



**POLITECNICO**  
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE  
E DELL'INFORMAZIONE

# Towards the development of an autonomous guide for visually impaired users for sport activities

TESI DI LAUREA MAGISTRALE IN  
AUTOMATION AND CONTROL ENGINEERING - INGEGNERIA  
DELL'AUTOMAZIONE

Author: **Manuel Nava, Virginia Di Mauro**

Student ID: 987815, 995736  
Advisor: Prof. Marcello Farina  
Academic Year: 2023-24



# Abstract

Visual impairment poses significant challenges, affecting daily activities such as navigating unfamiliar environments, avoiding obstacles, and participating in physical activities, thereby diminishing the overall quality of life. This project's main objective is to leverage technology to address the unique needs of visually impaired individuals, fostering a more inclusive environment.

The thesis is conducted within the BUDD-e project (Blind-assistive aUtonomous Droid Device), funded by Politecnico di Milano through the POLISOCIAL 2021 program and the PRIN 2022 project CARE (Control of Assistive Robots in crowded Environments). The project aims to tackle daily challenges encountered in accessing services, public spaces, sports arenas, healthcare facilities, shopping areas, and cultural centers. By developing a novel robotic guide, the goal is to provide a comprehensive solution that surpasses existing technological limitations and empowers users to engage in diverse environments, including sports activities. Although the BUDD-e prototype is promising for mobility concerns, it faces challenges related to reactivity and adaptability. Indeed, enhancements are necessary to support visually impaired individuals in various scenarios.

First, this thesis focuses on optimizing the smart tether system configuration to enhance practicality and user comfort, addressing issues, such as undesirable oscillations and discomfort in sports environments.

Secondly, to achieve a responsive and adaptable robotic guide, a new distance controller is devised to ensure more reactive and precise control of the robot's position relative to the user. Additionally, a critical stopping issue is addressed by introducing an innovative strategy, consisting of a prompt user brake detection and of a feedforward brake controller when necessary. These advancements address previous limitations and expand potential applications.

**Keywords:** assistive technology, robotic guide, smart tether system, braking detection, distance control, inclusive environment, mobility solutions.



## Abstract in lingua italiana

Le disabilità visive comportano sfide significative agli individui che ne soffrono, influenzando le attività quotidiane come orientarsi in ambienti non familiari, evitare ostacoli e la partecipare ad attività fisiche. Queste difficoltà riducono complessivamente la qualità della vita. L'obiettivo principale di questo progetto è sfruttare la tecnologia per rispondere alle esigenze uniche delle persone con disabilità visive, favorendo un ambiente più inclusivo.

La tesi è condotta nell'ambito del progetto BUDD-e (Blind-assistive aUtonomous Droid Device), finanziato dal Politecnico di Milano attraverso il programma POLISOCIAL 2021 e il progetto PRIN 2022 CARE (Control of Assistive Robots in crowded Environments). Il progetto si propone di affrontare le sfide quotidiane relative all'accesso a servizi e spazi pubblici, come impianti sportivi, strutture sanitarie, aree commerciali e centri culturali. Sviluppando una innovativa guida robotica, l'obiettivo è fornire uno strumento che superi le limitazioni delle tecnologie esistenti e consenta agli utenti di svolgere attività, anche sportive, in diversi ambienti.

Sebbene il prototipo BUDD-e sia una soluzione promettente per i problemi di mobilità, deve affrontare sfide legate alla reattività e all'adattabilità per migliorare il support a persone ipovedenti in diversi scenari.

Nella prima parte questa tesi si concentra sull'ottimizzazione della configurazione del cosiddetto smart tether system per migliorare il comfort dell'utente, affrontando problemi come le oscillazioni indesiderate e il disagio negli ambienti sportivi.

Nella seconda parte della tesi, per ottenere una guida robotica reattiva e adattativa, è stato ideato un nuovo controllore che garantisce un controllo più reattivo e preciso della posizione del robot rispetto all'utente. Inoltre, è stato affrontato un problema critico di frenata introducendo un approccio innovativo al problema di controllo della distanza nella fase in cui l'utente si ferma. La strategia proposta consiste nella rilevazione tempestiva della frenata dell'utente e prevede un'azione di frenata in feedforward quando necessario. Questi miglioramenti affrontano le limitazioni precedenti ed espandono le applicazioni potenziali.

**Parole chiave:** tecnologia assistiva, robot guida, smart tether system, riconoscimento di frenata, controllo di distanza, ambiente inclusivo, soluzioni di mobilità.

# Contents

<b>Abstract</b>	<b>i</b>
<b>Abstract in lingua italiana</b>	<b>iii</b>
<b>Contents</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Mobility challenges for visually impaired individuals . . . . .	1
1.2 State of the art in assistive technological solutions for visually impaired people . . . . .	2
1.3 The BUDD-e Robotic guide prototype and open issues . . . . .	7
1.3.1 Overview of the BUDD-e Robotic Guide . . . . .	7
1.3.2 The Yape Robotic Platform . . . . .	7
1.3.3 The Smart Tether system . . . . .	8
1.3.4 The Control System . . . . .	9
1.3.5 Tests and suggested improvements . . . . .	9
1.3.6 Open issues . . . . .	12
1.4 Thesis contributions . . . . .	13
<b>I New design of the smart tether system</b>	<b>15</b>
<b>2 System Structure and Components of the first setup</b>	<b>17</b>
2.1 Original setup . . . . .	17
2.2 New winch setup . . . . .	22
2.3 New load cell setup . . . . .	23
<b>3 Control of the first setup</b>	<b>27</b>
3.1 Model identification . . . . .	27
3.1.1 Identification setup description . . . . .	27

3.1.2	Winch model identification . . . . .	29
3.1.3	Tether model identification . . . . .	31
3.1.4	Model validation . . . . .	37
3.1.5	The comprehensive identified model of the smart tether system . . .	40
3.2	Frequency domain controller (Configuration 1) . . . . .	42
3.3	LMI Controller design (Configuration 2) . . . . .	45
3.3.1	The Control-Oriented Model . . . . .	45
3.3.2	Linear Matrix Inequalities and $\mathcal{H}_2$ control design . . . . .	47
3.3.3	$\mathcal{H}_2$ Controller design for the smart tether system . . . . .	50
<b>4</b>	<b>System Structure, components and control of the second setup (Configuration 3)</b>	<b>55</b>
4.1	New setup . . . . .	55
4.2	Control . . . . .	63
4.2.1	Motor setup . . . . .	63
4.2.2	Motor control . . . . .	64
4.2.3	Safety functions . . . . .	65
<b>5</b>	<b>Performance comparison between the different configurations</b>	<b>69</b>
<b>II</b>	<b>Distance control</b>	<b>83</b>
<b>6</b>	<b>Distance feedback control system</b>	<b>85</b>
6.1	Distance controller set-up . . . . .	85
6.1.1	LiDARs . . . . .	85
6.1.2	Target Tracking . . . . .	87
6.1.3	Target tracking on BUDD-e . . . . .	90
6.2	The Yape longitudinal speed dynamical model identification . . . . .	91
6.2.1	The Yape Longitudinal Speed Discrete-Time Switching Model . . .	93
6.2.2	$\mathcal{H}_2$ Controller Design . . . . .	94
6.3	Distance Controller . . . . .	95
6.3.1	Integral Action . . . . .	96
6.4	Experimental validation . . . . .	96
<b>7</b>	<b>Distance controller in braking conditions</b>	<b>101</b>
7.1	Braking control strategy . . . . .	101
7.2	Multivariate analysis . . . . .	102
7.2.1	Preliminary steps . . . . .	102

7.2.2	Mahalanobis distance . . . . .	103
7.3	Braking recognition in BUDD-e . . . . .	104
7.3.1	Experimental setup and data . . . . .	104
7.3.2	Braking recognition . . . . .	108
7.3.3	Experimental results . . . . .	109
7.4	Velocity-dependent braking detection . . . . .	116
<b>8</b>	<b>Conclusions and future work</b>	<b>121</b>
	<b>Bibliography</b>	<b>125</b>
	<b>List of Figures</b>	<b>127</b>
	<b>List of Tables</b>	<b>133</b>
	<b>Acknowledgements</b>	<b>135</b>



# 1 | Introduction

## 1.1. Mobility challenges for visually impaired individuals

Visual impairment poses significant challenges to individuals, affecting their daily activities. Navigating through unfamiliar environments, avoiding obstacles, and participating in physical activities become intricate tasks, leading to an overall diminished quality of life [15]. The main objective of this project is rooted in the commitment to leveraging technology to address the unique needs of visually impaired individuals, creating a more inclusive environment.

In [12] it is highlighted that various challenges are encountered in medical facilities and healthcare settings, such as hospitals, nursing homes, and day-care centers. Health service providers are obligated to ensure that visually impaired people have an equitable chance to engage in and derive benefits from all the offerings and services. Achieving this involves actions like guiding patients, promoting basic techniques for sighted assistance and mobility, using language and etiquette sensitive to disabilities, and providing appropriate devices upon request, that are necessary for the majority of the visually impaired people [18].

Similar issues are observed in the field of sports. Specifically, individuals with visual impairments face obstacles when doing sports like running, where constant guidance and the presence of trained guides are imperative. These guides play a crucial role, standing slightly ahead and to the side of the visually impaired runner, connected by a tether.

Traditional solutions, e.g. white canes and trained dogs, have limitations in the contexts discussed above. On the one hand, guide dogs require extensive training, incur maintenance costs, and may pose allergen concerns. On the other hand, white canes provide basic obstacle detection but lack the sophistication required for dynamic environments. They can provide little navigation information and the need for more supportive and precise technologies has been highlighted [6]. Assistive technologies, and in particular robotic

solutions, are envisioned as promising solutions to the mobility issue.

## 1.2. State of the art in assistive technological solutions for visually impaired people

In the last decades, various technologies have been proposed to face secure navigation, one of the numerous challenges for visually impaired people in daily life. Technological advances have significantly contributed to creating more inclusive environments for visually impaired individuals.

Visual assistive technologies (VATs) can be broadly categorized into two types: visual enhancement and visual substitution [9] aids.

First, visual enhancement devices capture, process, and display digital images on a screen, assisting partially sighted users by improving visual clarity. On the other hand, visual substitution systems bypass the need for visual input entirely by delivering sensory information through non-visual modalities, such as auditory or haptic signals. These devices take inputs from a variety of sensors and then translate them into a form that can be understood by those with visual impairments.

Visual substitution devices can be grouped into three categories:

- Electronic travel aids (ETAs), which help users navigate in the environment by converting visual information and additional sensor data into formats that can be heard or felt through tactile feedback.
- Electronic orientation aids (EOAs), which are often integrated into ETAs and guide users in choosing the best route through their environment.
- Position locator devices (PLDs), like GPS, which identify specific locations within a given area, facilitating users in finding their outdoor position relative to landmarks or other points of interest.

Among ETAs, we can find sophisticated models of white canes, designed to better assist users in navigating their environments. The *GuideCane*, described in [3] and illustrated in Figure 1.1, features a long handle attached to two steerable wheels integrated with a sensor unit. These ultrasonic sensors are pivotal in detecting obstacles, enabling the cane to navigate around them autonomously. Users perceive the direction through the force of the steering command, allowing them to follow the path determined by the *GuideCane*.



for outdoor navigation, especially on smartphone devices [9].

For example [2] presents the development and evaluation of a novel algorithm designed to enhance the detection and localization of zebra crossings using smartphone technology, primarily assisting visually impaired individuals in safely navigating road crossings. The ZebraRecognizer algorithm significantly improves upon previous solutions by incorporating a rectification phase that eliminates perspective distortion from the zebra crossing images, which allows for precise computation of the crossing's position relative to the user (Figure 1.3). This approach results in improved accuracy and reduced computation time, which enhances user interaction by providing real-time feedback.

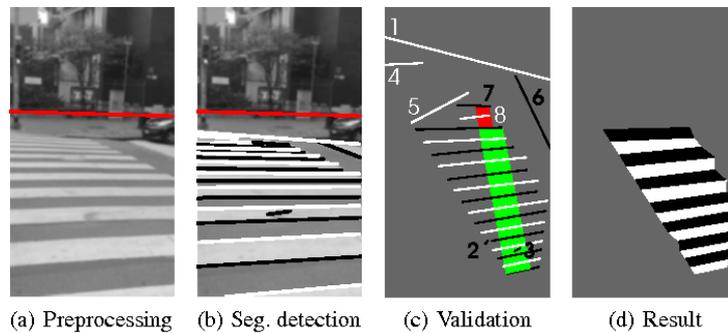


Figure 1.3: Example of ZebraRecognizer applied to an image

Robots, particularly those designed as mobility aids, represent a crucial intersection of robotics, sensory technology, and accessibility design. The primary purpose of robotic guides is to assist visually impaired individuals in navigating environments with greater ease, and enhance safety and independence. These systems use a variety of sensors to perceive their surroundings, including LiDARs, cameras, and ultrasonic sensors, which help in obstacle detection and path planning. Indeed, as highlighted in [18], it is of primary importance to integrate multiple sensor systems to precisely detect obstacles and to enhance the environmental awareness of robotic guides, which is crucial, e.g., in dynamic and unpredictable outdoor settings.

Significant research has been conducted to understand the specific needs and preferences of visually impaired users, which is critical for designing effective robotic guides. For example, [8] highlights the importance of incorporating end-user feedback into the design process to ensure that the robotic guides meet the actual needs of visually impaired individuals.

Autonomous navigation robots have been developed over the last years. For example, CaBot (Figure 1.4) is designed to guide users to their destinations while avoiding obstacles and providing directional feedback through a vibrotactile handle. The robot's design

incorporates LiDAR technology for localization and path planning and a stereo camera for object recognition[7].

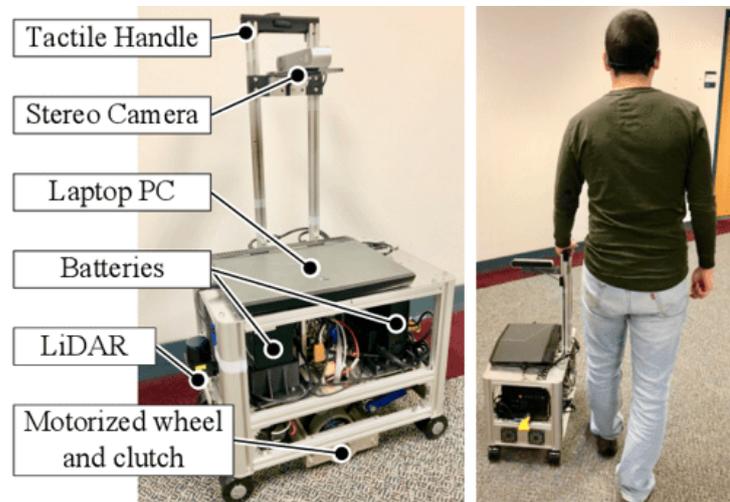


Figure 1.4: CaBot

A guiding device utilizing the Pioneer 3DX robot (Figure 1.5) has been equipped with a D-shaped user-friendly handle and software enabling audio interactions. It enhanced the usability of the robot through interactive audio features, physical design improvements, and customizability, retaining user data to personalize future interactions[10].



Figure 1.5: The Pioneer 3DX robot equipped with a wooden harness

The Robotic Guide Dog (Figure 1.6) is the quadrupedal device presented in [16] and is

designed to assist visually impaired individuals in navigating through narrow and cluttered spaces, realistically imitating a dog. The core of the innovation involves a leash-guided quadruped robot, which uses a hybrid physical Human-Robot Interaction (pHRI) model to manage the dynamics between humans and robots through leash tension, making the interaction easier and smoother. However, its small dimensions may potentially cause users injuries since it may not be perceivable.

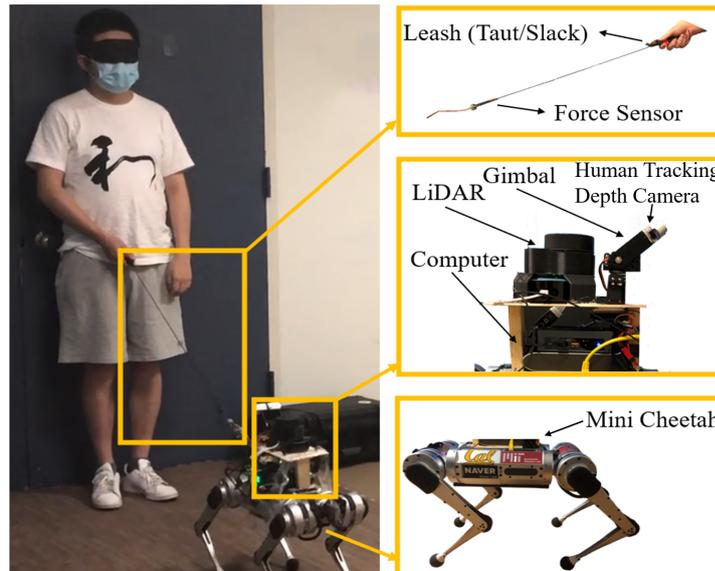


Figure 1.6: Robotic Guide Dog

Yet, the focus in the literature has been more on outdoor applications and collision avoidance capabilities, while less attention has been given to electronic travel aids (ETAs) that support indoor navigation for visually impaired people. Furthermore, solutions that manage the force exerted on the user and, at the same time, adjust their speed to match the user's pace still need to be devised.

The research presented in this thesis has been conducted in the framework of the BUDD-e project (Blind-assistive Autonomous Droid Device), funded by Politecnico di Milano through the POLISOCIAL 2021 program and the PRIN 2022 project CARE (Control of Assistive Robots in crowded Environments). It stems from the need to address the challenges daily faced by visually impaired individuals, such as accessing services, public spaces, sports arenas, healthcare facilities, shopping areas, and cultural centers for example. By developing a novel robotic guide, we aim to provide a comprehensive solution that not only overcomes the limitations of existing technologies but also empowers users to actively engage in diverse environments, including sports activities. This initiative emphasizes incorporating feedback from visually impaired users at every stage of the design

process.

### 1.3. The BUDD-e Robotic guide prototype and open issues

#### 1.3.1. Overview of the BUDD-e Robotic Guide

BUDD-e is an autonomous robotic guide designed to assist visually impaired users in navigating complex environments. The system consists of a self-balancing robotic platform, Yape, equipped with a Smart Tether System (a winch connected to a tether and a handle), and a sophisticated control architecture to ensure safe and effective guidance.

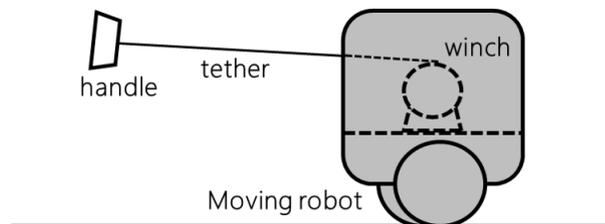


Figure 1.7: Budd-e components scheme

#### 1.3.2. The Yape Robotic Platform

Yape, produced by Yape S.r.l., is a differential drive mobile robot with a two-wheeled inverted pendulum structure. The torque required for stabilization and driving is provided by two brushless DC motors housed in the hubs of its two bicycle-like wheels. All components are contained within a steel chassis with a cubic structure of 50 cm per side, covered with high-density polystyrene, and the total mass is approximately 40 kg. Yape can reach a maximum velocity of about 20 km/h and can handle slopes up to 10°.

In the BUDD-e system, Yape has two main roles:

- It navigates and avoids obstacles, guiding the user to a pre-defined destination.
- It perceives the position of the user and uses this information to adapt to his/her velocity.

The stabilization, speed control, trajectory planning, and tracking in Yape are achieved using both proprioceptive and exteroceptive sensors. An Inertial Measurement Unit (IMU) with 6 degrees of freedom (a 3-axis accelerometer and a 3-axis gyroscope) is attached to the inner structure, while an RS16 Robosense 16-layer LiDAR sensor is mounted on top of

the vehicle. Data from these sensors are managed by the Vehicle Control Unit (VCU), a central microcontroller responsible for running the stabilization and speed tracking algorithms. Torque references are sent to the wheel motors via a CAN bus line. Additionally, Yape is equipped with a Navigation Control Unit (NCU), a Nvidia Xavier AGX board that collects data from the exteroceptive sensors, executes decision-making algorithms, and communicates velocity references to the VCU.

Yape can be controlled in two different modes of operation:

- **Manual mode:** a human driver can impose direction and speed commands through a Logitech F710 remote controller.
- **Autonomous mode:** Yape can drive autonomously relying on a pre-defined path and on the information acquired from its sensors.

### 1.3.3. The Smart Tether system

The smart tether system is an accessory developed for the YAPE robot, integrated into it and housed within a box located in Yape's payload compartment, which is originally intended for package transport. The smart tether system (in Figure 1.8 we show the first prototype) serves as a physical link between the robot and the visually impaired individual. Its structure presents a tether, with both an elastic and a rigid part.

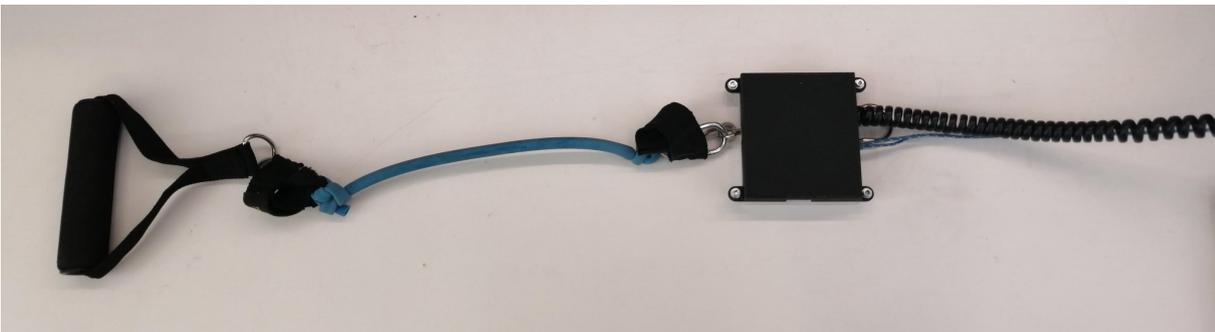


Figure 1.8: Smart tether from left to right: handle, elastic rope, load cell, non-elastic rope

The former aims to maintain a consistent force exerted on the user. This force is measured by a full-bridge load cell, located between the elastic and the non-elastic rope, as shown in Figure 1.8. At the same time, the rigid rope is directly attached to the winch and coils and uncoils on it, based on the angular velocity produced by the motor's rotation and the consequent rotation of the shaft, intending to adjust the distance based on the user's movement and speed. At the end of the elastic tether, a handle is positioned for a secure and comfortable grip for the user. The load is supported by bearings with fixed ends and

aluminum supports, whose material ensures lightness, compactness, and durability, and whose dimensions allow it to fit inside the internal compartment of Yape. In the next chapters, different setups will be described and compared through experimental validation tests.

### 1.3.4. The Control System

The objective of the overall control system is to safely guide a visually impaired user to a specific destination and, at the same time, to avoid moving obstacles along the path. BUDD-e can be roughly represented as a control system with three degrees of freedom and three main controllers.

For a smooth behavior and to ensure the safety of the user, a dedicated distance controller regulates the longitudinal dynamics of BUDD-e, maintaining the distance  $d$  from the user as close as possible to a reference one  $d_0$  and adapting its speed  $v$  to the user's velocity  $v_{BVI}$ . The control input used is the reference longitudinal velocity of Yape  $v_{ref}$ . This module needs to be fed with a measure, or estimate, of the robot-user distance.

Concerning the smart tether, the objective of the control system is to maintain the force exerted on the user constant and reject the disturbance, identified as the relative velocity between Yape and the user. Various controllers have been developed, depending on the configuration of the smart tether.

Ultimately, a complete autonomous driving stack must be implemented on the vehicle, encompassing localization, perception, and global and local planning algorithms. The latter module is especially important, as it aims to control the longitudinal and lateral dynamics of BUDD-e to follow a predefined trajectory and avoid unexpected obstacles.

### 1.3.5. Tests and suggested improvements

The BUDD-e robot was tested at the ASTT Grande Ospedale Metropolitano Niguarda, as presented in [14]. During these tests, some volunteers (either totally blind or visually impaired) allowed us to evaluate the system's functionality and usability.

During these tests, the participants completed a predefined route within the hospital, and their experiences were documented. BUDD-e successfully guided volunteers through the hospital, including complex spaces like narrow doorways and crowded areas. Data collected during the trials showed that BUDD-e could maintain a consistent distance from users and adapt to their walking speed.



(a) Inside pathway



(b) Outside pathway

Figure 1.9

Feedbacks from the volunteers were collected from these tests. They appreciated that BUDD-e enabled them to navigate unfamiliar places easily and walk at higher speeds. Volunteers found BUDD-e more useful than a white cane in many situations and less distracting than a guide dog. The system's ability to provide clear and understandable route indications was also positively noted.

However, several areas for improvement were highlighted. A general overview of the suggested improvements can be found in Table 1.1 [14]. Among other things, the user felt insecurity caused by the cable due to the load cell sensor in the middle of it and highlighted the necessity to better adapt to the users' velocity and to make less abrupt stops. These two issues will be addressed in Parts I and II, respectively.

<b>Audio</b>	<ul style="list-style-type: none"> <li>• Raise the volume of vocal indications</li> <li>• Interact with BUDD-e through vocal commands</li> <li>• Make BUDD-e inform the user about their location</li> <li>• Communicate the distance or time that remains to reach the destination</li> <li>• Alert earlier about the reaching of the destination</li> </ul>
<b>Unforeseen circumstances</b>	<ul style="list-style-type: none"> <li>• Create a button to contact the staff in case of problems</li> <li>• Modify the route as the user prefers</li> </ul>
<b>Robot Guide</b>	<ul style="list-style-type: none"> <li>• Reduce the dimensions</li> <li>• Add a keyboard to give commands</li> </ul>
<b>Smart Tether System</b>	<ul style="list-style-type: none"> <li>• Give a major feedback during curves</li> <li>• Adjust the reference force according to the user's preferences</li> <li>• Change the handle to a more comfortable one</li> <li>• Use a more rigid cable</li> </ul>
<b>Walking</b>	<ul style="list-style-type: none"> <li>• Improve the feature of BUDD-e to adapt to user velocity</li> <li>• Make the stops less aggressive</li> </ul>
<b>Visual impairment</b>	<ul style="list-style-type: none"> <li>• Adapt BUDD-e's usability based on the level of visual impairment</li> </ul>

Table 1.1: Possible improvements

### 1.3.6. Open issues

While it is acknowledged that the BUDD-e prototype is a valuable and promising solution to mobility concerns, as clear from Table 1.1, it presents challenges related to reactivity and adaptability, leaving room for improvements in supporting visually impaired individuals in diverse scenarios.

Some issues observed in the experimental tests carried out on the BUDD-e first prototype prompted our focus on enhancing reactivity. Reactivity is a pivotal factor, particularly in dynamic environments, such as sports activities. In fact, inaccuracies or delays in the system's response could compromise user safety and confidence.

In our pursuit of an improved and more reactive robotic guide, we first recognized the need to optimize the smart tether system configuration for practicality and user comfort. Previous designs exhibited undesirable oscillations and discomfort due to the load cell's placement on the cable, particularly in sports environments. Through careful analysis, adjustments were made to the smart tether system configuration to eliminate these issues, ensuring a practical and comfortable user experience in a variety of scenarios.

A critical aspect of reactivity lies also in the optimality of the distance controller. To achieve a responsive and adaptable robotic guide, a new controller needs to be devised to ensure more reactive and precise control of the position of the robot with respect to the user. In addition, a serious issue was observed when the user experiences a substantial change in speed, such as a sudden stop while running. In this case, Yape executes a smooth stop, halting at a greater distance from the user, leading to potential damages to the smart tether system, which physically connects the robot and the user.

To face this issue we need to introduce a novel additive component to the distance control system, allowing to promptly detect the user brake and endowed with a feedforward brake controller that takes over when necessary.

These enhancements not only address the limitations of previous versions but also expand the potential applications of the system, including participation in sports activities where rapid changes in movement are common.

Through a meticulous focus on reactivity, braking control, and load cell configuration, our work contributes to the ongoing effort to enhance the quality of life for visually impaired individuals.

## 1.4. Thesis contributions

The main contributions of the thesis are the following.

- Design of a new configuration of the smart tether system to relocate the load cell inside the vehicle, eliminating oscillations of the external cell.
- Development and implementation of a controller based on  $\mathcal{H}_2$  technique, integrated with a state observer, to enhance the smart tether system performance.
- Development of a new smart tether system leveraging on a better electric motor and the dedicated controller to overcome the limitations of the motor used in the previous configuration.
- Development and implementation of an optimal time-invariant distance control law with LMI arguments, robust with respect to the specific mode of operation and of the switching between them.
- Design of a dedicated algorithm that utilizes multivariate analysis to detect the instant when a pedestrian intends to stop, thereby enhancing the control system's responsiveness.

In line with the abovementioned contributions, the thesis is structured as follows.

- The structure and components of the first Smart Tether setup are illustrated in Chapter 2.
- In Chapter 3, the Smart Tether model is identified, and the design of the frequency domain controller (Configuration 1) and the  $\mathcal{H}_2$  controller (Configuration 2) are described.
- The structure of the new setup of the Smart Tether and its controller are illustrated in Chapter 4.
- In Chapter 5, the performance of different Smart Tether system configurations is compared through experimental.
- Chapter 6 discusses the design and implementation of the distance feedback control system.
- The development of a braking control strategy for the Smart Tether is presented in Chapter 7.
- The conclusions of the thesis are outlined in Chapter 8.



## Part I

# New design of the smart tether system



## 2 | System Structure and Components of the first setup

### 2.1. Original setup

The components of the original setup of the smart tether system include a 12 VDC brushed motor, a Roboclaw 2x45A motor driver, a 5kg full-bridge load cell, an HX711 amplifier, and a proximity sensor, all controlled by an Arduino board.

Additionally, the aluminum support structures have been specifically designed to fit the dimensions of the Yape inner compartment, ensuring proper alignment of the motor shaft and load shaft. The overall measures of the system are 30 cm in length, 20 cm in depth, and 22 cm in height, with a total weight of around 6 kg.

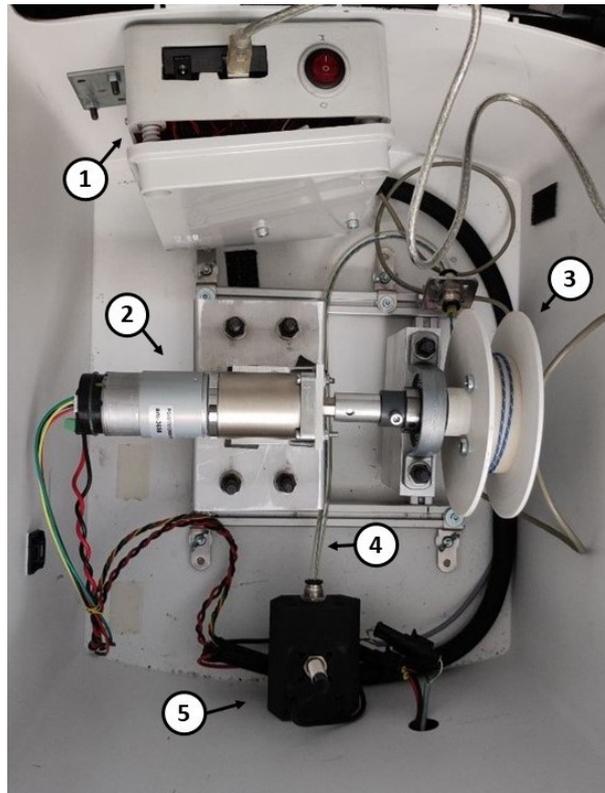


Figure 2.1: The Smart Tether System: 1. Box with electronic components; 2. Motor; 3. Shaft; 4. Polyurethane tube; 5. Proximity sensor.

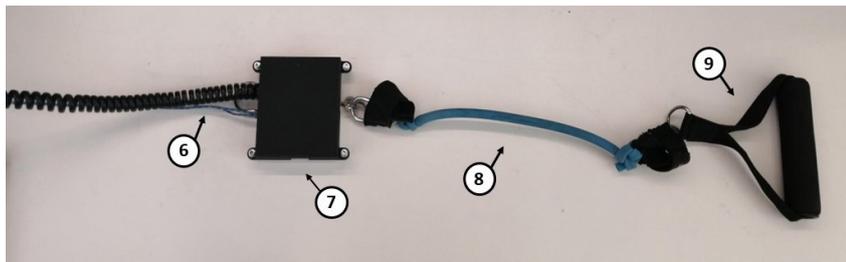


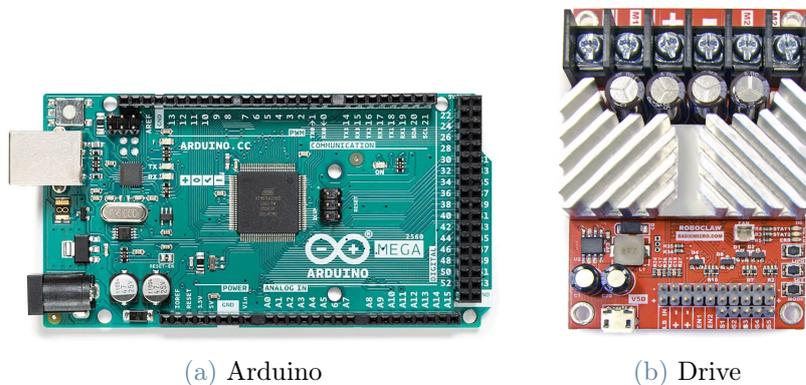
Figure 2.2: The Smart Tether System: Smart tether system cable: 6. Rigid cable; 7. Load cell; 8. Elastic cable; 9. Handle.

The motor selection was guided by specific requirements, including the ability to measure position and speed, and compatibility with the Yape battery. The motor specifications needed were a nominal torque of 6-7 Nm and a speed of 100 rpm. To fulfill these requirements, a 12 VDC brushed motor illustrated in Figure 2.3 was chosen, equipped with a 188:1 ratio gearbox and a 5 Volt DC encoder.



Figure 2.3: Motor, gearbox and encoder

For controlling the motor, the Roboclaw 2x45A driver (Figure 2.4b) was selected. This driver provides precise control in position, torque, and speed for two motors, supporting two encoder channels per motor. It offers a peak current of 60 A and a continuous current of 45 A per channel, with a voltage capacity of up to 34 VDC. Additionally, control can be managed using a microcontroller like Arduino (Figure 4.9a).



(a) Arduino

(b) Drive

Figure 2.4: Arduino and Drive

A 5kg full-bridge load cell was selected to measure the exerted force.

To read the measures of the full-bridge load cells, the HX711 amplifier (Figure 4.11) is used, together with the abovementioned Arduino board.

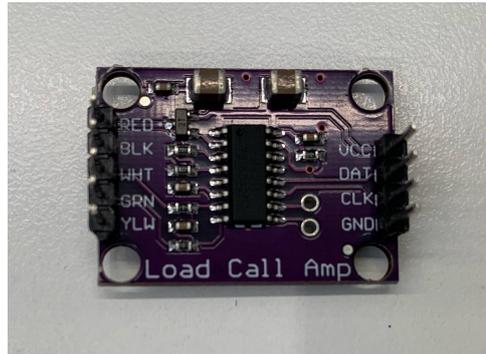


Figure 2.5: HX711 amplifier

The proximity sensor (Figure 2.6) has the fundamental function of identifying a particular segment of the cable wrapped with a metallic wire, which allows relaying exact position data to the encoder, avoiding possible calculation errors. The entire system's logic is managed by an Arduino board (Figure 4.9a), which is installed within a designated white enclosure along with the necessary cables and connections.



(a) Proximity sensor



(b) Proximity sensor in its enclosure

Figure 2.6

Finally, a DC-DC converter, shown in Figure 2.7, has been selected to integrate the smart tether into Yape. This step-down converter reduces the voltage from 24 V to 12 V with a maximum current of 10 A, allowing all the previous components to be directly powered by one of Yape's batteries.

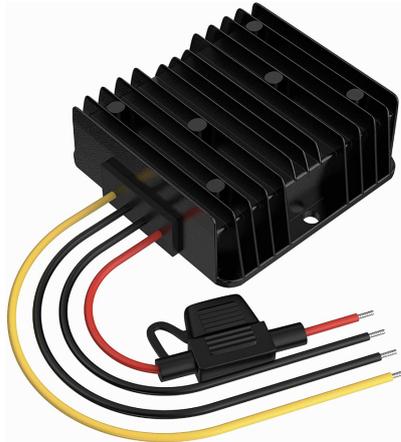


Figure 2.7: DC-DC converter

Key improvements involve a redesigned winch system and the load cell setup, for better stability and comfort of the tether.

