University of Trento

Department of Information Engineering and Computer Science

Doctoral School in

Information and Communication Technology

Authority-Sharing Control of Assistive Robotic Walkers

PhD Candidate

Marco Andreetto

Advisor

Prof. Luigi Palopoli

Co-Advisor

Prof. Daniele Fontanelli

To my family

Abstract

A recognized consequence of population aging is a reduced level of mobility, which undermines the life quality of several senior citizens. A promising solution is represented by assisitive robotic walkers, combining the benefits of standard walkers (improved stability and physical support) with sensing and computing ability to guarantee cognitive support.

In this context, classical robot control strategies designed for fully autonomous systems (such as fully autonomous vehicles, where the user is excluded from the loop) are clearly not suitable, since the user's residual abilities must be exploited and practiced. Conversely, to guarantee safety even in the presence of user's cognitive deficits, the responsibility of controlling the vehicle motion cannot be entirely left to the assisted person. The authority-sharing paradigm, where the control authority, i.e., the capability of controlling the vehicle motion, is shared between the human user and the control system, is a promising solution to this problem.

This research develops control strategies for assistive robotic walkers based on authority-sharing: this way, we ensure that the walker provides the user only the help he/she needs for safe navigation. For instance, if the user requires just physical support to reach the restrooms, the robot acts as a standard rollator; however, if the user's cognitive abilities are limited (e.g., the user does not remember where the restrooms are, or he/she does not recognize obstacles on the path), the robot also drives the user towards the proper corridors, by planning and following a safe path to the restrooms.

The authority is allocated on the basis of an error metric, quantifying the distance between the current vehicle heading and the desired movement direction to perform the task. If the user is safely performing the task, he/she is endowed with control authority, so that his/her residual abilities are exploited. Conversely, if the user is not capable of safely solving the task (for instance, he/is going to collide with an obstacle), the robot intervenes by partially or totally taking the control authority to help the user and ensure his/her safety (for instance, avoiding the collision).

We provide detailed control design and theoretical and simulative analyses of the proposed strategies. Moreover, extensive experimental validation shows that authority-sharing is a successful approach to guide a senior citizen, providing both comfort and safety. The most promising solutions include the use of haptic systems to suggest the user a proper behavior, and the modification of the perceived physical interaction of the user with the robot to gradually share the control authority using a variable stiffness vehicle handling.

Acknowledgments

Foremost, I would like to express my deepest gratitude to my advisors, Prof. Luigi Palopoli and Prof. Daniele Fontanelli, who guided me during my PhD.

Prof. Luigi Palopoli was of great inspiration for my research with his experience and foresight. The motivation he wisely offered, was fundamental in the finalization of my research ideas. His help was also precious to find an excellent position for my research period abroad.

Prof. Daniele Fontanelli was a true guide for all my PhD, with his personal and scientific advice. He carefully supervised my research, making a substantial contribution. He also partnered me in two international conferences which have been very important for my personal and professional development.

I would like to add a further thank for Prof. Luigi Palopoli for successfully coordinating the European project ACANTO, which was the sponsor of my PhD.

My sincere thanks also go to Prof. Francesco Bullo, who guided me during my research period abroad at University of California Santa Barbara with great patience and knowledge.

I would like to thank my colleagues of the Embedded Electronics and Computing Systems Group, which I synergically worked with. Stefano Divan and Fabiano Zenatti designed an innovative robotic hardware fundamental for the experimental validation of my research and they supported me in the control synthesis; Paolo Bevilacqua and Marco Frego developed an advance planning module; Valerio Magnago was our localization expert; Cristina Guerrero made a huge contribution in the software development of the robot; Bernardo Villalba Frias was very precious with his tremendous knowledge of computer science; Roberta Guidolin was fundamental and efficient in organizing almost every activity, ranging from international travels to experimental campaigns with seniors for project ACANTO. I would have never completed my research without their help.

I would like to thank all the friends of the Salesian College Piergiorgio Frassati, with whom I shared my permanence in Trento, and all the friends of the Juta Center gym, in the company of which I worked out during my PhD.

Last but not the least, I would like to thank my family for supporting me throughout my all life.

Contents

Chapter 1

Introduction

The scientific and technological development of modern world has increased the life expectancy, producing a constant aging of the population. Figure 1.1 shows the distribution of ages in 2000 and 2015, and the predictions for 2030 and 2050. Population increasing from 2000 to 2050 mainly affects the older groups (aged 25 or over): population aged 0-9 increases by 17%, population aged 10-24 increases by 18%, population aged 25-59 increases by 62%, and population aged 60 or over increases by 250%. The most relevant variation affects the senior group (aged 60 or over) and is a phenomenon typical of advanced countries. Figure 1.2 shows the percentage of population aged 60 or over among the years in different world regions. Notice that less advanced regions will slowly increase the percentage: Africa is supposed to arrive at 10% in 2050, while Europe and North America had a larger percentage (around 15%) in 1980. Europe is definitely the area with the oldest population. Predictions for 2050 state that almost 35% of European population, i.e., more than one individual every three, will be 60 years old or over.

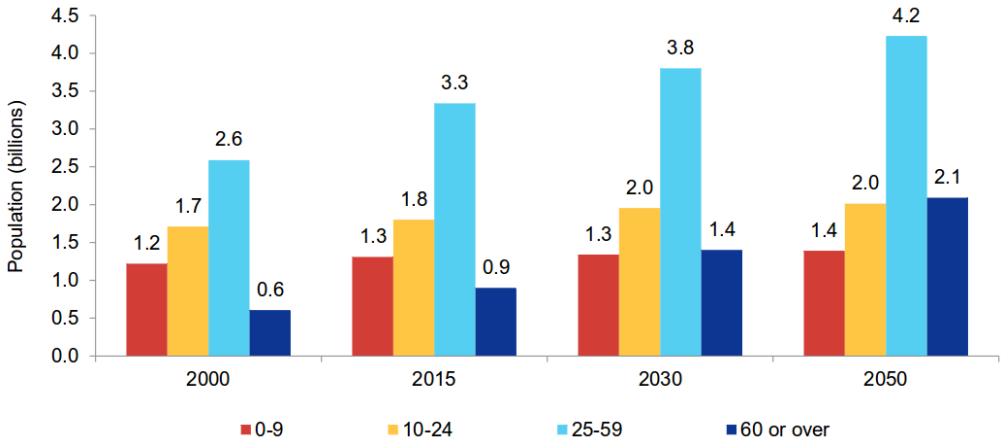

Figure 1.1: Distribution of the world population age among the years [86].

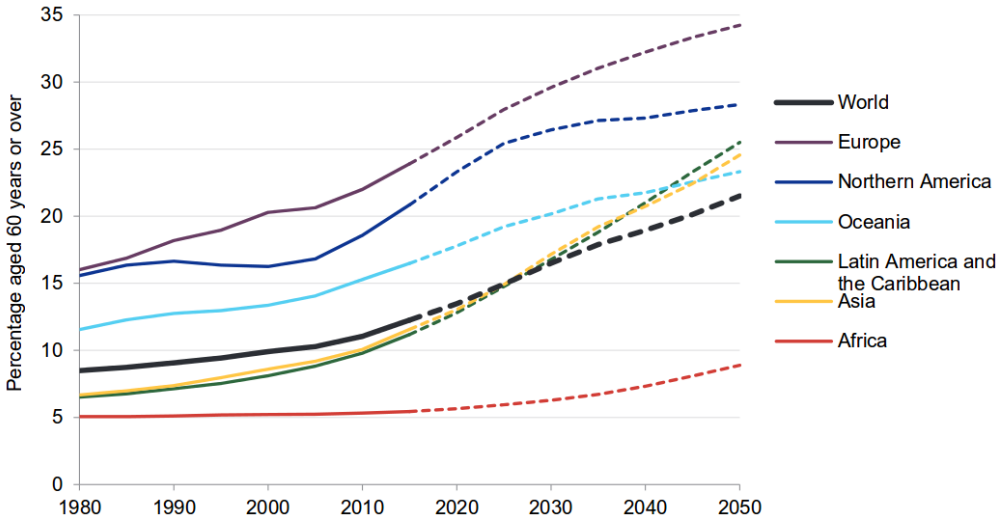

Figure 1.2: Percentage of over 60 years old among the years [86].

Population aging influences the rise of deambulation impediments, increasing the need of mobility devices. For instance, 24% of adults aged 65 or over used at least one mobility aid in 2011 in United States [36]. Figure 1.4 reports the percentage of population using assistive devices with respect to age and the type of aid. Notice that the diffusion of devices grows very rapidly (in a nonlinear way) with respect to the age. In particular almost one person every two aged 85 or over need a mobility device. Since the population in this age range is increasing, especially in advanced countries (see Figure 1.3), the importance of mobility aids is expected to grow in the next years.

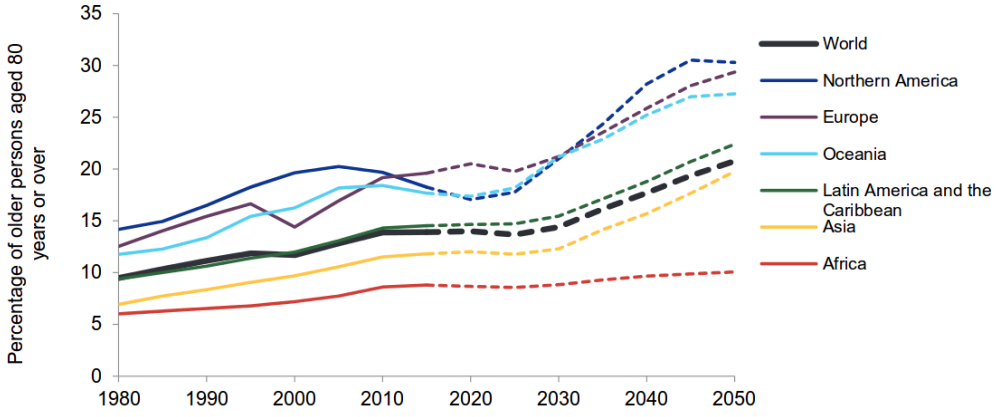

Figure 1.3: Percentage of over 80 years old among the older population (aged 60 years or over) [86].

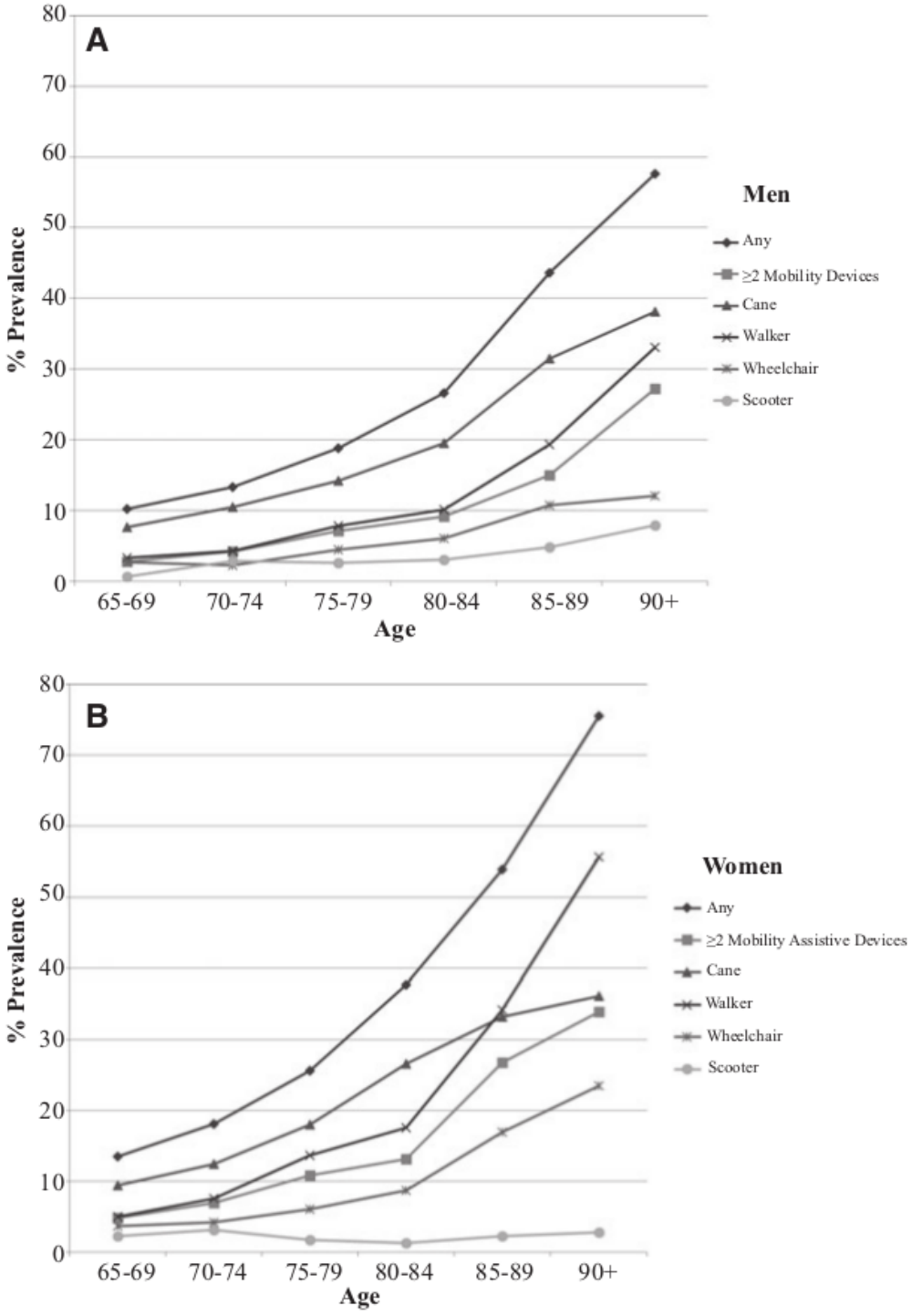

Figure 1.4: Prevalence of mobility device use with respect to age for men (A) and women (B) over 65 years old in Unated States in 2011 [36].

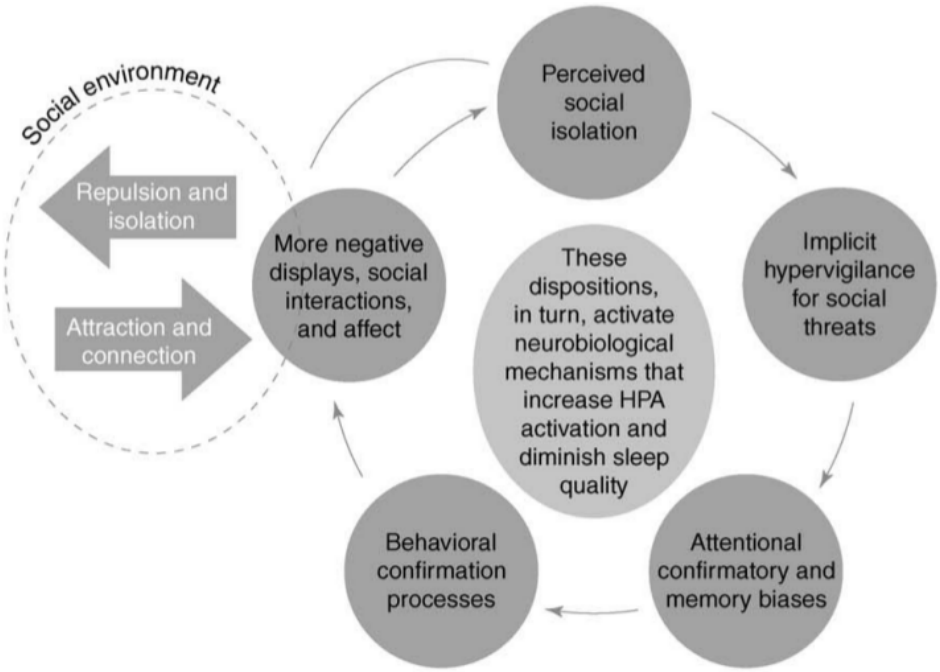

Figure 1.5: Effects of loneliness on human cognition [23].

The reduction of mobility highly affects the life of seniors, worsening both their physical and psychological health. Since they tend to spend the majority of their time at home, their social relationships are also penalized. Loneliness is proved to be a cause (not only a consequence) of reduced physical activity [44], therefore a self-reinforcing loop damaging physical health of the senior is generated. Loneliness is also extremely negative on human cognition (see Figure 1.5). Social isolation induces the individual to perceive the social environment as threatening. This stimulates neurobiological mechanisms that increase the activation of the hypothalamic pituitary adrenal (HPA) axis and diminish sleep quality [23].

1.1 Project Acanto

A promising way to face these nefarious effects of reduced mobility is the combination of deambulation aids and modern robotic technologies. The European project ACANTO [1] aims to develop an assistive robotic walker, called *FriWalk*, capable of physically and cognitively supporting the assisted person. The *FriWalk* is similar to a standard four-wheeled rollator (see figures 1.7 and 1.8), but it is endowed with actuators, sensors, and computing abilities to localize itself in the environment, and to compute and follow safe and comfortable paths to guide the user towards a desired

location. For example, the user asks the robot (using a tablet-based interface) to be guided towards the restrooms inside a building. The robot offers physical support and helps the user to reach the restrooms, ensuring that he/she moves safely, e.g., by avoiding obstacles. Moreover, ACANTO emphasizes the importance of social interaction among the users. Since social interaction is fundamental for correct aging, a dedicated social network is under development in ACANTO, e.g., to bring together two people with similar profiles. Overall, the workflow of ACANTO is depicted in Figure 1.6.

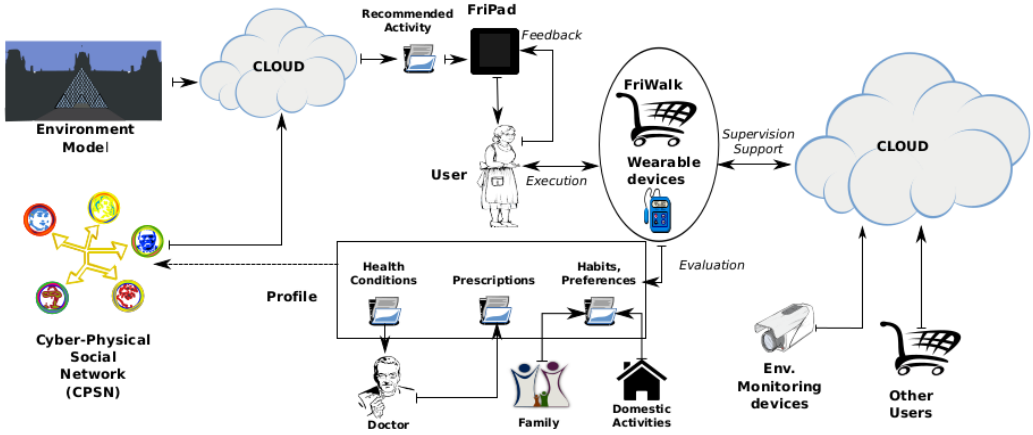

Figure 1.6: Workflow of ACANTO, from the ACANTO proposal.

Figure 1.7: *FriWalk* prototype equipped with the tablet interface [8].

Figure 1.8: *FriWalk* and user during experimental campaign validation within ACANTO [8].

1.2 Why authority-sharing?

An assistive robotic walker cannot act as a fully autonomous vehicle by excluding the user from the loop, since the user's residual abilities must be exploited and practiced. Conversely, the robot is supposed to guarantee user's safety, therefore the responsibility of controlling the vehicle motion cannot be entirely left to the assisted person.

This research aims to guarantee safety while leaving the user in the loop using control algorithms based on the authority-sharing paradigm [41, 22, 28], in which the control authority, i.e., the capability of controlling the vehicle motion, is shared between the human user and the control system. Ideally, the assistive robotic walker applying authority-sharing provides the user just the help he/she needs. For instance, if the user requires just physical support to reach the restrooms inside a building, the robot is supposed to act as a standard rollator; however, if the user's cognitive abilities are limited (e.g., the user does not remember where the restrooms are or he/she does not see an obstacle), the robot should also drive the user towards the proper corridors, by planning and following a safe path to the restrooms. This way, user's residual abilities are exploited and kept in training, the feeling of autonomy is enhanced, and safety is guaranteed.

1.3 Thesis organization

This essay is organized as follows. Chapter 2 presents the preliminary topics of the thesis, ranging form the models adopted in the research to describe the assistive walker, the existing guidance solutions for a robotic vehicle, and the authority-sharing strategies available in the literature. Chapter 3 introduces the robotic platforms used within this research (and in general, in the context of ACANTO), and the planning and localization modules implemented in the robot, which work synergically with the control module developed in this thesis. The proposed authority-sharing framework and the main theoretical contributions of this research are extensively described and analyzed in Chapter 4. In particular, the chapter introduces the attitude-based authority-sharing approach, which enables us to design authority-sharing controllers in the following chapters, on the basis of the available actuators on the robot. Chapter 5 applies the proposed framework when the robot is endowed just with a low-cost emergency braking system and a simple haptic interface based on vibrating bracelets. Chapter 6 proposes a guidance solution using active thrusting motors. Chapter 7 presents an authority-sharing strategy based on a passive walker equipped with front steering wheels. An extensive experimental validation is proposed in all these chapters. Further studies on authority-sharing are presented in chapters 8 and 9. Chapter 8 extends and validates our approach in a stochastic framework. Chapter 9 proposes the innovative application of authority-sharing in handling the motion of a group of humans assisted with robotic walkers. The last two studies are validated with preliminary experiments and extensive simulations. The thesis is concluded with some final remarks and discussions of the future research directions in Chapter 10.

1.3.1 Scientific contributions from the thesis

Our research contribution produced 9 scientific papers, comprising current submissions and conferences. Several chapters of the thesis are based on the scientific results obtained in the papers. Table 1.1 relates the produced scientific contributions with the chapters of the thesis.

1.3.2 Patents from the thesis

The main part of Chapter 7 presents an innovative authority-sharing guidance strategy which is currently under review for patent pending. As a consequence, the related paper has not been submitted yet and Chapter 7 is confidential.

Reference	Appearing in	Thesis main chapter
[10]	CDC	Chapter 5
[11]	IROS	Chapter 5
[5]	Robotics and Automation Letters	Chapter 5
[8]	Transaction on Haptics	Chapter 5
[6]	IROS	Chapter 6
[9]	Robotics and Automation Letters	Chapter 6
[12]	ICRA	Chapter 7
[72]	ICRA	Chapter 8
[7]	Transaction on Robotics	Chapter 9

Table 1.1: Scientific contributions of the thesis.

Chapter 2

Problem formulation and state of the art

2.1 Vehicle modeling

To design authority-sharing controllers, we describe the vehicle using standard non-holonomic models. The most adopted dynamic model for the *FriWalk* is the differential kinematics of a unicycle-like robot [103]. With reference to Figure 2.1, let $\{O_w, X_w, Y_w, Z_w\}$ be the world reference frame having the axis Z_w orthogonal to the plane of motion and the origin O_w located on the plane of motion. Let (x, y) be the coordinates of the vehicle reference point O_m with respect to the world frame, i.e., $O_m = [x, y, 0]^\top$ in the world frame. Accordingly to [103], the vehicle reference point is located in the mid point of the axle of the vehicle, which is the segment connecting the contact points of the non-caster wheels (which are the rear wheels for the *FriWalk*). The vehicle yaw θ (called also orientation or attitude) is defined as the orientation with respect to the world frame of a mobile reference frame $\{O_m, X_m, Y_m, Z_m\}$ attached to the vehicle, having Z-axis Z_m parallel to Z_w, and X-axis X_m oriented in the direction of motion. The positioning of mobile reference frame $\{O_m, X_m, Y_m, Z_m\}$ with respect to the real robot is shown in figures 2.2 and 2.3. Under the hypothesis of pure rolling motion of the wheels, the differential kinematics of a unicycle-like robot is

$$\begin{cases} \dot{x} &= v\cos\theta, \\ \dot{y} &= v\sin\theta, \\ \dot{\theta} &= \omega, \end{cases} \tag{2.1}$$

where the scalar quantities v and ω are the forward and the angular velocity of the vehicle, respectively. The forward velocity v is positive in case of forward motion, while the angular velocity ω is positive in case of counter-clockwise rotation. Model (2.1) is properly called differential kinematics, since, from a physical point of

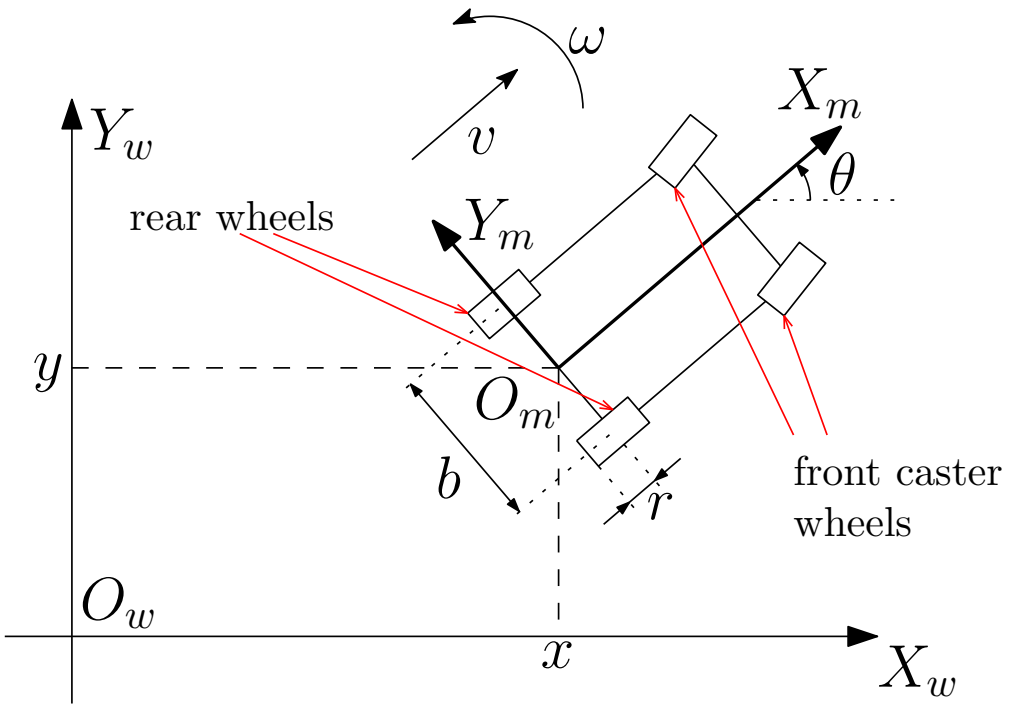

Figure 2.1: Unicycle modeling. The yaw θ is positive in the figure.

view, it is a kinematic model only, not considering vehicle inertia. However, from a mathematical point of view, equations (2.1) define a dynamic model since they are differential, the state of the system is $[x, y, \theta]^\top$, while the control inputs are the velocities v and ω. The control inputs are typically expressed as function of the angular velocities of the wheels. Let ω_R and ω_L the angular velocity of the rear right and left wheels, respectively. Under the hypothesis of pure rolling motion, the relation between vehicle velocities and wheel velocities is

$$\begin{aligned} v &= \frac{r(\omega_R + \omega_L)}{2}, \\ \omega &= \frac{r(\omega_R - \omega_L)}{b}, \end{aligned} \tag{2.2}$$

where $b > 0$ is the axle length and $r > 0$ is the wheel radius. A unicycle robot is also defined differentially driven, since its moving velocities depend on the sum/difference of its wheel velocities. In order to impose v and ω using the wheel velocities, each wheel has to be coupled with an independent motor.

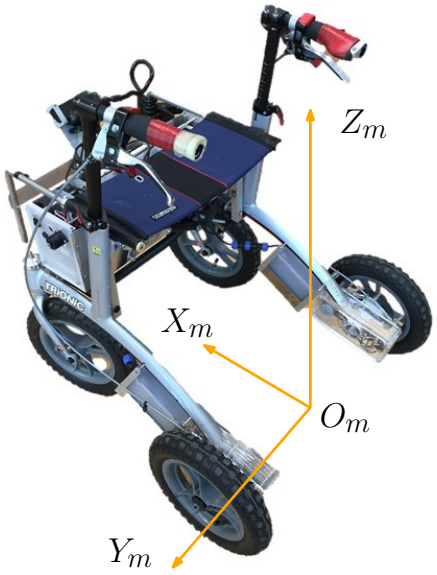

Figure 2.2: Positioning of mobile reference frame $\{O_m, X_m, Y_m, Z_m\}$ with respect to the real robot.

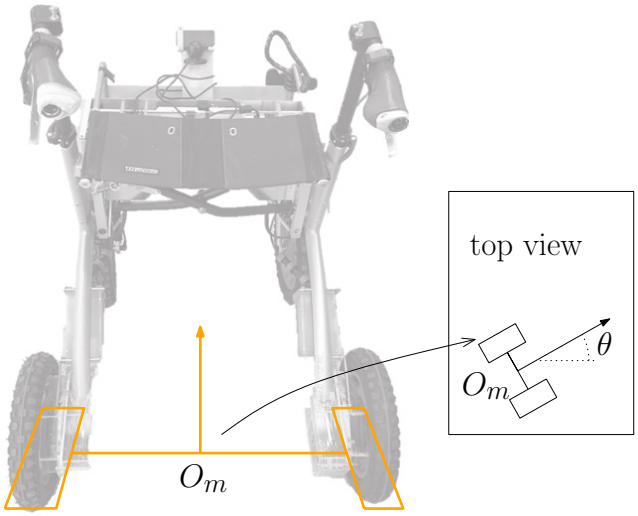

Figure 2.3: Schematic representation of the robot.

Front steering wheels

The front wheels of the *FriWalk* are equipped with motors, and can be used to steer the vehicle. In this case, the attitude dynamics of the vehicle (third equation in

Model (2.1)) is replaced by

$$\dot{\theta} = \frac{v}{d} \tan \varphi, \tag{2.3}$$

where φ is the steering angle of the virtual front wheel and $d > 0$ is the distance between the reference point O_m and the contact point of the virtual front wheel. If the motor can be commanded in position, the steering angle φ acts as control input. If the vehicle has two steering wheels (as the *FriWalk*), the virtual steering wheel is generated by imposing the Ackerman steering condition to the real steering wheels. The virtual steering angle φ is related to the left and right wheel angles φ_l and φ_r to ensure that the instantaneous center of rotation (ICR) of the vehicle is properly positioned (see Figure 2.4).

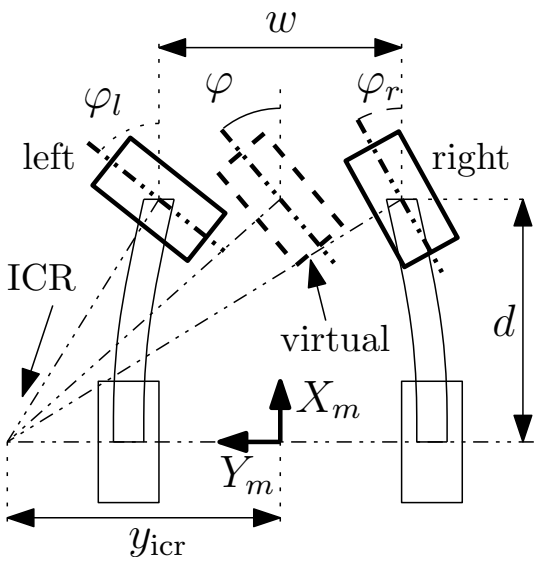

Figure 2.4: Steering configuration of the vehicle [13].

Let y_{icr} be the Y coordinate of the vehicle ICR in the vehicle frame $\{O_m, X_m, Y_m, Z_m\}$. By applying standard geometrical relations to the triangles in Figure 2.4, we get

$$\begin{aligned} y_{\text{icr}} &= \frac{d}{\tan(\varphi)}, \\ \varphi_l &= \arctan\left(\frac{d}{y_{\text{icr}} - \frac{w}{2}}\right), \\ \varphi_r &= \arctan\left(\frac{d}{y_{\text{icr}} + \frac{w}{2}}\right), \end{aligned} \tag{2.4}$$

where $w > 0$ is the distance between the contact points of the front steering wheels. If the motors of the front wheels are controlled in velocity, the virtual steering

angle φ acts as a state, and the attitude dynamics of the vehicle (third equation in Model (2.1)) is replaced by

$$\begin{aligned} \dot{\theta} &= \frac{v}{d}\tan\varphi, \\ \dot{\varphi} &= u_v, \end{aligned} \tag{2.5}$$

where the control input u_v is the commanded steering velocity of the virtual wheel. The relation with steering velocities of the real wheels is

$$\begin{aligned} \dot{\varphi}_l &= \frac{d^2\csc^2(\varphi)\dot{\varphi}}{(d\cot(\varphi)-\frac{w}{2})^2\left(\frac{d^2}{(d\cot(\varphi)-\frac{w}{2})^2}+1\right)}, \\ \dot{\varphi}_r &= \frac{d^2\csc^2(\varphi)\dot{\varphi}}{(d\cot(\varphi)+\frac{w}{2})^2\left(\frac{d^2}{(d\cot(\varphi)+\frac{w}{2})^2}+1\right)}. \end{aligned} \tag{2.6}$$

Notice that Relation (2.6) is not defined for $\varphi = 0$, however

$$\lim_{\varphi\to 0}\dot{\varphi}_l = \lim_{\varphi\to 0}\dot{\varphi}_r = \dot{\varphi},$$

hence Relation (2.6) can be easily implemented with a conditional instruction.

2.2 Path following

Path following is a typical problem of mobile robotics and is also a focus of this research. ACANTO is a typical scenario where the robot is required to solve a path following problem. Suppose for instance that a user requires to get to a specific room in a building. The robot is expected to plan and follow a path connecting its current location with the room. When solving a path following problem, we assume that the path is given as explained in Chapter 3.

2.2.1 Problem definition

The path is a curve in the plane parameterized by a curvilinear abscissa s. Formally a path $\Gamma : s \to [x_d(s), y_d(s), \theta_d(s)]^\top$ is a smooth function associating for each value of the curvilinear abscissa a desired X-coordinate $x_d(s)$, a desired Y coordinate $y_d(s)$, and a desired orientation $\theta_d(s)$. The path curvature $c(s)$ is defined as

$$c(s) = \frac{\mathrm{d}\,\theta_d}{\mathrm{d}\,s}(s).$$

Let us denote by t the standard time. We define the path following problem as follows.

Given a path Γ and a nonholonomic vehicle (2.1), find the vehicle velocities $v(t)$ and $\omega(t)$, and a time law $s(t)$ for the curvilinear abscissa, such that

$$\begin{aligned} \lim_{t\to+\infty} x(t) &= x_d(s(t)), \\ \lim_{t\to+\infty} y(t) &= y_d(s(t)), \\ \lim_{t\to+\infty} \theta(t) &= \theta_d(s(t)), \end{aligned} \tag{2.7}$$

Notice that the these conditions in (2.7) state that the vehicle position $[x, y]$ and attitude θ asymptotically converge to the desired values $[x_d(s), y_d(s), \theta_d(s)]^\top$ defined by the path (i.e., that the vehicle is located on the path and properly oriented). Then, a further condition

$$\lim_{t\to+\infty} v(t) > 0,$$

is added to require that the overall motion of the vehicle is forward (this avoids the undesired situation where the vehicle is on the path moving in the wrong direction).

Formulation (2.7) does not consider the presence of a human agent. Since the user cannot reach zero steady state error as in (2.7), the problem will be reformulated in Chapter 4.

2.2.2 Frenet frame

To properly represent a path following problem, it is a common practice to describe the robot using a reference frame $\{O_f, X_f, Y_f, Z_f\}$, called *Frenet frame* [64], moving along the path by following the vehicle motion. The origin O_f of the Frenet frame is located on the path (then it can be identified by the curvilinear abscissa s), and its X_f axis is tangent to the path (hence X_f has the desired orientation in the path point overlapping O_f, i.e., $\theta_d(s)$). Two methods are available to locate the Frenet frame on the path: either the Frenet frame position can be statically computed (i.e., it depends just on the position of the vehicle), or the position of the Frenet frame acts as a dynamical state (i.e., the positioning depends both on the vehicle state $[x, y, \theta]^\top$ and the past history).

Static Frenet frame

The simplest way to locate the Frenet frame on the path is to place its origin O_f in the point of the path closest to the vehicle reference point (see Figure 2.5). This way, the X-coordinate of the vehicle reference point O_m in the Frenet frame is always zero, and the Y-coordinate of the vehicle reference point O_m with respect to the Frenet frame, denoted by l, is the distance (with sign) of the vehicle from the path. Let $\tilde{\theta} := \theta - \theta_d$ be the vehicle attitude with respect to the Frenet frame. The path

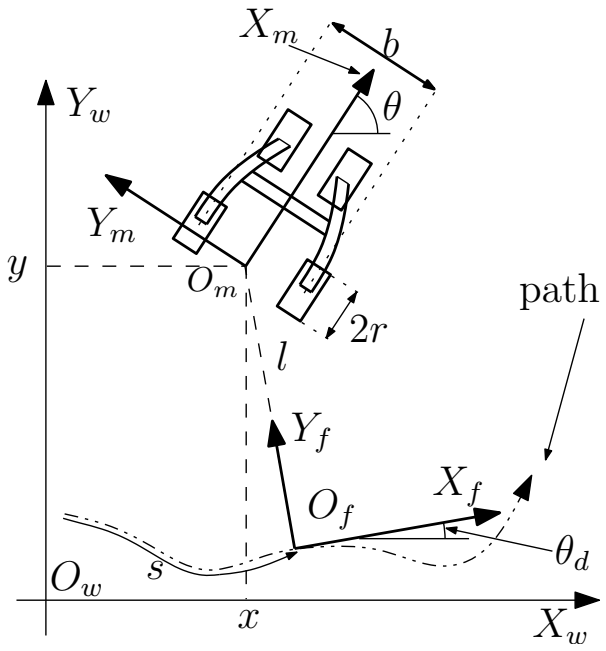

Figure 2.5: Positioning of the Frenet frame in the point of the path closest to the vehicle [6].

following problem is represented with respect to the Frenet frame by replacing the vehicle state $[x, y, \theta]^\top$ in (2.1) with the coordinates $[s, l, \tilde{\theta}]^\top$, having dynamics

$$\begin{cases} \dot{s} &= v\dfrac{\cos(\tilde{\theta})}{1 - c(s)l}, \\ \dot{l} &= v\sin(\tilde{\theta}), \\ \dot{\tilde{\theta}} &= \omega - c(s)\dot{s}. \end{cases} \tag{2.8}$$

Using this new set of coordinates (2.8), the path following conditions (2.7) are rewritten as the following asymptotic stability problem:

$$\begin{aligned} \lim_{t\to+\infty} l(t) &= 0, \\ \lim_{t\to+\infty} \tilde{\theta}(t) &= 0. \end{aligned} \tag{2.9}$$

Notice that the curvilinear abscissa s of the Frenet frame can be computed by calculating a geometrical problem, i.e., by founding the point of the path closest to the vehicle. The velocity $\dot{s}$ of the Frenet frame along the path is then determined by the vehicle velocity v and the position of the robot with respect to the path. This does not require the explicit design of $\dot{s}$, however it yields to a singularity in the expression of $\dot{s}$ in (2.9), kicking in whenever $1 - c(s)l = 0$, i.e., the vehicle is located

in the center of the curvature. Intuitively, this singularity takes place since if the vehicle is located at the center of a circular path, the point of the path closest to the vehicle is not well defined. To understand the practical meaning of the singularity, consider the vehicle moving close to the curvature center in Figure 2.6. Notice that, although between the time instants t_1 and t_2 the vehicle has performed a small displacement, the closest point of the path to the vehicle has traveled almost all the circular path. Then, even if the vehicle is very slow, the moving velocity of the Frenet frame can be arbitrarily large. The presence of the singularity may generate sudden and undesired motion of the vehicle. Then, theoretically, the singularity issue is of primary importance due to the particular class of users of the device, which are elderly with possible physical impediments, and hence particularly fragile. In a real practical scenario, the situation in which the vehicle is located in the curvature center is rather improbable (it has never taken place during our experimental tests with static Frenet frame [5, 6, 12, 11]). However, in order to guarantee the user's safety, the singularity should not be present. This safety issue is totally overcome using a dynamic Frenet frame.

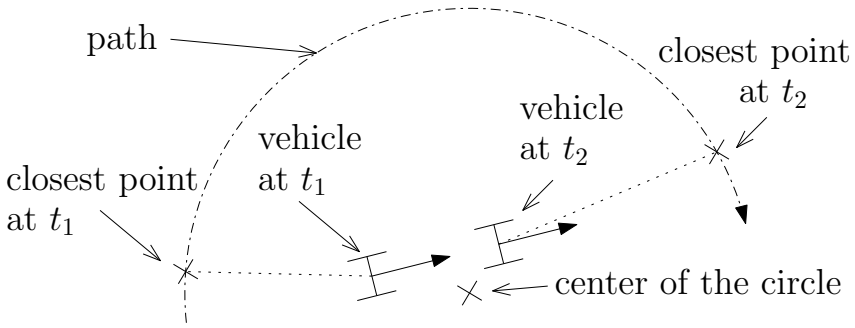

Figure 2.6: Vehicle moving close to the singular condition.

Dynamic Frenet frame

The Frenet frame can be located in an arbitrary point of the path (see Figure 2.7). In this case, the X-coordinate of the vehicle reference point in the Frenet frame is not necessarily zero. We denote by (l_x, l_y) the coordinates of the vehicle reference point O_m with respect to the Frenet frame placed in an arbitrary point of the path. Using the set of coordinates $[l_x, l_y, \tilde{\theta}]^\top$ instead of $[x, y, \theta]^\top$ in (2.1), the dynamic is written as [106]

$$\begin{cases} \dot{l}_x &= -\dot{s}(1 - c(s)l_y) + v\cos\tilde{\theta}, \\ \dot{l}_y &= -c(s)\,\dot{s}\,l_x + v\sin\tilde{\theta}, \\ \dot{\tilde{\theta}} &= \omega - c(s)\dot{s}. \end{cases} \tag{2.10}$$

Notice that the singularity has been removed. However the moving velocity of the Frenet frame $\dot{s}$ acts as auxiliary control input and then has to be designed. The

curvilinear abscissa s of the Frenet frame is obtained by integrating the auxiliary control input $\dot{s}$, i.e.,

$$s(t) = s(0) + \int_0^t \dot{s}\,\mathrm{d}\,\bar{t},$$

where $\bar{t}$ is a dummy variable and $s(0)$ is the initial curvilinear abscissa of the Frenet frame, which has to be initialized.

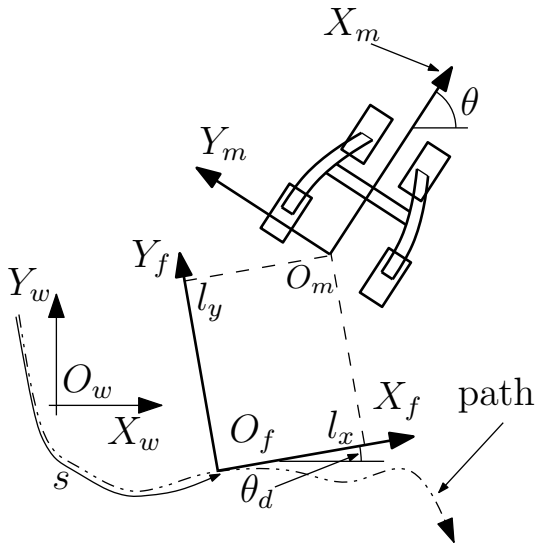

Figure 2.7: Positioning of the Frenet frame in an arbitrary point of the path.

Similarly to (2.9), the path following can be represented as asymptotic stability problem using the new set of coordinates $[l_x, l_y, \tilde{\theta}]^\top$, hence

$$\begin{aligned} \lim_{t\to+\infty} l_x(t) &= 0, \\ \lim_{t\to+\infty} l_y(t) &= 0, \\ \lim_{t\to+\infty} \tilde{\theta}(t) &= 0. \end{aligned} \tag{2.11}$$

Notice that, although the singularity issue is no more present, the controller has to stabilize three states (i.e., $[l_x, l_y, \tilde{\theta}]^\top$) instead of two as in (2.9).

2.2.3 Approaching angle solution

A common practice [81, 106] to deal with the path following problem is the use of an approaching angle function $\delta(\cdot)$, which is used as position-dependent attitude reference to solve the path following as an attitude tracking.

The approaching angle $\delta(\cdot)$ defines the attitude, expressed with respect to the Frenet frame, that the vehicle should have to approach and follow the path as

function of the lateral distance from the path (l_y for a dynamic Frenet frame, l for a static Frenet frame). Figure 2.8 depicts the approach to the path using approaching angle functions $\delta(\cdot)$. Notice that the larger the distance from the path, the larger $|\delta(l_y)|$. Moreover notice that, to ensure that the vehicle is steered towards the path, $\delta(l_y)l_y < 0$ if the vehicle is not on the path.

Consider for simplicity a path following problem parametrized using a static Frenet frame (2.8). It is possible to prove that, if the approaching angle function $\delta(\cdot)$ is continuous and odd, and satisfies $\delta(l)l < 0, \delta(0) = 0$ (e.g., as in Figure 2.9), then the path following conditions (2.9) hold if the attitude error $e_\theta = \tilde{\theta} - \delta(l)$ is equal to zero, i.e. $\tilde{\theta} = \delta(l)$. In other words, the path following problem is solved if the attitude $\tilde{\theta}$ of the vehicle tracks a proper reference $\delta(l)$.

In the case of dynamic Frenet frame (2.10) the properties of the approaching angle $\delta(l_y)$ are the same, however, to ensure the path following conditions (2.11), a proper design of the moving velocity $\dot{s}$ of the Frenet frame is needed to stabilize the longitudinal distance from the path l_x.

The approaching angle is not defined considering the presence of the human. However, we will use the approaching angle to define a user's behavior *desired* by the robot.

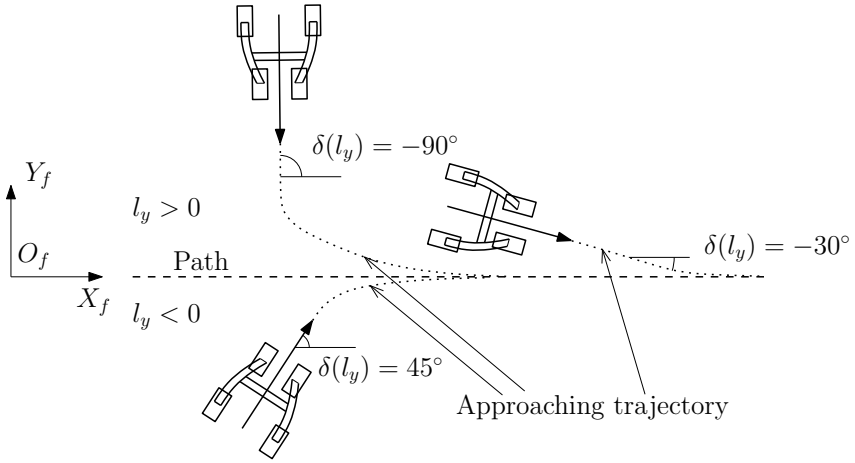

Figure 2.8: Possible approaching maneuvers corresponding to different functions $\delta(\cdot)$.

2.2.4 Example of path following controller

An example of performing path following controller based on a dynamic Frenet frame (2.10) is [106], as stated in the following theorem.

Theorem 1 (State of the art path following [106])**.** *Consider vehicle* (2.1) *described using a dynamic Frenet frame* (2.10)*. Pick an approaching angle* $\delta(\cdot)$ *as a continuous*

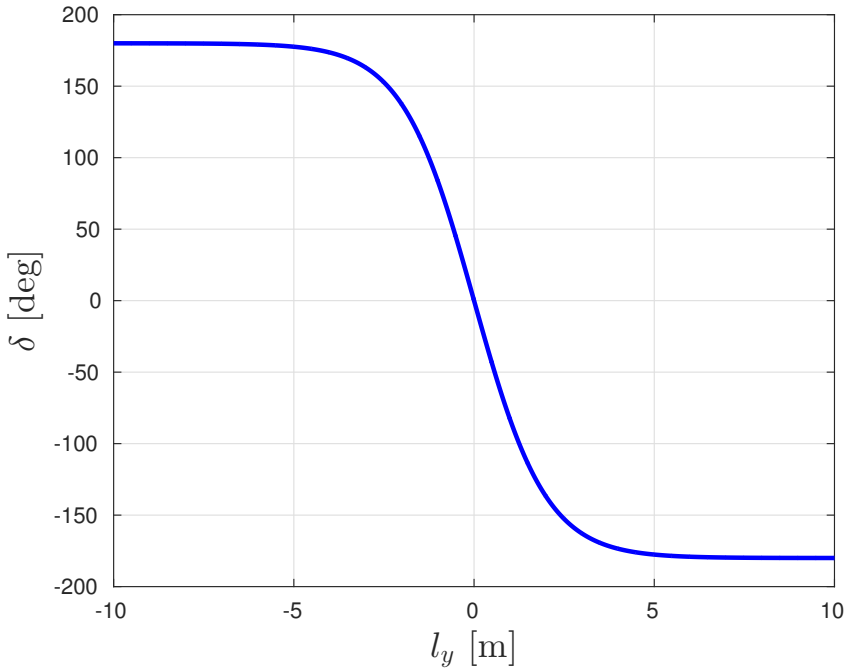

Figure 2.9: Approaching angle function $\delta(l_y) = -\frac{\pi}{2}\tanh\left(\frac{l_y}{2}\right)$.

and odd function of l_y *satisfying* $\delta(l_y)l_y \leq 0$ *for all* $l_y \in \mathbb{R}$ *and* $\delta(0) = 0$. *Assume the vehicle velocity satisfies* $\lim_{t\to+\infty} v(t) > 0$. *Then the classical path following problem* (2.11) *is solved by the control inputs*

$$\begin{aligned} \dot{s} &= v\,\cos(\tilde{\theta}) + \kappa_1 l_x, \\ \omega &= c(s)\dot{s} + \dot{\delta} - \kappa_3 l_y v \frac{\sin(\tilde{\theta}) - \sin(\delta)}{\tilde{\theta} - \delta} - \kappa_2(\tilde{\theta} - \delta), \end{aligned} \tag{2.12}$$

where κ_1, κ_2, *and* κ_3 *are positive gains.*

Proof of Theorem 1. The proof is based on the Lyapunov function [106]

$$\overline{V} = \frac{1}{2}(l_x^2 + l_y^2) + \frac{1}{2\kappa_3}(\tilde{\theta} - \delta)^2, \tag{2.13}$$

whose time derivative along the solutions, in accordance with (2.10), is

$$\begin{aligned} \dot{\overline{V}} &= l_x\dot{l}_x + l_y\dot{l}_y + \frac{1}{\kappa_3}(\tilde{\theta} - \delta)(\dot{\tilde{\theta}} - \dot{\delta}) \\ &= l_x(-\dot{s}(1 - c(s)l_y) + v\cos\tilde{\theta}) + l_y(-c(s)\,\dot{s}\,l_x + v\sin\tilde{\theta}) + \frac{1}{\kappa_3}(\tilde{\theta} - \delta)(\dot{\tilde{\theta}} - \dot{\delta}) \\ &= l_x(-\dot{s} + v\cos(\tilde{\theta})) + l_y v\sin(\tilde{\theta}) + \frac{1}{\kappa_3}(\tilde{\theta} - \delta)(\omega - c(s)\dot{s} - \dot{\delta}). \end{aligned}$$

We substitute $\dot{s}$ from (2.12) and $\sin(\tilde{\theta}) = \sin(\tilde{\theta}) - \sin(\delta) + \sin(\delta)$. We get

$$\begin{aligned}\dot{\overline{V}} &= -\kappa_1 l_x^2 + l_y v(\sin(\tilde{\theta}) - \sin(\delta) + \sin(\delta)) + \frac{1}{\kappa_3}(\tilde{\theta} - \delta)(\omega - c(s)\dot{s} - \dot{\delta}) \\ &= -\kappa_1 l_x^2 + l_y v \sin(\delta) + \frac{1}{\kappa_3}(\tilde{\theta} - \delta)\left(\omega - c(s)\dot{s} - \dot{\delta} + v l_y \kappa_3 \frac{\sin(\tilde{\theta}) - \sin(\delta)}{\tilde{\theta} - \delta}\right).\end{aligned}$$

We finally substitute ω from (2.12) and get

$$\dot{\overline{V}} = -\kappa_1 l_x^2 + l_y v \sin(\delta) - \frac{\kappa_2}{\kappa_3}(\tilde{\theta} - \delta)^2.$$

Condition $\dot{\overline{V}} \leq 0$ (and hence stability) follows from the assumptions on v and δ and standard Lyapunov arguments. □

2.2.5 Comments on simplified attitude control

It is remarked that the path following control presented in robotic literature is commonly based on differences between two angles (for instance $\tilde{\theta} - \delta$ in (2.12)). To have a proper behavior it is obviously needed to redefine the difference between angles, to proper represent the correct angular distance. The same operation is also needed in the definition of the vehicle attitude $\tilde{\theta}$ expressed in Frenet frame. An example of function redefining the difference between angle is reported in Listing 2.1. Notice that the reported function produces, for instance, the following outputs:

$$\begin{aligned}360^\circ - 0^\circ &= 0^\circ, \\ -170^\circ - 20^\circ &= 170^\circ, \\ 179^\circ - (-179^\circ) &= -2^\circ,\end{aligned}$$

which are always smaller or equal to a straight angle. Moreover the sign of the difference is always representative of the direction of the actuation. For instance, if $\tilde{\theta} = 179^\circ$ and $\delta = -90^\circ$, the function returns $\tilde{\theta} - \delta = 179^\circ - (-90^\circ) = -91^\circ < 0$. This means that the controller has to rotate the vehicle attitude $\tilde{\theta}$ in the negative direction (i.e., clockwise) of 91°. Clearly, by computing $179^\circ - (-90^\circ) = +269^\circ$, the controller is required to perform a positive (counterclockwise) rotation larger that a straight angle. This undesired phenomenon is well known in attitude control and is called *unwinding*. However, although several formal control solution have been proposed to face the problem [25], in robotics, it is preferred to represent the attitude difference in this more intuitive way.

```
function difference = diff_angle(beta,alpha)
% compute beta minus alpha
%% set alpha, beta between 0 and 2*pi
data = [alpha;beta];
```

```
for i = 1:length(data)
    angle = data(i);
    while abs(angle) > 2*pi
        angle = angle - 2*pi*sign(angle);
    end
    data(i) = angle;
    if data(i) < 0
        data(i) = data(i) + 2*pi;
    end
end
alpha = data(1);
beta  = data(2);

%% Compute difference

difference = beta - alpha;

if difference > pi
    difference = difference - 2*pi;
end

if difference < -pi
    difference = difference + 2*pi;
end
```

Listing 2.1: Example of Matlab code implementing the difference between two angles expressed in radians.

It is also remarked that robust global attitude stabilization is impossible using a continuous memory-less feedback as in Theorem 1 (see [78]). As a consequence, even results designed using a Lyapunov based approach (as in Theorem 1 from [106]) may not be considered global and robust. For instance, controller (2.12) may experience chattering problems in the presence of arbitrarily small measurement noise for attitude errors of 180°. These theoretical issues (which are rather improbable in practice but crucial for a formal analysis) are neglected in this thesis to promote a simpler and more intuitive structure of the proposed control algorithms.

2.3 Classification of authority-sharing strategies

In general, we have four ways to apply authority-sharing in the literature:

1. the user has full control of the vehicle motion and the robot simply suggests the user how to act (e.g., via haptic systems);

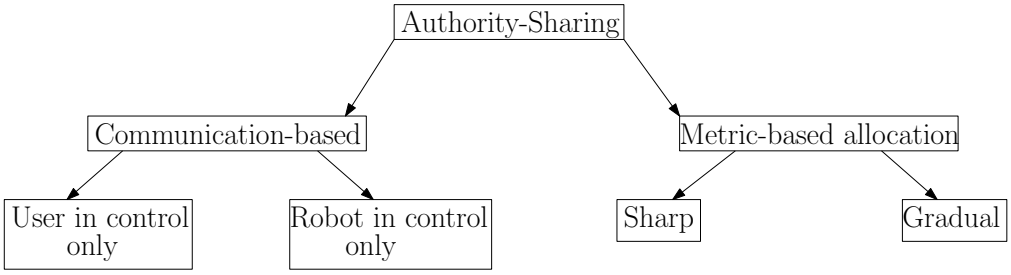

Figure 2.10: Classification of the authority-sharing strategies.

2. the robot has full control of its motion and the user simply suggests the robot how to act (via a system inferring user's intentions);

3. the control authority is completely given to the user or to the robot depending on the situation described by a proper metric (for instance the robot intervenes in dangerous situations only, e.g., close to the obstacles);

4. the control authority is given both to user and robot in the same time, but the relative importance is tuned on the basis of a given metric.

In the first two cases, only one agent (either the user or the robot) can directly act on the motion. We define this authority-sharing paradigm as *communication-based*, since one agent simply suggests the other the proper behavior (for instance the robot communicates with the user via haptic interfaces or the user commands the robot using a joystick).

In the last two cases, both the user and the robot can modify the vehicle motion depending on a given metric. Then, we defined this authority-sharing paradigm as *metric-based*. The allocation is defined *sharp* if the authority-sharing algorithm decides who sharply retains the control authority and establishes when switching such control authority from the robot to the user (or vice versa). For instance, if the metric is the distance from the obstacles, the control authority is allocated to the robot only close to the obstacles. The allocation is defined *gradual* if the metric is used to tune the relative importance between user and robot while they are both acting on the vehicle motion at the same time. For example, if the metric is the distance from the obstacles, the robot has a larger amount of the control authority close to the obstacles.

Clearly, a gradual transition of the authority from the robot to the person (or vice versa) is desired, but arguably much more challenging in terms of design.

2.4 Authority-sharing applications

The main issue in sharing the control authority with a human is that he/she may exhibit irrational or even dangerous low-level behavior (since in general, his/her

sensing system is slower that the sensor system of the robot). As a consequence, the robots are typically more reliable in performing low-level operations, e.g., obstacle avoidance. However, humans have a superior high-level decisional system, capable of rapidly generating heuristics to follow in the presence of unpredictable situations.

The *communication-based* with human in control authority-sharing exploits both advantages by letting the actual control authority to the human and endowing the robot with the capability of suggesting the human the desired behavior via, for instance, an haptic interface. The effectiveness of using haptic shared control is analyzed in [2]. The use of haptic technologies to assist a car driver has been extensively studied. In [110] bi-directional communication is ensured by the control interface, i.e., an haptic steering wheel, used by the human both to control the vehicle and receive suggestions. Experimental results run with a driving simulator show that the haptic system reduces the visual demand and improves the path-following performance. Further experiments are presented in [43], where the reduction of visual demand and the improvement of path following-performance are confirmed, while an improvement in the driver reaction time is also proved. In [85] the steering wheel acts again as haptic interface but the control system is capable of applying torques to it (similarly to the driver). Then, the control system continuously applies a steering torque on curve but the driver is still in the loop. The experiments, run on 12 young and experienced drivers, show again an improvement in the steering performance. Similar results are obtained by testing the same approach on elderly drivers in [84].

In the *communication-based* with robot in control authority-sharing, the user communicates his/her desired action, which is then executed by the robot. This strategy has been used in robotic walker control and it is linked to the inference of the user intentions [122]. In [79], user's intentions are estimated via two 6-degrees-of-freedom force and moment sensors mounted on the handles and a digital motion capture system. The experiments underline a strong correlation between the torque applied by the user with the intention to turn (related to the heading angle). In [52, 51], the user's state (walking state for a moving user, stooping state if the user does not want to move even if he/she applying forces to the robot, and emergency state) is estimated to choose the control strategy. The user's intentions are studied by a rotating infrared sensor to detect the user's lower limb movement in [65], and by a bar-like interface based on two potentiometers in [76]. In [82] user's short-term intended behavior is inferred using a Partially Observable Markov Decision Process based on minimal user input. Joysticks are used to estimate user intentions in [75, 74]. An further advance study on user's intention inference run with a more complex system including serevar sensors is proposed in [92].

In control of robotic walkers, sensors estimating forces and torques applied by the user are often used to estimate user's intentions. The sensed forces can be used to modify either vehicle velocity or vehicle acceleration [102]. In [42] the force information is elaborated to interpret the person's intentions both in terms

of longitudinal velocity (controlled via actuated wheels) and desired heading. The user's intentions are supported by allowing a mechanism of path deformation (in a path following problem); path deformation is used also in [120]. Force-sensing handles are combined with an intelligent learning scheme to properly drive the user in [71]. In [26, 107], force/torque sensing is combined with a learning scheme to apply a control action to the vehicle based on the definition of the instantaneous center of rotation.

The same *communication-based* with robot in control authority-sharing paradigm is used also in [104] for a smart wheelchair voice-controlled. A probabilistic reasoning construct is adopted to allocate the control authority between the wheelchair and the user.

In the *metric-based* authority-sharing, the control authority is given both to user and robot in the same time, but the relative importance is tuned on the basis of a given metric. A *sharp* solution is proposed in [79] for a robotic walker, where the user is provided with the maximum possible control authority, and the control system is activated and overrides the human commands only if it is needed to ensure the person's safety, for example to avoid an obstacle.

