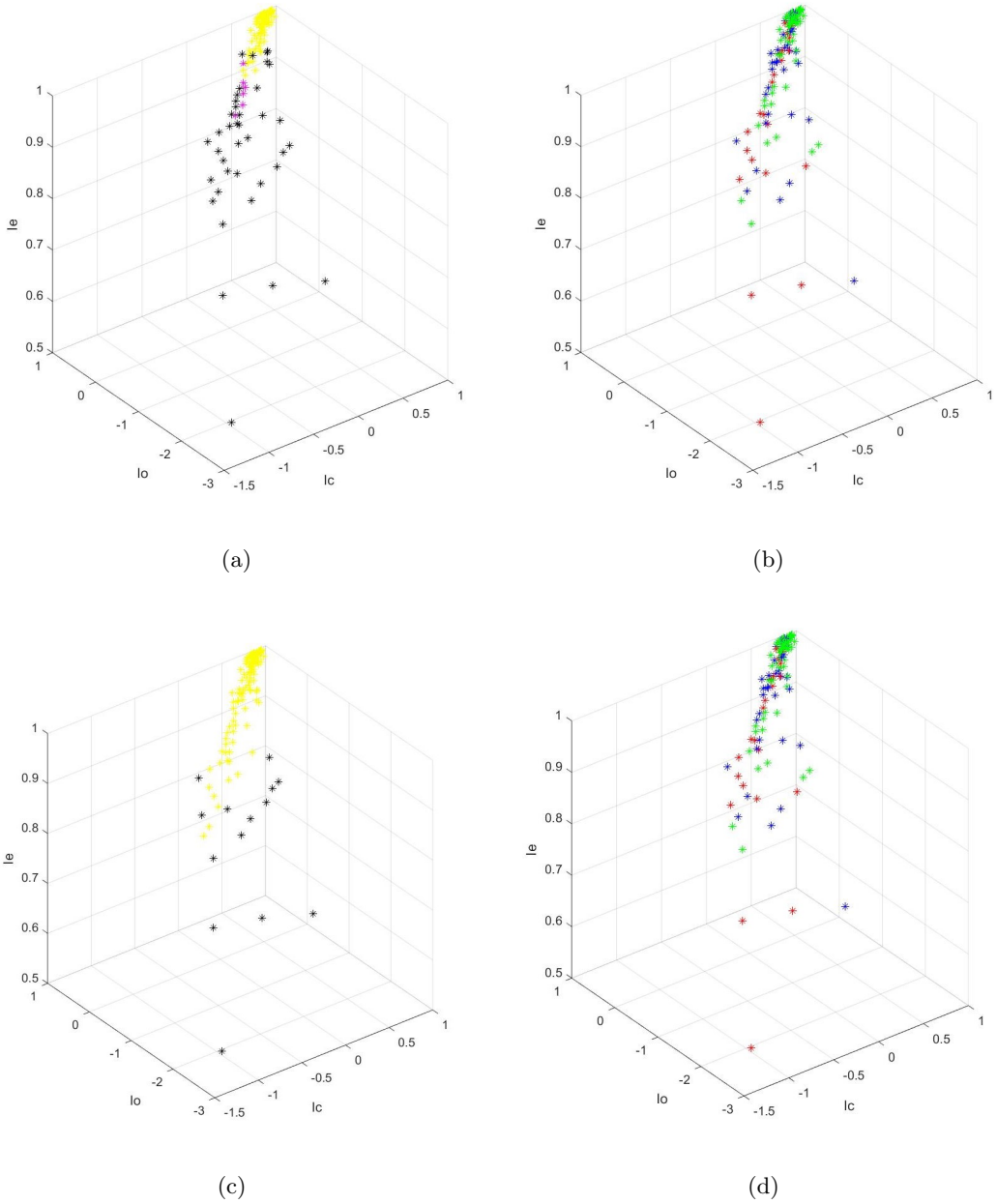


Figure 9: DBSCAN algorithm results comparison: a) ϵ equal to 0,15, $MinPts$ equal to 5, noise in black b) dataset points: SPB in green, SYNC in blue, L/F in red; c) ϵ equal to 0,30, $MinPts$ equal to 5, noise in black; d) dataset points: SPB in green, SYNC in blue, L/F in red

is decreasing. It is apparent from these plots that it is not possible to find clusters similar to the cooperation situations SPB, SYNC and L/F using the DBSCAN algorithm. Moreover, it is apparent how the distribution of the points in the dataset is not homogeneous: the density of points is much higher for values of the metrics approaching 1 than for values of the indexes lower than 0.9. The difference in density justifies the failure of the DBSCAN method that, as a general rule, is not able to classify clusters with large differences in density. Also the number of clusters identified by the algorithm is not meaningful: we were expecting from 3 to 7 clusters, but this method was able to identify, depending on the set parameters, only situations with 1-2 clusters or 8-10 clusters.

3.2.3 Agglomerative clustering

Once again, agglomerative clustering was implemented in MATLAB. Again, different definitions of the distance metric were implemented: Euclidean, cosine, correlation and cityblock. The maximum number of clusters ($MaxCl$) was set equal to 7 since we would like to find clusters similar to the three cooperation situations (SPB, SYNC, L/F) but also possibly sub-clusters for intermediate scenarios.

Graphical 3D results are shown in Figure 10. Also in this case, the obtained clusters are not similar to the expected ones and most of the time clusters with few elements (sometimes even only one) are found. This result is not surprising; the agglomerative clustering is a basic clustering method and most of the time it fails to correctly classify clusters of complex shapes.

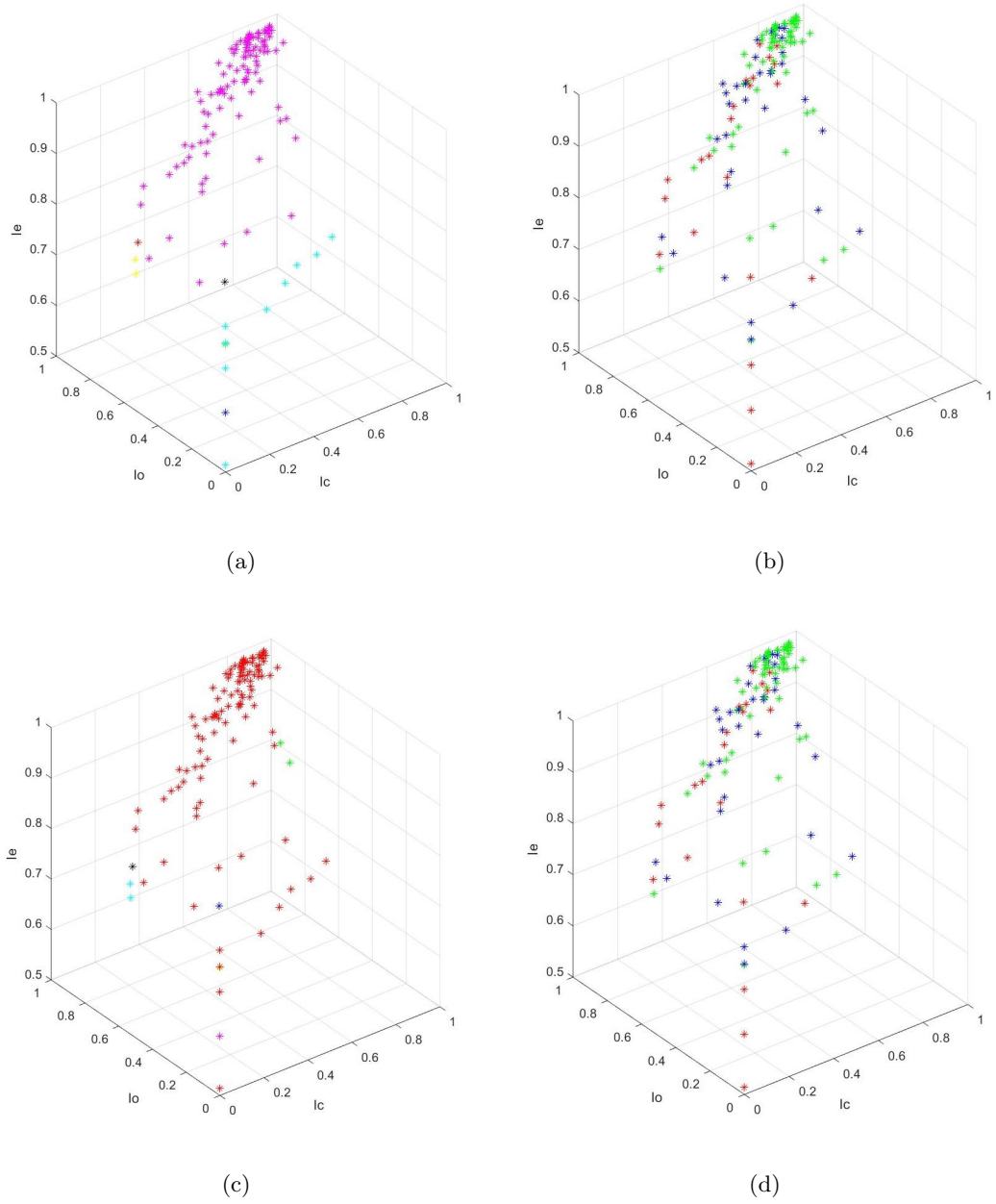


Figure 10: Agglomerative clustering algorithm results comparison: a) $MaxCl$ equal to 7, cosine distance metric; b) dataset points: SPB in green, SYNC in blue, L/F in red; c) $MaxCl$ equal to 7, correlation distance metric; d) dataset points: SPB in green, SYNC in blue, L/F in red