The new overall setup is shown in Figure 2.8.

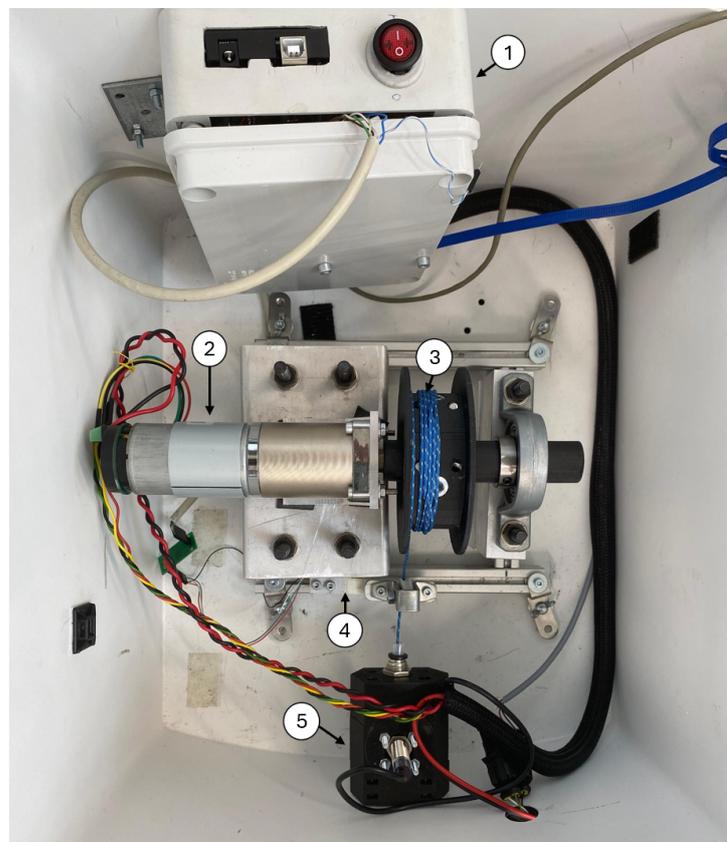


Figure 2.8: The smart tether system: 1.Box with electronic components; 2.Motor; 3.Winch; 4.Load cell; 5.Proximity sensor

## 2.2. New winch setup

Adjustments with respect to the initial version of the smart tether system ([12],[11],[14]) have been made to the winch, in terms of dimension and positioning. Firstly, we reduced its radius and its width by fixing it to the shaft internally to create a more compact system. Then the reel was placed between the motor and the bearing, to straightly pass the rigid cable from the winch to the proximity sensor, going through the little winch connected to the load cell.

Moreover, the shaft has been redesigned and reduced in dimension, resulting in a lighter and more compact structure. The CAD models of these components are shown in Figures 2.9 and 4.7.

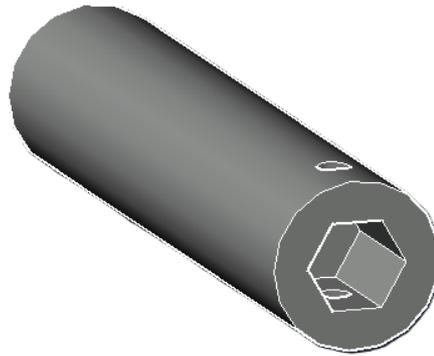


Figure 2.9: Shaft

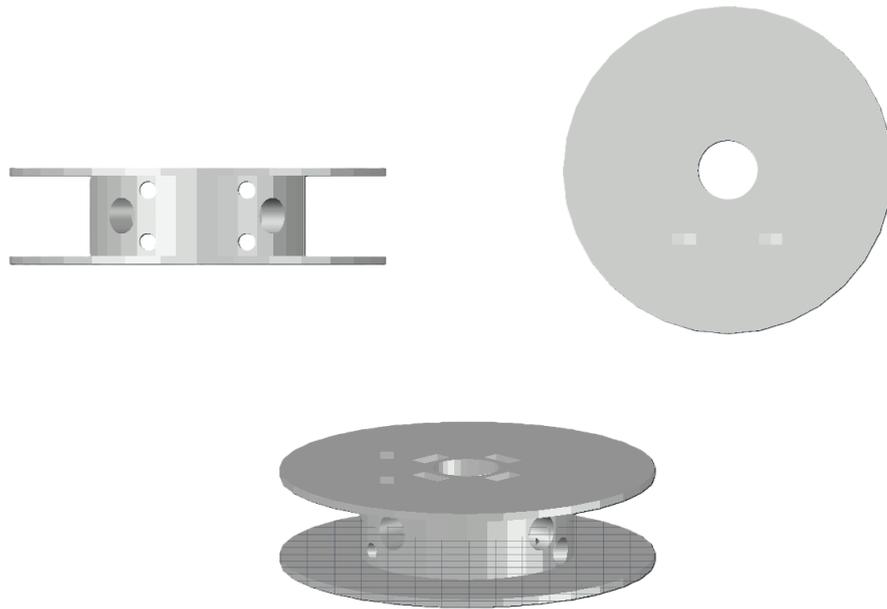


Figure 2.10: Winch with internal fixing mechanism

### 2.3. New load cell setup

In order to respond to the concerns raised during the tests conducted in ASTT Grande Ospedale Metropolitano Niguarda, illustrated in Section 1.3.5, we decided to face the enhancement of the security feeling of the visually impaired individuals, first by improving the comfort and the stability of the cable. To this aim, we have positioned the 5-kg load cell in the Yape compartment, downstream the reel.

This new configuration, shown in Figure 2.13, allows to remove the load cell sensor placed between the non-elastic and elastic sections of the tether, as shown in Figure 2.11b.

This change reduces the risk of sensor breakage and simultaneously makes the tether lighter and avoids fluctuations.

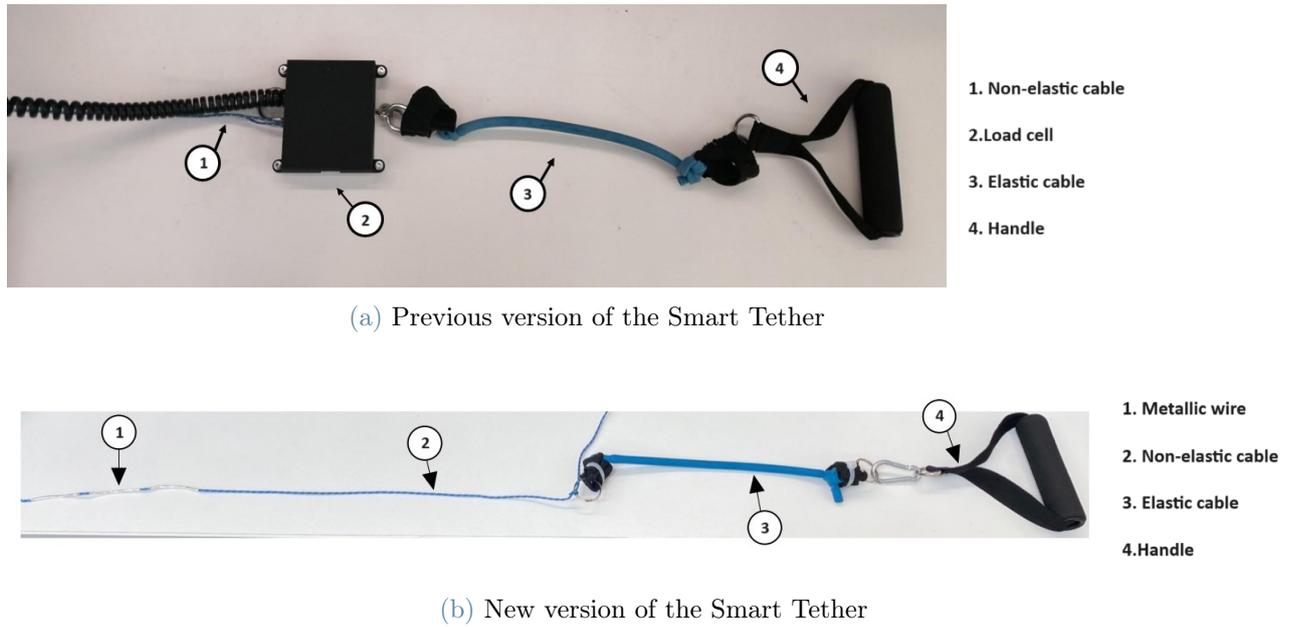


Figure 2.11

In the new configuration, as illustrated in Figure 2.14, the rigid rope uncoils from the winch and passes under a U-groove plastic bearing, with a diameter of 24mm and a width of 7mm, shown in Figure 2.12.

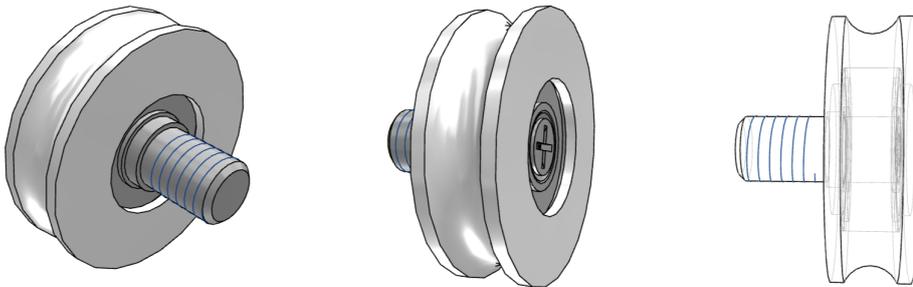


Figure 2.12: U-groove plastic bearing (diameter 24mm)



Figure 2.13: Winch, support structure and load cell

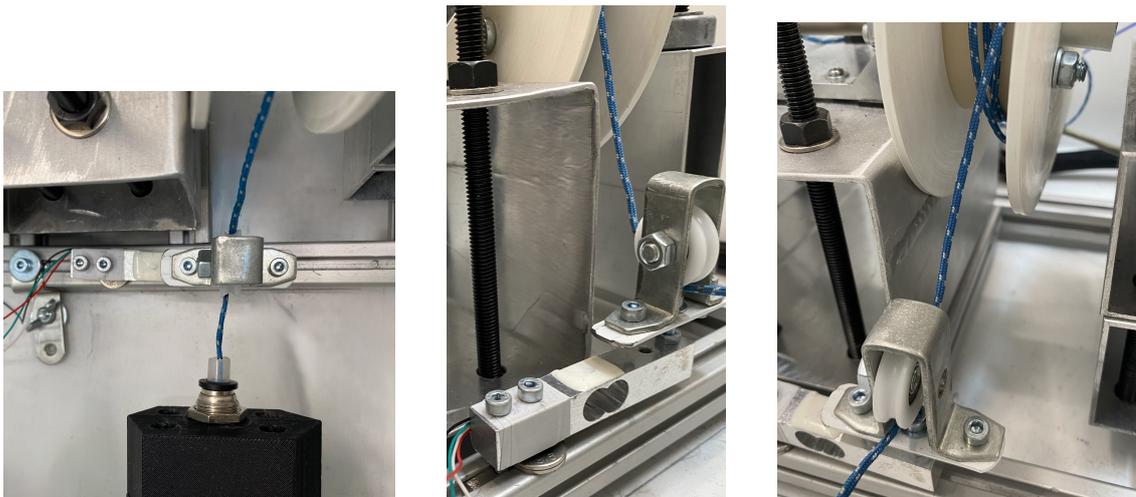


Figure 2.14: Winch configuration

The bearing is mounted on a structure designed for this purpose and fixed to a load cell through a screw. The same load cell sensor is attached to the extrusions that firmly secure the entire smart tether system to the interior compartment of Yape. The pressure exerted by the rigid rope on the lower part of the bearing causes the load cell to deform vertically and thus measure this deformation and convert it into a force measurement. The force exerted by the rigid rope is primarily along the vertical direction due to the steep inclination of the rope. In this way, the other dimensional force components (horizontal or lateral forces) are negligible and do not significantly affect the measurement. The cable is more comfortable and stable, as suggested in [14].

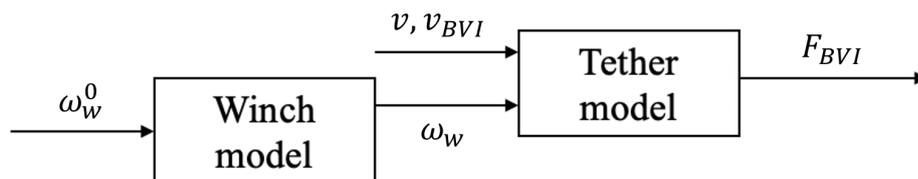


# 3 | Control of the first setup

In this section, we aim to design a controller that can maintain a constant force exerted on the cable. By doing so, the user will feel a consistent tension through the handle, making it easier to follow the robot. This controller, together with the one designed in Chapter 6 that maintains a constant distance between the user and the robot under working conditions, will address two main objectives: ensuring smooth and reliable navigation.

## 3.1. Model identification

To design a controller for the smart tether system, we must undergo a system identification phase. First, note that the smart tether is controlled by the reference angular velocity of the motor ( $\omega_w^0$ ). In this section, we first devise a model that represents the motor dynamics when given a reference angular velocity  $\omega_w^0$ . Next, we will examine how the exerted force is related to the motor's real angular velocity  $\omega_w$  and possible disturbances. The system block diagram is illustrated in Figure 3.1.



**Figure 3.1:** System block diagram. The input  $v_{BVI}$  to the tether model is the user's longitudinal speed; the  $F_{BVI}$  is the force exerted on the user; the  $\omega_w$  is the actual velocity of the motor and  $\omega_w^0$  is its reference velocity.

### 3.1.1. Identification setup description

A series of laboratory experiments were conducted to identify the tether model by attaching the end of the elastic tether to a stationary point, as illustrated in Figure 3.2. Yape is kept in a standing position and held stationary.



Figure 3.2: Experimental setup

The collected data, shown in Figure 3.3, include the normalized reference velocity  $\omega_w^0$  within the range  $[-1, 1]$ , the actual velocity  $\omega_w$ , the winch phase  $\theta_w$ , and the force  $F_{BVI}$  measured using a 5 kg load cell.

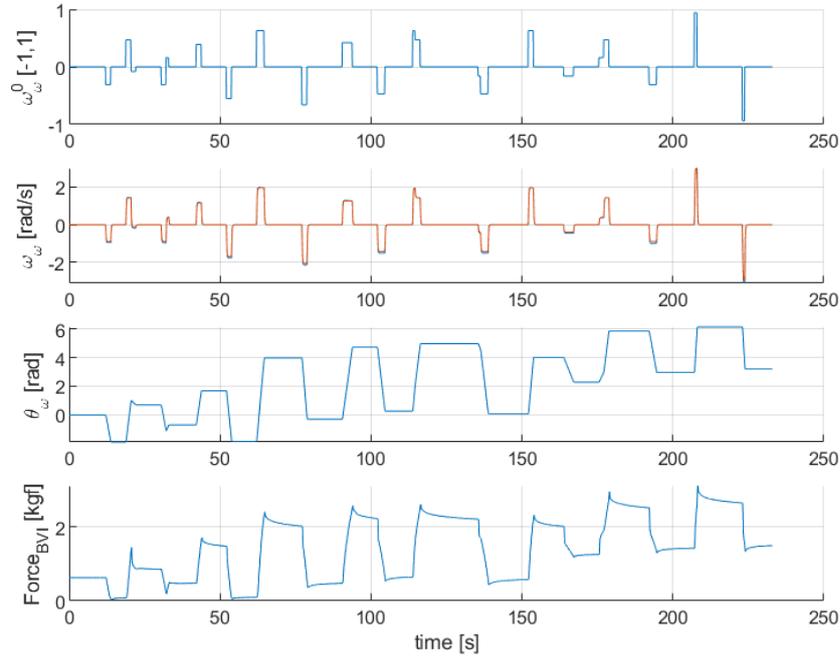


Figure 3.3: Identification data

### 3.1.2. Winch model identification

As an initial step, we examine the relationship between the reference velocity of the winch  $\omega_0^{ss}$  and the actual velocity  $\omega_{real}^{ss}$  in steady-state. Figure 3.4 shows that this relationship can be effectively modeled by a deadzone function, which is attributed to the motor's static friction and the inertia of the load. To define this deadzone, we divided the data into two regions. For each region, an affine relationship was derived to represent the positive and negative data accurately. These relationships are provided in equations (3.1) and (3.2).

$$\omega_{ss}^{real,positive} = 3.3293\omega_{ss}^{0,positive} - 0.14 \quad (3.1)$$

$$\omega_{ss}^{real,negative} = 3.3789\omega_{ss}^{0,negative} + 0.0916 \quad (3.2)$$

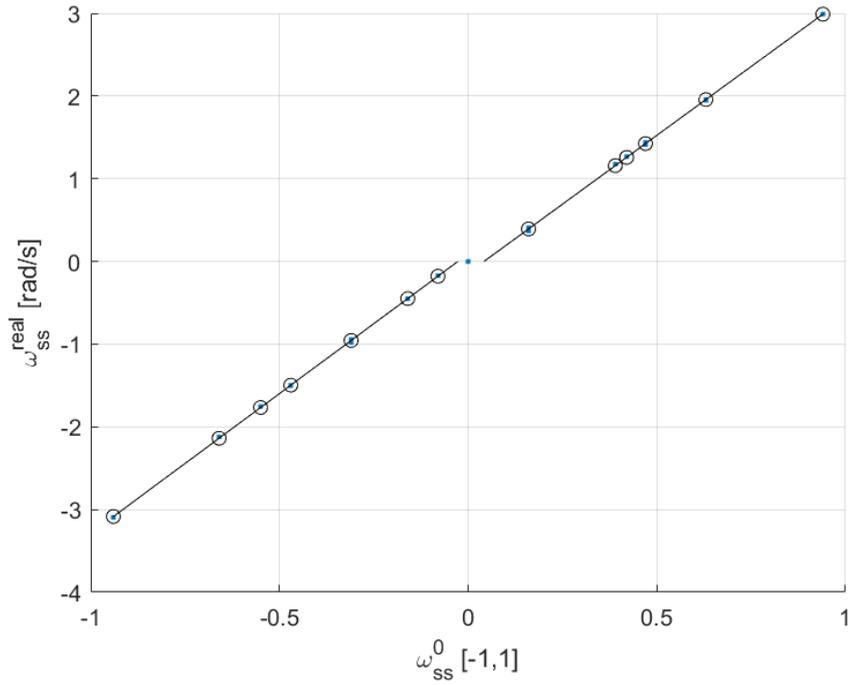


Figure 3.4: Identification of the deadzone

The necessity to relate the reference velocity after the deadzone  $\omega_w^{0,DZ}$  to the actual velocity  $\omega_w$  (see Figure 3.5) leads to the identification of a second-order transfer function, shown in equation (3.3), which effectively captures the dynamics between  $\omega_w^{0,DZ}$  and  $\omega_w$ .

$$F(s) = \frac{1}{(1 + \tau_w s)^2} \quad (3.3)$$

where  $\tau_w = 0.0911$  s.

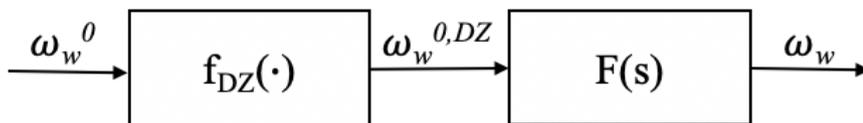


Figure 3.5: Transfer function block diagram between  $\omega_w^{0,DZ}$  and  $\omega_w$

### 3.1.3. Tether model identification

The data presented in Figure 3.3 are used also to identify the elastic tether model. Our aim is to find the relationship between the extension of the elastic tether  $\Delta L_T$  and the force exerted on the user  $F_{BVI}$ .

In case no disturbances act on the system, the extension  $\Delta L_T$ , in one setup, can be written as:

$$\Delta L_T = \Delta L_T^{0.6} + \theta_w R_w \quad (3.4)$$

where  $R_w$  is the radius of the winch and  $\Delta L_T^{0.6}$  is the extension of the tether when  $F_{BVI} = 0.6$  kgf.

The first step consists of identifying the steady-state relationship between  $\theta_w$  and  $F_{BVI}$ , defined in equation (3.5), i.e.,

$$F_{BVI}^{SS} = K_T^{SS}(\theta_w^{SS}) + 0.6 \quad (3.5)$$

To determine the function  $K_T^{SS}(\cdot)$ , a least-square approach is employed. Firstly, a linear polynomial regression (equation (3.6)) is identified, i.e.,

$$K_T^{SS}(\theta_w^{SS}) = \alpha_1 \theta_w^{SS} \quad (3.6)$$

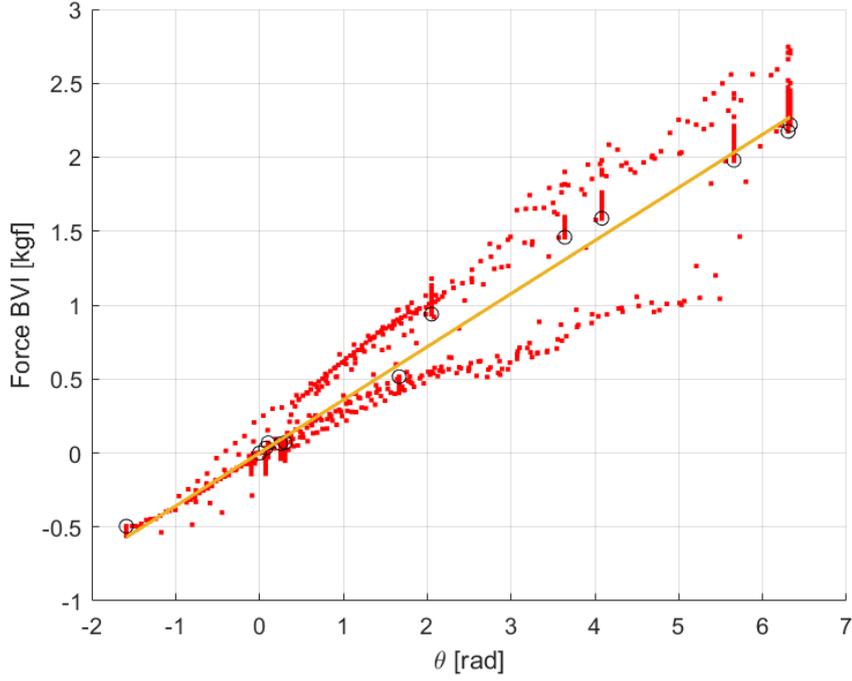


Figure 3.6: Plot of both the dynamic data (red dots) and the steady state data (black circles) for  $\theta_w - F_{BVI}$ . The estimated model with the polynomial regression method of order 1 is in yellow.

Since these results are not very satisfactory and representative, the model is identified with polynomial regression from the second order up to the fourth order, shown in Figure 3.7 and in equations (3.7), (3.8), and (3.9).

$$K_T^{SS}(\theta_w^{SS}) = \alpha_1 \theta_w^{SS} + \alpha_2 \theta_w^{SS2} \quad (3.7)$$

$$K_T^{SS}(\theta_w^{SS}) = \alpha_1 \theta_w^{SS} + \alpha_2 \theta_w^{SS2} + \alpha_3 \theta_w^{SS3} \quad (3.8)$$

$$K_T^{SS}(\theta_w^{SS}) = \alpha_1 \theta_w^{SS} + \alpha_2 \theta_w^{SS2} + \alpha_3 \theta_w^{SS3} + \alpha_4 \theta_w^{SS4} \quad (3.9)$$

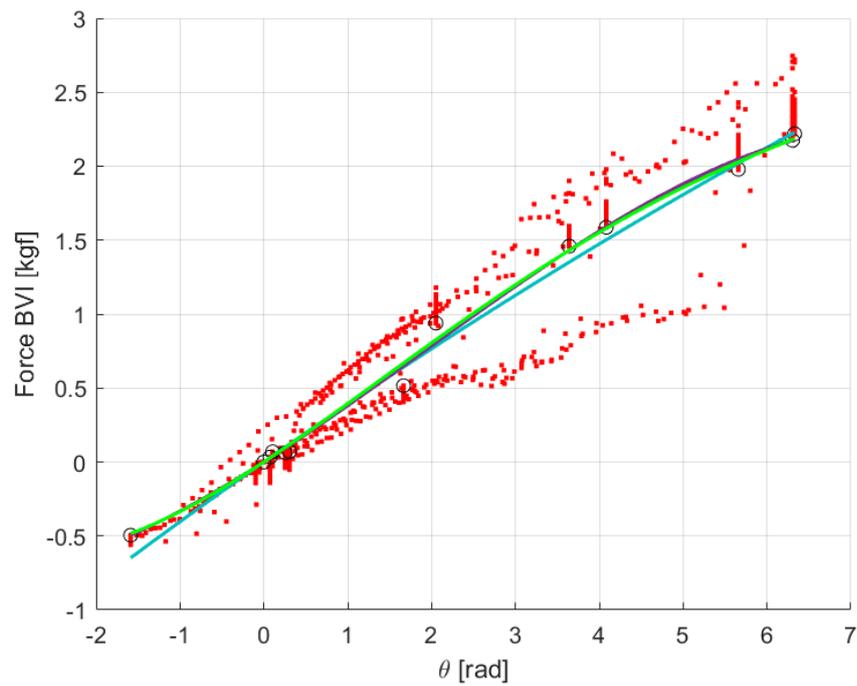


Figure 3.7: Plot of both the dynamic data (red dots) and the steady state data (black circles) for  $\theta_w - F_{BVI}$ . The estimated model with the polynomial regression method of order 2 is in blue, order 3 in purple, and order 4 in green.

Figure 3.8 provides a comparison between the real data and the trajectories generated by the four identified static models of different orders.

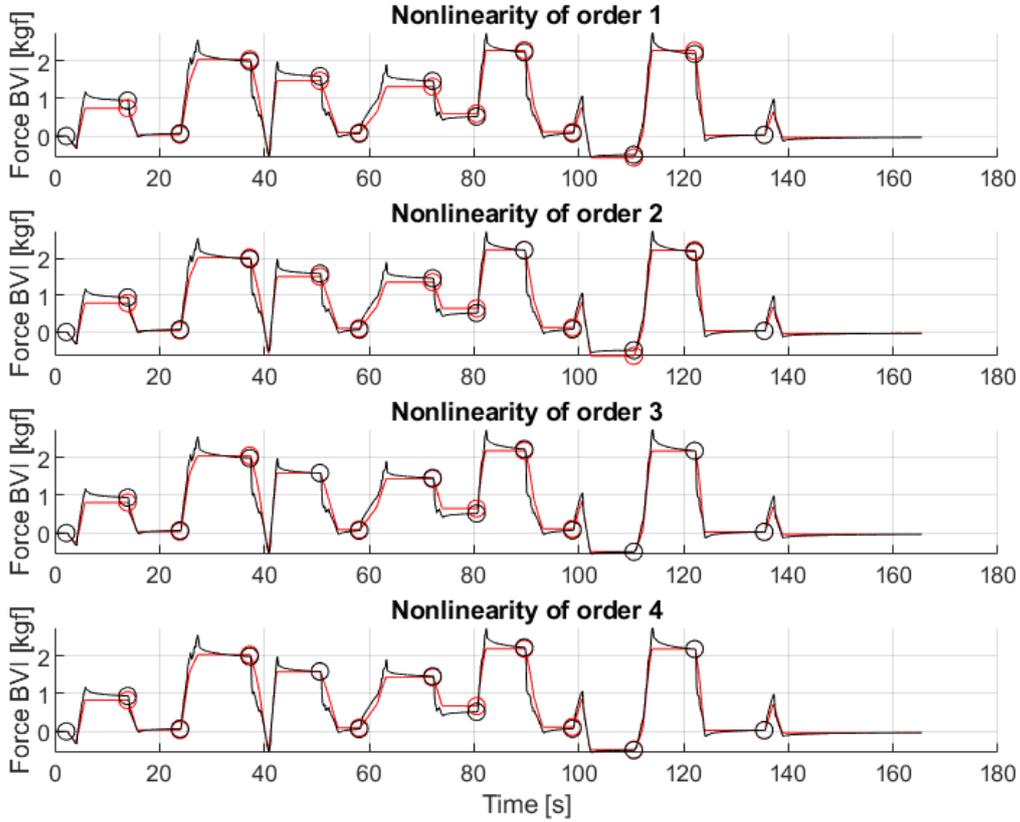


Figure 3.8: Comparison between the real data and the trajectory generated by the tether model.

Black line:  $F_{BVI}$ ; red line: data generated by the identified polynomial regression models; From the top to the bottom: polynomial regression model of first order; polynomial regression model of second order; polynomial regression model of third order; polynomial regression model of fourth order.

The models of orders 3 and 4 show an excellent steady-state agreement.

To implement the subsequent control strategy, the invertibility of the model is required. As a result, the third-order nonlinearity must be discarded, as it is not invertible within the range of our interest. Consequently, the fourth-order nonlinearity is the best option to be considered.

We decided at this point to improve the reconstruction during the transient by testing both a Wiener model and a Hammerstein model (see Figure 3.9).

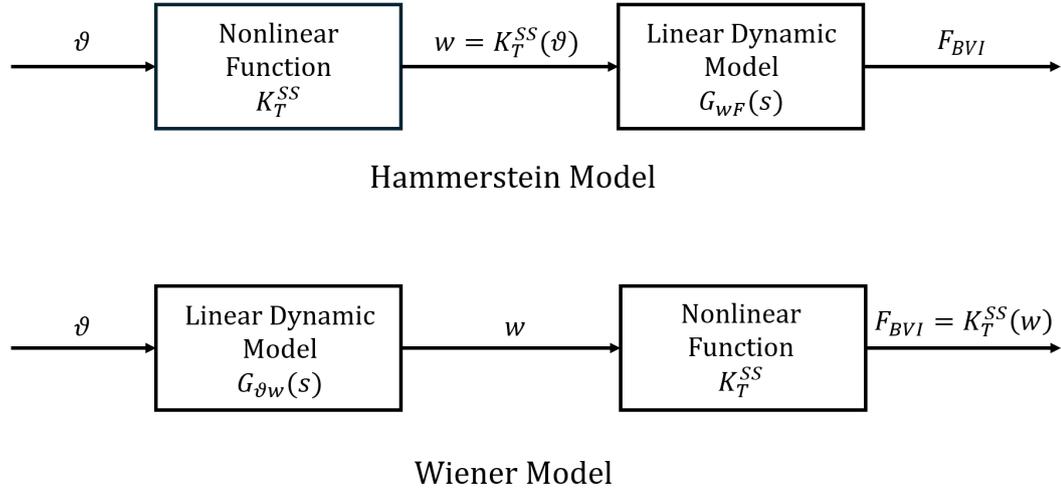


Figure 3.9: Block schemes of Hammerstein model and Wiener model

The linear transfer functions for both the Hammerstein model,  $G_{wF}(s)$ , presented in equation (3.10), and the Wiener model,  $G_{\vartheta w}(s)$ , which is shown in equation (3.11), are therefore identified. Note that the second-order choice for the linear transfer function is essential to accurately replicate the dynamic characteristics of the model. We obtained

$$G_{wF}(s) = K \frac{(s + T_{z1})(s + T_{z2})}{(s + T_{p1})(s + T_{p2})} \quad (3.10)$$

where:

$$K = 2.1306$$

$$T_{z1} = 55.05$$

$$T_{z2} = 2.984$$

$$T_{p1} = 98.55$$

$$T_{p2} = 3.607$$

$$G_{\vartheta w}(s) = K \frac{(s + T_{z1})(s + T_{z2})}{(s + T_{p1})(s + T_{p2})} \quad (3.11)$$

where:

$$K = 8.7372;$$

$$T_{z1} = 35.85$$

$$T_{z2} = 3.239$$

$$T_{p1} = 215.4$$

$$T_{p2} = 3.854$$

The simulated outputs obtained with the abovementioned identified models are compared with the actual data in Figures 3.10 and 3.11.

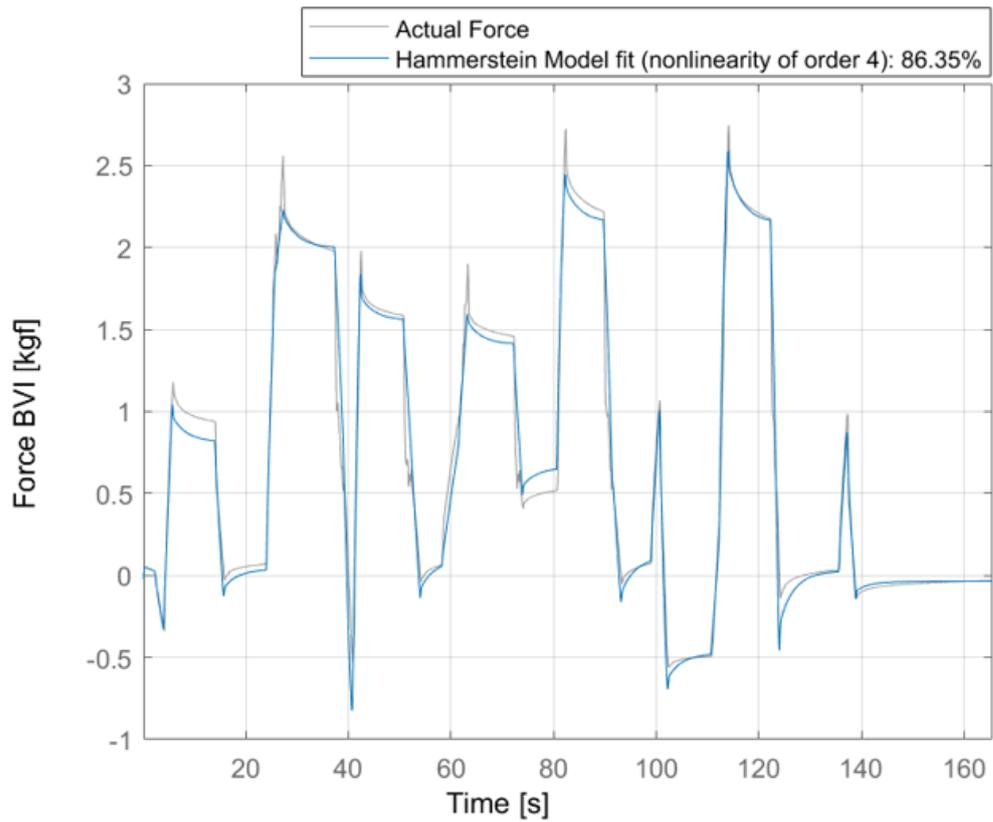


Figure 3.10: Comparison between the real data and the trajectory generated by the Hammerstein model of the tether with the fourth-order nonlinearity.

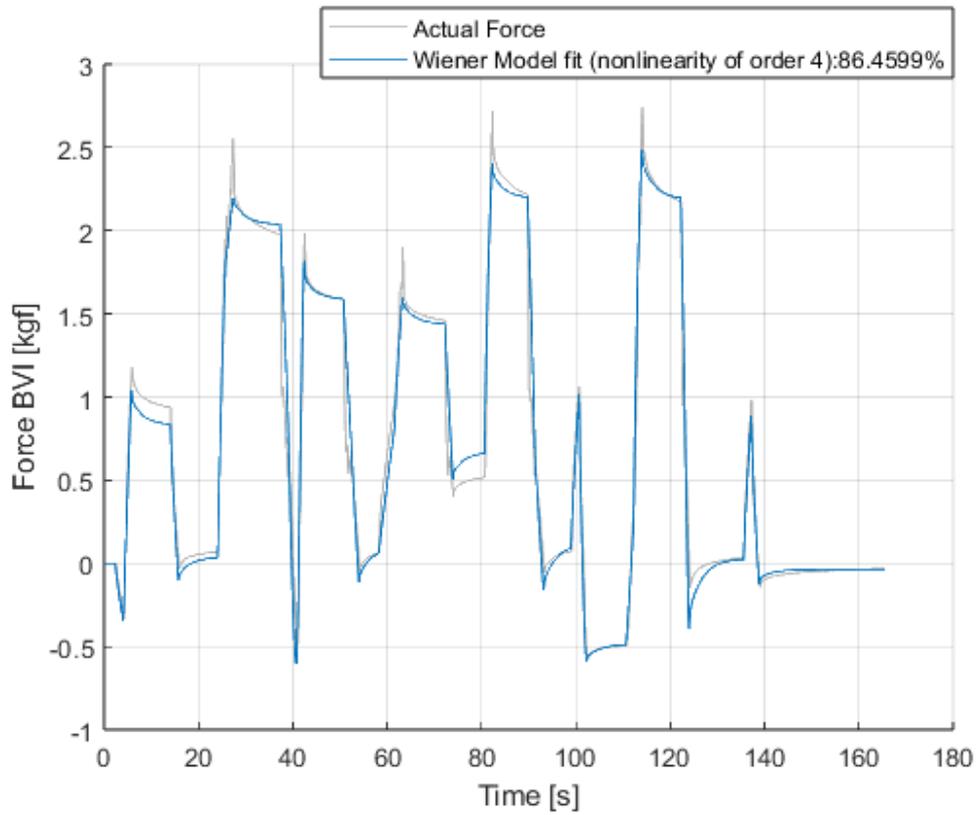


Figure 3.11: Comparison between the real data and the trajectory generated by the Wiener model of the tether with the fourth-order nonlinearity.