Control strategies based on *gradual* allocation fuse the human inputs with the robot inputs. In [55] the forces applied by the user to a robotic walker are estimated via motor currents and angular velocities, without using costly force/torque sensors. The velocity applied by the actuation system is obtained from a convex combination between the human forces and the forces that the robot would apply for autonomous navigation. The weights are dynamically changed to give more authority to the robot in dangerous situation (i.e., close to the obstacles and in the presence of large velocities). A similar approached is follow in [123], where the weights are tuned on the basis of the user's performance (e.g., the walker behaves as the user commands if he/she is properly follow the desired path).

2.5 Further references

The main goal of ACANTO is the develop of a robotic walker. A survey on assistive robotic walkers is available in [77].

Several assistive walkers are also omnidirectional robots. Combinations of these platforms with force sensing are described in [48]. In [108] the force are estimated on the basis of motor currents instead of force sensors. The controller developed in [113, 114] is designed to face modifications of the center of gravity of the system (including user and robot), for instance due to a load change.

Further studies on assistive robotic walkers range from fall prevention [50, 112], guidance of blind people [119], interaction with the user via voice commands [38], to adaptive control [119].

Control-sharing is applied to robots used in teleoperations (see e.g. [59, 45]) especially in the presence of actuation delay, since it is typical in space application.

Chapter 3

Robotic walker in Acanto

The development of the assistive robotic walker in ACANTO required a multidisciplinary approach. The main points on which the ACANTO team worked are the following:

- development of an innovative robotic hardware. The prototype is equipped with several actuation systems to test the most promising ways to guide the assisted person. This is the main topic covered in the PhD thesis of Stefano Divan;
- development of a suitable localization and sensing system: clearly, the robot needs to localize itself in the environment and to recognize obstacles. This is the main topic covered in the PhD thesis of Valerio Magnago;
- development of a planning strategy. A path planner is needed to define a safe path connecting the actual robot position with a point of interest required by the assisted person. This is the main topic covered in the PhD thesis of Paolo Bevilacqua;
- development of authority-sharing control strategies. Given the available actuators, the information on the vehicle localization, and a plan to follow, the robot has to guide the user towards the desired location by sharing the control authority.

This research is focused on authority-sharing control. However, because of the adopted multidisciplinary approach, a knowledge of the overall system is required to properly design the control strategies. This chapter provides an overview on the robotic platform, on the vehicle localization, and on the path planner.

3.1 Robotic hardware

Preliminary tests have been run on the low-cost robotic walker in Figure 3.1. This robot was equipped with two electromechanical brakes mounted on the rear wheels

Figure 3.1: Preliminary robotic hardware [12].

and two stepper motors controllable in position to steer the front wheels. This hardware allows the test of passive guidances only (i.e., without propelling motors), based on front steering or differential braking. Although some interesting results were obtained [12, 5, 11] using the platform in Figure 3.1, a more advance hardware has been developed to implement more sophisticated strategies. The ultimate ACANTO prototype is depicted in Figure 3.2. The robot is equipped with the following actuators:

- motors to steer the front wheels (Figure 3.4). The driver allows us to command these motors in position, velocity or current (torque);
- motors to propel the rear wheels (Figure 3.3). The driver allows us to command these motors in velocity or current (torque).

Since the robot is equipped both with front steering wheels and rear actuation, it is overactuated. Depending on the used actuators, we have:

- a passive rear driven walker, when the rear motors are used as brakes only and the front wheels act as caster (i.e., no actuation of the front motors). In this case the vehicle is controlled via differential braking. This configuration is used in the study proposed in Chapter 5;
- a passive front steering walker, when the front motors are used to steer the front wheels and the rear motors are not used to propel the vehicle (they can

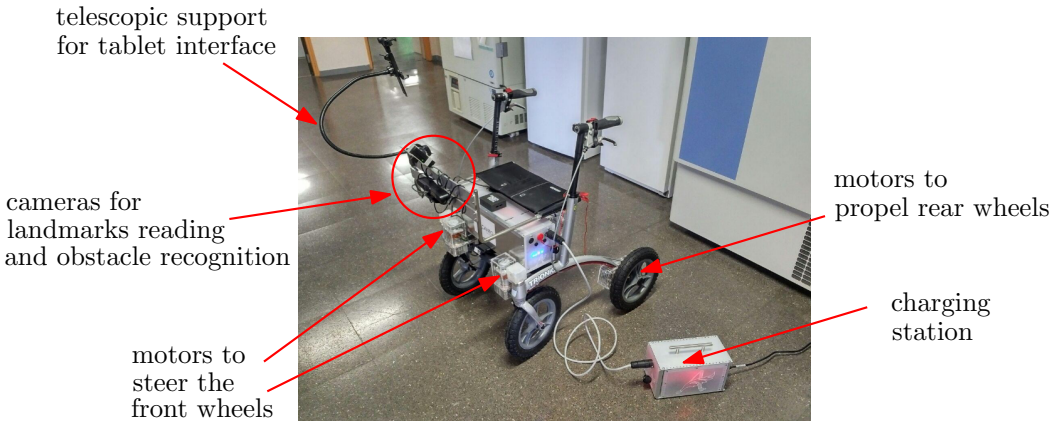

Figure 3.2: ACANTO robotic hardware.

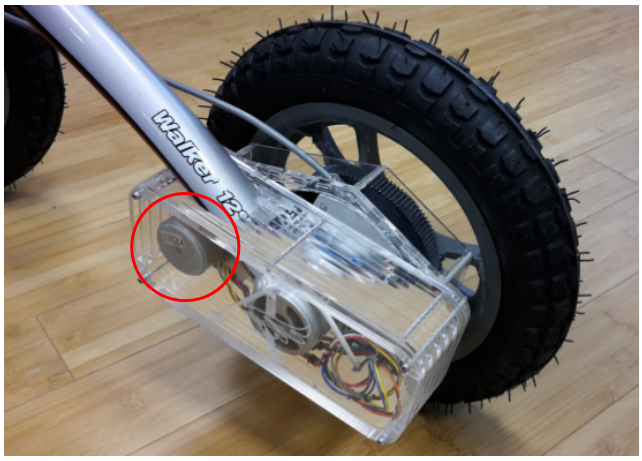

Figure 3.3: Rear wheel of the prototype. The motor is encircled in red.

still be used as brakes for instance in emergency situations). This configuration is used in the study proposed in Chapter 7;

- an active rear driven vehicle, if the rear motor are use to propel the vehicle and the front wheels act as caster. This configuration is used in the study proposed in Chapter 6.

In case of a commercial assistive walker, overactuation is clearly not necessary, however, in this research phase, having a robot with several working possibilities allows us to test several control strategies (e.g., algorithms for front steering vehicle, passive rear driven, active rear driven) on the same platform.

In order to guide the user using haptic feedback, the robot is also Bluetooth connected with two vibrating bracelets that are given to the user (Figure 3.5). From a

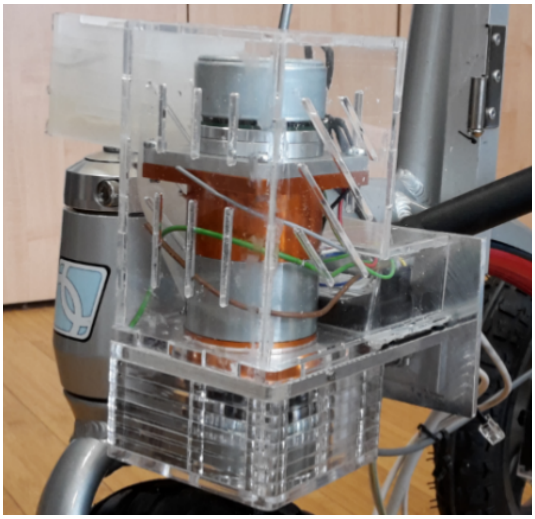

Figure 3.4: Motor mounted on the front right wheel inside the corresponding carter.

technical point of view, the vibrotactile bracelets are composed by cylindrical vibromotors, independently controlled via the Bluetooth communication protocol (see Figure 3.6). The communication is realized with an RN-42 Bluetooth antenna connected to a 3.3V Arduino pro-mini. The wireless connection baud rate is 57600bps. The microcontroller installed on the board is used to independently control the activation of each motor and receiving data from an external platform mounted on the walker. As the user's maximal sensitivity is achieved around 200-300Hz [96] (the human perceptibility range is between 20Hz and 400Hz), two Precision Microdrives Pico Vibe vibration motors are placed into two fabric pockets inside the bracelet (the width of the wristband is about 60mm), with vertically aligned shafts. The motors have a vibration frequency range of 100-300Hz, lag time of about 20ms, rise and stop time of 35ms. The bracelet guarantees about 4 hours of battery life with one motor always turned on. Each bracelet weights about 90g.

Several other sensors are preset on the robot to perform clinical evaluation of the user (for instance to recognize the user posture or the correctness of the user walking). The final version of the robot equipped with these devices for clinical use is supposed to have a higher price than the standard platform (whose final cost should instead range from 2000 to 2500 euros) and hence it will be available mainly to the hospitals instead of private users.

Figure 3.5: Zoom of Figure 1.8 where the vibrating bracelets are highlighted.

Figure 3.6: Picture of the vibrating bracelets. Placement of vibro motors and board (on the left), device available to the user (on the right) [8].

3.2 Vehicle localization

Localizing the vehicle in the environment is a fundamental problem in robot navigation. The localization algorithm aims to provide suitable estimates of the vehicle state. Let us denote by $\hat{a}$ the estimate of the quantity a and by σ_a the corresponding standard deviation. To provide $[\hat{x}, \hat{y}, \hat{\theta}]^\top$, the localization algorithm fuses the available information from the sensors and the vehicle model using an extended Kalman filter. The main sensors used for localization are wheel encoders and a camera reading landmarks.

The rear wheels of the vehicle are equipped with encoders (Figure 3.7), used for odometry. The vehicle state is estimated by discretizing the vehicle dynamics (2.1) via the second order Runge-Kutta method, i.e.,

$$\begin{cases} x_{k+1} &= x_k + v_k T_s \cos(\theta_k + \dfrac{\omega_k T_s}{2}), \\ y_{k+1} &= y_k + v_k T_s \sin(\theta_k + \dfrac{\omega_k T_s}{2}), \\ \theta_{k+1} &= \theta_k + \omega_k T_s, \end{cases}$$

where T_s is the sampling time and the notation x_k denotes that the time-varying quantity x is evaluated at the discrete time instances kT_s, $k \in \mathbb{N}$. The encoders are used to measure the wheel angular rotations $\Delta\phi_R$ and $\Delta\phi_R$ of the right and the left wheel, respectively. During a sampling time interval, the forward displacement of the vehicle Δs and the yaw rotation $\Delta\theta$ are computed as

$$\begin{aligned} \Delta s_k &= \frac{r}{2} \left(\Delta\phi_{R,k} + \Delta\phi_{L,k} \right), \\ \Delta\theta_k &= \frac{r}{b} \left(\Delta\phi_{R,k} - \Delta\phi_{L,k} \right), \end{aligned}$$

where b is the distance between the rear wheels and r is the rear wheel radius. The odometric localization is formulated by substituting $v_k T_s = \Delta s_k$ and $\omega_k T_s = \Delta\theta_k$, hence

$$\begin{cases} x_{k+1} &= x_k + \Delta s_k \cos(\theta_k + \dfrac{\Delta\theta_k}{2}), \\ y_{k+1} &= y_k + \Delta s_k \sin(\theta_k + \dfrac{\Delta\theta_k}{2}), \\ \theta_{k+1} &= \theta_k + \Delta\theta_k. \end{cases}$$

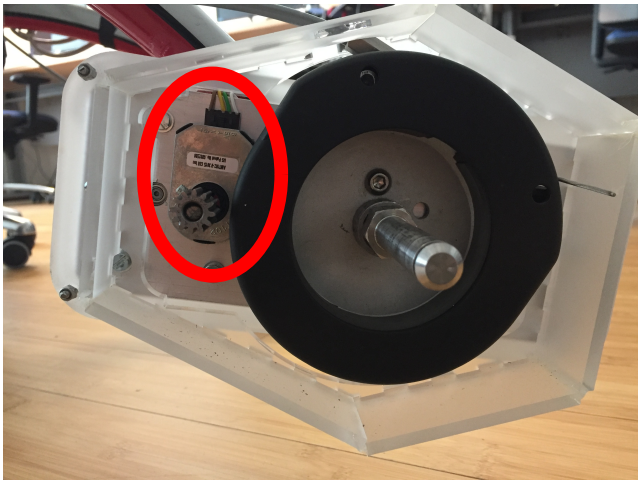

Figure 3.7: Encoder of a rear wheel.

The vehicle is also equipped with a camera reading landmarks placed on the floor, whose coordinates in the map are known. Notice that the measures of vehicle position and attitude obtained by the landmarks are absolute but available only when a landmark is in the field of view of the camera. Conversely, the odometric estimate is incremental (based on a discrete integration), then, since the estimation error is accumulated and grows step after step, the estimation error of the odometric localization is expected to diverge. However, the estimate computed via odometry is always available since the encoder data are read at each time step. Since the two measures (odometry and camera) are fused using a Bayesian observer, the estimator returns minimum variance estimates $\hat{x}, \hat{y}$ and $\hat{\theta}$ of the vehicle state and the corresponding estimation error covariance matrix

$$P = \mathrm{E}\left\{[x-\hat{x}, y-\hat{y}, \theta-\hat{\theta}]^\top [x-\hat{x}, y-\hat{y}, \theta-\hat{\theta}]\right\}, \tag{3.1}$$

where $\mathrm{E}\{\cdot\}$ is the expected value operator.

The localization software is designed to provide a state estimate every 7ms, which is also the sampling time of the control loop.

Figure 3.8: Vehicle and landmark.

3.3 Path planner

The planning algorithm computes a safe path connecting a start and a goal by optimizing the user's comfort. The ACANTO strategy [19] follows the following steps:

1. waypoint generation: a sequence of waypoints, i.e., points the vehicle has to travel through, is generated using a graph-searching algorithm on a graph representation of the environment;

2. fitting with clothoids: the each waypoint is connected with the following using a clothoid, i.e., a curve whose curvature varies linearly with respect to the arch length. In this phase a measure of the user's comfort is optimized;

3. feasibility check: the collision between each clothoid arch and the obstacle is verified.

The procedure returns a path that the robot has to follow to bring the user to the desired location. The path avoids all obstacles that are known a priori in the map (for instance the walls). This planning algorithm is further refined in [20], where a reactive component is implemented. The planner [20] modifies on-line the path generated via [19] to avoid unexpected obstacles that are not present in the map (e.g., other humans), by generating local modifications still considering user's comfort. This reactive planner is very reliable to avoid collisions with pedestrians, since it is based on an accurate model to predict the human motion [32].

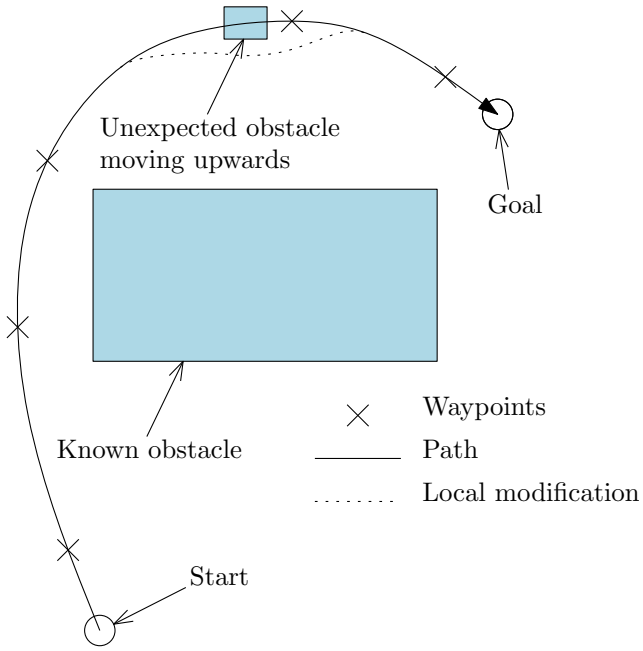

Figure 3.9: Basic idea of the planning strategy.

Chapter 4

A formal approach to authority-sharing

Chapter 2 defines the classical problems (and the corresponding solutions) that a fully autonomous wheeled robot has to solve. This chapter reformulates these problems in order to consider the presence of the human.

4.1 Path following with humans

Whenever the user requires to get to a specific location, the robot plans a path to reach such location and the vehicle has to guide the assisted person towards it. This path following problem differs from the standard formulation presented in Section 2.2 since the robot tracking the path is not fully autonomous but is has to interact with the user.

The robot is an assistive vehicle supporting the user, then the forward velocity v must be established by the assisted person. This way, we avoid uncomfortable situations where the robot pulls the assisted person by moving too rapidly, or where the user is forced by the robot to move slowly. This requirement states that the forward velocity $v(t)$ of the robot cannot be considered as control input. Since this effect is automatically obtained using a passive vehicle (i.e., without thrusting motors [49]), this problem is often defined *passive path following* [5]. Since the vehicle moves at the desired user's speed, we suppose that the user requires to advance, i.e., that the following assumption holds.

Assumption 1 (Persistent motion)**.** *The forward velocity $v(t)$ required by the user satisfies*

$$\lim_{t \to +\infty} v(t) > 0. \tag{4.1}$$

In practice, Assumption 1 is very weak, since it simply states that whenever the user stops, he/she restarts in finite time. This is clearly a necessary condition

to follow the path with a passive vehicle without thrusting motors, since it cannot move if none pushes it. Assumptions on positive forward velocity are very common and used also in unicycle path following [106] and Dubin's car path following [16].

Moreover, the classic path following formulation (2.7) requires zero steady state following errors, which is impossible (and unnecessary) if the user is in the loop. Therefore, to account for the human presence, we propose a relaxed version of the classic path following formulation as follows. Given a path Γ, a nonholonomic vehicle (2.1), and three positive constants x_∞, y_∞, and $\tilde{\theta}_\infty$, find the vehicle angular velocity $\omega(t)$ and a time law $s(t)$ for the curvilinear abscissa, such that

$$\begin{aligned} \lim_{t\to+\infty} |x(t) - x_d(s(t))| &\leq x_\infty, \\ \lim_{t\to+\infty} |y(t) - y_d(s(t))| &\leq y_\infty, \\ \lim_{t\to+\infty} |\theta(t) - \theta_d(s(t))| = |\tilde{\theta}| &\leq \tilde{\theta}_\infty. \end{aligned} \tag{4.2}$$

Conditions (4.2) state that the classical path following requirements (2.7) hold with a tolerated error. This tolerated distance defines a safe proximity of the path in which the control authority can be completely left to the user: if he/she is autonomously maintaining the vehicle in a neighborhood of the (safe) path, i.e., satisfying requirements (4.2), there is no need to actuate the vehicle. Notice that it would impossible for the user to autonomously satisfy the classical path following requirements (2.7) without relaxation, since they require infinite steady state accuracy (which is a low level task impossible for a human agent). The solution of a relaxed path following problem is then a requirement that *both* the user and the robot are capable to solve. Formally, the path following problem has been reformulated from asymptotic stability problem as in (2.7) to ultimate boundedness stability problem in (4.2).

If the path following problem is formulated using a static Frenet frame (2.9), the relaxed version is obtained as

$$\begin{aligned} \lim_{t\to+\infty} |l(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |\tilde{\theta}(t)| &\leq \tilde{\theta}_\infty, \end{aligned} \tag{4.3}$$

where $l_\infty > 0$ and $\tilde{\theta}_\infty > 0$ are the tolerated path following errors. If the path following problem is formulated using a dynamic Frenet frame (2.11), the relaxed version is obtained as

$$\begin{aligned} \lim_{t\to+\infty} |l_x(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |l_y(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |\tilde{\theta}| &\leq \tilde{\theta}_\infty. \end{aligned} \tag{4.4}$$

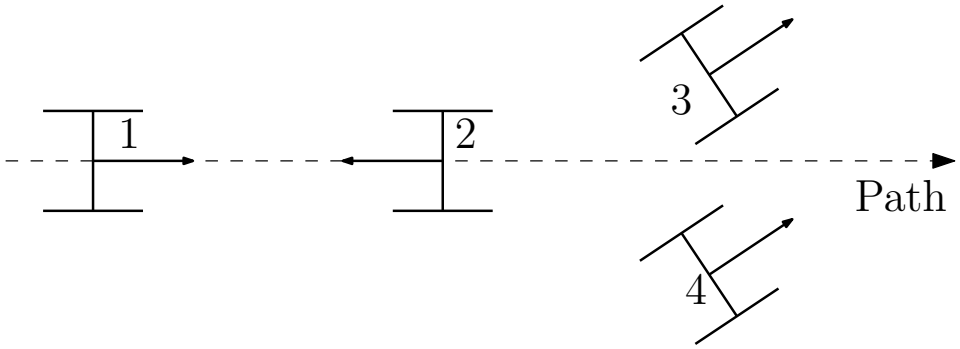

Figure 4.1: Possible relative positions between vehicle and path.

4.2 Attitude error solution

The basic idea to share the control authority is to let the user free to control the vehicle when he/she is autonomously following the path. To this end, a measure of the closeness of the path is required.

4.2.1 Path following metric

The most intuitive measure of the closeness of the path is the geometrical distance from the path (i.e., $|l|$ in (2.9) if a static Frenet frame is used). We propose to adopt a different metric based on the concept of approaching angle δ introduced in Section 2.2.3. In order to have an intuition on why the simple distance from the path is not sufficient, consider Figure 4.1. Both vehicle 1 and vehicle 2 are located exactly on the path, but vehicle 2 is oriented in the opposite direction with respect to the path, then it is definitely not following the path and requires a control intervention. Vehicle 3 and vehicle 4 are located at the same distance from the path, however vehicle 4 is properly oriented to approach the path (hence the control authority can be left to the user), while vehicle 3 requires a control intervention since it is departing from the path.

We propose to measure the path following performance using the attitude error $e_\theta = \tilde{\theta} - \delta$ introduced in Section 2.2.3 (hence the closeness of the path is measured as angular quantity). In the literature [10], it is stated that a wide class of approaching functions δ solves the classical path following problem if $e_\theta = 0$. In this research, we will prove that, if the approaching angle δ has an additional property (i.e., strict monotonicity) we can solve the relaxed path following (which is fundamental for authority-sharing) just by limiting the attitude error e_θ to a nonzero value. This result is based on the interpretation of the angle δ, defined as the desired orientation for the vehicle, as admissible attitude error to get closer to the path. Indeed, we

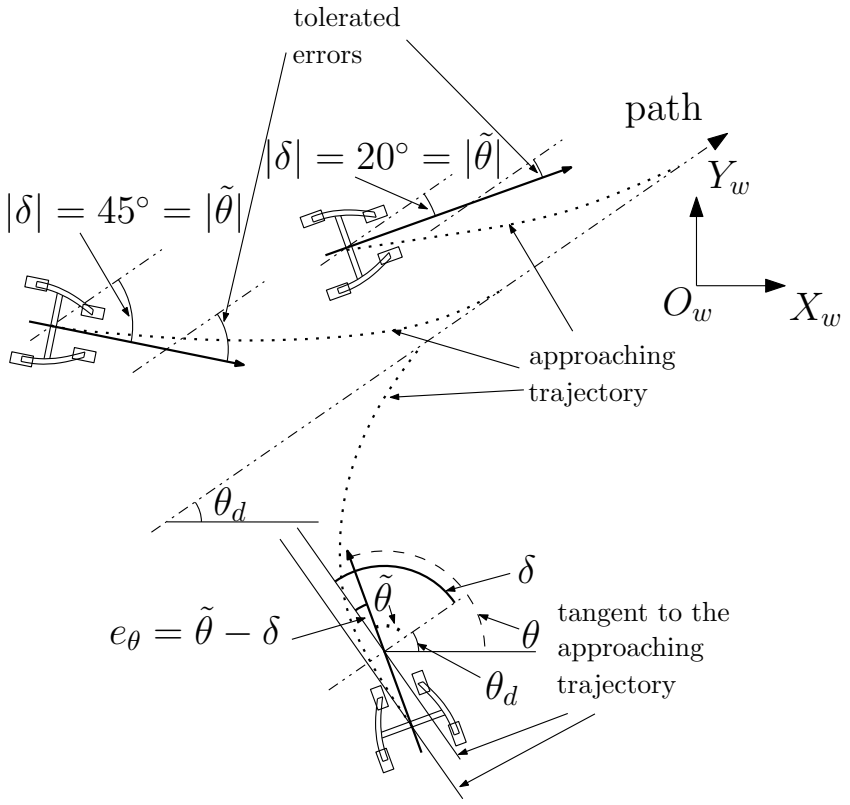

Figure 4.2: Representation of approaching angle as attitude tolerance [13].

notice from Figure 4.2 that the path is approached if

$$|e_\theta| = |\tilde{\theta} - \delta| < |\delta|. \tag{4.5}$$

4.2.2 Solution of relaxed path following

We aim to prove that, to solve the relaxed path following, it is sufficient to limit the attitude error $|e_\theta| = |\tilde{\theta} - \delta|$, provided that δ is strictly monotonic, both with static and dynamic Frenet frame.

Static Frenet frame

If a static Frenet frame is used, the path following is solved without tuning the velocity of the curvilinear abscissa $s(t)$, since the position of the Frenet frame depends on the robot position with respect to the path.

Theorem 2 (Relaxed path following with static Frenet frame)**.** *Consider vehicle* (2.1) *described using a static Frenet frame* (2.8)*. Define the attitude error as* $e_\theta = \tilde{\theta} - \delta(l)$*, where the approaching angle* $\delta(\cdot)$ *is a continuous, odd and strictly monotonic function satisfying* $\delta(l)l < 0$*,* $\delta(0) = 0$*, and* $|\delta(l)| < \frac{\pi}{2}$ *for all* $l \in \mathbb{R}$*. Under Assumption 1, given* $l_\infty > 0$ *and* $\tilde{\theta}_\infty > 0$ *in* (4.3)*, there exists a* $0 < \Theta < \frac{\pi}{2}$ *such that condition*

$$|e_\theta| < \Theta,$$

implies that the relaxed path following problem (4.3) *is solved.*

Intuitively, Theorem 2 states that the relaxed path following problem (4.3) is solved just by limiting the attitude error $|e_\theta|$. The following lemma is used in the proof of Theorem 2, in the following as well.

Lemma 1. *Let* $e_\theta = \tilde{\theta} - \delta$*. A sufficient condition to ensure* $\tilde{\theta}\delta > 0$ *is* (4.5)*.*

Proof of Lemma 1. Assume by contradiction that $\tilde{\theta}\delta \leq 0$. Then two cases are possible: either $\tilde{\theta}\delta < 0$ or $\tilde{\theta}\delta = 0$.

If $\tilde{\theta}\delta = 0$, at least one between $\tilde{\theta}$ and δ is zero. If $\tilde{\theta} = 0$, Relation (4.5) becomes

$$|e_\theta| = |\tilde{\theta} - \delta| = |-\delta| < |\delta|,$$

which is a contradiction. If $\delta = 0$, Relation (4.5) becomes

$$|e_\theta| = |\tilde{\theta} - \delta| = |\tilde{\theta}| < 0,$$

which is impossible.

Since $\tilde{\theta}\delta = 0$ is impossible and the contradictory hypothesis is $\tilde{\theta}\delta \leq 0$, $\tilde{\theta}\delta < 0$ must hold, i.e., $\tilde{\theta}$ and δ have opposite sign. Then two sub-cases are possible: either $\tilde{\theta} > \delta$ or $\tilde{\theta} < \delta$. If $\tilde{\theta} > \delta$, then $\tilde{\theta} > 0$ and $\delta < 0$, hence, Relation (4.5) becomes

$$|e_\theta| = |\tilde{\theta} - \delta| = \tilde{\theta} - \delta < |\delta| = -\delta \implies \tilde{\theta} < 0,$$

which contradicts $\tilde{\theta} > 0$. If $\tilde{\theta} < \delta$, then $\tilde{\theta} < 0$ and $\delta > 0$, hence, Relation (4.5) becomes

$$|e_\theta| = |\tilde{\theta} - \delta| = -\tilde{\theta} + \delta < |\delta| = \delta \implies \tilde{\theta} > 0,$$

which contradicts $\tilde{\theta} < 0$.

Therefore, the contradictory hypothesis $\tilde{\theta}\delta \leq 0$ is rejected, hence the proof. □

Proof of Theorem 2. Consider the Lyapunov function

$$V_l = \frac{1}{2}l^2,$$

whose time derivative is given by

$$\dot{V}_l = l\dot{l} = lv\sin\tilde{\theta}. \tag{4.6}$$

Because of Assumption 1, we have $v > 0$ in finite time. Consider the case where $\tilde{\theta}\delta \neq 0$ (the trivial case $\tilde{\theta}\delta = 0$ is discussed in the following). Notice that, we have $\dot{V}_l < 0$ if $l\tilde{\theta} < 0$. Since the approaching angle function $\delta(\cdot)$ is odd, $l\tilde{\theta} < 0$ is equivalent to $\tilde{\theta}\delta(l) > 0$, i.e. $\tilde{\theta}$ and $\delta(l)$ have the same sign. By assumption

$$|e_\theta| = |\tilde{\theta} - \delta(l)| < \Theta,$$

and by Lemma 1 $\tilde{\theta}$ and $\delta(l)$ have the same sign if

$$|e_\theta| = |\tilde{\theta} - \delta(l)| < |\delta(l)|.$$

This implies that, if $|\delta(l)| > \Theta$, $\tilde{\theta}$ and $\delta(l)$ have the same sign. Because of the strict monotonicity of $\delta(\cdot)$, $|\delta(l)| > \Theta$ holds if $|l| > l_\circ := |\delta^{-1}(\Theta)|$. It is remarked that the inverse function $\delta^{-1}(\cdot)$ exists since $\delta(\cdot)$ is continuous and strictly monotonic. Then $|l| > l_\circ \implies \dot{V}_l < 0$, i.e., $|l|$ converges in finite time and remains inside the region $|l| \leq l_\circ$. Moreover, inside the region $|l| \leq l_\circ$ we have that $|\delta(l)|$ is limited to $|\delta(l_\circ)| = \Theta$ because of monotonicity. Hence, when the solution enters the region $|l| \leq l_\circ$, since $|e_\theta| = |\tilde{\theta} - \delta(l)| \leq \Theta$, we also have $|\tilde{\theta}| \leq \tilde{\theta}_\circ := 2\Theta$. Notice that if $\tilde{\theta}\delta = 0$, which implies $\dot{V}_l = 0$ in (4.6), we have immediately $|l| \leq l_\circ$ and $\tilde{\theta} \leq \tilde{\theta}_\circ$ as trivial case. In fact

$$\begin{aligned} l = 0 &\implies |e_\theta| = |\tilde{\theta} - \delta(l)| = |\tilde{\theta} - \delta(0)| = |\tilde{\theta}| < \Theta < \tilde{\theta}_\circ, \\ \tilde{\theta} = 0 &\implies |e_\theta| = |\tilde{\theta} - \delta(l)| = |\delta(l)| < \Theta \implies |l| < |\delta^{-1}(\Theta)| = l_\circ. \end{aligned}$$

Finally, notice that

$$\lim_{\Theta \to 0} l_\circ = \lim_{\Theta \to 0} \tilde{\theta}_\circ = 0.$$

The solutions always enter and remain inside the regions $|l| \leq l_\circ$ and $|\tilde{\theta}| \leq \tilde{\theta}_\circ$. Then, give two arbitrary constants $l_\infty > 0$ and $\tilde{\theta}_\infty > 0$, the relaxed path following conditions (4.3) can be satisfied by choosing a sufficiently small Θ, hence the proof. □

In [5], the value $l_\circ$ is defined *critical*. In practice, fixed a tolerated attitude error Θ, the vehicle continues to get closer to the path until it is further than $l_\circ$ from the path, i.e., until $|l| > l_\circ$. This concept is shown in Figure 4.2: the closer the vehicle to the path, the smaller the admissible attitude error e_θ (which is mathematically equal to $|\delta(l)|$ by Lemma 1) to ensure that the vehicle gets closer to the path. If the vehicle is very far from the path, the admissible attitude error is larger than Θ (since $\delta(\cdot)$ is strictly monotonic with respect to l), so that the vehicle gets closer. As the vehicle gets closer, the admissible attitude error is reduced while Θ remains constant. This way, the vehicle gets closer to the path until the admissible attitude error reaches the value of Θ, represented by the condition $\Theta = |\delta(l)|$ (hence the definition of the critical value of l as $l_\circ = |\delta^{-1}(\Theta)|$). In the limit case $l = 0$, the admissible attitude error is $|\delta(l)| = |\delta(0)| = 0$, which confirms the intuition that if

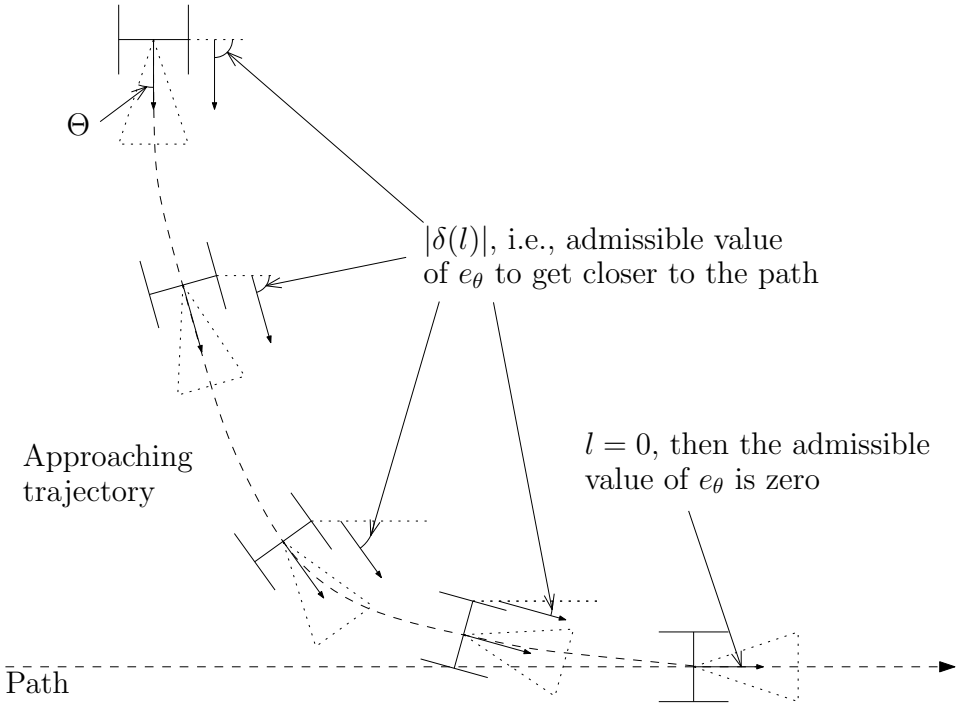

Figure 4.3: The closer the vehicle to the path, the smaller the admissible value of e_θ to get closer to the path.

the vehicle is located on the path but it is not properly oriented, it will necessarily depart from it. A further representation is given in Figure 4.4: the vehicle on the left has $|e_\theta| < \Theta$ and it is approaching the path since $|\delta(l)| > \Theta$; the vehicle on the right still has $|e_\theta| < \Theta$ but it is departing from the path since $|\delta(l)| < \Theta$.

Dynamic Frenet frame

If the Frenet frame is computed dynamically, it is still possible to solve the relaxed path following problem by properly choosing the moving velocity of the Frenet frame $\dot{s}$.

Theorem 3 (Relaxed path following with static Frenet frame)**.** *Consider vehicle* (2.1) *described using a dynamic Frenet frame* (2.10)*. Define the attitude error as* $e_\theta = \tilde{\theta} - \delta(l_y)$*, where the approaching angle* $\delta(\cdot)$ *is a continuous, odd and strictly monotonic function satisfying* $\delta(l_y)l_y < 0$*,* $\delta(0) = 0$*, and* $|\delta(l_y)| < \frac{\pi}{2}$ *for all* $l_y \in \mathbb{R}$*. Choose the moving velocity of the Frenet frame as*

$$\begin{aligned} \dot{s} &= v\dot{\xi}, \\ \dot{\xi} &= \cos\tilde{\theta} + \kappa_x l_x, \end{aligned} \tag{4.7}$$

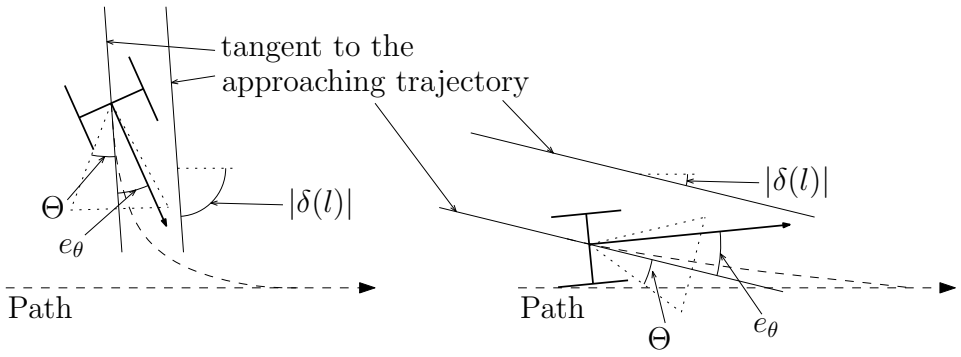

Figure 4.4: Effect of the admissible attitude error to get closer to the path.

where $\kappa_x > 0$ *is a gain. Under Assumption 1, given* $l_\infty > 0$ *and* $\tilde{\theta}_\infty > 0$ *in* (4.4), *there exists a* $0 < \Theta < \frac{\pi}{2}$ *such that condition*

$$|e_\theta| < \Theta,$$

implies that the relaxed path following problem (4.4) *is solved.*

Proof of Theorem 3. Consider the Lyapunov function

$$V_{xy} = \frac{1}{2} l_x^2 + \frac{1}{2} l_y^2,$$

whose time derivative along the solutions is

$$\begin{aligned} \dot{V}_{xy} &= l_x \left(-\dot{s}(1 - c(s) l_y) + v \cos\tilde{\theta} \right) + \\ &\quad + l_y \left(-c(s)\, \dot{s}\, l_x + v \sin\tilde{\theta} \right) \\ &= l_x \left(-\dot{s} + v \cos\tilde{\theta} \right) + l_y v \sin\tilde{\theta}. \end{aligned}$$

Using the expression of $\dot{s}$ from (4.7), we get

$$\dot{V}_{xy} = v \left(-\kappa_x\, l_x^2 + l_y \sin\tilde{\theta} \right).$$

The term $-\kappa_x\, l_x^2 < 0$ always has a stabilizing effect since $v > 0$ holds in finite time by Assumption 1. Notice that condition $l_y \sin\tilde{\theta} < 0$ implies $\dot{V}_{xy} < 0$. The expression $l_y \sin\tilde{\theta}$ is equivalent to equation (4.6), then, by following exactly the same steps of Proof of Theorem 2, we get that the solution enters in finite time and remains inside the regions $|l_y| \leq l_y^\circ := |\delta^{-1}(\Theta)|$ and $|\tilde{\theta}| \leq \tilde{\theta}_\circ := 2\Theta$. Moreover, in the region $|l_y| \leq l_y^\circ$, the derivative of the Lyapunov function satisfies

$$\dot{V}_{xy} = v \left(-\kappa_x\, l_x^2 + l_y \sin\tilde{\theta} \right) \leq v \left(-\kappa_x\, l_x^2 + l_y^\circ \right),$$

i.e.,

$$\dot{V}_{xy} < 0, \quad \forall l_x > \sqrt{\frac{l_y^\circ}{\kappa_x}} = \sqrt{\frac{|\delta^{-1}(\Theta)|}{\kappa_x}} =: l_x^\circ,$$

which implies that l_x enters in finite time and remains inside the region $|l_x| \leq l_x^\circ$. Finally, notice that

$$\lim_{\Theta \to 0} l_x^\circ = \lim_{\Theta \to 0} l_y^\circ = \lim_{\Theta \to 0} \tilde{\theta}_\circ = 0.$$

The solutions always enter and remain inside the regions $|l_x| \leq l_x^\circ$, $|l_y| \leq l_y^\circ$, and $|\tilde{\theta}| \leq \tilde{\theta}_\circ$. Then, give two arbitrary constants $l_\infty > 0$ and $\tilde{\theta}_\infty > 0$, the relaxed path following conditions (4.4) can be satisfied by choosing a sufficiently small Θ, hence the proof. □

4.3 Attitude-based authority-sharing

In Section 4.2, we prove that the relaxed path following problem can be solved just by limiting the attitude error $e_\theta = \tilde{\theta} - \delta$ since it is a measure of the distance between vehicle and path. With reference to the classification proposed in Section 2.3, we aim to design authority-sharing metric-based strategies, using the attitude error e_θ as metric. Since the planned path is always obstacle-free, we assume that if the relaxed path following conditions (4.3) or (4.4) hold (i.e., the attitude error $|e_\theta|$ is small), the user is safe (since he/she is properly following a safe path) then he/she can be left free to control the vehicle. Conversely, if the path following conditions do not hold (hence $|e_\theta|$ is large), the vehicle has to have the majority of the control authority to steer the user in the safe neighborhood of the path. Clearly, in the practice, the tolerated value of $|e_\theta|$, defining the tolerated path following error, is application-dependent. If, for instance, we want that the user remains very close to the path to avoid collisions with obstacles, the tolerated value of $|e_\theta|$ has to be small.

In this research, we design both sharp and gradual allocation of the control authority to maintain $|e_\theta|$ small.

A different and more classical path following metric is, for instance, the Lyapunov function $\overline{V} = \frac{1}{2}(l_x^2 + l_y^2) + \frac{1}{2\kappa_3}(\tilde{\theta} - \delta)^2$ in (2.13), where the constant $\kappa_3 > 0$ weights the relative importance between the attitude error $\tilde{\theta} - \delta$ and the distance error $l_x^2 + l_y^2$. However, the choice of $|e_\theta|$, with $\delta(\cdot)$ strictly monotonic, offers two fundamental advantages for authority-sharing. First, $|e_\theta|$ defines the distance in terms of angles, while $\overline{V}$ does not have an immediate physical meaning. As a consequence, metric $|e_\theta|$ is much more intuitive. Second, metric $|e_\theta|$ properly captures the correct behavior of the user, hence it is much more appealing for authority-sharing: consider, for instance, the case where the vehicle is far from the path in terms of linear distance but the user is properly orienting the heading towards the path, i.e., he/she is

properly behaving. We have $|e_\theta|$ small and $\overline{V}$ large. Therefore, the attitude error $|e_\theta|$ identifies also the correctness of the user's behavior.

4.3.1 Example of sharp allocation

In this research, authority-sharing algorithms based on sharp allocation are presented. We say that the vehicle works in the *robot in control mode* when the control authority is given to the robot, i.e., the motion of the vehicle is governed by the actuation system, and that the vehicle works in the *user in control mode* when the control authority is given to the user, i.e., the motion of the vehicle is governed by the user's actions (for instance the actuators are not active and the vehicle is passive).

The control law is then designed in order to maintain $|e_\theta|$ small and the switching (i.e., the transfer of the control authority from user to robot or vice versa) takes place with hysteresis. The hysteresis is defined by two attitude thresholds $\theta_{q_2} > \theta_{q_1} > 0$. The mechanism is illustrated in Figure 4.5. If initially the attitude error satisfies $|e_\theta| \geq \theta_{q_2}$ (point 1 in Figure 4.5), the control authority is given to the robot, and a control action is applied. Then, when the controller has reduced $|e_\theta|$ under the threshold θ_{q_1}(point 2 in Figure 4.5), the control authority is given back to the user. If the user wrongly uses his/her control authority, the attitude error can increase. If $|e_\theta|$ exceeds again the threshold θ_{q_1} (point 3 in Figure 4.5) the user in control mode is maintained. The robot in control mode is reactivated only if the attitude error hits the upper hysteresis threshold (point 4 in Figure 4.5).

To mathematical represent the hysteresis mechanism, we use hybrid system theory [40], by implementing the hybrid dynamics of a logic variable $q \in \{0, 1\}$. Intuitively, a hybrid system is a dynamical system combining a continuous and a discrete dynamics. The robot in control mode is defined by $q = 1$, while the user in control mode is defined by $q = 0$ (see Figure 5.4). The hybrid dynamics of q is defined as

$$\begin{cases} \dot{q} & = 0, \qquad [e_\theta, q]^T \in \mathcal{C}, \\ q^+ & = 1 - q, \quad [e_\theta, q]^T \in \mathcal{D}, \end{cases} \tag{4.8}$$

where $[e_\theta, q]^T$ is the overall state of the hybrid system, $\mathcal{C} := \mathcal{C}_0 \cup \mathcal{C}_1$ and $\mathcal{D} := \mathcal{D}_0 \cup \mathcal{D}_1$ are the flow and the jump set respectively, where

$$\begin{aligned} \mathcal{C}_0 &= \{|e_\theta| \leq \theta_{q_2} \ \wedge \ q = 0\}, \quad \mathcal{C}_1 = \{|e_\theta| \geq \theta_{q_1} \ \wedge \ q = 1\}, \\ \mathcal{D}_0 &= \{|e_\theta| \geq \theta_{q_2} \ \wedge \ q = 0\}, \quad \mathcal{D}_1 = \{|e_\theta| \leq \theta_{q_1} \ \wedge \ q = 1\}, \end{aligned} \tag{4.9}$$

where $\theta_{q_2} > \theta_{q_1} > 0$ are the hysteresis thresholds. Notice that variable q has discrete dynamics only, and its jumps take place with hysteresis. The hybrid notation (4.8) and (4.9) from [40] will be used in the thesis. In [40], a solution of the hybrid system is formally defined on a hybrid time domain $E \subset \mathbb{R}_{\geq 0} \times \mathbb{Z}_{\geq 0}$ where, for

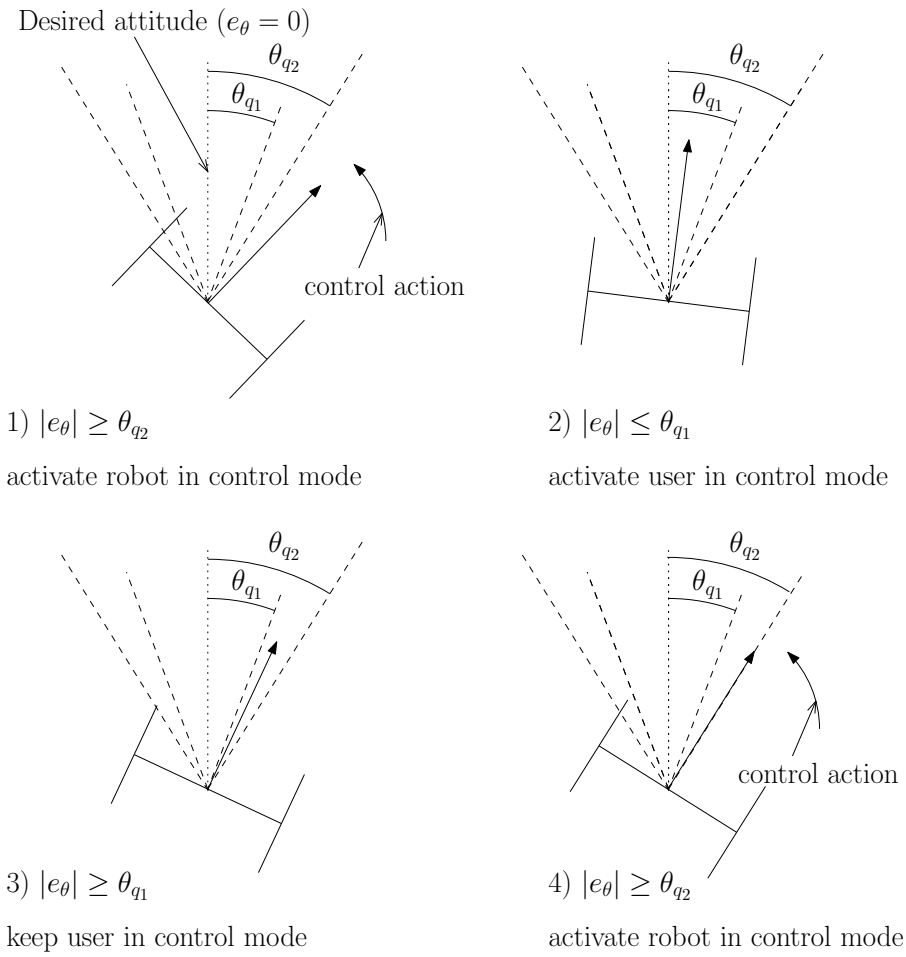

Figure 4.5: Example of sharp allocation of authority-sharing based on the attitude error.

each $(t, j) \in E$, t measures the amount of elapsed ordinary time and j measures the amount of jumps already performed by the solution. However, the formal stability study is just marginally faced in the thesis for simplicity's sake, in favor of a more intuitive analysis.

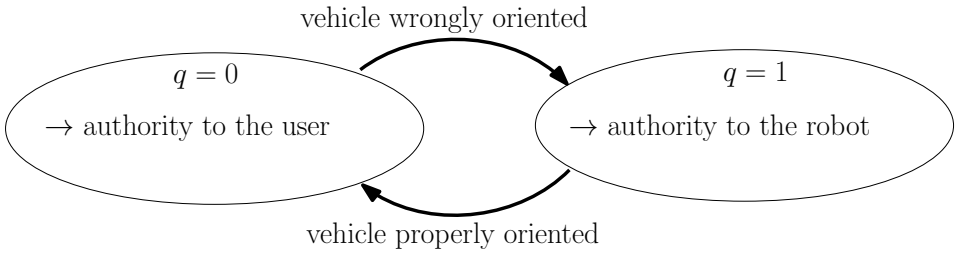

Figure 4.6: Hybrid labeling of the control modes.

4.4 Further considerations on the attitude error

The relaxed path following problem is solved by acting on the attitude error $e_\theta = \tilde{\theta} - \delta$. To do so, the additional property of strict monotonicity has been added (with respect to the literature) to the approaching angle function $\delta(\cdot)$. In the practice this requirement is not a relevant limitation, since non-monotonic approaching angles offer convergence to the path with unsatisfactory performance [10]. One of the most used approaching angle function in this research is

$$\delta(l_y) = -\frac{\pi}{2} \tanh\left(\frac{l_y}{2}\right), \tag{4.10}$$

which satisfies the requirements of the proposed theorems.

It is also remarked that the attitude error $e_\theta = \tilde{\theta} - \delta$ is defined as a difference between two angles and hence it has to be computed as in Section 2.2.5.

Chapter 5

Haptic-braking guidance

This chapter proposes a low cost authority-sharing path following strategy combining the advantages of a simple mechanical braking guidance, such as safety, passivity, and cheapness, and the ones of a vibrotactile haptic guidance, such as comfort and portability. The user is guided by providing indications on the directions of motion using the haptic interface so that he/she can autonomously and comfortably follow the planned path. However, whenever the user significantly departs from the path (for instance he/she gets too close to obstacles), the braking system kicks in to safely steer the user back along the proper direction. The formal correctnesses of the hybrid strategy ruling the combination of the two guidance systems is proved theoretically. Moreover, an extensive experimental study with users aged 75 to 100, comprising also psychological evaluations, has been performed. The hybrid combination of the braking and the haptic guidance systems is shown to outperform the two individual approaches in isolation. In fact, whilst it retains the same level of the users' perceived comfort typical of the haptic-only guidance, it ensures an adequate path following performance by means of the braking-only guidance.

5.1 Introduction to braking and haptic guidance

To follow a planned path, the robot requires an actuation system capable of modifying its motion. Clearly, one of the key requirements is to simplify the mechanical design in order to make the device cost affordable to a large number of potential customers. A possible way to achieve the goal is to utilize electro-mechanical brakes as actuation system. This solution has several advantages. First, the brakes are a dual use device. They can be used to stop the walker in case of emergency or when the user wants to use the seat. But, they can also be used to guide the walker by applying different torques on the two rear wheels (differential braking). Second, the absence of traction on the wheels makes the vehicle a passive robot, i.e., a robot that needs to be pushed by the user to move. Passive robots are arguably safer than their "active" counterparts, since they cannot generate accidental motions that

could cause injuries to the user or to other people in the surroundings.

Examples of passive robots are the ones developed around the COllaborative RObot (COBOT) paradigm [94, 39], a particular subclass of assistive robots [111]. A COBOTs is robotic systems composed of a cane with a caster wheel mounted on the tip. The wheel is equipped with a servo motor to control the steering angle. The passive robot idea has been extended and applied to a number of intelligent walking aids, ranging from steering-only controlled walkers [53] to fully actuated assistive carts [95]. More directly connected to the work presentedin this chapter is the passive walker proposed in [49] and further developed in [97]. In the work of these authors, the robotic device is a standard walker (pretty much like the *FriWalk*) with two caster wheels and a pair of electromagnetic brakes mounted on fixed rear wheels, whose braking action is modulated to steer the vehicle towards the desired path. A similar idea can be found in [33, 34], where braking system intervention is minimized to enhance the user's comfort. A common claim of all these papers is an excellent accuracy in following the path, which requires a robot always in control of the motion, denying the possibility of sharing the control authority with the assisted person, hence limiting the sense of independence of the user. Common limitation is also that they all require expensive devices for sensing (human applied forces and/or braking torques) and a precise actuation mechanisms, which can effectively modulate the braking torque.

In this work, we advocate a different philosophy, which can be condensed in the following statement: *the guidance system can rely on the user's abilities to navigate autonomously and needs to intervene only when the walker is not properly oriented to approach the desired path.* In other words, we are aiming at sharing the control authority with the user, without using complex systems to interpret the user's behavior.

Once the controller has intervened with a quick correction, the user could re-gain control of the motion. Translating this philosophy into a control algorithm requires different methods from those used by the authors cited above, who aim at a fine grained control of the position of the user on the path. A useful inspiration come form [16, 109]. In these cases the authors propose an all–or–nothing (bang-bang) control action, where to steer left or right, the controller has to block, respectively and alternatively, the left or right wheel.

Such ideas cannot be directly used in a service robot like the one of this work, since they generate a "chattering" behavior, which is annoying and difficult to interpret for the user, to guarantee asymptotic tracking of the path with zero error. To overcome these problems, we share the control authority with the assisted person in a neighborhood of the path to solve a relaxed path following problem: this way, the tracking is still guarantee (with nonzero error), but the chattering issue is totally removed and then the strategy can be implemented in a real prototype like the *FriWalk*.

Overall, this authority-sharing approach allows us to implement this bang-bang

braking strategy on the real prototype, while maintaining the aforementioned advantages:

1. simple sensing system: the forces applied by the assisted person are not required. Moreover, to block a wheel, it is not needed to modulate the braking torque since it is sufficient to command the maximum available current. This hardware limitation is present in the cheap assistive walker in Figure 3.1;

2. safety: the vehicle is totally passive, then the user is never harmed by accidental motion;

3. reduced cost of the actuators: an automated intelligent braking system is always a required feature (we could say the bare minimum) to guarantee safety in a robotic walker. The cost is reduced via a dual use of the passive braking system since it is also employed for guidance;

4. back-up solution: even if a more advance guidance system is available (e.g., front steering wheels), the bakes can be used as last resort for guidance if the the main guidance is out of work. In fact, the brakes are in general more reliable than other actuators because of safety issues.

The main disadvantage in using a bang-bang braking policy is the reduced comfort coming from curvature discontinuity [14] generated by rotations around a blocked wheel.

In principle, if the user is cooperative, i.e., he/she uses his/her control authority to follow the path, the destination can be reached without using the brakes, then improving the user's comfort since no braking actions occur. To suggest the user how to behave, i.e., where to steer the vehicle when he/she is on control of the motion, the haptic system is used. In other words, if the user follows the instructions provided by the haptic system, he/she is guided towards the destination being always in control of the vehicle. As a consequence, the braking system can be activated only for distracted users (which tend to depart from the planned path, for instance in direction of dangerous obstacles), an then the discomfort generated by the curvature discontinuity (rotations around a blocked wheel) is introduced only if the user is in danger.

To provide the user the information he/she needs to follow the path using an haptic interface, visual and auditory channels may be used. However, in real world scenarios, these channels may be overloaded with a huge quantity of information, resulting in some cases, unusable. A following consequence is the rapid error increasing and the overall user's performance reduction if cues are provided through these channels. A possible solution is to deliver necessary information exploiting an underutilized sense, i.e., the sense of touch. As the sound, a tactile stimulus is made up of a signal with varying frequency and amplitude, but different from the auditory feedback, tactile sensation directly engages our motor learning system with

extraordinary sensitivity and speed [47, 68]. Moreover, tactile communication can be used in situations where visual or auditory stimuli are distracting, impractical, unavailable or unsafe.

Vibrotactile haptic guidance has been successfully exploited in the last years. Closely related are the researches presented in [116, 29, 69, 99, 60, 15], and [18]. In [116], a vibrotactile belt is used for waypoint navigation in an outdoor environment. Another belt was exploited in [29] where the authors presented a navigation guidance system that guides a human towards a goal point. A torso-mounted vibrotactile display in [69] to provides cues for improving the situational awareness of soldiers in a simulated building-clearing exercise. Scheggi et al. in [99] presented a new paradigm for the assisted navigation of mixed human-robot teams using haptic information. Moreover, in [60], the authors proposed a mobile device for human navigation using multimodal communication (audio, visual, vibrotactile and directional skin-stretch stimuli). Finally, Benallegue et al. in [18] present an innovative head-neck system to balance the human steady gait trajectory. It exploits the tilt estimation within the visuo-vestibular system and contributes to the dynamics of walking due to the head stabilization.

While kinesthetic feedback is common in haptic systems, we tested vibrotactile interfaces, since tactile devices are generally more portable, less encumbering and have a wider range of action than the kinesthetic ones [70]. Different from existing strategies, in our case the user has the hands free from the devices, thus he/she can hold the handles of the walker.

The possibility of guiding humans using haptic interfaces were tested. Elastic bands were tested considering both their capability in displaying cues and investigating the after-effect problem. The idea of the preparatory experiments was to evaluate whether the vibro interface were able to suggest direction of motion.

Tactile vibratory sensitivity is influenced by the spatial location on the body, the distance between the stimulators, the frequency of stimulation and the age of the user. Studies have demonstrated that vibration is better sensed on hairy skin due to its thickness and nerve depth, and that vibrotactile stimuli are best detected in bony areas [37]. In particular, wrists and spine are generally preferred for detecting vibrations, with arms and ankles next in line [56]. Due to the aforementioned considerations and since our aim is to design an intuitive and non-obtrusive device which could be easily worn, we concentrated on the development of vibrotactile bracelets. We decided to use the bilateral configuration, that required two bracelets, one for each arm [100]. We investigated a solution in which two haptic bracelets, equipped with vibrating motors, are used.

Our strategy to solve the overall problem is then based on a hybrid controller, which is formalized and analyzed in this chapter. Its application allows a controlled blend of haptic and mechanical actions, thus reducing the robot control authority and guaranteeing the performance. The controller is based on the definition of safety regions around the desired path, where corrective actions simply do not occur.

The proposed hybrid controller has been successfully applied to a real walker and its effectiveness has been proved through extensive experiments with older adults. We compared the haptic and mechanical solutions in isolation with the proposed approach to highlight the benefits, in terms of user's comfort and safety, of the orchestrated fusion of the two approaches.

With reference to Section 2.3, it is remarked that the braking guidance is a metric-based sharp allocation approach, while the haptic guidance is a communication-based with user in control approach.

5.2 Guidance framework

The assistive walker is modeled as a unicycle-like robot having equations (2.1)

$$\begin{cases} \dot{x} &= v\cos\theta, \\ \dot{y} &= v\sin\theta, \\ \dot{\theta} &= \omega. \end{cases}$$

We denote by $R = b/2$ the length of the rear semi-axle (see Figure 2.1).

Since the robot is propelled by the user, the forward velocity $v > 0$ is completely determined by the assisted person, hence it is not a control variable. The angular velocity ω is, instead, the control input that should be generated by the actuators. The *FriWalk* is considered equipped with two front caster wheels (the front steering motors are not used in this study), while the rear wheels are equipped with electromagnetic brakes capable of blocking the wheels only, i.e., no modulation of the braking action is possible. This is an actual hardware limitation of the cheap assistive walker in Figure 3.1, which has been used in conjunction with the one in Figure 3.2 for the experimental validation of the work presented in this chapter[1]. This way, the vehicle rotates around the blocked wheel with angular velocity $|\omega| = v/R$, while it moves forward if the wheels are free to rotate, i.e., $\omega = 0$. This means that the control input ω generated by the braking system can assume only three values

$$\omega \in \left\{-\frac{v}{R}, 0, \frac{v}{R}\right\}. \tag{5.1}$$

Notice that when $\omega = -\frac{v}{R}$ and $\omega = +\frac{v}{R}$ the vehicle turns right and left, respectively, with fixed curvature radius R, while when $\omega = 0$ the vehicle can move freely.