3.2.4 Discussion

Different algorithms were implemented to cluster the dataset obtained applying the metrics to the human study results. All the algorithms failed to find meaningful clusters, no correlation between the clusters and the cooperation scenarios (SPB, SYNC, L/F) could be found. For these reasons, we concluded that no meaningful results could be obtained, in term of clusters, using the existing five metrics.

To further understand why the clustering fails, it is worthwhile considering again how the five metrics were defined. In fact, they were all computed [5] by integrating a certain quantity (depending on the applied/effective forces) from the initial time t_i to the final time t_f . It means that they can compute an average quality of the cooperation, not the quality at every instant. These indexes are thus not applicable if the quality of the cooperation during the task changes. In turn, the five metrics are not appropriate to be used for real-time applications.

Analyzing the force data obtained during the human study, we understood that the degree of cooperation changes during the task. In the case of SYNC scenario, it is not true that the two effective forces are always similar. In the L/F scenario, it is not true that a dominant force always exists, but in most of the tasks the ratio between the forces changes during the motion. In conclusion, it was not possible to identify which trial belongs to which specified cooperation scenario, SPB, SYNC or L/F, analyzing the force data.

3.3 Quotient Index Definition

From the analysis of section 3.2, it is clear that a different approach has to be used to study the cooperation. In particular, it is important to evaluate the cooperation at every instant

during the task and not by averaging the results. Based on this conclusion, a new metric was defined. We decided to relate this new index, named *quotient index* (I_q), directly to the force values, in particular the effective forces. I_q is defined as follows:

$$I_q = \frac{f_1^*}{f_2^*} \tag{3.19}$$

that is the quotient of the two effective forces. Note that I_q, as defined, is not bounded and can be either positive or negative. In case it is positive, the cooperation is said to be *active*; otherwise it is *passive*. What is important to note is that this new metric can be computed in real time and there are no integrals that need to be computed.

3.4 Quotient Index and Clustering

During the human study the force data were acquired at $1KHz$. To investigate clustering with the quotient index we down-sampled the data to $10Hz$. This means that we look at the information about the cooperation every $100ms$. This was done in order to reduce the amount of data and also because we assume that that the nature of cooperation does not change faster than $10Hz$. To perform the analysis, we computed the effective forces with PM-OFF and then the quotient index I_q for all the samples obtained from the human study. As explained in section 3.3, this new metric is not bounded and can be either positive or negative. To bound the value of the index, it has been redefined as follow:

$$I_q(t) = \begin{cases} \frac{f_1^*}{f_2^*}, & \text{if } f_1^* \leq f_2^* \\ \frac{f_2^*}{f_1^*}, & \text{if } f_1^* > f_2^* \end{cases} \tag{3.20}$$

With this adjustment, I_q is always bounded between -1 and +1. To apply the clustering methods to the new metric, we decided to analyze only scenarios corresponding to an active cooperation (the great majority of data), rejecting all the negative values of the quotient index. As a result, the values of I_q in our analysis were always bounded between 0 and 1.

3.4.1 K-means Clustering

Different clustering methods were applied to the set of quotient indexes computed for each data point in the experimental dataset. The first algorithm that was used is k-means. Multiple runs were attempted to find a reasonable number of clusters and a good definition of the distance metric. Figure 11 shows the results of the k-means algorithm when 5 clusters were specified and an Euclidean distance was adopted. Note how the algorithm groups the points in the same cluster computing the distance from the centroids. Similar results were observed specifying a different number of clusters or a different distance metric; in this case, 5 clusters were considered to be an appropriate number of cooperation situations to be identified. However, other results, such as when 3 or 4 clusters were specified, were also meaningful. A too small or too big number of cooperation situations could simplify or complicate the analysis too much, respectively.

3.4.2 DBSCAN Clustering

The DBSCAN method was implemented as well. Again, many attempts were made, changing the ϵ and the $MinPts$ parameters. A satisfactory result was found setting ϵ equal to 0.0175

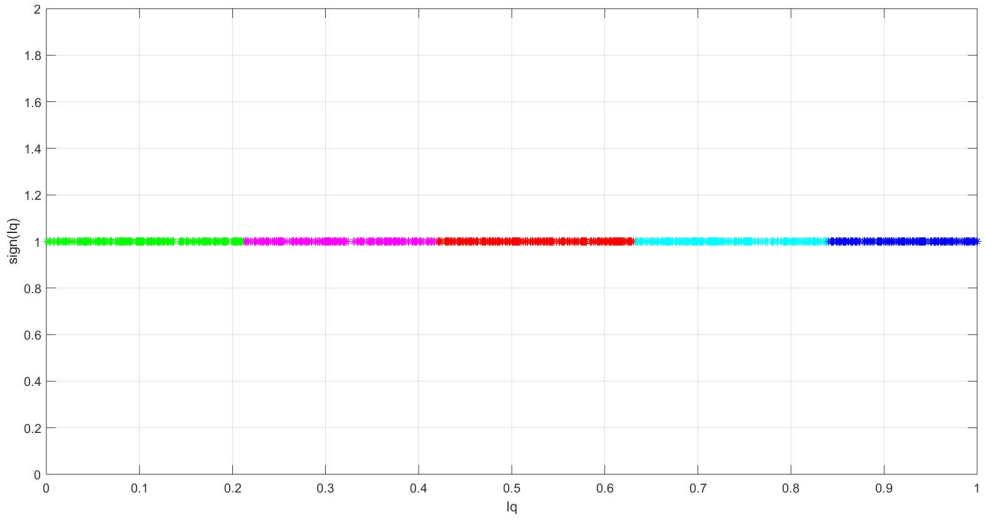


Figure 11: Results of the k-means algorithm application to the computed quotient indexes: 5 clusters specified

and $MinPts$ equal to 50 (Figure 12). With these parameters, 8 different clusters plus some noise elements were identified. In particular, three main clusters (red, orange, light green) were recognized for values of I_q larger than 0.4, four clusters (dark green, blue, cyan and purple) for the index going from 0.17 to 0.4 and another cluster when I_q is lower than 0.17. From the plot, it is clear how the density of the computed indexes increases when the values of I_q increases. On the contrary, for small value of the quotient index, a low density is reported (lots of noise elements for values of I_q close to zero).