#### 3.1.4. Model validation

The overall smart tether models identified in Sections 3.1.2 and 3.1.3 were validated and compared using a new data set (see Figure 3.12). The outcomes concerning the winch model are satisfactory since they exhibit the same behavior observed in the identification test. This consistency is demonstrated in Figure 3.13.

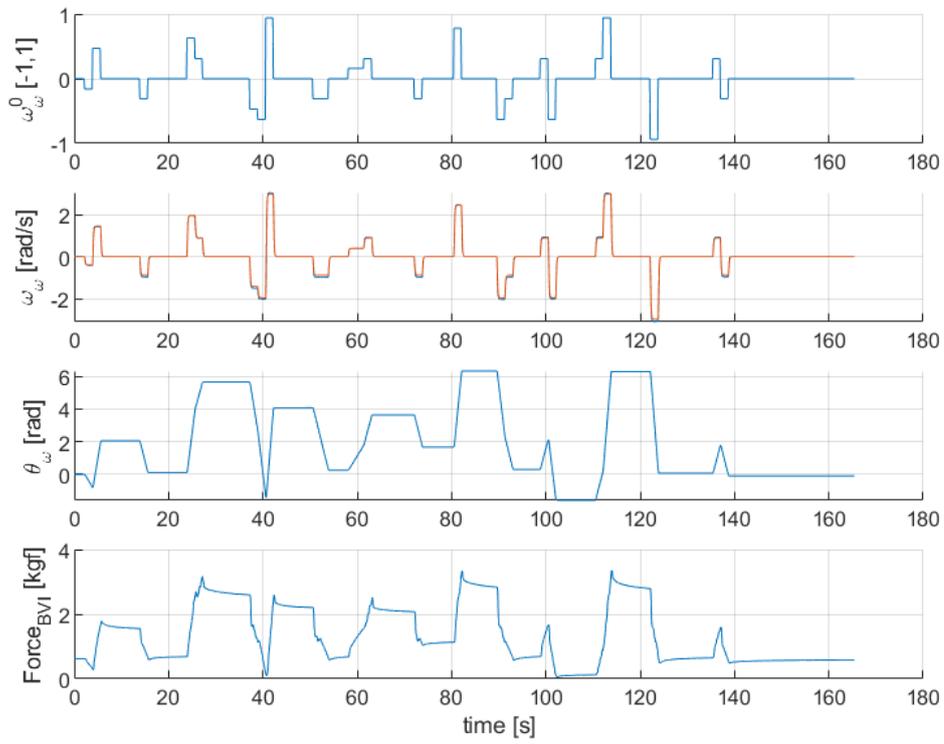


Figure 3.12: Validation data

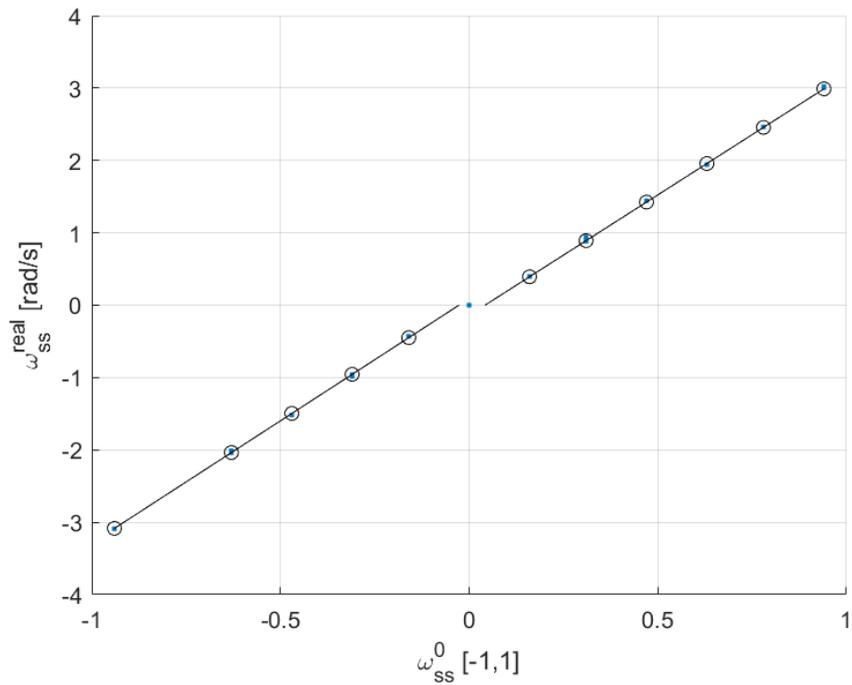


Figure 3.13: Validation of the deadzone model

Concerning the tether model validation, the outcomes are adequate, as shown in Figures 3.14 and 3.15.

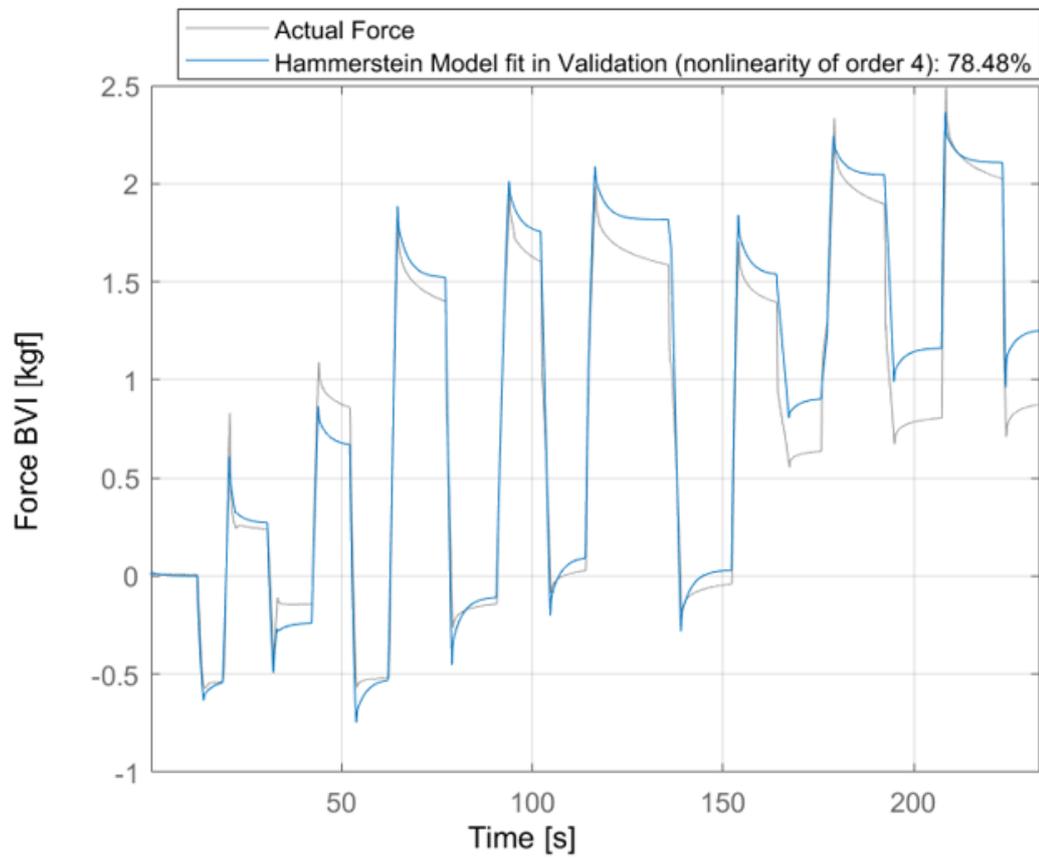


Figure 3.14: Comparison in validation between the real data and the trajectory generated by the Hammerstein model of the tether with the fourth-order nonlinearity.

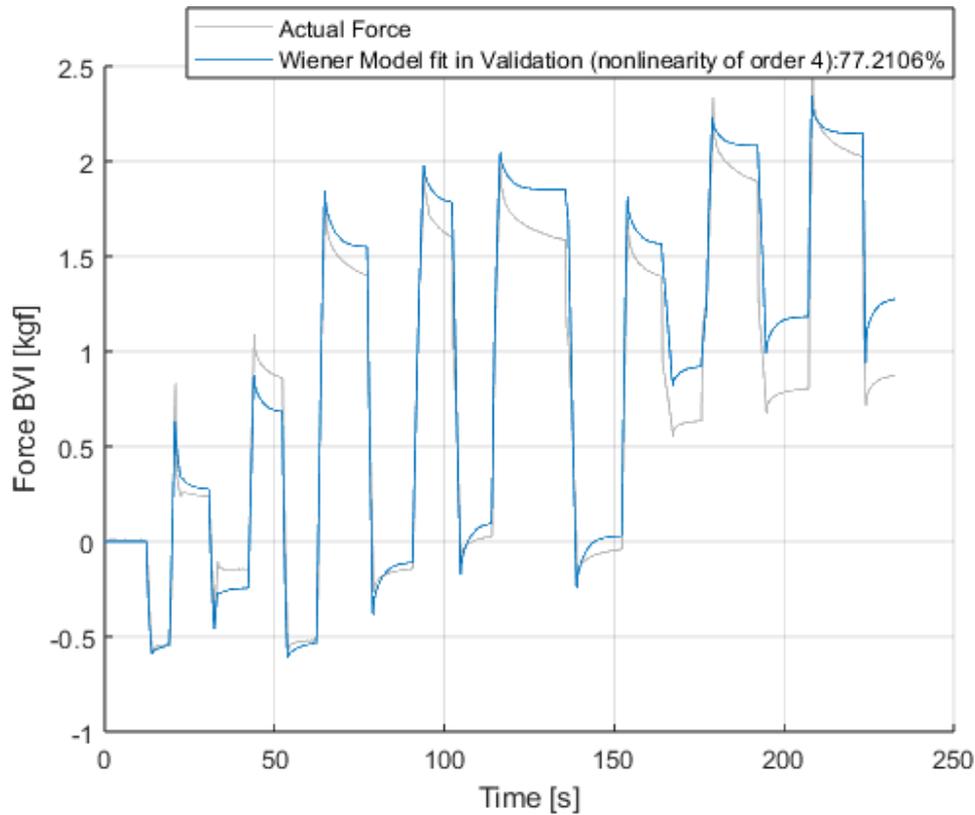


Figure 3.15: Comparison in validation between the real data and the trajectory generated by the Wiener model of the tether with the fourth-order nonlinearity.

Despite the similar behavior, when comparing the real data with the trajectories generated by the Hammerstein model and the Wiener model, the latter is preferred because it approximates the model's dynamics slightly better.

### 3.1.5. The comprehensive identified model of the smart tether system

To summarize, the overall selected model has the general structure described in Figure 3.16.

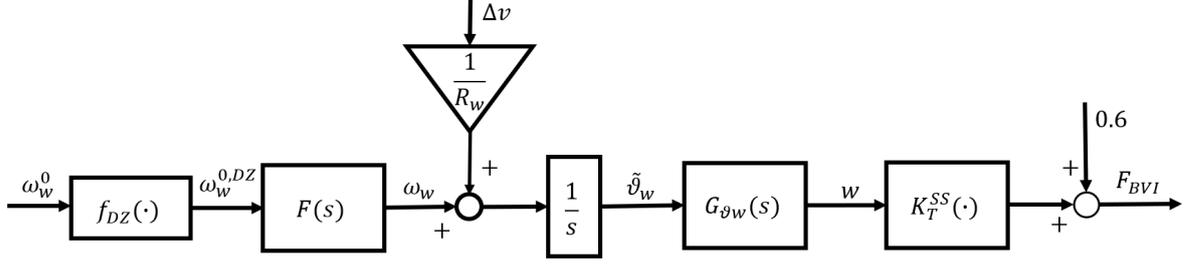


Figure 3.16: Smart Tether system;  $F(s)$  is the transfer function identified for the winch, while  $G_{\theta w}$  is the transfer function identified for the tether.

In the real setup, the user's longitudinal speed  $v_{BVI}$  and Yape's velocity  $v$  act as external disturbances. Indeed, we consider the actual disturbance as the difference  $\Delta v = v - v_{BVI}$ . This disturbance affects the tether elongation  $\Delta L_T$  as shown in Figure 3.16.

In case disturbances are absent, the elongation is defined in equation (3.4). In case disturbances are present, we define the difference of the actual winch phase with respect to the one providing  $F_{BVI} = 0.6\text{kgf}$  as

$$\tilde{\theta}_w = \frac{\Delta L_T - \Delta L_T^{0.6}}{R_w} \quad (3.12)$$

Our control objective is to maintain the elongation of the elastic tether  $\Delta L_T$  at  $\Delta L_T^{0.6}$ , ensuring a consistent force  $F_{BVI}$  exerted on the user. This, in turn, amounts to keeping  $\tilde{\theta}_w$  at zero.

Moreover, it is important to note that  $K_T^{SS}(\cdot)$  represents a static relationship between  $\tilde{\theta}_w$  and  $F_{BVI}$ . In real conditions, this relationship may present criticisms. In the following sections, for control purpose, it will be needed to invert the function  $K_T^{SS}(\cdot)$  (equation (3.9)) to translate the force measurements into angular positions, to express the variation in  $\tilde{\theta}_w$  as a function of the variation in  $F_{BVI}$ . Given equation (3.5), it is possible to linearize  $K_T^{SS}(\cdot)$  by deriving it with respect to  $\theta_w$ . We linearize  $K_T^{SS}(\cdot)$  around the steady state and define

$$\nabla K_T^{SS}(0) = \left. \frac{\partial K_T^{SS}(\theta_w)}{\partial \theta_w} \right|_{\theta_w=0} \quad (3.13)$$

$$\delta = \nabla K^{ss}(0)^{-1} = 2.6577 \quad (3.14)$$

### 3.2. Frequency domain controller (Configuration 1)

In this section, we illustrate the development of a simple smart tether system baseline controller obtained using frequency domain arguments. The design of this controller was conducted starting from the analysis of the Bode plot of the transfer function  $G(s)$ , displayed in Figure 3.17. The transfer function  $G(s)$  considers the linear part of the model shown in Figure 3.16 and is expressed as follows:

$$G(s) = \frac{1}{s} \cdot F(s) \cdot G_{\theta_w}(s) = \frac{389.59(s + 40.77)(s + 3.13)}{s(s + 68.84)(s + 13.73)^2(s + 3.814)} \quad (3.15)$$

where  $F(s)$  is provided in equation (3.3) and  $G_{\theta_w}(s)$  is given in equation (3.11). The latter transfer functions have been previously identified and discussed in detail in Sections 3.1.2 and 3.1.3, respectively.

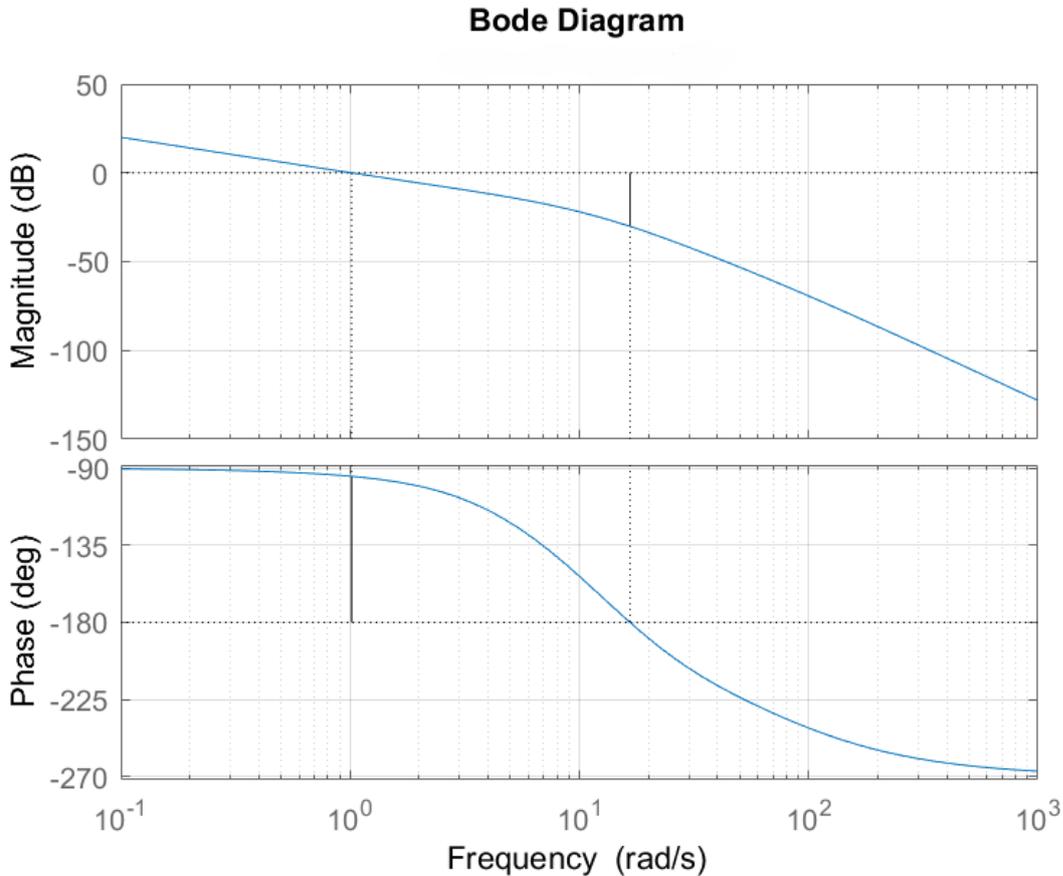


Figure 3.17: Bode plot of  $G(s)$

The initial step in the controller design process involves analyzing the open-loop Bode

plot of  $G(s)$ . This plot provides crucial insights on the system's frequency response characteristics. These parameters are fundamental for designing a stable and robust "ideal" control system, as in Figure 3.18.

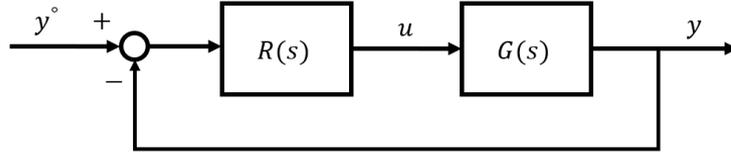


Figure 3.18: Stable and robust ideal control system

The Bode plot of  $G(s)$  (see Figure 3.17) indicates that, if we set  $R(s) = 1$ , the system has a phase margin ( $\phi_M$ ) of 85.2 degrees, a gain margin ( $G_M$ ) of 30.1 dB, and a cutoff frequency ( $\omega_C$ ) of 1.02 rad/s. These margins suggest that the system is stable with a significant buffer against gain and phase variations, ensuring robustness in the face of model uncertainties and external disturbances. In view of these considerations, a simple proportional regulator with  $R(s) = 1$  results to be a good trade-off between maintaining the system's responsiveness and keeping a high phase margin to ensure stability.

We chose not to introduce an additional integrator in  $L(s)$ . In fact, the offsets in  $v - v_{BVI}$  can be rejected by the distance controller, which is detailed in Chapter 6. Moreover, note that the current system transfer function  $G(s)$  includes an integrator to guarantee a unitary gain of the complementary sensitivity transfer function

$$T(s) = \frac{L(s)}{1 + L(s)} \quad (3.16)$$

where  $L(s) = R(s)G(s)$ .

In practical applications, the regulator is implemented in discrete time with a sampling period of  $T_s = 0.1$  seconds. The transition from continuous to discrete time introduces a phase loss, which can be calculated using the following expression:

$$\Delta\phi_M = -\omega_c \frac{T_s}{2} \frac{180}{\pi} = -2.92^\circ$$

This phase loss, caused by the discretization process, is negligible when compared to the overall phase margin of the control system, ensuring that the discrete implementation

retains the desired stability and performance characteristics.

In Figure 3.20 we show the simulation obtained with the closed-loop system, displayed in Figure 3.19, with the frequency domain proportional regulator. The disturbance, represented in gray, exhibits abrupt changes at various intervals. This disturbance is the offset  $v - v_{BVI}$  reported in angular velocity (rad/s). It is intended to test the robustness and stability of the control system. The reference force, in blue, is constant at approximately 0.6 kgf, indicating the desired force level that the controller aims to maintain.

The system is very responsive and the simulated force manages to track the reference force in a satisfactory way, with minor deviations during periods of high disturbance. This indicates that the proportional regulator is effectively compensating for the disturbances.

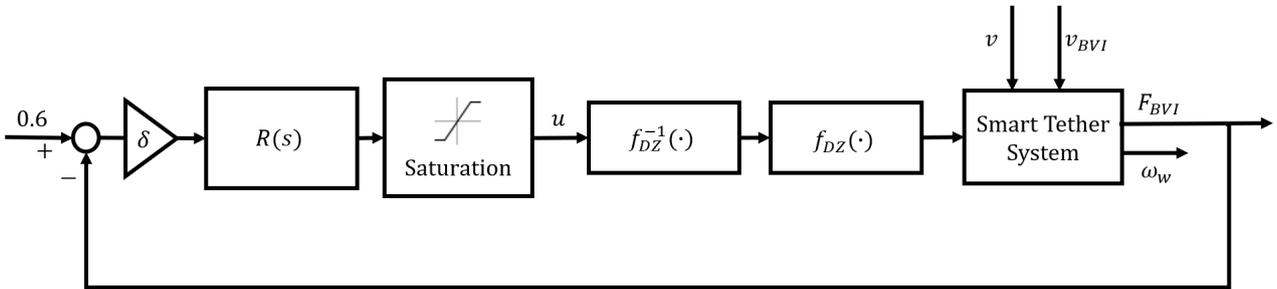


Figure 3.19: Control system;  $\delta$  (equation (3.14)) is the gain corresponding to the inverse of the derivative of the nonlinearity of fourth-order ( $K_T^{SS}$ ) evaluated in zero.

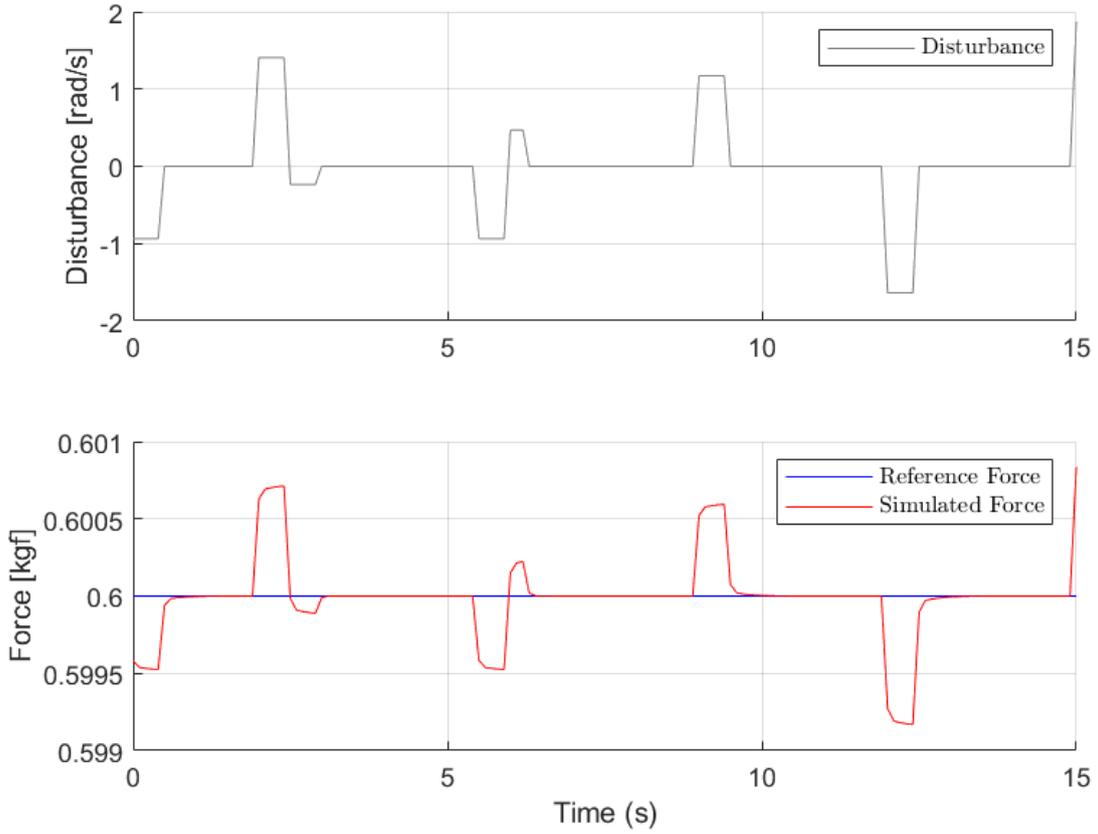


Figure 3.20: Simulation of the frequency control with disturbance, which is the offset  $v - v_{BVI}$ , already reported in angular velocity

### 3.3. LMI Controller design (Configuration 2)

#### 3.3.1. The Control-Oriented Model

Consider the scheme in Figure 3.21. In this section, we account for  $\Delta_{\theta_w}(s)$  as a multiplicative uncertainty and design a controller with some robustness properties. This is obtained by defining

$$G_{\theta_w}(s) = 1 + \Delta_{\theta_w}(s) \quad (3.17)$$

This notation, commonly used in robust control theory, accounts for the structured uncertainties in linear systems and will be introduced to design a control system with robust properties. It is important to note that  $\Delta_{\theta_w}(0) = 0$ , which means that the static gain is zero. We use such modeling approach for a manifold reason, i.e.,

- to limit the dimensionality of the model used in the control design

- to avoid, in the state-space model development, the propagation of the several uncertainties related to the identification of the class, the order, and the parameters of  $G_{\theta_w}(s)$ .

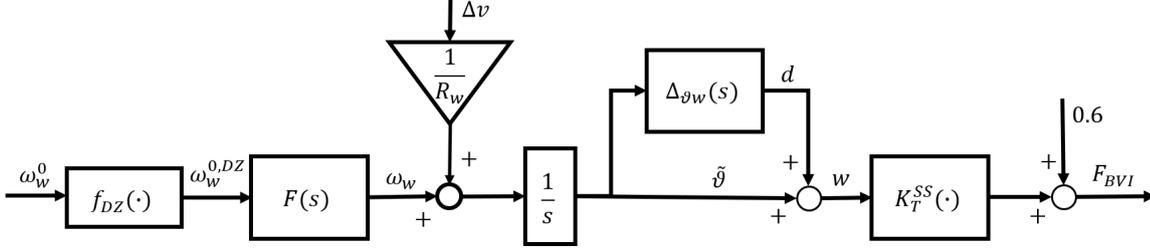


Figure 3.21: Smart Tether system;  $F(s)$  is the transfer function identified for the winch.  $\Delta_{\theta_w}(s)$  is the multiplicative uncertainty, given that  $G_{\theta_w} = 1 + \Delta_{\theta_w}(s)$ .

The linear model used for control design has input  $u = \omega_w^{o,DZ}$  (i.e., the motor's reference velocity after the deadzone function  $f_{DZ}$ , i.e.,  $\omega_w^{o,DZ} = f_{DZ}(\omega_w^o)$ ) and output  $w$  (i.e., the signal upstream the nonlinearity  $K_T^{SS}(\cdot)$ ). The disturbance is  $\Delta v = v - v_{BVI}$ .

The system dynamics can be defined as:

$$\omega_w = F(s)f_{DZ}(\omega_w^o) \quad (3.18)$$

$$\dot{\tilde{\theta}}_w = \frac{\Delta \dot{L}_T}{R_w} = \frac{\Delta v}{R_w} + \omega_w \quad (3.19)$$

With reference to the models developed in Sections 3.1.2, 3.1.3, and 3.1.5, the state vector is the following:

$$x = \left[ \omega_w \quad a_w \quad \tilde{\theta}_w \right]^T \quad (3.20)$$

where  $a_w = \dot{\omega}_w$  and the system dynamics is described by the following state-space equations:

$$\begin{cases} \dot{\omega}_w = a_w \\ \ddot{\omega}_w = -\frac{1}{\tau_w^2} \cdot \omega_w - \frac{2}{\tau_w} \cdot a_w + \frac{1}{\tau_w^2} \cdot \omega_w^{0,DZ} \\ \dot{\tilde{\theta}}_w = \omega_w + \frac{1}{R_w} \cdot \Delta v \end{cases} \quad (3.21)$$

The output, as discussed, is defined as

$$y = w = \tilde{\theta}_w + d \quad (3.22)$$

The model can be represented as follows:

$$\begin{cases} \dot{x} &= Ax + Bu + B_d \Delta v \\ y &= Cx + d \end{cases} \quad (3.23)$$

where the matrices  $A$ ,  $B$ ,  $B_d$ ,  $C$ , and  $D_d$  are given by:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{1}{\tau_w^2} & -\frac{2}{\tau_w} & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{1}{\tau_w^2} \\ 0 \end{bmatrix}, \quad B_d = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{R_w} \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}, \quad D_d = 0$$

This model is then discretized with a sampling time  $T_s = 0.1$  s, resulting in the discrete-time formulation:

$$x_{k+1} = Fx_k + Gu_k + G_d \Delta v_k \quad (3.24)$$

$$y_k = Hx_k + d_k \quad (3.25)$$

where  $d_k$  is considered as a measurement error. Note that  $y_k$  can be obtained as

$$y_k = K_T^{SS^{-1}}(F_{BVI} - 0.6) \quad (3.26)$$

. The latter can be linearized, as illustrated in Section 3.1.5.

### 3.3.2. Linear Matrix Inequalities and $\mathcal{H}_2$ control design

#### Linear Matrix Inequalities

Linear Matrix Inequalities (LMI) provide a powerful and versatile tool for controller design. This is due to their ability to systematically address various control problems. LMIs provide a framework for formulating and solving decision problems where the objective is to find matrices that simultaneously meet a set of linear conditions.

Consider, as a starting point, the discrete-time system

$$x_{k+1} = Fx_k + Gu_k \quad (3.27)$$

The aim is to design a suitable control gain  $K_x$  such that the static state-feedback control law is

$$u_k = K_x x_k \quad (3.28)$$

In this way, the closed-loop system dynamics is:

$$x_{k+1} = (F + GK_x)x_k \quad (3.29)$$

The matrix  $F + GK_x$  is asymptotically (Schur) stable if and only if there exists a symmetric matrix  $P > 0$  such that:

$$(F + GK_x)P(F + GK_x)^T - P < 0 \quad (3.30)$$

In control design, the unknowns are  $P$  and the gain  $K_x$ . The inequality above can be rewritten as:

$$\underbrace{FPF^T - P}_{\text{linear in } P} + \underbrace{FPK_x^T G^T + GK_x P F^T + GK_x P K_x^T G^T}_{\text{non linear in } P \text{ and } K_x} < 0 \quad (3.31)$$

Note that the one above is not a linear matrix inequality. However, by defining:

$$L = K_x P \quad (3.32)$$

it is possible to reformulate (3.31) as the following linear matrix inequality:

$$\underbrace{P - FPF^T - FL^T G^T - GLF^T}_{\text{linear in } P \text{ and } L} - \underbrace{GLP^{-1}L^T G^T}_{\text{non linear in } P \text{ and } L} > 0 \quad (3.33)$$

Inequality (3.33) is still nonlinear. However, it can be shown that, in view of the Schur complement [4], it is equivalent to:

$$\begin{bmatrix} P - FPF^T - FL^T G^T - GLF^T & GL \\ L^T G^T & P \end{bmatrix} > 0 \quad (3.34)$$

which is linear in both  $P$  and  $L$ .

To summarize, the LMI feasibility problem to be solved to design a control gain  $K_x$  that guarantees Schur stability of the closed-loop system matrix  $F + GK_x$  simply consists of checking if  $\exists P = P^T > 0$  and  $L$  such that (3.34) is satisfied. If such a solution exists, the gain matrix  $K_x = LP^{-1}$  guarantees such stability properties. The one formulated above does not guarantee optimality features to the closed-loop system. Optimality, however, can be enforced through further LMIs, as discussed in the following section.

## $\mathcal{H}_2$ control design for discrete-time systems

$\mathcal{H}_2$  control theory is fundamental in modern control engineering, focusing on designing controllers that minimize the norm of a given closed-loop transfer function. This approach is particularly relevant for systems where performance is measured by the energy of the error signal. The  $\mathcal{H}_2$  norm serves as a metric that quantifies the performance of a control system in terms of its ability to reject random disturbances, making it an essential method in the field of robust control.

Consider the discrete-time system:

$$\begin{aligned}x_{k+1} &= Fx_k + Gu_k + G_w w_k \\z_k &= Hx_k + D_u u_k\end{aligned}\tag{3.35}$$

where  $z_k$  is the so-called "performance output". Notably,  $H$  and  $D_u$  include the parameters that determine the influence of the inputs and states on the user-defined output.

In the following, we can derive the optimal control law  $u_k = K_x x_k$  that minimizes the  $\mathcal{H}_2$  norm of the transfer function  $G_{zw}(z)$ . In discrete-time, the  $\mathcal{H}_2$  norm of the transfer function  $G_{zw}(z)$  between  $w_k$  and  $z_k$ , under  $u_k = K_x x_k$ , can be computed as:

$$\|G_{zw}(z)\|_2^2 = \inf\{\text{trace}((H + D_u K_x)P(H + D_u K_x)^T)\}\tag{3.36}$$

subject to:

$$(F + GK_x)P(F + GK_x)^T - P + G_w G_w^T \leq 0\tag{3.37}$$

Let us define matrix  $S$  such that

$$S \geq (H + D_u K_x)P(H + D_u K_x)^T\tag{3.38}$$

If we define  $L = K_x P$ , we rewrite (3.37) and (3.38) as:

$$P - FPF^T - FL^T G^T - GLF^T - GLP^{-1}L^T G^T - G_w G_w^T \geq 0 \quad (3.39)$$

$$S - (HP + D_u L)P^{-1}(HP + D_u L)^T \geq 0 \quad (3.40)$$

Applying the Schur complement to constraints (3.39) and (3.40) we can cast the  $\mathcal{H}_2$  norm minimization problem as follows:

$$\min_{P,S,L} \text{trace}(S) \quad (3.41a)$$

subject to

$$\begin{bmatrix} P - FPF^T - FL^T G^T - GLF^T - G_w G_w^T & GL \\ L^T G^T & P \end{bmatrix} \geq 0 \quad (3.41b)$$

$$\begin{bmatrix} S & HP + D_u L \\ L^T D_u^T + PH^T & P \end{bmatrix} \geq 0 \quad (3.41c)$$

### 3.3.3. $\mathcal{H}_2$ Controller design for the smart tether system

First note that, in model (3.24), the state  $x_k$  is not measurable, as well as the disturbance  $\Delta v$ . Therefore, an observer needs to be initially designed to estimate both  $x_k$  and  $\Delta v_k$  from the available measurements of  $\omega_w$  and  $y_k$ . To this end, an extended model that includes  $\Delta v_k$  as a state variable is derived where the dynamics of  $\Delta v_k$  are assumed to be constant, as customary in disturbance estimation problems,  $\Delta v_{k+1} = \Delta v_k$ . Thus, the extended state is defined as  $\bar{x}_k = \begin{bmatrix} x_k^T & \Delta v_k \end{bmatrix}^T = \begin{bmatrix} \omega_w & a_w & \tilde{\theta}_k & \Delta v_k \end{bmatrix}^T$  with dynamics:

$$\bar{x}_{k+1} = \bar{F}\bar{x}_k + \bar{G}u_k \quad (3.42)$$

$$\bar{y}_k = \bar{H}\bar{x}_k \quad (3.43)$$

where:

$$\bar{F} = \begin{bmatrix} F & G_d \\ \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} & 1 \end{bmatrix}, \quad \bar{G} = \begin{bmatrix} G \\ 0 \end{bmatrix}, \quad \bar{H} = \begin{bmatrix} C & 0 \\ \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} & 0 \end{bmatrix}$$

A Kalman predictor is employed to estimate  $x_k$  and  $\Delta v_k$ , providing estimates  $\hat{x}_k$  and  $\hat{\Delta v}_k$ .