It is remarked that Equation (5.1) describes the vehicle behavior generated by the braking system. When the brakes are not active the vehicle is totally passive (as a standard rollator), then the user is capable to steer it. Using the vibrating bracelets, it is possible to adjust the heading by providing haptic stimuli. To limit

[1]The modulation of the braking torque would be possible with the walker in Figure 3.2 only. However, this hardware feature is not exploited in the control design of this chapter, where a limited hardware as the one in Figure 3.1 is considered.

the use of the tactile channel and the recognition time, the haptic system is used to provide the user with three simple indications, corresponding to basic behaviours of the human locomotion [100]:

1. *turn left*: the user is suggested to turn left, i.e., to apply an angular velocity $\omega > 0$, by activating the left bracelet;

2. *turn right*: the user is suggested to turn right, i.e., to apply an angular velocity $\omega < 0$, by activating the right bracelet;

3. *go straight*: no haptic stimuli are provided to the user, who is then is suggested to go straight, i.e., to apply an angular velocity $\omega \approx 0$.

To properly represent the path following problem, we adopt a dynamic Frenet frame (2.10)

$$\begin{cases} \dot{l}_x &= -\dot{s}(1 - c(s)l_y) + v\cos\tilde{\theta}, \\ \dot{l}_y &= -c(s)\,\dot{s}\,l_x + v\sin\tilde{\theta}, \\ \dot{\tilde{\theta}} &= \omega - c(s)\dot{s}. \end{cases}$$

We need to ensure that the vehicle approaches and follows a given path by considering the actuator limitations (recall that the brakes are capable of only blocking the wheels). In particular, we need to compute the angular velocity control law ω satisfying the actuation constraints (5.1) and the velocity of the Frenet frame $\dot{s}$ in (2.10) ensuring (4.4), i.e.,

$$\begin{aligned} \lim_{t\to+\infty} |l_x(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |l_y(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |\tilde{\theta}| &\leq \tilde{\theta}_\infty. \end{aligned}$$

Notice that this path following problem is *passive* and constrained since we are using passive actuators (i.e., brakes) without modulations (i.e., either blocking or not blocking the wheels) and the forward velocity v is determined by the assisted person. Notice also that the bracelets can not be considered actuators in the control design, since, even when they vibrate, the behavior of the vehicle is still determined by the user.

5.3 Sharp authority allocation in the braking guidance

Possible theoretical solutions to the classical path following problem (conditions (2.9)) are reported for example in [16, 109, 10]. In [16, 109] the input angular velocity $\omega \in \{-\frac{v}{R}, 0, \frac{v}{R}\}$ is selected on the basis of the actual path coordinates l, $\tilde{\theta}$. In particular, the state space $(l, \tilde{\theta})$ is partitioned into a set of non-overlapping regions, each

one corresponding to a value of the input ω in the set $\{-\frac{v}{R}, 0, \frac{v}{R}\}$. This solution has been generalized in [10] introducing the approaching angle δ, which defines the way in which the path is approached.

To limit the number of braking actions, the steering angle δ should have a varying rate smaller than the maximum angular velocity of the vehicle [10], i.e.,

$$|\dot{\delta}(l)| \leq \frac{v}{R}.$$

As a final remark, the almost global asymptotic stability property mentioned in [16, 109, 10] refers to the possible singular points when $\dot{s}$ is undefined, i.e., $c(s)l = 1$, since a static Frenet frame is used.

Although the solutions proposed in [16, 109, 10] are of relevance, they are unavoidably based on the chattering of the braking actions, i.e., the controller requires infinite switches between the control action $\{-\frac{v}{R}, 0, +\frac{v}{R}\}$ to accurately follow a generic feasible path. This is completely out of the question whenever the controller is applied to an actual human-robot mechanical interface. Indeed, besides the practical limits of generating a number of infinite switches in finite time because of the brakes limited dynamics, the human perceives a system that persistently brakes and releases the rear wheels, thus generating a sort of mechanical Pulse Width Modulation, which generates discomfort.

The effect of chattering is shown in the simulation of Figures 5.1 and 5.2. Notice from Figure 5.1 that the path is followed with zero steady state error. Figure 5.2 reports the sign of the angular velocity $\omega \in \{-\frac{v}{R}, 0, \frac{v}{R}\}$ applied to follow the path as in Figure 5.1. Notice that the sign continuously switches from -1 to $+1$ in a time interval equal to the simulation time step. This chattering actuation cannot be obtained with a real actuation system and highly penalizes the user's comfort.

To overcome these issues, the user is provided with control authority and charged to handle the vehicle in a neighborhood of the path using an authority-sharing approach based on sharp allocation. A hysteresis-based behavior, which limits the number of braking system interventions along the trajectory, is defined to the point. This way, the controller we are aiming at provides "most of the time" the user with complete control of the motion, hence the need of solving a relaxed version of the path following problem (4.4).

5.3.1 Chattering-free control algorithm

The hysteresis mechanism avoiding chattering is defined on the orientation error $e_\theta = \tilde{\theta} - \delta(l_y)$, which is used as metric to define the distance from the path. Let $g(e_\theta, \epsilon_q) : \mathbb{R} \times \mathbb{R}_{\geq 0} \mapsto \{-1, 0, 1\}$ defined as the function

$$g(e_\theta, \epsilon_q) := \begin{cases} \text{sign}(e_\theta), & \text{if } |e_\theta| \geq \epsilon_q, \\ 0, & \text{if } |e_\theta| < \epsilon_q. \end{cases} \tag{5.2}$$

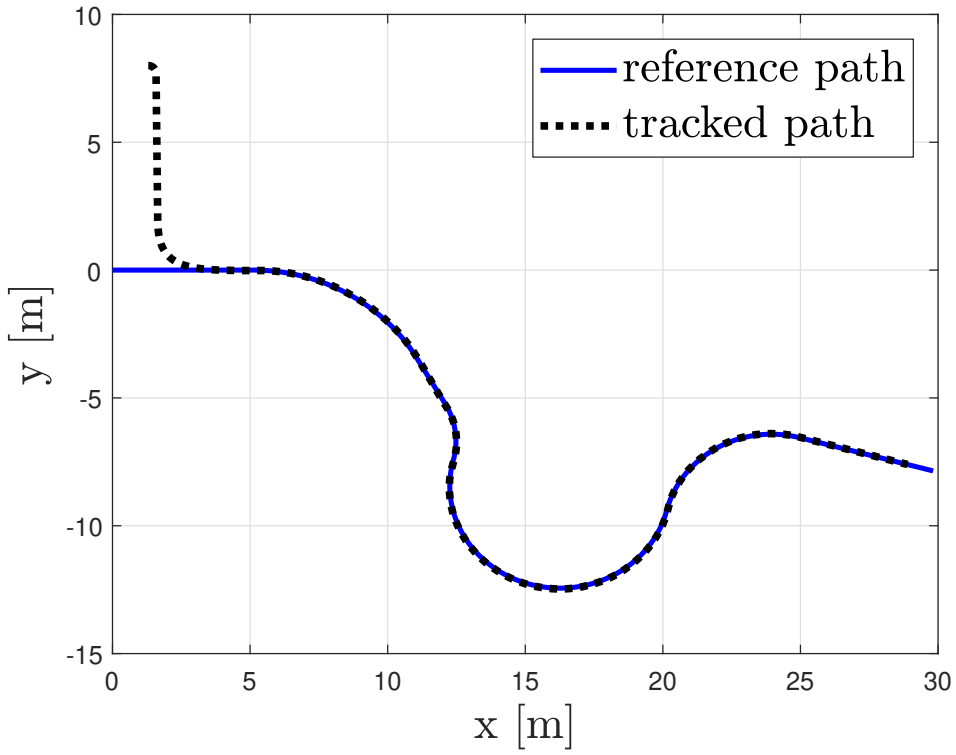

Figure 5.1: Path following with zero steady state error.

In practice, function $g(e_\theta, \epsilon_q)$ is the sign function of e_θ with a dead zone defined by $\epsilon_q > 0$ (see Figure 5.3). The hybrid controller is then defined as follows:

$$
\begin{cases}
\begin{cases}
\dot{x} &= v\cos\theta, \\
\dot{y} &= v\sin\theta, \\
\dot{\theta} &= \omega, \\
\dot{q} &= 0, \\
\dot{\omega} &= 0, \\
\dot{s} &= v(\cos(\tilde{\theta}) + \kappa_x l_x),
\end{cases} & \chi \in \mathcal{C}, \\
\begin{cases}
x^+ &= x, \\
y^+ &= y, \\
\theta^+ &= \theta, \\
q^+ &= 1 - q, \\
\omega^+ &= -g(e_\theta, \epsilon_q)\dfrac{v}{R}, \\
s^+ &= s,
\end{cases} & \chi \in \mathcal{D},
\end{cases}
\tag{5.3}
$$

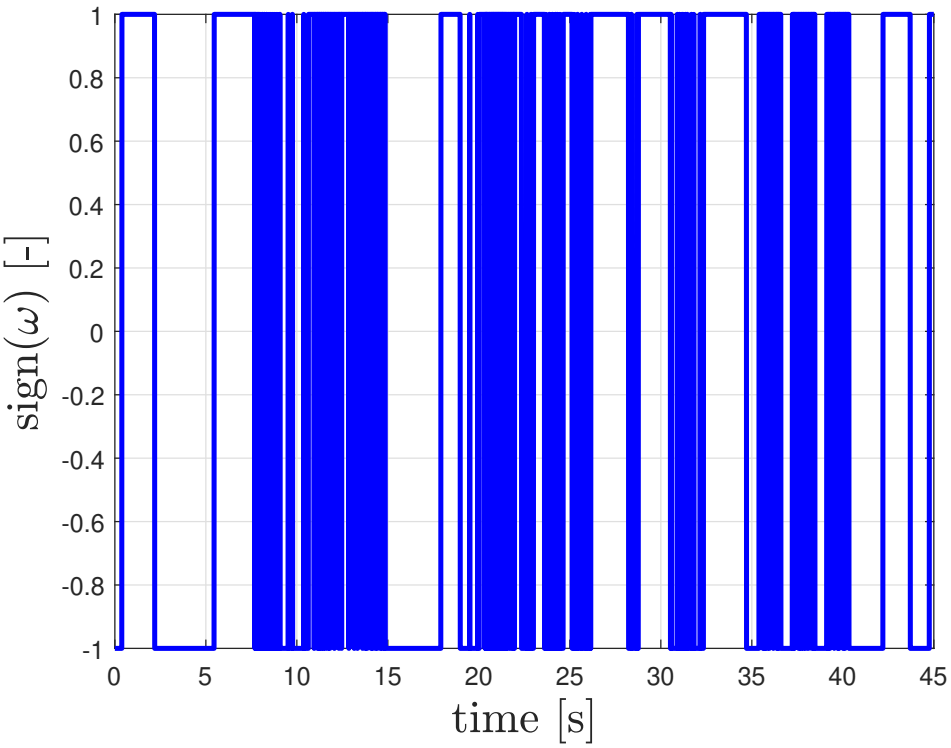

Figure 5.2: Chattering for control action in Figure 5.1.

where $\chi = [x, y, \theta, q, \omega, s]^\top$ is the overall hybrid state defined for convenience of notation and $\kappa_x > 0$ is a tunable gain. Notice that the velocity of the Frenet frame $\dot{s}$ is defined as in (4.7) since a dynamic Frenet frame is used[2]. The flow set is $\mathcal{C} = \mathcal{C}_0 \cup \mathcal{C}_1$ and the jump set is $\mathcal{D} = \mathcal{D}_0 \cup \mathcal{D}_1$, where

$$\begin{aligned} \mathcal{C}_0 &= \left\{\chi \in \mathbb{R}^6, |e_\theta| \leq \theta_{q_2} \ \wedge \ q = 0\right\}, \\ \mathcal{C}_1 &= \left\{\chi \in \mathbb{R}^6, |e_\theta| \geq \theta_{q_1} \ \wedge \ q = 1\right\}, \\ \mathcal{D}_0 &= \left\{\chi \in \mathbb{R}^6, |e_\theta| \geq \theta_{q_2} \ \wedge \ q = 0\right\}, \\ \mathcal{D}_1 &= \left\{\chi \in \mathbb{R}^6, |e_\theta| \leq \theta_{q_1} \ \wedge \ q = 1\right\}, \end{aligned} \tag{5.4}$$

where $\theta_{q_2} > \theta_{q_1}$ are two positive constants defining the hysteresis mechanism. The constant ϵ_q used in function $g(e_\theta, \epsilon_q)$ satisfies $\epsilon_q \in (\theta_{q_1}, \theta_{q_2})$. With this choice of θ_{q_1}, θ_{q_2} and ϵ_q, the hysteresis is well defined and the control system never brakes when the orientation error is smaller than the lower threshold θ_{q_1}. A graphical representation is depicted in Figure 5.4: the vehicle starts oriented as the dotted line, labeled with a 1. Since $|e_\theta| > \theta_{q_2}$ the control system is activated ($q = 1$) and

[2]Clearly, if a static Frenet frame is used, it is sufficient to replace the expression of $\dot{s}$ in (5.3) with the one in (2.8).

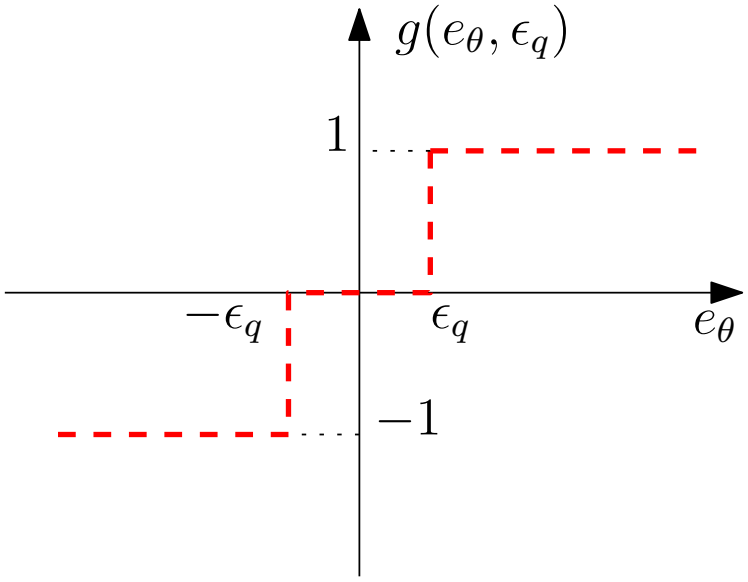

Figure 5.3: Function $g(e_\theta, \epsilon_q)$ in Equation (5.2) [8].

the vehicle turns right to reduce the orientation error ($\omega = -\frac{v}{R}$). $|e_\theta|$ is reduced below the threshold θ_{q_2} (dotted line labeled with a 2) until it becomes smaller than the lower threshold θ_{q_1} (dotted line labeled 3). Only at this point, the discrete dynamics of the hybrid controller in (5.3) is activated, turning $q = 0$ and stopping the braking action on the vehicle ($\omega = -g(e_\theta, \epsilon_m)\frac{v}{R} = 0$ since $|e_\theta| \leq \theta_{q_1} < \epsilon_q$).

In practice, the robot takes the control authority only to ensures that the attitude error $|e_\theta|$ remains small. The control authority is sharply allocated similarly to Section 4.3.1. The logic variable q defines who retains the control authority ($q = 0$ if the user is in control, $q = 1$ is the robot is in control), as depicted in Figure 5.5.

Notice that, in the control design the parameter $\epsilon_q \in (\theta_{q_1}, \theta_{q_2})$ is used. Observe that the performance of the controller does not depend on ϵ_q. The controller works equally for all $\epsilon_q \in (\theta_{q_1}, \theta_{q_2})$, regardless of the value in this interval. Then ϵ_q may be imposed to $\epsilon_q = (\theta_{q_2} + \theta_{q_1})/2$ and therefore it does not require tuning.

Stability issues

The proposed controller is define to limit the attitude error $|e_\theta|$ to solve the relaxed path following (4.4), as stated by the following theorem.

Theorem 4 (Path following via braking guidance)**.** *Consider the vehicle kinematic model* (2.10)*. Let us assume that the following local condition holds:*

$$|\gamma| < \frac{1}{R}, \tag{5.5}$$

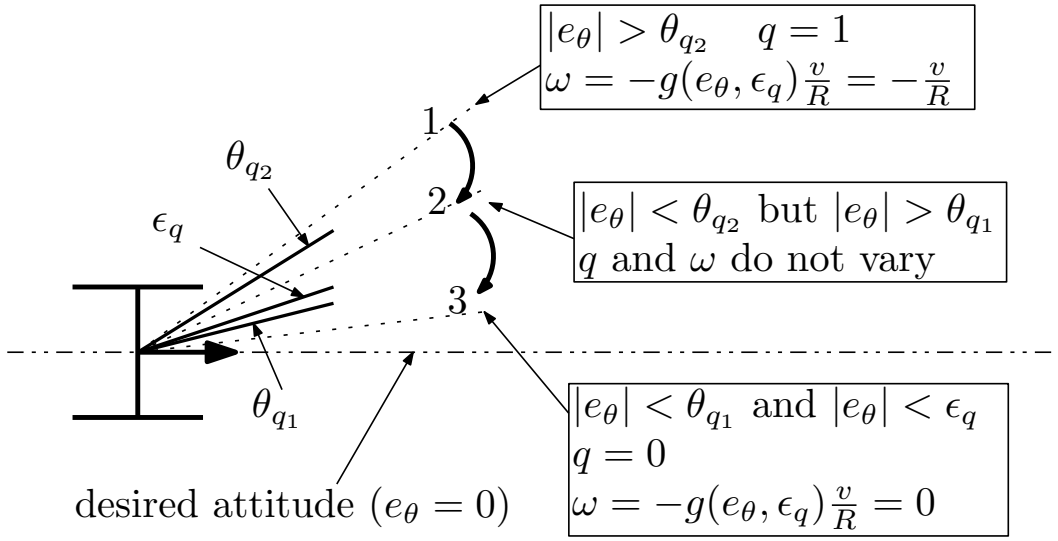

Figure 5.4: Angular hysteresis mechanism [8].

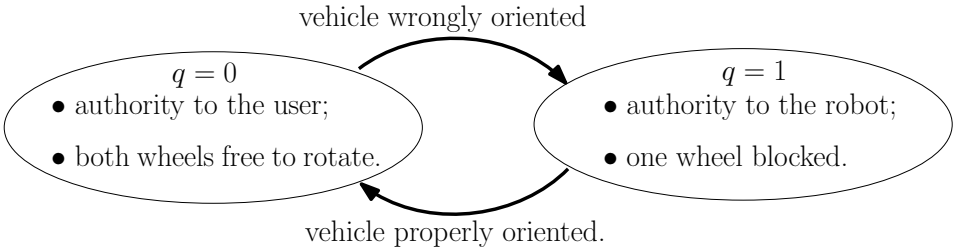

Figure 5.5: Schematic representation of the hybrid system ensuring hysteresis [5].

where

$$\begin{aligned} \gamma &= c(s)\dot{\xi} + \left(-c(s)\,\dot{\xi}\,l_x + \sin\tilde{\theta}\right)\frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y), \\ \dot{s} &= v\dot{\xi}, \\ \dot{\xi} &= \cos(\tilde{\theta}) + \kappa_x l_x. \end{aligned} \tag{5.6}$$

Under the assumptions of Theorem 3, given two arbitrary non-negative constants l_∞ *and* $\tilde{\theta}_\infty$ *in* (4.4)*, there exists an upper hysteresis threshold* θ_{q_2} *such that the controller defined in* (5.3) *ensures that the path following requirements* (2.11) *hold.*

Proof of Theorem 4. Consider precautionary that controller (5.3) is never active (i.e., $q = 0$) if $|e_\theta| \leq \theta_{q_2}$. Consider then the Lyapunov function

$$V = \frac{1}{2}e_\theta^2,$$

whose time derivative along the solutions is

$$\dot{V} = e_\theta \dot{e}_\theta = e_\theta \left(\omega - c(s)\dot{s} - \dot{l}_y \frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y) \right) = e_\theta \left(\omega - v\gamma \right). \tag{5.7}$$

When $|e_\theta| \leq \theta_{q_2}$, we have $\omega = -\operatorname{sign}(e_\theta)\frac{v}{R}$ in (5.2). Then, the Lyapunov function derivative (5.7) is rewritten as

$$\dot{V} = e_\theta \left(\omega - v\gamma \right) = v e_\theta \left(\frac{\operatorname{sign}(e_\theta)}{R} - \gamma \right).$$

Under the local condition (5.5), we have

$$\dot{V} = v e_\theta \left(\frac{\operatorname{sign}(e_\theta)}{R} - \gamma \right) < 0, \forall e_\theta \neq 0, \tag{5.8}$$

since $v > 0$ holds in finite time by assumptions of Theorem 3. Condition (5.8), in combination with the hybrid map defined by (5.11), ensures that $|e_\theta|$ enters in finite time and remains inside the region $|e_\theta| \leq \theta_{q_2}$. Then, since θ_{q_2} acts as upper limit of the attitude error e_θ, given the tolerated errors l_∞ and $\tilde{\theta}_\infty$, it is always possible by Theorem 3 to find an upper hysteresis threshold θ_{q_2} such that the path following conditions (2.11) hold. □

Notice that Theorem 4 holds only if the local condition (5.5), stating that $|\gamma|$ is small, is satisfied.

The term γ represents the curvature that the vehicle has to compensate to follow and approach the path via the trajectory defined by $\delta(l_y)$, in absence of attitude error, i.e., when $e_\theta = 0$. For instance, suppose that the vehicle is exactly on the path ($l_x = l_y = \tilde{\theta} = e_\theta = \delta(l_y) = 0$). In this case we have $\gamma = c(s)$, i.e., γ coincides with the path curvature. If the vehicle is not on the path, the term γ is, in general, larger since the vehicle has to compensate also the curvature of the approaching trajectory (notice that γ depends on the derivative of the approaching angle δ). As a consequence, the local condition (5.5) states that the curvature to be compensated has to be smaller than the minimum curvature the vehicle can physically follow (which is clearly $\frac{1}{R}$, since the vehicle is a Dubbin's car).

Notice that, the curvature $|\gamma|$ is very large, i.e., the local condition (5.5) is theoretically not satisfied, if

- the path curvature $|c(s)|$ is very large;
- the approaching trajectory is very sharp (i.e., large $\frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y)$);
- the vehicle is very far from the path (i.e., large l_x, l_y, $\tilde{\theta}$).

In the practice, the path curvature $c(s)$ and the approaching curvature $\frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y)$ are chosen small by the path planner algorithm and the control designer, respectively, to improve the user's comfort. Moreover the vehicle is supposed to be close to the path (i.e., small l_x, l_y, $\tilde{\theta}$). In fact, although theoretically the controller may allow us to approach a path very far from the vehicle, in practice the robot plans a path always starting from its current state (i.e., the path following error is almost zero at the initial time). This planning choice is needed for user's safety since the obstacles are avoided only if the vehicle is in a neighborhood of the path, hence it is meaningless to place the path very far from the vehicle in the initial time instant. As a consequence, the local condition (5.5) always holds in practice.

Delay compensation

The experimental results reported in [10] show that the actuation delay worsens the system performance. We propose a method to compensate this delay using predictions on the curvilinear coordinates, i.e. the control action ω in (5.3) is computed on a feedforward term. The expression $\omega^+ = -g(e_\theta, \epsilon_q)\frac{v}{R} = -g(\tilde{\theta} - \delta(l_y), \epsilon_q)\frac{v}{R}$ is replaced by $\omega^+ = -g(\tilde{\theta}_p - \delta(l_{y,p}), \epsilon_q)\frac{v}{R}$, where $\tilde{\theta}_p$ and $l_{y,p}$ denote the predicted values of $\tilde{\theta}$ and l_y, respectively. The prediction is computed via Euler's forward integration rule, i.e.,

$$\begin{aligned} \tilde{\theta}_p &= \tilde{\theta} + \dot{\tilde{\theta}}\Delta\tau, \\ l_{y,p} &= l_y + \dot{l}\Delta\tau, \end{aligned} \tag{5.9}$$

where $\Delta\tau$ is the estimated actuation delay and where the control inputs v and ω are supposed constant.

5.3.2 Simulation results

Simulation results are presented to analyze the path following performance of the braking guidance. For simplicity, a static Frenet (2.8) frame is used: as a consequence, the coordinate l reported in the following simulations represents the geometrical distance between vehicle and path.

In the reported simulation results, the initial conditions are $[x(0), y(0), \theta(0)]^T = [1.3\text{ m}, 8\text{ m}, 0^\circ]^T$, the actuation delay is set to $\Delta\tau = 100$ ms and we set the hysteresis threshold to $\theta_{q_2} = 40^\circ$ and $\theta_{q_1} = 5^\circ$. The approaching angle $\delta(l)$ is chosen as in (4.10). The user's behavior is simulated superimposing two different functions for the forward velocity v and the angular velocity ω. The implemented forward velocity v oscillates with amplitude of 0.5 m/s, has mean value of 1 m/s and frequency 0.3 Hz (see Figure 5.6). The highly oscillating behavior simulates a user that continuously slows down and accelerates, roughly emulating the effect of walking. Instead, the angular velocity, of relevance whenever no braking action is applied, is

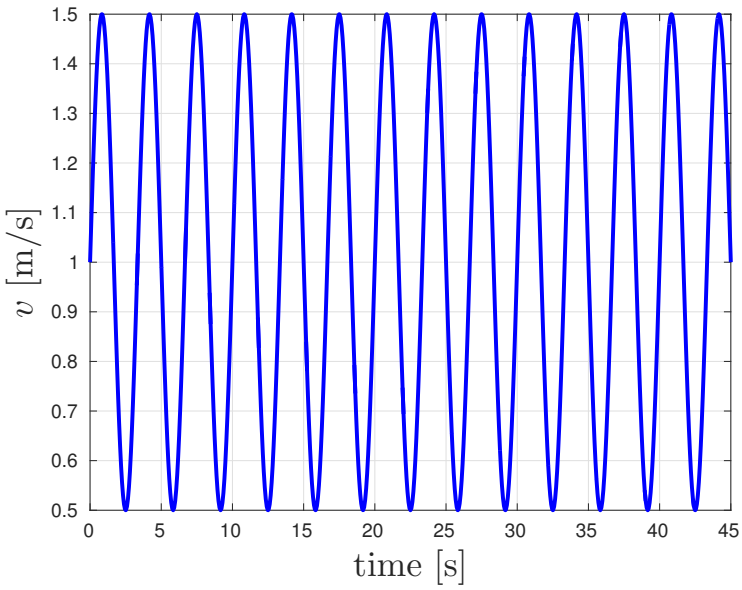

Figure 5.6: Simulated user's forward velocity v.

simulated with a low-pass filtered white noise, hence generating a random walk for the orientation θ in the reference model (2.1).

We first present the results regarding the actuation delay by comparing the path tracked in the ideal case, i.e., when the actuation delay is not present (dotted line in Figure 5.7), with the paths tracked when the actuation delay is actually present. Notice how the controlled paths solve the relaxed path following problem (2.9), i.e., l_∞ and $\tilde{\theta}_\infty$ are both greater than zero even in the ideal zero delay case to meet the requirements of the shared authority. Notice also from Figure 5.8 that the chattering problem has been eliminated with respect to Figure 5.2. Moreover, it is evident from Figure 5.7 how the presence of the delay implies a more erratic and, hence, uncomfortable path following, while the compensation lets the vehicle behave very close to the ideal situation. It is worthwhile to notice that the compensation strategy is robust with respect to the hypothesis of assumed constant forward velocity v. Finally, it has been observed that underestimating the delay generates a shorter number of braking actions of longer duration with respect to overestimating it. For user comfort, is then preferable the first choice. To clarify the impact of the thresholds θ_{q_1} and θ_{q_2} on the achievable trajectories and related performance, we reported in Figure 5.9 the results for different values of θ_{q_2} when $\theta_{q_1} = 5°$ (indeed, modifying both of the thresholds simultaneously makes the comparison more difficult). The figure reports the time evolution of the distance to the path with a set-up similar to Figure 5.7 and the different values of the critical value of l (i.e., $l^\circ = |\delta^{-1}(\theta_{q_2})|$ in the proof of Theorem 2, i.e., fixed a tolerated attitude error θ_{q_2}, the vehicle continues to get closer to the path until $|l| > l_\circ$). The critical values l° are depicted as horizontal

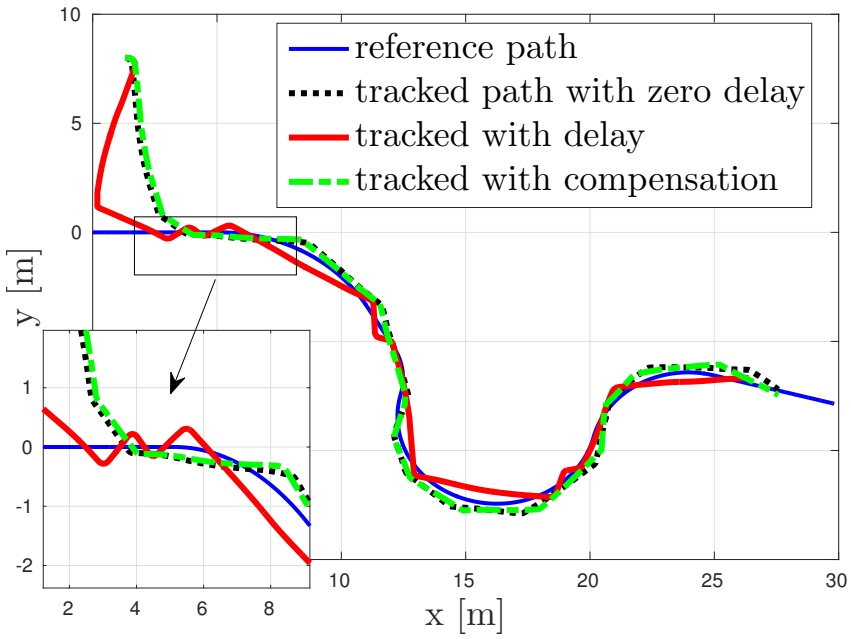

Figure 5.7: Comparison between the tracked paths: overall simulated trajectory, with an excerpt of the first portion of the paths [5].

lines, one for each choice of θ_{q_2}. The reported graph shows how the critical values can safely be adopted as (conservative) performance indexes for the path following. Notice that decreasing the value of θ_{q_2} (i.e., shifting the control authority to the robot) leads to a better approximation of the path (smaller errors), at the price of a higher number of braking actions. Finally, Table 5.1 reports quantitative measures, i.e., number of braking actions and overall braking time, for the trajectories in the simulations of Figure 5.9. The ratio between these two measures is also reported,

Threshold θ_{q_2} [deg]	n° of braking actions [-]	braking time [s]	mean braking time [s]
20	29	2.20	0.08
30	19	2.46	0.13
40	12	1.87	0.16

Table 5.1: Comparisons for different choices of the threshold θ_{q_2} [5].

i.e., mean braking time. As the authority of the control actions shifts toward the user (increasing θ_{q_2}) the number of control actions decreases but still we have an increasing mean braking time.

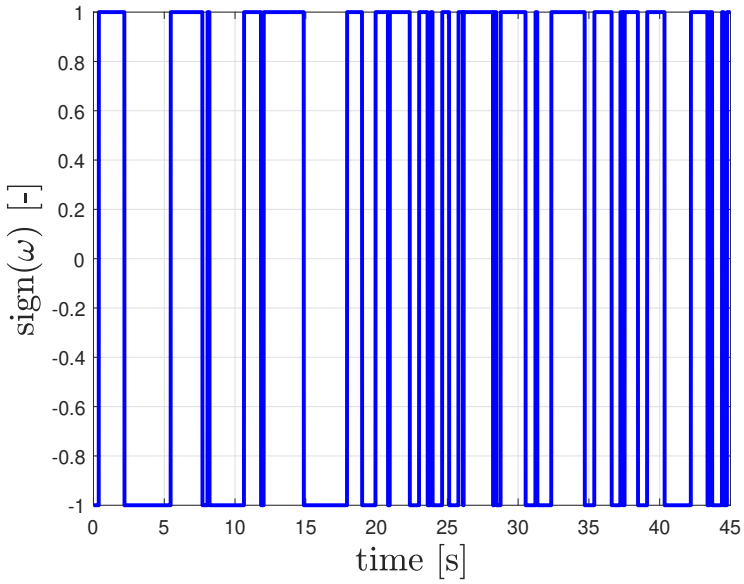

Figure 5.8: Control action with no chattering for the zero delay case in Figure 5.7.

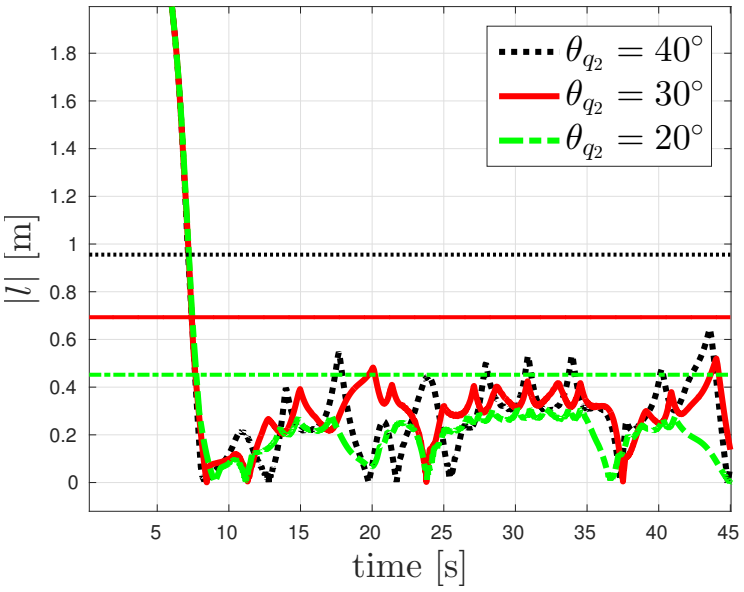

Figure 5.9: Time evolution of the path distance error for different choices of the threshold θ_{q_2} ($\theta_{q_1} = 5°$). The thin horizontal lines represents the critical distances $l°$ [5].

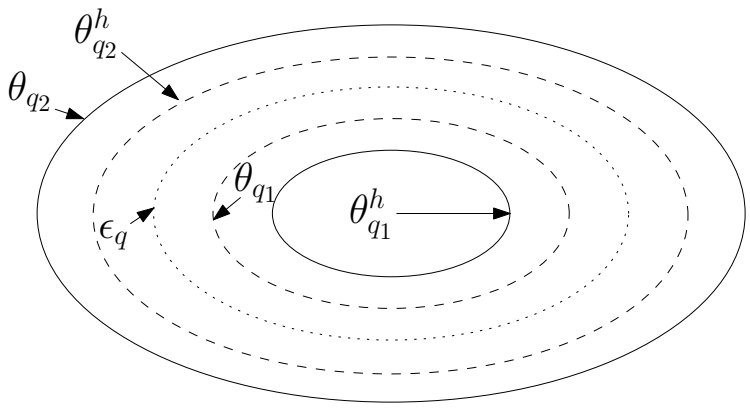

Figure 5.10: Graphical representation of the hysteresis thresholds [8].

5.4 Integration of haptic and braking guidance

The haptic guidance is used to suggest the user to steer the vehicle towards the path. In principle, if the user is cooperative, i.e. he/she properly steers the vehicle when the haptic system is active, the path following requirements (2.11) may be satisfied without using the brakes, then improving the user's comfort.

To design the activation policy of the bracelets, recall that we have shown that controller (5.3) solves the path following problem by maintaining the angular error $|e_\theta|$ small via an hysteresis mechanism. The same logic is also used to activate the haptic system. To improve the user comfort, we require that:

- if the user is leaving the path, the bracelets vibrate before the vehicle brakes. This way, the user is allowed to follow the path without braking interventions;
- the bracelets are deactivated at a smaller angular error $|e_\theta|$, i.e., the lower threshold of the bracelets hysteresis is smaller than the lower threshold of the brakes hysteresis. This way, the undesired situation where the haptic system does not provide suggestions to the user and the vehicle brakes is avoided.

Therefore, if we denotes by $\theta^h_{q_2} > \theta^h_{q_1}$ the hysteresis thresholds of the haptic system (the superscript h stands for "haptic"), we impose $\theta_{q_2} > \theta^h_{q_2}$, $\theta_{q_1} \geq \theta^h_{q_1}$ and $\theta^h_{q_2} > \theta_{q_1}$. Moreover, we pick $\epsilon_q \in (\theta_{q_1}, \theta^h_{q_2})$ in (5.2), as depicted in Figure 5.10.

Then, let $\psi \in \{-1, 0, 1\}$ be the working mode of the bracelets. We have $\psi = 1$ when the left bracelet vibrates (i.e., the user is suggested to turn left), $\psi = -1$ when the right bracelet vibrates (i.e., the user is suggested to turn right) and $\psi = 0$ when the haptic system is not active (no vibrations, then no indications are provided to the user). To describe the haptic system with an hybrid dynamics similar to (5.3), we introduce a logic state p having discrete dynamics only. Its meaning is similar to the meaning of q in (5.3). When $p = 1$, the bracelets vibrate since the attitude error

$|e_\theta|$ is large, while when $p = 0$ the bracelets do not vibrate since the attitude error $|e_\theta|$ is small. The hybrid dynamical system managing the haptic system is designed as follows:

$$\begin{cases} \dot{\psi} &= 0, \\ \dot{p} &= 0, \end{cases} \qquad \Xi \in \mathcal{C}^h,$$
$$\begin{cases} \psi^+ = -g(e_\theta, \epsilon_q), \\ p^+ = 1 - p, \end{cases} \qquad \Xi \in \mathcal{D}^h, \tag{5.10}$$

where $\Xi = [\psi, e_\theta, p]^T$, the flow set is $\mathcal{C}^h = \mathcal{C}_0^h \cup \mathcal{C}_1^h$ and the jump set $\mathcal{D}^h = \mathcal{D}_0^h \cup \mathcal{D}_1^h$, where

$$\begin{aligned} \mathcal{C}_0^h &= \left\{ \Xi \in \mathbb{R}^3, |e_\theta| \leq \theta_{q_2}^h \ \wedge \ p = 0 \right\}, \\ \mathcal{C}_1^h &= \left\{ \Xi \in \mathbb{R}^3, |e_\theta| \geq \theta_{q_1}^h \ \wedge \ p = 1 \right\}, \\ \mathcal{D}_0^h &= \left\{ \Xi \in \mathbb{R}^3, |e_\theta| \geq \theta_{q_2}^h \ \wedge \ p = 0 \right\}, \\ \mathcal{D}_1^h &= \left\{ \Xi \in \mathbb{R}^3, |e_\theta| \leq \theta_{q_1}^h \ \wedge \ p = 1 \right\}. \end{aligned} \tag{5.11}$$

Notice that also the correctness of the haptic controller (5.10) is proved once $\epsilon_q \in (\theta_{q_1}, \theta_{q_2}^h)$, hence it is chosen for simplicity as $\epsilon_q = (\theta_{q_2}^h + \theta_{q_1})/2$.

5.5 Experimental results

We conduct two studies with older adult participants (details can be found in Table 5.2) in which the testers completed different paths using the *FriWalk* in one of the laboratory of the University of Trento. In the first study (with 4 males, 10 females, aging between 65 and 75 years old), the participants were asked just to travel along a couple of paths using the passive braking system controller (5.3), while in the second study (with 6 males, 9 females, aging between 64 and 100 years old) a more extensive study, with more than eight paths for each participants were considered. In this second study, we tested the three combinations of guidance:

1. *bang-bang guidance*: the users are guided using the braking system (just capable of blocking the wheels) only. In this case the bracelets never vibrate;

2. *haptic guidance*: the users are guided using the bracelets only. In this case the vehicle never brakes, even if the user does not pay attention to the haptic clues and departs from the planned path;

3. *combined guidance*: the users are guided using both the brakes the bracelets. If the user does not pay attention to the haptic clues and departs from the planned path, the braking system is activated. It is recalled that the integration between brakes and bracelets is obtained by properly tuning the hysteresis thresholds as in Figure 5.10 .

	Males	Females	Oldest	Youngest	Walking Problems
Study 1	4	10	75	65	4 (28.6%)
Study 2	6	9	100	64	7 (43.8%)

Table 5.2: Cohort characteristic.

Participants were contacted through the Municipality of Pergine Valsugana and the senior center "Sempreverde" of Mattarello (both in the Trento province). Of this group, some of the participants usually use walking aids, such as crutches and/or a walker (28.6% of Study 1 and 43.8% of Study 2). All the participants were informed that data collection and the information provided are covered by the ethical rules conceived for the ACANTO project [1] and that they could quit the experiment at anytime. Once consent was obtained, they were invited to perform the tasks with the *FriWalk*. Before starting, an experimenter explained the features of the robotic walker and its guidance modalities. All participants completed a first trial (which was common for everybody) to take confidence with the robotic walker and its movements. Every participant completed at least one path per guidance system. The path to follow in every experiment was randomly selected in a portfolio of 15 paths generated using a planning algorithm [19] optimizing human comfort. In the laboratory arena, three tables were placed to emulate an actual indoor environment (see the rectangular obstacles in Figure 5.11).

5.5.1 Quantitative analysis

The controller hysteresis threshold are set to $\theta_{q_2} = 40°$, $\theta^h_{q_2} = 30°$ and $\theta_{q_1} = \theta^h_{q_1} = 20°$. Four sample trajectories along a randomly selected path are reported in Figure 5.11, Figure 5.13 and Figure 5.15 for the bang-bang braking guidance, haptic guidance and combined guidance, respectively. The localization is provided with an EKF [88] fusing the encoder data and the QR codes, positioned on the floor using the deployment [87] and read by the available front camera pointing downwards. Figure 5.12, Figure 5.14 and Figure 5.16 reports the time evolution of the attitude error e_θ for the Exp_1 in Figure 5.11, Figure 5.13 and Figure 5.15, respectively.

Bang–bang braking system

In this experiment, we asked the user to follow a desired path (blue solid line in Figure 5.11) while the *FriWalk* is controlled by the bang-bang braking strategy reported in Section 5.3. Recall that the user is always in control when the attitude error $|e_\theta|$ is below the inner threshold θ_{q_1} (see Figure 5.12), while the robot is always in control when $|e_\theta|$ is greater than the outer threshold θ_{q_2}. Within the two values, the control authority depends on the history of e_θ due to the hysteresis nature of the controller. Notice from Figure 5.12 that as soon as $|e_\theta| > \theta_{q_2}$, the controller

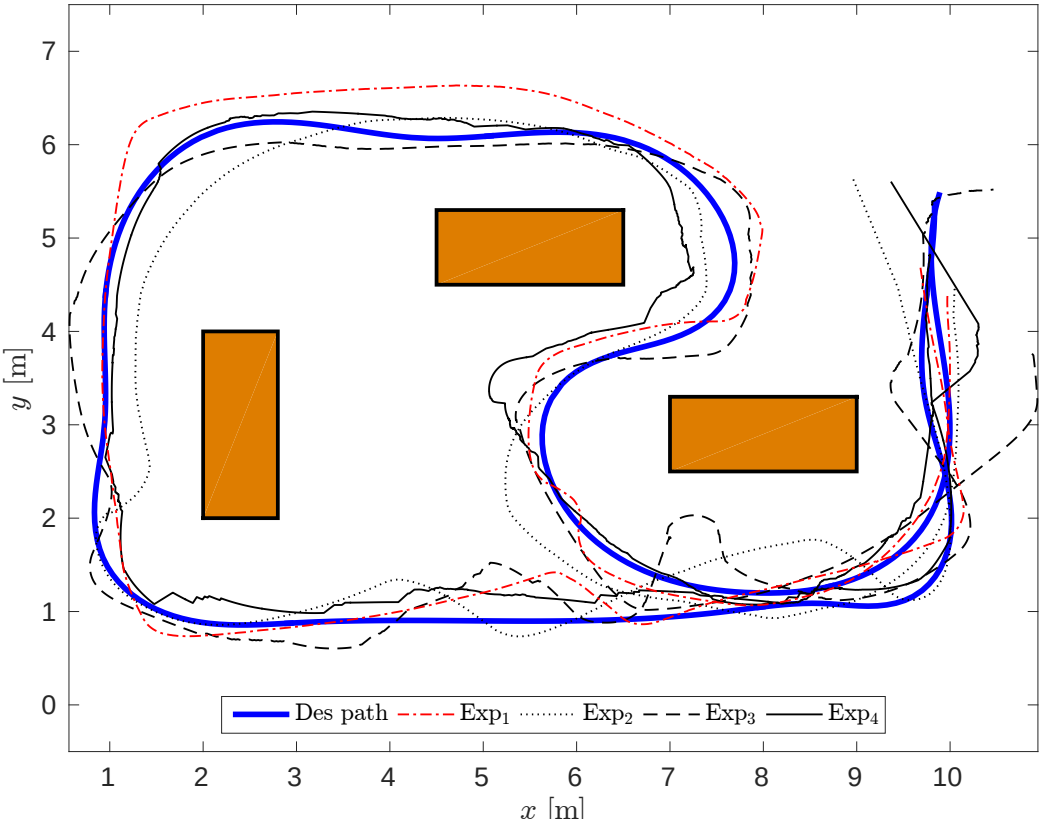

Figure 5.11: Experimental trajectories for four participants along a randomly selected path (solid thick line) followed with the *bang-bang* strategy. The rectangles represent the obstacles (i.e., tables) in the environment.

kicks in and $|e_\theta|$ does not increase anymore, since the controller is able to steer the user towards the desired path.

Haptic guidance

Examples of followed paths obtained via the haptic guidance described in (5.10) are reported in Figure 5.13. Since the vehicle never brakes, the user is always in control, independently from the value of e_θ, while steering suggestions are given by the haptic interfaces. The user can either follow the suggestion of the bracelets and then steer to the desired path, or ignore the stimuli. It may happen that the user wrongly interprets the information coming from the bracelets, as in *Exp*$_4$ of Figure 5.13, where the user follows a wrong path. The time evolution of e_θ for the haptic guidance is reported in Figure 5.14. Notice a larger deviation with respect to Figure 5.12, even if the bracelets are activated for smaller values of $|e_\theta|$ with respect

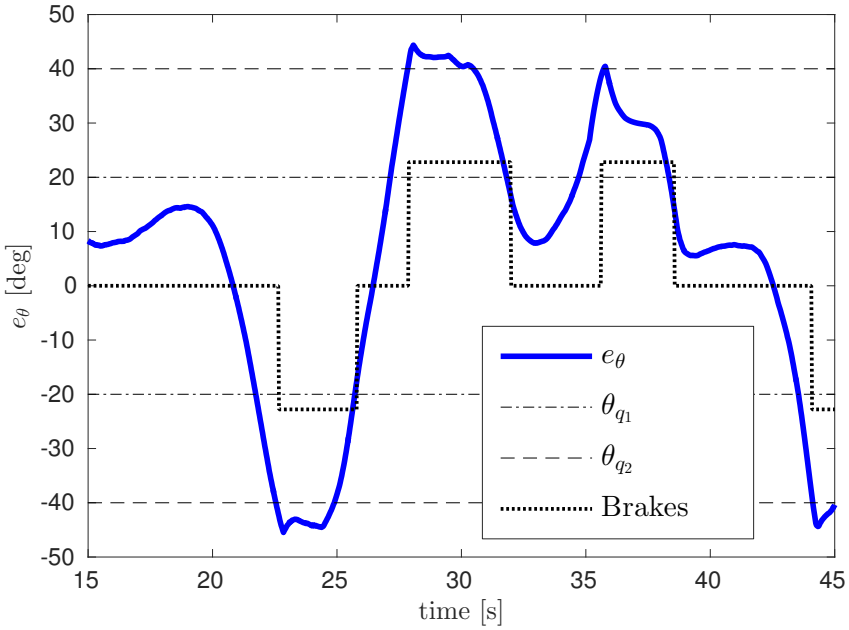

Figure 5.12: Time window of 30 seconds describing the evolution of the error e_θ for the Exp_1 in Figure 5.11. The curve *Brakes* reports the sign of the commanded angular velocity $\operatorname{sign}\omega$ in (5.3), scaled for visibility ($\omega > 0$ when the left wheel is blocked and the vehicle turns right, $\omega < 0$ when the right wheel is blocked and the vehicle turns left, $\omega = 0$ when the wheels are free and the user is in control of the motion). The controller thresholds θ_{q_1} and θ_{q_2} are also reported.

to the brakes since $\theta_{q_2} = 40° > \theta^h_{q_2} = 30°$. This takes place since, even if the haptic system is activate, the control authority remains to the user which is less efficient in controlling the attitude in comparison with the mechanical actuation (i.e., brakes) of the robot.

Combination between brakes and bracelets

The paths followed combining the braking guidance and the haptic guidance are reported in Figure 5.15, while the evolution of the attitude error is reported in Figure 5.16. Notice that, whenever the braking system is active, the bracelets vibrate. However, if the attitude error e_θ is limited, only haptic clues are provided to the user, hence with no activation of the brakes. In particular, around $t = 18$ seconds the attitude error exceeds the haptic threshold, i.e., $|e_\theta| > \theta^h_{q_2}$, and then the bracelets vibrate. Notice that the user is cooperative and recognizes the haptic clue, properly steering the vehicle. As a consequence, the attitude error $|e_\theta|$ remains below the braking threshold θ_{q_2} and the vehicle does not brake. In the following however, the attitude error exceeds also the braking threshold, i.e., $|e_\theta| > \theta_{q_2}$, then the vehicle

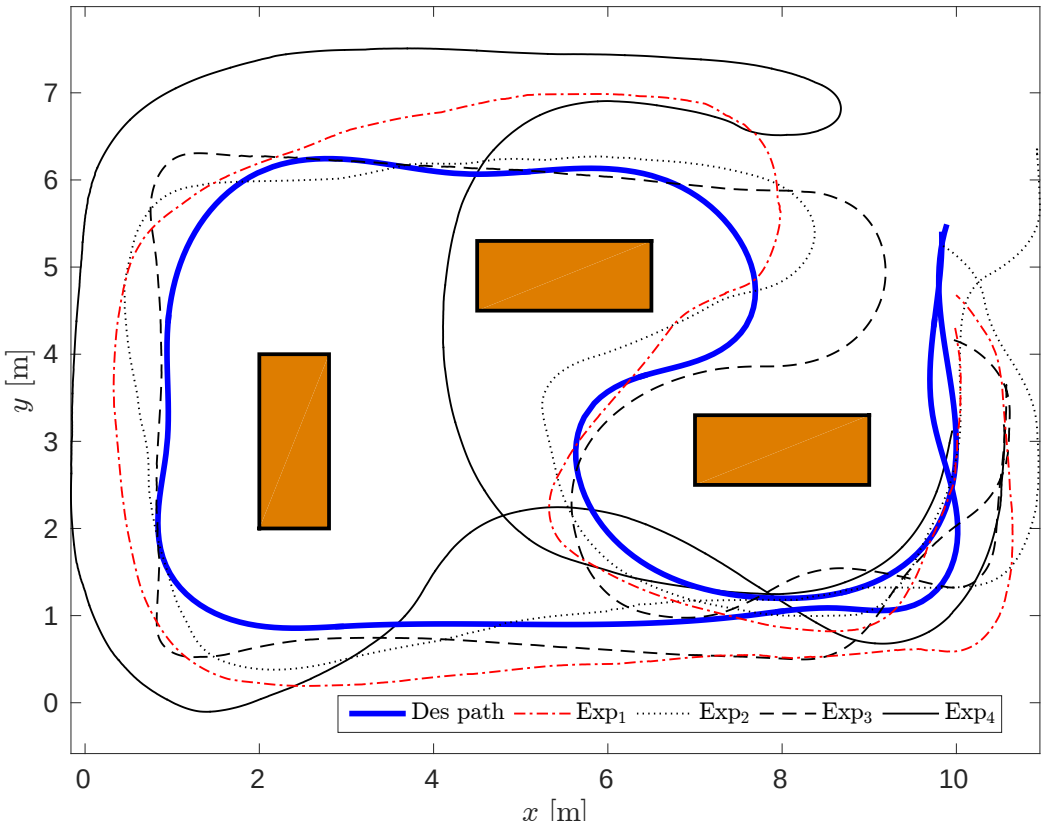

Figure 5.13: Experimental trajectories for four participants along a randomly selected path (solid thick line) followed with the *haptic* strategy. The rectangles represent the obstacles (i.e., tables) in the environment

brakes and the bracelets vibrate. Notice that the braking system ensures that the attitude error $|e_\theta|$ never reaches the large values observed in Figure 5.14, where only the bracelets are used.

Performance analysis

We report the average controller performance in Table 5.3 for all the run experiments. The reported features are:

- average lateral distance from the path, i.e., $|l_y|$ and its standard deviation, as indication of geometrical distance from vehicle and path;
- the number of interventions of the controller, i.e., the number of times in which the brakes or the bracelets are activated;

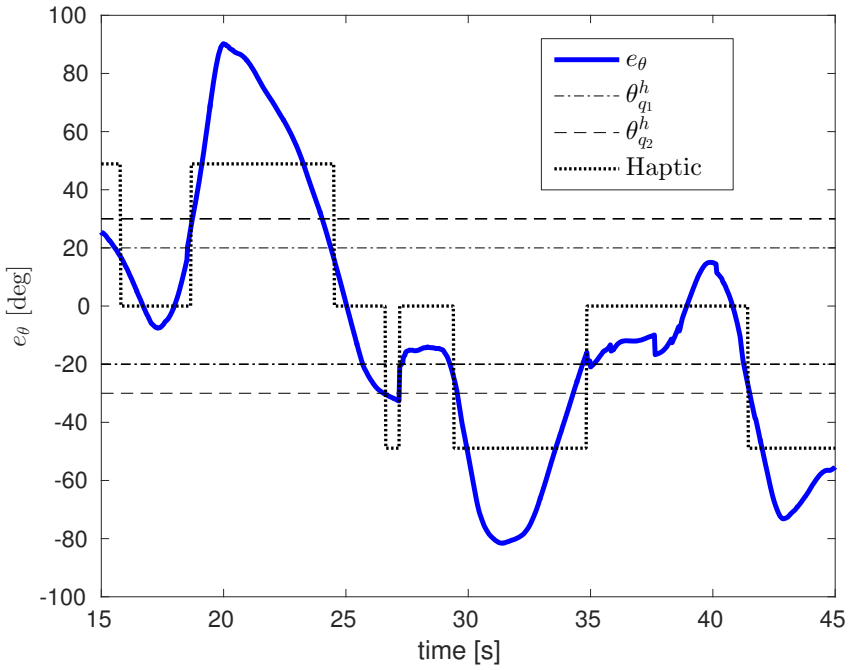

Figure 5.14: Time window of 30 seconds describing the evolution of the error e_θ for the Exp_1 in Figure 5.13. The curve *Haptic* reports the sign of the commanded angular velocity, i.e., ψ in (5.10), scaled for visibility ($\psi > 0$ when the user is suggested to turn right, $\psi < 0$ when the user is suggested to turn left, $\psi = 0$ when no haptic clues are provided). The controller thresholds $\theta^h_{q_1}$ and $\theta^h_{q_2}$ are also reported.

- the percentage of time in which the controller is active during the guidance.

Notice that the use of the braking system (both in the bang-bang guidance and in the combined guidance) ensures a much smaller lateral distance $|l_y|$ with respect to the use of bracelets only (haptic guidance) since the brakes allow the robot to take the control authority. The number of interventions, also reported in Figure 5.17-(a), is maximum for the bang-bang braking strategy. The haptic clues provided by the vibrating bracelets (haptic and combined guidance) considerably help the user to remain close to the path, hence reducing the number of interventions. Notice also from Table 5.3 and Figure 5.17-(b) that the controller remains active for 21% of the time in the case of bang-bang braking guidance and for 45% of the time in the case of haptic guidance. When the combined strategy is used, the percentage of time in which the controller is active is approximately halved for both brakes and bracelets.

Overall, the braking guidance requires corrections frequent (large number of interventions) but very short (small percentage of time), and ensures a small deviation $|l_y|$ from the path. Conversely, the haptic guidance requires sporadic corrections

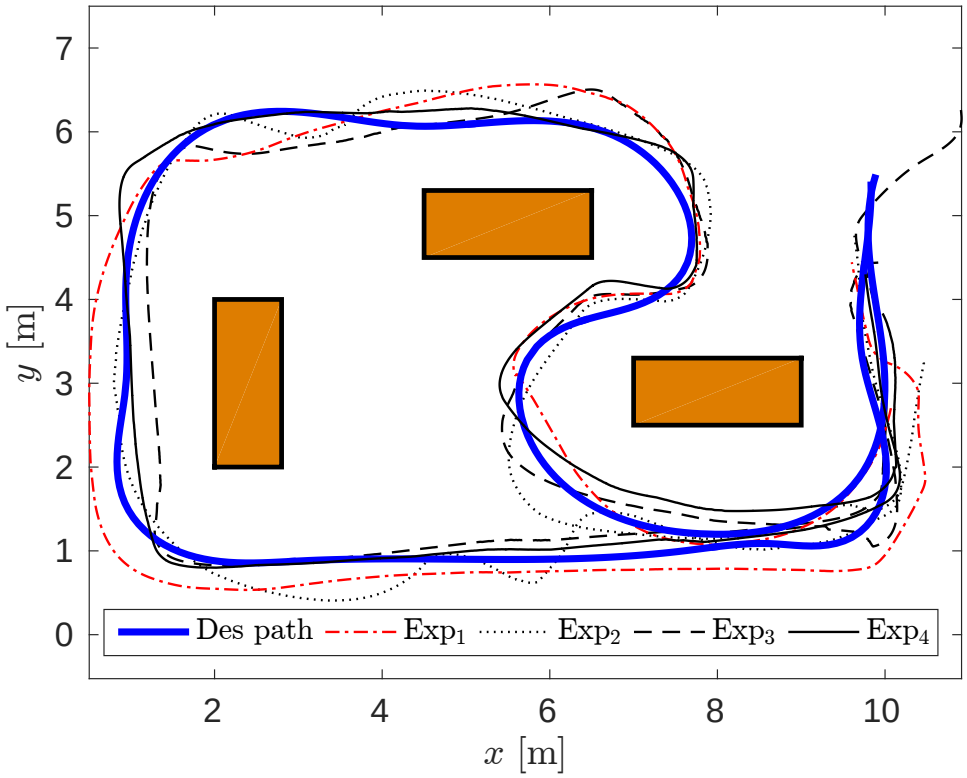

Figure 5.15: Experimental trajectories for four participants along a randomly selected path (solid thick line) followed by combining the braking and the haptic guidance. The rectangles represent the obstacles (i.e., tables) in the environment.

(small number of interventions) but significantly longer (large percentage of time), and is not capable of maintaining small the deviation $|l_y|$ from the path.

The data in Table 5.3 and Figure 5.17 show that the advantages can be combined using both brakes and bracelets. The combined guidance produces the same small lateral deviation $|l_y|$ from the path of the braking guidance. Although the number of activations of the bracelets slightly increases, the vehicle brakes half of the times and the amount of time in which brakes and bracelets are active is also halved, hence disturbing much less the user. Overall, the combination of the braking and the haptic guidance systems is then shown to outperform the two individual approaches in isolation. In fact, whilst it retains the same level of the users' perceived comfort typical of the haptic-only guidance, it ensures an adequate path following performance by means of the braking-only guidance.

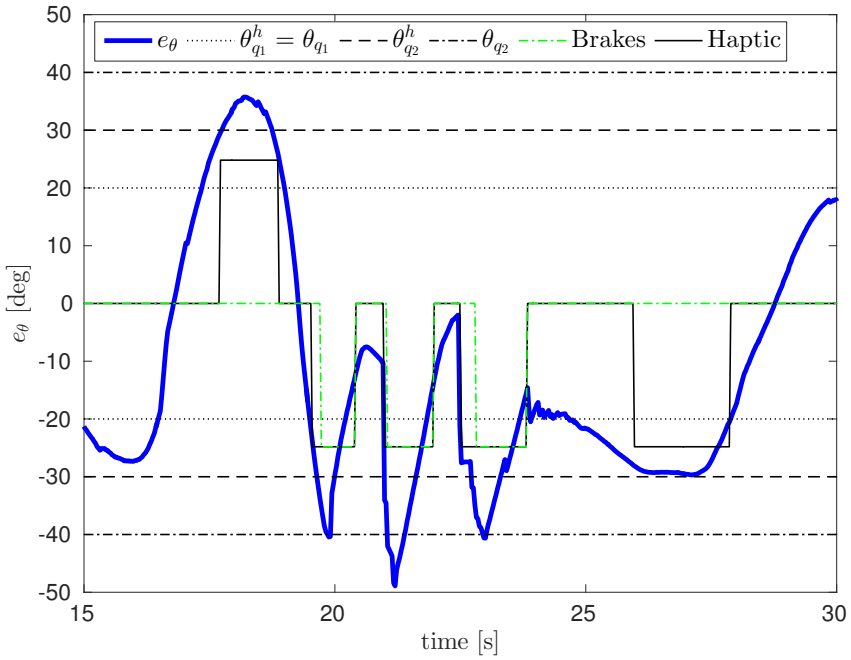

Figure 5.16: Time window of 15 seconds describing the time evolution of the error e_θ for the Exp_1 in Figure 5.15. Thresholds and signs of the suggested angular velocity are reported as in figures 5.12 and 5.14.