The region of values included in the interval [0.17 , 0.4] could be grouped in one cluster only. In fact, the presence of different clusters in this interval is a proof of the low density level

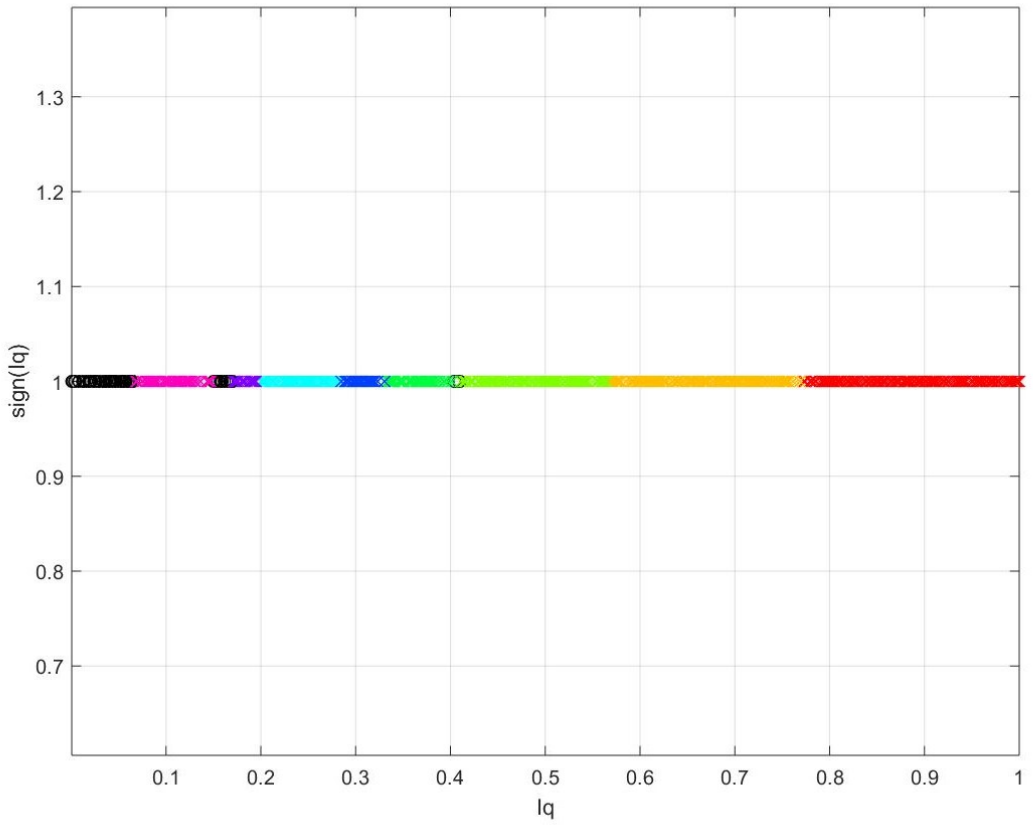


Figure 12: Results of the DBSCAN algorithm application to the computed quotient indexes: $\epsilon = 0.0175$, $MinPts = 50$

of the region; moreover, the entire interval is bounded by some noise elements. Grouping dark green, blue, cyan and purple clusters into one reduces the total number of clusters to 5 and allows a better identification of the possible cooperation scenarios.

3.4.3 Agglomerative Clustering

The last clustering method implemented on the quotient indexes in this phase of research is agglomerative clustering. In this clustering method it is possible to specify the maximum number of clusters to use ($MaxCl$ parameter) and to set different distance metrics. Good results were obtained using the Euclidean or cityblock distances; the cosine and correlation distances did not perform well. $MaxCl$ was set to 7, since this is a reasonable maximum number of cooperation scenarios that we wanted to identify.

Figure 13 shown the results obtained with the agglomerative clustering algorithm, setting $MaxCl$ equal to 7 and using the Euclidean distance. Five different clusters were identified, with different densities and extensions.

3.4.4 Clustering Results Analysis

Three different clustering algorithms, k-means, DBSCAN and agglomerative clustering, were implemented to identify possible cooperation scenarios based on the quotient index I_q. These methods generated comparable results.

Both k-means and agglomerative clustering identified 5 clusters (for k-means this parameter was specified) of similar densities and extensions. Note how Figure 11 and Figure 13 are graphically similar in terms of found clusters. The boundaries are obviously different and the k-means clusters seem to be a little bit shifted to the right with respect to agglomerative ones. The majority of the points not close to the boundaries are classified in the same way by both algorithms.

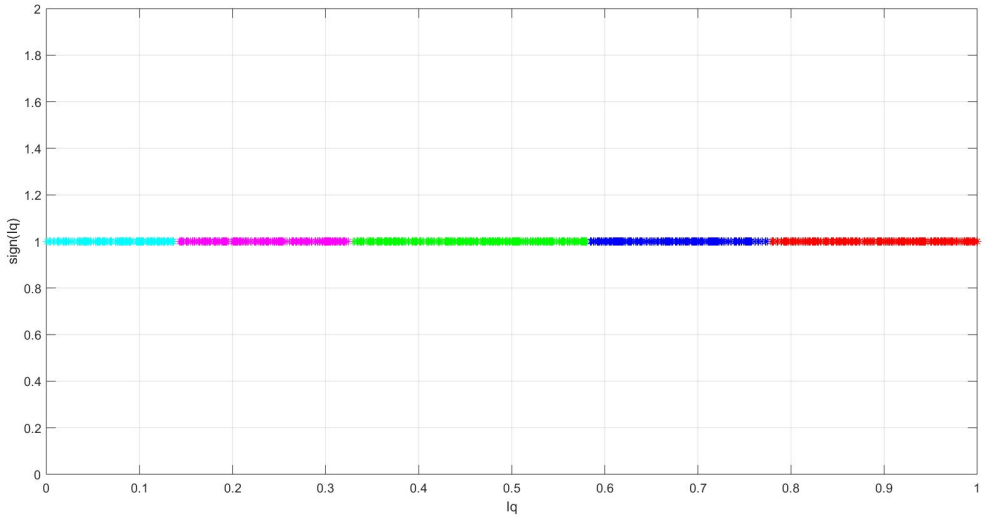


Figure 13: Results of the agglomerative clustering algorithm application to the computed quotient indexes: $MaxCl = 5$, Euclidean distance

The DBSCAN method produced slightly different results, identifying 8 clusters. Subsequently, 4 clusters were grouped into one (sub-section 3.4.2) because similar properties were observed, reducing the total number of clusters to 5. These final clusters resulted to be very similar to those identified by the agglomerative clustering algorithm and similar to k-means clusters. However, the DBSCAN method identified noise elements (for values of I_q close to zero), while the other methods were not able to do the same since they do not have the notion of noise.

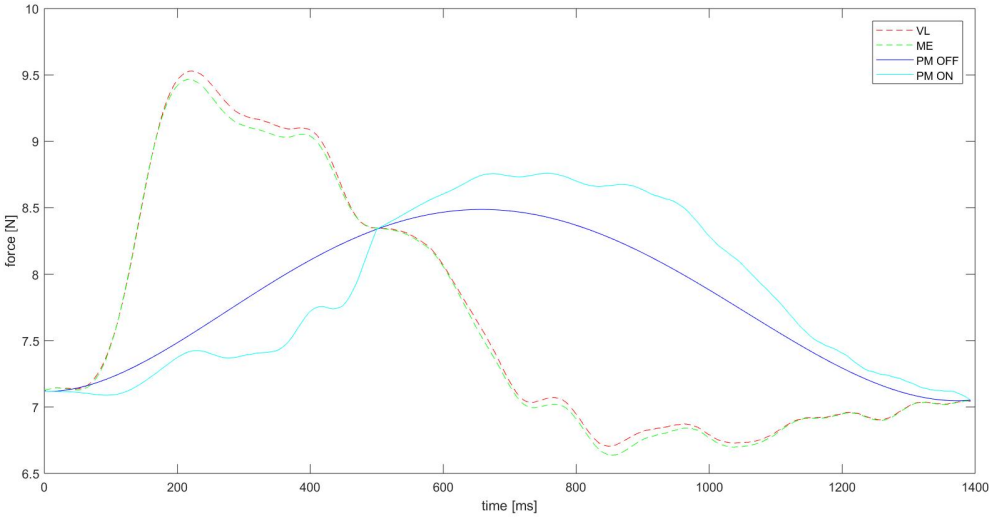
In general, the three clustering methods produced similar results in terms of the identified cooperation scenarios, but at the same time it is necessary to analyze in depth the results.

First, the quotient index is a 1D metric while the quantitative metrics presented in section 3.1 are 5D. The results of the clustering process in 1D cannot be directly compared with the clustering results on the quantitative metrics. The clustering process in 1D could simplify the analysis too much. Second, the choice of bounding the quotient index between 0 and 1 was necessary in order to find meaningful clustering results, but on the other hand does not allow to distinguish two situations with the same index but different larger effective force (f_1^* or f_2^*). For these reasons, the bounding constraint was removed for future analysis and the number of cooperation scenarios was selected based on their physical meaning.