The control law designed to track the reference is such that

$$u_k = K_x(\hat{x}_k - x_k^o) + u_k^o \quad (3.44)$$

The values  $x_k^o$  and  $u_k^o$  are used as feedforward terms to compensate for the disturbance estimate  $\hat{\Delta}v_k$ , i.e.,

$$x_k^o = \begin{bmatrix} -\hat{\Delta}v_k/R_w & 0 & 0 \end{bmatrix}^T \quad (3.45)$$

$$u_k^o = -\hat{\Delta}v_k/R_w \quad (3.46)$$

The control gain  $K_x$  is derived by solving this LMI  $\mathcal{H}_2$  optimization problem (equation (3.41)) with YALMIP toolbox in MATLAB and the MOSEK solver. The solution provides the optimal gain matrix  $K_x$  for the  $\mathcal{H}_2$  controller for the system described in equation (3.24), i.e.,

$$K_x = \begin{bmatrix} -0.9545 & -0.0320 & -7.5275 \end{bmatrix} \quad (3.47)$$

The weights which have been selected for the state variable and the control input respectively are

$$H = \begin{bmatrix} 0 & 0 & \sqrt{q} \\ 0 & 0 & 0 \end{bmatrix}$$

where  $q = 120$

$$D_u = \begin{bmatrix} 0 \\ \sqrt{r} \end{bmatrix}$$

where  $r = 1$ .

The real control input  $\omega_w^o$  is defined as  $\omega_w^o = f_{DZ}^{-1}(u)$ . The complete control architecture is depicted in Figure 3.22.

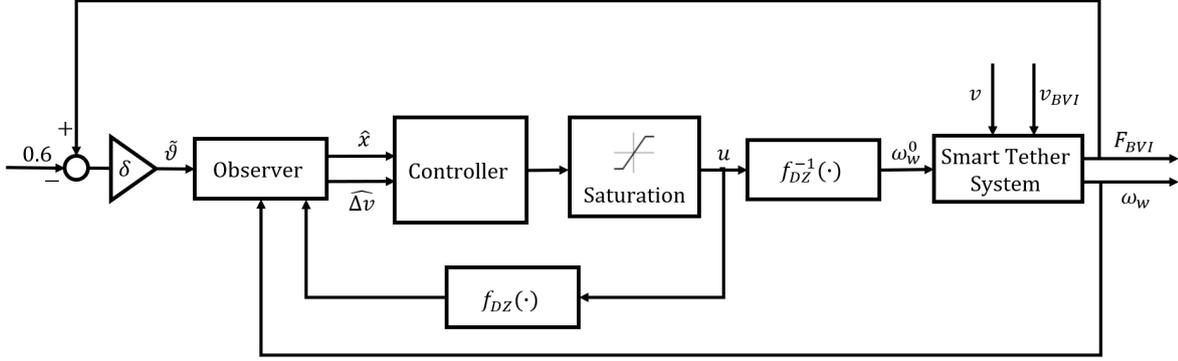


Figure 3.22: Control system;  $\delta$  (equation (3.14)) is the gain corresponding to the inverse of the derivative of the nonlinearity of fourth-order evaluated in zero

Some small modifications have been done with respect to the ideal control law (equation (3.44)). Firstly we noted that, in view of the significant measurement errors, the estimate of  $\theta_k$  was affected by relevant fluctuations. To overcome these oscillations, the estimate  $\hat{\theta}_k$ , provided by the Kalman filter, is further filtered using a low-pass filter with a bandwidth of 4.3655 rad/s.

Secondly, note that, to define the feedforward terms  $x_k^o$  and  $u_k^o$ , the control law (3.44) uses the estimate  $\hat{\Delta v}_k$  instead of the real disturbance  $\Delta v_k$ . The fact that  $\hat{\Delta v}_k$  is estimated makes terms  $x_k^o$  and  $u_k^o$  (ideally defined in open loop) to be defined in closed loop, thus affecting the control system performance.

To avoid this we compute

$$x_k^o = \begin{bmatrix} -dz(\hat{\Delta v}_k) & 0 & 0 \end{bmatrix}^T \quad (3.48)$$

$$u_k^o = -\frac{dz(\hat{\Delta v}_k)}{R_w} \cdot \mu \quad (3.49)$$

where  $dz(\cdot)$  is a deadzone function (see Figure 3.23) introduced to avoid the use of feedforward terms for small values of  $\hat{\Delta v}_k$ , avoiding the feedback action when close to stationary conditions.

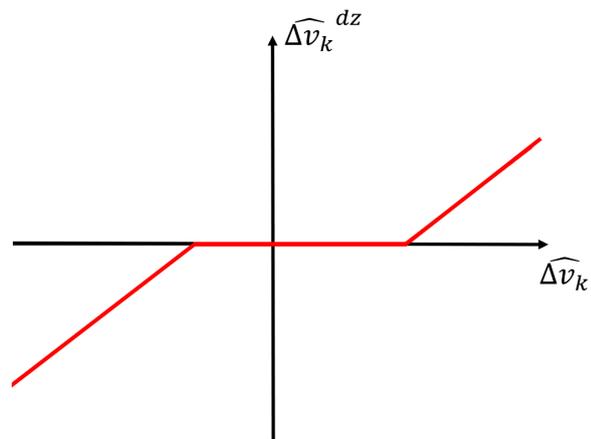


Figure 3.23: Deadzone  $dz(\cdot)$  introduced on  $\Delta \hat{v}_k$ . For  $\Delta \hat{v}_k$  within the interval  $[-0.015, 0.025]$ ,  $\Delta \hat{v}_k^{dz} = 0$ .

Also, the gain  $\mu = \frac{1}{0.9}$  has been introduced to make the control more reactive to speed changes and to compensate for the gain decrease occurring when applying the deadzone function  $dz(\cdot)$ .



# 4 | System Structure, components and control of the second setup (Configuration 3)

## 4.1. New setup

The development and implementation of a new setup of the smart tether system stems from the necessity to further enhance its performance, refining the system's precision, reliability, and responsiveness, thereby providing a safer and more effective user experience.

Central to this system is a high-performance motor, chosen for its ability to be directly controlled by torque and to deliver high-speed operations, ensuring that the system can handle the dynamic requirements of sports activities as well.

The new setup (Figure 4.1) integrates several other components, such as the N5-2-2 controller, a 5kg full-bridge load cell, employed for force monitoring, an HX711 amplifier, an Arduino board, a capacitor, and an absolute encoder.

The overall measures of the system are 29 cm in length, 25 cm in depth, and 23.5 cm in height, with a total weight of around 6 kg.

#### 4| System Structure, components and control of the second setup (Configuration 3)

56

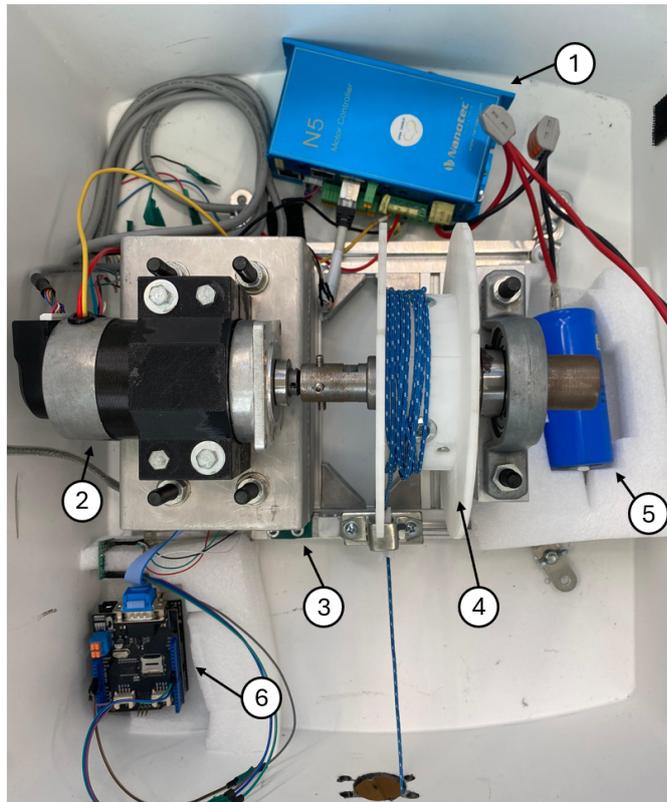


Figure 4.1: The Smart Tether System: 1. N5-2-2 controller; 2. Motor; 3. Load cell; 4. Winch; 5. Capacitor; 6. Arduino board.

The motor features a series of relevant technical specifications that define its performance and operating conditions. The motor, illustrated in Figure 4.2, operates at a nominal voltage of 24 V DC. The nominal current is 13.6 A, while the peak current can reach up to 40 A. Under nominal load conditions, the motor reaches a speed of 3500 rpm, while without load it can go up to 4500 rpm. The nominal torque of the motor is 0.044 Nm, with a peak of 0.135 Nm. The encoder integrated into the motor operates at a supply voltage of +5V DC and provides 18 pulses per revolution, allowing precise control of the motor's position and speed.



Figure 4.2: Motor

The Z-K10000/100 capacitor (Figure 4.3), with a standard capacitance of 10000  $\mu\text{F}$  and a dielectric rating of up to 100 V, is selected for its ability to store large amounts of energy and provide stable voltage levels.



Figure 4.3: The Z-K10000/100 capacitor (95 mm in length,  $\text{\O} 40\text{mm}$ )

The shaft (Figure 4.4) has been custom-designed and manufactured to connect the crankshaft and the winch.



Figure 4.4: Shaft designed to fit the gear-shaped crankshaft

The coil itself has been designed with the same specifications as the one used in the previous setup, as the requirements remained similar: lightness and compact dimensions

#### 4| System Structure, components and control of the second setup (Configuration 3)

58

(Figure 4.7). Its width has been reduced by internally fixing it to the shaft, creating a more compact system and saving space. This design allows the rigid cable to pass directly from the winch to outside the robot, through the small bearing, shown in Figure 4.5, connected to the load cell through the screw (Figure 4.10).

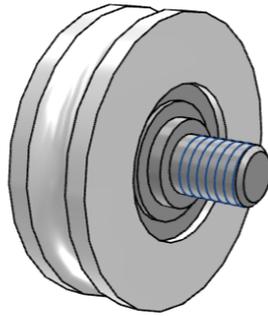


Figure 4.5: U-groove plastic bearing (diameter 30mm)



Figure 4.6: Load cell, bearing and winch

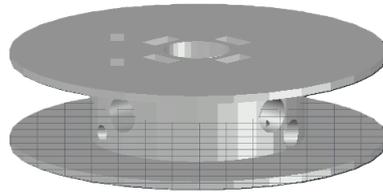


Figure 4.7: Winch with internal fixing mechanism

The N5-2-2 controller (Figure 4.8) produced by Nanotec was selected to control the motor. It is designed to control both stepper and BLDC motors, operating in either open-loop or closed-loop configurations. It supports a range of motor types and commutation methods, including open-loop stepper motors, closed-loop stepper motors with an encoder, BLDC motors with Hall sensors, and BLDC motors with encoders, the one of our interest. The controller is suitable for a variety of operating modes such as Profile Position Mode, Profile Velocity Mode, Profile Torque Mode, Velocity Mode, Homing Mode, Interpolated Position Mode, Cyclic Sync Position Mode, Cyclic Sync Velocity Mode, Cyclic Synchronous Torque Mode, and Clock-Direction Mode. As it will be deeply treated, the Velocity Mode is the operating mode selected for our purpose.

The N5-2-2 operates with an input voltage range of 12 V to 57.4 V DC ( $\pm 5\%$ ) for our hardware version (version w008). The number of pole pairs is 3 and the rated current is 18  $A_{rms}$ , with a peak current of 40  $A_{rms}$  for 5 seconds.

The controller interfaces include CANopen and Ethernet, supporting communication and control. It has four digital inputs (5 V/24 V, switchable via software), two wide-range inputs (5-24 V), and two analog inputs (-10 to +10 V or 0-20 mA, switchable via software). Outputs consist of two open-drain outputs (max 24 V and 500 mA). For motor feedback, it supports encoder inputs with a resolution of up to 65536 increments per revolution and can handle both differential and single-ended signals, which are switchable via software.

For effective commissioning, the N5-2-2 controller uses the Plug & Drive Studio software, which facilitates configuration and adaptation to the connected motor. The setup includes setting the IP address for Ethernet connections, with options for DHCP, Auto-IP, or static IP configurations.

#### 4| System Structure, components and control of the second setup (Configuration 3)

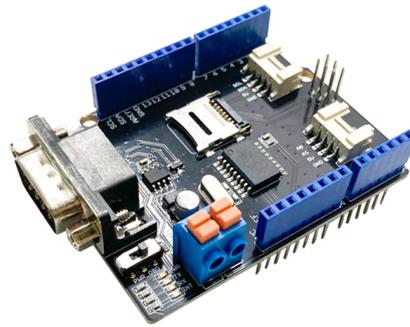


Figure 4.8: N5-2-2

Additionally, control can be managed using a microcontroller like Arduino (Figure 4.9a) and, since the interface chosen to communicate is CANopen due to its major manageability, a CAN-BUS shield (Figure 4.9b), to implement the CAN protocol directly in Arduino.



(a) Arduino



(b) CAN-BUS shield

Figure 4.9: Arduino and CAN-BUS shield

Similarly to the previous setup, illustrated in Chapter 2, a properly calibrated load cell is utilized to monitor the force applied to the user during usage. These measurements serve primarily as a means to ensure a more reliable verification of the forces at play, thereby enhancing the overall accuracy of the system. While the new Nanotec setup is equipped to measure torque, a functionality that is extensively leveraged in the control implementation discussed in the next section, the load cells provide an additional layer of verification.

#### 4| System Structure, components and control of the second setup (Configuration 3)

61

A significant advantage of this configuration, combining the load cell with its structural setup, is the ability to redirect the non-elastic rope from the winch directly outside the robot, as detailed in Chapter 2. This structural configuration reduces internal complexity and minimizes potential interference among components.

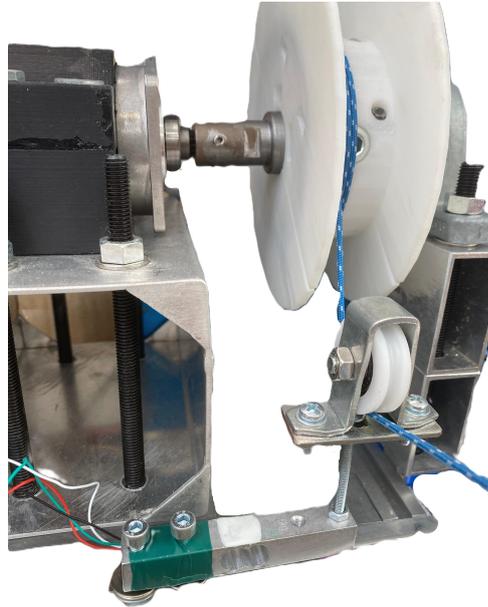


Figure 4.10: Load cell, bearing and winch

To accurately capture the data from the full-bridge load cell, we employed the HX711 amplifier, as detailed in Chapter 2.

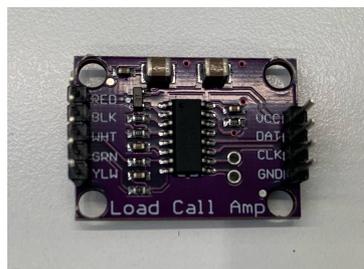


Figure 4.11: HX711 amplifier

The proximity sensor, which was previously used to prevent users from getting too close to the robot and which allows relaying exact position data to the encoder, avoiding possible calculation errors, has been deemed redundant and can be safely removed. This decision stems from the integration of an absolute encoder with the motor, which allows for enhanced safety measures.

#### 4| System Structure, components and control of the second setup (Configuration 3)

62

The encoder used in our system is a crucial component for precise position tracking and motor control. The one employed is the Nanotec NME2, which is a magnetic rotary encoder available in both incremental (UVW output) and absolute (SSI) variants, and the latter is of our interest.

It provides exact position data, ensuring that positional information is retained even after power loss, avoiding the need for re-homing or recalibration after power cycles.

The NME2 encoder employed can achieve resolutions up to 65536 positions per mechanical revolution, providing extremely detailed and precise feedback for the motor control system. This high resolution is facilitated by its robust design, capable of operating at mechanical speeds up to 12000 revolutions per minute, which makes it suitable for high-speed applications.

The electrical properties of the NME2 include a typical current consumption of 50 mA at 5 V or 40 mA at 12 V without load.

The encoder utilizes SSI (Synchronous Serial Interface) for data transmission, supporting clock frequencies up to 10 MHz, enabling fast and reliable data transmission, and outputs 24-bit data with a binary-encoded message, including position and error bits.

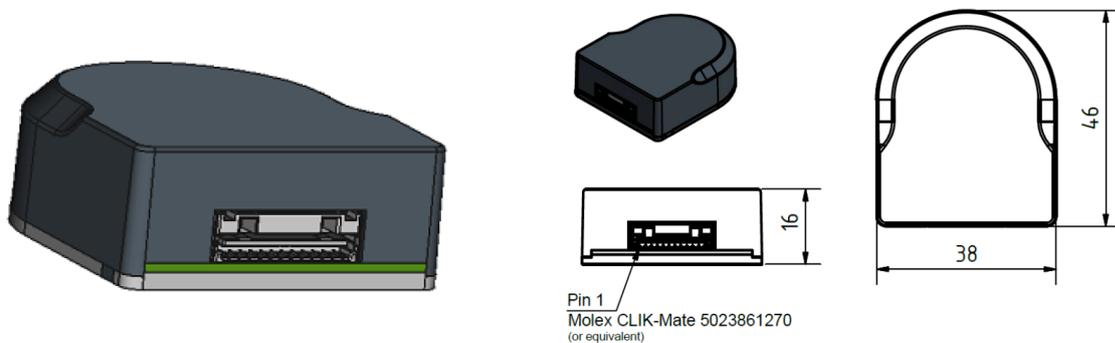


Figure 4.12: Encoder NME2, all dimensions are in millimeters

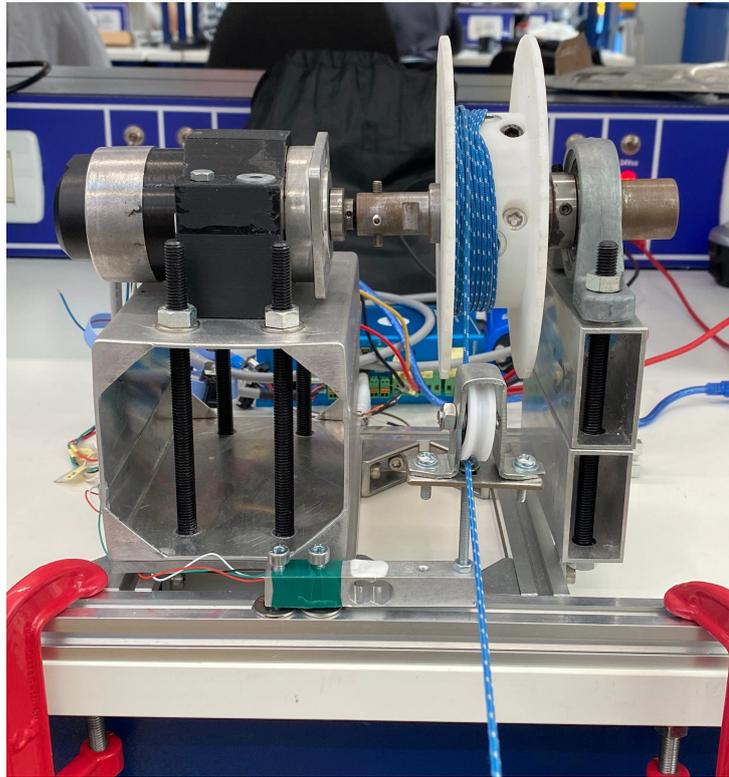


Figure 4.13: Complete new setup

## 4.2. Control

In this section, we illustrate the low-level controller of the new smart tether system setup. The implemented closed loop method is also referred to as sine commutation via an encoder with field-oriented control (FOC).

### 4.2.1. Motor setup

Prior to operation, the controller requires motor and encoder-specific data necessary for closed-loop operation. They are determined through the auto setup process, which involves several steps and must be successfully completed in a closed-loop system to ensure the system can accurately control the motor based on real-time feedback.

During the auto-setup process, the calibration of the motor and feedback systems is executed to their optimal settings. Initially, the resistance and inductance of the motor windings are measured. These values are fundamental as they influence the control algorithm's ability to model the motor's electrical behavior and they ensure that the system can apply the correct voltages and currents to achieve desired performance characteristics. Additionally, the encoder is aligned with the motor's electrical angle ensuring that the

control system has precise information about the rotor's position at all times, which is essential for synchronizing the motor's magnetic fields with its physical rotation. The encoder resolution is determined, defined by the number of pulses per revolution, and determines the precision of position feedback, which is crucial for accurate control of the motor's rotation.

Another crucial step in the auto-setup process is calibrating the current sensors. Its accuracy ensures that the control system can precisely measure and regulate the motor's current, which directly affects torque and speed control.

Moreover, the system estimates key motor parameters such as the back EMF constant, torque constant, and inertia during the auto-setup process. These parameters are used by the control algorithms to accurately predict the motor's behavior under different operating conditions. For instance, the back EMF constant relates to the voltage generated by the motor as it rotates, which is crucial for determining the appropriate voltage inputs needed to achieve a specific speed. The torque constant defines the relationship between the current supplied to the motor and the torque produced, which is essential for torque control applications. Inertia, which represents the resistance to changes in motion, affects the motor's acceleration and deceleration profiles.

Once the auto-setup is complete, and the parameters are correctly configured, the field-oriented control system can operate the motor efficiently. [1]

### 4.2.2. Motor control

The implemented control consists of speed control in velocity mode. In this mode, the system can operate at a preset target speed, with the ability to control torque based on specific requirements. It takes the target velocity as input and the actual speed as output.

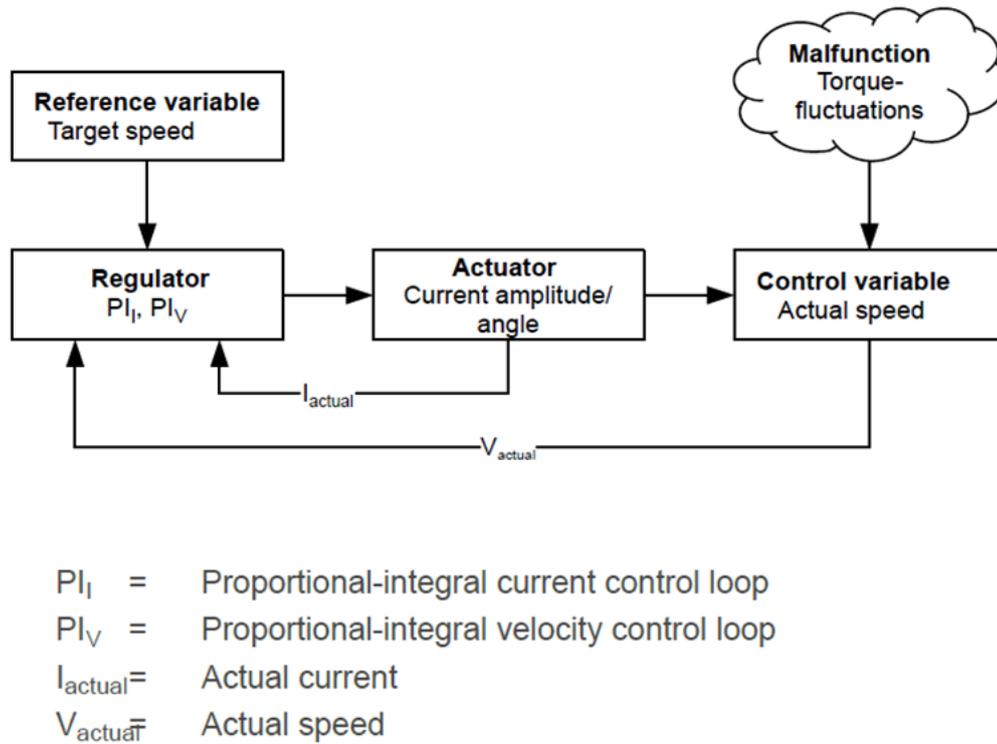


Figure 4.14

The target speed is set in hexadecimal format. Specifically, in our control it is set to -0x60 (hexadecimal) and translated to a specific speed value in revolutions per minute (rpm), the internal unit for speed. During operation, the motor adjusts its speed to match this target value.

Torque control in velocity mode is another critical aspect of the FOC system. Specifically, the speed is maintained to pull the rope effectively. The torque is set as a per thousand (‰) of the rated current, fixed at 400‰ initially, allowing the motor to handle the interaction with the user with a controlled amount of force.

This setup ensures that if the rope is pulled with a force exceeding the set torque, the reel will release the rope, letting the user go further, always exerting a constant torque on the person.

### 4.2.3. Safety functions

Additionally, the speed control system incorporates additional requirements. Under a specific position value of -2000 (-3.0680 rad and a distance of 70 cm between Yape and the user), individuated by the absolute encoder, the motor speed is reduced to zero to

prevent the user from being pulled too close to the robot. This speed reduction is essential to avoid damage, mishandling or injuries (Figure 4.15).

In a predetermined range of position values, between 1000 (1.5340 rad and a distance of 115 cm between Yape and the user) and the abovementioned too-close condition, the target speed, previously set at  $-0 \times 60$  (96 rpm and 10.0531 rad/s), decreases in a linearly proportional manner with respect to the decreasing phase, until the too-close condition and the zero speed are reached (Figure 4.15). When the user is too far, meaning above a position value of 8000 (12.2718 rad and a distance of 170 cm between Yape and the user) defined by the absolute encoder, the torque is increased to 700% of the rated current. This adjustment ensures that the motor can exert more force to pull the rope and the user closer effectively (Figure 4.16).

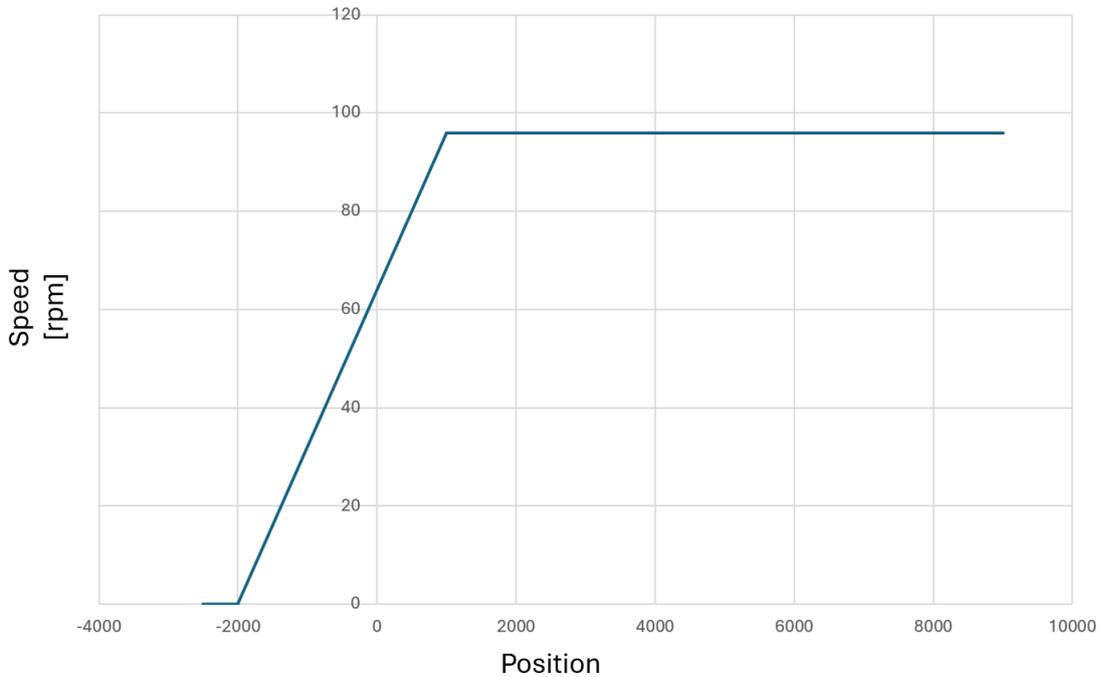


Figure 4.15: Safety functions implemented on speed

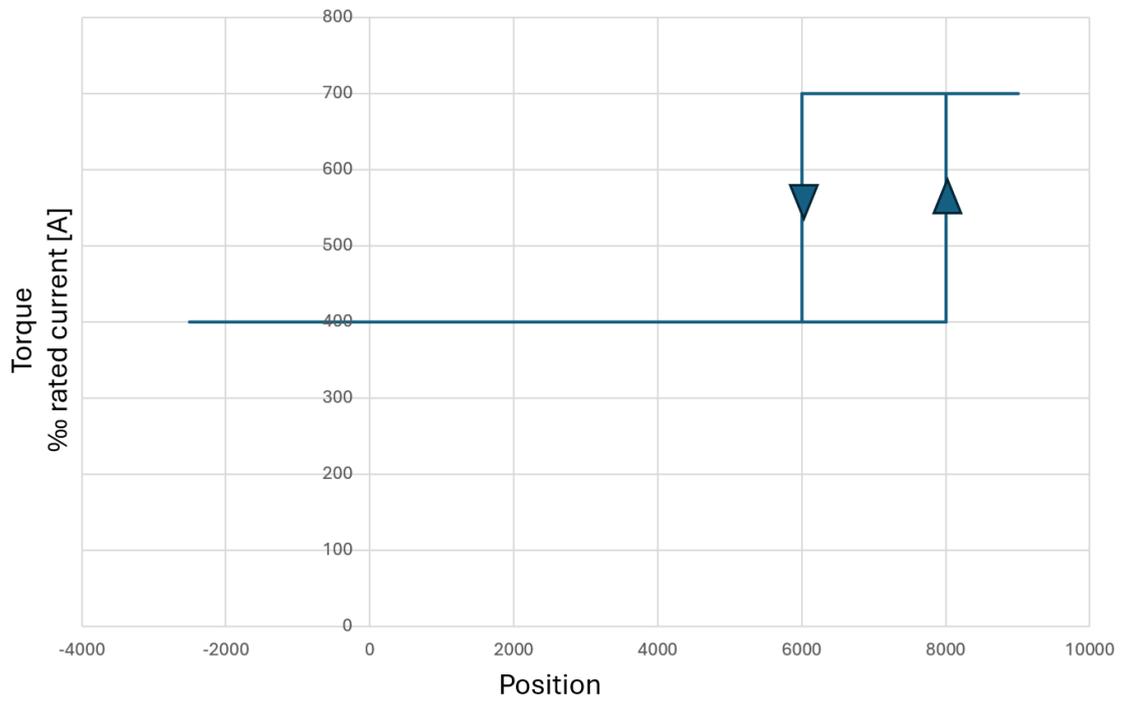


Figure 4.16: Safety functions implemented on torque

This setup provides precise control over the motor's speed and torque, ensuring safe and efficient operation and rapid response.

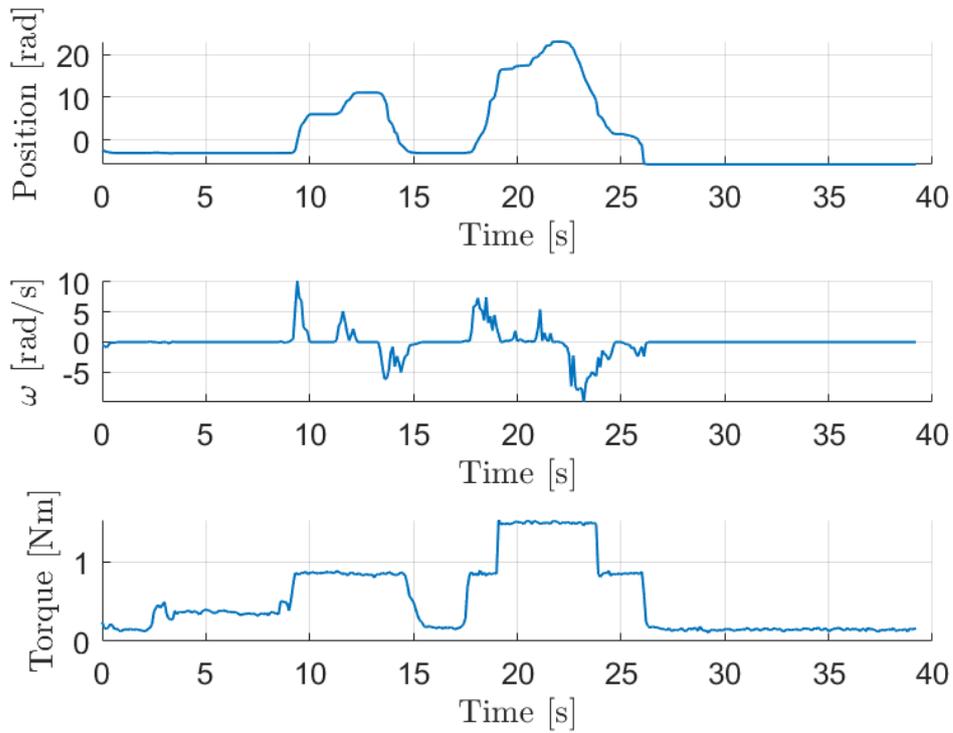


Figure 4.17: Experiment showing the behavior of the motor control system in response to various position values.

At a position value of  $-3.0680$  rad (distance of 70 cm between Yape and the user), the motor speed is reduced to zero to prevent the user from being pulled too close to the robot.

When the position is beyond 12.2718 radians (distance of 170 cm), the torque is increased to ensure the motor can pull the user closer effectively.

## 5 | Performance comparison between the different configurations

The performance and limitations of the different smart tether systems configurations discussed in Chapters 3 and 4 have been assessed through an experimental campaign at the Human Performance Lab, Politecnico di Milano - Sede territoriale di Lecco. This experimental scenario included a high-speed Technogym treadmill and two different setups: either Yape standing still (Figure 5.1) to test the frequency domain controller (Configuration 1) and the  $\mathcal{H}_2$  controller (Configuration 2), or a table positioned in front of the treadmill (Figure 5.2) to hold the smart tether system with the new motor (Configuration 3).



Figure 5.1: Configurations 1 and 2; first motor and Yape standing still



Figure 5.2: Configuration 3; new motor fixed on the table

The angular velocity  $\omega_w$  of the smart tether's winch and the exerted force  $F_{BVI}$  were measured using the smart tether electronic board. For the third configuration,  $F_{BVI}$  was measured both by the controller's torque recordings converted to force and by the electronic board.

Various volunteers tested the three different configurations by walking at speeds of 1.5 and 4 km/h and running at speeds of 6.5, 9, and 12 km/h, on the treadmill, holding the smart tether handle in their right hand. The position of the hand was recorded using an optoelectronic system comprising six cameras (BTS Smart DX 400 Bioengineering, Milan, Italy) and four reflective passive markers placed on the following anatomical landmarks: ulnar styloid process, radial styloid process, lateral head of metacarpal, and medial head of metacarpal (Figure 5.3). Additionally, the position of the smart tether was recorded by applying a passive marker either on Yape (for Configurations 1 and 2) or on the table in front of the treadmill (for Configuration 3).



Figure 5.3: Right hand musculoskeletal landmarks

First, the relative displacements between the volunteers' hands and the table were analyzed. By utilizing the abovementioned optoelectronic system, the positions of the hand in the  $x$ ,  $y$ , and  $z$  directions were accurately measured. For each participant, the mean value of the available marker positions at each time point was calculated. The Euclidean distance between the hand and the smart tether (denoted by the fixed marker either on Yape or on the table in front of the treadmill, referred to as  $x_{ST}$ ,  $y_{ST}$  and  $z_{ST}$ ) was determined using the following formula:

$$Distance = \sqrt{(x - x_{ST})^2 + (y - y_{ST})^2 + (z - z_{ST})^2} \quad (5.1)$$

The resulting Euclidean distances are illustrated in Figure 5.4.