5.5.2 Users' evaluation

In both studies, we used a questionnaire to conduct a structured interview in order to collect impressions and opinions of participants. After the tests with the robotic walker, participants were invited to sit next to an experimenter, who conducted the structured interview by reading the items of the questionnaire. Participants were asked to answer using yes or no and a 5 point Likert scale (1 "not at all", 2 "a little bit", 3 "moderately", 4 "very much", 5 "extremely"). The questions concerned different features of the interaction with the robotic walker (i.e., if they felt vibrations during the use, if it was clear if and when the *FriWalk* decided the path and if they felt pushed or blocked), followed by items on the pleasantness of usage, the ease of learning, the control over the robot and the adaptability of the walker. The questionnaire is reported in Table 5.4, while the results for *Study 2* are reported in Figure 5.18.

We observed an overall positive opinion on the *FriWalk* control modalities. Although the quantitative results presented in Section 5.5.1 underline that the *combined* strategy ensures the same path following error of the *braking* strategy, the psychological evaluation summarized in Figure 5.18 shows that the *combined* strategy offers superior comfort, especially in terms of easy of learning (item L3), control

Feature	Bang-Bang	Haptic	Combined	
Average $\|l_y\|$ [m]	0.24	0.59	0.25	
std $\|l_y\|$ [m]	0.10	0.27	0.06	
			Brakes	Haptic
n° of interventions	20.30	7	10.67	8.35
std n° of interventions	12.50	2	3.24	2.89
% time active controller	21.22	45.95	12.59	24.71
std % time active controller	10.40	18.60	10.54	19.45

Table 5.3: Summary of the controller performance.

(items C2 and C3) and adaptability (item A3). Participants stated they were aware that the system was programmed to decide the path to follow but they judged this feature as not disturbing. Moreover, when the suggested route is rendered uniquely with vibrations of wristbands on participants' arms, none reported the sensation to be pulled, blocked or pushed. Participants evaluated the interaction with the *FriWalk* as pleasant and not frustrating, indicating that it was easy to learn its use. Furthermore, they perceived the system in good accordance with their usual way of walking or their natural speed.

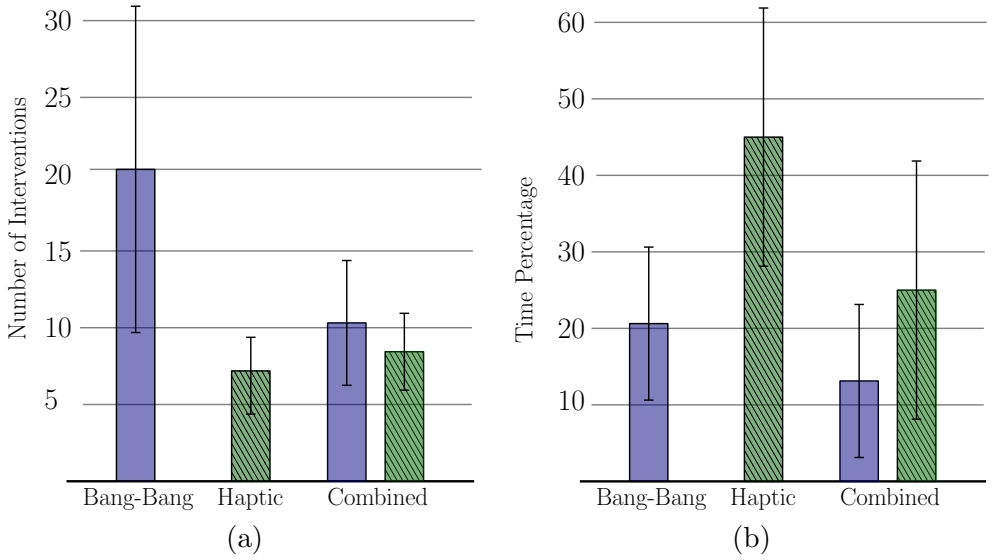

Figure 5.17: Comparison of the actuation actions between strategies: (a) number of controller interventions, (b) percentage of time for which the controller is active. In the combined strategy, the actuation actions of brakes (filled blue bar) and bracelets (dashed green bar) are reported separately.

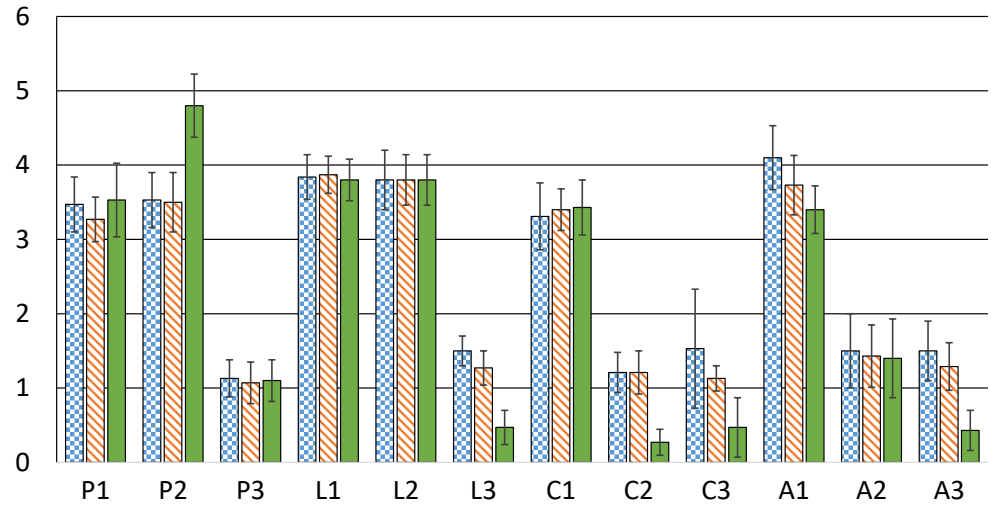

Figure 5.18: Means and standard deviations for the items on pleasantness (P1, P2, P3), ease of learning (L1, L2, L3), control perception (C1, C2, C3) and adaptability (A1, A2, A3) of the *FriWalk* for the *bang–bang* (squared pattern bars), *haptic* (raising tiling patterns bar) and *combined* (solid filled bars) [8]. The reversed items are normalized.

Characteristics of the interaction
Vibration: Have you felt vibrations?
Path: Was it evident that was the walker to decide the path to follow?
Blocked: Have you felt to be pulled, pushed, pulled, or stuck?
If yes, "How much unpleasant/annoying . . . ?" was each feature.
Pleasantness (in using the *FriWalk*) - P
P.1: The experience with the walker was pleasant.
P.2: You are satisfied with how he did the job with the walker.
P.3: It was frustrating to carry out the task with the walker. *
Ease of learning - L
L.1: It was easy to learn to use the walker.
L.2: You could use the walker properly in a short time.
L.3: You had trouble understanding how to move around. *
Control over the *FriWalk* - C
C.1: You were sure the walker would always respond.
C.2: You had the impression you could suddenly miss the control. *
C.3: You had the impression you did not have full control. *
Adaptability of the walker - A
A.1: The walker fits well with your movements.
A.2: You had to adjust to the movements decided by the walker. *
A.3: The walker hindered/prevented your usual way of walking. *
* = Reversed

Table 5.4: Items of the questionnaire for the users' evaluation [8].

5.6 Final comments

This research proposes a path following control strategy for a robotic walker combining a mechanical braking guidance and a vibrotactile haptic guidance. The bang-bang braking guidance in isolation ensures safety and guaranteed path following accuracy, but the user comfort is penalized because of the abrupt actuation due to the simple control strategy. Conversely, vibrotactile haptic guidance in isolation results much more comfortable, but it cannot ensure an adequate path following performance with uncooperative users, so user's safety is not guaranteed. The extensive experimental study with users aged 64 to 100, comprising also psychological evaluations, shows that the orchestration of the two solutions combines the safety and path following accuracy of the braking guidance and the comfort of the haptic guidance, outperforming the other two approaches in isolation.

Future research on the control strategy will focus on the automatic adaptation to the environment of the thresholds θ_{q_1}, θ_{q_2}, $\theta^h_{q_1}$, and $\theta^h_{q_2}$ defining the integrated controller. For instance, in an open environment (with no obstacles), a large value of the braking thresholds θ_{q_1} and θ_{q_2} (i.e., more freedom for the user) is desired. Conversely, in cluttered environments, where a better path following accuracy is needed to avoid collisions, a reduced value of the braking thresholds should be automatically set. In terms of user evaluation, future research will focus on longer and more ecological interactions of people with the robotic walker. Observing how individuals relate with the robotic walker in natural environments can provide a different kind of observations which cannot be detected in a controlled laboratory scenario.

Chapter 6

Simulated passivity guidance

In this chapter we address the problem to guide the user using motorized rear wheels. Our strategy aims to simulate a passive behavior in which the forward velocity is the one imposed by the user, who hence receives the impression of controlling the motion. The result is obtained by sharing the control authority: we leave the user in control (without any actuation) when he/she follows properly the path while estimating his/her desired speed by means of the robot sensing system, while the motors kick in when the user is departing from the path, reducing the path following error while moving at the estimated desired user's speed. We offer a theoretical analysis of our strategy. The technique has been validated via extensive experimentation with a large group of older adults.

6.1 Introduction to simulated passivity

Passive robotic walkers based on actuated front steering wheels or electromagnetic brakes for differential drive are widely accepted to guide the user along the path. The main advantage is that they are intrinsically safe since they leave the responsibility of the locomotion to the user. Clearly, if the robot is equipped with thrusting motors, the potentialities of the system largely increase. For instance an actuated walker can move autonomously and pick up a user in need in a remote location. Moreover, the actuators can be used to generate emergency brakes for safety. Unfortunately, the presence of actuation disrupts the system passivity, with potential safety problems whenever the actuated motion differs from the motion desired by the user (for instance the user is pulled). To ensure that the robot moves as the user requires, direct or indirect user interfaces can be used [77]. For the former, user commands/intentions are directly communicated to the device through joysticks [75], force sensors [42, 71, 121], turn buttons, and voice commands/navigator support [63]. Indirect interfaces recognizes user's movement and/or intent without requiring her/his input. For instance, the JAIST walker guesses the user's intention using laser scanned shin positions [65]. In [112] a depth camera is used to track

the user limbs; in [55] current sensors and wheel rotational encoders are used to estimate the user applied forces.

If the walker is active, i.e., propelled by motors, we propose the use of indirect interfaces to preserve a safe and intuitive behavior of the system without additional hardware, using *simulated passivity* with an actuated device to follow the desired path, i.e., to solve problem (4.4). The passive behavior is simulated by means of alternating phases when the user is in control and his/her forward speed is estimated, with phases in which the robot is in control and a slight braking steers it towards the desired path, if needed. As a result, the user receives the impression that he/she is moving at his/her desired pace as with a passive device (hence the name simulated passivity).

With reference to Section 2.3, the control authority is then shared between user and robot via a metric-based sharp allocation approach.

6.2 Strategy overview

The vehicle is modeled as a unicycle-like robot (2.1), i.e.,

$$\begin{cases} \dot{x} &= v\cos\theta, \\ \dot{y} &= v\sin\theta, \\ \dot{\theta} &= \omega, \end{cases}$$

where the control inputs are the linear velocity v and the angular velocity ω, computed on the basis of the motor velocities (2.2), i.e.,

$$\begin{aligned} v &= \frac{r(\omega_R + \omega_L)}{2}, \\ \omega &= \frac{r(\omega_R - \omega_L)}{b}, \end{aligned}$$

where $b > 0$ is the axle length and $r > 0$ is the wheel radius. The path following problem is described using a dynamic Frenet frame (2.10)

$$\begin{cases} \dot{l}_x &= -\dot{s}(1 - c(s)l_y) + v\cos\tilde{\theta}, \\ \dot{l}_y &= -c(s)\,\dot{s}\,l_x + v\sin\tilde{\theta}, \\ \dot{\tilde{\theta}} &= \omega - c(s)\dot{s}. \end{cases}$$

The relaxed path following problem (4.4),

$$\begin{aligned} \lim_{t\to+\infty} |l_x(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |l_y(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |\tilde{\theta}| &\leq \tilde{\theta}_\infty, \end{aligned}$$

has to be solved using the available actuators, i.e., the rear motors. The motors are controlled to impose the wheel velocities ω_R and ω_L to the right and to the left wheel, respectively. According to (2.2), whenever the wheel velocities ω_R and ω_L are chosen, the vehicle velocities v and ω are defined as well. While the vehicle angular velocity ω can be chosen to control the yaw θ to approach and follow the path, the forward velocity v must be chosen by the user. In fact, in assistive robotics, because of user balance issues, it is extremely important that the vehicle does not pull the assisted person (i.e., by moving at a forward velocity larger than the one of the user). A possible way to face this issue is the use of *passive* robots that by definition do not have the authority on the vehicle forward velocity v. In this work, instead, the robot is *active*, hence the motor velocities in (2.2) are used as input, therefore, we propose to *simulate* the passivity of the vehicle by sharing the control authority between the user and the robot by alternating the following two working modes as shown in Figure 6.1:

1. **Robot in control**: The control authority is given to the robot. The wheel velocities ω_R and ω_L are controlled and the forward velocity $v = v^\star$ and the angular velocity $\omega = \omega^\star$ are imposed to the vehicle as in (2.2).

2. **User in control**: The control authority is given to the user. The motors are not activated, hence the vehicle is totally passive. Consequently, the vehicle velocities $v = v_{\text{user}}$ and $\omega = \omega_{\text{user}}$ are completely determined by the user and measured by the vehicle sensors, e.g., wheel encoders.

The passive behavior in robot in control mode is here simulated by imposing a controlled velocity $v^\star$ close (or even equal) to v_{user}, estimated in the user in control mode. This way, the user feels in control of the vehicle forward motion as if the robot were passive. As a consequence, to ensure that the path following requirements (4.4) are satisfied, in the robot in control mode only the angular velocity $\omega^\star$ can be freely determined.

The switching strategy between the two modes in Figure 6.1 is based on the following rationale. Clearly, the control authority is given to the user if the vehicle is in a safe position (i.e., close to the path), otherwise to the robot. However, a further condition for the transition is needed. To ensure that the user's speed is measured sufficiently often, the robot in control mode cannot last for too long, while the user in control mode has to last for a sufficiently long time.

Therefore, the overall controller implementing simulated passivity via authority-sharing is then composed by two ingredients that will be formalized in the following sections:

1. a path following control law solving the relaxed path following (4.4) and simulating a passive robot, i.e., suitably computing the forward velocity $v^\star$ given the desired user velocity v_{user}, to be applied in the robot in control mode;

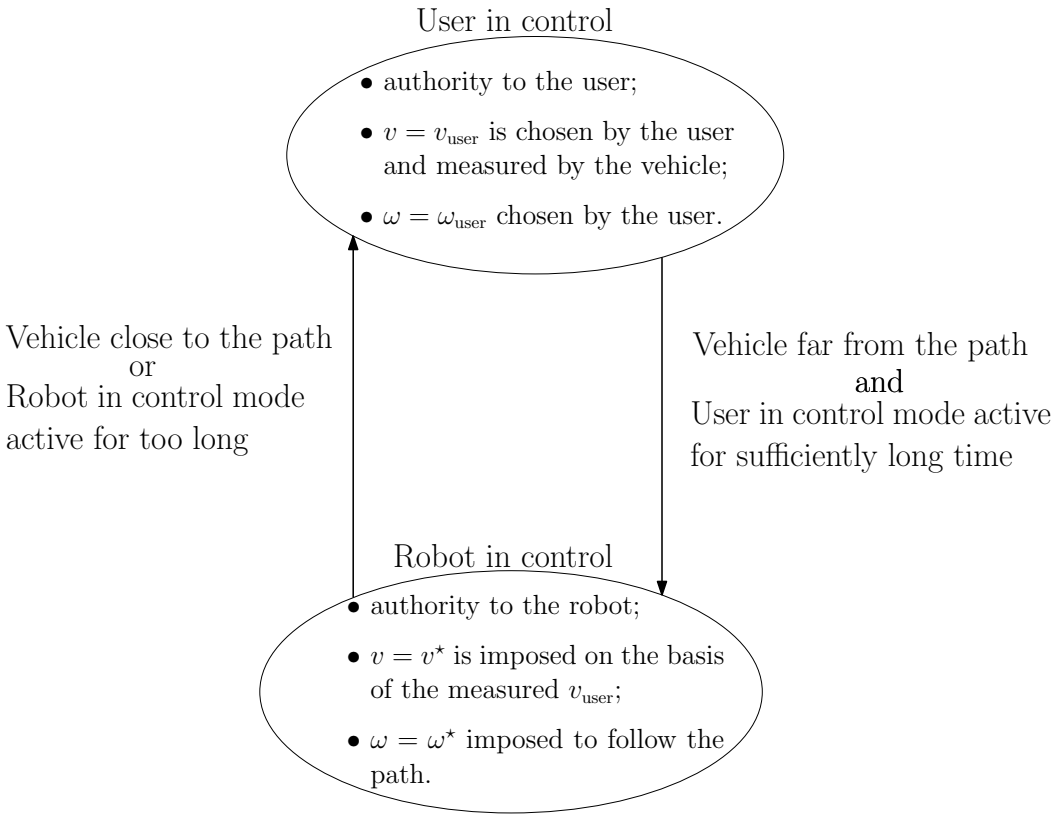

Figure 6.1: Simulation of passivity via authority-sharing [9].

2. a switching strategy between the two modes based on the user behavior and implementing the simulated passivity via authority-sharing paradigm.

6.3 Forward velocity selection to simulate passivity

6.3.1 Velocity projection

Whenever the angular velocity satisfies $\omega \neq 0$, one of the two wheels has a larger velocity than the vehicle reference point velocity v (according to Equation (2.2)). For instance, if the vehicle turns right, the left wheel is faster than both the right wheel and the vehicle reference point (having velocity v). Therefore, since the walker handles are approximately located above the rear wheels and even if the applied controlled velocity $v^\star \leq v_{\text{user}}$, the user may feel to be pulled by the fastest wheel. Hence, we impose that the fastest point of the vehicle has a forward velocity equal to v_{user}. In particular, if the requested angular velocity is positive, i.e., $\omega^\star > 0$, the vehicle turns left and the right wheel, the fastest one, is set to $\omega_R = v_{\text{user}}/r$.

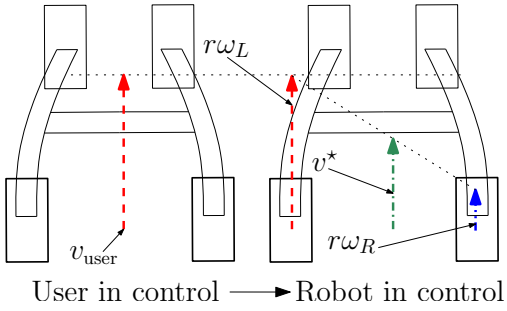

Figure 6.2: Computation of the vehicle velocities when the robot in control mode is enabled and the vehicle has to turn right [9].

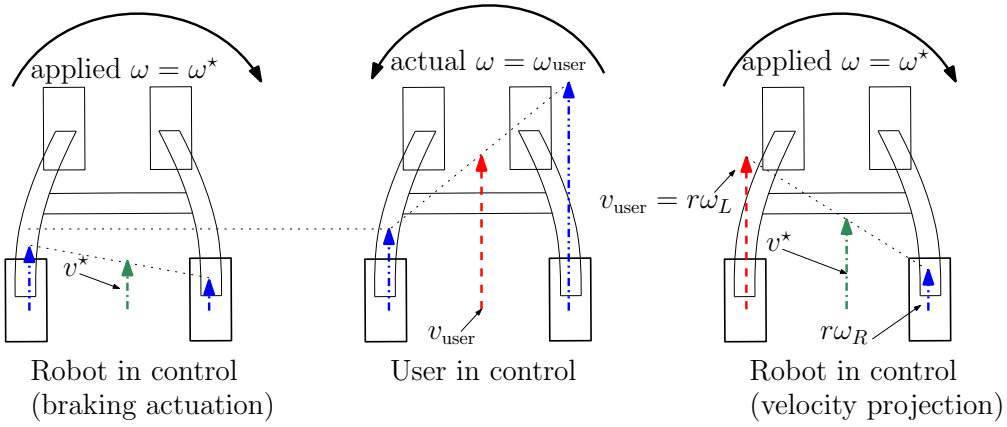

Figure 6.3: Difference between velocity projection and braking actuation for uncooperative users [9].

According to (2.2), we finally get

$$v^\star = v_{\text{user}} - \omega^\star \frac{d}{2},$$
$$\omega_L = \frac{v_{\text{user}}}{r} - \omega^\star \frac{d}{r}.$$

The case of $\omega^\star < 0$ is homologous. A compact formula to describe this strategy is $v^\star = v_{\text{user}} - |\omega^\star| \frac{d}{2}$. An example of the proposed algorithm is in Figure 6.2.

6.3.2 Braking actuation

Since the motors directly command the wheel velocities, projecting the user's velocity v_{user} on the fastest wheel might still generate discomfort if the correction that the robot has to apply is relevant. Consider for example Figure 6.3. In the user in control mode (center of the figure) the user is steering the vehicle left. Suppose

that, when the robot in control mode is enabled, the vehicle has to turn right. If the velocity projection strategy is applied, the velocity of the left wheel may increment and still compromise the user's balance if the difference between the angular velocities is relevant (right part of Figure 6.3). To avoid such a condition, the vehicle is braked by setting $v^\star = \alpha v_{\text{user}}$, where $\alpha < 1$ in order to ensure a braking action and avoid that the user is pulled by the left handle (left part of Figure 6.3).

6.3.3 Choice of the forward velocity

The rationale of the forward velocity $v^\star$ choice in the robot in control mode is the following. If the robot intervention considerably varies the vehicle angular velocity ω, the braking actuation method is applied to guarantee the user's safety. Conversely, if the robot can apply a small correction only (i.e., the required angular velocity $\omega^\star$ is close to the actual ω), the velocity projection method is applied to improve the user's comfort. Overall, the forward velocity simulating passivity applied in the robot in control mode is

$$v^\star = \begin{cases} v_{\text{user}} - |\omega^\star|\dfrac{d}{2}, & \text{when } |\omega - \omega^\star| \leq \Omega, \\ \alpha v_{\text{user}}, & \text{when } |\omega - \omega^\star| > \Omega, \end{cases} \tag{6.1}$$

where $\Omega > 0$ and α are two comfort parameters to be tailored on the specific user's requirements.

6.4 Attitude-based passive path following controller

The path following problem is solved by stabilizing the attitude error $e_\theta = \tilde{\theta} - \delta(l_y)$. In this section we develop a stabilizing angular velocity law $\omega = \omega^\star$ working for every forward velocity v. This way, computed the forward velocity $v^\star$ applied in the robot in control mode on the basis of the user's forward velocity v_{user}, it is always possible to apply an angular velocity $\omega = \omega^\star$ in the robot in control mode steering the user towards the path.

Theorem 5 (Angular velocity attitude stabilization)**.** *Consider the vehicle kinematic model* (2.10)*. Under Assumption 1, the control law*

$$\omega = \omega^\star = v\left(\gamma - \kappa\left(\tilde{\theta} - \delta(l_y)\right)\right), \tag{6.2}$$

where $\kappa > 0$ is a constant gain and

$$\begin{aligned} \gamma &= c(s)\dot{\xi} + \left(-c(s)\,\dot{\xi}\,l_x + \sin\tilde{\theta}\right)\frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y), \\ \dot{s} &= v\dot{\xi}, \\ \dot{\xi} &= \cos(\tilde{\theta}) + \kappa_x l_x, \end{aligned}$$

with $\kappa_x > 0$, ensures that the attitude error $e_\theta = \tilde{\theta} - \delta(l_y)$ converges to zero.

Proof of Theorem 5. Consider the Lyapunov function

$$V_\theta = \frac{1}{2} e_\theta^2,$$

whose derivative along the solutions is

$$\dot{V}_\theta = e_\theta \dot{e}_\theta = e_\theta \left(\omega - c(s)\dot{s} - \dot{l}_y \frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y) \right) = e_\theta \left(\omega - v\gamma \right).$$

By imposing $\omega = \omega^\star$ in (6.2), and recalling that $v > 0$ in finite time by Assumption 1, we get

$$\dot{V}_\theta = -v\kappa e_\theta^2 < 0 \quad \forall e_\theta \neq 0,$$

that ensures the convergence of e_θ to zero. □

Notice from this proof that the Lyapunov function remains constant for zero velocity, i.e., $v = 0$, confirming the intuition that the path following error should not vary for a still vehicle. Notice that this is ensured by a control angular velocity ω in (6.2) *linear* with respect to the vehicle forward velocity v. This way, when the vehicle does not advance, the angular velocity is zero. This choice is not mandatory in terms of control design. Indeed, a choice $\omega = -\kappa e_\theta$ is sufficient to stabilize the attitude error e_θ in a still vehicle (i.e., if $v = 0$). Commanding a nonzero angular velocity to a still unicycle-like robot differentially driven is definitely possible by imposing opposite rotational velocities on the wheels (see (2.2)). For instance, this is obtained using the control angular velocity in [106]. However, experimental evidence shows that the linear dependence (6.2) is preferred by the users, who generally want a vehicle completely still (with both linear and angular velocities equal to zero) whenever they stop walking. This way, the vehicle rotates (i.e., it has an angular velocity $\omega \neq 0$) only when moving forward, exhibiting a behavior necessary for a car-like robot instead of a unicycle-like one. This preference is totally in accordance with the experimental results in [14], showing the human comfort is maximize by a car-like motion instead of a unicycle-like one. It is anticipated that, if this linear dependence between the velocities is not respected when the walker uses actuated front wheels, a singularity totally compromising the performance may arise (see Section 7.6.1).

6.5 Simulated passivity via authority-sharing

The overall vehicle passive behavior is simulated by sharing the control authority by switching between the user in control mode and the robot in control mode, as depicted in Figure 6.1. The switching strategy is synthesized with a synergistic use of two different ideas: a time based and a behavioral based approaches. More precisely, in the user in control mode, the vehicle behaves passively and estimates the user's velocity v_{user} for a time window T_U. In the robot in control mode, the

motors impose the velocities $v = v^\star$ and $\omega = \omega^\star$ for a maximum time window of T_R: if the user is autonomously following the path, the control authority is given back to the user. This complex switching strategy is formalized using hybrid system theory [40].

6.5.1 Behavioral authority-sharing

When the robot in control mode is active but the user is autonomously following the path, i.e., when the path following errors are limited, the control authority is given back to the user. This is implemented by designing an hysteresis mechanism to the control law $\omega = \omega^\star$. The mathematical formulation uses the hybrid dynamics of a logic variable $q \in \{0, 1\}$. When the vehicle is far from the path, we set $q = 1$ and the motors are engaged with velocities described in Section 6.3.3. When the robot is close to the path, we set $q = 0$ and the motors are disengaged, i.e., totally passive walker. The switch between the two modes is activated with hysteresis on the basis of the distance from the path measured as $|e_\theta|$. In fact, because of the monotonicity property of the approaching angle $\delta(\cdot)$, it is sufficient to limit the attitude error e_θ to ensure that the path following requirements (2.11) hold. The hybrid dynamics of q is defined as in (4.8) to implement an authority-sharing based on sharp allocation

$$\begin{cases} \dot{q} &= 0, & [e_\theta, q]^T \in \mathcal{C}, \\ q^+ &= 1 - q, & [e_\theta, q]^T \in \mathcal{D}, \end{cases}$$

where $[e_\theta, q]^T$ is the overall state of the hybrid system, $\mathcal{C} := \mathcal{C}_0 \cup \mathcal{C}_1$ and $\mathcal{D} := \mathcal{D}_0 \cup \mathcal{D}_1$ are the flow and the jump set respectively, where

$$\begin{aligned} \mathcal{C}_0 &= \{|e_\theta| \leq \theta_{q_2} \ \wedge \ q = 0\}, \quad \mathcal{C}_1 = \{|e_\theta| \geq \theta_{q_1} \ \wedge \ q = 1\}, \\ \mathcal{D}_0 &= \{|e_\theta| \geq \theta_{q_2} \ \wedge \ q = 0\}, \quad \mathcal{D}_1 = \{|e_\theta| \leq \theta_{q_1} \ \wedge \ q = 1\}, \end{aligned}$$

where $\theta_{q_2} > \theta_{q_1} > 0$ are the hysteresis thresholds. Recalling (6.1) and (6.2), the actual velocities of the vehicle are then

$$\begin{aligned} v &= (1 - q)\, v_{\text{user}} + q v^\star, \\ \omega &= (1 - q)\, \omega_{\text{user}} + q \omega^\star, \end{aligned} \tag{6.3}$$

and ensure the solution of the path following problem (2.11), as stated by the following theorem.

Theorem 6 (Path following via simulated passivity)**.** *Consider the vehicle kinematic model* (2.10)*. Under the assumptions of Theorem 3, given two arbitrary non-negative constants l_∞ and $\tilde{\theta}_\infty$ in* (4.4)*, there exists an upper hysteresis threshold θ_{q_2} such that the controller* (6.3) *ensures that the path following requirements* (2.11) *hold.*

Proof of Theorem 6. Consider precautionary that controller (6.3) is never active (i.e., $q = 0$) if $|e_\theta| \leq \theta_{q_2}$. Theorem 5, in combination with the hybrid map defined by (4.9), ensures that $|e_\theta|$ enters in finite time and remains inside the region $|e_\theta| \leq \theta_{q_2}$. Then, since θ_{q_2} acts as upper limit of the attitude error e_θ, given the tolerated errors l_∞ and $\tilde{\theta}_\infty$, it is always possible by Theorem 3 to find an upper hysteresis threshold θ_{q_2} such that the path following conditions (2.11) hold. □

6.5.2 Time based authority-sharing

We introduce an additional logic variable p to describe the robot in control mode. When $p = 0$, the user in control mode is active, while when $p = 1$ the robot in control mode *can be activated* on the basis of the user's behavior logic variable q, described previously. Intuitively, p determines if the authority *can be given* to the robot, while q determines if the authority *is required* by the robot. To model the activation time of the two control modes, we introduce two additional hybrid states τ_U and τ_R acting as timers. The timer of the user in control mode has hybrid dynamics

$$\begin{cases} \dot{\tau}_U &= 1 - p, & \tau_U \leq T_U, \\ \dot{p} &= 0, & p \in \{0, 1\}, \\ \dot{h} &= 0, & h \in \{0, 1\}, \end{cases} \tag{6.4}$$

$$\begin{cases} \tau_U^+ &= 0, & \tau_U = T_U, \\ p^+ &= \min(q, 1 - p), & p \in \{0, 1\}, \\ h^+ &= \max\left(\text{sign}\left(\Omega - |\omega - \omega^\star|\right), 0\right), & h \in \{0, 1\}, \end{cases} \tag{6.5}$$

where the logic variable $h \in \{0, 1\}$ determines how the forward velocity should be computed, according to (6.1), i.e.,

$$v^\star = h\left(v_{\text{user}} - |\omega^\star|\frac{d}{2}\right) + (1 - h)\alpha v_{\text{user}}. \tag{6.6}$$

The discrete dynamics $p^+ = \min(q, 1 - p)$ in (6.5) ensures that the robot in control mode (i.e., the jump of p to 1) is activated when $\tau_U = T_U$ only if the user is not following the path, i.e., $q = 1$. The timer of the robot in control mode has hybrid dynamics

$$\begin{cases} \dot{\tau}_R &= p, & \tau_R \leq T_R, \\ \dot{p} &= 0, & p \in \{0, 1\}. \end{cases} \tag{6.7}$$

$$\begin{cases} \tau_R^+ &= 0, & \tau_R = T_R, \\ p^+ &= 1 - p, & p \in \{0, 1\}. \end{cases} \tag{6.8}$$

Notice that, when $p = 1$, the continuous dynamics in (6.7) is design to increment the timer τ_R up to $\tau_R = T_R$. Then τ_R is reset to zero by the discrete dynamics

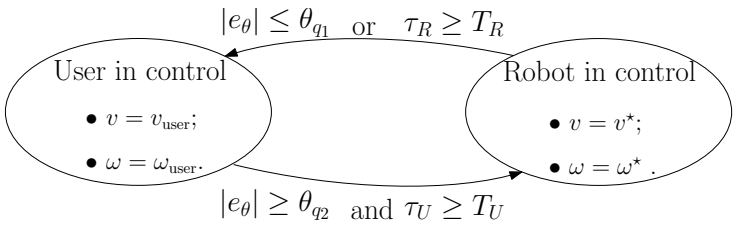

Figure 6.4: Simulated passivity as hybrid automaton.

in (6.8) and p jumps to 0 activating the dynamics (6.4) and (6.5). Finally, to ensure that the user in control mode immediately restarts when q jumps from 1 to 0, the discrete dynamics (4.8) is modified as

$$\begin{cases} \dot{q} &= 0, \\ \dot{\tau}_R &= p, \\ \dot{p} &= 0, \end{cases} \qquad \begin{matrix} [e_\theta, q]^T \in \mathcal{C}, \\ [\tau_R, p]^T \in \mathbb{R}^2. \end{matrix} \tag{6.9}$$

$$\begin{cases} q^+ &= 1 - q, \\ \tau_R^+ &= (1-q)\tau_R, \\ p^+ &= (1-q)p, \end{cases} \qquad \begin{matrix} [e_\theta, q]^T \in \mathcal{D}, \\ [\tau_R, p]^T \in \mathbb{R}^2. \end{matrix} \tag{6.10}$$

The result of the hybrid controller described in (6.4), (6.5), (6.7), (6.8), (6.9) and (6.10) is succinctly

$$\begin{aligned} v &= \begin{cases} v_{\text{user}} & q = 0 \text{ or } p = 0, \\ v^\star & \text{otherwise}, \end{cases} \\ \omega &= \begin{cases} \omega_{\text{user}} & q = 0 \text{ or } p = 0, \\ \omega^\star & \text{otherwise}. \end{cases} \end{aligned} \tag{6.11}$$

A representation of the proposed control algorithm as hybrid automaton is reported in Figure 6.4.

6.5.3 Activation time tuning

The lengths of the activation times T_R and T_U clearly influence the user's comfort and the path following performance and, hence, can be used as tuning parameters. Nevertheless, the effective values of these parameters are constrained. Indeed, the larger T_R, the better are the path following performance (authority is shifted to the robot). However, since in the robot in control mode the user can not modify the vehicle velocities, T_R is a precautionary upper bound. Conversely, the larger T_U, the larger is the user's comfort. However, the path following error is larger with uncooperative users. As a consequence, to ensure that the path following

requirements (2.11) hold even with an uncooperative user, the user in control mode should last for a small amount of time. According to Theorem 6, the performance is guaranteed as long as $|e_\theta| < \theta_{q_2}$ remains valid. The perturbation induced by the user can be quantified as

$$|\dot{e}_\theta^{\text{user}}| \leq |\omega_{\text{user}}| + |v_{\text{user}}\gamma|,$$

hence the maximum increment of attitude error that the user can generate is $\Delta e_\theta = \int_{t_0}^{t_0+T_U} |\dot{e}_\theta^{\text{user}}| \mathrm{d}\tau$, where t_0 is the time instant in which the robot in control mode is activated. Clearly, if the user velocities are large, the time window T_U has to be small to limit Δe_θ. Therefore, an adaptive discrete dynamics may be added to (6.8) and (6.10) (i.e., at the end of each user in control mode), reported next

$$T_U^+ = \min\left(\frac{a_1}{a_2|\hat{v}| + a_3|\hat{\omega}|}, T_U^{\max}\right), \quad T_U \in \mathbb{R}. \tag{6.12}$$

$\hat{v}$ and $\hat{\omega}$ are the measured vehicle velocities while a_1, a_2, and a_3 are constant parameters to be tuned on the specific user. $T_U^{\max} > 0$ is the upper bound to T_U in the user in control mode. The implementation of this $\min(\cdot)$ function is needed to avoid the undesired condition $T_U \to \infty$ whenever the vehicle is still (i.e. $\hat{v} = \hat{\omega} = 0$). Notice that these parameters could be set automatically in an adaptive tuning algorithm or by a GUI-based question & answers with the user. In both cases, the system should be used for a longer time than the one available for the field tests reported in Section 6.7. Therefore, those parameters have been set via experiments with young testers (omitted for brevity) to guarantee an average behavior between aggressive and loose control.

6.6 Simulation results

To firstly show the performance of the proposed solution by simulations, we considered a user moving at a constant forward speed of $v_{\text{user}} = 1$ m/s and starting from a generic initial condition $[x(0), y(0), \theta(0)]^T = [5, -5, 0]^T$. To show the effectiveness of the building blocks separately, we first consider the path following controller (6.2) acting all the time in isolation (i.e., robot fully controlled in feedback) and assuming the gain $\kappa = 4$ and the approaching angle $\delta(\cdot)$ chosen as in (4.10). The controller (6.2) solves the path following with zero steady state tracking error, as depicted in Figure 7.7 for the trajectory labeled "Controller 1". By adding the hysteresis mechanism presented in (6.3), with hysteresis thresholds $\theta_{q_2} = 20^\circ$ and $\theta_{q_1} = 5^\circ$, the path is instead followed with a given tolerance (see the trajectory labeled "Controller 2" in Figure 7.7). Notice that in this case ω_{user} is assumed equal to the last angular velocity $\omega^\star$ to mimic a user's behavior observed in the experiments. Finally, to promote the interaction and cooperation between the user and the robot, the final passive controller based on active actuators of (6.11) is simulated with $T_R = T_U = 1$ s and $\alpha = 1$ for the dynamics (6.5). The trajectory, dubbed "Controller 3" in Figure 7.7, is followed with a higher tolerance, as expected.

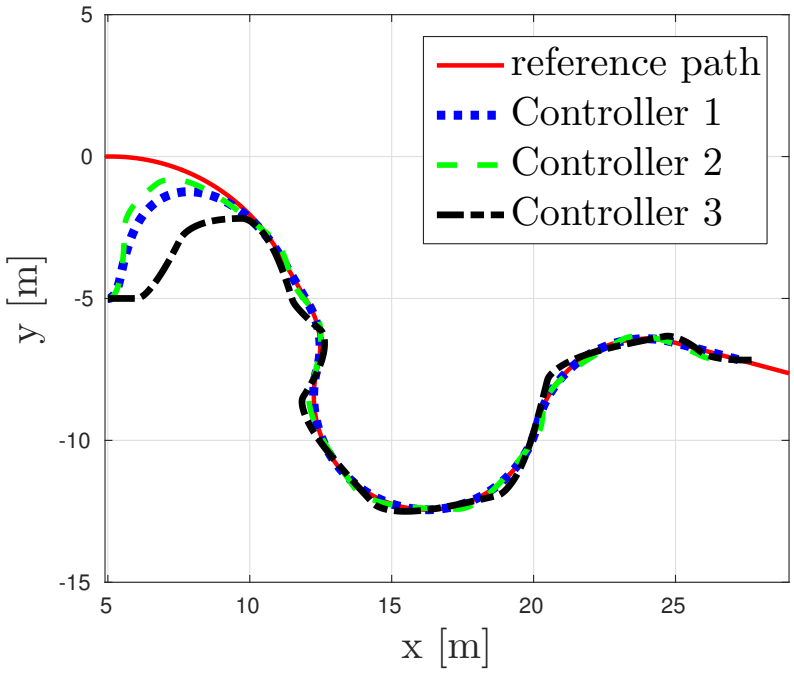

Figure 6.5: Trajectories generated by the path following controller (6.2) (Controller 1), the hysteresis controller (6.3) (Controller 2) and the final passive controller (6.11) (Controller 3) [6].

6.7 Experimental results

In this section we present the experimental results of the proposed approach. Two studies were conducted in which the older adult participants (details can be found in Table 6.1) completed different paths using the *FriWalk* in one laboratory of the University of Trento. In the first study (with 4 males, 10 females, aging between 65 and 75 years old), the participants were asked just to travel along a couple of paths, while in the second study (with 6 males, 9 females, aging between 64 and 100 years old) a more extensive study, with more than eight paths for each participants were considered. Some of the participants usually use walking aids, such as crutches and/or a walker (28.6% of *Study 1* and 43.8% of *Study 2*). Participants were contacted through the Municipality of Pergine Valsugana and the senior center "Sempreverde" of Mattarello (both in the Trento province) and invited to participate. They were informed that data collection and that all information provided are covered by the ethical rules conceived for the ACANTO project [1] and that they could quit the experiment at anytime. Once consent was obtained they were invited to perform the tasks with the *FriWalk*. Before starting, an experimenter explained the features of the robotic walker and its motion mode. All participants completed a first trial (which was common for everybody) to take confidence with the robot

	Males	Females	Older	Younger	Walking Problems
Study 1	4	10	75	65	4 (28.6%)
Study 2	6	9	100	64	7 (43.8%)

Table 6.1: Cohort characteristic [9].

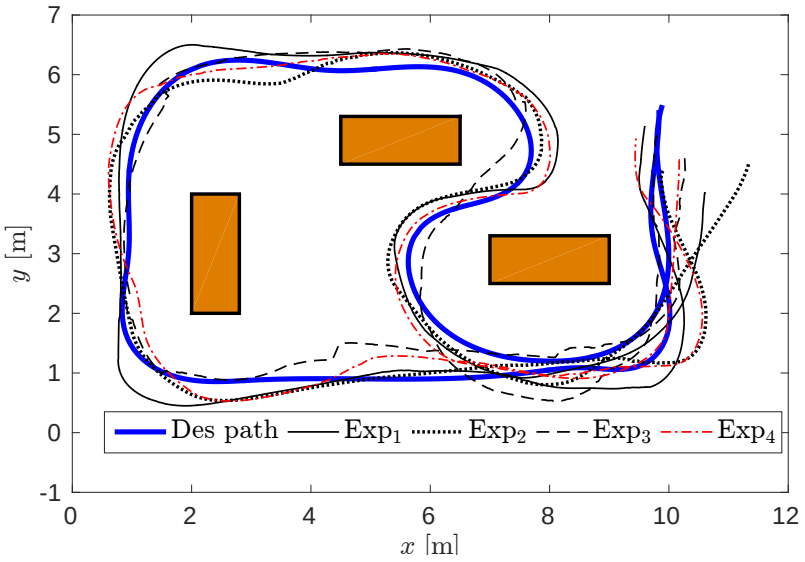

Figure 6.6: Experimental trajectories for four participants along a randomly selected path (solid thick line). The rectangles represent the obstacles (i.e., tables) in the environment [9].

walker and its movements. More than ten different paths, starting and ending in the same home position, were randomly chosen for each participant, that completed at least one of them. In the laboratory arena, three tables were placed to emulate an actual indoor environment (see the rectangular obstacles in Figure 6.6).

6.7.1 Quantitative analysis

We first present the quantitative analysis of the experimental results. The controller parameters (Table 6.2) adopted in the experiments are: $\alpha = 0.37$ and $\Omega = 0.08$ [rad/s] for (6.1), the hysteresis thresholds are set to $\theta_{q_1} = 8°$ and $\theta_{q_2} = 15°$, the maximum time for the robot in control mode is $T_R = 2$ s, while the parameters for the user in control mode in (6.12) are $T_U^{max} = 4$ s, $a_1 = 2$, $a_2 = 2$ and $a_3 = 1$. Finally, we select $\delta(l_y) = -\pi/2 \tanh(l_y)$. Four sample trajectories along a randomly selected path are reported in Figure 6.6. The localization is provided with an extended Kalman filter [88] fusing the encoder data and the QR codes, positioned on the

α	0.37	Ω	0.08	θ_{q_1}	8°
θ_{q_2}	15°	T_R	2 s	T_U^{max}	4 s
a_1	2	a_2	2	a_3	1

Table 6.2: Controller parameters adopted in the experiments [9].

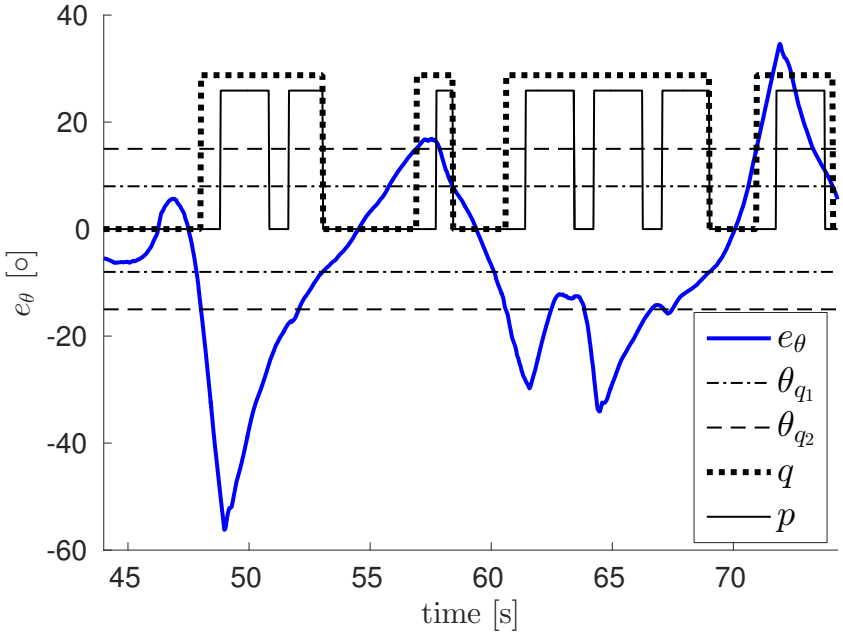

Figure 6.7: Last 30 seconds of the time evolution of the error e_θ for the Exp_1 in Figure 6.6. The evolution of the discrete hybrid variable p and q (scaled for visibility) and the controller thresholds θ_{q_1} and θ_{q_2} are also reported [9].

floor using the deployment [87] and read by the available front camera pointing downwards. It may happens that a QR code reading is missed, hence a localization jump can be detected in the estimated trajectory (see the dashed trajectory of Exp_3 in Figure 6.6). Nonetheless, the controller is able to correctly steer the user towards the desired path. Figure 6.7 and Figure 6.8 report the time evolution of the error e_θ and of wheel rotational velocity, respectively, for the last 30 seconds of the Exp_1 in Figure 6.6. It is important to recall that the user is in control when at least one of the two variables q or p is 0, which may happens, but for a limited amount of time (almost 0.8 seconds in the experiments) even if the robot is outside the hysteresis thresholds (see the portion of Figure 6.7 where $q \neq 0$ and $p = 0$). This is the essence of the time based authority sharing presented in Section 6.5.2, whereas the behavioral authority sharing (Section 6.5.1) takes place whenever $q = p = 0$. Finally, it has to be noted how the controller is very effective in controlling the error

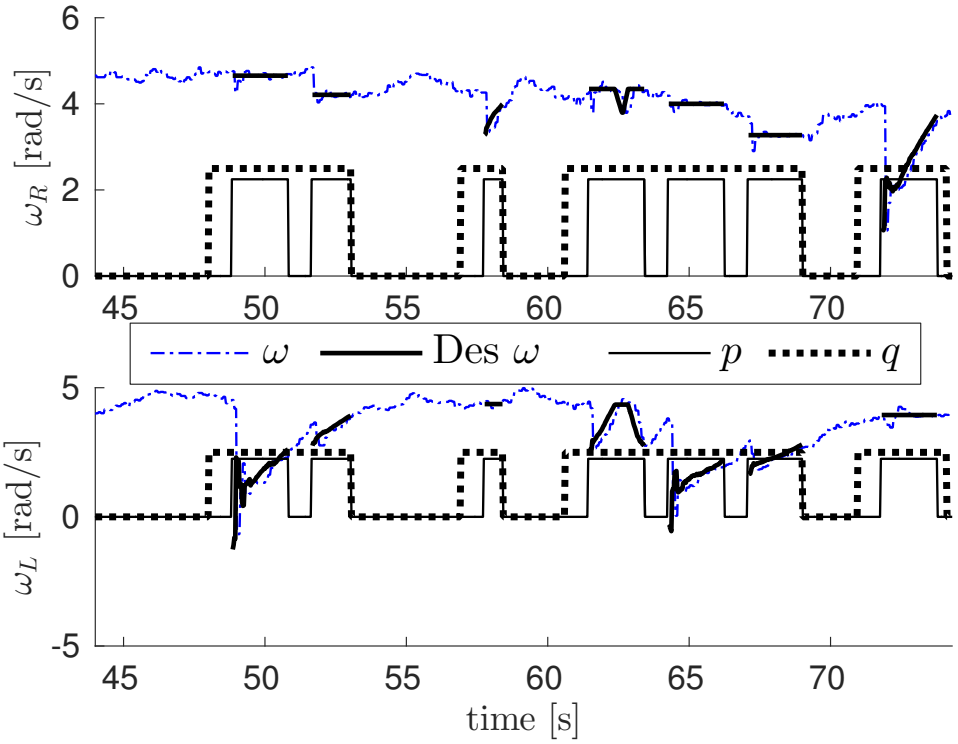

Figure 6.8: Last 30 seconds of the time evolution of the right ω_R (upper plot) and left ω_L (lower plot) angular velocities for the Exp_1 in Figure 6.6. The evolution of the discrete hybrid variable p and q (scaled for visibility) is also reported [9].

e_θ in the region of the hysteresis (Figure 6.7).

For the actuation, it is evident from Figure 6.8 when the actuation kicks in by means of the desired velocity (thick solid lines superimposed to the actual wheels velocity, represented with a dash-dotted line). It is also noticeable how the wheel dynamic and the user applied forces generate a small tracking error of the desired velocity.

6.7.2 User's evaluation

In both studies, we used a questionnaire to conduct a structured interview to collect the impressions and opinions of people who participated in the studies. After the session with the robotic walker, the participants were invited to sit next to an experimenter who conducted the structured interview reading the items of the questionnaire. The aim of the structured interview was to collect the impressions of people on the proposed control approach. To this end, we included different questions (open ended and closed ended). In the present work, we present the analysis of closed ended questions. Participants were asked to answer using yes or no and/or a

Characteristics of the interaction
Vibration: Have you felt vibrations?
Path: Was it evident that was the walker to decide the path to follow?
Blocked: Have you felt to be pulled, pushed, pulled, or stuck?
If yes, "How much unpleasant/annoying... ?" was each feature.
Pleasantness (in using the *FriWalk*) - P
P.1: The experience with the walker was pleasant.
P.2: It was frustrating to carry out the task with the walker. *
P.3: You are satisfied with how he did the job with the walker.
Ease of learning - L
L.1: It was easy to learn to use the walker.
L.2: You could use the walker properly in a short time.
L.3: You had trouble understanding how to move around. *
Control over the *FriWalk* - C
C.1: You were sure the walker would always respond.
C.2: You had the impression you could suddenly miss the control. *
C.3: You had the impression you did not have full control. *
Adaptability of the walker - A
A.1: The walker fits well with your movements.
A.2: You had to adjust to the movements decided by the walker. *
A.3: The walker hindered/prevented your usual way of walking. *
* = Reversed

Table 6.3: Items of the questionnaire for the users' evaluation [9].

5 point Likert scale (1 "not at all", 2 "a little bit", 3 "moderately", 4 "very much", 5 "extremely"). The questions concerned different features of the interaction with the robotic walker, followed by items on the pleasantness of usage, the ease of learning, the control over the robot and its adaptability. Items used for the structured interview are shown in Table 6.7.2. We first report the results of the characteristics of interaction in Table 6.4. The percentage of affirmative responses, with their relative mean **M** and standard deviations **SD** on how much annoying/disturbing were the different features of the interaction on the Likert scale, are reported.

For the other items, the results, with mean and standard deviation, are summarized in tables 6.5 and 6.6, and Figure 6.9.

Discussion The results of the studies showed an overall positive impression of the *FriWalk*. Concerning the characteristics of interaction that in both studies most of the participants were aware that the *FriWalk* decided the path to follow, whereas a low percentage of them reported they felt the vibration and had the sensation of being blocked or pushed. In any case, we observed that participants did not

	Study 1		***Study 2***	
Item	**Yes**	**M (SD)**	**Yes**	**M (SD)**
Vibration	33.3%	1.75 (0.50)	53.3%	2.00 (0.76)
Path	91.7%	1.82 (0.98)	93.3%	1.31 (0.63)
Blocked	25%	2.00 (0.00)	66.7%	1.80 (0.79)

Table 6.4: Answers on the characteristics of the interaction [9].

P - M (SD)	**L - M (SD)**	**C - M (SD)**	**A - M (SD)**
P1: 3.58 (0.79)	L1: 3.75 (0.45)	C1: 2.83 (1.34)	A1: 3.33 (1.07)
P2: 4.83 (0.58)	L2: 3.67 (0.49)	C2: 4.67 (0.89)	A2: 3.42 (1.24)
P3: 3.83 (0.72)	L3: 4.83 (0.39)	C3: 3.67 (1.23)	A3: 4.33 (0.89)

Table 6.5: Mean and standard deviation (in parenthesis) for the items used to collect the impressions on the *FriWalk* in *Study 1*.

perceive these features as disturbing or annoying, thus validating our definition of comfort. The results also showed that participants evaluated the experience as moderately pleasant and that they felt happy with their performance with the robot. Moreover, they reported they did not feel frustrated by the interaction with the walker. Importantly, from a user experience point of view, the participants reported they had the feeling they could always easily control the *FriWalk*. Finally, we found that participants had the feeling that the *FriWalk* well adapted to their speed and natural pace, so that the walker was not an obstacle to their usual way of walking.

Furthermore, it has been noted that participants showed good confidence in interpreting the walker suggestions with low path following errors. The fact that the *FriWalk* corrected the users by slightly slowing down was considered fundamental in this respect. Notice that in a few minutes participants understood the functioning of the robot and that with a clear explanation of its features and capabilities they did feel they were in control of the system.

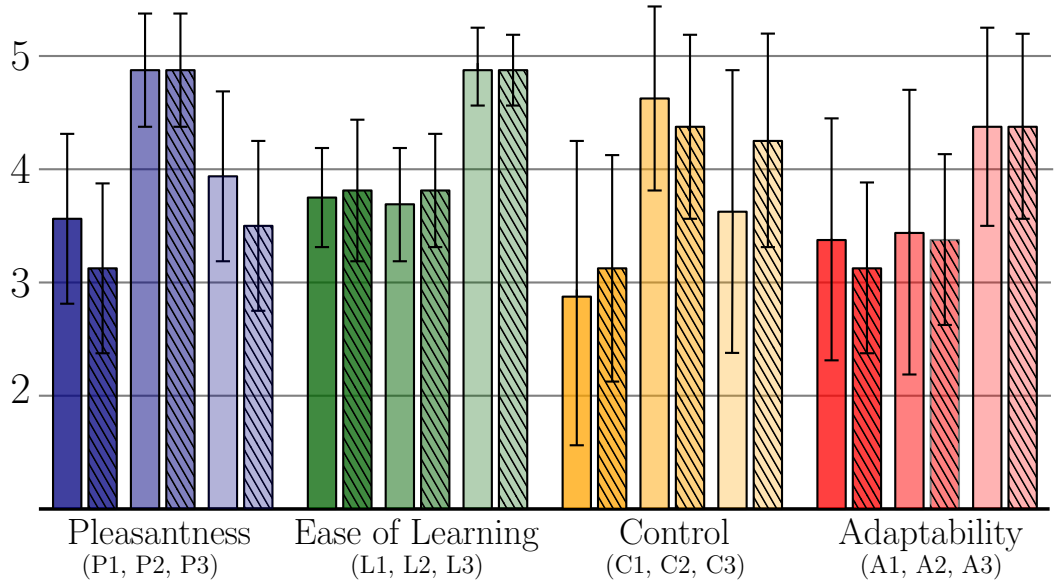

Figure 6.9: Means and standard deviations for the items on pleasantness (P1, P2, P3), ease of learning (L1, L2, L3), control perception (C1, C2, C3) and adaptability (A1, A2, A3) of the *FriWalk* in *Study 1* (solid fill bars) and *Study 2* (falling tiling pattern).

P - M (SD)	**L - M (SD)**	**C - M (SD)**	**A - M (SD)**
P1: 3.13 (0.74)	L1: 3.80 (0.68)	C1: 3.13 (0.92)	A1: 3.13 (0.74)
P2: 4.80 (0.56)	L2: 3.80 (0.56)	C2: 4.33 (0.82)	A2: 3.36 (0.74)
P3: 3.47 (0.74)	L3: 4.87 (0.35)	C3: 4.23 (0.93)	A3: 4.33 (0.82)

Table 6.6: Means and standard deviation (in parenthesis) for the items used to collect the impressions on the *FriWalk* in *Study 2*.

6.8 Final comments

"Simulated passivity" is a novel guidance approach for robotic walkers. The solution is based on alternating intervals in which the system is not engaged and the user is in control with other intervals in which the system comes into play to execute turns. The impression is that of a passive system in which the user is never "pulled" even if the turns are imposed using the motorized rear wheels. The system has been validated with a large base of senior users. We report both quantitative analysis and user's evaluation of the *FriWalk*.

From a technical perspective, future works will concentrate in changing dynamically the thresholds $\theta_{q_2} > \theta_{q_1} > 0$ according to the actual free space in front of the robot and still preserving the convergence properties. Moreover, future studies will focus on comparing different mechanical solutions and longer interactions with the robot walker using an ecological approach, and the possibility of orchestrating them with a visual feedback. Furthermore, learning algorithms to improve the user experience (i.e., tunable parameters or forward velocity adaptation) will be developed and tested.

Chapter 7

Variable stiffness handling

This chapter proposes a path following controller via authority-sharing between a human and an assistive robotic walker equipped with front steering wheels. Front steering offers several advantages for assistive robotic walker path following, such as passivity, accuracy, and comfort. The control authority is gradually shared between human and robot using a variable stiffness vehicle handling paradigm. This approach modifies the perceived physical interaction of the user with the robot on the basis of a path following metric. If the user is close to the path, he/she perceives the vehicle as compliant, having the impression than it is not actuated and he/she is free to move. The larger the distance, the stiffer the front steering system. Hence, the user perceives a vehicle gradually stiffer and stiffer, i.e., more and more difficult to steer away, as he/she gradually deviates from the path. The compliance of the steering system is generated via software by a varying-gain approach. If the walker is close to the path, a low-gain is used, so that the user inputs are dominant in the robot dynamics. Conversely, as the user departs from the path, the gains are increased to stiffen the steering system and to guarantee performance recovery and safety by feedback. We provide extensive theoretical, simulative, and experimental analysis of the control strategy, even in the presence of uncooperative users. The experimental study also proposes and validates a quantitative index of the human-robot cooperation.

7.1 Introduction to variable stiffness vehicle handling

Although guidance solutions based on rear actuation offer several advantages, such as cheapness (brakes) and possibility of autonomous motion (motorized wheels), the use of a front steering vehicle, studied in this chapter, arguably improves the system quality at the price of a small increase in the mechanical complexity. First, the walker remains passive, hence, as in the case of a braking guidance, no accidental motion of the vehicle can be generated since it is completely propelled by the user. Second, a front steering actuation allows a superior localization accuracy

since wheel slipping (and hence inaccurate odometry) is clearly more probable when the velocities of the rear wheels (which are coupled with the encoders) are controlled by actuators with faster dynamics (e.g., blocking the wheels as in Chapter 5 and in [8, 5]). Third, the car-like kinematics generates trajectories with superior perceived comfort with respect to the unicycle-like kinematics [14].

On the other hand, existing path following solutions are not readily available. Car-like path following is solved for instance by the well-known chained form coordinate transformation [98] and by Lyapunov techniques [81]. Unfortunately, these approaches make an explicit use of the forward velocity as a control input, which is not possible for the problem at hand where the robot is propelled by the user.

A possible solution would be to steer the front wheels to generate an angular velocity solving the path following problem without the need to control the forward velocity. However, this approach is, in general, singular at null forward velocity. This issue is irrelevant for fully auonomous robots, where the condition of non-zero velocity is clearly necessary for convergence to the path, then the case of zero velocity is always neglected (by assumption) in car-like control [98, 81]. However, an undesired behavior of the controller for a still vehicle cannot be neglected in assistive robotics, where the user often stops or slows down.

A further problem comes from the need of sharing the control authority with the user. A front steering system is very efficient to follow the path [12] since it can completely force the user to move in the direction imposed by the front wheels without sharing the authority. This is not the case of the braking guidance [8], where authority-sharing is needed to avoid chattering, and of the simulated passivity approach [9], where the control must be periodically given to the user to estimate his/her desired forward speed. To share the control authority, we may apply a sharp allocation based on the path following error, giving the authority to the robot only far from the path. Unfortunately, if the front actuation were activated only for large deviations, a sudden reorientation of the front wheels would stiffly happen whenever the robot takes the control, hence mining the comfort advantage of the car-like kinematics.

To avoid this nuisance and properly share the authority with the user, a more compliant actuation of the front wheels is required. A compliant behavior in human-robot physical interaction is typical of soft robotics, where the compliance ensures safety (for instance in case of a collision between a human operator and a robotic arm). Despite a more complex structure, the most promising solution in soft robotics is the use of a compliant element in the mechanics. For instance, in variable stiffness transmissions [21], the compliance can be realized with adjustable-length leaf springs or two-way air cylinders (coordinately increasing air pressure in both chambers to tune the stiffness). Unfortunately, the actuators available in the *FriWalk* are ordinary brushless motors coupled with standard transmission belts, so we generate compliance by an *active impedance by control* approach [117], where the actuators mimic a compliant behavior on the basis of the control software. In terms of control,

guaranteeing both compliance and performance requires advance strategies (e.g., based on feedforward actions [31]) for soft robots. In fact, high-gain control offers good performance but clearly stiffens the robot, while low-gain control maintains the compliance at the price of a reduced working precision. In particular, low-gain control cannot compensate disturbances in the presence of small tracking errors, since the feedback action is typically very small.