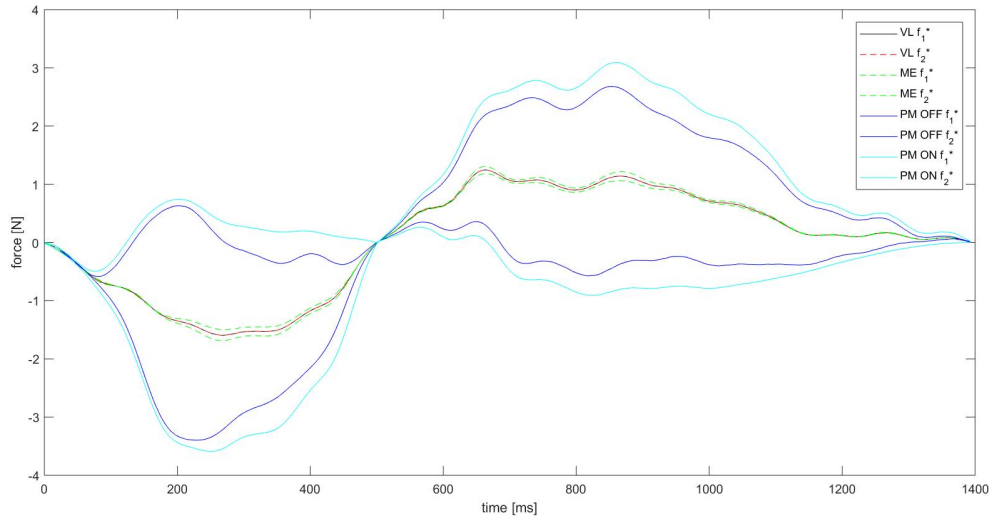
3.5 Quotient Index and Interaction Force Models

The quotient index metric was defined at every time instant as the ratio of the two effective forces at that time instant. In order to compute the quotient index it is thus necessary to calculate the interaction force, which in turn allows the computation of the effective forces from the applied ones, at every time instant. In chapter 2, VL, ME, PM-OFF and PM-ON models were introduced and can be used to compute the interaction force. In section 3.4, the effective forces were computed using PM-OFF, but as discussed previously, this method cannot be used for a real-time application. It is thus necessary to devise an on-line (real-time) method to compute the interaction force that can provide meaningful results when used with the quotient index.

Let's analyze how VL, ME, PM-OFF and PM-ON methods are used to compute the interaction force and the effective forces. Figure 14(a) shows the interaction force computed by the four models for the same force data. Note how VL and ME models compute similar values



(a)



(b)

Figure 14: VL model, ME model, PM-OFF and PM-ON comparison: a) computation of the interaction force; b) computation of the effective forces

for the interaction force, but that are completely different from those obtained with PM-OFF and PM-ON. There is a similarity, sometimes stronger, sometimes less evident, between the results of PM-OFF and PM-ON: in fact PM-ON is directly obtained from PM-OFF, adding a prediction on the future values of the applied force.

Since the effective forces are directly computed from the applied forces (that are measured by force sensors) by adding or subtracting the interaction force, we expect that similar patterns will be observed in the effective forces. This is in fact demonstrated in Figure 14(b) that shows the results of the computation of the effective forces using the four different models. Note how the results for VL and ME models are very similar, but substantially different from PM-OFF. As explained in sub-section 2.3.1, VL model computes two equal effective forces in case of dyadic manipulation task; that is why the two effective forces are the same in the plot. The minimum energy model is able to differentiate two effective forces, visible on the plot, but the difference between them is small. PM-OFF and PM-ON, instead, both produce effective forces that are quite asymmetric.

After the computation of the effective forces, it was possible to extract the quotient indexes. The data were again down-sampled to $10Hz$. The quotient indexes in the case of VL model resulted to be all equal to 1, since this model assumes two equal effective forces in case of dyadic manipulation. Therefore, in the case of VL model, the I_q index is meaningless. The ME model also produces quotient indexes close to 1: more than 90% of the computed indexes are bounded between 0.9 and 1. Also in this case, the quotient index thus fails to identify possible cooperation scenarios.

Different outcome was generated by the PM-ON model (the results for the PM-OFF model were already presented in section 3.4). To investigate the PM-ON model we decided to use the original (unbounded) quotient index

$$I_q = \frac{f_1^*}{f_2^*} \tag{3.21}$$

and to consider only active cooperation (positive values of I_q). Then, three different cooperation scenarios were defined, based on the value of the index. Only three intervals were chosen in order to simplify the eventual implementation on the robot since each scenario needs to be mapped to a different robot controller. The scenarios are described in Table I. The parameter β is bounded between 0 and 1 and identifies (with β^{-1}) three different cooperation situations: L/F, COOP and F/L. Even though the names resemble those used for the human study, there is no direct mapping between them. As said in the previous sections, it was not possible to recognize the specified cooperation scenarios during the human study, SPB, SYNC and L/F, analyzing the force data. For this reason, we propose three new cooperation scenarios, L/F, COOP and F/L, based on the physical meaning of the ratio of the applied forces. We use the ground data, obtained during the human study, in order to test our proposed labels (L/F, COOP, L/F) and neglecting the specified cooperation scenarios (SPB, SYNC, L/F). The parameter β is setting the boundaries of the three proposed labels. This parameter is very important and must be selected taking into account the physical meaning of the quotient index, that is the ratio of the two effective forces. L/F stands for leader/follower, F/L for follower/leader. In these two situations, the quotient index is much larger or smaller than 1 and one effective

I_q **range**	**Cooperation scenario**
$I_q \leq \beta$	L/F
$\beta < I_q < \beta^{-1}$	COOP
$I_q \geq \beta^{-1}$	F/L

TABLE I: PROPOSED COOPERATION SCENARIOS

β	β^{-1}	PM-ON accuracy w.r.t PM-OFF
0.01	100	89.12%
0.1	10	65.71%
0.2	5	53.53%
0.4	2.5	47.87%
0.5	2	49.32%
0.6	1.67	50.28%
0.8	1.25	56.84%
0.9	1.11	61.35%

TABLE II: PM-ON PERCENTAGE ACCURACY OF CORRECT CLASSIFIED COOPERATION SCENARIOS WITH RESPECT TO PM-OFF

force is significantly larger than the other. The leader is the partner that is applying the larger effective force; the other participant is the follower. The work is entirely or mostly provided by the leader; the follower just follows the motion. In the cooperation scenario (COOP) instead, the value of I_q is bounded between β and β^{-1}, that means that the magnitude of the two effective forces is quite similar and the work is almost equally split between the two partners.

Finally, the indexes for the PM-ON model were computed and the results were compared with the PM-OFF model. In particular, we computed the agreement between the on-line model and the off-line model. For every time instant (remember the data were down sampled at $10Hz$), the quotient index was computed applying PM-OFF and PM-ON; if the identified cooperation scenario was the same, a correct classification was registered; if instead, the scenarios were different, a misclassification occurred. The test was repeated varying the parameter β in order to see how the accuracy of PM-ON with respect to PM-OFF changed. Table II shows the results of the comparison. Again, the table shows the percentage accuracy of the on-line model with respect to the off-line model in recognizing the same cooperation scenarios among the three proposed, L/F, COOP and F/L. There is no any kind of relation with the specified cooperation scenarios during the human study, SPB, SYNC and L/F.

A high percentage of correctly classified scenarios was obtained in case of very small or very large (always lower than 1) value of β. However these results are misleading. In fact, a low value of β corresponds to a situation in which the COOP scenario covers a much wider range than the other two scenarios: in this case all the situations would be interpreted as COOP so the results are meaningless. On the other hand, in the case of very high values of β, close to 1, the range for the COOP scenario is reduced and the only two scenarios that are effectively used are L/F and F/L. This case is again meaningless. To clarify, an example is proposed. If a low value of β is chosen, 0.1 for instance, trails where one effective force is ten times larger than the other would be classified as good cooperation cases, but obviously these results are wrong

from a physical point of view. As said previously, β sets the boundaries of the proposed labels and its value must be selected taking into account the physical meaning of the quotient index.

The range of values for β that produce meaningful results from a physical point of view seems to be somewhere between 0.4 and 0.7. With these values of β the accuracy of PM-ON with respect to PM-OFF significantly decreases to approximately 50%. It means that one time out of two, PM-ON is able to recognize in real-time the same cooperation scenario as PM-OFF (which cannot be computed in real-time). This is an encouraging result that promises to provide a reasonable pathway to subsequently implement a control methodology for pHRI on a robot.

CHAPTER 4

COOPERATION SCENARIOS AND ROBOT CONTROL

This chapter describes how the information on the cooperation intent can be used to control the robot as it physically interacts with a human. In the first two sections, an overview of the Robot Operating System (ROS) and the Baxter Robot is given. Then a set of robot controllers and a switching scheme that selects how they should be activated is proposed that optimizes the interaction. Preliminary results on the implementation of the individual robot controllers on the Baxter Robot are presented next. Finally, we discuss some strategies that could improve the accuracy of the PM-ON model and thus potentially improve the performance of the human-robot physical interaction.