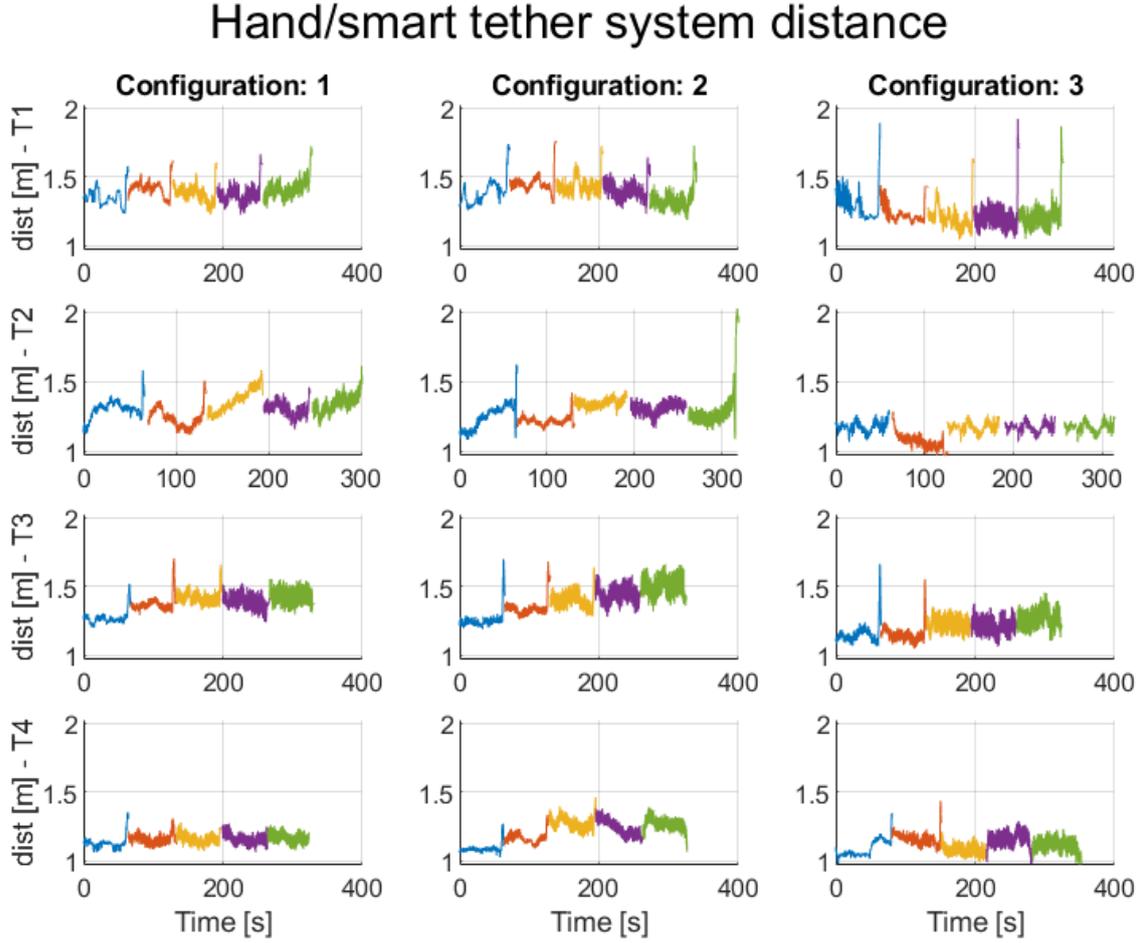


Figure 5.4: Distance between the user’s hand and the smart tether system. The three configurations are shown and ordered increasingly from left to right. The four testers (T1-T4) are displayed in increasing order from the top to the bottom. The data referring to  $v_{BVI} = 1.5$  km/h are shown in blue,  $v_{BVI} = 4$  km/h in orange,  $v_{BVI} = 6.5$  km/h in yellow,  $v_{BVI} = 9$  km/h in purple and  $v_{BVI} = 12$  km/h in green.

Additionally, the collected signals related to  $F_{BVI}$  and  $\omega_w$  are presented in Figures 5.5, 5.6, 5.7 and 5.8.

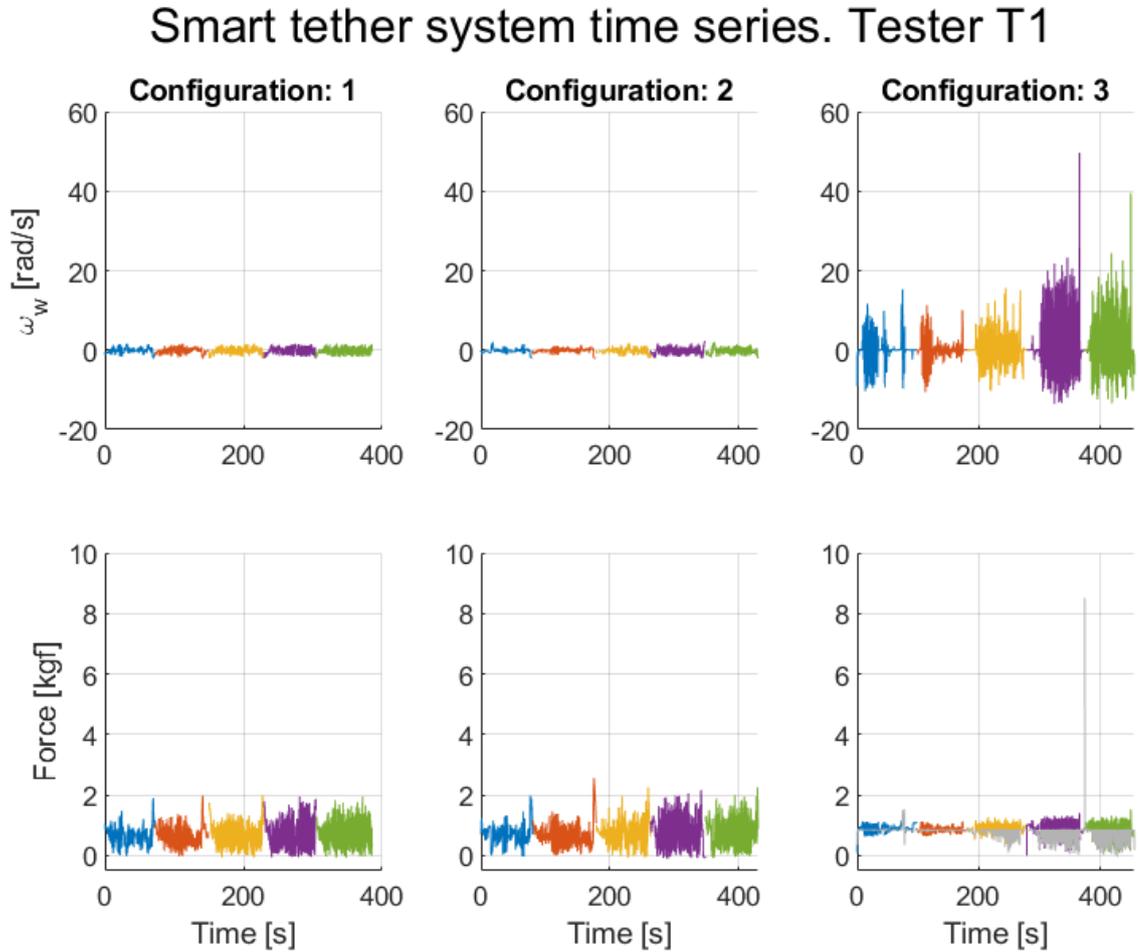


Figure 5.5: Winch velocity and Force for Tester T1. The three configurations are shown and ordered increasingly from left to right. The data referring to  $v_{BVI} = 1.5\text{km/h}$  are shown in blue,  $v_{BVI} = 4\text{km/h}$  in orange,  $v_{BVI} = 6.5\text{km/h}$  in yellow,  $v_{BVI} = 9\text{km/h}$  in purple and  $v_{BVI} = 12\text{km/h}$  in green.

In grey, we present the measurements of the actual torque provided by the Nanotec controller, already reported in kgf.

### Smart tether system time series. Tester T2

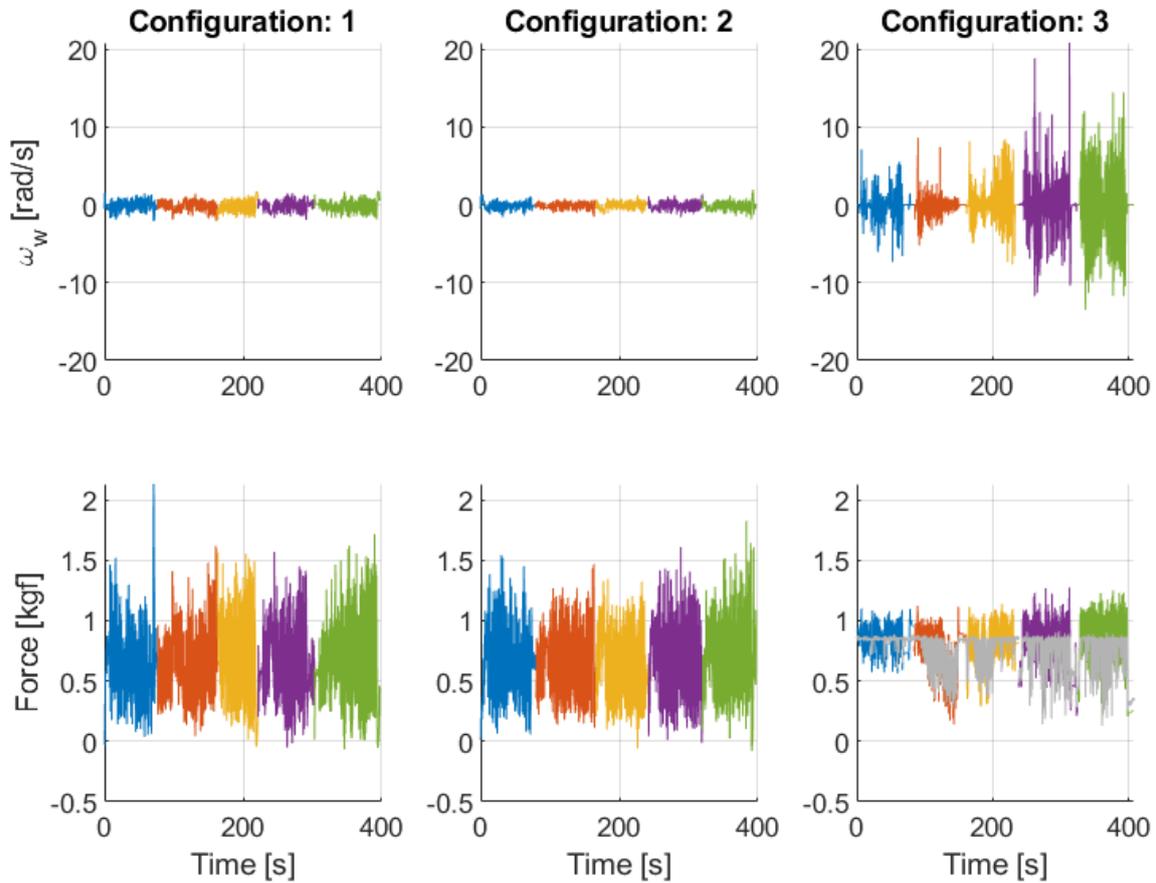


Figure 5.6: Winch velocity and Force for Tester T2. The three configurations are shown and ordered increasingly from left to right. The data referring to  $v_{BVI} = 1.5\text{km/h}$  are shown in blue,  $v_{BVI} = 4\text{km/h}$  in orange,  $v_{BVI} = 6.5\text{km/h}$  in yellow,  $v_{BVI} = 9\text{km/h}$  in purple and  $v_{BVI} = 12\text{km/h}$  in green.

In grey, we present the measurements of the actual torque provided by the Nanotec controller, already reported in kgf.

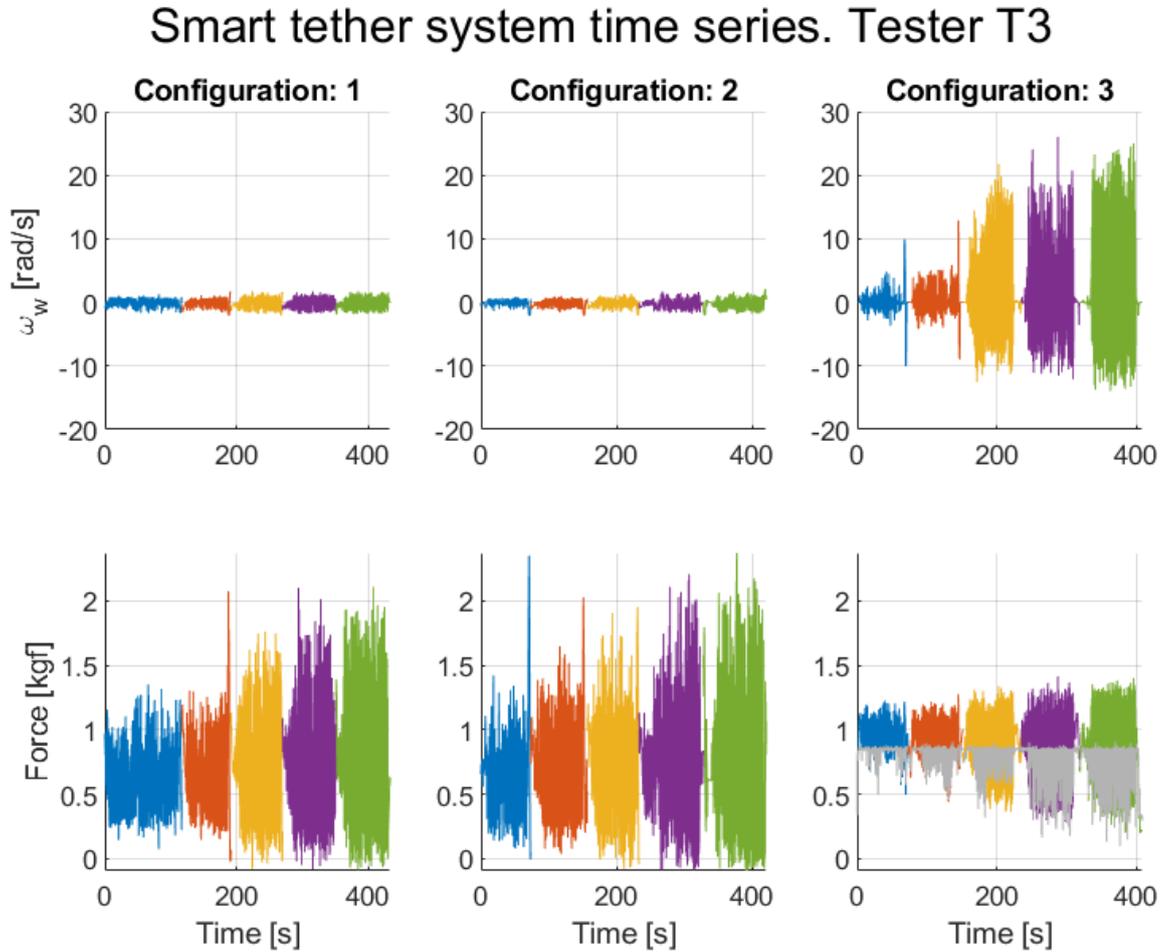


Figure 5.7: Winch velocity and Force for Tester T3. The three configurations are shown and ordered increasingly from left to right. The data referring to  $v_{BVI} = 1.5\text{km/h}$  are shown in blue,  $v_{BVI} = 4\text{km/h}$  in orange,  $v_{BVI} = 6.5\text{km/h}$  in yellow,  $v_{BVI} = 9\text{km/h}$  in purple and  $v_{BVI} = 12\text{km/h}$  in green.

In grey, we present the measurements of the actual torque provided by the Nanotec controller, already reported in kgf.

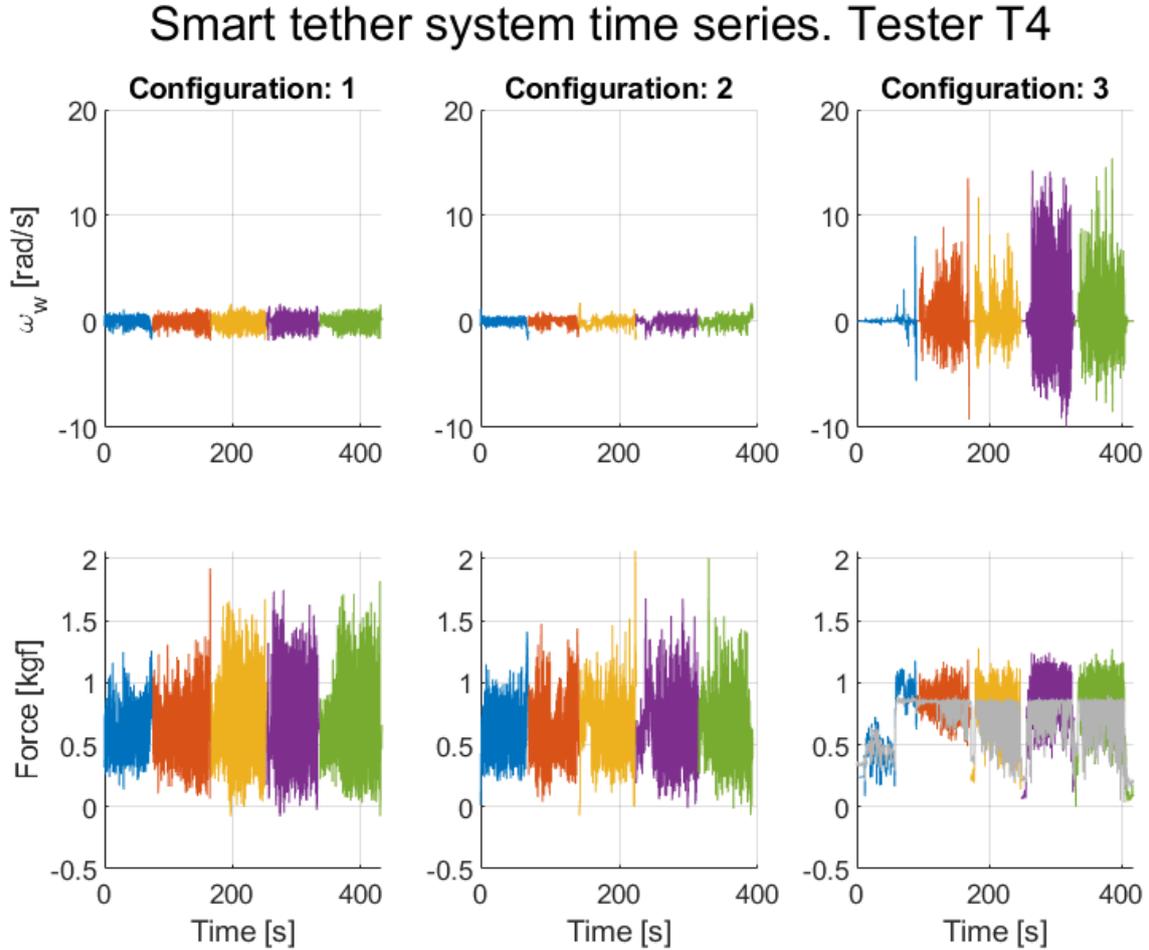


Figure 5.8: Winch velocity and Force for Tester T4. The three configurations are shown and ordered increasingly from left to right. The data referring to  $v_{BVI} = 1.5\text{km/h}$  are shown in blue,  $v_{BVI} = 4\text{km/h}$  in orange,  $v_{BVI} = 6.5\text{km/h}$  in yellow,  $v_{BVI} = 9\text{km/h}$  in purple and  $v_{BVI} = 12\text{km/h}$  in green.

In grey, we present the measurements of the actual torque provided by the Nanotec controller, already reported in kgf.

Configurations 1 and 2 demonstrate similar performance. The results indicate that the control system effectively controls the force exerted on the user, particularly at low speeds where system disturbances are minimal. At higher speeds, however, the motor's limitations cause the velocity  $\omega_w$  to saturate. Consequently, as the treadmill speed increases, the disturbances impacting the system become more pronounced, and the system struggles to control the force promptly. Nevertheless, it is important to note that even when disturbances are significant, the system maintains the force within a reasonable range, i.e.,  $F_{BVI} \leq 2$  kgf, except during the final instants of each speed range when braking

occurs and the distance increases (Figure 5.4).

For the 3<sup>rd</sup> Configuration, the enhanced responsiveness and reactivity of the new motor enable significantly more precise force control, even at elevated speeds. In fact, as shown in Figures 5.5, 5.6, 5.7 and 5.8, the winch velocity  $\omega_w$  reaches -10 rad/s when the motor pulls the cable to coil it, whereas, when the cable is released towards the user, it can reach up to 40 rad/s. This differs from Configurations 1 and 2, where the winch velocity saturates at -2 and 2 rad/s. Moreover, even at the maximum treadmill velocity, i.e., 12 km/h, the force exerted on the user is kept within a limited and acceptable range ( $F_{BVI} \leq 1.4$  kgf).

In the plots related to Configuration 3, comparing force and torque (reported to kgf), it is evident that the torque does not exceed the value of 0.85 kgf when releasing the cable, in contrast to the readings from the load cell sensor. Conversely, when rewinding the cable, both the torque and force measurements exhibit approximately the same values.

To obtain a different viewpoint on the results, the force exerted on the user and the distance between the hand and the smart tether were analyzed in the frequency domain. The Fourier transform has been applied to different segments of the signals, each representing a distinct treadmill speed. The results are shown in Figures 5.9, 5.10, 5.11, and 5.12. The x-axis is represented on a logarithmic scale with base 10, while in the y-axis we display the amplitude of the Fourier transform on a linear scale.

## FFT: Tester 1

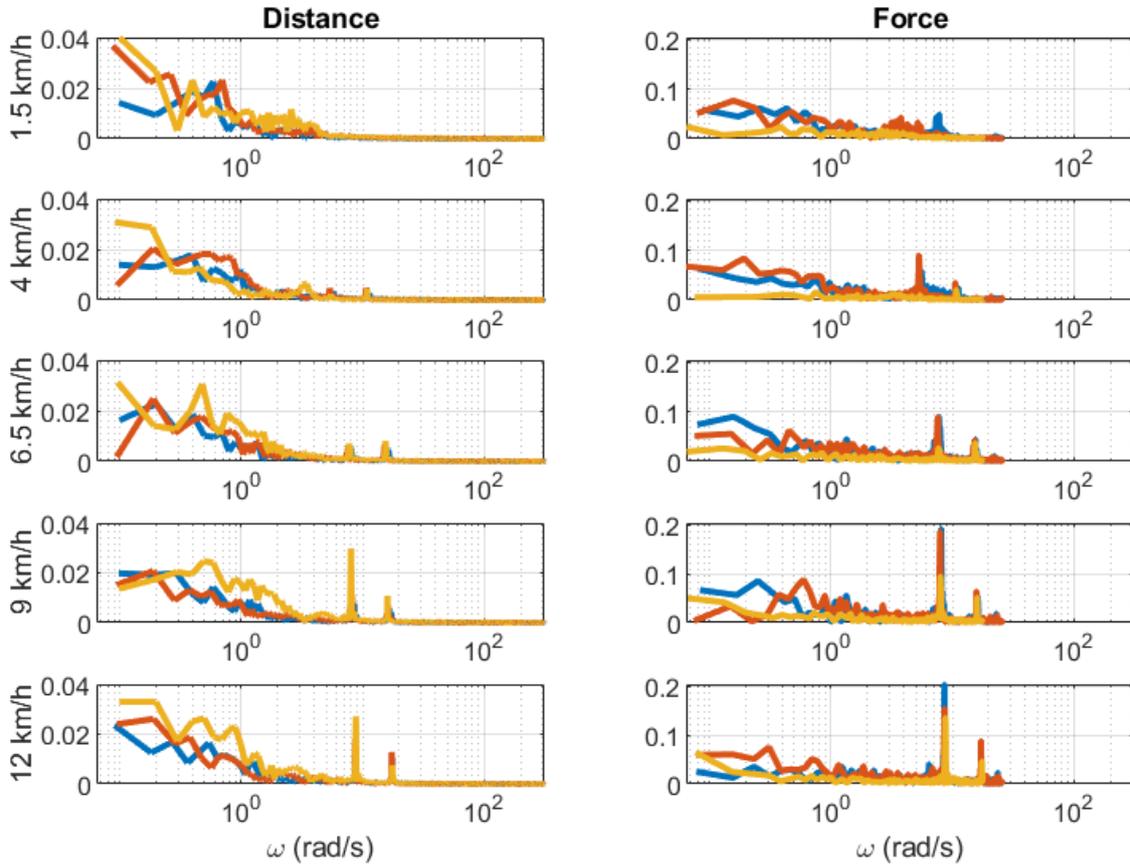


Figure 5.9: Fourier transforms of the hand-table or hand-Yape distance (on the left) and of the force  $F_{BVI}$  (on the right) for different treadmill velocities (from the top to the bottom):  $v_{BVI} = 1.5\text{km/h}$ ,  $v_{BVI} = 4\text{km/h}$ ,  $v_{BVI} = 6.5\text{km/h}$ ,  $v_{BVI} = 9\text{km/h}$  and  $v_{BVI} = 12\text{km/h}$ . For each panel, we can find in blue the Fourier transform of 1<sup>st</sup> Configuration, in red the 2<sup>nd</sup> Configuration, and in yellow the 3<sup>rd</sup> Configuration.

## FFT: Tester 2

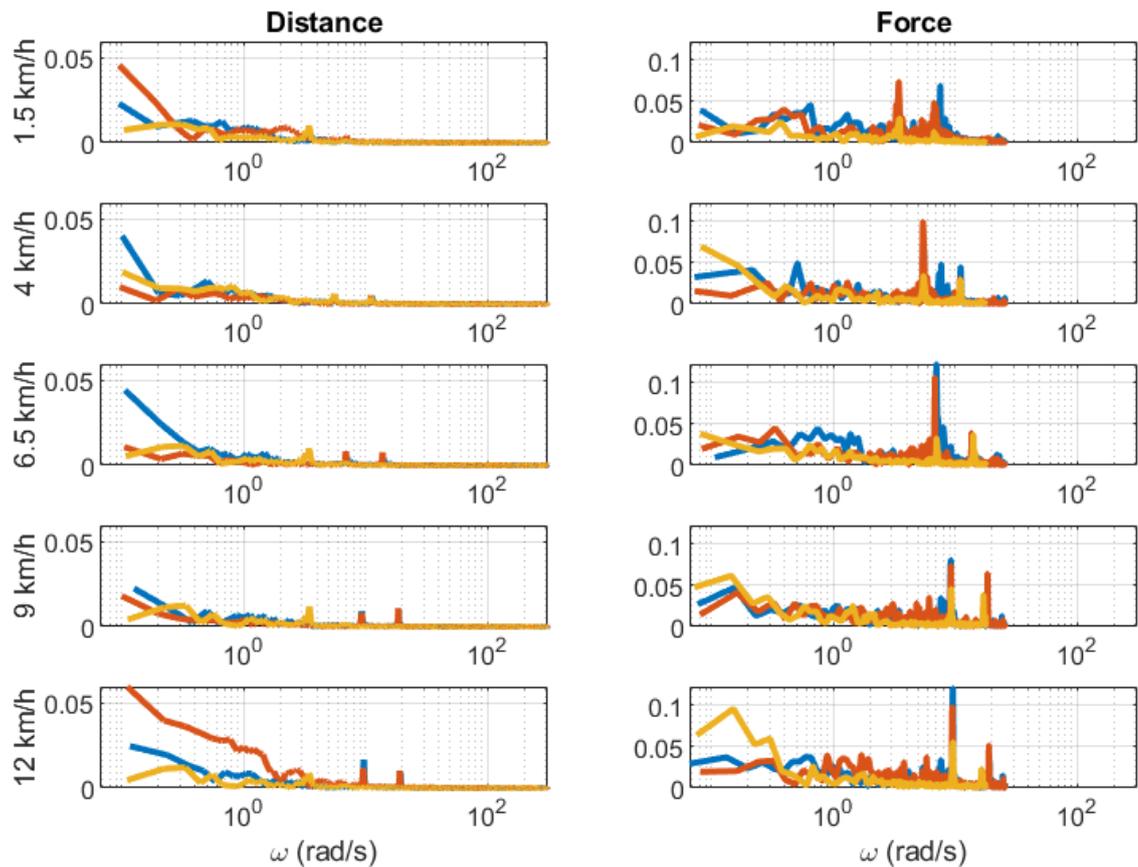


Figure 5.10: Fourier transforms of the hand-table or hand-Yape distance (on the left) and of the force  $F_{BVI}$  (on the right) for different treadmill velocities (from the top to the bottom):  $v_{BVI} = 1.5\text{km/h}$ ,  $v_{BVI} = 4\text{km/h}$ ,  $v_{BVI} = 6.5\text{km/h}$ ,  $v_{BVI} = 9\text{km/h}$  and  $v_{BVI} = 12\text{km/h}$ . For each panel, we can find in blue the Fourier transform of 1<sup>st</sup> Configuration, in red the 2<sup>nd</sup> Configuration, and in yellow the 3<sup>rd</sup> Configuration.

## FFT: Tester 3

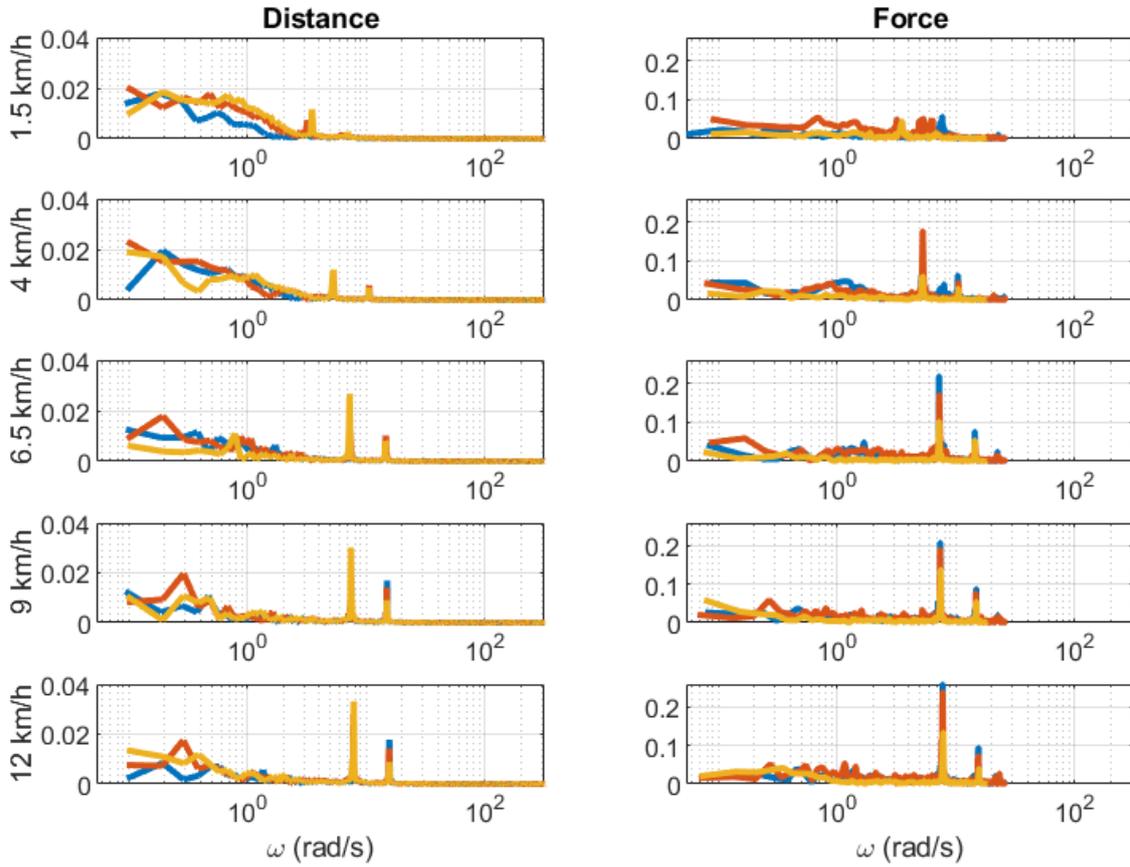


Figure 5.11: Fourier transforms of the hand-table or hand-Yape distance (on the left) and of the force  $F_{BVI}$  (on the right) for different treadmill velocities (from the top to the bottom):  $v_{BVI} = 1.5\text{km/h}$ ,  $v_{BVI} = 4\text{km/h}$ ,  $v_{BVI} = 6.5\text{km/h}$ ,  $v_{BVI} = 9\text{km/h}$  and  $v_{BVI} = 12\text{km/h}$ . For each panel, we can find in blue the Fourier transform of 1<sup>st</sup> Configuration, in red the 2<sup>nd</sup> Configuration, and in yellow the 3<sup>rd</sup> Configuration.

## FFT: Tester 4

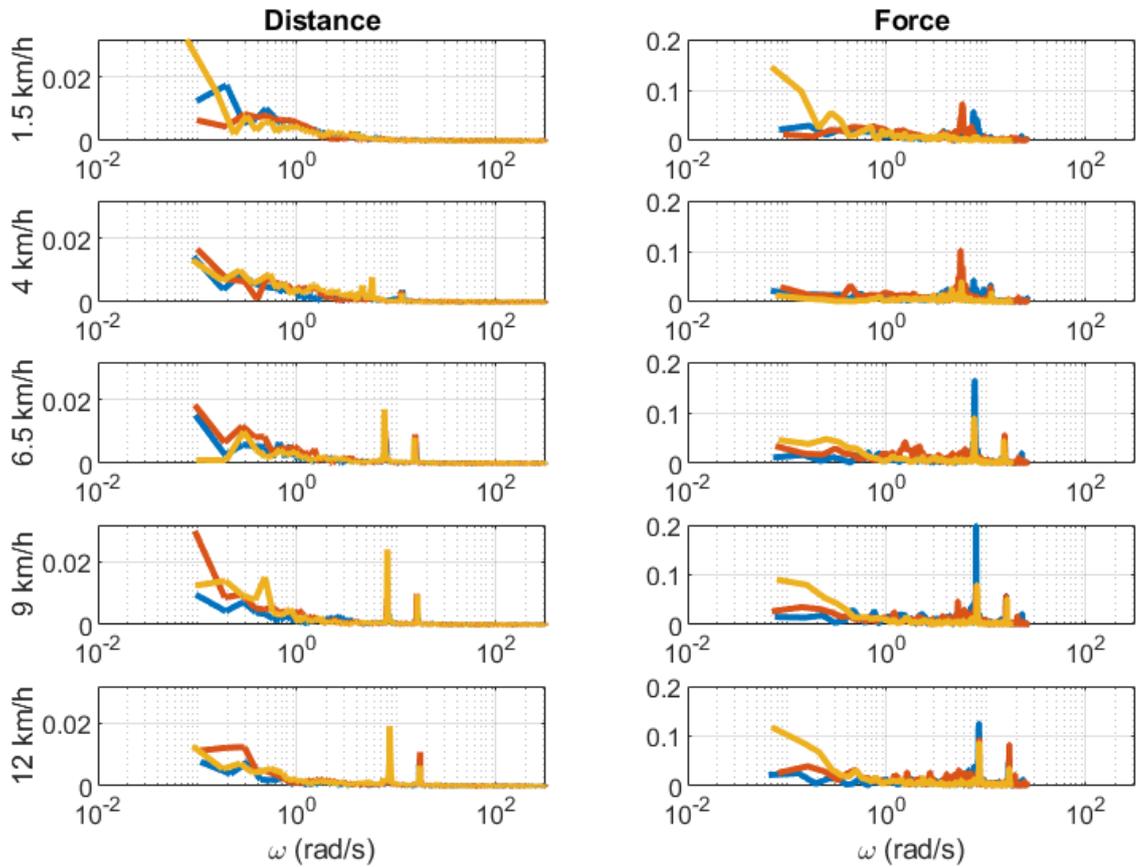


Figure 5.12: Fourier transforms of the hand-table or hand-Yape distance (on the left) and of the force  $F_{BVI}$  (on the right) for different treadmill velocities (from the top to the bottom):  $v_{BVI} = 1.5\text{km/h}$ ,  $v_{BVI} = 4\text{km/h}$ ,  $v_{BVI} = 6.5\text{km/h}$ ,  $v_{BVI} = 9\text{km/h}$  and  $v_{BVI} = 12\text{km/h}$ . For each panel, we can find in blue the Fourier transform of 1<sup>st</sup> Configuration, in red the 2<sup>nd</sup> Configuration, and in yellow the 3<sup>rd</sup> Configuration.