In this work, we propose a *variable stiffness vehicle handling*, implementing a *gradual* transition of the control authority from user to robot and vice versa. This approach modifies the perceived physical interaction of the user with the robot on the basis of the path following metric. If the user is close to the path, he/she perceives the vehicle as compliant, having the impression than it is not actuated and he/she is free to move. The larger the distance, the stiffer the front steering system. Hence, the user perceives a vehicle gradually stiffer and stiffer, i.e., more and more difficult to steer away, as he/she gradually deviates from the path. This way, handling the vehicle in the (safe) direction of the path (towards which the robot is compliant), is perceived much more intuitive and comfortable than a further deviation (for which the robot is stiff). The compliance of the steering system is generated by a varying-gain approach. In particular the incapability of low-gain control to compensate disturbances in the presence of small errors is used to gradually share the control authority with the user. If the walker is close to the path, a low-gain approach is used, so that the user inputs are dominant in the robot dynamics. Conversely, as the user departs from the path, the gains are increased to stiffen the steering system and to guarantee performance recovery and safety by feedback. Notice that this approach uses the compliance in a different way with respect standard soft robotics. In fact, an increasing stiffness is used to guarantee safety, since it ensures that the user is brought back in a neighborhood of the path (e.g., far from the obstacles).

7.2 Strategy overview

The assistive walker is pushed forward by the assisted person and it is equipped with two independent steering wheels. The two front steering wheels are modeled as a single virtual wheel, i.e., single track model. The model is then the differential kinematics of a front-steering rear-driven bicycle (2.3) and (2.5)

$$\begin{cases} \dot{x} &= v\cos(\theta), \\ \dot{y} &= v\sin(\theta), \\ \dot{\theta} &= \dfrac{v}{d}\tan(\varphi), \\ \dot{\varphi} &= u_v, \end{cases} \tag{7.1}$$

where the coordinates $[x, y]$ denote the position of the vehicle reference point, i.e., the mid point O_m of the rear wheel axle, with respect to the ground reference frame $\{O_w, X_w, Y_w, Z_w\}$, θ is the vehicle yaw, i.e., the orientation of the vehicle-fixed

reference frame $\{O_m, X_m, Y_m, Z_m\}$ with respect to ground, $v \geq 0$ is the forward velocity, d is the distance between O_m and the contact point of the virtual front wheel, φ is the steering angle of the virtual front wheel, and u_v is the rotational velocity of the virtual front wheel. Since the vehicle is passive, the forward velocity v is an exogenous control input determined by the user's thrust, while the steering velocity u_v is a control input.

Since the walker is equipped with two independent steering wheels, the virtual steering angle φ is related to the left and right wheel angles φ_l and φ_r to ensure a proper positioning of the instantaneous rotation center of the vehicle (see Figure 2.4). The relations between steering angles φ, φ_r, and φ_l, and steering velocities $\dot{\varphi}$, $\dot{\varphi}_r$, and $\dot{\varphi}_l$ are (2.4) and (2.6), i.e.,

$$
\begin{aligned}
y_{\text{icr}} &= \frac{d}{\tan(\varphi)}, \\
\varphi_l &= \arctan\left(\frac{d}{y_{\text{icr}} - \frac{w}{2}}\right), \\
\varphi_r &= \arctan\left(\frac{d}{y_{\text{icr}} + \frac{w}{2}}\right), \\
\dot{\varphi}_l &= \frac{d^2 \csc^2(\varphi)\dot{\varphi}}{(d\cot(\varphi) - \frac{w}{2})^2 \left(\frac{d^2}{(d\cot(\varphi) - \frac{w}{2})^2} + 1\right)}, \\
\dot{\varphi}_r &= \frac{d^2 \csc^2(\varphi)\dot{\varphi}}{(d\cot(\varphi) + \frac{w}{2})^2 \left(\frac{d^2}{(d\cot(\varphi) + \frac{w}{2})^2} + 1\right)}.
\end{aligned}
$$

where y_{icr} is the Y coordinate of the vehicle ICR in the vehicle frame $\{O_m, X_m, Y_m, Z_m\}$ and $w > 0$ is the distance between the contact points of the front steering wheels.

The relaxed path following problem is represented using a dynamic Frenet reference frame (2.10)

$$
\begin{cases}
\dot{l}_x &= -\dot{s}(1 - c(s)l_y) + v\cos\tilde{\theta}, \\
\dot{l}_y &= -c(s)\,\dot{s}\,l_x + v\sin\tilde{\theta}, \\
\dot{\tilde{\theta}} &= \omega - c(s)\dot{s}, \\
\dot{\varphi} &= u_v,
\end{cases}
$$

where the last equation $\dot{\varphi} = u_v$ is added to (2.10) to recall that the car-like model (7.1) has four states with respect to the unicycle model (2.1). The state of the vehicle can be equivalently represented as $\chi = [l_x, l_y, \tilde{\theta}, \varphi]^\top$ or $\bar{\chi} = [x, y, \theta, \varphi]^\top$.

We need to design a steering controller solving the relaxed path following prob-

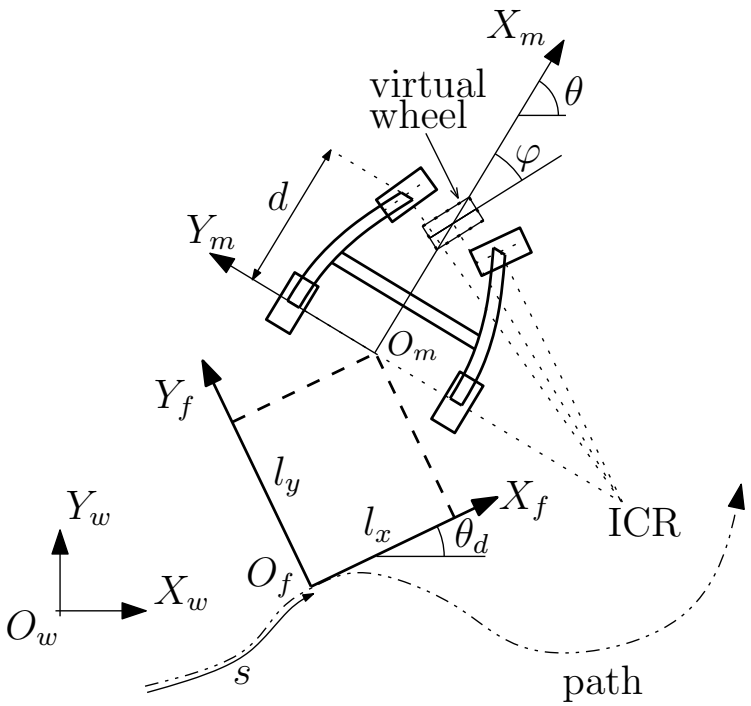

Figure 7.1: Adopted reference frames [13].

lem (4.4)

$$\lim_{t\to+\infty} |l_x(t)| \leq l_\infty,$$
$$\lim_{t\to+\infty} |l_y(t)| \leq l_\infty,$$
$$\lim_{t\to+\infty} |\tilde{\theta}(t)| \leq \tilde{\theta}_\infty.$$

The purely geometric requirements (4.4) are clearly not sufficient to guarantee that the control authority is shared with a gradual allocation mechanism between the robot and the user, which is instead one of the main goals of this work. This sharing policy is reproduced using a *variable stiffness vehicle handling*: the vehicle has a larger control authority when it is far from the path and hence it is physically perceived "stiff" by the user; conversely, the vehicle has a smaller control authority when it is close the path, i.e., the user perceives a compliant vehicle as if it were not actuated. This effect is viewed as a virtual spring-damper system acting on the steering wheels, where the spring stiffness (hence the term "stiff" vehicle) and the damping coefficient vary with a suitable measure of the distance from the path, considering both geometric distance and relative orientation (see Section 4.2.1). Notice that the classical car-like model (7.1) is based on a first order dynamics $\dot{\varphi} = u_v$, i.e., it is supposed that the velocity is commanded by the motor with

negligible settling time. This working mode of the motor is denoted as *velocity-tracking*. Although this hypothesis is reasonable and widely accepted [103], we need to consider a second order dynamics to implement the variable stiffness paradigm. This requires to use the motors in *torque-tacking* mode. The motor dynamics and the corresponding working modes of its controller are described in the following section.

7.2.1 Low-level motor control

Each wheel of the walker is coupled with a permanent magnet synchronous motor (PMSM) via a gear box having a gear ratio n_g. A PMSM can be used both to command the steering velocity u_v (velocity-tracking mode) and to apply a torque to the steering wheels (torque-tracking mode), as explained in the following.

The dynamic model of a PMSM in the dq-rotor reference frame can be expressed as in [61]. The wheel is considered as a passive rotating system. The dynamic model of each wheel-motor system is expressed as

$$\begin{cases} \dfrac{\mathrm{d}i_d}{\mathrm{d}t} &= \dfrac{1}{L_d}\left(-R_s i_d + n_g n_p L_q i_q v_w + \nu_d\right), \\ \dfrac{\mathrm{d}i_q}{\mathrm{d}t} &= \dfrac{1}{L_q}\left(-R_s i_q - n_g n_p L_d i_d v_w - n_g n_p K_e v_w + \nu_q\right), \\ \dfrac{\mathrm{d}v_w}{\mathrm{d}t} &= \dfrac{3 n_g n_p}{2 J_{eq}}\left(K_e i_q + (L_d - L_q) i_d i_q\right), \end{cases} \tag{7.2}$$

where i_d and i_q are the direct and quadrature current components, ν_d and ν_q are the direct and quadrature input voltage components, v_w is the steering velocity, R_s is the phase resistance, L_d and L_q are the phase inductances along the direct and quadrature axes respectively, n_p is the number of permanent magnet pole pairs, K_e is the back electromotive force (EMF), and $J_{eq} = J_w + n_g^2 J_m$ is the equivalent inertia moment of the motor-wheel system, where J_m and J_w are the inertia moments of motor and of the wheel (around the steering axis), respectively.

We adopt the field oriented control strategy [118, 67] for system (7.2). In particular it is possible to design the two control inputs ν_d and ν_q, by means of a state feedback, as

$$\nu_d = -n_g n_p L_q i_q v_w + \nu'_d, \tag{7.3}$$

$$\nu_q = n_g n_p L_d i_d v_w + n_g n_p K_e v_w + \nu'_q, \tag{7.4}$$

where ν'_d and ν'_q are two auxiliary control input designed in order to assign the

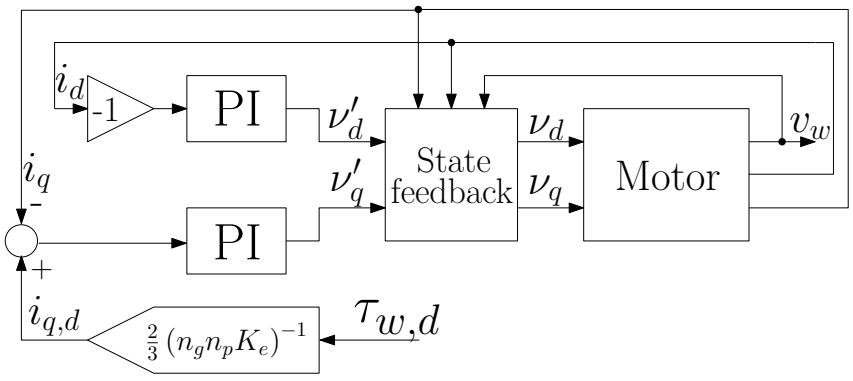

Figure 7.2: Block diagram of the torque control [13].

desired behavior to the system

$$\begin{cases} \dfrac{\mathrm{d}i_d}{\mathrm{d}t} &= \dfrac{1}{L_d}\left(-R_s i_d + \nu'_d\right), \\ \dfrac{\mathrm{d}i_q}{\mathrm{d}t} &= \dfrac{1}{L_q}\left(-R_s i_q + \nu'_q\right), \\ \dfrac{\mathrm{d}v_w}{\mathrm{d}t} &= \dfrac{3n_g n_p}{2J_{eq}}\left(K_e i_q + (L_d - L_q) i_d i_q\right). \end{cases} \tag{7.5}$$

By means of the state-feedback (7.3) and (7.4) the dynamics of the stator currents are made linear and decoupled between each other: a variation of ν'_d produces only a variation of i_d and a variation of ν'_q produces only a variation of i_q. Now we can act on the input ν'_d to force the current i_d to zero by means of a PI controller. In such a case, after a transient in which the i_d goes to zero, the nonlinear term $i_d i_q$ in the speed equation is also zero and the system can be viewed as a linear system, in which the speed, depends only on the current i_q which can be controlled by the input ν'_q.

In other words, it is possible to design the speed controller (motor in velocity-tracking mode) or the torque controller (motor in torque-tracking mode) by the dynamics

$$\begin{cases} \dfrac{\mathrm{d}i_q}{\mathrm{d}t} &= \dfrac{1}{L_q}\left(-R_s i_q + \nu'_q\right), \\ \dfrac{\mathrm{d}v_w}{\mathrm{d}t} &= \dfrac{3n_g n_p}{2J_{eq}} K_e i_q, \end{cases} \tag{7.6}$$

and by recalling that the torque τ_w applied by the motor on the steering wheel is given by

$$\tau_w = \frac{3}{2} n_g n_p K_e i_q. \tag{7.7}$$

Summarizing, the two working modes are obtained as follows.

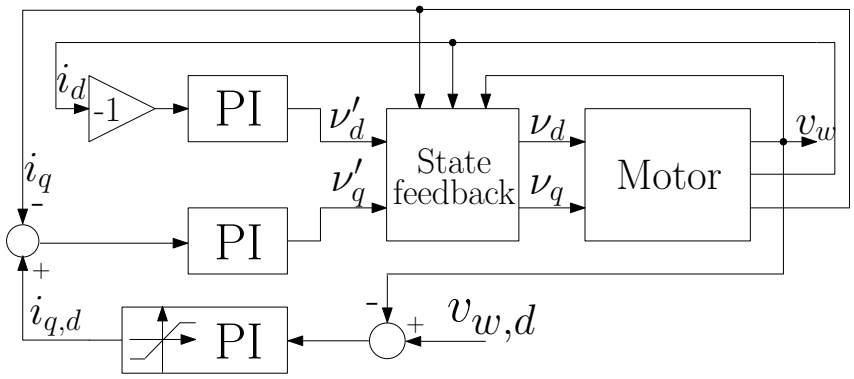

Figure 7.3: Block diagram of the speed control [13].

Torque-tracking. The motor controller tracks the desired torque $\tau_{w,d}$ via the control scheme in Figure 7.2. The desired torque is transformed in a desired quadrature current $i_{q,d}$ by inverting relation (7.7). The auxiliary control input ν'_d acts as a PI controller based on the quadrature current error $i_{q,d} - i_q$ in the quadrature current dynamics $\frac{\mathrm{d}i_q}{\mathrm{d}t}$ in (7.6).

Velocity-tracking. The motor controller tracks the desired velocity $v_{w,d}$ via the control scheme in Figure 7.3 by acting on dynamics (7.6). The velocity error $v_{w,d} - v_w$ is fed to a saturated PI controller (endowed with anti-wind up scheme), whose output is interpreted as desired quadrature current $i_{q,d}$, to be tracked by means of the control input ν'_d as in the torque-tracking mode.

It is a well-know result that, since the currents have a first order dynamics forced by a PI controller, the settling-time of the current loop can be made arbitrarily small by proper tuning the PI controller accordingly with the upper bound on the source voltage. In other words, the current dynamics is negligible with respect to the other mechanical variable dynamics (i.e., steering velocity and steering position). Moreover for a standard car-like robot (7.1), also the steering velocity dynamics is negligible with respect to the dynamics of the steering angle φ (and indeed with respect to the vehicle position $[x, y, \theta]$), hence the choice to model the steering velocity u_v as a control input.

7.2.2 Variable stiffness overview

The variable stiffness paradigm aims to gradually share the control authority between user and vehicle on the basis of the distance from the path. If the vehicle is far from the path, the controller is supposed to have the control authority to reduce the path following errors $|l_x|$, $|l_y|$ and $|\tilde{\theta}|$. Conversely, if the vehicle is close to the path, it gradually releases the control authority to the user. To implement this effect, we

use the two working modes of the motor as shown in Figure 7.4 and explained in the following.

Stiff velocity-tracking. The controller in Figure 7.3 ensures that the actual motor velocity approximately coincides with the velocity reference. In the classical robotic literature, this working mode is considered in the car-like model (7.1) by equation $\dot{\varphi} = u_v$, as explained in Section 7.2.1. Obviously, the velocity command u_v (related to the virtual wheel) is reproduced by imposing the actual velocity of the motors via (2.6). When this mode is enabled, the driver commands up to the maximum available current to reproduce any velocity reference, therefore the user perceives a stiff vehicle since any attempt to modify the vehicle motion is rejected as a disturbance. This "stiff" mode (where the control authority is completely given to the robot) will be use to override the user's command in dangerous situations.

Compliant torque-tracking. The controller in Figure 7.2 ensures that the actual torque produced by the motor approximately coincides with a torque reference. When this mode is enabled, the dynamic equation $\dot{\varphi} = u_v$ is replaced by the augmented model

$$\begin{cases} \dot{\varphi} &= v_\varphi, \\ \dot{v}_\varphi &= \dfrac{u_\tau}{J_{eq}}, \end{cases} \tag{7.8}$$

where state φ is the position of the steering angle of the virtual wheel, the state v_φ is the steering velocity (which is no more control input) and the actual control input u_τ is the torque provided to the virtual wheel. It is remarked that, when the torque-tracking mode of the motor is enabled, the control input acts on the second derivative of the steering angle $\ddot{\varphi}$. This model is even more precise than the previous $\dot{\varphi} = u_v$, since the current dynamics is indeed negligible. If the commanded torque is small, the user perceives a soft vehicle since the driver commands limited currents. This mode is then used to vary the vehicle stiffness, i.e., to implement the spring-damper variable stiffness system.

In other words, a motor is supposed to reproduce either a velocity reference u_v or a torque reference u_τ on the basis of the working mode (see Figure 7.4). The working modes are labeled with a logic variable $q \in \{0, 1\}$. In particular, $q = 1$ indicates that the velocity-tracking mode is enabled, while $q = 0$ indicates that the torque-tracking mode is enabled.

To implement the overall variable stiffness path following controller, we have to design:

1. a feedback control law $u_v(\chi, v)$ solving the path following problem (2.11) when the vehicle has full control authority;

2. a feedback control law $u_\tau(\chi, v)$ to implement the spring-damper variable stiffness system to share the control authority. The stiffness of this spring-damper

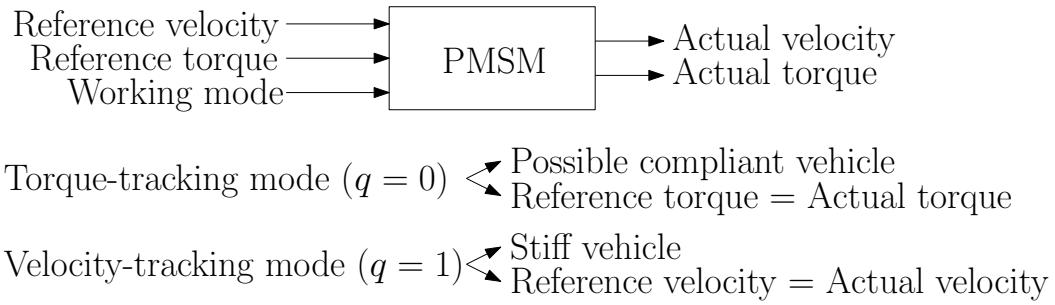

Figure 7.4: Motor schematic behavior [13].

system will vary with respect to the distance from the path, measured as the attitude error e_θ defined in Section 4.2.1;

3. the switching law between the two modes, i.e., the conditions under which q jumps from 0 to 1 and vice versa. We will use hybrid system theory [40] to represent the switching behavior of the model between the dynamics $\dot{\varphi} = u_v$ and (7.8).

In the proposed control scheme, the control input $\dot{s}(\chi, v)$, i.e., the moving velocity of the Frenet frame, will be independent from the working mode of the motors. Moreover the control synthesis of $u_v(\chi, v)$ (i.e., the control input when the velocity-tracking mode is enabled) is based on a backstepping approach, where the steering velocity u_v is designed to ensure that the steering angle φ converges to a desired steering angle φ_d: this desired value φ_d (and its derivative $\dot{\varphi}_d$) will be used to define the equilibrium position of the spring-damper variable stiffness system acting in the torque-tracking mode.

7.3 Steering singularity in assistive robotics

The controller $u_v(\chi, v)$ acts when the motor is the (rigid) velocity-tracking mode. However, it defines also the equilibrium position of the spring-damper variable stiffness system by means of the desired steering angle φ_d and the desired steering velocity $\dot{\varphi}_d$. Therefore it is fundamental that the steering command φ_d avoids the singularity of zero velocity [12]. Using the standar terminology of backstepping, we refer to φ_d as *virtual control input*. Notice that, using the relation $\omega = \frac{v}{d}\tan(\varphi)$ in (7.1), the dynamics of a unicycle-like robot is obtained. Using a control law $\omega(\chi, v)$ and $\dot{s}(\chi, v)$ (e.g., [106]) solving the path following problem (2.11), a virtual control input is immediately obtained as

$$\varphi_d(\chi, v) = \arctan\left(\frac{d}{v}\omega(\chi, v)\right). \tag{7.9}$$

However, a control law $\varphi_d(\chi, v)$ as in (7.9) is singular for $v = 0$. Then, whenever the user stops for any reason the commanded steering angle is $\varphi \approx \varphi_d \approx \pm 90°$. In this case the comfort is really penalized since the user has to force wheel sleeping to restart the motion. Moreover large deviations from the path may be generated by the steering dynamics when the motion is restarted. Although the assumption $v \neq 0$ is a necessary condition for any control law (since the vehicle is propelled by the user), such undesired behavior cannot be tolerated, since the condition $v = 0$ is very common in assistive robotics, where the users often stop and the vehicle always starts from a standstill.

7.4 Path following via variable stiffness

To implement the variable stiffness paradigm, we firstly design a control law solving the path following problem by defining a desired steering angle φ_d and a desired steering velocity $\dot{\varphi}_d$. These two quantities will be used in the computation of the motor torques when the variable stiffness is enabled. The design of the path following controller is reported in Section 7.4.1, the variable stiffness strategy is presented in Section 7.4.2. The two paradigms are combined in Section 7.4.3, where the final variable stiffness path following controller is synthesized and its performance guarantee is formally proved.

7.4.1 Rigid path following controller

The stiff path following controller is designed using the motor in the velocity-tracking mode ($q = 1$), i.e., the classical car-like model (7.1) with dynamics $\dot{\varphi} = u_v$.

Theorem 7 (Stiff path following)**.** *Consider the vehicle kinematic model* (2.10)*. Then, under the assumptions of Theorem 3 the passive path following problem* (4.4) *is solved with* $l_\infty = \tilde{\theta}_\infty = 0$ *by the control actions*

$$\begin{aligned} u_v &= \dot{\varphi}_d - \kappa_\varphi \sin\left(\frac{e_\varphi}{2}\right) - v\,\eta, \\ \dot{s} &= v\,\dot{\xi}, \end{aligned} \tag{7.10}$$

where

$$
\begin{aligned}
\dot{\xi} &= \cos\tilde{\theta} + \kappa_x l_x, \\
\ddot{\xi} &= -\sin\tilde{\theta}\left(\frac{v}{d}\tan\varphi - c(s)v\,\dot{\xi}\right) - \kappa_x v\,\dot{\xi}(1 - c(s)l_y) + v\cos\tilde{\theta}, \\
\gamma &= \dot{\xi}c(s) + \frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y)\,\left(-c(s)\dot{\xi}l_x + \sin\tilde{\theta}\right), \\
\varphi_d &= \arctan\left(d\,(\gamma - \kappa_\theta e_\theta)\right), \\
\dot{\varphi}_d &= \frac{d}{1 + \left(d\,(\gamma - \kappa_\theta e_\theta)\right)^2}\left\{c(s)\ddot{\xi} + \frac{\mathrm{d}\,c}{\mathrm{d}\,s}(s)v\dot{\xi}^2 + + \frac{\mathrm{d}^2\,\delta}{\mathrm{d}\,l_y^2}(l_y)\,v\,\left(-c(s)\,\dot{\xi}\,l_x + \sin\tilde{\theta}\right)\right)^2 + \\
&\quad - \frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y)(\ddot{\xi}\,c(s)\,l_x + \dot{\xi}\,\left[\frac{\mathrm{d}\,c}{\mathrm{d}\,s}(s)\,v\dot{\xi}\,l_x + \,c(s)\,\dot{l}_x\right] - \dot{\tilde{\theta}}\cos\tilde{\theta})\,-\kappa_\theta\left(\dot{\tilde{\theta}} - \frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y)\dot{l}_y\right)\Bigg\}, \\
e_\theta &= \tilde{\theta} - \delta(l_y), \\
e_\varphi &= \varphi - \varphi_d, \\
\eta &= \frac{4e_\theta\cos\left(\frac{e_\varphi}{2}\right)}{d\cos(\varphi_d)\cos(\varphi)},
\end{aligned}
\tag{7.11}
$$

and $\kappa_\theta > 0$, $\kappa_\varphi > 0$, *and* $\kappa_x > 0$ *are constants.*

Remark 1 (Velocity independence)**.** *The desired steering angle* φ_d *is independent from the forward velocity* v*. This implies that the steering command is completely insensitive to the singularity problem* (7.9)*. Moreover, since* v *does not appear in the expression of the steering angle* φ_d *in* (7.11)*, the desired steering velocity* $\dot{\varphi}_d$ *is independent from the forward acceleration* $\dot{v}$*, hence it can be feedback computed via* (7.11)*, without any numerical differentiation.*

The proof of Theorem 7 is based on the capability of controller (7.10) of stabilizing the vehicle attitude e_θ. In other words, controller (7.10) ensures that the vehicle orientation $\tilde{\theta}$ is steered towards the desired approaching angle $\delta(l_y)$, as established by the following theorem.

Theorem 8 (Attitude stabilization)**.** *Under the hypothesis of Theorem 7, controller* (7.10) *ensures that*

$$
\lim_{t\to\infty}|e_\theta| = \lim_{t\to\infty}|\tilde{\theta} - \delta(l_y)| = 0. \tag{7.12}
$$

Proof of Theorem 8. Backstepping is used to stabilize the attitude error e_θ. Then suppose that in the vehicle attitude dynamics (7.1)

$$
\dot{\theta} = \frac{v}{d}\tan\varphi,
$$

the steering angle φ is a virtual control input. We define the Lyapunov function

$$V_1 = \frac{1}{2}(\tilde{\theta} - \delta(l_y))^2 = \frac{1}{2}e_\theta^2,$$

whose time derivative along the solutions, using the definitions of $\dot{l}_y$, $\dot{s}$, $\dot{l}_y$, and γ in (2.10), (7.10), and (7.11), respectively, is rewritten as

$$\begin{aligned}\dot{V}_1 &= e_\theta(\dot{\tilde{\theta}} - \dot{\delta}(l_y)) = e_\theta\left(\frac{v}{d}\tan\varphi - c(s)\dot{s} - \frac{\mathrm{d}\delta}{\mathrm{d}l}(l_y)\dot{l}_y\right) = \\ &= e_\theta\left(\frac{v}{d}\tan\varphi - v\gamma\right).\end{aligned}$$

Notice that, if $\varphi = \varphi_d$, chosen as in (7.11), were the actual control input, we would get $\dot{V}_1 = -v\kappa_\theta e_\theta^2 < 0$, $\forall e_\theta \neq 0$. Since the steering angle φ is not a control input, we have $\varphi = e_\varphi + \varphi_d$, where the steering error $e_\varphi = \varphi - \varphi_d$ is defined in (7.11). Then consider the new Lyapunov function

$$V_2 = V_1 + 1 - \cos\left(\frac{e_\varphi}{2}\right),$$

whose time derivative along the solutions is

$$\begin{aligned}\dot{V}_2 &= e_\theta(\dot{\tilde{\theta}} - \dot{\delta}(l_y)) + \frac{1}{2}\sin\left(\frac{e_\varphi}{2}\right)(\dot{\varphi} - \dot{\varphi}_d) = \\ &= e_\theta\left(\frac{v}{d}\tan(\varphi_d + e_\varphi) - v\gamma\right) + \frac{1}{2}\sin\left(\frac{e_\varphi}{2}\right)(u_v - \dot{\varphi}_d),\end{aligned}$$

where $\dot{\varphi}_d$ is computed symbolically in (7.11) without any numerical differentiation. Using the identity $\tan(\varphi_d + e_\varphi) = \tan(\varphi_d) + \tan(\varphi_d + e_\varphi) - \tan(\varphi_d)$, we get

$$\dot{V}_2 = e_\theta\left(\frac{v}{d}\tan(\varphi_d) - v\gamma\right) + e_\theta\frac{v}{d}\left(\tan(\varphi_d + e_\varphi) - \tan(\varphi_d)\right) + \frac{1}{2}\sin\left(\frac{e_\varphi}{2}\right)(u_v - \dot{\varphi}_d).$$

Using the prosthaphaeresis formula

$$\tan(\varphi_d + e_\varphi) - \tan(\varphi_d) = \frac{\sin(\varphi_d + e_\varphi - \varphi_d)}{\cos(\varphi)\cos(\varphi_d)} = \frac{\sin(e_\varphi)}{\cos(\varphi)\cos(\varphi_d)},$$

and the definition of φ_d in (7.11), we have

$$\dot{V}_2 = -v\kappa_\theta e_\theta^2 + \frac{v}{d}e_\theta\frac{\sin(e_\varphi)}{\cos(\varphi)\cos(\varphi_d)} + \frac{1}{2}\sin\left(\frac{e_\varphi}{2}\right)(u_v - \dot{\varphi}_d).$$

We use the definition of η in (7.11) and the duplication formula $\sin(e_\varphi) = 2\sin\left(\frac{e_\varphi}{2}\right)\cos\left(\frac{e_\varphi}{2}\right)$, to get

$$\begin{aligned}\dot{V}_2 &= -v\kappa_\theta e_\theta^2 + \frac{v\,\eta}{2}\sin\left(\frac{e_\varphi}{2}\right) + \frac{1}{2}\sin\left(\frac{e_\varphi}{2}\right)(u_v - \dot{\varphi}_d) = \\ &= -v\kappa_\theta e_\theta^2 + \frac{1}{2}\sin\left(\frac{e_\varphi}{2}\right)(u_v - \dot{\varphi}_d + v\,\eta).\end{aligned}$$

The control input u_v in (7.10) ensures

$$\dot{V}_2 = -v\kappa_\theta e_\theta^2 - \frac{\kappa_\varphi}{2}\sin^2\left(\frac{e_\varphi}{2}\right) < 0, \quad \forall [e_\theta, e_\varphi]^T \neq [0,0]^T,$$

hence the proof. □

Notice that the control velocity u_v compensates a term $v\eta$, which is singular for $\varphi = 90°$ or $\varphi_d = 90°$. This issue should not be relevant in practice: condition $\varphi = 90°$ is almost impossible because of mechanical constraints; similarly, $\varphi_d = 90°$ is verified only if the controller gains are wrongly tuned or the vehicle is at infinite distance from the path. However, simulation analyses show that the compensation of term $v\eta$ may still worsen the controller performance if the product $\cos(\varphi)\cos(\varphi_d)$ is too small. The control law u_v in (7.10) may be replaced by

$$u_v = \dot{\varphi}_d - \kappa_\varphi \sin\left(\frac{e_\varphi}{2}\right) - v\,\eta\,\max\left(0, \operatorname{sign}\left(\frac{\eta}{2}\sin\left(\frac{e_\varphi}{2}\right) - \frac{\kappa_\theta}{m_\varphi}e_\theta^2\right)\right), \tag{7.13}$$

where $m_\varphi > 1$. This choice commands $u_v = \dot{\varphi}_d - \kappa_\varphi \sin\left(\frac{e_\varphi}{2}\right) - 0$ whenever the second argument $\operatorname{sign}\left(\frac{\eta}{2}\sin\frac{e_\varphi}{2} - \frac{\kappa_\theta}{m}e_\theta^2\right)$ of the $\max(\cdot,\cdot)$ function in (7.13) is negative. By rewriting the Lyapunov function derivative as

$$\dot{V}_2 = -v\left(k_\theta e_\theta^2 + \frac{\eta}{2}\sin\left(\frac{e_\varphi}{2}\right)\right) + \frac{1}{2}\sin\left(\frac{e_\varphi}{2}\right)(u_v - \dot{\varphi}_d),$$

we notice that the term $v\,\eta$ is compensated by the control input u_v in (7.13), only when $\operatorname{sign}\left(\frac{\eta}{2}\sin\frac{e_\varphi}{2} - \frac{\kappa_\theta}{m_\varphi}e_\theta^2\right) = +1$, i.e., when $\frac{\eta}{2}\sin\left(\frac{e_\varphi}{2}\right)$ has a destabilizing effect since it cancels the stabilizing contribution of $\kappa_\theta e_\theta^2$. The constant $m_\varphi > 1$ is then a safety factor which is used to preserve the effect of $\kappa_\theta e_\theta^2$. As aforementioned, although a control law $u_v = \dot{\varphi}_d - \kappa_\varphi \sin\frac{e_\varphi}{2} - v\eta$ is more intuitive, simulation results show that the compensation of the term $v\eta$ when unnecessary can considerably worsen the controller performance. Moreover, experimental analyses show that a simplified controller

$$u_v = \dot{\varphi}_d - \kappa_\varphi \sin\left(\frac{e_\varphi}{2}\right), \tag{7.14}$$

is sufficient to exhibit performance comparable with (7.13) although its much simpler structure.

Theorem 7 holds since controller (7.10) reduces the attitude error $|e_\theta|$. If fact, if $|e_\theta|$ is small, the vehicle approaches and follows the path with limited errors $|l_x|$, $|l_y|$ and $|\tilde{\theta}|$, as guaranteed by Theorem 3.

Proof of Theorem 7. From Theorem 8 we have that $|e_\theta| \to 0$, and then conditions (4.4) hold with $l_\infty = \tilde{\theta}_\infty = 0$ as direct implication of Theorem 3. □

7.4.2 Varying vehicle stiffness

Controller (7.10) ensures that the vehicle follows the path as stated in Theorem 7 by overriding the user's command. If the vehicle is close to the path (i.e., in a safe region), a different controller is engaged to share the control authority with the user via a variable stiffness paradigm.
The vehicle compliance is reproduced by enabling the torque-tracking mode of the motor, by introducing in dynamics (7.8) a virtual spring-damper system having stiffness and damping coefficient depending on the vehicle position. The closer the vehicle to the path, the more compliant the spring-damper (i.e., smaller stiffness and damping coefficient). The distance between vehicle and path is measured via the attitude error $|e_\theta|$, then the vehicle is close to the path if $|e_\theta| \leq \Theta \in [0, \frac{\pi}{2}]$, where Θ is the tolerated threshold. The equilibrium position of the mass-spring damper is defined by the desired steering angle φ_d and the desired steering velocity $\dot{\varphi}_d$ computed by the "rigid" controller (7.11). Hence, the final torque applied by the motor is

$$u_\tau = -\kappa_p \ (\varphi - \varphi_d) - \kappa_v \ (v_\varphi - \dot{\varphi}_d), \tag{7.15}$$

where the positive gains κ_p and κ_v vary on the basis of the distance between vehicle and path. The gains are computed as cycloidal functions of the attitude error, i.e.,

$$\begin{aligned} \kappa_p(e_\theta, \Theta, \bar{\kappa}_p) &= \bar{\kappa}_p \left(\frac{|e_\theta|}{\Theta} - \frac{1}{2\pi} \sin\left(\frac{2\pi |e_\theta|}{\Theta} \right) \right), \\ \kappa_v(e_\theta, \Theta, \bar{\kappa}_v) &= \bar{\kappa}_v \left(\frac{|e_\theta|}{\Theta} - \frac{1}{2\pi} \sin\left(\frac{2\pi |e_\theta|}{\Theta} \right) \right), \end{aligned} \tag{7.16}$$

where the constants $\bar{\kappa}_p > 0$ and $\bar{\kappa}_v > 0$ are the maximum values for the gains κ_p and κ_v, and $\Theta \in [0, \frac{\pi}{2}]$ is the tolerated attitude error threshold. An example of cycloidal profile is reported in Figure 7.5. Notice that, because of the nonlinear behaviour (7.16), for small attitude errors we have $\kappa_p \approx 0$ and $\kappa_v \approx 0$, i.e $u_\tau \approx 0$ in (7.15). In other words, where the vehicle is close to the path, the controller has a limited authority since the commanded torque is very small. Conversely, if the attitude error $|e_\theta|$ increases, the controller authority increases as well since the gains κ_p and κ_v become large. Notice also that the controller (7.15) is highly nonlinear since the control action is proportional to the steering error $\varphi - \varphi_d$ and the steering velocity error $v_\varphi - \dot{\varphi}_d$ via two gains that vary on the basis of the attitude error $e_\theta = \delta(l_y) - \tilde{\theta}$. It is recalled that, theoretically, a PD-like controller as (7.15) ensures limited but nonzero tracking error (since a feed-forward term $\ddot{\varphi}_d$ is missing) only if the gains are large.

7.4.3 Hybrid blend of the two controllers

In this section the path following controller (7.10) is combined with the variable stiffness approach (7.15) using hybrid system theory.

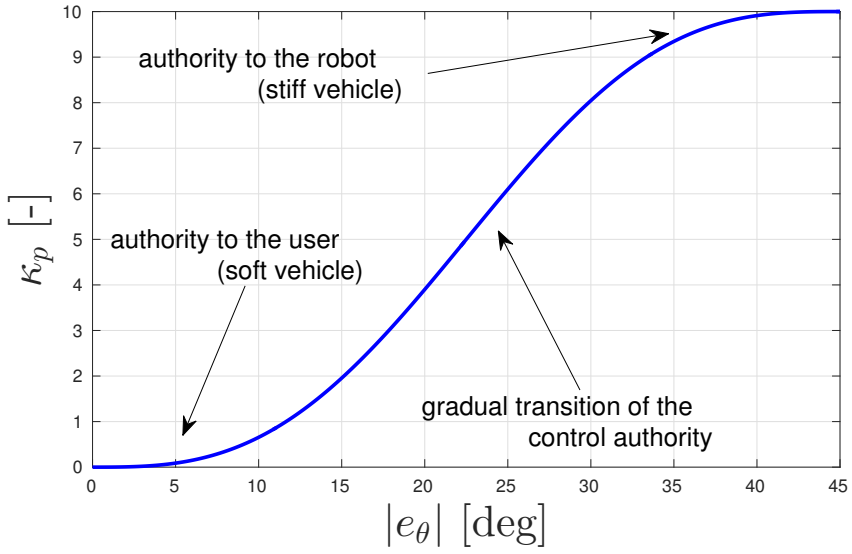

Figure 7.5: Example of cycloidal stiffness with $\bar{\kappa}_p = 10$ and $\Theta = 45°$ [13].

We model the switching behavior of the model between the first order dynamics $\dot{\varphi} = u_v$ and the second order dynamics (7.8). The hybrid model embeds the dynamics of the steering velocity v_φ by introducing two states v_{φ_0} and t_j with discrete dynamics only, representing the initial condition of the integrator and the time instant in which the working mode is updated by the jump map, respectively. We have

$$\begin{cases} \dot{\varphi} & = q u_v + (1-q)\left(v_{\varphi_0} + \displaystyle\int_{t_j}^{t} \frac{u_\tau}{J_{eq}} d\bar{t}\right), \\ \dot{v}_{\varphi_0} & = 0, \\ \dot{q} & = 0, \\ \dot{t}_j & = 0, \end{cases} \tag{7.17}$$

$$\text{for } [\chi, v_{\varphi_0}, q, t_j]^\top \in \mathcal{C},$$

$$\begin{cases} \varphi^+ & = \varphi \\ v_{\varphi_0}^+ & = u_v, \\ q^+ & = 1-q, \\ t_j^+ & = t, \end{cases} \tag{7.18}$$

$$\text{for } [\chi, v_{\varphi_0}, q, t_j]^\top \in \mathcal{D},$$

where u_v and u_τ are computed as (7.10) and (7.15), respectively, and $\mathcal{C} = \mathcal{C}_0 \cup \mathcal{C}_1$

and $\mathcal{D} = \mathcal{D}_0 \cup \mathcal{D}_1$ are the flow set and the jump set, respectively, defined via

$$\begin{aligned} \mathcal{C}_0 &= \{V(\chi) \leq V_{\text{out}} \ \wedge \ q = 0\}, \\ \mathcal{C}_1 &= \{V(\chi) \geq V_{\text{in}} \ \wedge \ q = 1\}, \\ \mathcal{D}_0 &= \{V(\chi) \geq V_{\text{out}} \ \wedge \ q = 0\}, \\ \mathcal{D}_1 &= \{V(\chi) \leq V_{\text{in}} \ \wedge \ q = 1\}, \end{aligned} \tag{7.19}$$

where $V(\chi) \geq 0$ is a function to be designed that triggers the switching of the motor mode q, and $V_{\text{out}} > V_{\text{in}} > 0$ are positive constants. The condition $V_{\text{out}} > V_{\text{in}}$ ensures that the jumps of q takes place with hysteresis. In order to guarantee the path following performance, function $V(\cdot)$ is chosen as the Lyapunov function used in the design of the attitude stabilizer in Theorem 8, i.e.,

$$V(\chi) = \frac{1}{2} e_\theta^2 + 1 - \cos\left(\frac{e_\varphi}{2}\right). \tag{7.20}$$

The rationale of the proposed controller is the following and sketched in Figure 7.6. If the vehicle is far from the path ($V(\chi)$ is large, i.e., $V(\chi) \geq V_{\text{in}}$ in the definition of $\mathcal{C}_1$ in (7.19)) we have $q = 1$ hence $\dot{\varphi} = u_v$, therefore the velocity-tracking mode is enabled. In this condition the control authority is completely taken by the robot an the vehicle is perceived as stiff. As shown in the proof of Theorem 8, the control input u_v in (7.10) ensures that $V(\chi)$ decreases. When the Lyapunov function $V(\chi)$ becomes small (implying small attitude error $|e_\theta|$) the condition $V(\chi) \leq V_{\text{in}}$ in $\mathcal{D}_1$ is triggered, then the motor switches in the torque-tracking mode (q jumps from 1 to 0 and hence the solutions enter the set $\mathcal{C}_0$) and the control input becomes u_τ in (7.15), ensuring gradual transition of the control authority via the variable stiffness. The velocity-tracking mode is re-enabled only if $V(\chi)$ (and hence the attitude error $|e_\theta|$) becomes large via the condition $V(\chi) \geq V_{\text{out}}$ in $\mathcal{D}_0$.

The hybrid control law defined by dynamics (7.17) and (7.18) solves the path following problem (2.11) for arbitrary tolerated errors l_∞ and θ_∞ via a proper tuning of the hysteresis threshold $V_{\text{out}} > V_{\text{in}}$, as stated by the following theorem.

Theorem 9 (Compliant path following)**.** *Consider the vehicle kinematic model* (2.10) *and the hybrid controller defined by* (7.17) *and* (7.18)*. Under the assumptions of Theorem 3, for all $l_\infty > 0$ and $\tilde{\theta}_\infty > 0$ there exists an upper hysteresis threshold $V_{out} > 0$ such that the passive path following conditions* (4.4) *hold.*

Proof of Theorem 9. By Theorem 8, the hybrid controller (7.17) and (7.18) ensures that $V(\chi)$ in (8.1) enters in finite time and remains inside the region $V(\chi) \leq V_{\text{out}}$, since, whenever $V(\chi) \geq V_{\text{out}}$, the hybrid controller commands $\dot{\varphi} = u_v$ that implies $\dot{V} < 0 \, \forall [e_\theta, e_\varphi]^\top \neq [0, 0]^\top$. This implies that the attitude error $|e_\theta|$ enters in finite time and remains inside the region $|e_\theta| \leq \sqrt{2V_{\text{out}}}$. Therefore, by Theorem 3, for all $l_\infty > 0$ and $\tilde{\theta}_\infty > 0$ there exists a sufficiently small $V_{\text{out}} > 0$ such that conditions (2.11) hold, hence the proof. □

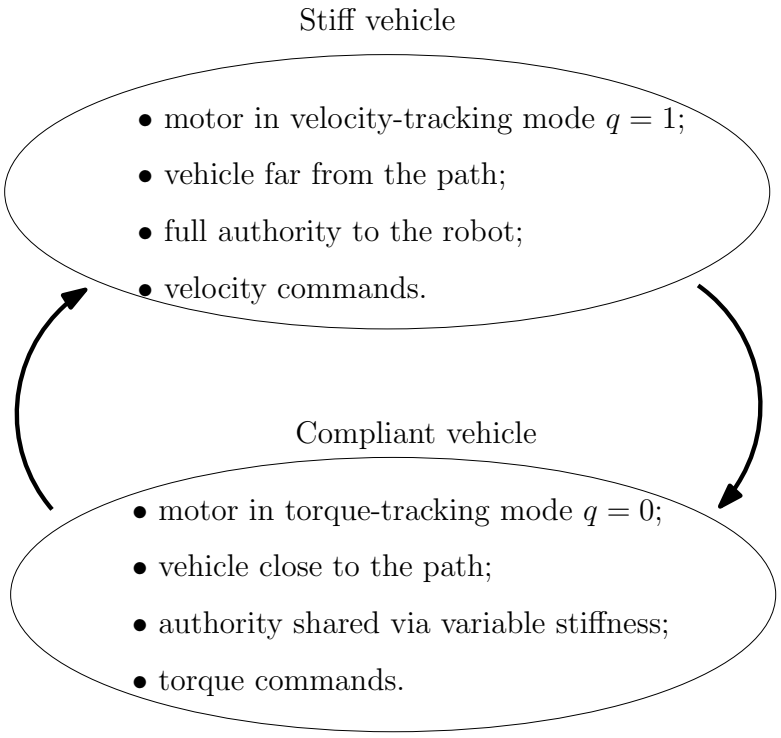

Figure 7.6: Final hybrid controller [13].

7.5 Simulation results

The proposed controller is firstly tested by several simulations. In the following simulations, the control velocity u_v is (7.13), the controller gains are $\kappa_x = 1$, $\kappa_\theta = 2$, $\kappa_\varphi = 10$, and $m_\varphi = 2$, the approaching angle function $\delta(\cdot)$ is chosen as (4.10), and the implemented forward velocity of the user is $v(t) = (1 + 0.2\sin(0.6\pi t))$ m/s. The variable stiffness parameter are $\bar{\kappa}_p = 2.5$, $\bar{\kappa}_v = 0.1$ and $\Theta = 70°$. The hysteresis thresholds are set to $V_{\text{out}} = \frac{1}{2}\Theta^2 = 0.75$ and $V_{\text{in}} = 0.14$. Moreover the complete dynamics of the actuator (7.2) is simulated. The motor is controlled as explained in Section 7.2.1 to reproduce the velocity and torque commands u_v and u_τ of the path following controller (7.17). In the motor dynamics, the load torque generated by the friction between steering wheel and floor is simulated using a Gaussian model comprising coulomb and viscous friction. The vehicle initial conditions are $[x(0), y(0), \theta(0)]^T = [13\text{ m}, -7\text{ m}, 80°]^T$, while the motor initial condition is $[\varphi(0), v_\varphi(0), i_d(0), i_q(0)]^T = [0°, 0\text{ rad/s}, 0\text{ mA}, 0\text{ mA}]^T$.

In the simulations labeled "rigid" and "uncooperative" of Figure 7.7, the user applies a constant counter-clockwise torque to the steering wheel of the vehicle. In simulation labelled "rigid", the rigid controller u_v in (7.10) is always applied (i.e., the torque-tracking mode of the motor is not allowed by setting $V_{\text{out}} < 0$). Notice

that the disturbance introduced by the user is completely rejected and the path is perfectly followed in accordance with Theorem 7. In simulation "uncooperative", the variable stiffness is enabled. Then the controller (7.17) initially applies a rigid velocity command u_v to reduce the distance from the path by rejecting the disturbance. Then, when the vehicle gets closer to the path, the hybrid controller (7.18) switches in compliant mode (i.e., the torque-tracking mode on the motor is enabled). The compliant controller u_τ in (7.15) maintains limited the Lyapunov function V in (8.1) without requiring the switch in rigid mode (i.e., the discrete dynamics (7.18) is no more activated). However, a zero attitude error is not possible, since if $e_\theta = 0$ the control authority would be completely given to the user (that is pushing the vehicle away from the path) since the gains in (7.15) would be $\kappa_p = \kappa_v = 0$. The overall result is that the followed path is parallel to the reference (see also Figure 7.8). In simulation labeled "go straight" the simulated user always tries to go straight by applying a torque equal to -10φ straightening the steering wheel. This way, the vehicle steers to follow a corner only if the attitude error $|e_\theta|$ (and therefore the gains in (7.15)) is sufficiently large, hence with delay with respect to the planned path. Figure 7.8 shows the values of the stiffness κ_p used in control law u_τ in (7.15) after that the variable stiffness mode $q = 0$ is enabled. Notice that, for users that do not autonomously follow the path ("uncooperative" and "go straight"), the stiffness has a similar initial behavior. Initially κ_p decreases since the steering wheels are properly oriented to reduce the attitude error because of the previous action of the velocity controller u_v. Once κ_p becomes very small, the disturbance is no more rejected, hence the attitude error $|e_\theta|$ increases. Once $|e_\theta|$ and κ_p are sufficiently large, the control action u_τ in (7.15) is large enough to ensure that the vehicle remains close to the path. In case of a constant disturbance (label "uncooperative"), the stiffness κ_p remains almost constant since the constant disturbance is compensated by the controller (7.15): in this equilibrium condition, the motion of the vehicle is parallel to the path (see Figure 7.7). Figure 7.8 shows also a simulation with a cooperative user (label "cooperative"), who applies a torque $-10(\varphi - \varphi_d)$ on the steering wheel (i.e., the user always acts as a proportional controller to stabilize the steering position). In this case, the stiffness κ_p converges to zero and the path followed is similar to the path "rigid" in Figure 7.7. In other words, if the user is cooperative, the path is followed by leaving to the user full control authority (the controller has no authority since the gains are zero).

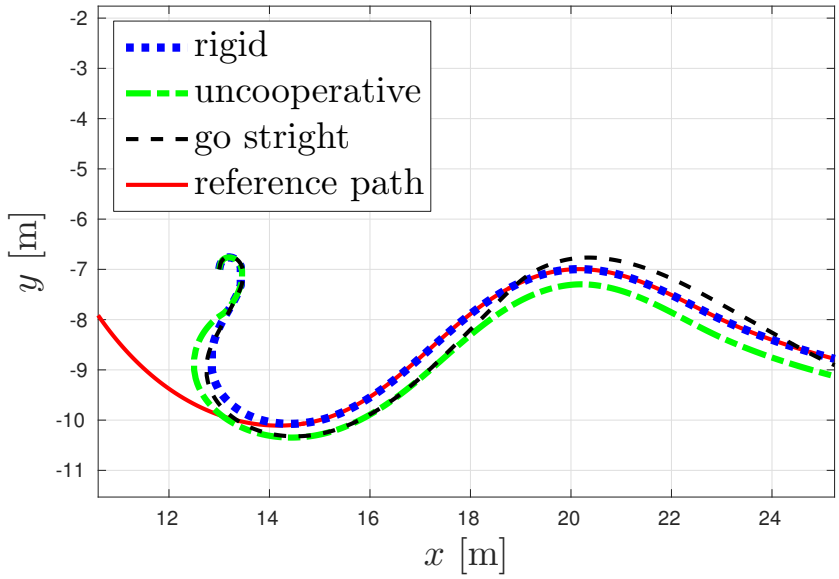

Figure 7.7: Simulations of the path following in the presence of disturbances introduced by the assisted person [13].

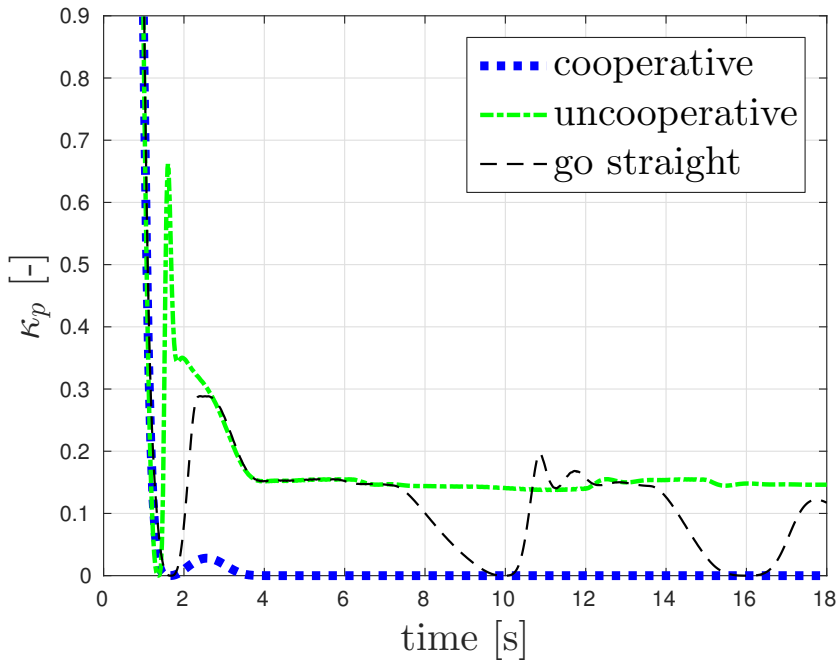

Figure 7.8: Stiffness behavior with different users [13].

7.6 Experimental results

Several tests, executed in facilities of University of Trento (see e.g., Figure 7.9), are presented to evaluate the proposed controller. Section 7.6.1 discusses the avoidance of the steering singularity described in Section 7.3. Section 7.6.2 shows how the gradual authority allocation of the variable stiffness handling deals with different user profiles, proposing also a quantitative measure of user's cooperation. Further experiments are reported and analyzed in Section 7.6.3.

7.6.1 Harnessing steering singularities

We first present the capability of the proposed controller to handle the assistive robotics steering singularity described in Section 7.3.
To this end, the users were requested to push the vehicle forward on a straight path, and after few seconds, they were requested to stop in a marked position, i.e., $v = 0$. After 5 seconds, the users were asked to complete the path.
To simplify the experiments and to focus on the singularity handling, the authority-sharing varying stiffness behavior of the guidance is not implemented. The front motors are used to track with approximately zero steady state error a given virtual front steering angle. In particular, Figure 7.10 shows the experimental trajectories tracked by one user to follow the straight reference path. The virtual steering angle φ in (7.1) is commanded to the motors as follows:

- case labeled "φ_d": the commanded steering angle is equal to the desired velocity-independent virtual steering angle φ_d of the proposed controller in (7.11), i.e., in steady state we get

$$\varphi \approx \varphi_d = \arctan\left(d\left(\gamma - \kappa_\theta e_\theta\right)\right);$$

- case labeled "φ_1": the commanded steering angle is used to reproduce the angular velocity ω° in [106] solving the path following problem (2.11) for a unicycle like robot, i.e.,

$$\varphi \approx \varphi_1 = \arctan\left(\frac{d}{v}\omega^\circ\right);$$

- case labeled "φ_2": the commanded steering angle is obtained as in the case labeled "φ_1", but a saturation of 45° is added to ensure that the commanded steering angle remains limited, i.e.,

$$\varphi \approx \varphi_2 = \mathrm{sat}_{\frac{\pi}{4}}\left(\arctan\left(\frac{d}{v}\omega^\circ\right)\right),$$

  where $\mathrm{sat}_L(\cdot)$ denotes the symmetric saturation function with saturation limits $\pm L$.

Figure 7.9: Experimental setup in the Department of Information Engineering and Computer Science of University of Trento.

Notice that all the three control laws allow the user to follow the path in the initial part of the experiment. When the user stops, the singularity $v = 0$ is triggered. The proposed controller φ_d is completely insensitive to this issue, allowing a smooth path path following when the user restarts. Conversely, when the singularity is triggered, the controller φ_1 steers the wheels at 90° ($\varphi_1 = \arctan\left(\frac{d}{v}\omega^\circ\right)) = \arctan\left(\frac{d}{0}\omega^\circ\right)) \rightarrow$ 90°. When the user starts over, since the steering wheels cannot be instantaneously turned, the vehicle moves in the non-straight direction imposed by the steering system, as reported by the dotted trajectory in Figure 7.10 and dubbed φ_1. Notice that the vehicle slightly comes back (i.e., its curvilinear abscissa decreases) since, to reproduce a virtual steering angle of 90°, the differential steering strategy requires a wheel (the right wheel in the experiment in Figure 7.10) to steer at 90°. Then, when the user pushes the vehicle forward, the right steering wheel is turned to an angle $\varphi_r > 90^\circ$ because of the friction with the ground. To overcome this nuisance, a saturation is applied as $\varphi_2 = \text{sat}_{\frac{\pi}{4}}(\varphi_1)$ in the last case. Albeit the trajectory obtained is better than in the previous case (see the thick dashed line in Figure 7.10 named φ_2), the perceived comfort and the following performance are clearly worsened with respect to the proposed solution.

To justify the experimental results in Figure 7.10, let us explicetly compare the the angular velocity ω° in [106] solving the path following problem (2.11) and our proposed strategy. We have

$$\omega^\circ = c(s)(v\,\cos(\tilde{\theta}) + \kappa_x l_x) + \dot{\delta} - \kappa_3 l_y v \frac{\sin(\tilde{\theta}) - \sin(\delta)}{e_\theta} - \kappa_2(\tilde{\theta} - \delta), \tag{7.21}$$

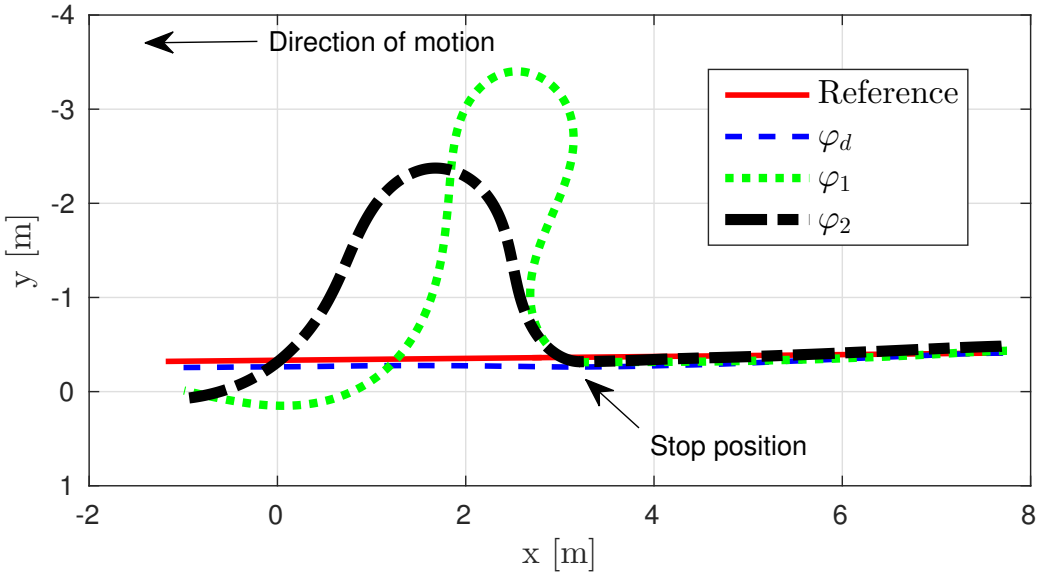

Figure 7.10: Experimental effects of singular velocities. The proposed algorithm (labeled "φ_d") lets the vehicle remain in the path when the singularity happens, which instead is not the case for the compared controllers (labeled "φ_1" and "φ_2").

where κ_x, κ_2, and κ_3 are positive gains. Combining the commanded steering angle $\varphi \approx \varphi_d$ in (7.11) with the car like attitude dynamics (7.1), we have

$$\begin{aligned}\omega &= \frac{v}{d}\tan(\varphi) \approx \frac{v}{d}\tan(\varphi_d) = \frac{v}{d}\tan(\arctan\left(d\left(\gamma - \kappa_\theta e_\theta\right)\right)) = v(\gamma - \kappa_\theta e_\theta) \\ &= v\left((\cos\tilde{\theta} + \kappa_x l_x)c(s) + \frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y)\,\left(-c(s)(\cos\tilde{\theta} + \kappa_x l_x)l_x + \sin\tilde{\theta}\right) - \kappa_\theta e_\theta\right),\end{aligned} \tag{7.22}$$

which coincides with (6.2). The key difference between the angular velocity ω° in (7.21) from [106] and the proposed controller in (7.22) is that we command in (7.22) an angular velocity ω *linear* with respect to the forward velocity v chosen by the user. This way, when the vehicle is still, we require no rotations (i.e., $\omega = 0$). It is remarked that for a unicycle-like robot differentially driven, commanding a nonzero angular velocity when the vehicle is still, is indeed possible (by just imposing opposite rotational velocities on the wheels). Conversely, the attitude dynamics $\omega = \frac{v}{d}\tan\varphi$ in (7.1) of a car-like robot states that, in order to have nonzero angular velocity when the vehicle is still, the only theoretical possibility is $\varphi = 90^\circ \implies \tan\varphi \to \infty$, hence the singularity. When the vehicle is moving, i.e., $v > 0$, the linear relation between the vehicle velocities is less relevant (indeed all the controllers work in the experiments of Figure 7.10) since the singularity is never triggered.