4.1 Robot Operating System (ROS)

ROS, Robot Operating System, is an open source flexible framework for writing robot software. In particular, it enables developers to provide services such as hardware abstraction, low-level device control, device communication, etc. ROS uses a graph architecture in which processes are identified as nodes and can be connected via topics. On node, called the Master, stands on top of the others and is always directly connected to all the other nodes. Communication between nodes is achieved using the publish/subscribe paradigm. A publisher that generates an item publishes it as a message on the ROS network. The receiver nodes subscribe to topics of interest and when a message on the topic of interest is published, they receive the

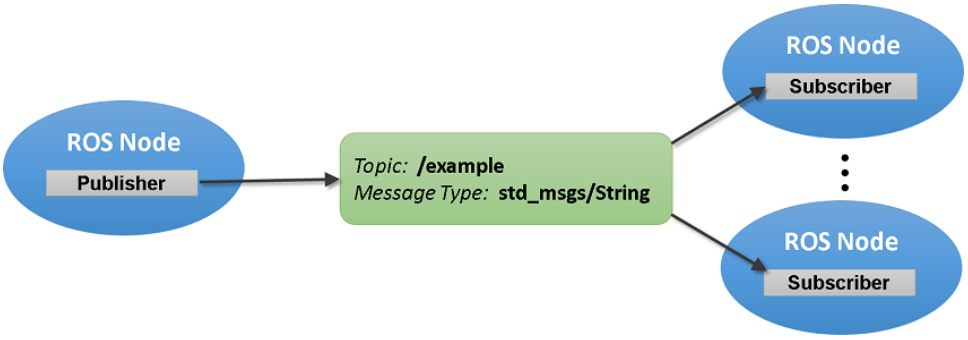


Figure 15: Schematic representation of the publisher and subscriber communication method in ROS. Reprinted from Exchange Data with ROS Publishers and Subscribers, by The MathWorks, Inc. Copyright 1994-2018 The MathWorks, Inc.

message. An example is a sensor which publishes on the ROS network the measured quantities at a certain frequency. Processes that need the sensor data subscribe to the appropriate topic and receive the sensor packets as they get published. Figure 15 shows a situation in which a message is published by a node and is forwarded to a group of subscriber nodes.

The main client libraries for ROS are written in C++, Python or LISP and are used in various scenarios such as motion recognition, perception, gesture synthesis, etc. The application programming interface (API) for a robot is typically provided by the robot manufacturer, usually in C++ or Python. But the main attraction of ROS is that due to hardware abstraction layer, the code written in ROS that provides a certain high-level functionality for a particular robot can be easily used on other robots. In case of the Baxter Robot, the robot maker, Rethink Robotics, provides libraries written in C++ and Python for robot control. These libraries have been used in our experiments.

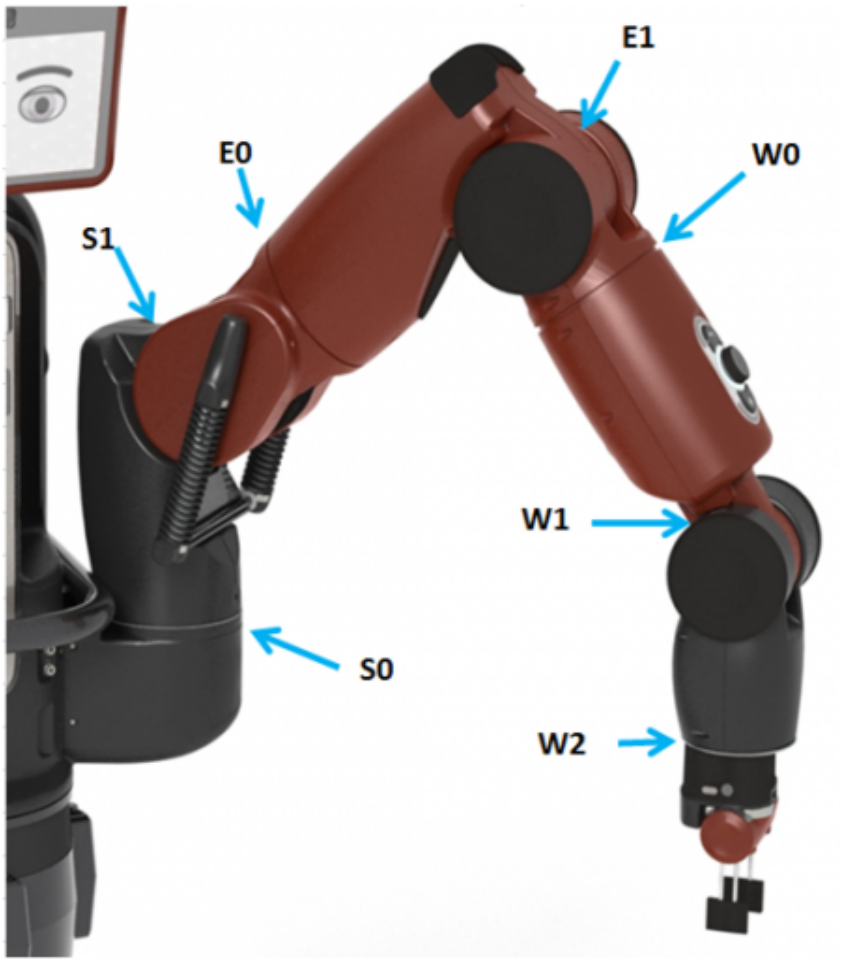


Figure 16: Names and positions of the seven joints in the Baxter left arm. Reprinted from Arms, by Rethink Robotics, 2015. Copyright 2015 Rethink Robotics.

4.2 Baxter Robot Description and Control Modes

Baxter is a robot built by Rethink Robotics. It is widely used for academic research and in industrial applications (the industrial version). Baxter is made of a torso-like movable pedestal, two arms that can be outfitted with grippers and a head-pan with a screen that can be rotated horizontally. The arms are composed of seven revolute joints connected by eight links; from the robot's torso to the hand, in sequence, the joints are $s_0, s_1, e_0, e_1, w_0, w_1, w_2$, where s, e, w stand for shoulder, elbow and wrist. The end-effector of the robot is located at the joint w_2. Figure 16 shows the joint positions for the left arm.

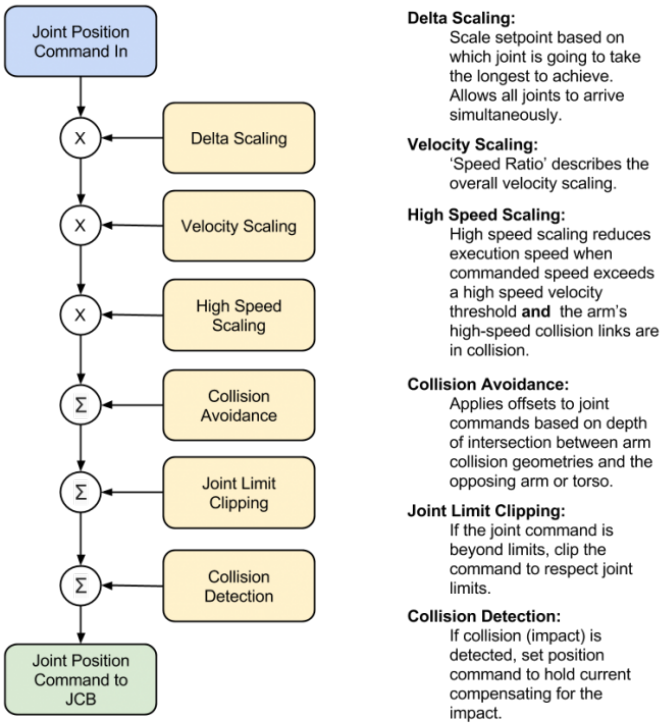
Four different control modes are available for the Baxter Robot: joint position control, raw joint position control, joint velocity control and joint torque control. We discuss each in turn.