Regarding Configurations 1 and 2, both exhibit effective attenuation primarily at low frequencies, up to approximately 2 rad/s. However, at higher frequencies, noticeable peaks indicate less effective attenuation. As the treadmill velocity increases, the smart tether system's ability to attenuate disturbances diminishes due to saturation effects. These limitations can only be mitigated by employing a more efficient motor with less restrictive saturation limits.

Since the new motor overcomes these limitations, Configuration 3 demonstrates significant attenuation of frequency components compared to the other configurations at almost all tested speeds, indicating a more stable response with fewer oscillations. For instance, at

speeds of 1.5 km/h and 4 km/h, Configuration 3 shows lower peaks in the low frequencies (0-10 rad/s) compared to Configurations 1 and 2.

In some cases, e.g. the test conducted with Tester 1 (Figure 5.9), the amplitude of the yellow peaks (Configuration 3) is notably higher. This increase can be attributed to the enhanced system responsiveness, which provided users with greater freedom of hand movement and reduced their apprehension about causing potential damage.

Regarding the force  $F_{BVI}$ , Configurations 1 and 2 exhibit good attenuation up to around 5 rad/s. Beyond this frequency, they show more pronounced peaks in the high-frequency components, especially at higher speeds, indicating less responsiveness in the force control exerted on the user, primarily due to the motor's limitations. Configuration 3, however, effectively attenuates the majority of the spectral components. In fact, although peaks are still present, this configuration shows a widespread reduction in high-frequency components compared to the other configurations, suggesting more precise and stable force control. For example, at higher speeds (9 km/h and 12 km/h), Configuration 3 maintains lower force peaks, particularly in the frequencies between 10 and 100 rad/s.

Overall, Configuration 3 offers superior performance in terms of responsiveness, effectively addressing the limitations observed in Configurations 1 and 2 by utilizing a more efficient motor capable of handling higher frequencies without saturation effects.

## Part II

### Distance control



# 6 | Distance feedback control system

## 6.1. Distance controller set-up

To ensure smooth operation and user safety, a dedicated distance controller regulates the longitudinal dynamics of BUDD-e. This controller maintains the distance  $d$  from the user as close as possible to a reference distance  $d_0$ , while adapting its speed  $v$  to match the user's velocity  $v_{BVI}$ .

The distance control system relies on accurate distance measurements and appropriate user velocity estimates at each time instant.

The estimation of the distance and velocity of an obstacle in the environment can be obtained using various exteroceptive sensors. Any sensor capable of measuring distance can be adapted for this purpose, e.g., radars, LiDARs, and cameras are the most commonly used ones.

### 6.1.1. LiDARs

LiDAR (Light Detection and Ranging) is a remote sensing technology that uses laser light to measure distances and create detailed and high-resolution maps of environments. This technology is widely used in various fields, including autonomous vehicles, geography, forestry, and environmental monitoring.

A LiDAR system emits laser pulses and measures the time it takes for each pulse to travel to an object and back to the sensor. By calculating this time delay, the system can determine the distance to the object with high precision. The basic components of a LiDAR system include a laser source, a scanner and a photodetector.

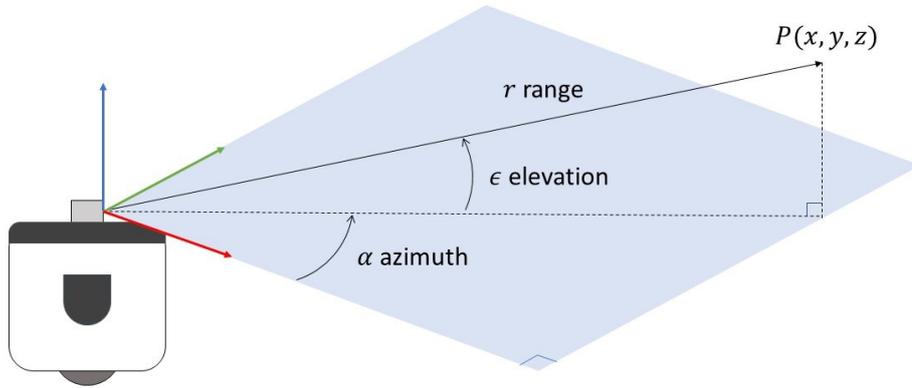


Figure 6.1: LiDAR spherical coordinates

Each pulse that is reflected generates a point in space. The environment is constituted by the collection of all these points, which takes the name of point cloud. From the point cloud, it is possible to reconstruct extremely accurate 3D maps of the environment. Points are usually measured in spherical coordinates, namely range  $r$ , elevation  $\epsilon$  and azimuth  $\alpha$  (see Figure 6.1) converted in  $(x, y, z)$  space coordinates according to the following equation.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r \cos(\alpha) \cos \epsilon \\ r \sin(\alpha) \cos \epsilon \\ r \sin \epsilon \end{bmatrix} + n \quad (6.1)$$

In 6.1,  $n$  is a source of noise due to many possible factors, such as uncertainty in measuring the time of arrival, uncertainty in measuring the exact orientation, interaction with the target (e.g., since black objects absorb the light, not enough of the light gets back to the sensor), and the Jitter effect (due to the relative movement of objects during the time required for a full scan).

The target tracking in Yape is performed using a Robosense R16 LiDAR (see Figure 6.2) mounted on the top lid. Here are some specifications of the LiDAR:

- Accuracy of  $\pm 2$  cm.
- Detection range of 150 m (High-resolution).
- Horizontal Field of View (FOV) of  $360^\circ$ .
- Vertical Field of View of  $30^\circ$ .

- Measure 300,000 points per second.



Figure 6.2: Robosense R16 LiDAR

### 6.1.2. Target Tracking

Yape is capable of tracking multiple dynamic objects simultaneously, a process known as Multiple Target Tracking (MTT). The application of MTT is inherently complex because each LiDAR scan generates numerous measurement points for every obstacle. It is not possible to determine in advance the number of obstacles or how many measurements correspond to each one, a challenge referred to as Extended Target Tracking (ETT).

To manage this complexity, a cascade ETT algorithm is employed to facilitate the tracking of multiple obstacles. The main functionality of this algorithm is summarized as follows:

- Point Cloud Processing: The 3D point cloud (see Figure 6.4) is processed to create a 2D grid representation of the environment. This conversion reduces computational complexity and is more suitable for vehicle navigation in a two-dimensional space.
- Dynamic Object Extraction: Dynamic objects must be distinguished from background obstacles.
- Low-level Target Tracking: The target tracking algorithm is applied to each data point individually.
- Clustering: Measurements are grouped based on their relative position and speed to form clusters, with each cluster representing an individual object within the

LiDAR's range.

- High-level Target Tracking: Finally, a second target tracking algorithm is applied to the clustered measurements. The output of this algorithm is an estimate of the position and speed of the entire obstacles in the scene.

This approach simplifies the extended target tracking problem to a standard target tracking problem. This technique does not assume any specific shape for the detected obstacles, making it capable of simultaneously tracking objects of various sizes, such as pedestrians and cars. However, it is important to note that this process requires significant computational resources.

## Point cloud processing

The data processing steps are as follows:

- Measurements taken when the laser beam hits the ground can be misinterpreted as obstacles. These erroneous measurements are removed from the point cloud using a RANSAC plane fitting algorithm.
- LiDAR's position is dependent on Yape therefore detected objects can appear to move in a manner opposite to the vehicle's motion. This apparent motion is corrected through a scan-matching algorithm.
- The point cloud becomes sparser with increasing distance from the sensor. To address this, the cell values are normalized based on the maximum number of points each cell can potentially contain.
- To convert from a 3D to a 2D representation, a grid representation is constructed.

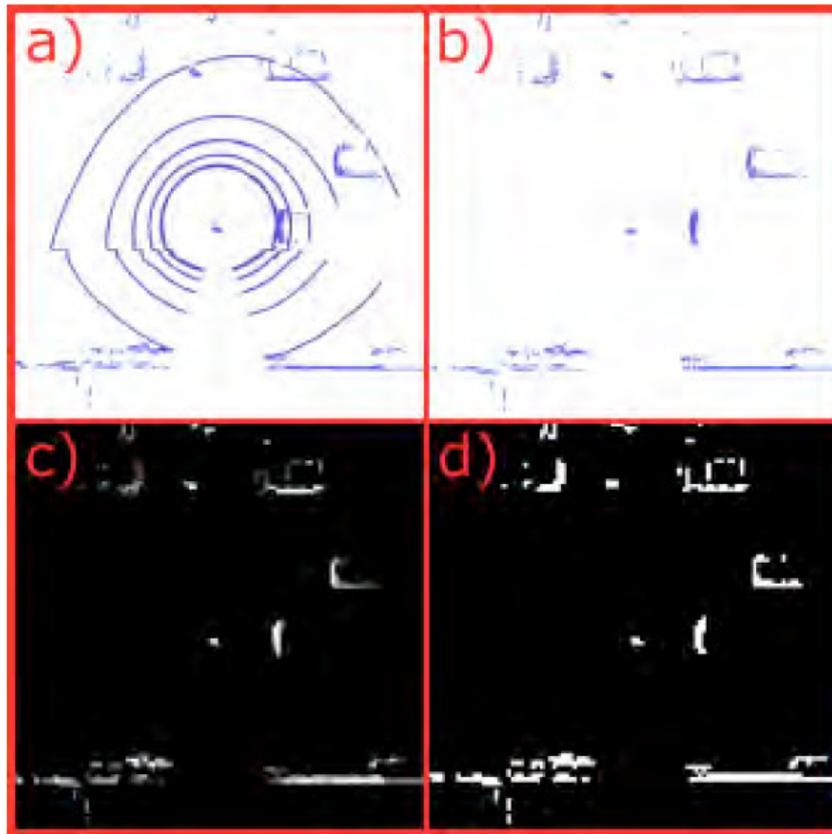


Figure 6.3: Point cloud preprocessing: a) original image, b) background removal, c) grid quantization, d) grid binarization

### Dynamic Object Extraction

- The background is filtered out from the measurements. This is achieved by the LiDAR computing a threshold moving average from previous data points. If a pixel remains occupied for a certain duration, it is classified as background.
- The resulting image is then processed to eliminate noisy pixels and group together pixels that are likely part of the same object.
- Obstacles can cast shadows, causing objects behind them to disappear and reappear. Additionally, moving objects can degrade background removal performance. To address these issues, the shadow zone for each obstacle is estimated, and points passing through these zones are marked as constants until they become visible again.

### Low-level Target Tracking

- A Kalman filter is utilized to predict the movement of particles.

- A gating operation is performed to select measurements that are potentially compatible with each track. Note that each measurement can be associated with multiple tracks. Following this, an association problem is addressed where each track is matched with the most likely measurements using a Global Nearest Neighbor algorithm. In this step, only one track is associated with each measurement.
- A track manager is implemented to handle the detection of new objects, the disappearance of objects, and possible tracking errors.

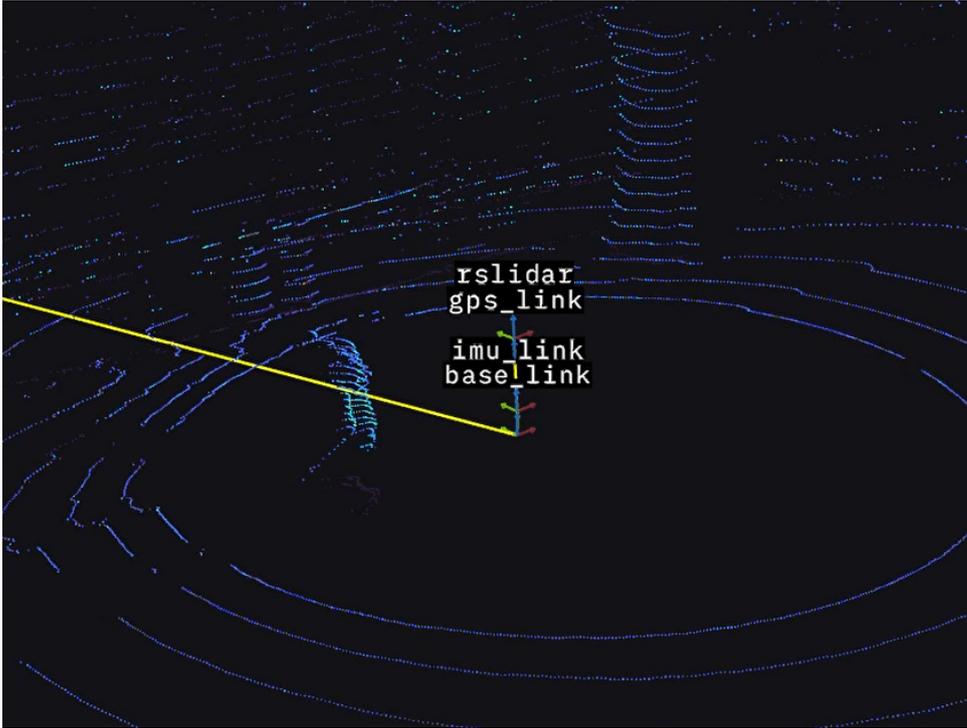


Figure 6.4: LiDAR detecting different objects (the surrounding wall, a tree, a person)

### 6.1.3. Target tracking on BUDD-e

In BUDD-e, the LiDAR is used (besides for obstacle avoidance) for the measurement of the user distance  $d(t)$ . In particular, the nearest object detected at the back of Yape is considered as the user, and the target distance  $d(t)$  is calculated as

$$d(t) = \frac{v_{\text{beam}} \Delta t(t)}{2} \quad (6.2)$$

In 6.2,  $v_{\text{beam}}$  is the velocity of the beam, assumed known, and  $\Delta t(t)$  is the time measured by the LiDAR between the laser beam emission and its comeback. By analyzing the variation in distance calculated in (6.2) and knowing Yape's velocity, it is straightforward

to estimate the user's velocity  $v_{BVI}$ , which will be useful in Chapter 6.2 as an estimation of the random disturbance acting on the model.

## 6.2. The Yape longitudinal speed dynamical model identification

The longitudinal velocity of Yape is regulated by a dedicated embedded control system, with  $v_{ref}$  as the input and  $v$  as the output. For this reason,  $v_{ref}$  is regarded as a controllable input. The distance control system aims to maintain the robot-user distance  $d$  as close as possible to a reference value  $d_0$ , despite potential random disturbances acting on the model, which correspond to the user's speed  $v_{BVI}$ . This speed can be estimated by the target tracking algorithm using LiDAR data as shortly discussed in Chapter 6.1.3.

The identification of Yape's dynamics conducted in [14] revealed a delay in the acceleration dynamics that needed to be considered when designing an appropriate controller. Further investigation indicated that the true source of the delay was the detection and sampling of data. As a result, data collection procedures were corrected, and an accurate model, free from delay, was identified. The new identification of the Yape dynamics was performed through step experiments at various speeds, as illustrated by the data in Figure 6.6. These experiments reveal that Yape operates in two distinct modes, depending on whether it is accelerating or decelerating. As clear from figure 6.6, the acceleration dynamics has a larger settling time compared to the deceleration one.

- During acceleration, Yape's longitudinal dynamics can be described by the transfer function  $F_{acc}(s)$ , i.e.,

$$F_{acc}(s) = \frac{1 - 0.3423s}{1 + 2.3728s + 0.9681s^2} \quad (6.3)$$

- When Yape is decelerating, its longitudinal dynamics can be represented by the transfer function  $F_{dec}(s)$ , i.e.,

$$F_{dec}(s) = \frac{1 - 0.4255s}{1 + 0.6187s + 0.2059s^2} \quad (6.4)$$

Notice that both acceleration and deceleration modes are characterized by two asymptotically stable poles and a non minimum-phase zero. Also, delay is not part of transfer function (6.3), contrarily to the transfer function identified in [14].

In Figure 6.6 we show the comparison of the real velocity with the one obtained by feeding

the switching model with the input  $v_{ref}$  used in the experiment. The switching condition used here is extremely simple and illustrated in Figure 6.5. It depends upon the step-wise input provided to the system.

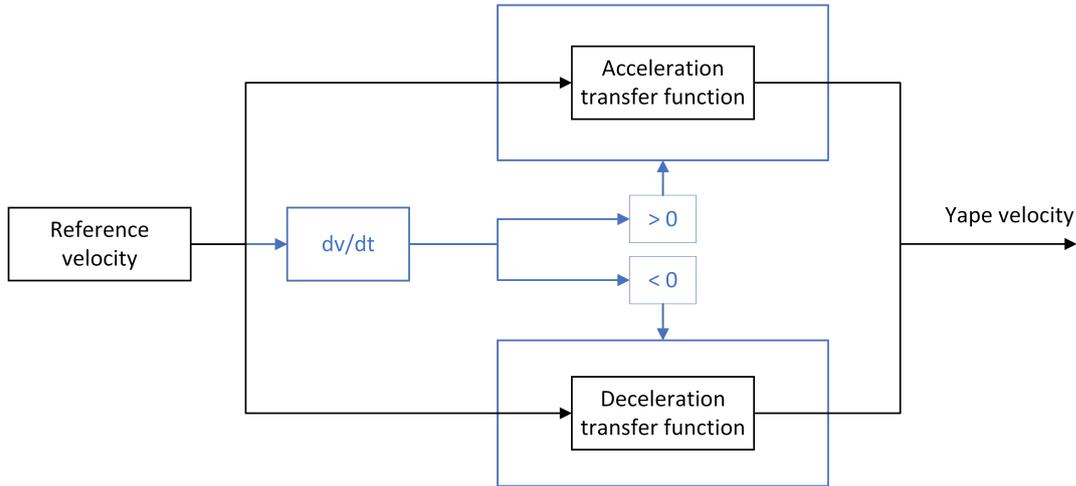


Figure 6.5: Yape switching model

Figure 6.6 shows that, when the commutation condition is correctly identified, the switching model is capable of capturing the system behavior precisely.

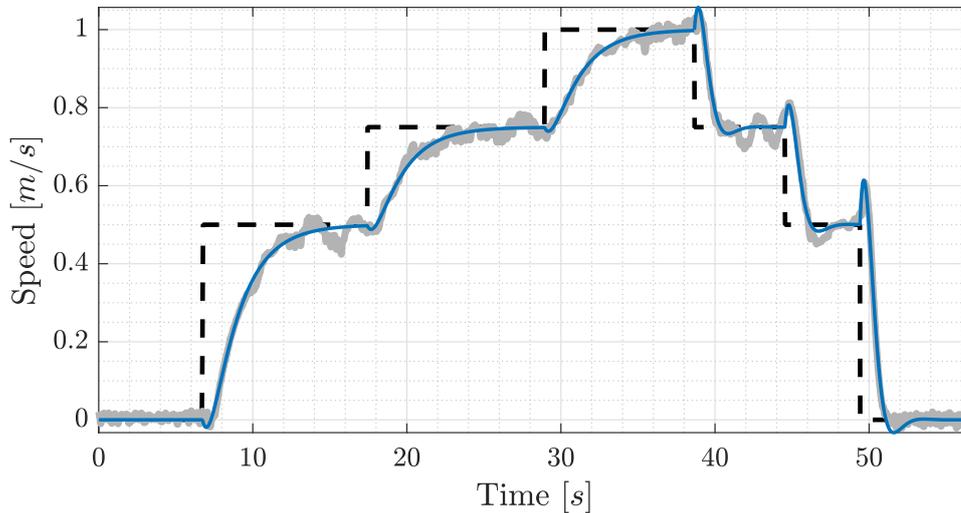


Figure 6.6: Example of the data used to identify the longitudinal dynamical model of Yape. Black dashed line:  $v_{ref}$ ; gray solid line: measured Yape velocity  $v$ ; blue solid line: simulated Yape velocity  $v$ .

However, for more complex input signals, it was not possible to precisely identify the switching condition between these two modes of operation. As a result, implementing a

switching or time-varying control law could be problematic. Therefore, in the next section, we derive a time-invariant control law using LMI arguments, which is robust against the specific mode of operation and the transitions between them.

### 6.2.1. The Yape Longitudinal Speed Discrete-Time Switching Model

For LMI-based control design, a discrete-time state-space realization of the system model will be derived. To do so, we define

$$u(t) = a_{ref}(t) = \dot{v}_{ref}(t) \quad (6.5)$$

as new "virtual" input. Also, the system state vector is defined as follows

$$x(t) = [v_{ref}(t) \ p(t) \ v(t) \ a(t)]^T \quad (6.6)$$

where  $a(t) = \dot{v}(t)$  and  $p(t)$  is the robot curvilinear abscissa. Recalling that the transfer function between  $v_{ref}(t)$  and  $v(t)$ , for both  $i = acc, dec$  has the form

$$F_i(s) = \frac{1 + sT_i}{1 + \alpha_i s + \beta_i s^2} \quad (6.7)$$

we can write

$$\begin{cases} \dot{v}_{ref} = u(t) \\ \dot{p}(t) = v(t) \\ \dot{v}(t) = a(t) \\ \dot{a}(t) = \frac{1}{\beta_i}(-\alpha_i a(t) - v(t) + v_{ref}(t) + T_i a_{ref}(t)) \end{cases} \quad (6.8)$$

System (6.8) can be written in compact form as

$$\begin{bmatrix} \dot{v}_{ref}(t) \\ \dot{p}(t) \\ \dot{v}(t) \\ \dot{a}(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{1}{\beta_i} & 0 & -\frac{1}{\beta_i} & -\frac{\alpha_i}{\beta_i} \end{bmatrix} \begin{bmatrix} v_{ref}(t) \\ p(t) \\ v(t) \\ a(t) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ \frac{T_i}{\beta_i} \end{bmatrix} u(t) \quad (6.9)$$

This model can be discretized, with a sampling time  $T_s = 0.1s$ , obtaining

$$x_{k+1} = A_i x_k + B_i u_k \quad (6.10)$$

### 6.2.2. $\mathcal{H}_2$ Controller Design

For controller design, we apply to the switching system (6.10) an  $\mathcal{H}_2$  norm minimization approach similar to the one discussed in [5]. Before defining it, we first derive the controller in the case of regulation, i.e., where  $x_k = 0$  is the desired steady state and  $u_k = Kx_k$  is the control law. The application of this approach requires the definition of the performance output

$$z_k = C_p x_k + D_p u_k \quad (6.11)$$

Note that, for LTI systems, in order to make the  $\mathcal{H}_2$  approach essentially equivalent to the design of an LQ controller, matrices  $C_p$  and  $D_p$  must be properly defined as

$$C_p = \begin{bmatrix} \sqrt{Q} \\ 0 \end{bmatrix}, \quad D_p = \begin{bmatrix} 0 \\ \sqrt{R} \end{bmatrix} \quad (6.12)$$

where  $Q$  and  $R$  are the weighting matrices commonly used in the LQ control cost. In this chapter we minimize the  $\mathcal{H}_2$  cost by considering the dynamics of the system in both operating modes, i.e., we solve

$$\inf_{P,K} \text{trace}((C_p + D_p K)P(C_p + D_p K)^T) \quad (6.13)$$

subject to the constraint

$$(A_i + B_i K)P(A_i + B_i K)^T - P + I < 0 \quad (6.14)$$

for both  $i = acc, dec$ . The latter is done, first by introducing the new optimization variable  $S$  fulfilling

$$S \geq (C_p + D_p K)P(C_p + D_p K)^T \quad (6.15)$$

Then, we define a new gain matrix  $L = KP$ . In this way, following the same method described in Chapter 3, by resorting to the Schur complement [4], the optimization problem above can be formulated as the LMI

$\min_{P,S,L} \text{trace}(S)$  subject to

$$\begin{bmatrix} S & C_p P + D_p L \\ (C_p P + D_p L)^T & P \end{bmatrix} \geq 0 \quad (6.16)$$

$$\begin{bmatrix} P - A_i P A_i^T & P - A_i L^T B_i \\ -B_i L A_i^T - I & B_i L \end{bmatrix} \geq 0 \quad (6.17)$$

for  $i = acc, dec$ .

### 6.3. Distance Controller

The control law designed in this chapter has the goal of making the robot speed  $v$  adapt to the user's speed  $v_{BVI}$  and, at the same time, to steer the robot curvilinear abscissa  $p$  to  $p_{BVI} + d_o$ , i.e., to  $d_o$  ahead with respect to the user. Note that  $p - (p_{BVI} + d_o) = d - d_o$ : indeed, the latter goal is equivalent to steer the robot-user distance  $d$  to the reference value  $d_o$ . In light of this discussion, the control law derived in the previous section is modified as follows.

$$u_k = a_{BVI,k} + K(x_k - x_{o,k}) \quad (6.18)$$

where the reference state vector  $x_{o,k}$  is defined as:

$$x_{o,k} = [v_{BVI,k} \ p_{BVI} + d_o \ v_{BVI,k} \ a_{BVI,k}]^T \quad (6.19)$$

and where  $a_{BVI,k}$  denotes the acceleration of the user. Denoting with  $k_j$ ,  $j = 1, \dots, 4$  the elements of the gain row vector  $K$ , and according to the control law defined in (6.18), we have to set, at each time instant,

$$a_{ref,k} = k_1(v_{ref,k} - v_{BVI,k}) + k_2(d_k - d_k^o) + k_3(v_k - v_{BVI,k}) + k_4(a_k - a_{BVI,k}) \quad (6.20)$$

For practical implementation, we approximate  $a_{ref,k}$ ,  $a_k$ , and  $a_{BVI,k}$  using the implicit Euler discretization method to avoid stability issues from the regulator equation, and we obtain:

$$\begin{aligned} v_{ref,k} - v_{BVI,k} = & v_{ref,k-1} - v_{BVI,k-1} + T_s(k_1(v_{ref,k} - v_{BVI,k}) + k_2(d_k - d_k^o) \\ & + (k_3 + \frac{k_4}{T_s})(v_k - v_{BVI,k}) - \frac{k_4}{T_s}(v_{k-1} - v_{BVI,k-1})) \end{aligned} \quad (6.21)$$

In view of this, the real input is, at each time instant, set to

$$v_{ref,k} = v_{BVI,k} + \frac{1}{1 - k_1 T_s} ((v_{ref,k-1} - v_{BVI,k-1}) + k_2 T_s (d_k - d_k^o)) + (k_3 T_s + k_4)(v_k - v_{BVI,k}) - k_4(v_{k-1} - v_{BVI,k-1}) \quad (6.22)$$

### 6.3.1. Integral Action

The control action developed in the previous section may not display good performance in case of uncertainties in the estimation of the user speed  $v_{BVI,k}$ . For this reason, the control law can be robustified thanks to the use of a suitable integrator with anti-windup, placed in parallel with respect to the previous controller. In this way, the control action is computed as  $a_{ref,k} = k_1(v_{ref,k} - v_{BVI,k}) + k_2(d_k - d_k^o) + k_3(v_k - v_{BVI,k}) + k_4(a_k - a_{BVI,k}) + k_5 i_k$ , where

$$i_{k+1} = \text{sat}_{v_{MAX}/(3|k_5|)}(i_k + T_s(d_k - d_k^o)) \quad (6.23)$$

Note that,  $\text{sat}_{\bar{x}}(x) = \min(\max(x, -\bar{x}), \bar{x})$ . This results in the following control law

$$v_{ref,k} = v_{BVI,k} + \frac{1}{1 - k_1 T_s} ((v_{ref,k-1} - v_{BVI,k-1}) + k_2 T_s (d_k - d_k^o)) + (k_3 T_s + k_4)(v_k - v_{BVI,k}) - k_4(v_{k-1} - v_{BVI,k-1}) + k_5 T_s i_k \quad (6.24)$$

## 6.4. Experimental validation

The controller developed in this section was implemented on the real system and subjected to testing. These tests were conducted without using the Smart Tether System and aimed at evaluating the performance of the controllers described in this section (Controller 1 and Controller 2 refer to (6.22) and (6.24), respectively). The initial tests (see Figures 6.7 and 6.8) were performed with the user in a fixed position, while Yape's reference distance changed over time in a step-wise manner.

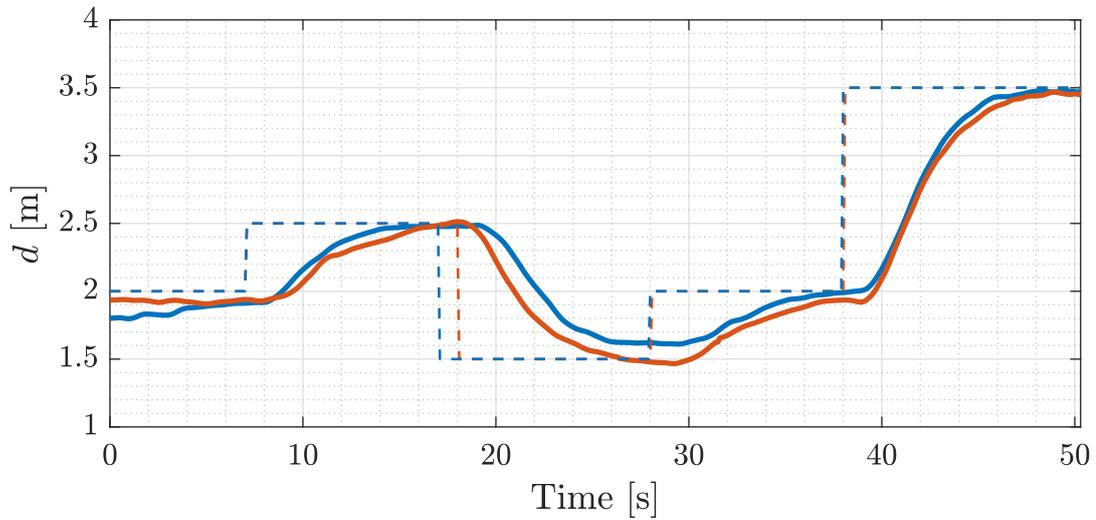


Figure 6.7: Response (i.e., measured user-robot distance  $d$ ) to step-wise inputs of the distance control systems designed: (6.22) (blue line) and (6.24) (red line). Dashed lines: reference distances  $d_o$ .

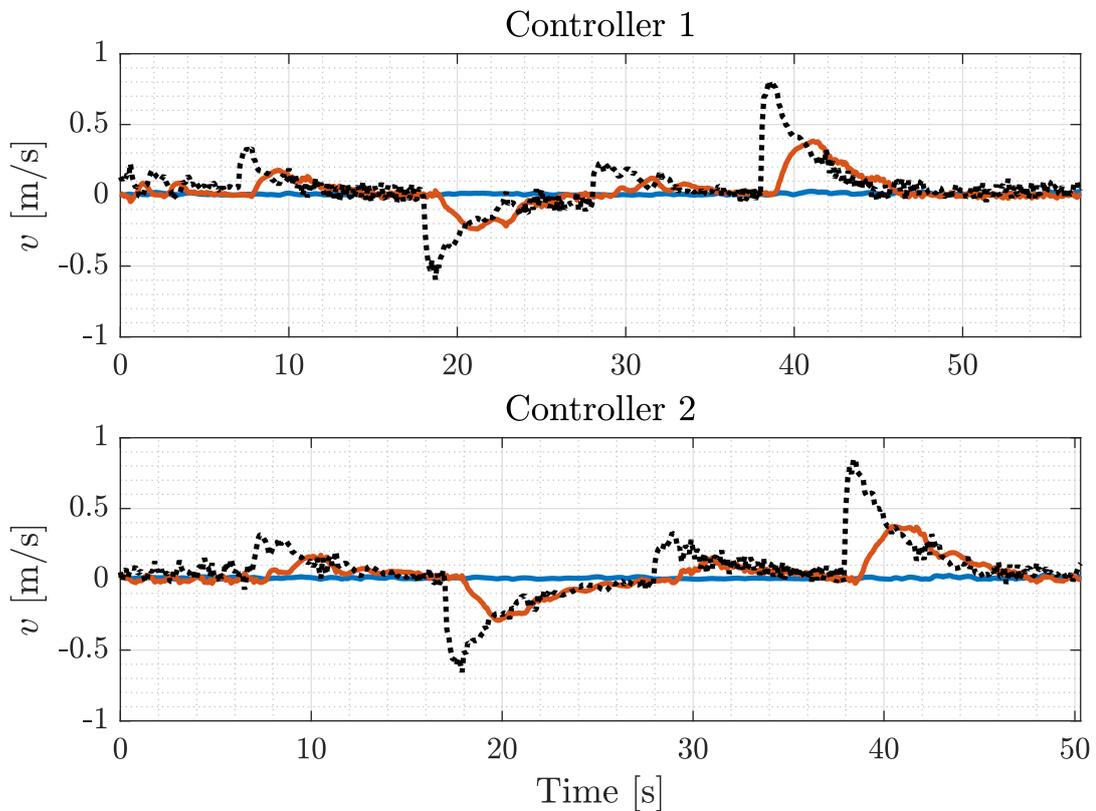


Figure 6.8: Response to step-wise inputs of the distance control systems 6.22 (Controller 1), and 6.24 (Controller 2). For all panels we show: reference Yape velocity  $v_{ref}$  (black dashed line); Yape velocity  $v$  (red line); user estimated velocity  $v_{BVI}$  (blue line).

As observed in Figure 6.7, the  $\mathcal{H}_2$ -norm minimizing controllers exhibit a generally smooth behavior, quickly achieving a steady-state condition. Controller 2, in particular, which includes integral action, demonstrates superior static performance. In the second test, the user moved at a low speed (approximately 0.5-1 m/s) while the reference distance between Yape and the user remained constant. The results of this test are shown in Figure 6.9. Note that Figure 6.9 cannot be used for comparison purposes, since the user velocities in the two scenarios, as depicted in the bottom panels, display significantly different dynamic behaviors.

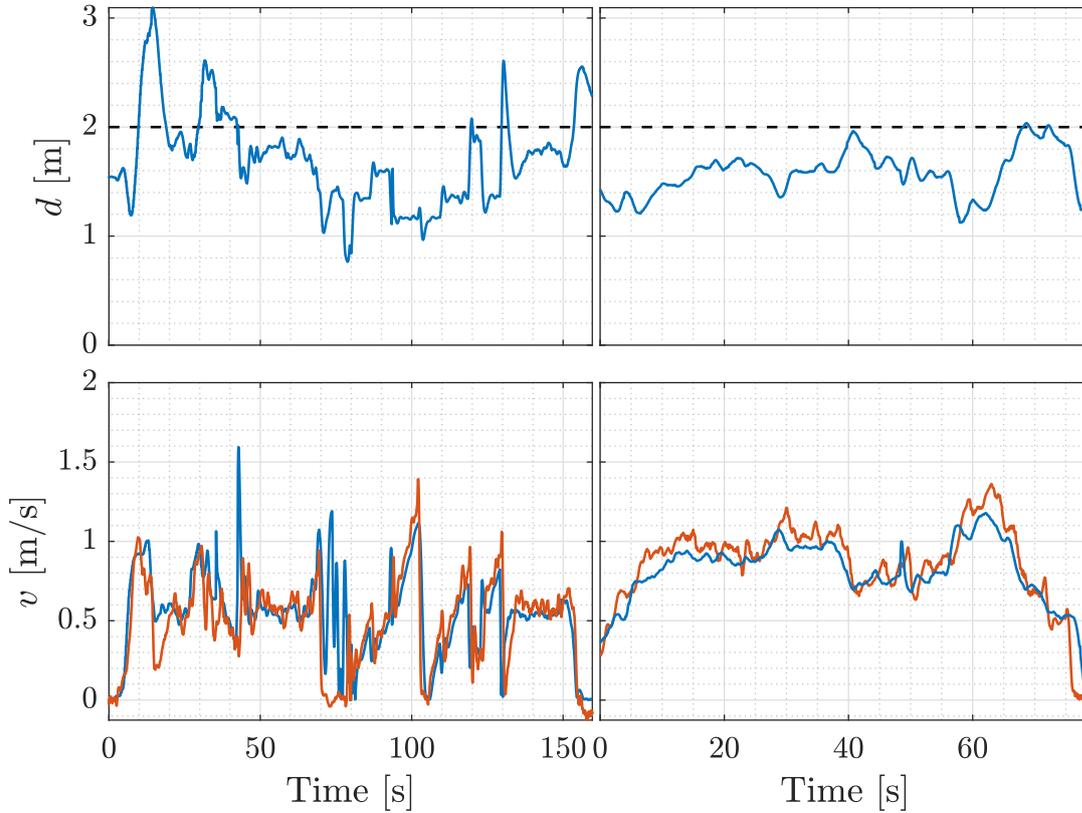


Figure 6.9: Response to varying user speed of the distance controllers (6.22) (Controller 1, left panel) and (6.24) (Controller 2, right panel). Top panels: reference distance  $d_o$  (black dashed line); measured distance  $d$  (blue solid line). Bottom panels: Yape velocity  $v$  (red line); user estimated velocity  $v_{BVI}$  (blue line).