7.6.2 Control authority gradual allocation

This section compares the effect of the gradual allocation of the control authority on cooperative and uncooperative users. The testers in the experiments of Figure 7.11 are *uncooperative*, i.e., they are required to try to push the vehicle far from the path. Conversely, the testers in the experiments of Figure 7.15 are *cooperative* since they were required to follow the path with the help of the tablet showing the position of the vehicle (similar to a standard navigation system).

Experimental paths tracked by uncooperative users are shown in Figure 7.11, while the corresponding attitude errors are reported in Figure 7.12. Notice that in the initial straight part of the experiments, all the users push the vehicle on the left. The result is a motion almost parallel to the reference path as in the simulations of Figure 7.7. Figure 7.13 shows the logic variable q for Test 3 of Figure 7.13. Notice that, although the user is trying to constantly push the vehicle away, the rigid velocity-tracking mode $q = 1$ is rarely used, hence the torque-tracking mode is generally sufficient to perform the path following task (indeed it is the case of the other two tests in Figure 7.11, where q always remains zero). The currents commanded by the drivers of the front motors are reported in Figure 7.14. It is remarked that, when the torque-tracking mode is enabled, a safe saturation of 2A is imposed to the current corresponding to the commanded torque (7.15), while when the velocity-tracking mode is enabled, the commanded currents are chosen without limitations by the motor drivers, as in Section 7.2.1. Notice that, when the rigid velocity-tracking mode is enabled, large currents are applied to track the velocity command u_v in (7.14). In this case, the vehicle is very efficient in reducing the attitude error e_θ (see Figure 7.12), in accordance with Theorem 7.

Experimental paths tracked by cooperative users are shown in Figure 7.15, while the corresponding attitude errors are reported in Figure 7.16. Notice that the users exploit their control authority to improve the path following performance with respect to the uncooperative case of Figure 7.11. Notice also that the attitude errors e_θ are considerably smaller than the uncooperative case of Figure 7.12, hence confirming that the users are properly using their control authority. In all the experiments of Figure 7.15, the torque-tracking mode is always active, since no "rigid" intervention by the robot is needed. Figure 7.17 shows the currents commanded by the drivers of the front wheels. Notice that much smaller currents are needed to follow the path with respect to Figure 7.14, confirming that the user is properly contributing to the path following.

The commanded currents may be considered representative of the cooperation between user and robot, via the cooperation index

$$I_c = \left(\frac{1}{L_{\text{path}}} \int_0^{s_{\text{final}}} \left(i_r^2 + i_l^2 \right) \mathrm{d}\, \bar{s} \right)^{-1}, \tag{7.23}$$

where i_r and i_l are the right and left current, respectively. The integral is computed with respect to $\bar{s}$, i.e., the curvilinear abscissa of the closest point of the vehicle to

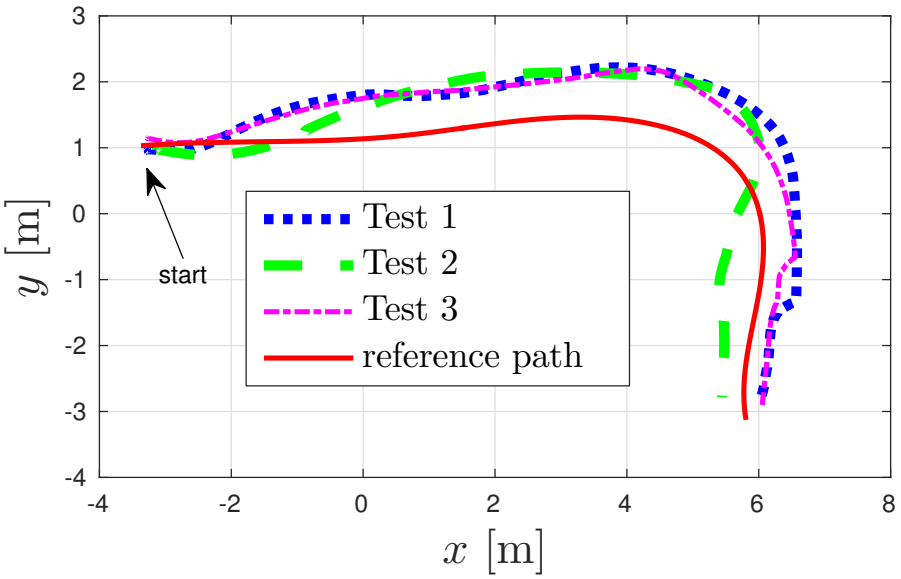

Figure 7.11: Paths followed with uncooperative users.

the path. This choice is preferable with respect to an integration in time, since if the vehicle is still, $\bar{s}$ does not vary and hence the cooperation index I_c remains constant. s_{final} is the last value assumed by $\bar{s}$ (for all the experiments it coincides with the curvilinear abscissa of the last point of the path, i.e., the length of the path L_{path}).

The larger the cooperation index, the less the energy $\int_0^{s_{\text{final}}} (i_r^2 + i_l^2)\, \mathrm{d}\, \bar{s}$ the robot spends to counteract against the user. Notice that, if the user ideally follows the path exactly as controller (7.10), no actuation is needed (i.e., $i_r = i_l = 0$) and the cooperation index I_c tends to infinity. Conversely, if the user is uncooperative, the robots spends a large amount of energy to compensate the disturbing action of the user and to limit the path following error: in this case, large currents imply a smaller cooperation index I_c.

Notice from Table 7.1 that the cooperation indexes for the experiments in Figure 7.15 (where a cooperative behavior is reproduced by the users) are at least one order of magnitude larger than the ones of the experiments in Figure 7.11 (where an uncooperative behavior is reproduced by the users), confirming the goodness of (7.23) to measure the human-robot cooperation.

7.6.3 Quantitative analysis

Further quantitative studies are proposed in this section. The walker was tested by 20 users. Each user was requested to follow a path (randomly selected in a pool of 4 paths starting close to the walker position) after a first trial to take confidence with the system. The controller parameters are the same used in the

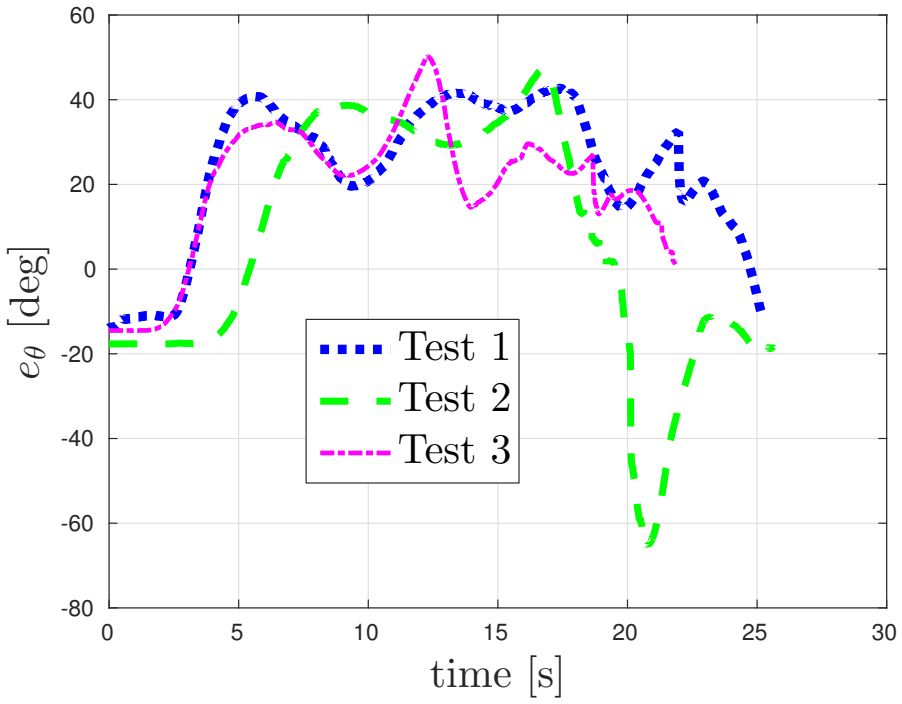

Figure 7.12: Attitude errors for the tests with uncooperative users of Figure 7.11.

previous sections. In all these experiments, the rigid velocity-tracking mode was never activated, hence the variable stiffness approach resulted sufficient to maintain the robot in the proximity of the path.

Figure 7.18 shows examples of paths tracked by the users, while Figure 7.19 reports the corresponding attitude errors. Notice that the reference path (red solid line) has two difficult corners (i.e., portions with large curvature). All the users successfully followed the path. Notice that in Test 1, the user performed the first corner with a larger curvature radius, producing large attitude errors with respect to the other tests. As a consequence he/she continued the task by moving almost parallel to the path. In this case the vehicle is still very compliant (i.e., large human authority), hence the user was not forced to approach the path.

Figure 7.20 reports the user's speed for the experiments in Figure 7.18, showing that the passivity of the variable stiffness handling properly fits with all user profiles. Notice that the user in Test 1 moves approximately at double speed with respect to the user in Test 4, who hence takes twice the time to follow the path. Notice also that in Test 4 the user stops and no steering singularity problems take place in accordance with the results of Section 7.6.1.

Figure 7.21 compares the actual steering angle φ and its desired value φ_d in (7.11) for Test 3 in Figure 7.18. Since the user maintains the vehicle close to the path (see Figure 7.18), the corresponding attitude error is small (see Figure 7.19), hence

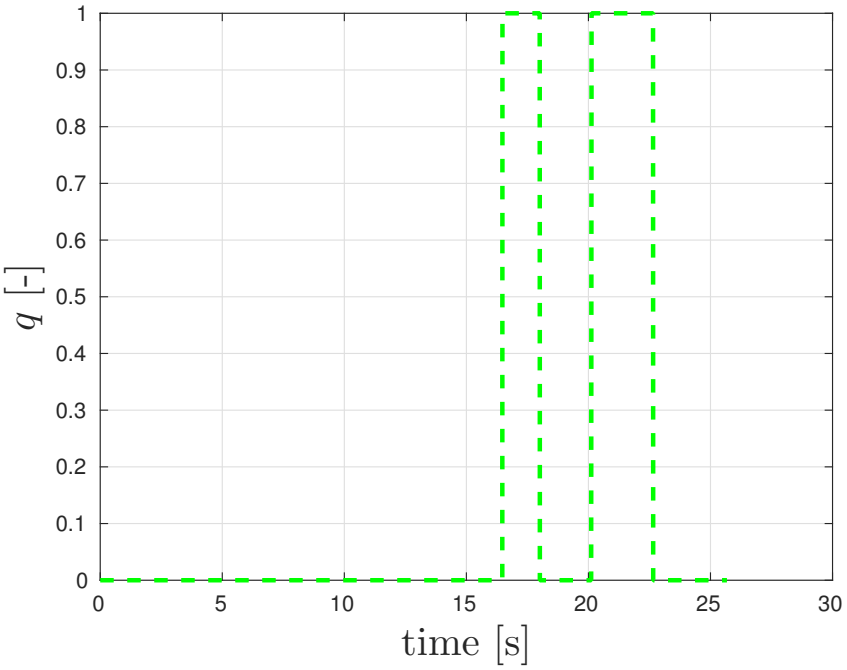

Figure 7.13: Logic variable for Test 3 in Figure 7.11.

the vehicle is compliant, i.e., a low-gain approach is used to stabilize the steering position. As a consequence, the steering angle is imposed by the user since the robot inputs are almost negligible with respect to the forces applied by the user, even though the steering error $e_\varphi = \varphi - \varphi_d$ is large: this behavior further highlights the authority-sharing paradigm. In the first part of the experiment the vehicle is on the left of the path, hence the robot requires to steer right ($\varphi_d < 0$). However, the user prefers to move forward ($\varphi \approx 0$). In the proximity of the curves, the attitude error increases, but its value is not sufficiently large to ensure high-gains and good tracking performance for the steering angle. However, the user understands that the robot is slightly stiffening, hence steering the vehicle towards the path is perceived as easier and more natural. As a consequence, the user imposes a steering angle which has at least the correct sign with respect to the desired one.

The mean cooperation index in (7.23) between all the experiments is 3.07A^{-2}, and its minimum is 0.64A^{-2}. Notice that this mean value is close to the cooperative case of Section 7.6.2.

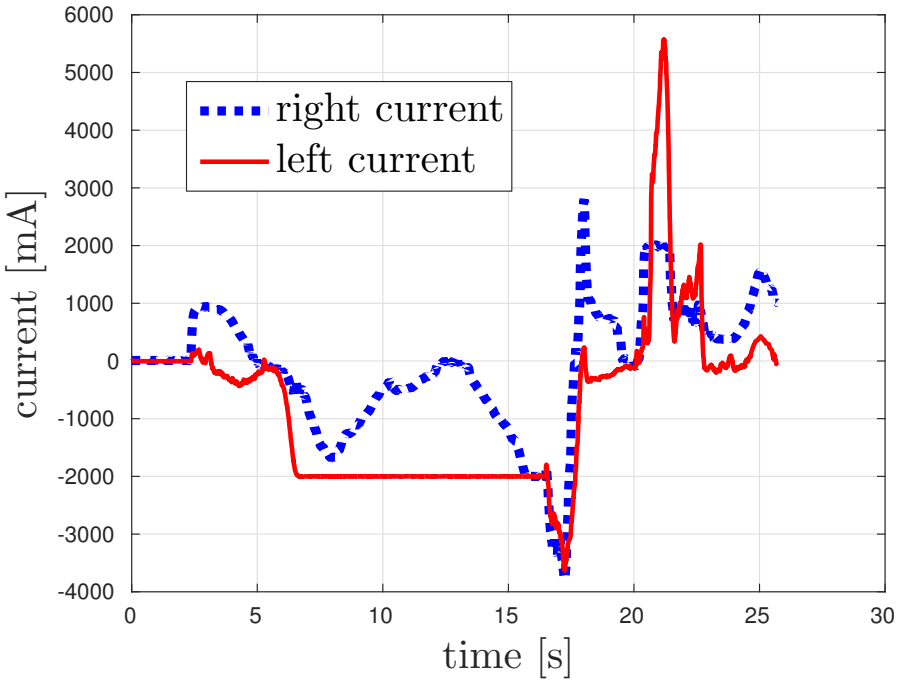

Figure 7.14: Commanded currents for Test 3 in Figure 7.11.

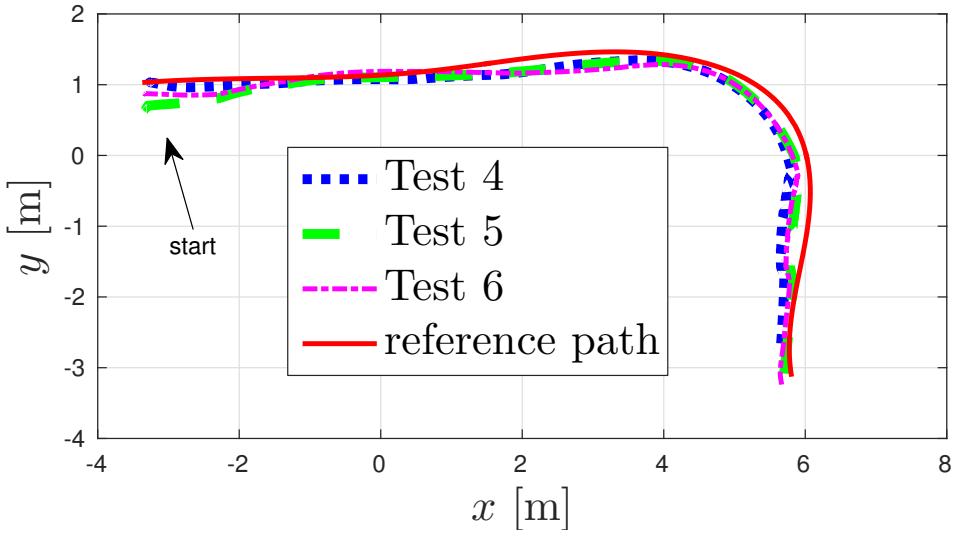

Figure 7.15: Paths followed with cooperative users.

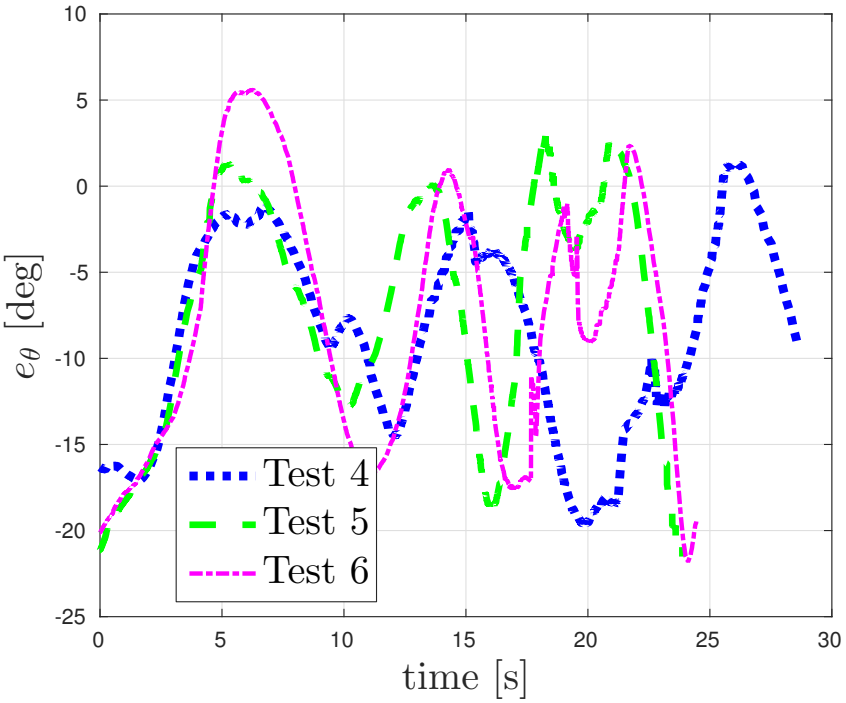

Figure 7.16: Attitude errors for the tests with cooperative users of Figure 7.15.

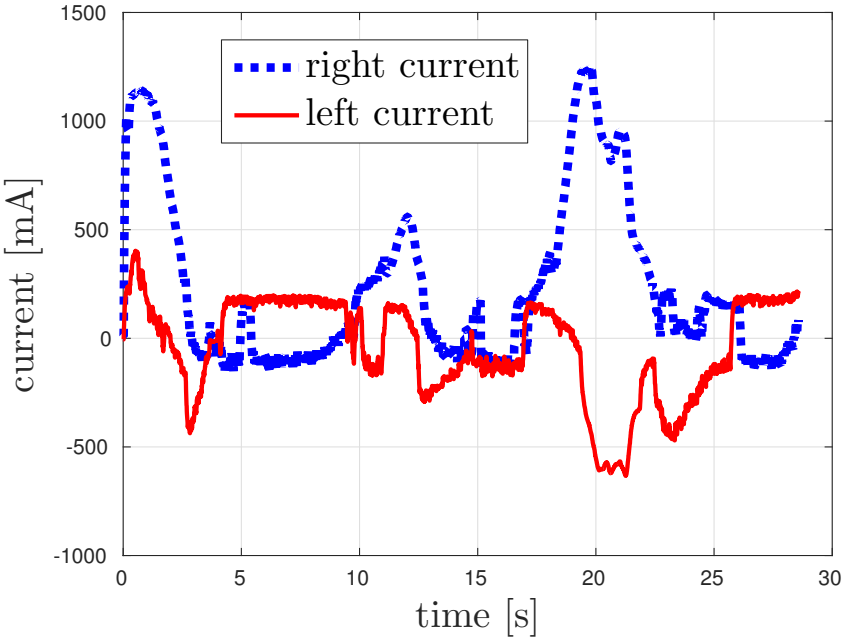

Figure 7.17: Commanded currents for Test 4 in Figure 7.15.

Behavior	Test name	$I_c[\mathrm{A}^{-2}]$
Uncooperative	Test 1	0.267
Uncooperative	Test 2	0.234
Uncooperative	Test 3	0.232
Cooperative	Test 4	5.096
Cooperative	Test 5	6.566
Cooperative	Test 6	4.573

Table 7.1: Cooperation index for the experiments in figures 7.11 and 7.15.

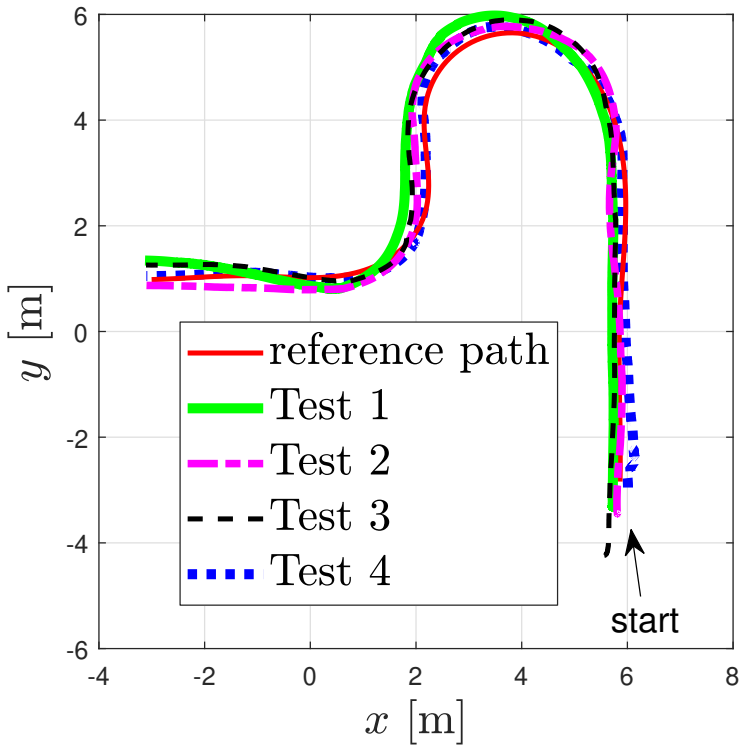

Figure 7.18: Examples of followed paths by users guided with variable stiffness handling.

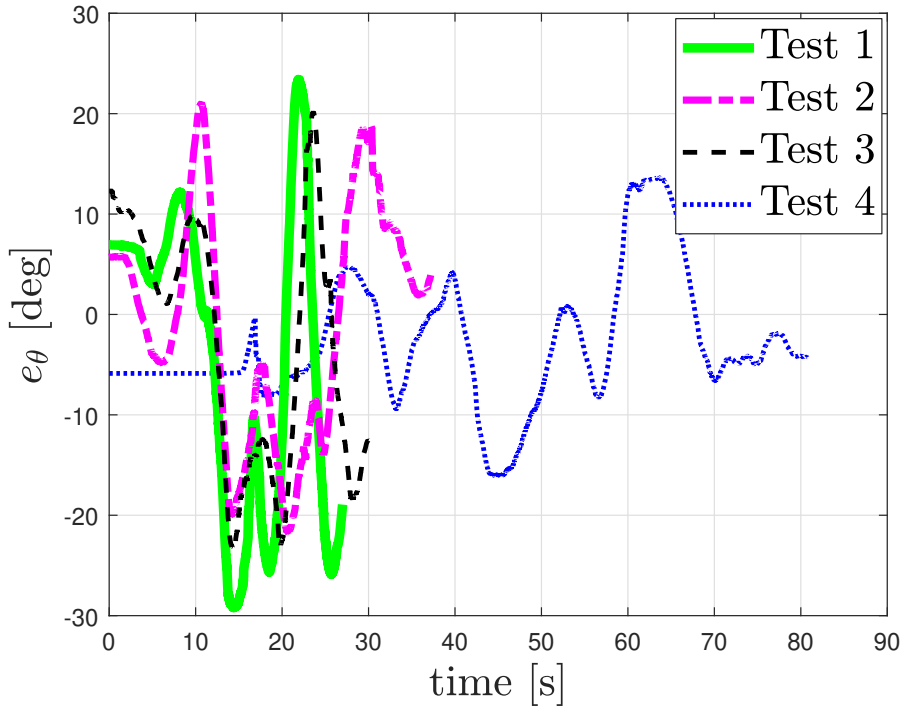

Figure 7.19: Attitude errors for experiments in Figure 7.18.

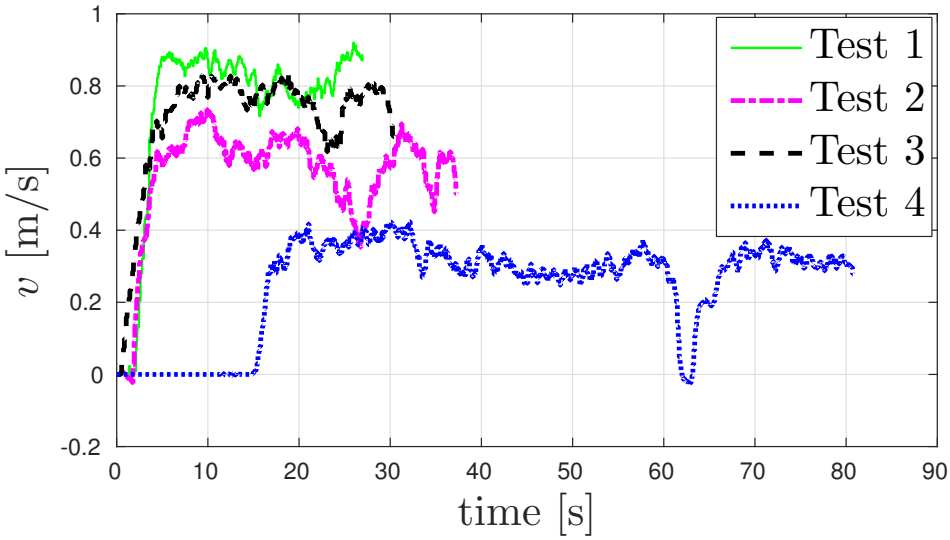

Figure 7.20: Users' velocities for experiments in Figure 7.18.

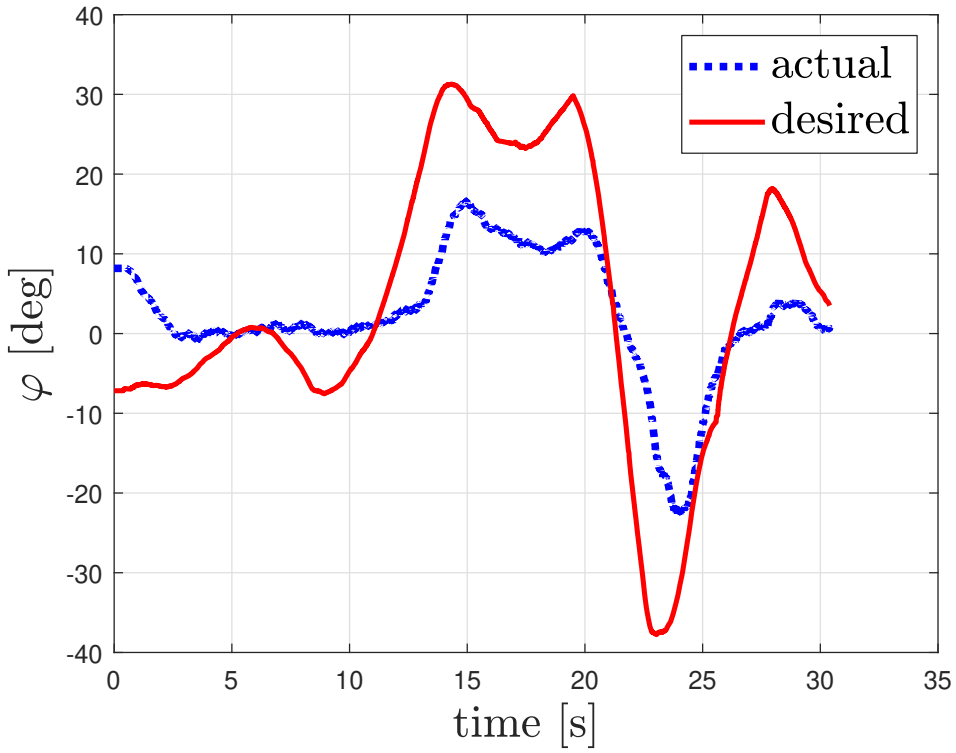

Figure 7.21: Steering commands for Test 3 in Figure 7.18.

7.7 Final comments

This chapter describes an authority-sharing control strategy in which the assistive walker is equipped with front steering wheels, hence offering several advantages, such as passivity, accuracy, and comfort. The control authority is gradually shared between human and robot using a variable stiffness vehicle handling paradigm, modifying the perceived physical interaction of the user with the robot on the basis of the path following error. The interaction is tuned by acting on the compliance of the steering system using a varying-gain approach. We provide extensive validation of the approach comprising theoretical, simulative, and experimental studies. The controller may exhibit a totally rigid behavior (i.e., no compliance) in emergency situations by activating the velocity-tracking mode, which is also necessary to ensure theoretical error bounds; however, experimental evidence shows that the gradual allocation of the control authority obtained by compliance is sufficient to maintain the vehicle close to the path, and hence to ensure the user's safety. We also propose the cooperation index, i.e., an effective measure of the cooperation between human and robot, defining the energy spent by the robot to guide the user: the larger the cooperation index, the less the energy used to "fight" the user.

Future work will focus on changing dynamically the vehicle compliance on the basis of the environmental conditions (e.g., the vehicle is required to be stiffer in cluttered environments, where the risk of collision is higher). Moreover, the importance of the cooperation index will be investigated. For instance, small cooperation indexes may indicate a user with limited cognitive abilities (e.g., he/she is not capable to recognize the obstacles and hence the robot has to frequently act). A further possibility is the use of the cooperation index to infer user's intentions: if the user continuously tries to departure from the path (i.e., small cooperation index), he/she may be willing to reach a different location with respect to the one pointed by the path.

Chapter 8

Probabilistic authority-sharing

Most of the solutions available in the literature solving a path following problem require an accurate localization of the robot in the environment. This is not always available in the practice, for instance because of the dead-reckoning problem taking place if a fiducial marker reading is missed because of occlusions. However, the control system is still required to perform actions when the localization is not available (indeed, even stopping the robot is an action).

This chapter proposes to face the localization loss via authority-sharing, hence extending our authority-sharing framework in a stochastic case. We estimate the probability the robot is performing a proper action in the presence of poor localization accuracy, i.e., the controller reliability, using a Lyapunov-based approach. Whenever this reliability is smaller than the expected human reliability, the user is provided with control authority since he/she offers better performance (in a probabilistic sense) than the control system. A huge advantage offered by this approach is that we can instrument the environment with a heavy infrastructure to have accurate localization only in certain areas. The proposed authority-sharing controller is extensively analyzed in simulations and validated experimentally.

8.1 Introduction to authority-sharing based on information precision

The solution to the path following problem is typically designed by supposing that the localization algorithm is "accurate enough" to produce a negligible error in the estimate of the vehicle state [64]. Assuming a good localization accuracy is fairly acceptable for robots relying on exogenous sensor readings (absolute measures) always or most of the times. Indeed, endogenous sensors (relative measures) are affected by the well known dead-reckoning effect that produces an unbounded growth of the position uncertainty [17]. Several ways have been proposed to improve the accuracy of the robot localization, such as optimal deployment of landmarks to

meet a desired target accuracy [24, 87, 124], or using active sensors [73] or mapping detected landmarks [101]. Whatever the strategy used to deploy and use markers in the environment, a certain fact with this type of solution is that the absolute position measures come intermittently. It is well known [105, 88] that closing a control loop with intermittent observation can lead to a poor performance (possibly even to instability) if the average rate is not sufficient to compensate for the system dynamics. On the other hand, a massive deployment of landmarks is inconceivable in realistic environments (e.g., a museum, or a shopping mall).

Authority-sharing offers an elegant escape from this quandary. The key observation is that even a user with mild cognitive impairments is able to maintain a direction of motion when the environment does not require choices (e.g., a corridor). Only in presence of decision points (e.g., bifurcations, cross-roads, doors) a constant intervention of the system is required. A possible way to see this is that the intelligence of the user can be used to compensate the reduced information precision on the environment. This behavior mimics a human being driving his/her car and overriding the suggestions coming from the navigator if the GPS localization is evidently wrong or if an unforeseen obstacle, i.e., road works, blocks the suggested way. Similarly, the autonomous driving system of modern cars gives back the control to the human in case, e.g., of heavy weather [66]. We can translate this simple idea into a design principle: use a heavy infrastructure (dense landmarks) when a close support is required for the user and a light infrastructure (i.e., sparse landmarks) when we can shift the authority to the user. This natural strategy has to be complemented by a control algorithm that decides the balance of the authority according to the accuracy of the information on the system state, which is the most important contribution of this Chapter. Specifically, we propose to allocate the control authority according to the available localization precision. The performance on the path following maximum error are experimentally characterized as a function of the uncertainty growth due to dead-reckoning. To the best of our knowledge, this is the first work that directly considers data uncertainties to rule the controller behavior, being most of the literature devoted to the compensation of parametric model uncertainties (e.g., [3]) or to the disturbance rejection (e.g., [27]).

8.2 Problem recall

The vehicle is modeled as a unicycle-like robot (2.1), i.e.,

$$\begin{cases} \dot{x} &= v\cos\theta, \\ \dot{y} &= v\sin\theta, \\ \dot{\theta} &= \omega, \end{cases}$$

where the control inputs are the linear velocity v and the angular velocity ω, while the path following problem is described using a dynamic Frenet frame (2.10)

$$\begin{cases} \dot{l}_x &= -\dot{s}(1 - c(s)l_y) + v\cos\tilde{\theta}, \\ \dot{l}_y &= -c(s)\,\dot{s}\,l_x + v\sin\tilde{\theta}, \\ \dot{\tilde{\theta}} &= \omega - c(s)\dot{s}. \end{cases}$$

Similarly to Chapter 7, the state of the vehicle can be equivalently represented as $\chi = [l_x, l_y, \tilde{\theta}]^\top$ or $\bar{\chi} = [x, y, \theta]^\top$. The problem at hand is the relaxed path following (4.4),

$$\begin{aligned} \lim_{t\to+\infty} |l_x(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |l_y(t)| &\leq l_\infty, \\ \lim_{t\to+\infty} |\tilde{\theta}| &\leq \tilde{\theta}_\infty. \end{aligned}$$

8.3 Path following solution

The solution to the path following problem is typically designed by supposing that the localization algorithm is "accurate enough" to yield a negligible error on the estimate of the vehicle state χ. Of course, when intermittent observations are adopted, as in the localization system running on the *FriWalk* and reported in [88], the effect of the feedback control can be highly wrong and, hence, the control should be given to the user. To implement this authority-sharing, how the localization accuracy is derived and a description of the controller implemented is needed, which is the purpose of this section.

Vehicle localization

Let us denote $\hat{a}$ the estimate of the quantity a and σ_a the corresponding standard deviation. With localization algorithm we intend the execution of an estimator that provides "suitable estimates" $\hat{x}, \hat{y}$ and $\hat{\theta}$ of the vehicle states of (2.1). For the rollator in Figure 1.8, the available sensors are encoders mounted on the rear wheels (odometry-based localization) and a camera reading landmarks (QR codes placed on the floor, the ceiling or on the walls) whose positions in the map are known. The odometry data are always available but affected by dead-reckoning. The measures of vehicle position and attitude obtained by the landmarks are absolute but available only when a landmark is in the field of view of the camera. The two measures are fused using a Bayesian estimator, such as an extended Kalman filter [17]. The estimator returns minimum variance estimates $\hat{x}, \hat{y}$ and $\hat{\theta}$ of the vehicle state and the corresponding estimation error covariance matrix (3.1)

$$P = \mathrm{E}\left\{[x - \hat{x}, y - \hat{y}, \theta - \hat{\theta}]^\top [x - \hat{x}, y - \hat{y}, \theta - \hat{\theta}]\right\},$$

where $\mathrm{E}\{\cdot\}$ is the expected value operator.

Path following controller

A control law solving the classical path following problem (2.11), i.e., ensuring the convergence of the error e_θ to zero, is given by (6.2) and (5.6), i.e.,

$$\begin{aligned}\omega(\chi) &= v\left(\gamma(\chi) - \kappa\left(\tilde{\theta} - \delta(l_y)\right)\right),\\ \gamma(\chi) &= c(s)(\cos(\tilde{\theta}) + \kappa_x l_x) + \left(-c(s)\left(\cos(\tilde{\theta}) + \kappa_x l_x\right) l_x + \sin(\tilde{\theta})\right)\delta'(l_y),\end{aligned}$$

where $\kappa > 0$ is a gain and $\delta'(l_y) = \frac{\mathrm{d}\,\delta}{\mathrm{d}\,l_y}(l_y)$. Indeed, using the Lyapunov function

$$V = \frac{1}{2}e_\theta^2, \tag{8.1}$$

its time derivative is

$$\begin{aligned}\dot{V} = e_\theta \dot{e}_\theta &= e_\theta\left(\omega - c(s)\dot{s} - \dot{l}_y\delta'(l_y)\right) =\\ &= e_\theta\left(\omega - v\gamma(\chi)\right) = -v\kappa e_\theta^2 < 0, \quad \forall e_\theta \neq 0,\, v > 0.\end{aligned} \tag{8.2}$$

8.4 Probabilistic authority-sharing controller

The controller (6.2) ensures asymptotic tracking of the path in ideal conditions (i.e., the estimation error of χ is zero). Intuitively, if the estimation error is limited, controller (6.2) is expected to ensure that the path is followed with an error due to (8.2). However, if no landmark is detected, hence no absolute measure is available, the localization is affected by dead-reckoning of odometry and hence the estimation error grows potentially unbounded. Hence, the path following error grows as well. In this following section we will show the guard adopted to shift the control authority between the robot and the user and how this authority-sharing idea can be formally modeled using tools from hybrid systems [40].

8.4.1 Controller probabilistic analysis

To explain the rationale of the probabilistic analysis of the controller, let us specify what changes in (8.2) when the state χ is not known, the control input $\omega(\cdot)$ in (6.2) is computed using the available estimate $\hat{\chi}$, which is affected by the estimation error noise ε, i.e.,

$$\varepsilon = \begin{bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_\theta\end{bmatrix} = \begin{bmatrix}l_x - \hat{l}_x\\ l_y - \hat{l}_y\\ \tilde{\theta} - \hat{\tilde{\theta}}\end{bmatrix} = \chi - \hat{\chi}. \tag{8.3}$$

Using the Taylor expansion for the nonlinear functions in (6.2) about the estimated quantities, and recalling (8.3), one gets

$$\gamma(\chi) = c(s)\Lambda(\hat{\chi},\varepsilon) + \left(\delta'(\hat{l}_y) + \delta''(\hat{l}_y)\varepsilon_y\right) \left[-c(s)\Lambda(\hat{\chi},\varepsilon)(\hat{l}_x+\varepsilon_x) + \sin(\hat{\tilde{\theta}}) + \cos(\hat{\tilde{\theta}})\varepsilon_\theta\right] + \mathcal{O}(\varepsilon^2),$$

where $\delta'(\hat{l}_y) = \frac{\mathrm{d}\delta}{\mathrm{d}l_y}(l_y)\Big|_{l_y=\hat{l}_y}$ and $\delta''(\hat{l}_y) = \frac{\mathrm{d}^2\delta}{\mathrm{d}l_y^2}(l_y)\Big|_{l_y=\hat{l}_y}$, where, with a light abuse of notation, we denote with $\mathcal{O}(\varepsilon^2)$ high order error terms, and where

$$\Lambda(\hat{\chi},\varepsilon) = \cos(\hat{\tilde{\theta}}) - \sin(\hat{\tilde{\theta}})\varepsilon_\theta + \kappa_x\hat{l}_x + \kappa_x\varepsilon_x.$$

Hence

$$\gamma(\chi) = \gamma(\hat{\chi}) + H(s,\hat{\chi})\varepsilon + \mathcal{O}(\varepsilon^2), \tag{8.4}$$

where $H(s,\hat{\chi})$ is a row vector equals to

$$\begin{bmatrix} c(s)\kappa_x - c(s)\delta'(\hat{l}_y)\left(\cos(\hat{\tilde{\theta}}) + 2\kappa_x\hat{l}_x\right) \\ \left[-c(s)\left(\cos(\hat{\tilde{\theta}}) + \kappa_x + \hat{l}_x\right)\hat{l}_x + \sin(\hat{\tilde{\theta}})\right]\delta''(\hat{l}_y) \\ c(s)\sin(\hat{\tilde{\theta}})\left(\hat{l}_x\delta'(\hat{l}_y) - 1\right) + \cos(\hat{\tilde{\theta}})\delta'(\hat{l}_y) \end{bmatrix}^T.$$

We can therefore compute the first order approximation of (8.2) as

$$\dot{V} = e_\theta\dot{e}_\theta = e_\theta\left(\omega(\hat{\chi}) - v\gamma(\chi)\right) = e_\theta\left(v\gamma(\hat{\chi}) - \kappa v\hat{e}_\theta - v\gamma(\chi)\right),$$

and then, noticing that $\hat{e}_\theta = \hat{\tilde{\theta}} - \delta(\hat{l}_y)$ and that

$$e_\theta = \hat{\tilde{\theta}} + \varepsilon_\theta - \left[\delta(\hat{l}_y) + \delta'(\hat{l}_y)\varepsilon_y + \mathcal{O}(\varepsilon_y^2)\right] = \hat{e}_\theta + G(\hat{l}_y)\varepsilon + \mathcal{O}(\varepsilon_y^2),$$

where $G(\hat{l}_y) = [0, -\delta'(\hat{l}_y), 1]$, and then plugging (8.4), we finally have

$$\begin{aligned} \dot{V} &= e_\theta\left(v\gamma(\hat{\chi}) - \kappa v\hat{e}_\theta - v\gamma(\chi)\right) \\ &= \left(\hat{e}_\theta + G(\hat{l}_y)\varepsilon + \mathcal{O}(\varepsilon_y^2)\right)\left(v\gamma(\hat{\chi}) - \kappa v\hat{e}_\theta - v\gamma(\hat{\chi}) - vH(s,\hat{\chi})\varepsilon + \mathcal{O}(\varepsilon^2)\right) \\ &= -v\kappa\hat{e}_\theta^2 - v\hat{e}_\theta\left(\kappa G(\hat{l}_y) - H(s,\hat{\chi})\right)\varepsilon + \mathcal{O}(\varepsilon^2) \\ &= -v\kappa\hat{e}_\theta^2 - v\hat{e}_\theta\Xi(s,\hat{\chi})\varepsilon + \mathcal{O}(\varepsilon^2) \\ &= -v\kappa\hat{e}_\theta^2 + f(\varepsilon), \end{aligned} \tag{8.5}$$

where $f(\cdot)$ is a nonlinear function of the estimation error and $\Xi(s,\hat{\chi}) = \kappa G(\hat{l}_y) - H(s,\hat{\chi})$. It is evident that the negative definiteness cannot be established. Moreover, if the noise affecting the measures is Gaussian, ε could be unbounded, which rules out

standard techniques, such as proving the boundedness of the Lyapunov function [62]. More importantly, even if a bound can be determined, it is not given for granted that the human using the *FriWalk* could not do anything better. Instead, notice that the Lyapunov function derivative $\dot{V}$ in (8.5) is a random variable since it depends on ε.

Definition 1 (Controller reliability)**.** *Given $\Gamma \leq 0$, the reliability $p_\Gamma(\hat{\chi})$ of a control action $\omega(\hat{\chi})$ is given by probability*

$$p_\Gamma(\hat{\chi}) = \Pr \dot{V} < v\Gamma, \tag{8.6}$$

where $\Pr \dot{V} < v\Gamma$ denotes the probability that the event $\dot{V} < v\Gamma$ takes place.

The constant $\Gamma \leq 0$ is a minimum convergence speed that the controller is required to guarantee. Roughly speaking, the reliability $p_\Gamma(\hat{\chi})$ is the probability that the controller ensures at least such convergence speed. Scaling Γ by v is not strictly necessary but it comes handy since $\dot{V}$ is linear with respect to v as well. In fact, if the controller were deterministic as in (8.2), we would get $\dot{V} < v\Gamma \iff -v\kappa e_\theta^2 < v\Gamma \iff -\kappa e_\theta^2 < \Gamma$.

Using (8.5), it is now possible to compute a first order approximation of the mean value $\overline{\dot{V}}$ and of the standard deviation $\sigma_{\dot{V}}$, which are required to compute the controller reliability by Definition 1. Assuming as customary that $\mathrm{E}\{\varepsilon\} = 0$, we can readily have

$$\begin{aligned} \overline{\dot{V}} &= \mathrm{E}\left\{\dot{V}\right\} = -v\kappa\hat{e}_\theta^2 \\ \sigma_{\dot{V}}^2 &= \mathrm{E}\left\{\left(\dot{V} - \mathrm{E}\left\{\dot{V}\right\}\right)^2\right\} \\ &= \mathrm{E}\left\{v^2\hat{e}_\theta^2\Xi(s,\hat{\chi})\varepsilon\varepsilon^T\Xi(s,\hat{\chi})^T\right\} \\ &= v^2\hat{e}_\theta^2\Xi(s,\hat{\chi})P\Xi(s,\hat{\chi})^T. \end{aligned} \tag{8.7}$$

Under the assumption of Gaussian distribution, using mean value and covariance from (8.7), the probability (8.6) can be explicitly computed.

The idea proposed in this chapter is to allocate the control authority on the basis of the controller reliability (8.6). To intuitively describe this approach, we compare `case 1` and `case 2` in Figure 8.1. Suppose for simplicity that $\Gamma = 0$ in the definition of controller reliability (8.6). The mean value of $\dot{V}$ in `case 1` is smaller (i.e., larger convergence rate) than `case 2`, while its covariance is much larger than the covariance of `case 2`. This implies that the reliability of the controller is larger in `case 2`, since the probability to get $\dot{V} < 0$ is larger than `case 1`. Consider also `case 3`, where the covariance tends to infinity, i.e., absence of information. Since the controller reliability is in this case 0.5, any action the robot performs has 50% chance of reducing the attitude error e_θ.

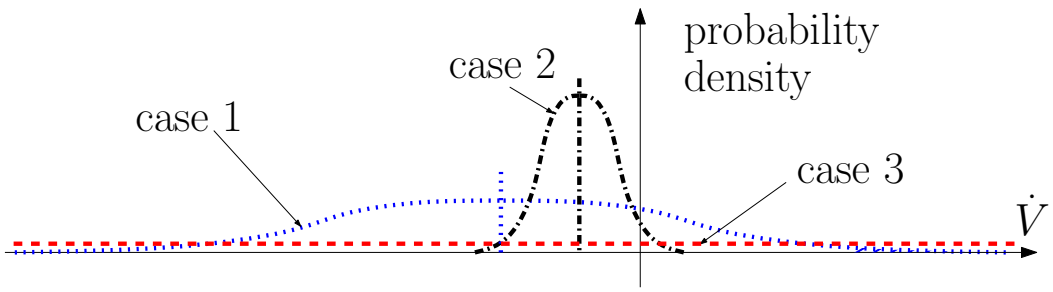

Figure 8.1: Examples of distributions of Lyapunov function derivative [72].

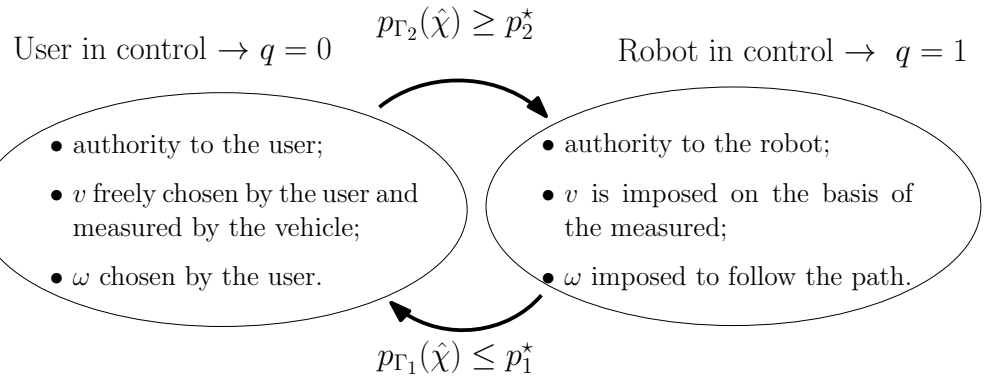

Figure 8.2: Control authority sharing of the hybrid controller (4.9) [72].

8.4.2 Allocation of the control authority

The control authority is shared with the user on the basis of the controller reliability, as shown in Figure 8.2. To properly implement a smooth transition, we define a hysteresis mechanism by formulating the control law as a hybrid system [40]. More in depth, let $q \in \{0, 1\}$ be a logic variable defining who retains the control authority. If $q = 0$ the controller reliability is small and then the user is in control of the vehicle, i.e., the vehicle actuators are not active (*user in control* state in Figure 8.2). While if $q = 1$ the controller reliability is large and hence the robot is in control (*robot in control* state in Figure 8.2 and the control action (6.2) is applied to steer the vehicle towards the path). The hysteresis is defined on the basis of two constants $\Gamma_2 > \Gamma_1 \geq 0$ representing convergence speed thresholds. Let $p_1^\star \in (0, 1)$ and $p_2^\star \in (0, 1)$, $p_1^\star \leq p_2^\star$, be the minimum tolerated reliabilities that, respectively, activate and disengage the controller. The overall controller is formalized as the following hybrid system having state $[e_\theta, q]^T$.

$$\begin{cases} \dot{q} & = 0, & [e_\theta, q]^T \in \mathcal{C}, \\ q^+ & = 1 - q, & [e_\theta, q]^T \in \mathcal{D}, \end{cases} \tag{8.8}$$

where $\mathcal{C} := \mathcal{C}_0 \cup \mathcal{C}_1$ and $\mathcal{D} := \mathcal{D}_0 \cup \mathcal{D}_1$ are the flow and the jump set respectively, where

$$\begin{aligned} \mathcal{C}_0 &= \{p_{\Gamma_2}(\hat{\chi}) \leq p_2^\star \ \wedge \ q = 0\}, \\ \mathcal{C}_1 &= \{p_{\Gamma_1}(\hat{\chi}) \geq p_1^\star \ \wedge \ q = 1\}, \\ \mathcal{D}_0 &= \{p_{\Gamma_2}(\hat{\chi}) \geq p_2^\star \ \wedge \ q = 0\}, \\ \mathcal{D}_1 &= \{p_{\Gamma_1}(\hat{\chi}) \leq p_1^\star \ \wedge \ q = 1\}. \end{aligned} \tag{8.9}$$

This way, the angular velocity of the vehicle is $\omega = qv\left(\gamma(\hat{\chi}) - \kappa\hat{e}_\theta\right) + (1-q)\omega_{\text{user}}$, where ω_{user} is the angular velocity that the user imposes when he/she has the control authority.

Remark 2. *Assume that perfect localization is available, i.e., no covariance on the Lyapunov function derivative $\sigma^2_{\dot{V}} = 0$. Notice that the discrete dynamics of the logic variable q in* (8.9) *is activated just by comparing the exact value of the Lyapunov function derivative $\overline{\dot{V}} = \dot{V} = -v\kappa e_\theta^2$ with the numeric thresholds $v\Gamma_1$ and $v\Gamma_2$. This is clearly equivalent to comparison between the norm of the attitude error $|e_\theta|$ and the numeric thresholds θ_{q_1} and θ_{q_2} used in the definition of the sharp authority allocation in the previous chapters of the thesis, e.g.,* (4.9) *(indeed writing $-v\kappa e_\theta^2 \leq v\Gamma_1$ is equivalent to write $|e_\theta| \leq \theta_{q_1}$ with $\theta_{q_1} = \sqrt{-\Gamma_1/\kappa}$ and $v \neq 0$). As a consequence, the control algorithm presented in this chapter can be interpreted as the stochastic generalization of the sharp authority allocation of the previous chapters.*

8.5 Simulation results

The proposed controller has been extensively tested in simulations with different choices of the controller tuning parameters. In the results here reported, those parameters are set as follows: $\kappa_x = 1$ and $\kappa = 0.5$ in (8.5), $p^\star = p_1^\star = p_2^\star = 0.9$ in (4.9), and $\Gamma_1 = -0.03$ and $\Gamma_2 = -0.24$ for the thresholds of Definition 1. Since Γ_1 and Γ_2 are compared with $-\kappa\hat{e}_\theta^2$ in (8.9), the corresponding mean tolerated attitude errors are 15° and 40°, respectively. The implemented localization algorithm computing (3.1) is an extended Kalman filter fusing the odometric data with the absolute position measure from the landmarks [88]. The landmarks are deployed following [87] to ensure that at least one marker is always in the field of view of the camera (depicted with squares in Figure 8.3). The landmark reading uncertainty is 10° for the vehicle orientation and 10 cm for the position. The uncertainty due to encoders is of 13 mm per wheel revolute.

Recall that the underlying assumption of the proposed solution is that the path following performance of the proposed solution depends on the ability of the user to follow the path when the uncertainty grows. In fact, if the user is cooperative, rely on her/him is quite rewarding, while if the user is completely uncooperative (i.e., he/she constantly moves away from the path on purpose), the path following error grows. Notably, the user behavior cannot be known in advance (and also, it is a challenging

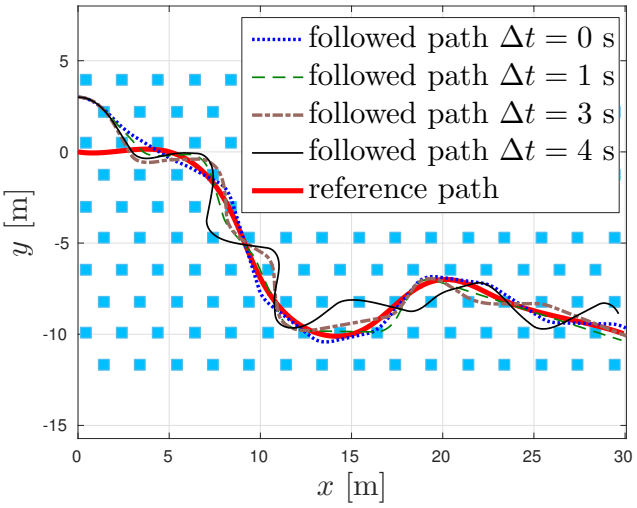

Figure 8.3: Paths followed for different reading periods Δt. The squares represent the landmark positions [72].

problem to define a suitable "cooperativeness" measure). Nevertheless when the path following error grows, the controller reliability as per Definition 1 grows as well, thus limiting the deviation from the planned path. In the simulations, when the user has the control authority, he/she is modeled with a neutral behavior, that is he/she pushes the vehicle forward (i.e., $\omega_{\text{user}} = 0$). Figure 8.3 shows the paths followed by the robot varying the landmark reading time interval Δt. Notice that if Δt is small (0, that is continuous reading, or 1 second) the vehicle is maintained close to the path, which is a trivial consequence of the small covariance (3.1) due to frequent landmarks readings: as a consequence, the robot remains in control most of the time. If Δt is larger (3 or 4 seconds), the controller reliability is, on average, smaller. In other words, the control authority is given to the robot only when the reliability exceeds the threshold $p^\star$, which happens for larger mean values of $\dot{V}$ (i.e., for larger attitude errors $|e_\theta|$, see (8.7)). Further simulations are presented in Figure 8.4, where the influence of landmark reading time Δt on the norm of the orientation error is shown in a probabilistic sense. For each Δt, 100 simulations are executed. Notice that, the larger Δt, the larger the attitude error e_θ, since the user is endowed with more control authority in the presence of large uncertainty. A similar behavior is obtained for the worst case distance to the path (see Figure 8.4, bottom plot).

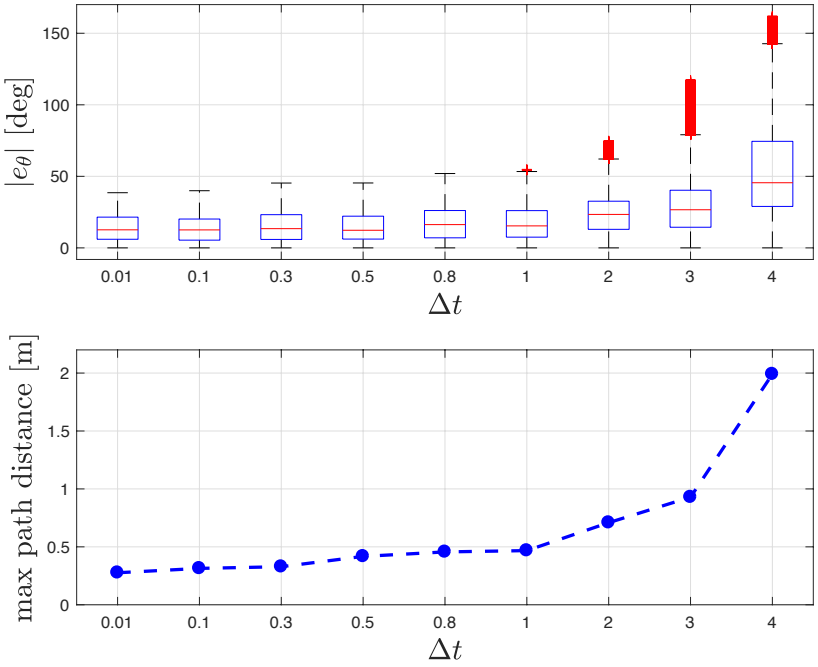

Figure 8.4: Box and whiskers plot of $|e_\theta|$ (top) and maximum path following error (bottom) for Montecarlo simulations with growing reading periods [72].

8.6 Experimental results

The experimental results have been collected using the *FriWalk*. The controller parameters adopted in the experiments are: $\kappa_x = 1$, $\kappa = 0.5$, $p_1^\star = 0.7$, $p_2^\star = 0.9$, $\Gamma_1 = -0.004$ and $\Gamma_2 = -0.137$. With respect to the simulation results in Section 8.5, the probability $p_1^\star$ has been reduced to give more authority to the user to increase the comfort. Similarly, both the mean tolerant attitude errors Γ_1 and Γ_2 have been reduced to 5° and 30°, respectively.

The experimental scenario is the Department of Information Engineering and Computer Science of the University of Trento, comprising corridors and rooms in Figure 8.5. The starting point, of the *FriWalk* is inside one room, represented with a blue circle in Figure 8.5. The landmarks are placed only in proximity of difficult decision points, i.e., landmark #1 is in the starting room in the vicinity of the exit door, landmark #2 has been collocated at the beginning of the corridor, while landmark #3 is deployed before two intersecting corridors. In the corridor, due to the particular desired path considered (dash-dotted black line of Figure 8.5), has no landmark since the only available choice is to maintain on the course. The depicted yellow solid triangle pointing forward represents the field of view of the camera attached to the vehicle and used to detect the landmarks, while the dotted blue

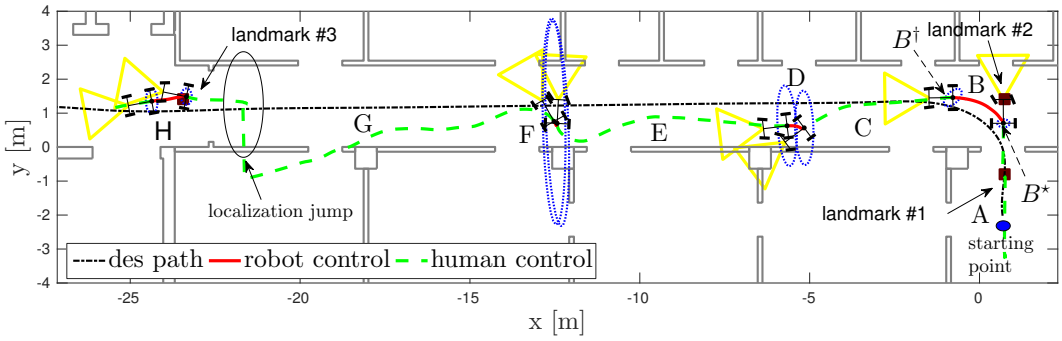

Figure 8.5: A sample trajectory of the experimental trial with localization covariance depicted in selected points. The picture reports the desired path (dash-dotted line) and the estimated trajectory obtained by the localization algorithm (dashed line). This trajectory is divided into sub-paths for reading easiness [72].

ellipses represent the localization error covariance P_{xy} (upper 2×2 matrix of (3.1)) in selected positions. To better analyze the experiments, the path is divided in the following parts:

Sub-path A: the user is in control of the robot ($q = 0$ in (4.8)) and pushes the *FriWalk* outside the room since the localization error is very high (i.e., kidnapped robot problem, dashed green line in Figure 8.5).