4.2.1 Joint Position Control

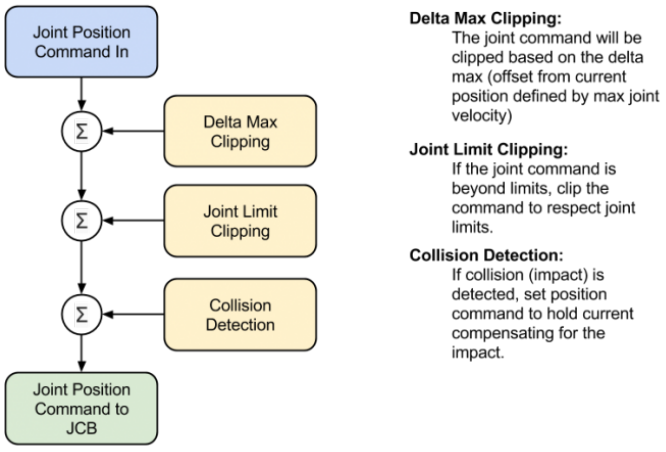
The joint position control mode allows to command the angle, in radiants, of each of the seven joints. The desired configuration of the robot for the controller is thus described by a vector of seven elements. With this method it is possible to specify an initial and a final point for a point-to-point motion, or to specify an entire trajectory, providing a stream of points. Figure 17(a) describes in detail the sequence of steps involved in joint position control. Note how a series of safety checks are added to avoid dangerous situations for humans, the environment or the robot itself.

4.2.2 Raw Joint Position Control

The raw joint position control is an extension of the basic joint position control. In this control mode lots of safety features are removed to guarantee a more direct control. Obviously,



(a)



(b)

Figure 17: Schematic representation of the safety features in joint position control (a) and raw joint position control (b). Reprinted from Arm Control Modes, by Rethink Robotics, 2015. Copyright 2015 Rethink Robotics.

this option must be used with caution, since it can lead to dangerous situations. Figure 17(b) represents the schematic of this control mode.

4.2.3 Joint Velocity Control

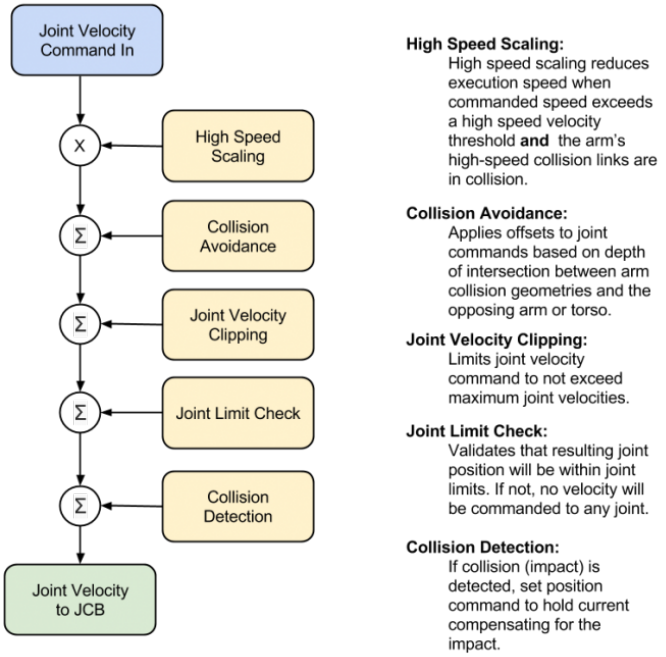
The joint velocity control is an advanced control mode in which a vector of seven elements containing the desired velocities (in radiants per second) of all the joints is forwarded to the joint control boards (JCB_s) for execution. This control mode includes some safety features but not as many as in the case of joint position control, so it must be used with caution (Figure 18(a)).

4.2.4 Joint Torque Control

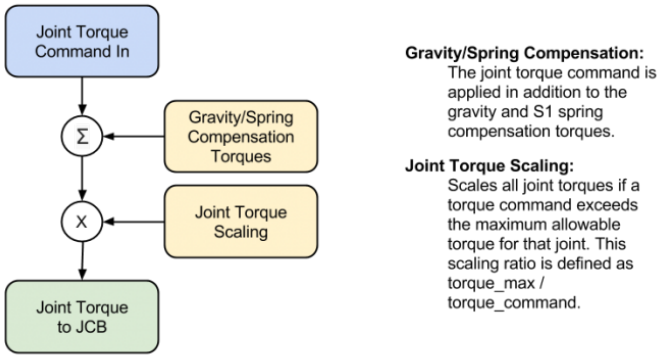
This control mode allows the most direct access to the JCB_s, commanding directly the torque values to be applied by the actuators. In this case, the safety checks are few and dangerous situations may occur quite easily. Particular attention has to be paid to the *gravity/spring compensation torques* block (Figure 18(b)). This term adds to the torque we are commanding the torques necessary to compensate the weight of the arm and the spring force exerted by the s_1 joint.

4.2.5 Experimental Results

Among the four available control modes, since the aim of the research was to control the interaction that is in turn described with forces, we tried to implement several control algorithms using the joint torque control. In particular, in order to test this control mode, a simple proportional-integral-derivative (PID) controller was implemented. The goal of the test was to move the end effector from an initial pose to a final one, directly commanding the torques of



(a)



(b)

Figure 18: Schematic representation of the safety features in joint velocity control (a) and joint torque control (b). Reprinted from Arm Control Modes, by Rethink Robotics, 2015. Copyright 2015 Rethink Robotics.

the actuators. The real aim was to understand the precision and the safety of the joint torque control mode.

A first set of tests was performed maintaining the gravity compensation torques active. That means the robot control software was adding the torques necessary to compensate the weight of the arm to the commanded values. In this case, the arm began to be unstable after few seconds and the emergency button had to be pressed. We tried to adjust the controller parameters (the PID parameters were tested in a simulation environment), but a successful result was never obtained. Subsequently, the PID controller was tested by switching the gravity compensation torques off. The weight of the arm was thus not compensated any more. In this case, the robot did not go unstable, but the results were still not satisfactory. In most of the cases, the commanded torques were not able the compensate the high friction inside the joints. And when they were able to do so, oscillations were always present.

In conclusion, the tests performed using the joint torque control mode never produced satisfactory results. We believe that the main reason for lackluster performance of the joint torque control was an inadequate gravity compensation model (Rethink Robotics stated that they are trying to improve this model) and the presence of high friction inside the joints that is very difficult to model. For these reasons, we decided that the joint torque control mode is not suitable to control the Baxter Robot for human-robot interaction.

We performed many other tests implementing different controllers and testing all the control modes available. The aim of these tests was to understand the precision, repeatability and safety level of the robot. It turned out that the robot is absolutely safe when controlled in position

control mode and not completely safe when other control modes are selected. The performances in terms of precision and repeatability are quite poor with all control modes. We concluded that Baxter is not a good robot for human-robot interaction and satisfactory results cannot be achieved with this hardware. Baxter Robot can be used for the initial testing phase, but a more precise robot has to be used in order to get meaningful results in pHRI.

4.3 Control Framework

In section 3.5, three possible cooperation scenarios, L/F, COOP, and F/L, were proposed based on the value of the quotient index I_q. Encouraging results were obtained when the quotient index was computed on-line using the PM-ON model. To improve the quality of human-robot interaction and adapt the robot behavior to what is expected by the human, we propose to link each cooperation scenario to a different robot controller. In the case of three scenarios, we thus propose the theoretical behavior of three possible controllers.

In case of robot leader and human follower (L/F), the Baxter Robot has to provide the entire or the majority of the work for the task, while the human is just following the motion. The robot should move trying to replicate the most natural point-to-point trajectory used by humans which is the minimum-jerk motion. Figure 19 shows the results of a test where a minimum-jerk trajectory is commanded to the Baxter using joint position control and raw joint position control. The initial and final positions on the x-axis are 0.0 and 0.5 (meters, in the Baxter reference frame), the duration of the task 3 seconds. The mean-square-errors resulted to be equal to $0.279mm$ and $0.281mm$ for the joint position control and the raw joint position control respectively.