The presented results demonstrate that the controllers developed in this section exhibit good both dynamic and static performance, making them suitable for deployment in real-case scenarios.

A significant consideration is in order. The results shown in this section indicate that, when the user experiences a gradual change in speed, whether accelerating or decelerating,

performance remains good and the distance lies within an acceptable range. A slight delay in detection is generally tolerated. However, problems occur when the user experiences a significant change in speed, such as a sudden stop while running. In these scenarios, Yape executes a smooth stop, halting at a greater distance from the user. This is because the velocity is derived from the target tracking module, making it susceptible to inherent delays. The issue described above is addressed in Chapter 7.



# 7 | Distance controller in braking conditions

In the previous chapter a new issue was highlighted: Yape executes smooth stops not only when the user slowly decreases his/her speed, but also in case the user performs a sudden stop while running. In this case, however, the robot halts at an unacceptable distance from the user. The main source of this problem was identified in the fact that the velocity is derived from the target tracking module, making it susceptible to inherent delays.

To solve this problem and avoid critical situations, Yape must swiftly detect the user's intent to stop and respond promptly. Due to delays, relying solely on velocity data from the target tracking module could lead to unacceptable results. The data collected from the smart tether system proves valuable, as they allow for rapid identification of the user's intentions through the force exerted on the designed cable. This can result in a precise and immediate control of the vehicle.

## 7.1. Braking control strategy

The braking control approach we propose consists of two steps:

- Braking recognition: braking recognition is approached as a fault detection problem, where the nominal state represents the user's normal behavior and the faulty state corresponds to the braking action that leads to the user's stop. A method inspired by multivariate analysis was implemented on the smart tether data to correctly identify these states.
- Feedforward braking: the sudden braking system should not rely on feedback from the system state measurement (like traditional feedback control systems). Instead, it should act proactively when a user brake is detected. This can minimize the reaction time and allow the robot to stop almost simultaneously with the user. By implementing a feedforward brake, the robot can enhance safety and comfort for the user.

The following section describes the approach used to identify the braking condition, while the forward braking operation will be designed and implemented in future work.

## 7.2. Multivariate analysis

The approach used in this work is inspired by multivariate analysis. Multivariate analysis encompasses a set of statistical techniques used to identify patterns and relationships among multiple variables simultaneously. This form of analysis is critical when multiple variables interact in complex ways, as discussed in [13].

Consider a dataset known as the training set, which captures the nominal operating conditions of a process. Let the training set be constituted by  $n$  observations of  $m$  process variables  $x_i(k)$ ,  $k = 1, \dots, N$ , and  $i = 1, \dots, m$ . We define the vector  $x(k)$  as

$$x(k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_m(k) \end{bmatrix} \quad (7.1)$$

### 7.2.1. Preliminary steps

In order to extract useful information from the data of the training set, it is necessary to perform the following operations:

- Removal of largely incorrect measurements (outliers), or measurements referred to different operating conditions.
- Unit variance normalization of the variables to avoid the effects of different measurement units.

The unit variance normalization is done variable-wise, each variable with its mean and variance. In our work, however, this step will not be performed as discussed below.

### 7.2.2. Mahalanobis distance

Consider the training set defined possibly normalized data with covariance matrix  $S$ . The latter is computed as

$$S = \frac{1}{N-1} \sum_{k=1}^N (x(k) - \mu_x)(x(k) - \mu_x)^T \quad (7.2)$$

where  $\mu_x = \frac{1}{N} \sum_{k=1}^N x(k)$ .

At time  $k$ , the statistical distance (Mahalanobis distance)  $T^2$  of the measurement vector  $x(k)$  from the training set is defined as:

$$T^2(k) = x(k)^T S^{-1} x(k) \quad (7.3)$$

$T^2$  is characterized by a  $F$  distribution since the expected variance is estimated from data. Therefore the confidence region is defined by:

$$T_\alpha^2 = \frac{(m)(n-1)}{(n-m)} F_\alpha(m, n-m) \quad (7.4)$$

where  $F_\alpha(m, n-m)$  is the upper  $100\alpha\%$  of the  $F$  distribution with  $m$  and  $n-m$  degrees of freedom. Given a significance level  $\alpha$ , the ellipsoidal confidence region is defined by:

$$T^2 \leq T_\alpha^2 \quad (7.5)$$

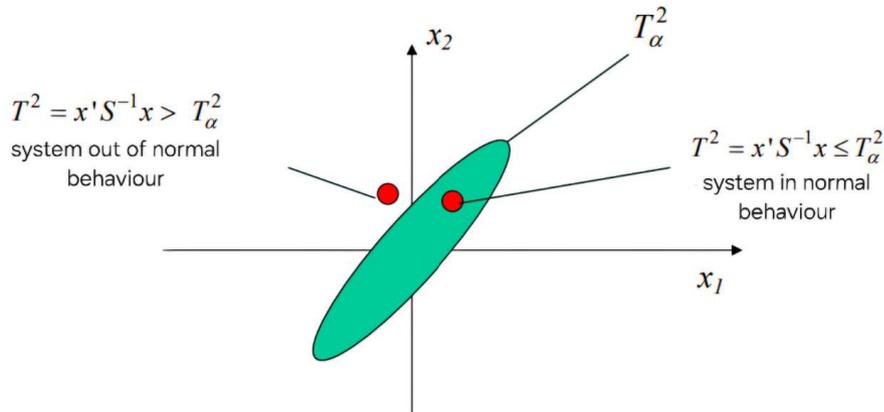


Figure 7.1: Ellipsoidal confidence region

Therefore, in case  $x(k)$  is unbiased, the statistical test we perform is the following

- If  $T^2(k) \leq T_\alpha^2$  we declare the system in normal behavior at time  $k$ .
- If  $T^2(k) > T_\alpha^2$  we declare the system out of normal behavior at time  $k$ .

Note that the computation of the confidence region is valid as long as the sampled variances match the actual ones.

### 7.3. Braking recognition in BUDD-e

Braking detection is performed in this work using a method inspired by the multivariate analysis described in the previous section. Since many possible variables can be used for multivariable analysis, we will compare the results obtained using different sets of variables: by comparing the results, the optimal set of variables to be used for detection which allows to minimize the probability of false positives and the detection delay will be selected for each of the configurations described in Part I of this thesis. Specifically, instead of computing the thresholds  $T_\alpha$  as functions of the probability  $\alpha$ , we will evaluate, using the available experimental data, the performance of the detection algorithm as a function of the threshold values. This will be done, for all available smart tether system configurations and for different user speeds. In this way, we will evaluate the optimal threshold values to be used in the proposed stop detection scheme, for each user speed. To evaluate the performance of the stop detection algorithm we use two (partially conflicting) performance indicators:

- the detection delay, computed as the time distance of the detected brake from the actual braking action of the user. A time delay penalty of 10 seconds for every missed detection was introduced to better visualize the results of each analysis. This decision aims to penalize the threshold values that are unable to correctly detect all the brakings.
- the rate of data points, during non-braking operations, where the value  $T(k)$  exceeds the given threshold value. This rate, later denoted as *threshold exceedance count*, is strictly related to the probability of false positives.

#### 7.3.1. Experimental setup and data

Data were collected during tests where the user stopped multiple times while walking or running on the athletics track at Giuriati Sports Center (as shown in Figure 7.2). During these tests, before the stops the robot speed was set and maintained constant. The test

was repeated at different velocities: 1 m/s, 1.5 m/s, 2 m/s, 2.5 m/s, and 3 m/s. The procedure was repeated for each configuration, i.e.

- Configuration 1: first setup with the control designed in the frequency domain using the frequency domain criterion, see Section 3.2.
- Configuration 2: first setup with the control designed with the  $\mathcal{H}_2$  method, see Section 3.3.
- Configuration 3: new setup, see Chapter 4.



Figure 7.2: Test performed in Milan, on the athletics track at Giuriati Sport Center to gather data for the braking detection procedure

During these tests, data from both Yape and the smart tether system were gathered, and the latter was used for the multivariate analysis. As an example, Figures 7.3, 7.4, 7.5 show the data collected from three tests conducted at a velocity of 1.5 m/s, each with one of the three configurations.

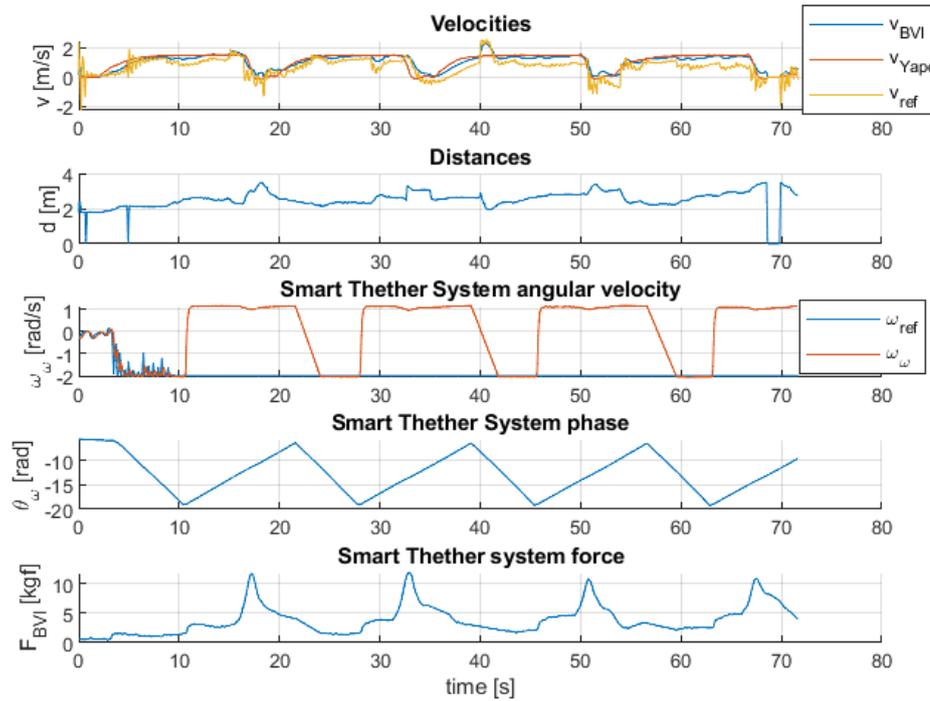


Figure 7.3: Test performed with Configuration 1 and velocity  $v_{BVI} = 1.5$  m/s

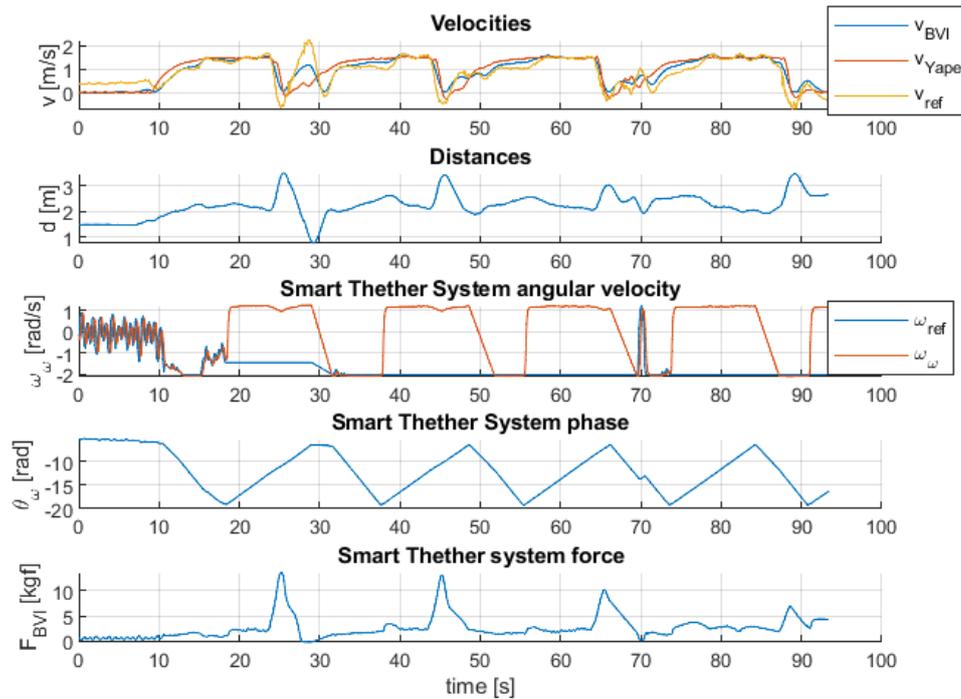


Figure 7.4: Test performed with Configuration 2 and velocity  $v_{BVI} = 1.5$  m/s

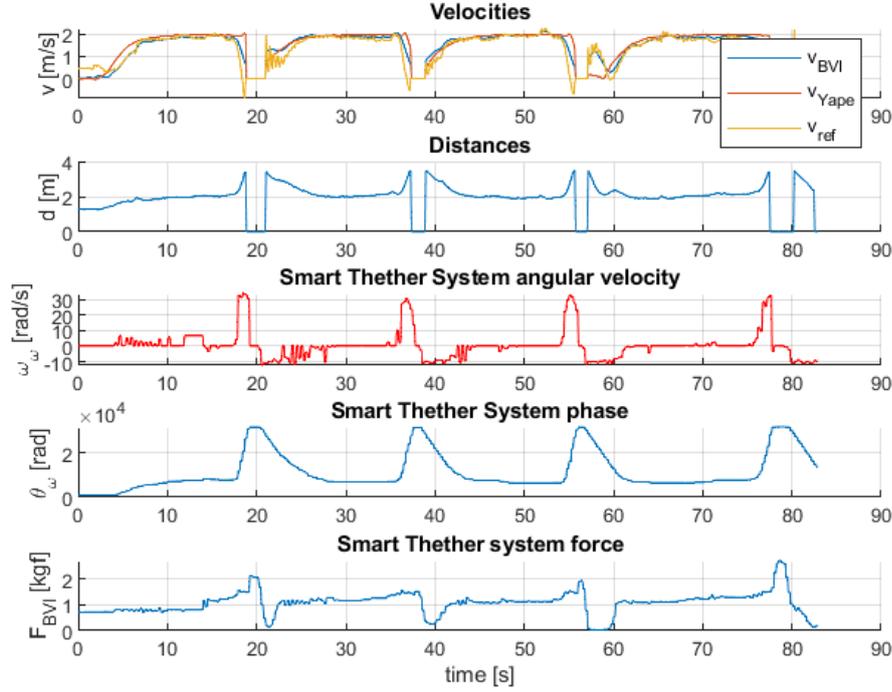


Figure 7.5: Test performed with Configuration 3 and velocity  $v_{BVI} = 1.5$  m/s

Figures 7.3 and 7.4 display the data collected from Configurations 1 and 2, which share the same hardware. These figures underscore a problematic behavior of the smart tether system which is detected for all the velocities. Specifically, the angular velocity of the winch remains constant at the saturation value for that motor. This issue is triggered by the "Too Far" safety condition within the system, which, once activated, sets the winch's angular velocity to retract the user back into the permitted region close to the robot. The "Too Far" condition was activated too frequently because the feedback force was too weak. As a result, the user, who relied solely on the force feedback to navigate the athletics track, did not stay sufficiently close to the robot in order to make the smart tether system work in normal conditions.

In view of this issue, in Configurations 1 and 2, the smart tether system's angular velocity does not provide significant information for braking detection. Therefore this variable will be discarded and, for the latter configurations, the only useful information for braking detection will be the tugging force exerted by the system on the user, i.e.,  $F_{BVI}$ .

### 7.3.2. Braking recognition

For each velocity defined previously in Section 7.3.1, a segment of the test, during which the user is not stopping, is isolated and used as the training set. The training set is used to compute matrix  $S$  as the variance of the selected datum  $x(k)$ , see (7.2).

Here is a summary of the algorithm implemented in Yape.

#### Offline phase:

- Select the most representative variable vector  $x(k)$  for brake detection.
- Collect data in normal operating conditions (without braking) to define the training set according to (7.1).
- Compute the covariance  $S$  of  $x(k)$  according to equation (7.2).

**Online phase:** At any new set of collected measurement  $x(k)$  we compute  $T^2(k) = x(k)^T S^{-1} x(k)$  and compare it to the defined threshold  $T^2$ , which is possibly varying as a function of Yape's speed  $v$ .

To perform the analysis illustrated in Section 7.2 and resumed above, Configurations 1 and 2 use only the force value  $F_{BVI}$  since it is the only one carrying useful information, i.e.,

$$x(k) = F_{BVI}(k) \quad \forall k \quad (7.6)$$

For Configuration 3, instead, we can use the additional information from the winch angular velocity. Hence, the candidate sets of variables for braking detection are

$$\text{Case A: } x(k) = F_{BVI}(k) \quad \forall k \quad (7.7a)$$

$$\text{Case B: } x(k) = \tilde{\omega}_w(k) \quad \forall k \quad (7.7b)$$

$$\text{Case C: } x(k) = \left[ F_{BVI}(k) \quad \tilde{\omega}_w(k) \right]^T \quad \forall k \quad (7.7c)$$

To correctly identify the braking condition, the variable  $\tilde{\omega}_w(k)$  is defined as follows:

$$\tilde{\omega}_w(k) = \max(0, \omega_w(k)) \quad (7.8)$$

In this way,  $\tilde{\omega}_w$  takes nonzero values only if  $\omega_w$  is positive, i.e., in case the user is gaining distance from the robot, e.g., in braking.

### 7.3.3. Experimental results

Data collected during the tests have been manually separated in a "braking set" and a "normal set", displayed in Figures 7.6-7.8. For Configurations 1 and 2, the histograms show the distribution of the samples of  $F_{BVI}(k)$ . On the other hand, in the case of Configuration 3, both force and winch angular velocity were taken into consideration.

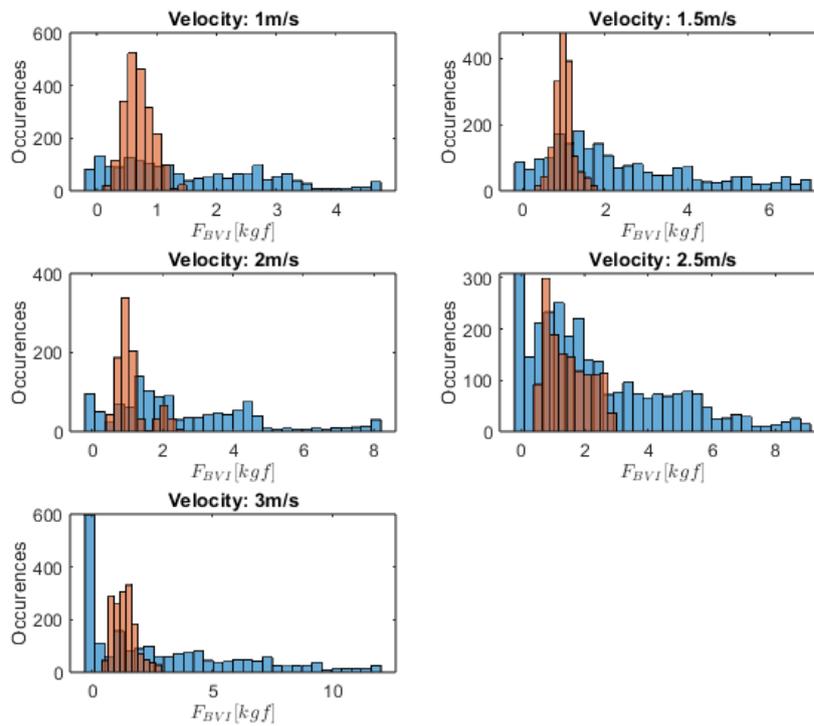


Figure 7.6: Configuration 1: distribution of the force variable in the "normal dataset" in red and in the "braking dataset" in blue.

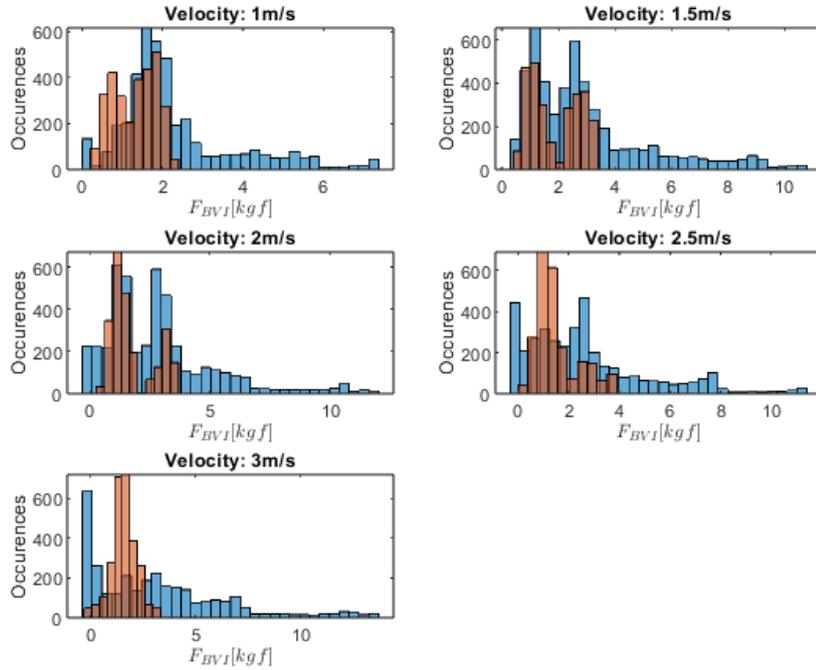


Figure 7.7: Configuration 2: distribution of the force variable in the "normal dataset" in red and in the "braking dataset" in blue.

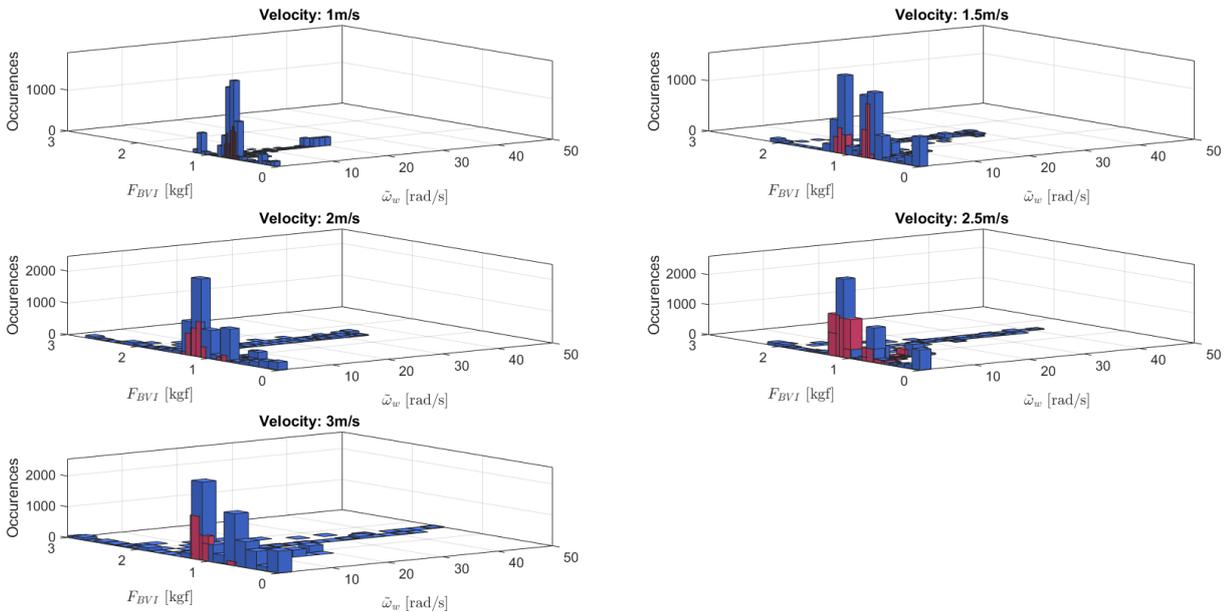


Figure 7.8: Configuration 3: distribution of both the force and the winch angular velocity variables in the "normal dataset" in red and in the "braking dataset" in blue.

The ellipsoidal confidence region defined, e.g., by (7.5) is depicted in Figure 7.9 for Con-

figuration 3.

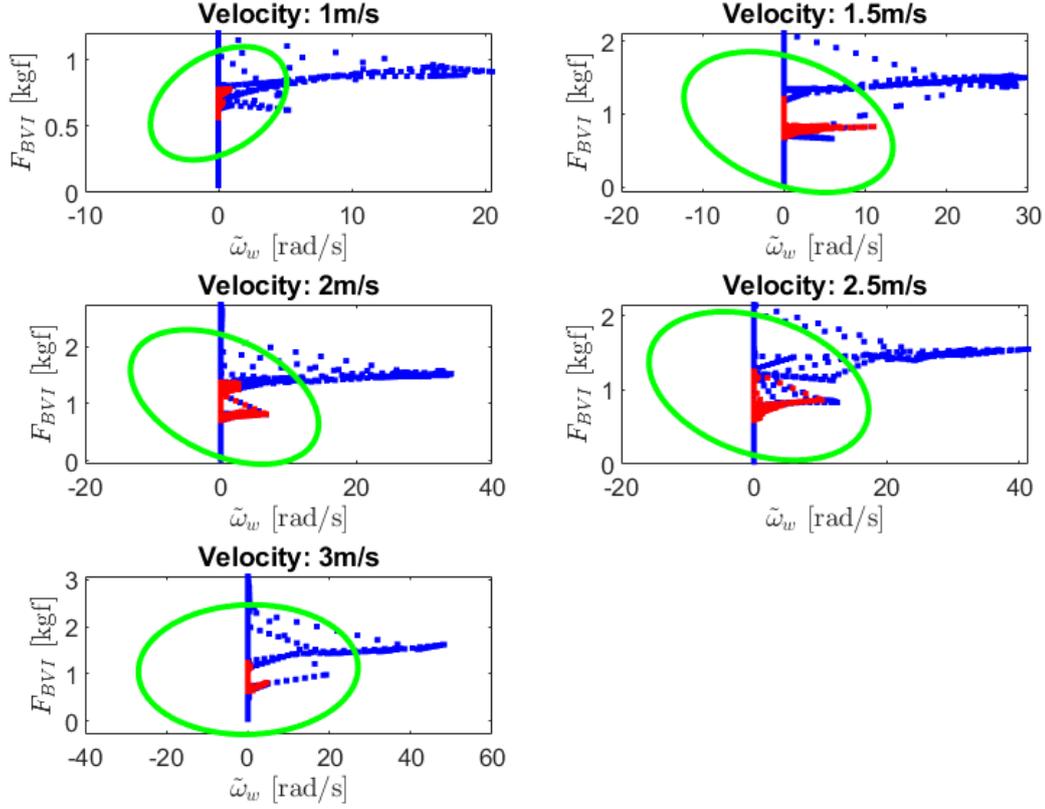


Figure 7.9: Configuration 3, Case C, the blue dots are the braking data, the red dots are the normal data and the green ellipse is the threshold of one of the possible confidence regions, for a given value of the threshold.

At this point, we are in the position to evaluate the performances obtainable by applying the braking recognition strategy described in Section 7.3.2. However, before doing it, some elucidations are due.

- For Case c of Configuration 3, i.e., when  $x(k) = [F_{BVI}, \tilde{\omega}_w(k)]^T$ , the offline computation of matrix  $S$  is performed using a slightly different training vector, i.e.,  $x_{raw}(k) = [F_{BVI}, \omega_w(k)]^T$ . In view of this, for this case, we compute

$$S = \frac{1}{N-1} \sum_{n=1}^N (x_{raw}(k) - \mu_x^{raw})(x_{raw}(k) - \mu_x^{raw})^T$$

This is done since, as clear from Figures 7.5 and 7.8, the number of occurrences where  $\tilde{\omega}_w(k) = 0$  is very large. For this reason, if we compute  $S$  using  $\tilde{\omega}_w(k)$ ,  $S$

would not capture in a correct way the variability of the angular velocity variable.

- In all cases, the data are not pre-normalized (contrarily to the prescription given in Section 7.2.1). This is done since the orders of magnitude of the variables at hand are similar, and there is no need for such a normalization for numerical consistency.
- For the computation of the variable  $T^2(k)$  in (7.3), the data are not depolarized, i.e. they have not been subtracted by their mean value. This is due to the fact that, for both  $F_{BVI}$  and  $\tilde{\omega}_w$  and in all configurations, in stopping conditions, the absolute value of  $F_{BVI}$  and  $\tilde{\omega}_w$  (and not their deviation from the mean value) reach higher peaks. This will make the brake detection more reliable and the number of false positives will be reduced. A final clarification is due: as apparent from Figure 7.8, the number of cases when  $F_{BVI}$  reaches low peaks in stopping conditions in Configuration 3 and for high velocities is large. However, such low peaks occur after long delays after the brakes, as visible from Figure 7.5. Therefore, identifying the brakes in those cases would make the brake detection ineffective.

The results of the tests are analyzed in terms of threshold exceedance count and detection delays and plotted for every combination of configurations and velocities in Figures 7.10-7.14 for varying threshold values. The left upper plots of these figures show the threshold exceedance count, while the left lower plots show the detection delay, both computed for varying values of  $T$ . The right plots depict the correlation between these two variables (threshold exceedance count and detection delay). As a general rule, note that a curve closer to the origin indicates better signal performance.

In case of Configuration 3, the analysis is repeated for each set of variables selected for the detection, see equation (7.7), i.e. for Cases A, B and C.

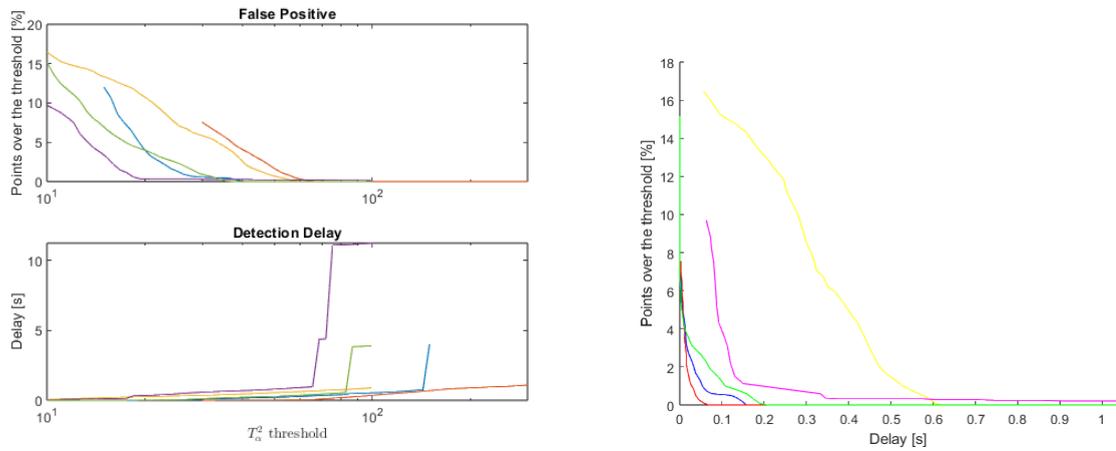


Figure 7.10: Braking detection performed with Configuration 1 at different velocities: in blue 1 m/s, in red 1.5 m/s, in yellow 2 m/s, in purple 2.5 m/s, and in green 3 m/s.

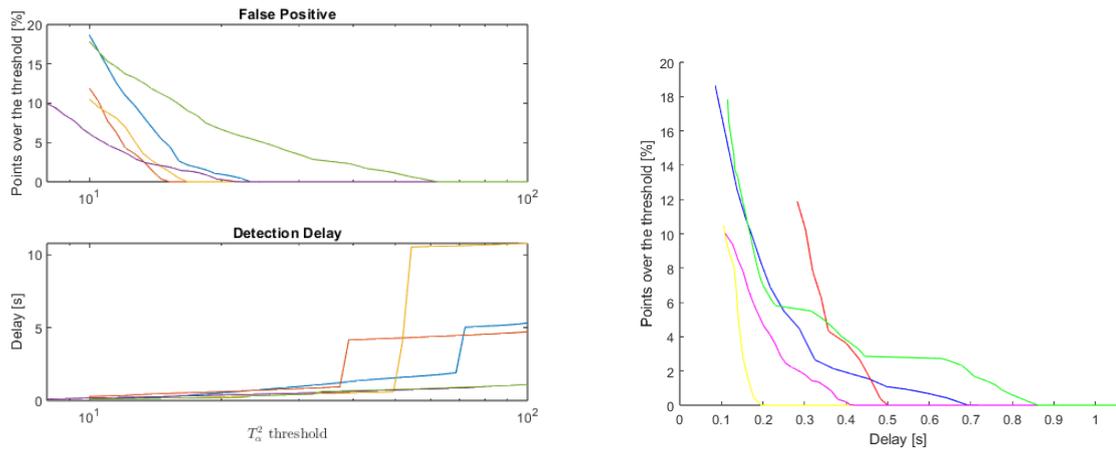


Figure 7.11: Braking detection performed with Configuration 2 at different velocities: in blue 1 m/s, in red 1.5 m/s, in yellow 2 m/s, in purple 2.5 m/s, and in green 3 m/s.

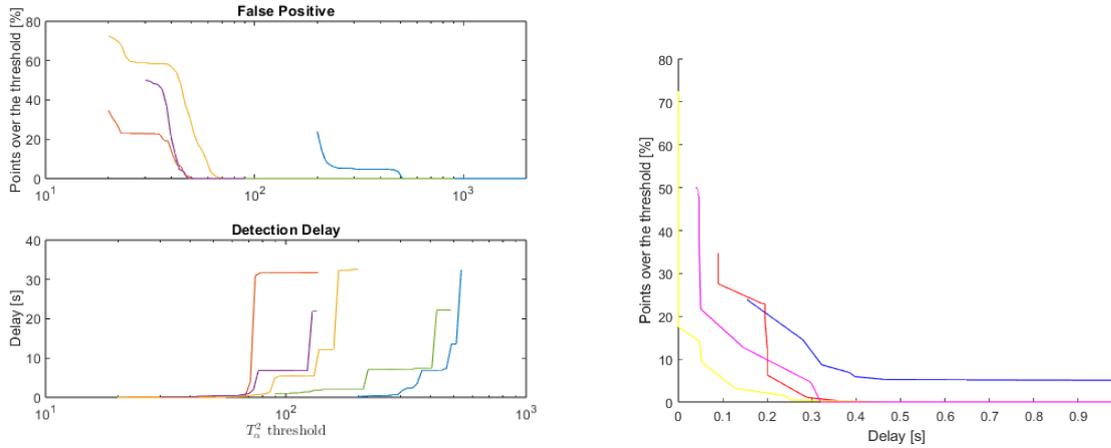


Figure 7.12: Braking detection performed with variable  $x = F_{BVI}$  (i.e. Case A) in Configuration 3 at different velocities: in blue 1 m/s, in red 1.5 m/s, in yellow 2 m/s, in purple 2.5 m/s, and in green 3 m/s.