Sub-path B: when the vehicle detects a landmark in position $B^\star$, $p_{\Gamma_2}(\hat{\chi}) > p_2^\star$ and the controller (4.9) enters in the jump set $\mathcal{D}_1$ so that $q \to 1$. The robot is hence in control ($q = 1$ in Figure 8.2). The Gaussian probability density function (pdf) of $\dot{V}$ in point $B^\star$ is reported with dash-dotted black line in Figure 8.6. During the *robot in control* mode, ω is imposed by the control law and steers the walker toward the desired path (red solid line in Figure 8.5). At point $B^\dagger$, $p_{\Gamma_1}(\hat{\chi}) < p_1^\star$ and the authority is given back to the user since $q \to 0$ (the solid green Gaussian pdf in Figure 8.6).

Sub-path C: in this section the user is in control and the covariance grows (no landmark detected), hence the pdf flattens, so that it is more difficult for the controller to kick in. Nonetheless, at the end of sub-path **C**, the orientation error becomes so large (indeed, $\dot{V}$ is a quadratic function of $\hat{e}_\theta$ (8.5)) that $p_{\Gamma_2}(\hat{\chi}) \geq p_2^\star$ and the controller intervenes to align the user toward the path.

Sub-path D: due to the shape of the Gaussian, which is flatter than in sub-path **B**, it takes a smaller time to reach the condition $p_{\Gamma_1}(\hat{\chi}) < p_1^\star$. However, the user receives the input to realign towards the desired path.

Sub-path E: the user has the possibility to move freely since the covariance of the localization error is very large. At the end of this sub-path the user tried to perform an U-turn, but the controller does not allow this maneuver as at the end of Sub-path **C**.

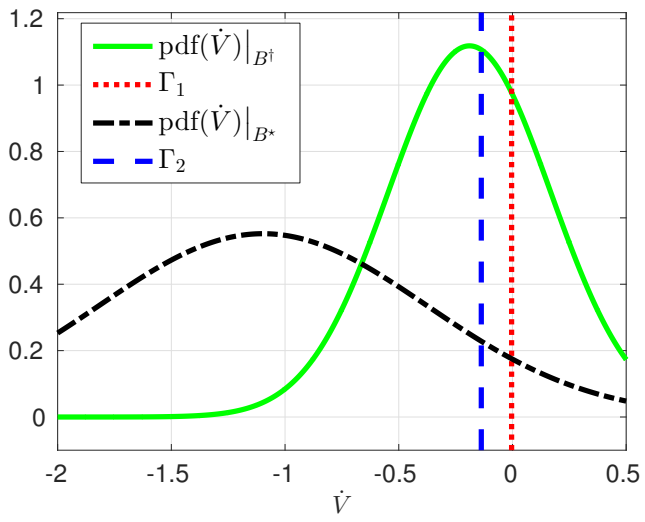

Figure 8.6: Distribution of $\dot{V}$ at the beginning of section $B \to B^\star$ (dash–dotted black line) and at the end of section $B \to B^\dagger$ (solid green line) [72].

Sub-path F: the same of sub-path **D**, but even shorter.
Sub-path G: from the beginning of this sub-path, no landmark is in view since 12.5 meters, so that the uncertainty grows unbounded. Notice that the walker wrongly localizes itself through a wall, which is obviously not true: however, if the *robot in control* was active, the vehicle would be guided over the desired path and hence, aligning the green dashed line over the dash-dotted desired path, the *FriWalk* would be steered towards the wall on the other side of the corridor. Instead, after landmark #3 is detected and the uncertainty drops, it can be seen that the vehicle was correctly very close to the path, guided by the user.
Sub-path H: finally, the controller takes the control of the robot since $p_{\Gamma_2}(\hat{\chi}) \geq p_2^\star$.

8.7 Final comments

In this Chapter, we have applied the authority-sharing paradigm in a stochastic framework, to face the localization loss coming from dead-reckoning. The probabilistic sharing provides the user with control authority if he/she can act more reliably (in a probabilistic sense) than the robot. Since the user is typically reliable in environments where the direction of motion is clear (e.g., corridors), a precise localization (hence dense landmark deployment) is needed only in the proximity of complex decision points. Extensive simulations and preliminary experiments validates the proposed approach.

There are several open problems that deserve future investigation. From the theoretical point of view, the most interesting problem that needs to be addressed is

to offer "certifiable" performance guarantees based on the knowledge of the vehicle and the environment. Another important goal is to test the idea with a number of actual users and carry out a quantitative and qualitative study on their performance and impressions.

Chapter 9

Authority-sharing in multi-vehicle human groups

The problem of coordinating a group of senior users assisted by robotic walkers is considered. A force field-based control strategy is proposed to guide the users in complex dynamic environments. In particular, the well-known social force model is used to define the human-like reference motion for the robots, i.e., the headings that the assistive vehicles should have to safely move in the environment towards the goal in a way similar to humans. Hence, if the user properly steers his/her assistive vehicle, he/she is left in control of the robot. An attitude controller sets a range for the possible headings that allow the vehicles to move in accordance with the reference motion avoiding collisions. As far as the users choose a direction of motion within this range, they are left free to drive their walkers without any interference. Therefore, the control authority is shared between the user and the robot. We also propose an important improvement on the force field to include a novel dynamic obstacle avoidance strategy based on oval shaped limit cycles. The asymmetry of the oval generates paths that are safer and more comfortable than the paths generated by the the circular limit cycles (customary in the literature). The proposed approach is validated through a massive amount of simulation results in very realistic environments and in the presence of dynamic and uncooperative objects following real human trajectories.

9.1 Introduction to social force control

Providing elders with a good navigation support is helpful, but not sufficient to persuade them to move out of their home. Social relations can be very important [57]: a walk in a large public space is arguably more appealing in company of trusted friends than if the user goes alone. If all the users use a robotic walker, a good guidance solution is one that: 1. preserves the autonomy of each user (e.g., to change the person he/she wants to have close), 2. guides the users to their destinations

without loosing the group coherence.

In the field of distributed robotics, the problem of cooperative motion of a team of robots towards a common direction is known as flocking. In [89], a control law based on potential fields is used to steer a group of agents towards a common direction by forcing the robots to converge to a lattice structure. Non-holonomic behaviors are considered in [91] and [83], where the flocking problem is solved using nonlinear control that operates on the angular velocity of the agents leaving the forward velocity unchanged. In [115], flocking with collision avoidance is obtained using a potential field approach for non-holonomic robots. A common idea to these papers is to achieve non-holonomic flocking by aligning all the robots along a common orientation as quickly as possible in order to avoid collisions (attitude synchronization). All subsequent maneuvers maintain the same orientation for all the agents, thus achieving flocking and collision avoidance. This strategy is not applicable to our context for two reasons: 1. a quick convergence to the same orientation requires setting high angular velocity and could compromise the user's balance, 2. maintaining the same orientation for all the agents (or imposing a lattice structure) would completely deprive the users of their freedom inside the group.

There are different models in the literature that guarantee cohesion of human groups without forcing a rigid positioning or the same direction of motion for all the components. In the social force model (SFM) [46], each human is modelled as a particle subjected to attractive and repulsive forces. The SFM is very popular in several applications, such as detecting and localizing anomalies in the behavior of a crowd from a video stream [80]. In [35] the SFM is used to determine the behavior of a team of robots, which guide a group a pedestrian towards a desired destination. Despite its persistent popularity, the SFM has important limitations, which have suggested several modifications and improvements. An example is offered by the headed social force model (HFSM) [32], which modifies the SFM by inserting a non-holonomic behavior that models the fact that humans have a preferred direction of motion (i.e., they prefer headway over sideways motions). Importantly, the HFSM explicitly considers the effect of cohesion in group motion. The SFM and its derivations have been validated by different researchers using experiments and field trials.
This chapter applies a force field used to model human motion to control a group of robotic walkers. Each human in the group is guided by his/her walker along trajectories that are as close as possible to the one that she/he would follow inside the group. In particular, the field generated by the SFM [46] is used to identify the reference heading that the vehicle has to follow in order to move toward the goal in a safe and human-like way. If the user drives by maintaining an orientation close to the reference, he/she is left in control of the robot. An attitude controller just restricts the set of headings that each user can choose in order to avoid collisions

and move towards the goal. This policy qualifies our technique as an example of authority-sharing. The adopted force field defining the reference motion also considers the group cohesion force as defined in the HSFM [32].
The chapter also improves the dynamic obstacle avoidance abilities of the SFM based by applying a limit cycle generation on top of the repulsive forces between the agents. This approach is based on a nonlinear differential equation, whose solutions converge to a closed trajectory encircling the obstacle. Our limit cycles are based on an innovative oval shape, whose asymmetric structure offers several improvements with respect to the existing circular shapes.

9.2 Social force control framework

Each vehicle in the group is modeled as a unicycle robot, having dynamics

$$\begin{cases} \dot{x} &= v\cos\theta, \\ \dot{y} &= v\sin\theta, \\ \dot{\theta} &= \omega, \\ \dot{v} &= a, \end{cases} \tag{9.1}$$

where (x, y) is the position of the mid point of the rear axle in the plane of motion, θ is the vehicle yaw, v is the scalar forward velocity. a is the acceleration applied to the vehicle, and ω is the scalar angular velocity.

The acceleration a and angular velocity ω act as control inputs and they can be applied either by the user or by the actuation system. If the actuation system is not activated, the vehicle acts as a fully passive rollator moving in accordance with the inputs of the user, who is in control of the motion. Conversely, if the actuators are engaged, the robot is in control of the motion. This simple kinematic model, already proved accurate enough for assistive robots, is also used in the literature to analyze vehicle flocking [115].

9.2.1 Vehicle working modes

In order to avoid operating conditions that could compromise the user's balance, the vehicle is not allowed to pull the user; hence, the only possible actuation for controlling linear motions is by brakes. If the braking system is not activated, the user can push the vehicle to the desired velocity. Conversely, if the braking system is active, the forward velocity is reduced. We model the overall behavior as

$$\begin{aligned} &a = a_{\text{robot}}, \quad &&\text{if the brakes are engaged,} \\ &a = a_{\text{user}}, \quad &&\text{if the brakes are not engaged,} \end{aligned} \tag{9.2}$$

where $a_{\text{robot}} \leq 0$ is the braking acceleration applied by the actuation system, while $a_{\text{user}} \geq 0$ is the acceleration applied by the users when the brakes are disengaged.

Since the user is standing behind the vehicle, backward motion is not allowed for safety reasons: $a_{\text{robot}} = 0$ if $v \leq 0$. The brakes are designed to completely stop the vehicle regardless of the human input (e.g., to perform emergency braking [5]); therefore, equation (9.2) does not consider the user's input when the actuators are engaged, i.e., when the braking system is active, the forward velocity v always decreases. Notice that relation $a_{\text{user}} \geq 0$ assumes that each user is propelling his/her vehicle. When the user is in control of the forward motion, we assume that his/her input $a_{\text{user}} \geq 0$ is limited and so is the velocity that is imposed on the vehicle.

The actuation system can be activated or deactivated to act on the vehicle angular motion. Similarly to the forward motion, if the robot is passive, the user is in control of the motion and can freely steer the vehicle. Conversely, when the actuation system is active, the angular velocity is set by the robot and the user is guided. In particular, we have

$$\begin{aligned} &\omega = \omega_{\text{robot}}, \quad \text{if the actuators are engaged,} \\ &\omega = \omega_{\text{user}}, \quad \text{if the actuators are not engaged,} \end{aligned} \tag{9.3}$$

Overall, combining the all the possibilities, we define the following working modes:
User in control mode: the user is in control of both the linear and the angular motion. The acceleration applied to the vehicle is then determined by the user via $\dot{v} = a_{\text{user}}$ in (9.2), and so is the angular velocity $\omega = \omega_{\text{user}}$ in (9.3);
Robot in control mode: the robot is in control of both the linear and angular motion. The user's inputs are completely rejected and the motion is totally determined by the actuators: $\omega = \omega_{\text{robot}}$ in (9.3) and $\dot{v} = a_{\text{robot}} \leq 0$ in (9.2).
Braking mode: the robot is in control of the linear motion, while the user is in control of the angular velocity: $\omega = \omega_{\text{user}}$ chosen by the user in (9.3) and $\dot{v} = a_{\text{robot}} \leq 0$ in (9.2). This operation mode is obtained by applying the same braking torque to the right and left wheels.

9.3 Navigation via potential fields

Our work is based on the use of potential fields for navigation [103], which we shortly summarize here.

9.3.1 Holonomic robot

Suppose that a holonomic robot having position $p = [x, y]^\top$ has to reach a goal position p_g. A potential guidance is based on a scalar potential function

$$U : \mathbb{R}^2 \to \mathbb{R},$$

having the following properties:

1. $U(\cdot)$ is differentiable and positive definite about p_g, i.e., $U(\bar{p}) \geq 0, \forall \bar{p} \in \mathbb{R}^2$ and $U(\bar{p}) = 0 \iff \bar{p} = p_g$. As a consequence, the potential function has its minimum at the goal position, i.e.,

$$\arg\min_{\bar{p}} U(\bar{p}) = p_g;$$

2. the potential function has no other local minima, i.e.,

$$\nabla U(\bar{p}) = 0 \iff \bar{p} = p_g. \tag{9.4}$$

If the velocity of the holonomic robot can be chosen, i.e., the robot has dynamics

$$\dot{p} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} u_x \\ u_y \end{bmatrix},$$

where the velocities u_x and u_y are control inputs, the control law steering the robot to the goal is

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = -\nabla U(p).$$

An example of potential field U and its negative gradient $-\nabla U$ is reported in figures 9.1 and 9.2. The function is designed to assume large values in the neighborhood of obstacles to avoid collisions, while its minimum is located in the goal position $p_g = [1, -1]^\top$. This way, since the robot always move to reduce the potential, collisions with obstacles never occur by construction.

Although constructing a potential fields with no local minima is almost impossible for environments with arbitrarily-shaped obstacles, in the practice, in standard planar indoor environments, potential fields are widely used for robot navigation since the local minima problem is easily bypassed via techniques such as vortex fields [30] and waypoint design [103].

9.3.2 Robot with heading as control input

Consider a robot having dynamics

$$\dot{p} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \end{bmatrix} = vh,$$

where $h = [\cos\theta, \sin\theta]^\top$ is the heading of the robot acting as control input. In practice, the robot moves in the chosen direction h at forward speed v. In this set-up, the navigation to the goal is still possible by ensuring that the heading of the robot h and the heading of the gradient

$$h_U = -\frac{\nabla U(p)}{|\nabla U(p)|}, \tag{9.5}$$

form an angle smaller than 90° (i.e., $h \cdot h_U > 0$, where the notation $h \cdot h_U$ denotes the scalar product between h and h_U), as stated by the following theorem.

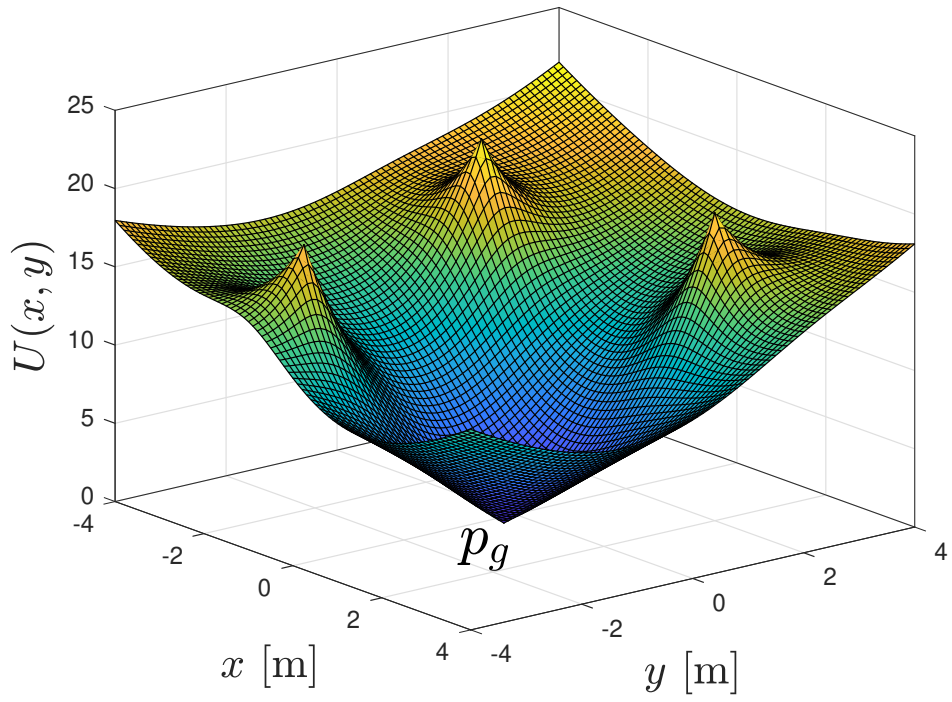

Figure 9.1: Example of potential field.

Theorem 10. *Let $U : \mathbb{R}^2 \to \mathbb{R}$ a potential field satisfying the properties in Section 9.3.1. If the vehicle heading h satisfies for all time*

$$h \cdot h_U \in (0, 1], \tag{9.6}$$

where h_U is the heading of the gradient defined in (9.5)*, then the robot reaches a neighborhood of the goal position, i.e., $\forall \varepsilon > 0$ we have*

$$\lim_{t \to \infty} |p - p_g| < \varepsilon.$$

Proof of Theorem 10. First notice that the heading h_U of the field is well defined if $|p - p_g| \geq \varepsilon$ since $\nabla U(p) = 0 \iff p = p_g$. The time evolution of the potential function $U(p)$ is given by

$$\dot{U}(p) = \nabla U(p) \cdot \dot{p} = -v |\nabla U(p)| h_U \cdot h. \tag{9.7}$$

We interpret the potential function U as a Lyapunov function. We have $\dot{U}(p) < 0$, $\forall p : |p - p_g| \geq \varepsilon$ by assumptions (9.6) and (9.4), hence the proof. □

9.3.3 Robot with angular velocity as control input

Consider a robot having dynamics (9.1). Theorem 10 ensures that if the robot is properly oriented as in (9.6), it arrives at the goal position. Then, we design

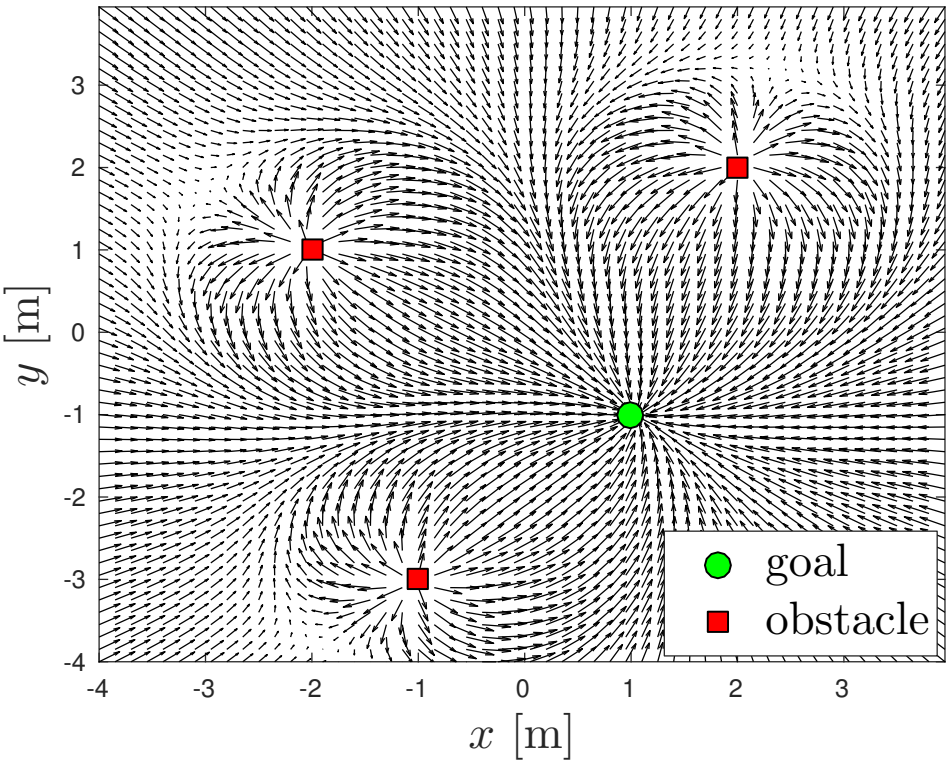

Figure 9.2: Gradient of the potential field in Figure 9.1.

the angular velocity control input ω in (9.1) to ensure that the vehicle heading $h = [\cos\theta, \sin\theta]^\top$ converges to the heading of the field h_U in (9.5).

Angular velocity command

The following Lyapunov function is considered to measure the distance between the vehicle heading h and the reference direction h_U:

$$V = 1 - h \cdot h_U. \tag{9.8}$$

Notice that $V \in [0, 2]$, and $h_U = h \iff V = 0$. For convenience of notation we write the components of the unit vectors as $h = [h^x, h^y]^\top$ and $h_U = [h_U^x, h_U^y]^\top$. It is recalled that the dynamics of the vehicle heading h is

$$\dot{h} = \begin{bmatrix} \dot{h}^x \\ \dot{h}^y \end{bmatrix} = \omega \begin{bmatrix} -h^y \\ h^x \end{bmatrix}. \tag{9.9}$$

Theorem 11 (Attitude controller)**.** *Consider the vehicle heading* (9.9) *and the error measure* (9.8)*. Suppose that the initial condition satisfies* $h(0) \cdot h_U \neq -1$ *and that the*

field heading h_U is constant. Then almost global stability is ensured by the control law

$$\omega = -\kappa\, V^{\alpha}\, \mathrm{sign}(h_U^x h^y - h_U^y h^x), \tag{9.10}$$

where $\alpha \in (0, \frac{1}{2})$ and $\kappa > 0$. Moreover the convergence takes place in finite time.

Proof of Theorem 11. Straightforward computations show that the derivative of the Lyapunov function (9.8) is given by

$$\dot{V} = \omega(h_U^x h^y - h_U^y h^x).$$

The control law is rewritten as

$$\omega = -\kappa\,(1 - h \cdot h_U)^{\alpha}\, \mathrm{sign}(h_U^x h^y - h_U^y h^x),$$

to get

$$\dot{V} = -\kappa\,(1 - h \cdot h_U)^{\alpha}\, \left|h_U^x h^y - h_U^y h^x\right| \leq 0. \tag{9.11}$$

Notice that $\dot{V} = 0 \iff \left|h_U^x h^y - h_U^y h^x\right| = 0 \iff V \in \{0, 2\}$ since $1 - h_U \cdot h = 0 \implies \left|h_U^x h^y - h_U^y h^x\right| = 0$. Since the assumption $h(0) \cdot h_U \neq -1$ ensures $V \neq 2$, then the only possible solution satisfying $\dot{V} = 0\, \forall t \geq 0$ provides $V = 0$, hence asymptotic stability follows from standard La Salle arguments. To prove that the convergence takes place in finite time, recall the identity $|h \times h_U| = |h_U^x h^y - h_U^y h^x| = \sqrt{1 - |h \cdot h_U|^2}$, where the notations $h \times h_U$ denotes the cross product between the vectors h and h_U. From asymptotic stability it follows that $V < 1$ holds in finite time. Since $V < 1 \implies 0 < h \cdot h_U \leq 1$, we have that there exists a finite time instant $t_1 > 0$ such that the condition

$$|h \times h_U| = \sqrt{1 - |h \cdot h_U|^2} \geq \sqrt{1 - |h \cdot h_U|},$$

holds for all $t \geq t_1$. Then, for all $t \geq t_1$, the Lyapunov function derivative (9.11) satisfies

$$\dot{V} = -\kappa V^{\alpha} |h \times h_U| \leq -\kappa V^{\alpha} \sqrt{1 - |h \cdot h_U|} = -\kappa V^{\alpha + \frac{1}{2}}.$$

Since $\alpha \in (0, \frac{1}{2})$ we have $\alpha + \frac{1}{2} < 1$, and finite time convergence follows from standard nonlinear system argumentation. □

Braking command

Control law (9.10) is used to steer the vehicle heading h towards the field heading h_U as in (9.6). However, two issues have to be considered:

1. Theorem 11 ensures the tracking of a constant reference h_U, while, if the vehicle moves (i.e., $v \neq 0$) h_U varies since the position p of the vehicle varies. In particular, the larger the vehicle velocity v, the larger the tracking error for law (9.10);

2. controller (9.10) requires some amount of time to properly orient the vehicle heading as in (9.6). During the transient, if relation (9.6) does not hold, collisions with obstacles may occur.

To overcome these issues, we reduce the forward velocity v of the vehicle via the braking action

$$a = a_{\text{robot}} = -\kappa_b \, v, \tag{9.12}$$

where $\kappa_b > 0$ is a gain. Intuitively, by choosing a braking gain κ_b sufficiently large, it is possible to reduce the tracking error for law (9.10) (since the derivative of h_U is reduced for small velocities). Moreover, collisions do not occur since condition (9.6) is satisfied in a time smaller than the time needed to hit an obstacle.

9.4 Social force model guidance

The social force model is used to describe the natural motion of human groups [46]. Each human is modeled as a particle subjected to attractive and repulsive forces. The model is widely used in several applications and shown to be very realistic via experimental data. The main forces acting on each human are depicted in Figure 9.3 and described in the following:

- attractive force to the goal. Given a waypoint to reach $[x_{\text{wp}}, y_{\text{wp}}]$, the attractive force F_a is defined as
$$F_a = -\nabla U_a(x, y),$$
where
$$U_a(x, y) = \kappa_a d_{\text{wp}},$$
where $d_{\text{wp}} = \sqrt{(x - x_{\text{wp}})^2 + (y - y_{\text{wp}})^2}$ is the distance between the human located in position $[x, y]^\top$ and the waypoint, and κ_a is a positive parameter.

- repulsive force from the obstacles. Given the closest point of the obstacle $[x_{\text{obs}}, y_{\text{obs}}]$, the repulsive force F_{obs} is defined as
$$F_{\text{obs}} = -\nabla U_{\text{obs}}(x, y),$$
where
$$U_{\text{obs}}(x, y) = U_0^{\text{obs}} \, e^{-\frac{d_{\text{obs}}}{R_{\text{obs}}}},$$
where $d_{\text{obs}} = \sqrt{(x - x_{\text{obs}})^2 + (y - y_{\text{obs}})^2}$ is the distance between human and obstacle, and U_0^{obs} and R_{obs} are positive parameters.

- repulsive force from the other humans in the group. Each agent in the group located in position $[x_h, y_h]^\top$ applies a repulsive force to the human located in position $[x, y]^\top$ defined as
$$F_h = -\nabla U_h(x, y),$$

where

$$U_h(x,y) = U_0^h \, e^{-\frac{d_h}{R_h}},$$

where $d_h = \sqrt{(x-x_h)^2 + (y-y_h)^2}$ is the distance between the two agents, and U_0^h and R_h are positive parameters.

- cohesion force [32]. This force maintains the compactness of the group by attracting the far agents to the centroid of the group. Let $[x_{\text{centr}}, y_{\text{centr}}]^\top$ be the position of the centroid. The cohesion force F_c is defined as

$$F_c(x,y) = \begin{cases} 0 & \text{if } d_c \leq d_1, \\ F_c^{\max} \dfrac{d_c - d_1}{d_2 - d_1}; & \text{if } d_1 \leq d_c \leq d_2, \\ F_c^{\max} & \text{otherwise,} \end{cases}$$

where $d_c = \sqrt{(x-x_{\text{centr}})^2 + (y-y_{\text{centr}})^2}$ is the distance from the centroid, $d_2 > d_1 > 0$ and $F_c^{\max}$ are positive parameters. Notice that the cohesion force is not active if the centroid is close ($d_c \leq d_1$), it reaches its maximum $F_c^{\max}$ if centroid is far ($d_c \geq d_2$), and it has a linear behavior for intermediate distances ($d_1 \leq d_c \leq d_2$).

In this work, the total force generated by the social force model is used to define the desired heading of the vehicle h_U.

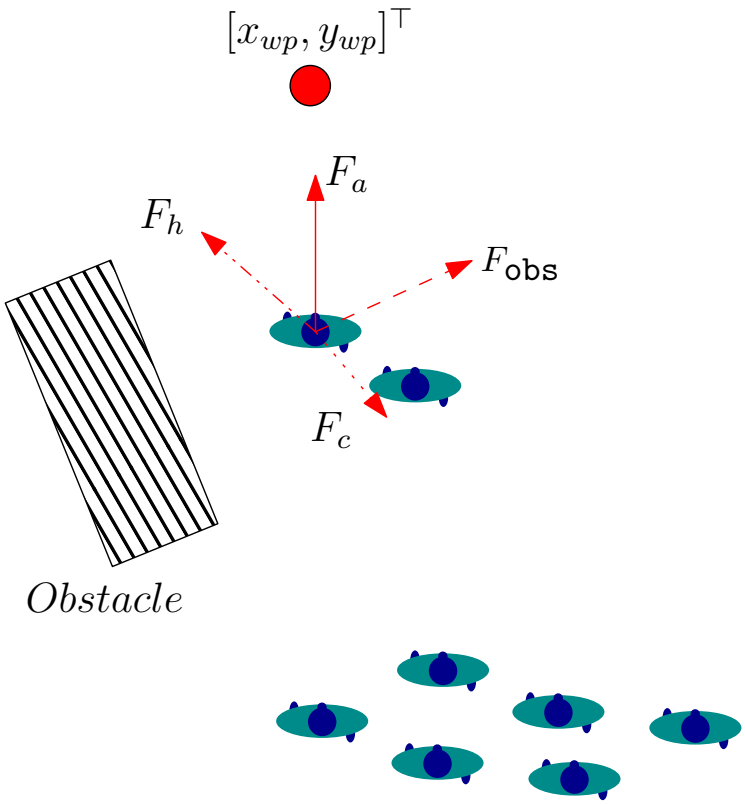

Figure 9.3: Forces considered in the headed social force model [7].

9.5 SFM improvement: avoiding local minima

Since the SFM generates a field that we use for guidance, the presence of local minima may theoretically hinder the robot's convergence to the destination. This issue is not considered when the SFM is used to analyze the motion of crowds, but it can be relevant for our SFM-based control.

We propose to modify (Sections 9.5.1 and 9.5.2) and classify (Section 9.5.3) the standard force field generated by the SFM in order to avoid local minima and to make the force field suitable for control.

9.5.1 Multiple waypoints

Local minima may occur if an obstacle is between the robot and the goal, since the attractive force of the goal is balanced by the repulsive force of the obstacle (see Figure 9.4). To overcome this issue, we provide the robot with a sequence of waypoints generated by a planning algorithm (e.g., RRT, RRT*, A* or other planning approaches) and connecting its current position with the goal. For instance,

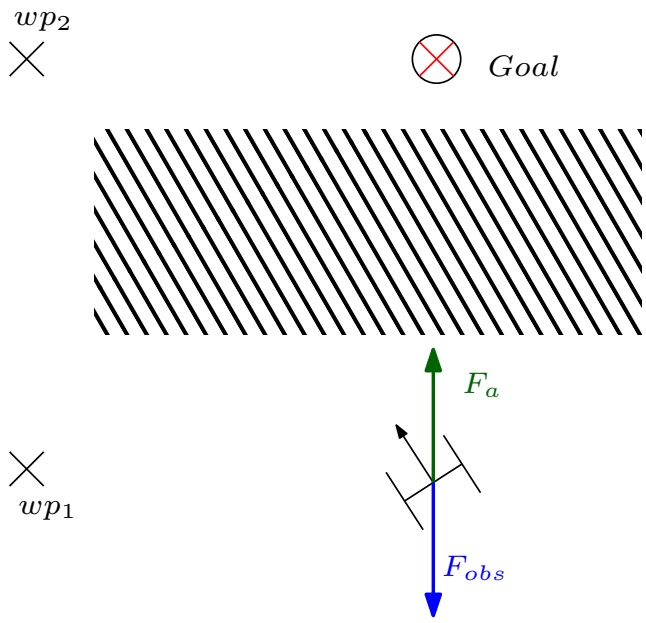

Figure 9.4: Example of local minimum generated by the obstacle and use of waypoints to reach the goal [7].

the robot in Figure 9.4 is not attracted by the goal but by the first waypoint wp_1. Once the first waypoint is reached, the robot is attracted by the second waypoint wp_2. The third waypoint coincides with the goal. This way, the robot avoids local minima and reaches the goal.

9.5.2 Vortex fields

A further strategy to avoid local minima is the use of vortex fields. In general, obstacles generate repulsive forces oriented along the line connecting robot and obstacle. A vortex field generates a force orthogonal to this line, suggesting the robot to pass around the obstacle (see Figure 9.5). Given the repulsive force F_{obs} generated by an obstacle, a vortex force is generated by rotating the repulsive force of 90° as

$$F_{\text{vortex}} = \begin{bmatrix} \cos(\pm 90°) & -\sin(\pm 90°) \\ \sin(\pm 90°) & \cos(\pm 90°) \end{bmatrix} F_{\text{obs}},$$

where the sign of the rotation is used to distinguish counter clockwise rotation (positive sign) from clockwise rotation (negative sign). We choose the direction of rotation such that the scalar product between the vortex force F_{vortex} and the vector pointing from robot to waypoint is positive (see Figure 9.6). The rationale is that this orientation ensures that the vortex force does not turn the robot away from the attractive waypoint when it approaches the obstacle.

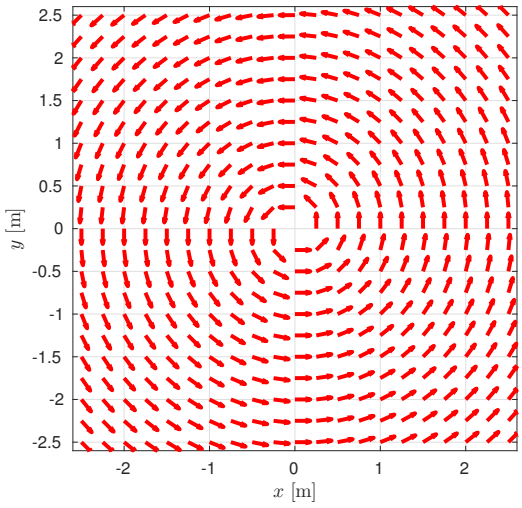

Figure 9.5: Example of counter clockwise vortex fields generated by an obstacle located in the origin [7].

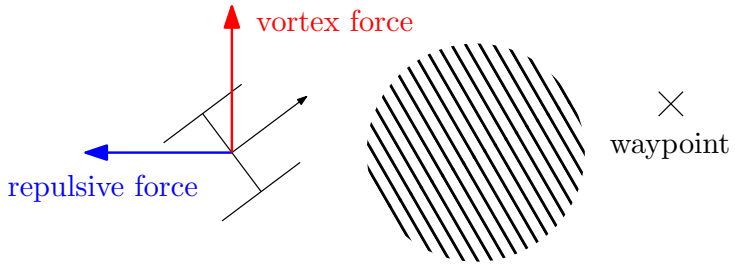

Figure 9.6: Orientation of vortex fields [7].

9.5.3 Permanent and transient forces

Although the use of multiple waypoints and vortex fields generally makes the robots to avoid local minima generated by the repulsive force of the obstacles F_{obs}, it may happen that a local minimum is generated by the repulsive force F_h exerted by other agents. An example is reported in Figure 9.7, where both vehicles are attracted by the waypoint inside a corridor, but the front vehicle is still. In this case, the rear vehicle cannot pass around the front vehicle because of the walls. Moreover the attractive force F_a of the waypoint is balanced by the repulsive force of the front agent F_h, hence the total force on the rear vehicle is zero. Since both vehicles are moving towards the waypoint, it is reasonable to suppose that the front user will restart his/her motion, hence that the repulsive action F_h on the rear vehicle will decrease. Conversely, the attractive force F_a of the waypoint will remain and hence the motion of the rear vehicle will restart. As a consequence, a reasonable action of

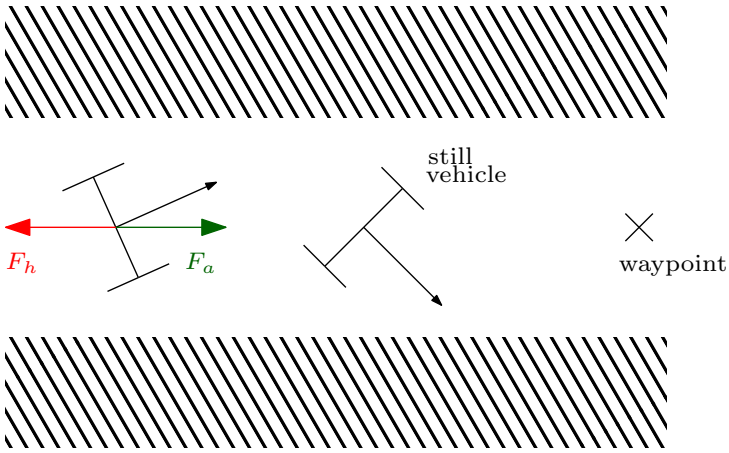

Figure 9.7: Example of vehicle waiting because of a minimum [7].

the rear agent is to wait that the front agent moves away. With the reference to the example in Figure 9.7, we define two categories of forces:

- *permanent forces*, i.e., the forces generated by static obstacle (known in the map) and waypoints. These forces are permanent since they are established offline given the map and the goal. Given a generic point in the map, the permanent force is constant in time. The permanent forces are then the attractive force of the waypoints F_a, the repulsive force of the obstacle F_{obs}, and the vortex fields F_{vortex} of the obstacles;

- *transient forces*, i.e., forces given by the temporary presence of other agents. These forces are transient since they are not known offline from the map and the goal. Given a generic point in the map, the transient force is generally not constant in time since the other agents are moving. The transient forces are then the repulsive force F_h generated by the other agents and the cohesion force F_c. The forces introduced in Section 9.7 for dynamic obstacle avoidance will be also classified as transient forces.

This classification of forces is used in Section 9.6 to define the control actions.

9.6 Authority-sharing vehicle control

The social force model is used to define the desired attitude of each assistive vehicle h_U in (9.5). The total force applied on each agent is given by

$$F_{\text{tot}} = F_a + F_c + \sum_{i=1}^{n_{\text{obs}}} (F_{\text{obs},i} + F_{\text{vortex},i}) + \sum_{j=1}^{n_{\text{h}}} F_{h,j},$$

where F_a is the attractive force of the waypoint, F_c is the cohesion force, $F_{\text{obs},i}$ and $F_{\text{vortex},i}$ are the repulsive and vortex forces generated by the closest point of the i–th obstacle, respectively, $F_{h,j}$ is the repulsive force generated by the j–th human of the group, n_{obs} is the number of obstacles and n_{h} is the number of humans in the group. It is remarked that the permanent force acting on each agent is

$$F_{\text{per}} = F_a + \sum_{i=1}^{n_{\text{obs}}} \left(F_{\text{obs},i} + F_{\text{vortex},i}\right).$$

We denote by h_{tot} and h_{per} the headings of the total force F_{tot} and the permanent force F_{per}, respectively, i.e.,

$$h_{\text{tot}} = \frac{F_{\text{tot}}}{|F_{\text{tot}}|},$$
$$h_{\text{per}} = \frac{F_{\text{per}}}{|F_{\text{per}}|}.$$

Notice that a proper selection of the waypoints and the vortex fields ensures $|F_{\text{per}}| \neq 0$, hence the heading of the permanent force h_{per} is always well-defined.

9.6.1 Choice of the braking action

The robot brakes via (9.12) in undesired situations, i.e., when the vehicle is either wrongly oriented or close to a local minimum. Otherwise, the user is left in control of the vehicle. We have

$$\begin{aligned} \dot{v} = a_{\text{robot}} = -\kappa_b\, v, \quad & \text{if } h \cdot h_U < \cos\Theta, \text{ or } h_{\text{per}} \cdot h_{\text{tot}} < 0, \\ & \text{or } |F_{\text{tot}}| < \bar{F}, \\ \dot{v} = a_{\text{user}}, \quad & \text{otherwise,} \end{aligned}$$

where $\Theta \in (0, 90°)$ and $\bar{F} > 0$ are two thresholds. Notice that vehicle brakes in three cases:

1. $h \cdot h_U < \cos\Theta$, i.e., the vehicle heading h has an angular distance larger than Θ from the reference heading h_U. This situation is undesired since, in accordance with Theorem 10, the angular distance between the vehicle heading h and reference heading h_U has to be small for proper navigation;

2. $h_{\text{per}} \cdot h_{\text{tot}} < 0$, i.e., the permanent force F_{per} and F_{tot} form an angle larger than 90°. In this case the robot brakes since the transient forces (i.e., the cohesion force F_c and the repulsive force generated by the others $\sum_{j=1}^{n_{\text{h}}} F_{h,j}$) are pushing the vehicle in a direction opposite with respect to the permanent force F_{per} (i.e., far from the waypoint or close to the obstacles);

3. $|F_{\text{tot}}| < \bar{F}$, i.e., the total force is small, hence the vehicle is close to a local minimum.

9.6.2 Choice of the attitude reference h_U

The reference heading h_U for the vehicle is selected between the heading of the total force h_{tot} and the heading of the permanent force h_{per}. In particular, in absence of local minima issues, we consider the total force h_{tot} as reference heading in order to steer the vehicle in the direction of motion established by the social force model. Conversely, if there are local minima issues, we consider the permanent force h_{per} as reference heading in order to steer the vehicle (which is braking) towards the waypoint and away from the obstacles. We have

$$\begin{aligned} h_U &= h_{\text{per}}, \quad \text{if } h_{\text{per}} \cdot h_{\text{tot}} < 0, \text{ or } |F_{\text{tot}}| < \bar{F}, \\ h_U &= h_{\text{tot}}, \quad \text{otherwise}, \end{aligned}$$

Notice that the local minima conditions are the same used for the choice of the braking action in Section 9.6.1. Notice also that, even in the presence of local minima issues (for instance $|F_{\text{tot}}| \approx 0$), the reference of the vehicle heading is well-defined since we orient the vehicle towards the permanent force h_{per} instead of the total force h_{tot}.

9.6.3 Vehicle reorientation

Theorem 10 states that, if the vehicle heading h is close to the field heading h_U, the robot navigates towards the goal. Hence, if the vehicle is not properly oriented, i.e., $h \cdot h_U \leq \cos\Theta$, where Θ is an angular threshold, the control action $\omega = \omega_{\text{robot}}$ in (9.10) is applied to reorient the heading h. Conversely, if the vehicle heading h is close to the desired attitude h_U, the control authority is given to the user since he/she is pushing the vehicle in the correct direction. We have

$$\begin{aligned} \omega &= \omega_{\text{robot}}, \quad \text{if } h \cdot h_U < \cos\Theta, \\ \omega &= \omega_{\text{user}}, \quad \text{otherwise}, \end{aligned}$$

where $\omega_{\text{robot}} = -\kappa\, V^{\alpha}\, \text{sign}(h_U^x h^y - h_U^y h^x)$ as in (9.10).

9.6.4 Overall authority-sharing strategy

The overall hybrid automaton defining the authority-sharing control is obtained by combining sections 9.6.1, 9.6.2, and 9.6.3. To facilitate the implementation and avoid chattering phenomena, we adopted an hysteresis mechanism using two angular thresholds $\Theta_2 > \Theta_1$ instead of a single threshold Θ. The automaton is reported in Figure 9.8.

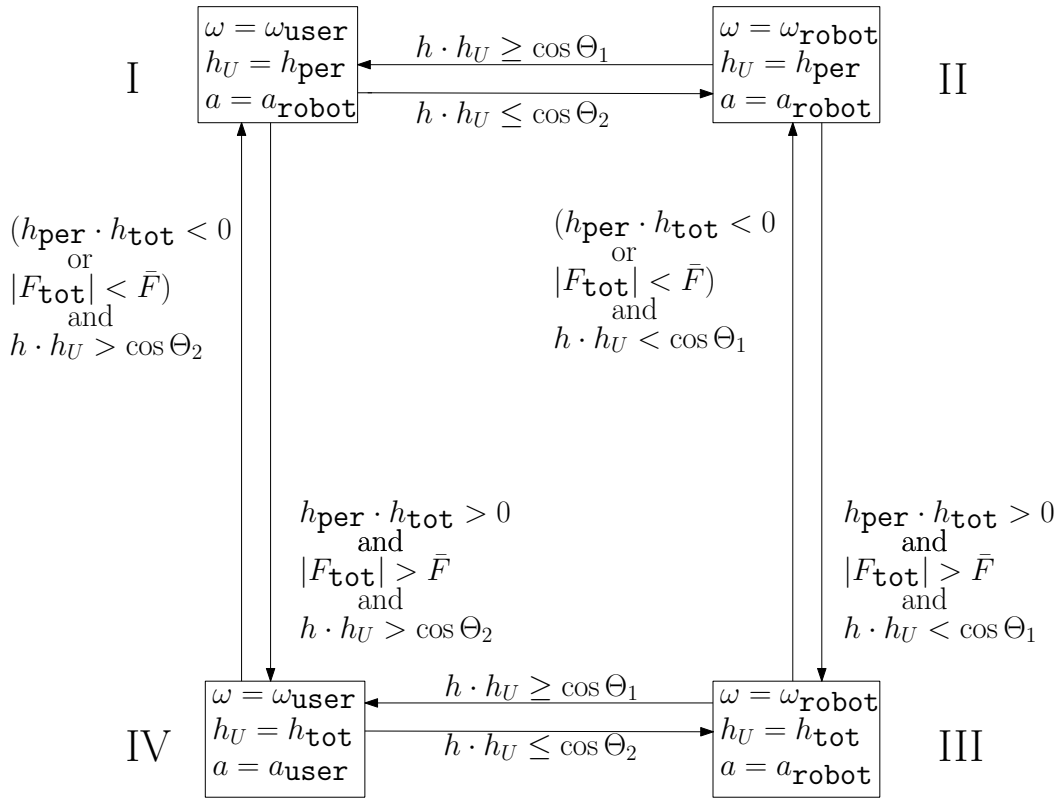

Figure 9.8: Overall hybrid automaton defining the authority-sharing control [7].

9.7 Dynamic obstacle avoidance using limit cycles

The social force model includes an intrinsic dynamic obstacle avoidance behavior because of the repulsive force between two agents. However, several studies on potential field multiagent navigation (see e.g., [58]) show that smoother obstacle avoidance maneuvers are obtained using limit cycles. This approach is widely used in multiagent framework since it is easily combined with a force field control and its computational cost is very limited (no path planning is required). The strategy to avoid obstacles using limit cycles in a 2D space is to define a closed trajectory encircling the obstacle, and then a nonlinear differential equation whose solutions converge and spiral into the closed trajectory.

In this chapter, we implement obstacle avoidance by combining the force field generated by the SFM, with the limit cycle approach. Our specific contribution is also to use limit cycles of a peculiar shape: ovals. A shown in Section 9.7.2 the asymmetric structure of ovals offers several advantages over the circular shapes usually adopted in the literature and presented in Section 9.7.1.

9.7.1 Circular limit cycles

Consider, for instance, the differential equation

$$\begin{aligned}\dot{x}_1 &= -\sigma x_2 + \alpha x_1 \left(1 - \left(\frac{x_1}{R}\right)^2 - \left(\frac{x_2}{R}\right)^2\right),\\ \dot{x}_2 &= \sigma x_1 + \alpha x_2 \left(1 - \left(\frac{x_1}{R}\right)^2 - \left(\frac{x_2}{R}\right)^2\right),\end{aligned} \tag{9.13}$$

where x_1 and x_2 are the states, $\sigma \in \{-1, 1\}$ is the direction of rotation, R and $\alpha > 0$ are two constants. Notice from Figure 9.9 that the limit cycle is a circle, hence parameter R is the corresponding radius. Notice also that the direction of rotation satisfies

$$\sigma = \begin{cases} +1 & \text{clockwise rotation,}\\ -1 & \text{counterclockwise rotation.}\end{cases}$$

By a Lyapunov based approach the almost global asymptotic stability of the limit cycle is proved [58] (i.e., almost all trajectories converge to the circle as the time goes to infinity). To understand the limit cycle behavior, Equation (9.13) is decomposed in two contributes

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \sigma\, x_{ff} + \alpha\, x_{fb},$$

where

$$\begin{aligned}x_{ff} &= \begin{bmatrix}-x_2\\ x_1\end{bmatrix},\\ x_{fb} &= \begin{bmatrix}x_1\\ x_2\end{bmatrix}\left(1 - \left(\frac{x_1}{R}\right)^2 - \left(\frac{x_2}{R}\right)^2\right).\end{aligned}$$

The term x_{ff} acts as feedforward action ensuring that the solutions spiral into circular orbits (left picture in Figure 9.10). In fact, if a solution spins counterclockwise around a circle of any radius $\tilde{R} > 0$, its velocity field satisfies

$$\begin{aligned}x_1 = \tilde{R}\cos t \quad &\Longrightarrow \quad \dot{x}_1 = -\tilde{R}\sin t = -x_2,\\ x_2 = \tilde{R}\sin t \quad &\Longrightarrow \quad \dot{x}_2 = \tilde{R}\cos t = x_1.\end{aligned}$$

The term x_{fb} acts as feedback action ensuring that the solutions are attracted towards the circle of radius R. In fact notice that, if $1 - \left(\frac{x_1}{R}\right)^2 - \left(\frac{x_2}{R}\right)^2 > 0$ (i.e., inside the circle) x_{fb} acts as repulsive action from the center of the circle, while if $1 - \left(\frac{x_1}{R}\right)^2 - \left(\frac{x_2}{R}\right)^2 < 0$ (i.e., outside the circle), x_{fb} acts as attractive action to the center of the circle. In both cases, the feedback action "pushes" the solution towards the circle (right picture in Figure 9.10). Notice that the feedback component is defined by measuring a normalized distance ρ from the circle via its implicit equation, i.e., $\rho = 1 - \left(\frac{x_1}{R}\right)^2 - \left(\frac{x_2}{R}\right)^2$.

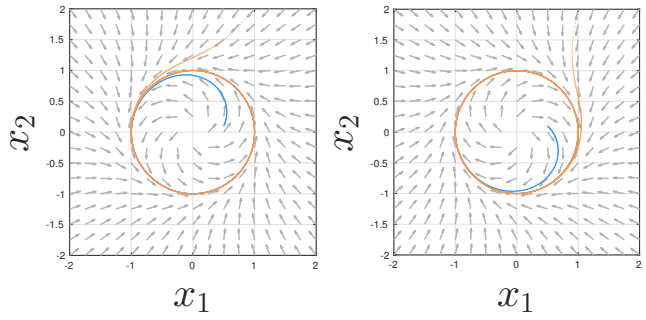

Figure 9.9: Phase portraits of Equation (9.13). The evolution of two solutions starting inside and outside the circle are reported. The direction of rotation is $\sigma = 1$ on the left and $\sigma = -1$ on the right [7].

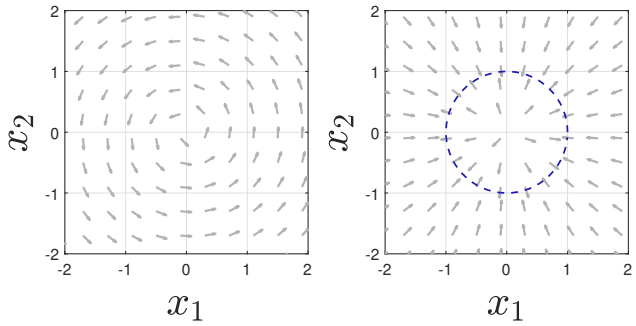

Figure 9.10: Phase portraits of the feedforward (on the left) and feedback (on the right) components of Equation (9.13) [7].

The force F_{lc} generated by the dynamic obstacle to the vehicle is added to the transient forces, defined in Section 9.5 and it is oriented as the vector field in (9.13). The heading of the force F_{lc} generated by the dynamic obstacle is

$$h_{\text{lc}} = \frac{[\dot{x}_1, \dot{x}_2]^\top}{|[\dot{x}_1, \dot{x}_2]^\top|},$$

while its modulus is

$$|F_{\text{lc}}| = \begin{cases} 0, & \text{if } \rho < 0, \\ F_{\max} - (F_{\max} - F_{\min})\rho, & \text{otherwise,} \end{cases} \tag{9.14}$$

where $F_{\max} \geq F_{\min} \geq 0$ are the maximum and the minimum values for the norm of the limit cycle force $|F_{\text{lc}}|$. Notice that, outside the limit cycle trajectory ($\rho < 0$) the force F_{lc} is set to zero. Conversely, inside the circle ($\rho > 0$), the force F_{lc} grows linearly with respect to the distance ρ.

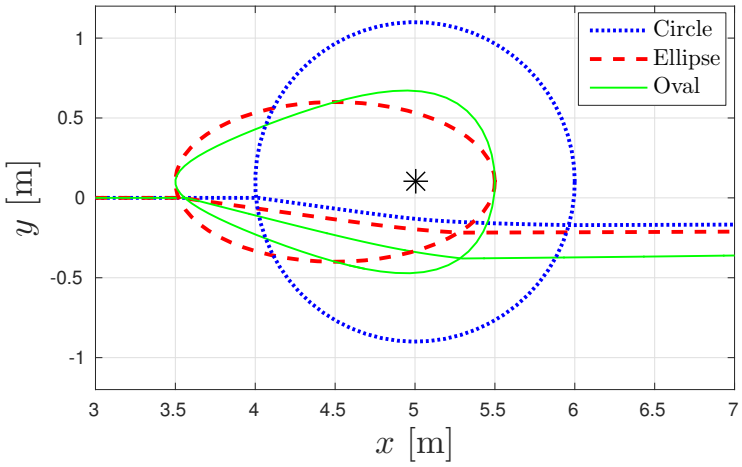

Figure 9.11: Trajectories (solid lines) executed by the robot moving from left to right with different shapes of the limit cycle (dashed lines) [7].

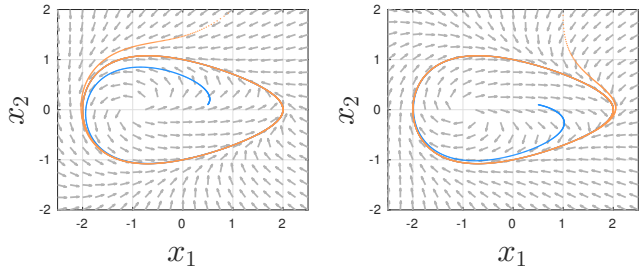

Figure 9.12: Desired phase portrait for the oval limit cycle. The evolution of two solutions starting inside and outside the oval are reported. The direction of rotation is $\sigma = 1$ on the left and $\sigma = -1$ on the right [7].

9.7.2 Oval limit cycles

Since the robot is moving with respect to the obstacle, a limit cycle not having circular symmetry is indeed a more reasonable choice to avoid the collision. Figure 9.11 shows the trajectory of the vehicle to avoid a point-shaped obstacle with a limit cycle of three shapes (circle, ellipse, and oval). Notice that using the ellipse or the oval, the avoidance maneuver starts at larger distance from the obstacle. Moreover the oval limit cycle maintains the robot further from the obstacle with respect to the ellipse in the last part of the maneuver.

We aim to generalize the strategy of Section 9.7.1 to generic elliptical and oval limit cycle trajectories to improve the obstacle avoidance maneuver.

The oval implicit equation is

$$\left(\frac{x_1}{b_1}\right)^2 + \left(\frac{x_2}{b_2}\right)^2 \mathrm{e}^{\nu\, x_1} = 1,$$

where b_1 and b_2 are the two axis lengths and ν is a deformation parameter (i.e., if $\nu = 0$ the ellipse case is obtained). To obtain the equations of the limit cycle as in (9.13), we compute separately the feedforward and the feedback terms. For the feedforward term, we consider a solution following an oval trajectory

$$\begin{aligned} x_1 &= b_1 \cos t, \\ x_2 &= b_2 \mathrm{e}^{-\frac{\nu}{2} x_1} \sin t, \end{aligned}$$

whose velocity is

$$\begin{aligned} \dot{x}_1 &= -b_1 \sin t, \\ \dot{x}_2 &= b_2 \mathrm{e}^{-\frac{\nu}{2} x_1} \cos t + b_2 \mathrm{e}^{-\frac{\nu}{2} x_1} \dot{x}_1 \frac{-\nu}{2} \sin t, \end{aligned}$$

then, by substituting $\cos t = x_1/b_1$ and $\sin t = \mathrm{e}^{\frac{\nu}{2} x_1} x_2/b_2$, the feedforward component is

$$x_{ff} = \begin{bmatrix} -\frac{b_1}{b_2} x_2 \mathrm{e}^{\frac{\nu}{2} x_1} \\ \frac{b_2}{b_1} x_1 \mathrm{e}^{-\frac{\nu}{2} x_1} + x_2^2 \mathrm{e}^{\frac{\nu}{2} x_1} \nu \frac{b_1}{2 b_2} \end{bmatrix}. \tag{9.15}$$

For the feedback term, we have

$$x_{fb} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \left(1 - \left(\frac{x_1}{b_1}\right)^2 - \left(\frac{x_2}{b_2}\right)^2 \mathrm{e}^{\nu\, x_1} \right) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \rho.$$

Notice that the normalized distance from the oval to be used in (9.14) is $\rho = 1 - (x_1/b_1)^2 - (x_2/b_2)^2 \mathrm{e}^{\nu\, x_1}$.

The overall nonlinear differential equations is then

$$\begin{aligned} \dot{x}_1 &= -\sigma \frac{b_1}{b_2} x_2 \mathrm{e}^{\frac{\nu}{2} x_1} + x_1 \rho, \\ \dot{x}_2 &= \sigma \left(\frac{b_2}{b_1} x_1 \mathrm{e}^{-\frac{\nu}{2} x_1} + x_2^2 \mathrm{e}^{\frac{\nu}{2} x_1} \nu \frac{b_1}{2 b_2} \right) + x_2 \rho. \end{aligned} \tag{9.16}$$

In order to anticipate the avoidance maneuver as in Figure 9.11, the obstacle should not be located in the center of the closed trajectory defining the limit cycle (but for instance in a focus). It is remarked that it is not sufficient to translate the obstacle position in the focus to obtain the desired behavior. Consider the left picture in Figure 9.13 and assume that the obstacle is located in $[-1, 0]^\top$. Notice that, if the initial condition is chosen between center and obstacle, the solution gets closer to

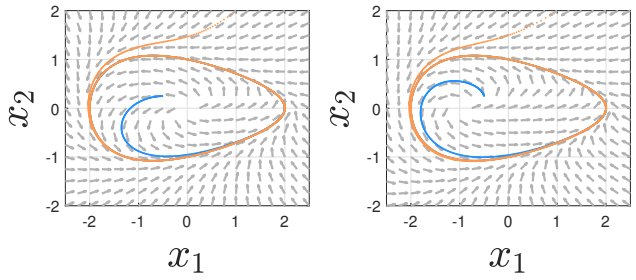

Figure 9.13: Phase portrait for Equation (9.16), on the left, and desired phase portrait for the obstacle avoidance, on the right [7].

the obstacle, which is clearly undesired to avoid the collision. The desired behavior is reported in the right picture in Figure 9.13, where, even if the solution starts between center and obstacle, it moves away from the point $[-1, 0]^\top$. Notice that in the left picture of Figure 9.13, the center of the oval $[0, 0]^\top$ is an unstable equilibrium, while in the right picture of Figure 9.13, the unstable equilibrium point is on the obstacle, i.e., $[-1, 0]^\top$. To translate the unstable equilibrium from the point $[0, 0]^\top$ to the point $[x_c, 0]^\top$, where x_c is x_1 coordinate of the new unstable equilibrium, we first modify the feedback term as

$$x_{fb} = \begin{bmatrix} x_1 - x_c \\ x_2 \end{bmatrix} \rho.$$

To modify the feedforward term, notice from Figure 9.14 that the spiraling oval trajectories defined by the desired feedforward component x_{ff} have different centers. The translation of the center position depends on the oval dimension. In particular we impose a linear relation between the translation x_{trasl} of the center and the semi-axle of the oval, i.e.,

$$x_{\text{trasl}}(r) = x_c \left(1 - \frac{r}{b_1}\right),$$

where r is the semi-axle of the oval centered in $[x_{\text{trasl}}, 0]^\top$. Notice that if $r = b_1$ (i.e., the semi-axle is equal to the semi-axle of the main oval), we get $x_{\text{trasl}}(b_1) = 0$, which is the center of the main oval. Similarly, if $r = 0$, we have $x_{\text{trasl}}(0) = x_c$, which coincides with the desired unstable equilibrium. In the computation of the feedforward term we also require that each oval in Figure 9.14 has the same ratio $\epsilon = \frac{b_1}{b_2}$. Each oval in Figure 9.14 has center $[x_{\text{trasl}}(r), 0]^\top$ and semi-axle r, then it satisfies the implicit oval equation

$$\left(\frac{x_1 - x_{\text{trasl}}(r)}{r}\right)^2 + \left(\frac{x_2\,\epsilon}{r}\right)^2 \mathrm{e}^{\nu\, x_1} = 1. \tag{9.17}$$

The analytic expression of $r^\star(x_1, x_2)$, i.e., the solution of (9.17) with respect to r can be obtained symbolically but it is here omitted for brevity's sake. Therefore, the

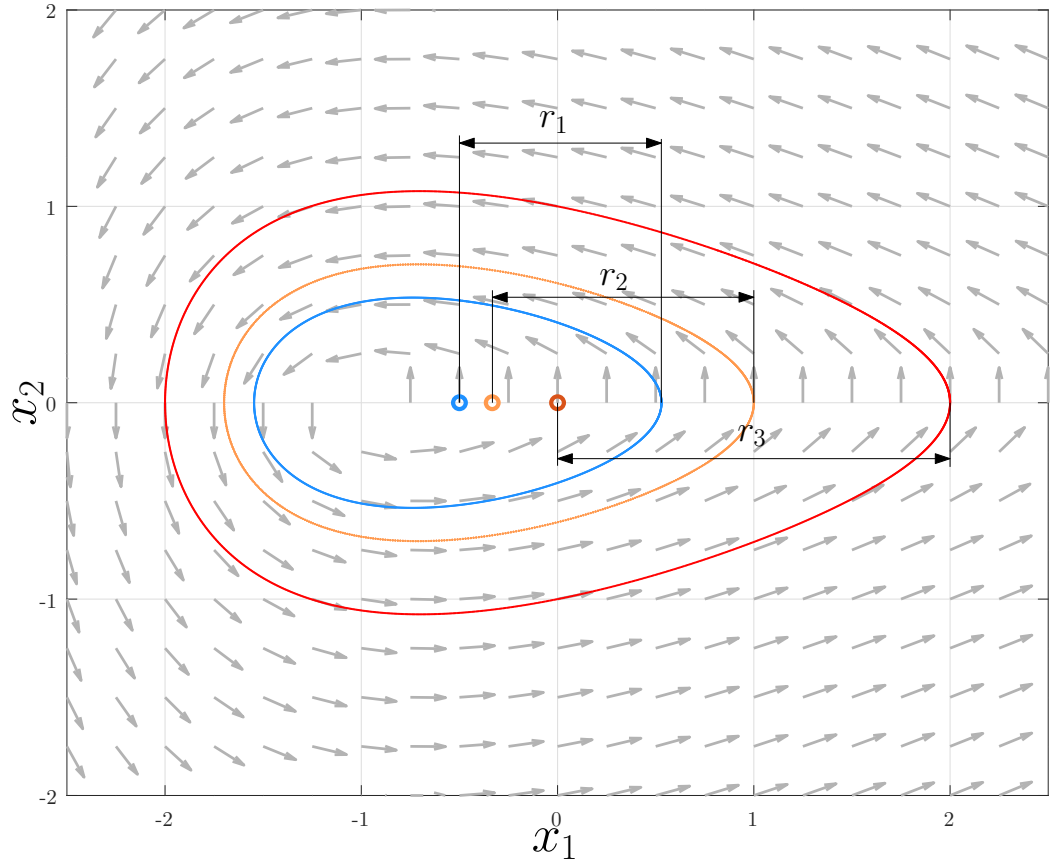

Figure 9.14: Pencil of coaxial ovals. r_1, r_2, and r_3 are the horizontal semi-axes of the plotted ovals. For each oval, the position of the center is shown. The arrows represent the feedforward component of the differential equation generating the desired oval limit cycle [7].

feedforward component is obtained by translating (9.15) in $[x_{\text{trasl}}(r^\star(x_1, x_2)), 0]^\top$, i.e.,

$$x_{ff} = \begin{bmatrix} -\frac{b_1}{b_2} x_2 \mathrm{e}^{\frac{\nu}{2}\bar{x}_1} \\ \frac{b_2}{b_1}\bar{x}_1 \mathrm{e}^{-\frac{\nu}{2}\bar{x}_1} + x_2^2 \mathrm{e}^{\frac{\nu}{2}\bar{x}_1} \nu \frac{b_1}{2b_2} \end{bmatrix},$$

where $\bar{x}_1 = x_1 - x_{\text{trasl}}(r^\star(x_1, x_2))$.