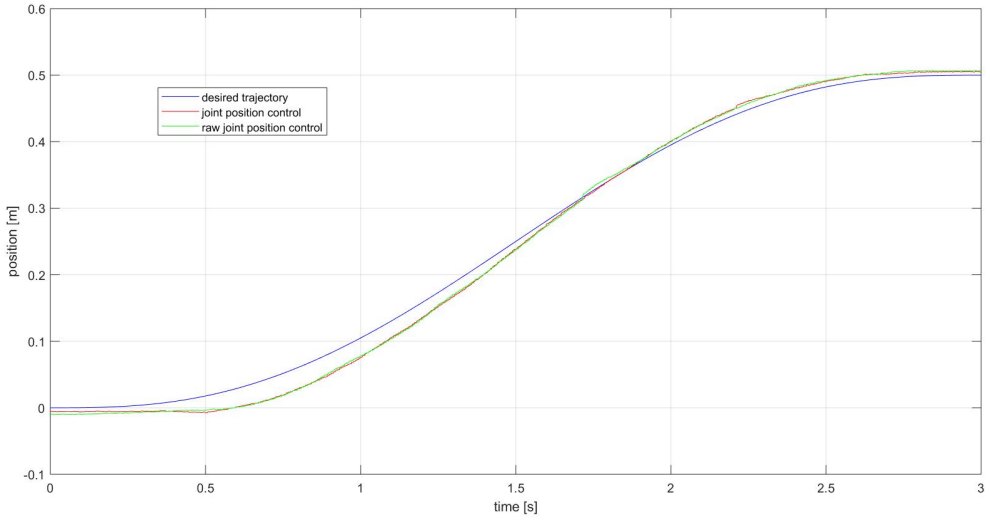


Figure 19: Comparison between the desired minimum jerk trajectory (a) and the actual ones using joint position control (b) and raw joint position control (c)

In the case of cooperative motion scenario (COOP), the work has to be provided by both the human and the robot. The robot cannot simply follow the human and, at the same time, cannot just impose its motion on the human. Instead, it must able to react in real-time to optimize the interaction. A good candidate controller for this scenario is the impedance control, defined as

$$f_R^* = M\ddot{x}_d + k_d(\dot{x}_d - \dot{x}) + k_s(x_d - x) - \frac{1}{2}M\bar{g} \tag{4.1}$$

where M stands for the mass of the object to be moved, $x_d, \dot{x}_d, \ddot{x}_d, x, \dot{x}, \ddot{x}$ are the desired and actual position, velocity and acceleration respectively, k_d and k_s are the vectors of damping and

spring constants, $\bar{g}$ is the gravity, f_R^* the robot's effective force. The desired values (position, velocity, acceleration) are provided by the minimum-jerk trajectory. The damping and the spring constants need to take into account all the variations from the nominal minimum-jerk trajectory, reacting appropriately to the human effective forces. The tuning of these parameters allows to interact with a more stiff or more compliant robot; a good solution could be to select these parameters so that the robot impedance matches the impedance of the human arm [21] [22].

Based on the results of the tests with the joint torque control mode, a direct force control (torque control) cannot be implemented on Baxter. Researchers obtained satisfactory results implementing position-based or velocity-based impedance control in which an impedance control is simulated controlling the position or the velocity and not directly the torque/force [23] [24].

The last cooperation scenario assumes the robot to be the follower and the human the leader (F/L). In this case, the Baxter Robot must follow the motion imposed by the human, that is providing the entire or the majority of the requested work. One of the most used controllers for this kind of applications is the impedance control, that allows the robot to follow the human motion. We propose a different controller. Human effective force can be computed in real-time using PM-ON, then applying Newton's laws a relation between force and velocity is found:

$$f_H^* = M\frac{dv}{dt} \tag{4.2}$$

where f_H^* is the human effective force, M the object's mass and v the object's velocity. Integrating this relation in time, it is possible to find the velocity as function of the human effective force:

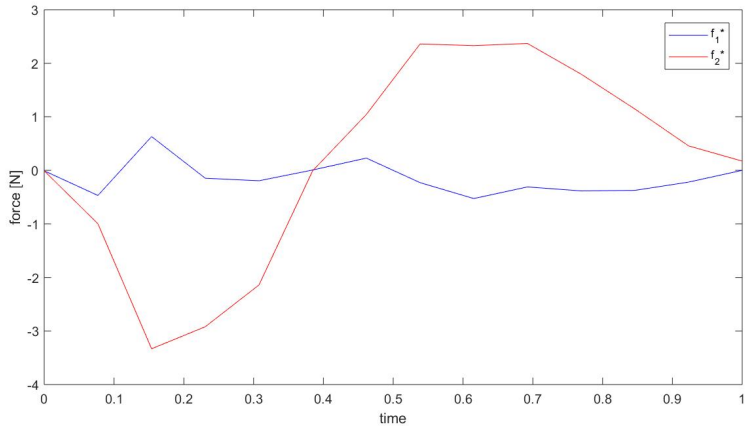
$$v = \int_{t_i}^{\tau} \frac{f_H^*}{M} d\tau \tag{4.3}$$

where t_i is the initial time and τ the current time. The robot could now be controlled with the joint velocity control mode: the end effector velocity is known, the joints velocities can be computed applying inverse kinematics (IK).

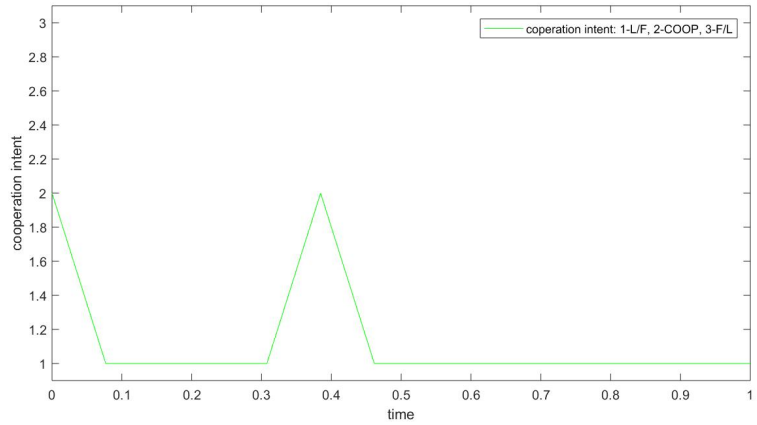
With respect to the impedance control that makes the robot reactive to the human force tuning the damping and spring constants, the controller we propose just follows the human motion. The key idea is that in this cooperation scenario, the robot simply has to follow the motion; if the human force changes, then a different quotient index is computed and a different cooperation label, and robot controller, is probably selected.

4.4 Improving the Cooperation Strategy

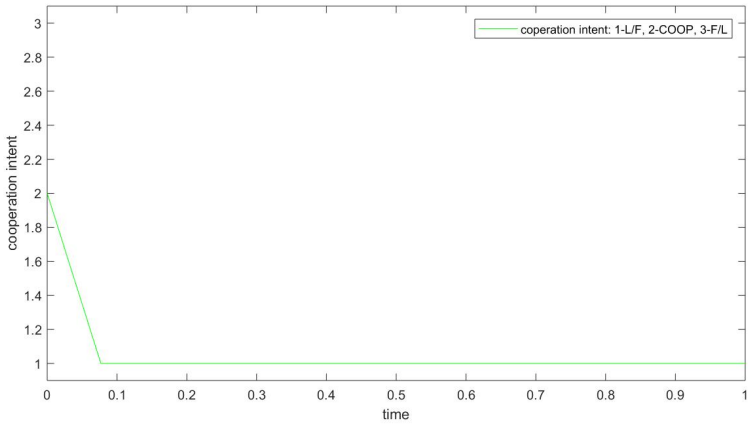
In the previous chapter, we showed that the quotient index I_q can be used to recognize the cooperation intent and to improve the quality of human-robot interaction. However, when the effective forces are small, the ratio between them becomes very sensitive to noise, possibly resulting in miss-classifying the cooperation intent. To clarify the issue, imagine a particular instant in which f_1^* and f_2^* are equal to $0.1N$ and $0.05N$, respectively. I_q will be 2 (or 0.5) and the identified cooperation situation will be L/F (or F/L). From a physical point of view instead, no motion will probably occur in this situation because the forces are so small.



(a)



(b)



(c)

Figure 20: a) effective forces in a clear L/F scenario; b) quotient index calculated without considering the magnitude of the effective forces; c) quotient index computed following the adjusted cooperation strategy

Figure 20 shows another situation in which the computed I_q can lead to a questionable outcome. Figure 20(a) shows a clear L/F scenario: f_2^* (in red) provides the entire effort for the task and acts as a leader while f_1^* (in blue) just follows. Figure 20(b) reports the cooperation intent computed applying the quotient index, giving values of 1, 2 and 3 to indicate L/F, COOP and F/L, respectively. The collaboration is classified as L/F for almost the entire duration of the task. Only two instants are identified as COOP. In particular, note how at instant 0.4, there is a jump in the graph. This jump is caused by small values of the effective forces at instant 0.4 that are interpreted by I_q as COOP scenario, even if the entire motion is a clear L/F situation.

In order to avoid this misleading intent identification, a little adjustment of the control strategy is introduced: at every time instant, if the biggest effective force is smaller than a set threshold force, the cooperation intent is not calculated and the controller of the previous time instant is adopted. Figure 20(c) reports the results of the application of this new strategy with a force threshold equal to $0.2N$. Note how at instant 0.4, the cooperation intent is still L/F (no more COOP) and a satisfactory result is achieved. This adjustment introduces possible delays in the robot's reaction, but since the forces are sampled at very high frequency in case of real-time applications (around $1KHz$), these delays will likely not be noticeable by humans.