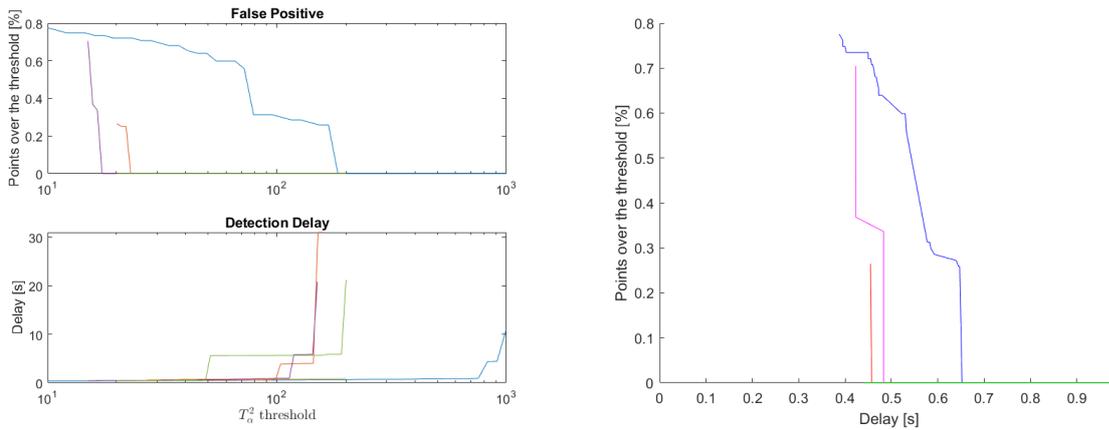


Figure 7.13: Braking detection performed with variable  $x = \tilde{\omega}_w$  (i.e. Case B) in Configuration 3 at different velocities: in blue 1 m/s, in red 1.5 m/s, in yellow 2 m/s, in purple 2.5 m/s and in green 3 m/s.

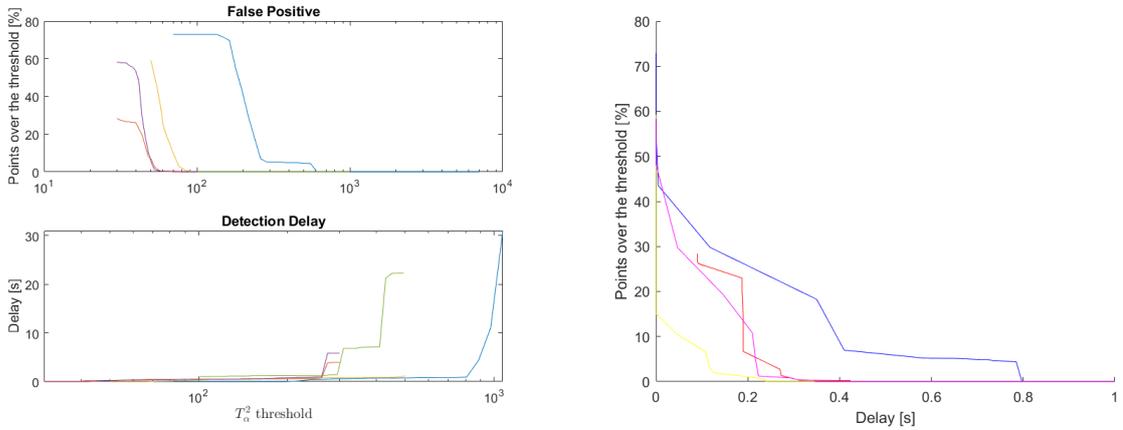


Figure 7.14: Braking detection performed with both variables  $x = [F_{BVI} \quad \tilde{\omega}_w]^T$  (i.e. Case C) in Configuration 3 at different velocities: in blue 1 m/s, in red 1.5 m/s, in yellow 2 m/s, in purple 2.5 m/s and in green 3 m/s.

Figures 7.10-7.14 illustrate that the selected analysis procedure correctly enables to recognize braking when it occurs. The choice of the most suitable option for the threshold  $T$ , for all velocities, requires a compromise between threshold exceedance count and time delay. In this work, the metric used to determine the best threshold value is the normalized Euclidean distance  $D_{norm}$  between the origin and the points from the graph, as shown below:

$$D_{norm} = \sqrt{\left(\frac{TD}{TD_{max}}\right)^2 + \left(\frac{TEC}{TEC_{max}}\right)^2} \quad (7.9)$$

where TD represents the time delay and TEC represents the number of threshold exceedance count.

The optimal values are determined as the threshold values minimizing  $D_{norm}$ . This represents the most favorable balance between time delay and the threshold exceedance count in the system.

Figures 7.12, 7.13, and 7.14 show the results for Configuration 3. It is evident that braking detection based solely on the force  $F_{BVI}$  is not the optimal approach. Although the time delay is the lowest when considering the force variable (0.2s less), the threshold exceedance count is too high to justify selecting a braking detection method based on force alone. Additionally, due to the high threshold exceedance count, it would be challenging to achieve good results when merging the optimal thresholds found at different velocities to develop a procedure that works in real application scenarios.

Conversely, using both force  $F_{BVI}$  and velocity  $\tilde{\omega}_w$  provides similar outcomes to using

$F_{BVI}$  alone. On the other hand, a braking detection based solely on  $\tilde{\omega}_w$  resulted in a very low and consistent threshold exceedance count while maintaining a low detection delay. Therefore, a braking detection method that considers only the velocity variable  $\tilde{\omega}_w$  is preferred, as it would be more robust in real-world scenarios.

## 7.4. Velocity-dependent braking detection

For the online implementation of the braking detection algorithm, different strategies can be adopted to cope with the fact that the user and robot velocities are time-varying and may not correspond to those defined in the braking recognition algorithm design.

In this work, for all configurations (in which, we recall that the finally selected variable  $x(k)$  under test is scalar), we adopt the following test:

- If  $|x(k)| \leq \bar{x}(v)$ , then we declare the system in nominal behavior at time  $k$ .
- If  $|x(k)| > \bar{x}(v)$ , then we declare the system in non-nominal behavior at time  $k$ .

The peculiarity of the adopted approach is that the threshold  $\bar{x}(v)$  is a function of the (possibly suitably filtered over time) robot speed  $v$ . The threshold  $\bar{x}(v)$  is a curve attempting to fit the optimal thresholds previously computed for different velocities. First of all, note that, since for all configurations  $x(k)$  is scalar, the covariance matrix  $S(v)$ , computed in (??) and dependent on the robot velocity  $v$  (for this reason we now write  $S(v)$ ), reduces to a scalar variance. We define  $\sigma(v_i) = S(v_i) \cdot T^2(v_i)$ , where  $T^2(v_i)$  is the optimal threshold value for each velocity  $v_i$ ,  $i = 1, \dots, 5$ , where  $v_i = 0.5 + 0.5i$  m/s.

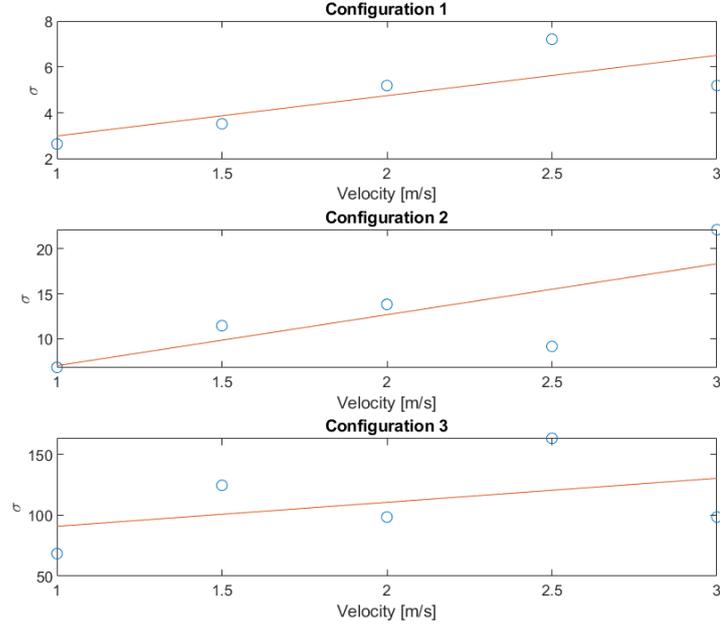


Figure 7.15: All configurations in increasing order from top to bottom. The blue dots are the optimal threshold values  $\sigma$  for each velocity  $v_i$ . The red line represents the interpolation of these points,  $\hat{\sigma}(v_i)$ .

In Figure 7.15 we analyze  $\sigma(v)$  and we derive an interpolating curve for  $\sigma(v)$ , named  $\hat{\sigma}(v)$ . The function  $\hat{\sigma}(v)$  is computed as

$$\hat{\sigma}(v) = \bar{\sigma} + \alpha_1 v \quad (7.10)$$

where  $\bar{\sigma}$  and  $\alpha_1$  are obtained by minimizing the least square cost function

$$\sum_{i=1}^5 (\sigma(v_i) - \bar{\sigma} - \alpha_1 v_i)^2 \quad (7.11)$$

Eventually, the time-varying threshold  $\bar{x}(v)$  is computed as  $\bar{x}(v) = \sqrt{\hat{\sigma}(v)}$ . The outcome of the braking detection done with these new thresholds in comparison with the optimal value found at the end of Section 7.3.3 in terms of threshold exceedance count and detection delay are shown in Figures 7.16, 7.17 and 7.18.

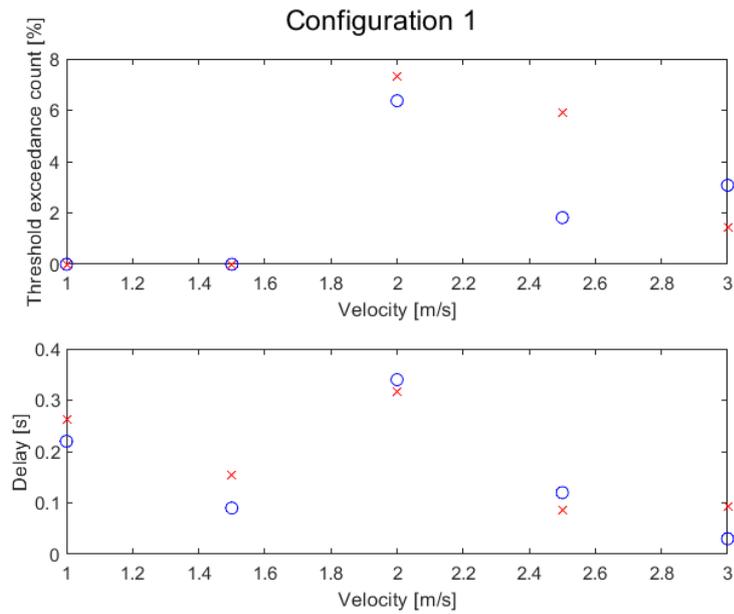


Figure 7.16: Configuration 1, the blue dots are the optimal points, the red crosses are points found after the interpolation. The threshold exceedance count is on the upper plot, while the detection delay is on the lower one.

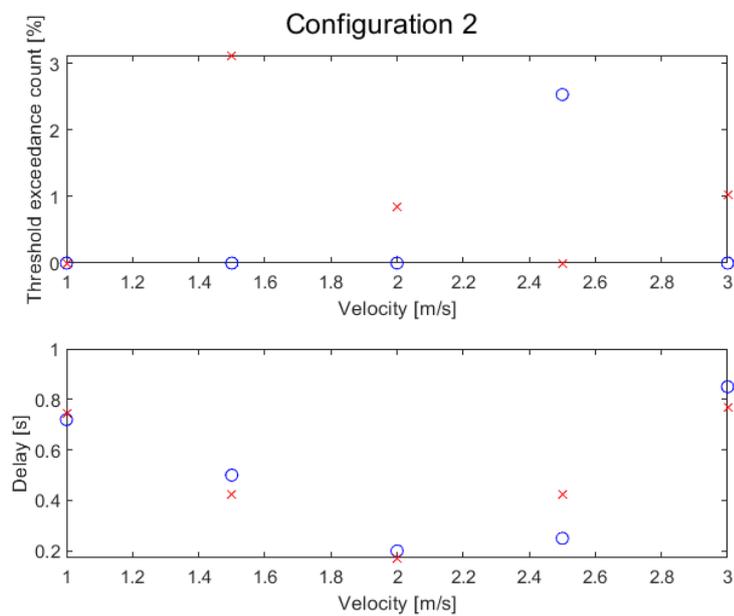


Figure 7.17: Configuration 2, the blue dots are the optimal points, the red crosses are points found after the interpolation. The threshold exceedance count is on the upper plot, while the detection delay is on the lower one.

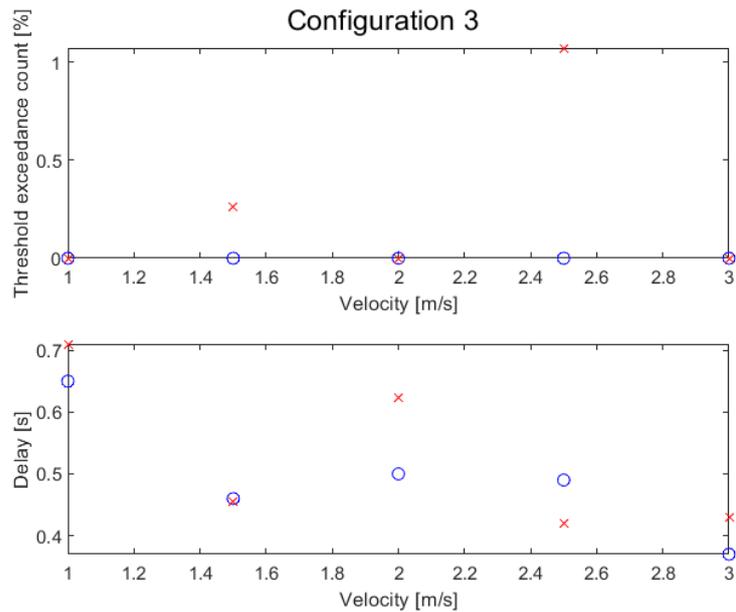


Figure 7.18: Configuration 3, the blue dots are the optimal points, the red crosses are points found after the interpolation. The threshold exceedance count is on the upper plot, while the detection delay is on the lower one.

The implementation of a velocity-dependent threshold resulted in braking detection with similar time delays but generally higher threshold exceedance counts. Configurations 1 and 2 generally exhibit higher threshold exceedance counts compared to Configuration 3 due to the behavior previously highlighted in Section 7.3.3. Nonetheless, these results are acceptable although they need to be validated through field testing.



## 8 | Conclusions and future work

The research presented in this thesis addresses the significant challenges faced by visually impaired individuals, particularly in navigating unfamiliar environments, avoiding obstacles, and participating in physical activities. These challenges essentially reduce their quality of life. The primary goal of this project is to leverage technology to create a more inclusive environment for visually impaired individuals, thus enhancing their independence and engagement in daily activities.

This work was conducted within the framework of the BUDD-e project (Blind-assistive Autonomous Droid Device), funded by Politecnico di Milano through the POLISOCIAL 2021 program and the PRIN 2022 project CARE (Control of Assistive Robots in crowded Environments). The project was driven by the need to address the everyday challenges faced by visually impaired individuals in accessing various services and public spaces, including sports arenas, healthcare facilities, shopping areas, and cultural centers. The project aims to develop a novel robotic guide aimed at providing a comprehensive solution that surpasses the limitations of existing technologies.

While the initial BUDD-e prototype was promising, it also highlighted challenges related to reactivity and adaptability in diverse scenarios. This necessitated further improvements to better support visually impaired individuals.

A significant focus of this work was on optimizing the smart tether system configuration to enhance practicality and user comfort. Previous designs exhibited undesirable oscillations and discomfort, particularly in sports environments, due to the placement of the load cell on the cable. Adjustments made to the smart tether system configuration successfully eliminated these issues, resulting in a practical and comfortable user experience across various scenarios.

To achieve a more responsive and adaptable robotic guide, a new controller was devised to ensure precise control of the robot's position relative to the user.

Moreover, the new motor, which provides higher speeds and ensures better performances, was employed to overcome the saturation limitation of the velocity of the winch while running.

Although the new configurations devised with the old setup, i.e., Configurations 1 and 2, provide interesting results as compared with the configuration proposed previously in [12], the experimental results shown in Chapter 5 witness that the new configuration (i.e. Configuration 3), where the new motor is used, has much better dynamic performances with respect to the previous ones. A critical issue addressed in this thesis was also related to the robot's response to changes in user speed, such as sudden stops while running. Indeed, during brakes, the existing system, Yape, executed a smooth stop that halted the robot at a greater distance from the user, potentially causing damage to the smart tether system. To resolve this, a novel strategy was proposed to the distance control system, allowing for prompt detection of user braking and incorporating a feedforward brake controller to take over when necessary.

These enhancements allow to improve significantly the system's reactivity, braking control, and load cell configuration, expanding the potential applications of the robotic guide. The refined system now is much more efficient in supporting visually impaired individuals in participating in sports activities where rapid changes in movement are common, thus addressing a broader range of mobility concerns.

Future research and development should continue to build upon these advancements, further refining and expanding the capabilities of assistive robotic guides.

Future work can concentrate on several key areas to further enhance the capabilities and applications of the assistive robotic guide for visually impaired individuals.

One potential improvement could be upgrading the speaker to function as an interactive tool, enhancing communication between the user and the robot. This could involve adding a microphone, enabling the user to communicate more effectively with BUDD-e. Additionally, vocal communication could be improved, with BUDD-e providing clearer information about key details like the remaining distance or estimated time to the destination. It could also offer better descriptions of potentially hazardous situations. To further enhance user safety, a button should be implemented to allow users to call for assistance from staff if problems arise.

Another crucial development area is enabling users to modify their route by choosing among predefined options. This could be facilitated through various means, such as a keyboard interface or a smartphone application. The smartphone application could also address the booking issue for BUDD-e, allowing visually impaired individuals to book the service autonomously whenever needed.

Extensive field testing of the braking detection and control system in real-world conditions,

including various environments and specifically in sports facilities, is crucial and must be conducted to ensure the system's robustness and reliability. Furthermore, the development and implementation of running programs tailored to the needs of visually impaired users would be advantageous. Such programs should prioritize safety while encouraging physical activity, thereby ensuring that the robotic guide can support users in a variety of athletic pursuits.



## Bibliography

- [1] *Technical Manual N5 - Fieldbus: CANopen - For use with the following devices: N5-1-2, N5-2-2*, volume Technical Manual Version 2.0.3.
- [2] D. Ahmetovic, C. Bernareggi, A. Gerino, and S. Mascetti. Zebrarecognizer: Efficient and precise localization of pedestrian crossings. *2014 22nd International Conference on Pattern Recognition*, pages 2566–2571, 2014. doi: 10.1109/ICPR.2014.443.
- [3] J. Borenstein and I. Ulrich. The guidecane - a computerized travel aid for the active guidance of blind pedestrians. *Proceedings of International Conference on Robotics and Automation*, 2:1283–1288, 1997.
- [4] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume Volume 15 of Studies in Applied Mathematics Society for Industrial and Applied Mathematics (SIAM). 1994.
- [5] M. de Oliveira, J. Geromel, and J. Bernussou. An LMI optimization approach to multiobjective controller design for discrete-time systems. In *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No.99CH36304)*, volume 4, pages 3611–3616, 1999.
- [6] Z. Fei, E. Yang, H. Hu, and H. Zhou. Review of machine vision-based electronic travel aids. In *23rd International Conference on Automation and Computing (ICAC 2017)*, pages 1–7. IEEE, 2017. doi: 10.23919/ICoNAC.2017.8082021.
- [7] J. Guerreiro, D. Sato, S. Asakawa, H. Dong, K. M. Kitani, and C. Asakawa. Cabot: Designing and evaluating an autonomous navigation robot for blind people. *Proceedings of ASSETS '19 Conference*, pages 68–82, 2019.
- [8] M. Hersh and M. Johnson. A robotic guide for blind people part 2: Gender and national analysis of a multi-national survey and the application of the survey results and the cat model to framing robot design specifications. *Applied Bionics and Biomechanics*, 9:29–43, 2012. doi: 10.3233/ABB-2011-0034.
- [9] I. Kim. Recent advancements in indoor electronic travel aids for the blind or visually

- impaired: a comprehensive review of technologies and implementations. *Universal Access in the Information Society*, 2024. doi: 10.1007/s10209-023-01086-8.
- [10] A. Kulkarni, A. Wang, L. Urbina, A. Steinfeld, and B. Dias. Robotic assistance in indoor navigation for people who are blind. *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 461–462, 2016.
- [11] P. Marson. Development, control and validation of a guide robot for blind and partially sighted users. Master’s thesis, Politecnico di Milano, 2021-2022.
- [12] P. Rattamasanaprapai. Design, realization, and control of a smart tether system for a robotic guide for blind and visually impaired users. Master’s thesis, Politecnico di Milano, 2021-2022.
- [13] R. Scattolini. *Safety in Automation Systems*. 2022-2023.
- [14] E. Tagarelli. Development of a distance controller for a robotic guide for blind and visually impaired users and customization for wayfinding in healthcare facilities. Master’s thesis, Politecnico di Milano, 2022-2023.
- [15] World Health Organization. Blindness and vision impairment, 2021. URL <https://www.who.int/standards/classifications/classification-of-diseases>.
- [16] A. Xiao, W. Tong, L. Yang, J. Zeng, Z. Li, and K. Sreenath. Robotic guide dog: Leading a human with leash-guided hybrid physical interaction. *CoRR*, abs/2103.14300, 2021.
- [17] C. Ye, S. Hong, X. Qian, and W. Wu. Co-robotic cane: A new robotic navigation aid for the visually impaired. *IEEE Systems, Man, and Cybernetics Magazine*, 2(2): 33–42, 2016. doi: 10.1109/MSMC.2015.2501167.
- [18] L. Zeng. A survey: Outdoor mobility experiences by the visually impaired. *Mensch Computer – Workshopband*, 2015. doi: 10.1515/9783110443905-056.

## List of Figures

1.1	Shorter caption . . . . .	3
1.2	Co-robotic cane . . . . .	3
1.3	Example of ZebraRecognizer applied to an image . . . . .	4
1.4	CaBot . . . . .	5
1.5	The Pioneer 3DX robot equipped with a wooden harness . . . . .	5
1.6	Robotic Guide Dog . . . . .	6
1.7	Budd-e components scheme . . . . .	7
1.8	Smart tether from left to right: handle, elastic rope, load cell, non-elastic rope . . . . .	8
1.9	Shorter caption . . . . .	10
2.1	The Smart Tether System: 1. Box with electronic components; 2. Motor; 3. Shaft; 4. Polyurethane tube; 5. Proximity sensor. . . . .	18
2.2	The Smart Tether System: Smart tether system cable: 6. Rigid cable; 7. Load cell; 8. Elastic cable; 9. Handle. . . . .	18
2.3	Motor, gearbox and encoder . . . . .	19
2.5	HX711 amplifier . . . . .	20
2.6	Shorter caption . . . . .	20
2.7	DC-DC converter . . . . .	21
2.8	The smart tether system: 1.Box with electronic components; 2.Motor; 3.Winch; 4.Load cell; 5.Proximity sensor . . . . .	21
2.9	Shaft . . . . .	22
2.11	Shorter caption . . . . .	24
2.13	Winch, support structure and load cell . . . . .	25
3.1	System block diagram. The input $v_{BVI}$ to the tether model is the user's longitudinal speed; the $F_{BVI}$ is the force exerted on the user; the $\omega_\omega$ is the actual velocity of the motor and $\omega_\omega^0$ is its reference velocity. . . . .	27
3.2	Experimental setup . . . . .	28
3.3	Identification data . . . . .	29

3.4	Identification of the deadzone . . . . .	30
3.5	Transfer function block diagram between $\omega_w^{0,DZ}$ and $\omega_w$ . . . . .	30
3.6	Plot of both the dynamic data (red dots) and the steady state data (black circles) for $\theta_w - F_{BVI}$ . The estimated model with the polynomial regression method of order 1 is in yellow. . . . .	32
3.7	Plot of both the dynamic data (red dots) and the steady state data (black circles) for $\theta_w - F_{BVI}$ . The estimated model with the polynomial regression method of order 2 is in blue, order 3 in purple, and order 4 in green. . . . .	33
3.8	Comparison between the real data and the trajectory generated by the tether model. Black line: $F_{BVI}$ ; red line: data generated by the identified polynomial regression models; From the top to the bottom: polynomial regression model of first order; polynomial regression model of second order; polynomial regression model of third order; polynomial regression model of fourth order. . . . .	34
3.9	Block schemes of Hammerstein model and Wiener model . . . . .	35
3.10	Comparison between the real data and the trajectory generated by the Hammerstein model of the tether with the fourth-order nonlinearity. . . . .	36
3.11	Comparison between the real data and the trajectory generated by the Wiener model of the tether with the fourth-order nonlinearity. . . . .	37
3.12	Validation data . . . . .	38
3.13	Validation of the deadzone model . . . . .	38
3.14	Comparison in validation between the real data and the trajectory generated by the Hammerstein model of the tether with the fourth-order nonlinearity. . . . .	39
3.15	Comparison in validation between the real data and the trajectory generated by the Wiener model of the tether with the fourth-order nonlinearity. . . . .	40
3.16	Smart Tether system; $F(s)$ is the transfer function identified for the winch, while $G_{\theta_w}$ is the transfer function identified for the tether. . . . .	41
3.17	Bode plot of $G(s)$ . . . . .	42
3.18	Stable and robust ideal control system . . . . .	43
3.19	Control system; $\delta$ (equation (3.14)) is the gain corresponding to the inverse of the derivative of the nonlinearity of fourth-order ( $K_T^{SS}$ ) evaluated in zero. . . . .	44
3.20	Simulation of the frequency control with disturbance, which is the offset $v - v_{BVI}$ , already reported in angular velocity . . . . .	45
3.21	Smart Tether system; $F(s)$ is the transfer function identified for the winch. $\Delta_{\theta_w}(s)$ is the multiplicative uncertainty, given that $G_{\theta_w} = 1 + \Delta_{\theta_w}(s)$ . . . . .	46

3.22	Control system; $\delta$ (equation (3.14)) is the gain corresponding to the inverse of the derivative of the nonlinearity of fourth-order evaluated in zero . . . .	52
3.23	Deadzone $dz(\cdot)$ introduced on $\Delta\hat{v}_k$ . For $\Delta\hat{v}_k$ within the interval $[-0.015, 0.025]$ , $\Delta\hat{v}_k^{dz} = 0$ . . . . .	53
4.1	The Smart Tether System: 1. N5-2-2 controller; 2. Motor; 3. Load cell; 4. Winch; 5. Capacitor; 6. Arduino board. . . . .	56
4.2	Motor . . . . .	57
4.3	The Z-K10000/100 capacitor (95 mm in length, $\varnothing$ 40mm) . . . . .	57
4.4	Shaft designed to fit the gear-shaped crankshaft . . . . .	57
4.5	U-groove plastic bearing (diameter 30mm) . . . . .	58
4.6	Load cell, bearing and winch . . . . .	58
4.10	Load cell, bearing and winch . . . . .	61
4.11	HX711 amplifier . . . . .	61
4.13	Complete new setup . . . . .	63
4.14	. . . . .	65
4.15	Safety functions implemented on speed . . . . .	66
4.16	Safety functions implemented on torque . . . . .	67
4.17	Motor control system behavior in response to position values. . . . .	68
5.1	Configurations 1 and 2; first motor and Yape standing still . . . . .	69
5.2	Configuration 3; new motor fixed on the table . . . . .	70
5.3	Right hand musculoskeletal landmarks . . . . .	71
5.4	Distance between the user's hand and the smart tether system. The three configurations are shown and ordered increasingly from left to right. The four testers (T1-T4) are displayed in increasing order from the top to the bottom. The data referring to $v_{BVI} = 1.5$ km/h are shown in blue, $v_{BVI} = 4$ km/h in orange, $v_{BVI} = 6.5$ km/h in yellow, $v_{BVI} = 9$ km/h in purple and $v_{BVI} = 12$ km/h in green. . . . .	72
5.5	Winch velocity and Force for Tester T1. The three configurations are shown and ordered increasingly from left to right. The data referring to $v_{BVI} = 1.5$ km/h are shown in blue, $v_{BVI} = 4$ km/h in orange, $v_{BVI} = 6.5$ km/h in yellow, $v_{BVI} = 9$ km/h in purple and $v_{BVI} = 12$ km/h in green. In grey, we present the measurements of the actual torque provided by the Nanotec controller, already reported in kgf. . . . .	73

- 5.6 Winch velocity and Force for Tester T2. The three configurations are shown and ordered increasingly from left to right. The data referring to  $v_{BVI} = 1.5\text{km/h}$  are shown in blue,  $v_{BVI} = 4\text{km/h}$  in orange,  $v_{BVI} = 6.5\text{km/h}$  in yellow,  $v_{BVI} = 9\text{km/h}$  in purple and  $v_{BVI} = 12\text{km/h}$  in green. In grey, we present the measurements of the actual torque provided by the Nanotec controller, already reported in kgf. . . . . 74
- 5.7 Winch velocity and Force for Tester T3. The three configurations are shown and ordered increasingly from left to right. The data referring to  $v_{BVI} = 1.5\text{km/h}$  are shown in blue,  $v_{BVI} = 4\text{km/h}$  in orange,  $v_{BVI} = 6.5\text{km/h}$  in yellow,  $v_{BVI} = 9\text{km/h}$  in purple and  $v_{BVI} = 12\text{km/h}$  in green. In grey, we present the measurements of the actual torque provided by the Nanotec controller, already reported in kgf. . . . . 75
- 5.8 Winch velocity and Force for Tester T4. The three configurations are shown and ordered increasingly from left to right. The data referring to  $v_{BVI} = 1.5\text{km/h}$  are shown in blue,  $v_{BVI} = 4\text{km/h}$  in orange,  $v_{BVI} = 6.5\text{km/h}$  in yellow,  $v_{BVI} = 9\text{km/h}$  in purple and  $v_{BVI} = 12\text{km/h}$  in green. In grey, we present the measurements of the actual torque provided by the Nanotec controller, already reported in kgf. . . . . 76
- 5.9 Fourier transforms of the hand-table or hand-Yape distance (on the left) and of the force  $F_{BVI}$  (on the right) for different treadmill velocities (from the top to the bottom):  $v_{BVI} = 1.5\text{km/h}$ ,  $v_{BVI} = 4\text{km/h}$ ,  $v_{BVI} = 6.5\text{km/h}$ ,  $v_{BVI} = 9\text{km/h}$  and  $v_{BVI} = 12\text{km/h}$ . For each panel, we can find in blue the Fourier transform of 1<sup>st</sup> Configuration, in red the 2<sup>nd</sup> Configuration, and in yellow the 3<sup>rd</sup> Configuration. . . . . 78
- 5.10 Fourier transforms of the hand-table or hand-Yape distance (on the left) and of the force  $F_{BVI}$  (on the right) for different treadmill velocities (from the top to the bottom):  $v_{BVI} = 1.5\text{km/h}$ ,  $v_{BVI} = 4\text{km/h}$ ,  $v_{BVI} = 6.5\text{km/h}$ ,  $v_{BVI} = 9\text{km/h}$  and  $v_{BVI} = 12\text{km/h}$ . For each panel, we can find in blue the Fourier transform of 1<sup>st</sup> Configuration, in red the 2<sup>nd</sup> Configuration, and in yellow the 3<sup>rd</sup> Configuration. . . . . 79
- 5.11 Fourier transforms of the hand-table or hand-Yape distance (on the left) and of the force  $F_{BVI}$  (on the right) for different treadmill velocities (from the top to the bottom):  $v_{BVI} = 1.5\text{km/h}$ ,  $v_{BVI} = 4\text{km/h}$ ,  $v_{BVI} = 6.5\text{km/h}$ ,  $v_{BVI} = 9\text{km/h}$  and  $v_{BVI} = 12\text{km/h}$ . For each panel, we can find in blue the Fourier transform of 1<sup>st</sup> Configuration, in red the 2<sup>nd</sup> Configuration, and in yellow the 3<sup>rd</sup> Configuration. . . . . 80

5.12	Fourier transforms of the hand-table or hand-Yape distance (on the left) and of the force $F_{BVI}$ (on the right) for different treadmill velocities (from the top to the bottom): $v_{BVI} = 1.5\text{km/h}$ , $v_{BVI} = 4\text{km/h}$ , $v_{BVI} = 6.5\text{km/h}$ , $v_{BVI} = 9\text{km/h}$ and $v_{BVI} = 12\text{km/h}$ . For each panel, we can find in blue the Fourier transform of 1 <sup>st</sup> Configuration, in red the 2 <sup>nd</sup> Configuration, and in yellow the 3 <sup>rd</sup> Configuration. . . . .	81
6.1	LiDAR spherical coordinates . . . . .	86
6.2	Robosense R16 LiDAR . . . . .	87
6.3	Point cloud preprocessing: a) original image, b) background removal, c) grid quantization, d) grid binarization . . . . .	89
6.4	LiDAR detecting different objects (the surrounding wall, a tree, a person) .	90
6.5	Yape switching model . . . . .	92
6.6	Example of the data used to identify the longitudinal dynamical model of Yape. Black dashed line: $v_{ref}$ ; gray solid line: measured Yape velocity $v$ ; blue solid line: simulated Yape velocity $v$ . . . . .	92
6.7	Response (i.e., measured user-robot distance $d$ ) to step-wise inputs of the distance control systems designed: (6.22) (blue line) and (6.24) (red line). Dashed lines: reference distances $d_o$ . . . . .	97
6.8	Response to step-wise inputs of the distance control systems 6.22 (Controller 1), and 6.24 (Controller 2). For all panels we show: reference Yape velocity $v_{ref}$ (black dashed line); Yape velocity $v$ (red line); user estimated velocity $v_{BVI}$ (blue line). . . . .	97
6.9	Response to varying user speed of the distance controllers (6.22) (Controller 1, left panel) and (6.24) (Controller 2, right panel). Top panels: reference distance $d_o$ (black dashed line); measured distance $d$ (blue solid line). Bottom panels: Yape velocity $v$ (red line); user estimated velocity $v_{BVI}$ (blue line). . . . .	98
7.1	Ellipsoidal confidence region . . . . .	103
7.2	Test performed in Milan, on the athletics track at Giuriati Sport Center to gather data for the braking detection procedure . . . . .	105
7.3	Test performed with Configuration 1 and velocity $v_{BVI} = 1.5 \text{ m/s}$ . . . . .	106
7.4	Test performed with Configuration 2 and velocity $v_{BVI} = 1.5 \text{ m/s}$ . . . . .	106
7.5	Test performed with Configuration 3 and velocity $v_{BVI} = 1.5 \text{ m/s}$ . . . . .	107
7.6	Configuration 1: distribution of the force variable in the "normal dataset" in red and in the "braking dataset" in blue. . . . .	109

7.7	Configuration 2: distribution of the force variable in the "normal dataset" in red and in the "braking dataset" in blue. . . . .	110
7.8	Configuration 3: distribution of both the force and the winch angular velocity variables in the "normal dataset" in red and in the "braking dataset" in blue. . . . .	110
7.9	Configuration 3, Case C, the blue dots are the braking data, the red dots are the normal data and the green ellipse is the threshold of one of the possible confidence regions, for a given value of the threshold. . . . .	111
7.15	All configurations in increasing order from top to bottom. The blue dots are the optimal threshold values $\sigma$ for each velocity $v_i$ . The red line represents the interpolation of these points, $\hat{\sigma}(v_i)$ . . . . .	117
7.16	Configuration 1, the blue dots are the optimal points, the red crosses are points found after the interpolation. The threshold exceedance count is on the upper plot, while the detection delay is on the lower one. . . . .	118
7.17	Configuration 2, the blue dots are the optimal points, the red crosses are points found after the interpolation. The threshold exceedance count is on the upper plot, while the detection delay is on the lower one. . . . .	118
7.18	Configuration 3, the blue dots are the optimal points, the red crosses are points found after the interpolation. The threshold exceedance count is on the upper plot, while the detection delay is on the lower one. . . . .	119

# List of Tables

1.1 Possible improvements . . . . . 11



## Acknowledgements

Ringraziamo il Prof. Marcello Farina, nostro relatore in questo percorso. Il suo grande supporto, la sua disponibilità e la sua infinita umanità sono stati fondamentali per partire e, soprattutto, ripartire.

Un ringraziamento va a Paolo Falcone per il suo ruolo all'interno del progetto CARE.

Ringraziamo Alessandro Mattioli, con il suo aiuto siamo riusciti ad integrare all'interno di BUDD-e un motore in grado di fornirci tutto quello di cui avevamo bisogno.

Grazie a Filippo Bertozzi e allo Human Performance Lab, che ci hanno permesso di svolgere le accurate analisi di confronto presenti in questa tesi.

Thank you Jinyang Li for your time and your precious help.

Grazie a Luca Mozzarelli e Luca Cattaneo, sempre disposti ad aiutarci e schiarirci le idee quando ne abbiamo avuto bisogno.

Un ringraziamento va al Prof. Alessandro Colombo, che ci ha messo a disposizione il suo tempo e la sua conoscenza.

Ringraziamo Vasco Previtali per l'interesse che ha mostrato verso il progetto. Il suo aiuto all'interno del laboratorio è stato fondamentale.