CHAPTER 5

CONCLUSION AND FUTURE WORK

This research investigated human-robot collaboration trying to recognize intents and how to increase robot performances during the interaction. First, results from other researches have been analyzed, in particular a study conducted at the Robotics Lab at University of Illinois at Chicago (UIC). We showed how traditional approaches failed in computing the collaboration intent and cannot be used in real-time applications.

Then we proposed a new metric for the recognition of different cooperation scenarios, named quotient index, I_q. The properties and the physical meaning of this index have been presented too. Different models for the computation of the interaction force have been implemented and tested with I_q. The on-line polynomial model resulted to be a valid candidate to be used with the new metric during a real-time human-robot interaction in order to identify the collaboration intent. The other models, virtual linkage and minimum-energy were not able the recognize different collaboration intents, while the off-line polynomial model cannot be implemented in real-time applications.

The number of collaboration scenarios were then reduced to three, for simplicity, and each scenario was associated to a particular robot controller able to improve the quality of the interaction. An improvement of the collaboration strategy was also proposed; it is able to take into account particular cases in which the computed intent is misleading.

5.1 Future Work

This research contributed in studying human-robot interaction and presented useful results for a future real-time application. Through this direction, it would be interesting to implement the proposed controllers and study if the robot is able to react in the desired way when commanded by them. Then a new human study have to be set up and tests have to be performed; the humans have to be asked about their feeling when interacting with the robot.

The accuracy of the actual on-line polynomial model with respect to the off-line version resulted to be around 50%. For future real-time applications, it would be interesting to work on a better prediction of the future force values used in the on-line model in order to increase its accuracy.

CITED LITERATURE

1. Hughes, V. A., Roubenoff, R., Wood, M., Frontera, W. R., Evans, W. J., and Fiatarone Singh, M. A.: Anthropometric assessment of 10-y changes in body composition in the elderly. The American Journal of Clinical Nutrition, 80(42):475–482, 2004.

2. Szulc, P., Beck, T. J., Marchand, F., and Delmas, P. D.: Low skeletal muscle mass is associated with poor structural parameters of bone and impaired balance in elderly menthe minos study. Journal of Bone and Mineral Research, 20(5):721–729, 2005.

3. Williams, D. and Khatib, O.: The virtual linkage: a model for internal forces in multi-grasp manipulation. In 1993 Proceedings IEEE International Conference on Robotics and Automation, pages 1025–1030 vol.1, May 1993.

4. Groten, R., Feth, D., Goshy, H., Peer, A., Kenny, D. A., and Buss, M.: Experimental analysis of dominance in haptic collaboration. In RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication, pages 723–729, Sept 2009.

5. Noohi, E., Žefran, M., and Patton, J.: A model for human-human collaborative object manipulation and its application to humanrobot interaction. IEEE Transactions on Robotics, 32:1–17, 08 2016.

6. Todorov, E.: Optimality principles in sensorimotor control (review). Nature Neuroscience, 7(9):907–915, 2004.

7. Flash, T. and Hogan, N.: The coordination of arm movements: an experimentally confirmed mathematical model. Journal of Neuroscience, 5(7):1688–1703, 1985.

8. Uno, Y., Kawato, M., and Suzuki, R.: Formation and control of optimal trajectory in human multijoint arm movement. Biological Cybernetics, 61(2):89–101, Jun 1989.

9. Garvin, G. J., Žefran, M., Henis, E. A., and Kumar, V.: Two-arm trajectory planning in a manipulation task. Biological Cybernetics, 76(1):53–62, Jan 1997.

10. Noohi, E., Parastegari, S., and Žefran, M.: Computational model for dyadic and bimanual reaching movements. In 2015 IEEE World Haptics Conference (WHC), pages 260–265, June 2015.

11. Nehaniv, C. L., Dautenhahn, K., Kubacki, J., Haegele, M., Parlitz, C., and Alami, R.: A methodological approach relating the classification of gesture to identification of human intent in the context of human-robot interaction. In ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005, pages 371–377, Aug 2005.

12. Kulic, D. and Croft, E. A.: Estimating intent for human-robot interaction. 2003.

13. Breazeal, C. and Aryananda, L.: Recognition of affective communicative intent in robot-directed speech. Autonomous Robots, 12(1):83–104, Jan 2002.

14. Kelley, R., Wigand, L., Hamilton, B., Browne, K., Nicolescu, M., and Nicolescu, M.: Deep networks for predicting human intent with respect to objects. In Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction, HRI '12, pages 171–172, New York, NY, USA, 2012. ACM.

15. Kelley, R., Tavakkoli, A., King, C., Nicolescu, M., Nicolescu, M., and Bebis, G.: Understanding human intentions via hidden markov models in autonomous mobile robots. In Proceedings of the 3rd ACM/IEEE International Conference on Human Robot Interaction, HRI '08, pages 367–374, New York, NY, USA, 2008. ACM.

16. Erden, M. S. and Tomiyama, T.: Human-intent detection and physically interactive control of a robot without force sensors. IEEE Transactions on Robotics, 26(2):370–382, April 2010.

17. Kazerooni, H.: Human-robot interaction via the transfer of power and information signals. IEEE Transactions on Systems, Man, and Cybernetics, 20(2):450–463, Mar 1990.

18. Duchaine, V. and Gosselin, C.: Safe, stable and intuitive control for physical human-robot interaction. In 2009 IEEE International Conference on Robotics and Automation, pages 3383–3388, May 2009.

19. Mielke, E. A., Townsend, E. C., and Killpack, M. D.: Analysis of rigid extended object co-manipulation by human dyads: Lateral movement characterization. CoRR, abs/1702.00733, 2017.

20. Shadmehr, R. and Wise, S. P.: The Computational Neurobiology of Reaching and Pointing: A Foundation for Motor Learning. Cambridge, MA, USA, MIT Press, 2005.

21. Puzi, A. A., Sidek, S. N., and Sado, F.: Mechanical impedance modeling of human arm: A survey. IOP Conference Series: Materials Science and Engineering, 184(1):012041, 2017.

22. Artemiadis, P. K., Katsiaris, P. T., Liarokapis, M. V., and Kyriakopoulos, K. J.: Human arm impedance: Characterization and modeling in 3d space. In 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3103–3108, Oct 2010.

23. Pelletier, M. and Doyon, M.: On the implementation and performance of impedance control on position controlled robots. In Proceedings of the 1994 IEEE International Conference on Robotics and Automation, volume 2, pages 1228–1233, May 1994.

24. Jlassi, S., Tliba, S., and Chitour, Y.: An online trajectory generator-based impedance control for co-manipulation tasks. In 2014 IEEE Haptics Symposium (HAPTICS), pages 391–396, Feb 2014.

25. Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G.: Robotics: Modelling, Planning and Control. Springer Publishing Company, Incorporated, 1st edition, 2008.

VITA

NAME	Stefano Castagneri
EDUCATION	
	Master of Science in Electrical and Computer Engineering, University of Illinois at Chicago, July 2018, USA
	Master of Science in Mechatronic Engineering , Polytechnic of Turin, July 2018, Italy
	Bachelor's Degree in Mechanical Engineering, Polytechnic of Turin, July 2016, Italy
	Bachelor's Degree in Mechanical Engineering, Tongji University, July 2016, China
LANGUAGE SKILLS	
Italian	Native speaker
English	Working proficiency
	A.Y. 2017/18: One Year of study abroad in Chicago, Illinois
	A.Y. 2014-2018: Lessons and exams attended exclusively in English
Chinese	Basic level
	A.Y. 2014-2015: One year of study abroad in Shanghai, China
SCHOLARSHIPS	
Fall 2017	Italian scholarship for final project (thesis) at UIC
Fall 2017	Italian scholarship for TOP-UIC students
Fall 2014	Italian scholarship for Politong students

Spring 2015	Italian scholarship for Politong students

TECHNICAL SKILLS

Basic level	Project management
Average level	Basic and applied mechanics, electrical and hydraulic machines, model-based design for automotive, controllers
Advanced level	Robotics and mechatronic systems

WORK EXPERIENCE AND PROJECTS

July 2017	Participant at European Innovation Academy (EIA), Cofounder and Chief Technology Officer (CTO) at OnePlug, tech start-up for electrical home appliances.
Jan 2016 - July 2016	Internship at Gruppo Torinese Trasporti (GTT). Detection of improvement actions to reduce the operation's duration on the tramway in the central workshop.