Overall, the differential equation defining the desired shape of the oval limit cycle (Figure 9.12) is

$$\begin{aligned} \dot{x}_1 &= -\sigma\epsilon x_2 \mathrm{e}^{\frac{\nu}{2}\bar{x}_1} + (x_1 - x_c)\rho, \\ \dot{x}_2 &= \sigma\left(\frac{\bar{x}_1}{\epsilon}\mathrm{e}^{-\frac{\nu}{2}\bar{x}_1} + x_2^2 \mathrm{e}^{\frac{\nu}{2}\bar{x}_1}\nu\frac{\epsilon}{2}\right) + x_2\rho. \end{aligned} \tag{9.18}$$

9.7.3 Definition of the direction of rotation

In this section we propose a method to chose the direction of rotation $\sigma \in \{-1, 1\}$ of the limit cycle in (9.18). We denote by v_{obs} the velocity of the obstacle, by $[x_{\text{obs}}, y_{\text{obs}}]^\top$ its coordinates, and by h_{obs} its heading, i.e., $v_{\text{obs}} = h_{\text{obs}}|v_{\text{obs}}|$. Moreover we denote by h_+ the unit vector defining the direction of motion counterclockwise with respect to the obstacle (see Figure 9.15), i.e.,

$$h_+ = \frac{[-(y - y_{\text{obs}}), x - x_{\text{obs}}]^\top}{|[-(y - y_{\text{obs}}), x - x_{\text{obs}}]^\top|}.$$

We propose to choose the direction of rotation σ in (9.18) as

$$\sigma = \text{sign}\left(w_1 f_1 + w_2 f_2 + w_3 f_3\right), \tag{9.19}$$

where w_1, w_2, and $w_3 > 0$ are weights, and f_1, f_2, and f_3 are functions of the relative motion between obstacle and robot defined as

$$\begin{aligned} f_1 &= h_+ \cdot h_{\text{per}}, \\ f_2 &= h_+ \cdot h, \\ f_3 &= -\text{sign}(h_+ \cdot h_{\text{obs}}) f_4(v_{\text{obs}}, \bar{v}_{\text{obs}}) f_5(h, h_{\text{obs}}, \Theta), \end{aligned}$$

where $\bar{v}_{\text{obs}} > 0$ is a velocity threshold, $\Theta \in (0°, 90°)$ is an angular threshold, and functions $f_4(v_{\text{obs}}, \bar{v}_{\text{obs}})$ and $f_5(h, h_{\text{obs}}, \Theta)$ are reported in Figure 9.17. The functions f_1, f_2, and f_3 suggest to the robot to pass around the obstacle clockwise or counterclockwise. In particular:

- f_1 is positive if the robot follows the permanent field by passing around the obstacle in the counterclockwise direction. Then, since the counterclockwise direction is defined by $\sigma = +1$, if the weights in (9.19) are chosen as $w_1 > 0$ and $w_2 = w_3 = 0$, the robot passes around the obstacle by always moving towards the goal. Then, w_1 weights the importance of moving towards the goal;

- f_2 is positive if the counterclockwise direction is oriented as the robot heading h. Then, if the weights in (9.19) are chosen as $w_2 > 0$ and $w_1 = w_3 = 0$, the robot passes around the obstacle by moving in approximately the same direction of its heading h. Then, w_2 penalizes obstacle avoidance maneuvers that require the robot to significantly change its heading h;

- f_3 is positive (depending on $-\text{sign}(h_+ \cdot h_{\text{obs}})$) if the counterclockwise direction is such that the robot safely passes behind the moving obstacle. Notice that, if the weights in (9.19) are chosen as $w_3 > 0$ and $w_1 = w_2 = 0$, the robot always passes behind the obstacle: this choice is safer that passing in front of the obstacle since, if the obstacle does not invert its velocity, the collision

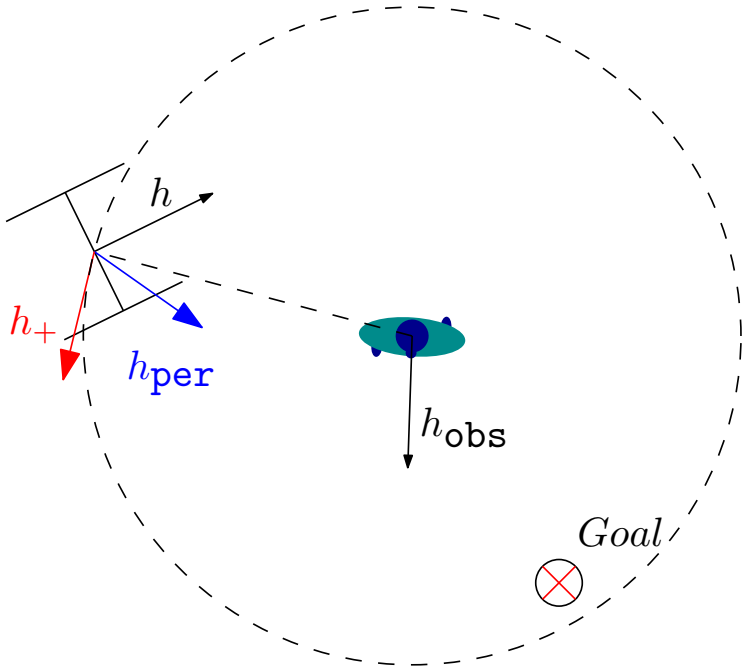

Figure 9.15: Notation in the collision avoidance of dynamic obstacles [7].

is surely avoided. f_3 is obtained as product of the two penalty functions f_4 and f_5 reported in Figure 9.17. $f_4(v_{\text{obs}})$ assumes values smaller than 1 if the obstacle is moving at speed smaller than the threshold $\bar{v}_{\text{obs}}$ since slowly moving obstacles are less dangerous. f_5 assumes values smaller than 1 if the obstacle and the robot are moving in the same direction to reduce the importance of the passing behind behavior in case of overtaking. Overall w_3 weights the safety of the obstacle avoidance maneuver.

The effect of the weights in (9.19) on the direction of rotation are depicted in Figure 9.16.

9.7.4 Rototranslation of the oval

Since the oval does not have circular symmetry, it has to be oriented in the plane to facilitate the obstacle avoidance maneuver (see Figure 9.18). Let h_{oval} the unit vector defining the x-axis of the oval in Figure 9.12, i.e., the oval heading. To proper place the oval limit cycle in the map, we need to define the orientation of the oval heading h_{oval}.

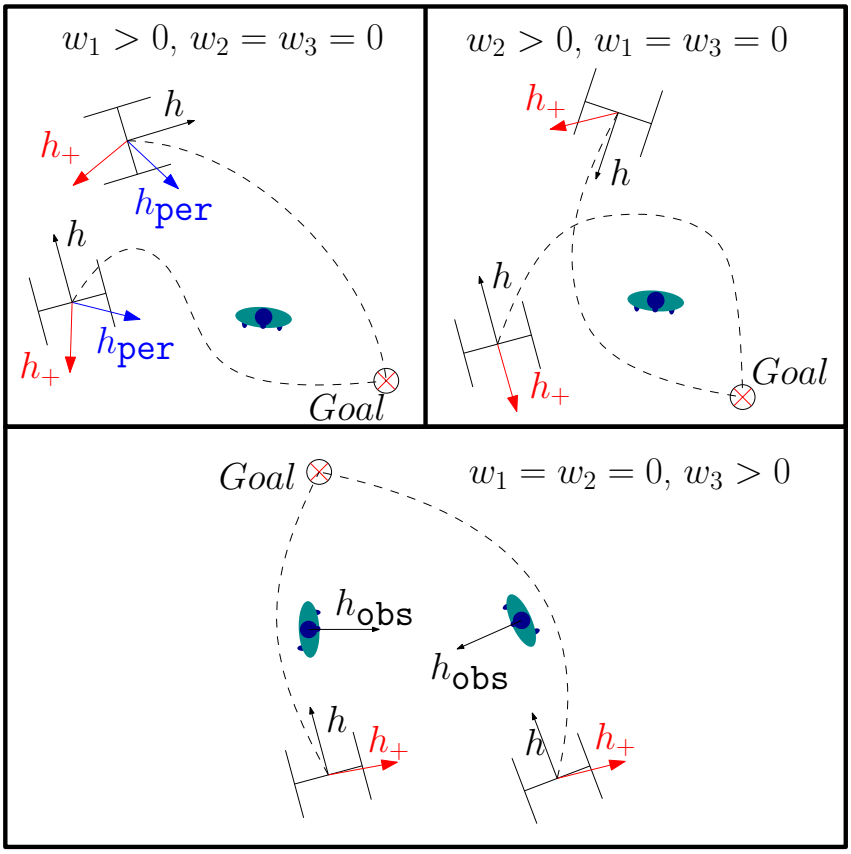

Figure 9.16: Effect of the weights in (9.19) on the direction of rotation [7].

Initialization of the oval heading

Notice from Figure 9.18 that a good thumb rule to orient the oval, is to ensure that the oval heading h_{oval} points towards the robot. The oval heading h_{oval} is then initialized on the basis of the expected robot motion (Figure 9.19). Let wp_k be the waypoint of the sequence of waypoints defined in Section 9.5.1 closest to the obstacle and let wp_{k+1} be the waypoint subsequent to wp_k. We initialize the oval heading h_{oval} as

$$h_{\text{oval}} = \frac{wp_k - wp_{k+1}}{|wp_k - wp_{k+1}|}. \tag{9.20}$$

Rotation of the oval heading

Even if the oval heading h_{oval} points towards the robot, the trajectory may not be the desired one. Consider for instance the examples reported in Figure 9.21. In the examples of Figure 9.21, it is established via relation (9.19) to pass around the

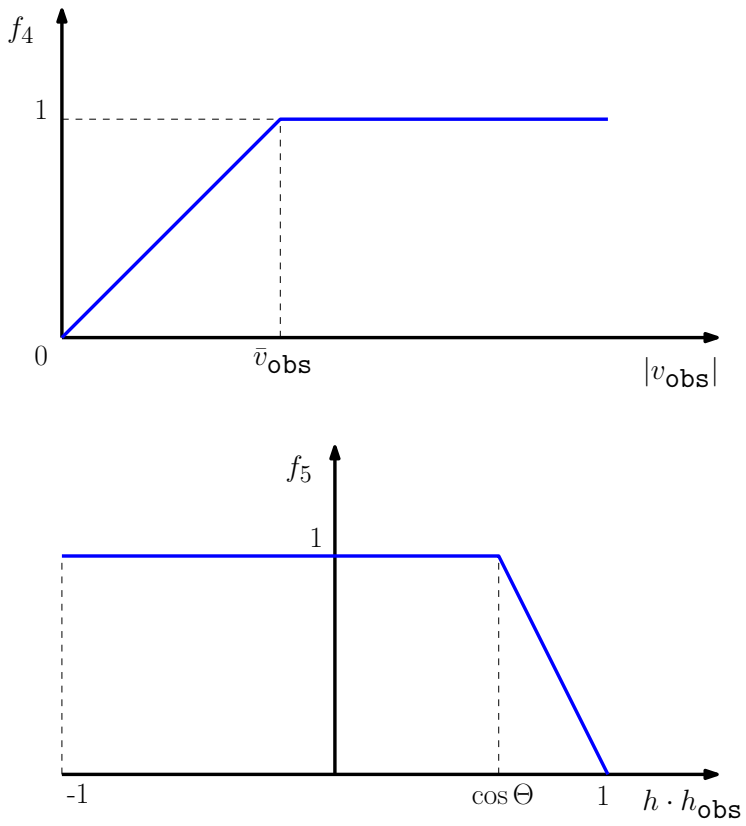

Figure 9.17: Penalty functions in the definition of f_3 in (9.19) [7].

obstacle counterclockwise. Notice that in the top picture the trajectory commanded by the limit cycle pushes undesirably far from the goal. Conversely, on the bottom picture, the oval heading h_{oval} is slightly rotated and therefore the counterclockwise rotation is executed with a commanded trajectory that does not push the vehicle backward. To ensure that the commanded trajectory is always desirable, the oval heading h_{oval} is computed as

$$\begin{aligned} h_{\text{oval}} &= \mathcal{R}(-\sigma\alpha_0)h_{o\to r}, \quad && \text{if } \sigma h_{\text{per}} \cdot h_+ < 0, \text{ and} \\ & && \quad h \cdot h_{o\to r} < 0, \\ h_{\text{oval}} &= \frac{wp_k - wp_{k+1}}{|wp_k - wp_{k+1}|}, && \text{otherwise,} \end{aligned} \tag{9.21}$$

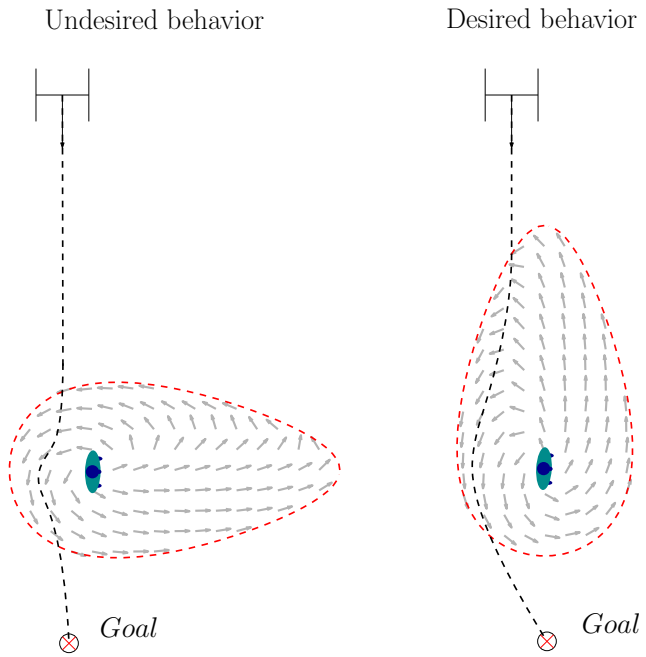

Figure 9.18: Positive effect of the orientation of the oval towards the robot [7].

where $\alpha_0 > 0$ is a positive small angle, $\mathcal{R}(-\sigma\alpha_0)$ is the rotation matrix of angle $(-\sigma\alpha_0)$, and $h_{o\to r}$ is the unit vector pointing from the obstacle to the robot, i.e.,

$$h_{o\to r} = \frac{[x - x_{\text{obs}}, y - y_{\text{obs}}]^\top}{|[x - x_{\text{obs}}, y - y_{\text{obs}}]^\top|},$$

$$\mathcal{R}(-\sigma\alpha_0) = \begin{bmatrix} \cos(-\sigma\alpha_0) & -\sin(-\sigma\alpha_0) \\ \sin(-\sigma\alpha_0) & \cos(-\sigma\alpha_0) \end{bmatrix}.$$

The choice $h_{\text{oval}} = \mathcal{R}(-\sigma\alpha_0)h_{o\to r}$ in (9.21) ensures that the oval heading h_{oval} is such that the commanded trajectory does not push the vehicle backward. In fact, the vehicle is pushed backward if oval heading h_{oval} is chosen as in (9.20) and $\sigma h_{\text{per}} \cdot h_+ < 0$. Notice that we reorient the oval heading in (9.21) only if the robot is getting close to the obstacle, i.e., $h \cdot h_{o\to r} < 0$. The geometrical construction is reported in Figure 9.20.

Rototranslation of the coordinate system

Once the oval heading $h_{\text{oval}} = [h^x_{\text{oval}}, h^y_{\text{oval}}]^\top$ is computed via (9.21). The values of x_1 and x_2 used to compute the direction of the force applied by the limit cycle in (9.18) are defined as

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} h^x_{\text{oval}} & -h^y_{\text{oval}} \\ h^y_{\text{oval}} & h^x_{\text{oval}} \end{bmatrix}^\top \begin{bmatrix} x - x_{\text{obs}} \\ y - y_{\text{obs}} \end{bmatrix} - \begin{bmatrix} x_c \\ 0 \end{bmatrix}.$$

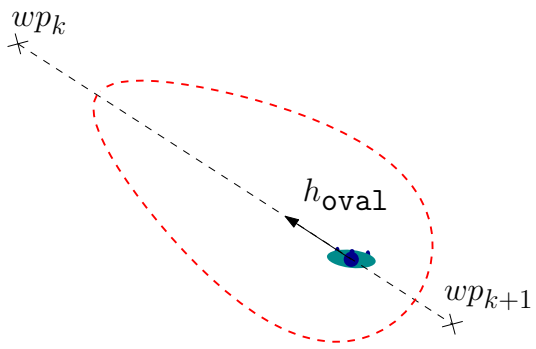

Figure 9.19: Initialization of the oval heading h_{oval} [7].

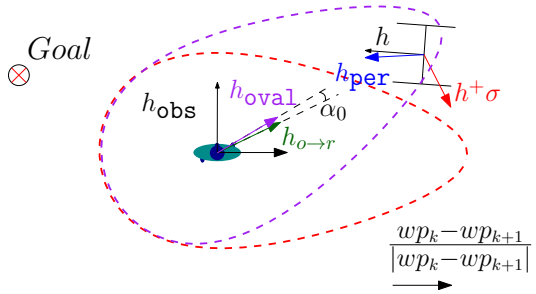

Figure 9.20: Rotation of the oval heading h_{oval} [7].

This way the obstacle is centered in position $[x_c, 0]^\top$ in a reference frame attached to the oval (as in Section 9.7.2) and oriented as the desired oval heading h_{oval}.

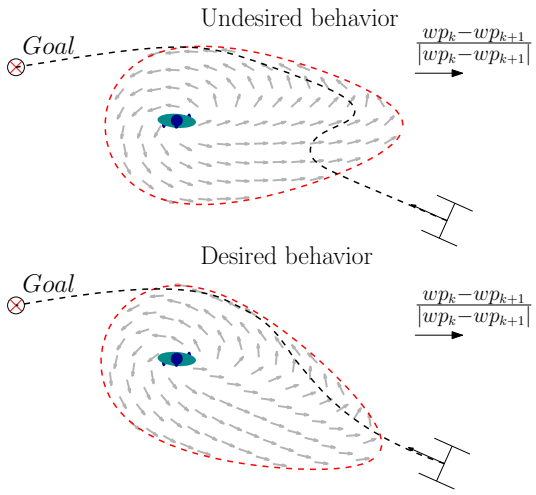

Figure 9.21: Effect of the rotation of the oval heading on the robot trajectory [7].

9.8 Simulation results

The proposed approach has been extensively tested in simulations. The adopted indoor environment is the map of the Department of Information Engineering and Computer Science of the University of Trento. The static obstacles in the map (e.g., walls) are depicted as grey boxes. Since we share the control authority with the assisted person, the robot trajectories depend on the implemented human behavior. When the user is in control of the forward motion, i.e., $\dot{v} = a_{\text{user}}$ in (9.2), we model the user as a proportional controller stabilizing the forward speed of 1 m/s. When the user is in control of the angular motion, i.e., $\omega = \omega_{\text{user}}$ in (9.3), we analyze two opposite user behaviors:

- uncooperative user: the assisted person applies a control action $\omega = \omega_{\text{user}} = 0$, hence he/she pushes the vehicle forward even if a collision with an obstacle is going to occur (and hence we classify the user as *uncooperative*);
- cooperative user: the assisted person applies a control action $\omega = \omega_{\text{user}}$ to reduce the attitude error (9.8). In other words, the user orients the vehicle heading h (i.e., the direction of his/her velocity) towards the field heading h_U (and hence we classify the user as *cooperative*).

It is remarked that the uncooperative user applying $\omega = \omega_{\text{user}} = 0$ is a very precautionary assumption. In fact, in the practice, a senior without cognitive deficits is capable of recognizing if he/she is going to collide with an obstacle and hence he/she helps to modify the vehicle motion to avoid the collision. Moreover, the user is provided with graphical indications on the direction of motion he/she is supposed to follow to reach the desired location using a tablet. Hence, the situation in which

users apply $\omega = \omega_{\text{user}} = 0$ models people with modest cognitive abilities (or serious visual impairments). Conversely, a cooperative user orienting the vehicle heading h towards the field heading h_U, does not have necessarily an ideal behavior. In fact, for instance, a real user may exploit his/her control authority to get closer to another user to establish a relationship which was not predicted by the social force model. In other words, the trajectories executed by real users are expected to be more "natural" than the trajectories executed by the modeled cooperative users (which, for example, tend to maintain single line motion) since the social force model cannot capture all the social dynamics.

9.8.1 Standard navigation

We test the control approach to move a group of three users inside the map. The waypoints (red crosses in the figures) are placed to ensure that:

1. they travel along a corridor;
2. they enter the first room by passing through a door;
3. they enter the second room by passing through a second door;
4. they return in the corridor by passing through a third door.

In figures 9.22 and 9.23 the motion of the robots is executed from right to left.

Figure 9.22 shows the navigation with cooperative users. They travel along the corridor in parallel trajectories up to the first door, where they turn left to enter the first room. Notice that vehicle 2 (which is closer to the door) enters first, while vehicles 1 and 3 wait since the condition $h_{\text{per}} \cdot h_{\text{tot}} < 0$ in Figure 9.8 is triggered since the transient repulsive force generated by vehicle 2 pushes the other vehicles in a direction opposite with respect to the permanent force. In the following, since they enter the first room one by one, they cross the two rooms and the second and third doors in approximately single line. Once in the corridor, the motion continues in single line. It is remarked that the authority-sharing controller in Figure 9.8 does not force the users to move in single line (indeed, in the first part of the simulation they follow parallel trajectories). The single line motion is executed because each user steers the heading of his/her vehicle towards the last waypoint although they are free to assume a different relative position (for instance the "V" shape at the beginning of the simulation).

Figure 9.23 shows the robustness of the authority-sharing controller with uncooperative users. Notice that the users move very close to the walls in the corridors since they push the vehicles towards the wall up to a point where the control authority is taken from the robot to steer the trajectories away. Notice also that the users travel from the first room to the second room by taking a bend corner instead of moving straight to the door. This takes place since the users are endowed with sufficient control authority to perform this unnatural corner. Overall, although the

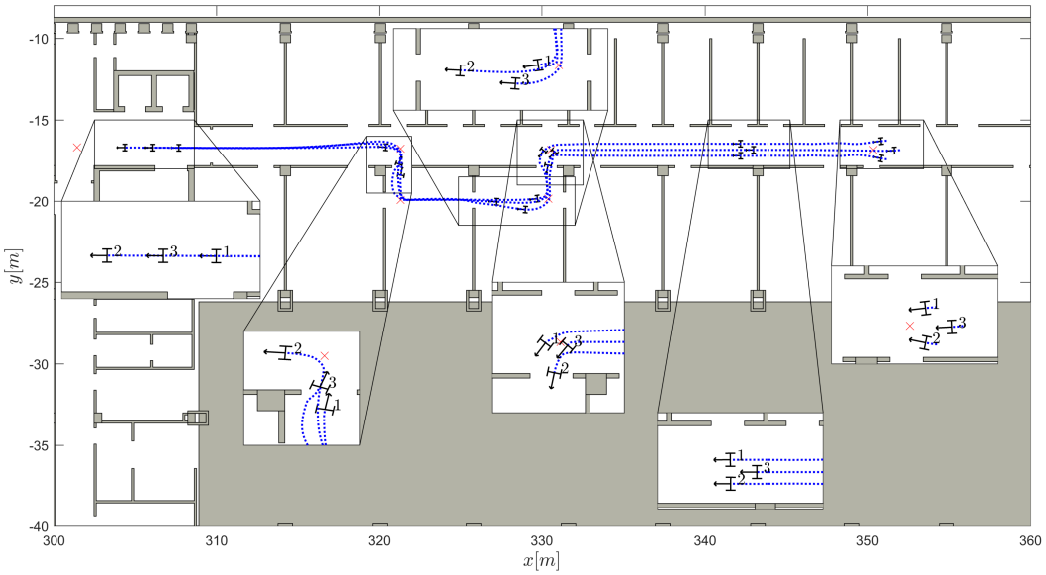

Figure 9.22: Standard navigation with cooperative users [7].

very uncooperative user behavior, the control strategy finally manages to steer the vehicles at the end of the corridor.

9.8.2 Effect of the cohesion force

We show the effect of the cohesion force on the vehicle motion. In the simulation of Figure 9.24, the three vehicles are supposed to follow together the corridor, but the initial condition is such that vehicle 3 is much further than vehicles 1 and 2. In this condition the transient cohesion force F_c applied on vehicle 3 points towards vehicles 1 and 2, i.e., in the opposite direction of the permanent attractive force F_a pointing towards the waypoint. Hence the condition $h_{\text{per}} \cdot h_{\text{tot}} < 0$ is triggered in the automaton of Figure 9.8, then vehicle 3 brakes (i.e., its velocity remains zero) while orienting its heading towards the waypoint. Conversely, the transient cohesion force F_c applied on vehicle 1 and 2 pointing towards vehicle 3, is oriented as the permanent attractive force F_a pointing towards the waypoint, hence $h_{\text{per}} \cdot h_{\text{tot}} > 0$ and then vehicles 1 and 2 are allowed to move forward. The motion of vehicle 3 starts only when vehicle 1 and 2 are sufficiently close, showing a waiting behavior very common in human group walking.

It is remarked that the cohesion force points towards the centroid of the formation (highlighted as a cyan square in Figure 9.24). Therefore the computation of the direction of the cohesion force requires the knowledge of the centroid position. In the proposed simulations, the centroid is computed by the agents in a distributed way. Assuming that each robot knows the position of its neighbors, i.e., the other

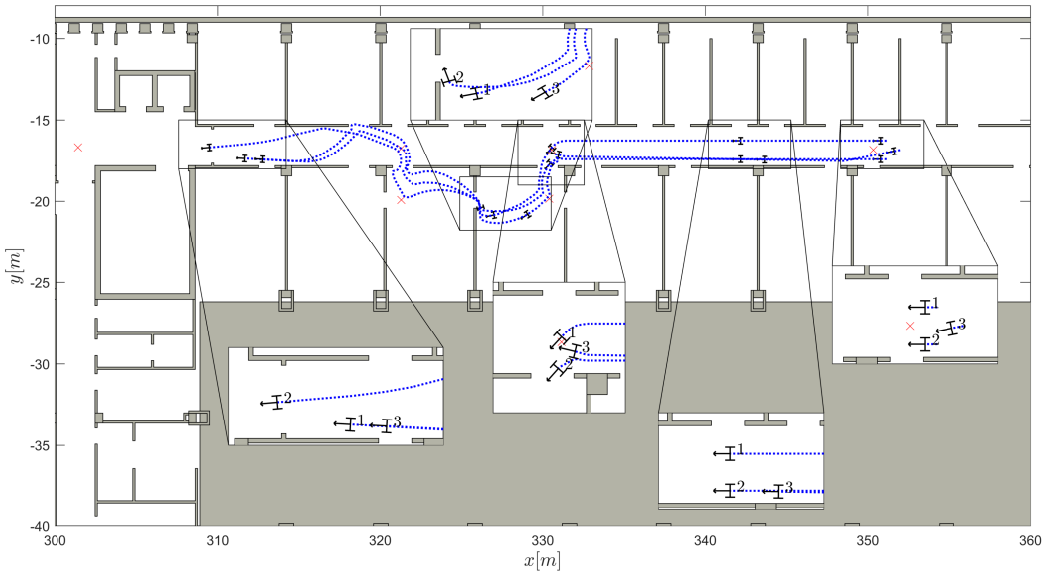

Figure 9.23: Standard navigation with very uncooperative users [7].

robots at a distance smaller than a given communication range, the centroid position may be simply estimated by averaging the robot position $[x, y]^\top$ with the position of its neighbors since in the practice the communication graph is expected to be complete. However, in the rather improbable case in which the communication range is very limited (hence non-complete communication graph), we implemented the the well-known distributed Metropolis algorithm to average the neighbor positions.

9.8.3 Interaction with dynamic obstacles

We tested the dynamic collision avoidance strategies using oval limit cycles and cooperative users. The advantages of oval cycles with respect to circle and ellipse has been already shown in the simulation of Figure 9.11. In the simulations proposed in this section we implement totally uncooperative obstacles, i.e., each obstacle maintains its velocity by ignoring the presence of the assistive walkers. Therefore the collision does not occur only if the vehicles avoid the obstacles. This is a very precautionary assumption since, in practice, the dynamic obstacle encountered in an standard environment are typically humans. It is well-know that the avoidance maneuver involving two humans is cooperative [90, 4], i.e., both agents modify their motion to accommodate the avoidance of the collision. Moreover experimental studies on social dynamics [54] show that, when one of the two humans involved in the collision avoidance maneuver has mobility impairments, the other one (i.e., the able-bodied human) almost totally takes care of the collision avoidance, by significantly modified his/her trajectory with respect to the case in which the other involved

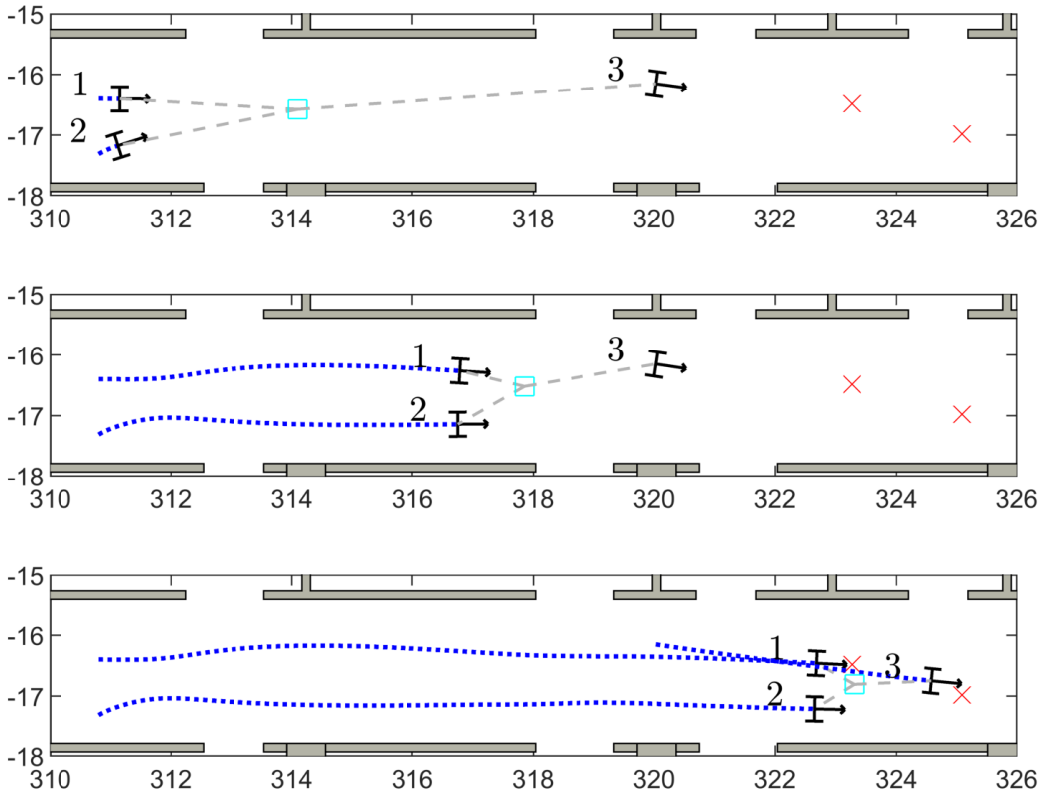

Figure 9.24: Waiting behavior induced by the cohesion force in the cooperative following of a corridor. The three figures represent the vehicle positions in three subsequent time instants [7].

human is able-bodied. This way, in practice, the senior with mobility impairments can maintain his/her motion most of the time. In the proposed simulation, this cooperative behavior of the obstacles is precautionary not implemented.

First, we analyze how the hybrid automaton in Figure 9.8 handles the control authority between user and robot during an obstacle avoidance maneuver. Then we present the collision avoidance approach in the previous indoor environment (University of Trento), where the obstacle trajectories are predetermined functions of time. Finally, the strategy is analyzed by simulating the interaction with humans moving in an outdoor real scenario recorded at ETH [93].

Analysis of the authority-sharing automaton

Figure 9.26 shows how the hybrid automaton in Figure 9.8 handles the control authority during an obstacle avoidance maneuver. The bottom part of the figure reports the difference $\theta_{\text{per}} - \theta_{\text{tot}}$ between the angle of the permanent force θ_{per}

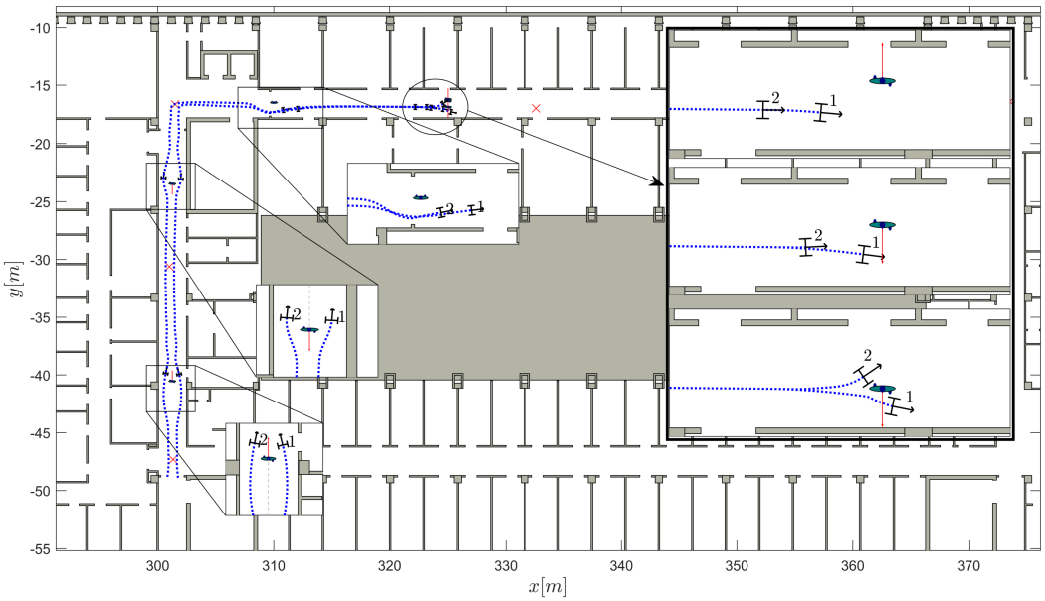

Figure 9.25: Navigation in the presence of uncooperative dynamic obstacles [7].

(i.e., $h_{\text{per}} = [\cos\theta_{\text{per}}, \sin\theta_{\text{per}}]^\top$) and the angle of the total force θ_{tot} (i.e., $h_{\text{tot}} = [\cos\theta_{\text{tot}}, \sin\theta_{\text{tot}}]^\top$), and the difference $\theta - \theta_{\text{tot}}$ between the vehicle yaw θ and the angle of the total force θ_{tot}. In the first part of the simulation, the user is in control of the motion (then $\omega = \omega_{\text{user}}$ and $a = a_{\text{user}}$, i.e., mode IV in Figure 9.8) and sufficiently far from the obstacle, then total force coincides with the permanent force since it is not influenced by the limit cycle (i.e., $\theta_{\text{per}} - \theta_{\text{tot}} \approx 0$). As the user approaches the obstacle, the effect of the limit cycle increases the difference between total and permanent force, hence $\theta_{\text{per}} - \theta_{\text{tot}}$ increases. At $t = t_1$ the difference between the vehicle yaw θ and the desired angle θ_{tot} exceeds the threshold $\Theta_2 = 10°$, then the automaton in Figure 9.8 moves the control authority to the robot to perform the avoidance maneuver (then $\omega = \omega_{\text{robot}}$ and $a = a_{\text{robot}}$ in mode III). At $t = t_2$ the difference $\theta - \theta_{\text{tot}}$ between the vehicle yaw θ and the angle of the total field θ_{tot} falls beyond the threshold $\Theta_1 = 5°$, then the control authority is given back to the user (mode IV in Figure 9.8). Notice that the same switches of the control authority of times t_1 and t_2 take place at time t_3 and t_4 respectively. For $t > t_4$ the avoidance maneuver is competed, then the difference $\theta_{\text{per}} - \theta_{\text{tot}}$ between the angle of the permanent force θ_{per} and the angle of the total field θ_{tot} converges to zero since the effect of the limit cycle vanishes.

Standard navigation

The navigation of two agents interacting with dynamic obstacles in an indoor environment is shown in Figure 9.25. The vehicles are moving from the left bottom

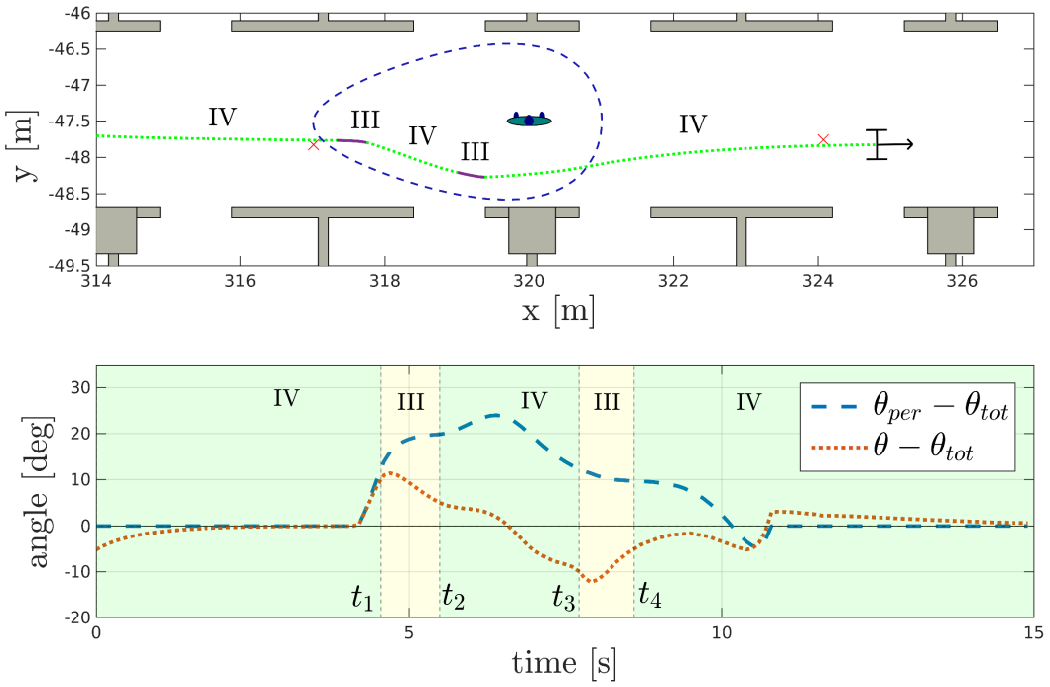

Figure 9.26: Effect of the hybrid automaton in Figure 9.8 during an obstacle avoidance maneuver. The vehicle trajectory is reported with different linestyle to enlighten different working modes of Figure 9.8 [7].

corner of the map to the right top corner. Four avoidance maneuvers are enlightened.

In the first situation the obstacle is moving in the same direction of the vehicles but at smaller speed. Then, the robots execute an overtaking maneuver. Notice that vehicle 1 passes on the right of the obstacle, while vehicle 2 passes on its left. This natural behavior takes place since the terms f_1 and f_2 in the definition of the rotation of the limit cycle in (9.19) are dominant with respect to $f_3 \approx 0$, and hence the vehicles tend to slightly modified their trajectories.

A similar behavior is exhibited in the second situation, where the obstacle is moving towards the vehicles, that again split to pass one on the left and one on the right. In this case $f_3 \neq 0$ in (9.19), however the suggested rotational direction for the limit cycle by f_3 is in accordance with terms f_1 and f_2.

In the third situation the obstacle is still. Both vehicle turn right since a smaller modification of their headings is required (dominant effect of term f_2 in (9.19)).

In the fourth situation (which is encircled and associated with three magnifications corresponding to three subsequent time instants) the obstacle is moving back and forward along the Y-axis, then almost orthogonally to the vehicle headings (i.e., large effect of term f_3 in (9.19)). When vehicle 1 is approaching, the obstacle is moving up, then the term f_3 in (9.19) commands to deviate on the right to precautionary

pass behind the obstacle. Afterwards, vehicle 2 approaches the same obstacle which has inverted its velocity, then the term f_3 in (9.19) commands to deviate on the left. Notice also that the obstacle reorients its velocity towards vehicle 1. In this case, if the obstacle had accelerated significantly, the only way to avoid the collision for vehicle 1 would have been to significantly accelerate as well. Since we cannot pull the assisted person (constraint $a_{\text{robot}} \leq 0$) the collisions would have occurred. This shows a rather intuitive limit of the safety ensured by the proposed approach: in the improbable case where the obstacle exhibits a very competitive behavior by trying to force the collision, since the vehicle velocity cannot be augmented, the user safety is not necessarily guaranteed. Fortunately, in the practice, a much more cooperative behavior of the other humans is expected.

Interaction with humans in a real scenario

The dynamic collision avoidance strategy based on oval limit cycles has been tested also by simulating the interaction with humans moving in a real scenario recorded at ETH. The simulations are precautionary still based on uncooperative obstacles since the obstacles neglect the presence of the walkers; however the obstacle motions correspond to real human trajectories.

Figure 9.27 shows two users moving from the top to the bottom. In the left picture they interact with a single human moving upwards. In this case they perform a split maneuver by passing the obstacle on different sides. Notice that $f_3 \neq 0$ in (9.19), however the suggested rotational direction for the limit cycle by f_3 is in accordance with terms f_1 and f_2. In the left picture, the two users avoid a collision with two approaching humans by passing both on the same side. Notice that the user on the left part deviates his/her trajectory to accommodate the maneuver of the other user, which is closer to the approaching humans, because of the effect of the repulsive force F_h.

Figure 9.28 shows two users moving orthogonally to a single human moving upwards. Notice that the implemented weights in the choice of the direction of rotation of the limit cycle (9.19) are such that the robots precautionary pass behind the human (dominant effect of the term f_3 in (9.19)).

Figures 9.29 and 9.30 show a crossing situation similar to Figure 9.28 in which two human obstacles moving in opposite direction are involved. Figure 9.29 enlightens the interaction with first human. Notice that the vehicles are going to pass behind the first human (dominant effect of the term f_3 in (9.19)). Since the vehicle headings considerably depart from the orientation suggested by the limit cycles (i.e., $h \cdot h_U < \cos\Theta$) the braking maneuver $\dot{v} = a_{\text{robot}} = -\kappa v$ is enabled by the automaton in Figure 9.8. As a consequence, the vehicles decelerate. Notice that the deceleration without relevant deviations in the robot trajectories is sufficient to avoid the collision. Figure 9.30 enlightens the interaction with second human. In this case the vehicle deceleration is not sufficient to ensure the collision is avoided. As a consequence, the vehicles deviate their trajectories to pass behind the second

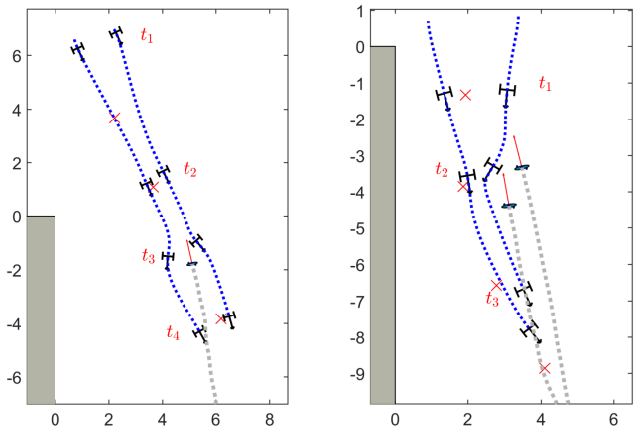

Figure 9.27: Users depicted in subsequent time instants interacting with approaching human obstacles. The human obstacle in the left figure is reported at time instant t_3, while the ones in the right figure are reported at time instant t_2 [7].

human.

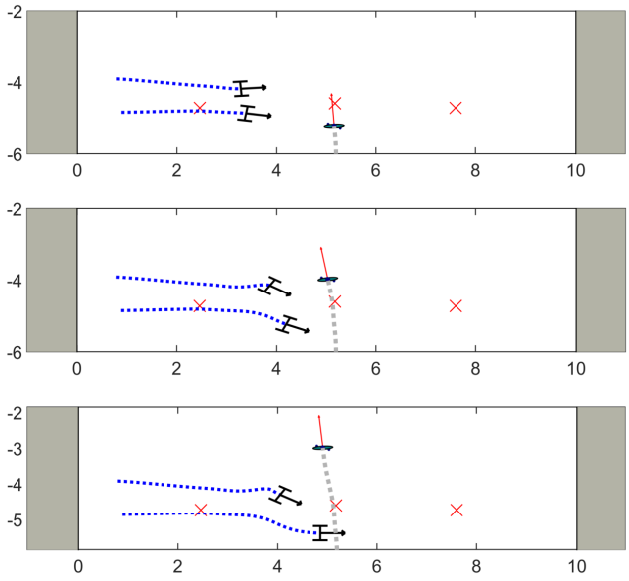

Figure 9.28: Users and single human obstacle interacting in a cross maneuver. The three figures represent subsequent time instants [7].

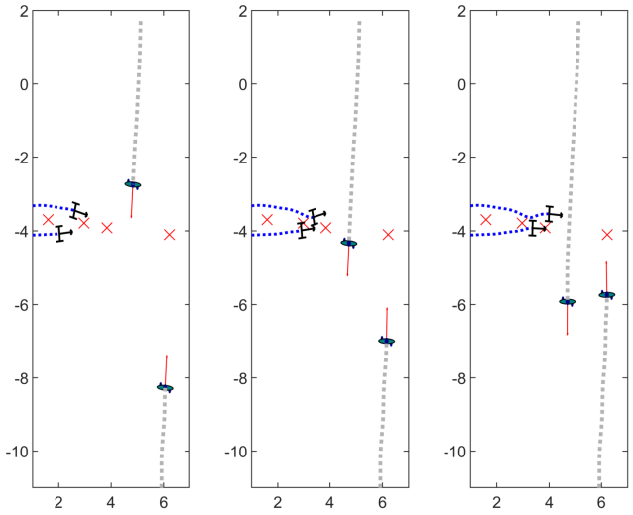

Figure 9.29: Users interaction with two humans moving in opposite direction in a crossing maneuver. The interaction with the first human is enlighten. The three pictures represent subsequent time instants [7].

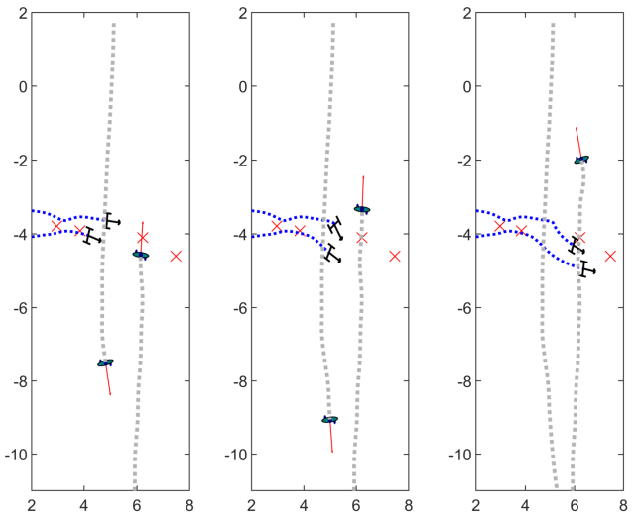

Figure 9.30: Users interaction with two humans moving in opposite direction in a crossing maneuver. The interaction with the second human is enlighten. The three pictures represent subsequent time instants [7].

9.9 Final comments

In this work we propose a control strategy to guide a group of assistive robotic walkers interacting and supporting a group of humans moving in a complex environment.

We adopt the force field used in the well-known social force model for human-like motion generation, to define the vehicle reference headings, i.e., the headings that the assistive vehicles should have to safely move in the environment towards the goal. The control authority is shared between the user and the robot by leaving the human in control of the motion whenever he/she is safely moving in the environment. The robot just limits the set of headings that the user can impose to his/her assistive vehicle in order to avoid collisions and move towards the goal.

The research also proposes to improve the force field of the social force model by including an innovative dynamic obstacle avoidance strategy based on limit cycles: the new oval shape of the trajectory offers several improvements with respect to the existing circular shapes because of its asymmetric structure.

The overall strategy is successfully tested by simulations in real environments and in the presence of moving uncooperative obstacle following real human trajectories.

Further studies will be mainly focused on the experimental validation of the proposed technique. These challenging experiments may comprise evaluation of the comfort offered by the strategy via proper questionnaires prepared by psychology experts. We also aim to provide a formal proof on the asymptotic stability of the innovative oval limit cycles.

Chapter 10

Conclusions

10.1 Research evaluation

The thesis proposes several strategies to control an assistive robotic walker by sharing the control authority with the user. The robot is supposed to be capable of solving autonomously the task (e.g., moving from a starting position to a point of interest), but, in order to exploit user's residual abilities and improve the comfort, it shares the control authority. The authority is allocated on the basis of an error metric, quantifying the distance between the current vehicle heading and the desired movement direction to perform the task. If the user is safely performing the task, he/she is endowed with control authority, so that the residual abilities are kept in training and the feeling of autonomy is enhanced. Conversely, if the user is not capable of safely solving the task (for instance, he/she is going to collide with an obstacle), the robot intervenes by partially or totally taking the control authority to help the user and ensure his/her safety (for instance, avoiding the collision with the obstacle). We offered theoretical analyses and experimental validation of several guidance systems (see Table 10.1). In particular:

- braking guidance: the user is guided by differentially braking the rear wheels. Although the offered comfort is inferior to all the other guidance systems, the brakes are always available in every walker model (because of safety reasons) and very reliable, making the braking guidance the cheapest solution presented in the thesis;
- haptic guidance: the user is guided using vibrating bracelets. The followed trajectory is definitely the most comfortable since it is totally chosen by the user on the basis of the indication from the haptic system. However, the control of the vehicle motion is totally entrusted to the user, hence safety cannot be guaranteed;
- haptic and braking guidance: the haptic approach is extremely versatile, since it can be easily integrated with all the mechanical guidance solutions presented

in the thesis to improve the user's comfort. We studied the integration with the braking guidance, showing that we can obtain both comfort (from the haptic component) and performance guarantees (from the mechanical actuation);

- simulated passivity: the user is guided using actuated rear wheels. The robot alternates phases in which the user is in complete control of the motion and the sensing system estimates the desired user's velocity, with phases in which the robot has the control authority and steers the vehicle towards the path at the estimated desired user's speed. Overall, although this guidance system is active, the user has the impression the vehicle is moving at his/her desired speed, i.e., the robot simulates a passive behavior. Having an active vehicle is not necessary to guide the user, however it offers several extra potentialities (for example, an actuated walker can move autonomously and pick up a user in need in a remote location);

- variable stiffness handling: the user is guided by front steering wheels, whose compliance varies depending on the path following metric. The vehicle exhibits a variable stiffness behavior, gradually sharing the authority with the user. When the user is close to the path, he/she perceives the vehicle as compliant, having the impression than it is not actuated and he/she is free to move. The larger the distance, the stiffer the front steering system. Hence, the user perceives a vehicle gradually stiffer and stiffer, i.e., more and more difficult to steer away, as he/she gradually deviates from the path. The proposed analysis defines also a performance index quantifying the degree of collaboration between user and robot. Although this guidance system requires motors to steer the font wheels, several advantages are offered (passivity, more comfort, better localization accuracy...), and it allows a gradual transition of the control authority.

Overall, the use of haptic systems is recommended since it boosts the comfort and can be easily integrated with all the other solutions, which instead ensure safety. In terms of mechanical guidance, although the additional cost of the front motors, the gradual allocation by modifying the perceived physical interaction using a variable stiffness handling, is the most promising solution for comfort, safety, and accuracy.

The thesis also proposes two further studies on the concept of authority-sharing in stochastic and multiagent frameworks. The probabilistic sharing provides the user with control authority if he/she acts more reliably (in a probabilistic sense) than the robot. Finally, the social force control applies authority-sharing in a multiagent scenario, by defining the human-like reference motion for the robots via the social force model an limiting the set of headings that each user can impose to his/her vehicle to avoid collisions and move towards the goal. These last two studies are successfully validated with preliminary experiments and extensive simulations.

Authority-sharing strategy	Advantages	Disadvantages
Braking	cheapness, reliability	comfort
Haptic	comfort, versatility	safety
Braking + haptic	cheapness, comfort, reliability	sharp authority allocation
Simulated passivity	potentialities of active actuation	sharp authority allocation
Variable stiffness	graduality, comfort, accuracy	front motors required

Table 10.1: Comparison between authority-sharing strategies.

10.2 Future work

A first possible improvement of the proposed strategies is an adaptive threshold system. If the user is uncooperative and/or very far from the safe reference path, the robot intervenes and takes the control authority. The condition in which the robot has full authority to ensure safety (e.g., stiff vehicle in the variable stiffness vehicle handling of Chapter 7) corresponds to a threshold value, representing the maximum deviation that the robot tolerates. The larger the threshold (and hence the tolerated deviation), the larger the user's freedom. However, larger deviations may correspond to a relevant proximity of the obstacles and hence a higher risk of collision. In our research, such thresholds have been chosen as a reasonable trade off between user's freedom and safety, known the experimental environment. A real-time adaptation is clearly expected to be more suitable, since the importance of small path following errors depends on the environment. For instance, in a narrow corridor a small threshold ensures small deviations from the path, avoiding collisions with the walls. Conversely, in an open hall, a larger threshold is preferred since the collision may happen only very far from the path.

In our work we have not provided a common definition of user's comfort for all the guidance systems. We related the comfort as number of control interventions (for braking and haptic guidance), qualitative psychological evaluation (for braking, haptic, and simulated passivity guidance), and human-robot cooperation index (for variable stiffness handling). Future developments will focus on the definition of a single common comfort metric suitable for all guidance systems.

Relevant improvements may be addressed in terms of experimental evaluation:

the thesis provides only preliminary experiments on probabilistic authority-sharing and an extensive simulation analysis for the multiagent framework. Experimental studies are required to further validate these strategies. Moreover, the proposed experiments for the other guidance systems took less than one hour per user. Longer and more ecological interactions with the robotic walker will be investigated.

As final remark, the robotic walker developed in ACANTO is still a prototype, so the hardware robustness and the complete integration between the software modules needed for a commercial purpose are still missing.

References

[1] ACANTO: A CyberphysicAl social NeTwOrk using robot friends. EU Project, February 2015.

[2] David A Abbink, Mark Mulder, and Erwin R Boer. Haptic shared control: smoothly shifting control authority? *Cognition, Technology & Work*, 14(1):19–28, 2012.

[3] A Pedro Aguiar and Joao P Hespanha. Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. *IEEE Transactions on Automatic Control*, 52(8):1362–1379, 2007.

[4] Daniel Althoff, Dirk Wollherr, and Martin Buss. Safety assessment of trajectories for navigation in uncertain and dynamic environments. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 5407–5412. IEEE, 2011.

[5] M. Andreetto, S. Divan, D. Fontanelli, and L. Palopoli. Path following with authority sharing between humans and passive robotic walkers equipped with low-cost actuators. *IEEE Robotics and Automation Letters*, 2(4):2271–2278, 2017.

[6] M. Andreetto, S. Divan, D. Fontanelli, L. Palopoli, and F. Zenatti. Path following for robotic rollators via simulated passivity. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6915–6922, Sept 2017.

[7] Marco Andreetto, Manuel Boldrer, Daniele Bullo, Daniele Fontanelli, and Luigi Palopoli. Social force model control of assistive robotic walkers. *Transactions on Robotics*, 2018. Submitted.

[8] Marco Andreetto, Stefano Divan, Francesco Ferrari, Daniele Fontanelli, Luigi Palopoli, and Domenico Prattichizzo. Combining Haptic and Bang-Bang Braking Actions for Passive Robotic Walker Path Following. *Transaction on Haptics*, 2018. Submitted.

[9] Marco Andreetto, Stefano Divan, Francesco Ferrari, Daniele Fontanelli, Luigi Palopoli, and Fabiano Zenatti. Simulating passivity for robotic walkers via authority-sharing. *IEEE Robotics and Automation Letters*, 3(2):1306–1313, 2018.

[10] Marco Andreetto, Stefano Divan, Daniele Fontanelli, and Luigi Palopoli. Hybrid feedback path following for robotic walkers via bang-bang control actions. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 4855–4860. IEEE, 2016.

[11] Marco Andreetto, Stefano Divan, Daniele Fontanelli, and Luigi Palopoli. Passive robotic walker path following with bang-bang hybrid control paradigm. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 1054–1060. IEEE, 2016.

[12] Marco Andreetto, Stefano Divan, Daniele Fontanelli, and Luigi Palopoli. Harnessing steering singularities in passive path following for robotic walkers. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 2426–2432. IEEE, 2017.

[13] Marco Andreetto, Stefano Divan, Daniele Fontanelli, Luigi Palopoli, and Antonino Sferlazza. Variable Stiffness. 2018. Submitted for Patent Pending.

[14] Gustavo Arechavaleta, Jean-Paul Laumond, Halim Hicheur, and Alain Berthoz. An optimality principle governing human walking. *IEEE Transactions on Robotics*, 24(1):5–14, 2008.

[15] Tommaso Lisini Baldi, Stefano Scheggi, Marco Aggravi, and Domenico Prattichizzo. Haptic guidance in dynamic environments using optimal reciprocal collision avoidance. *IEEE Robotics and Automation Letters*, 3(1):265–272, 2018.

[16] Andrea Balluchi, Antonio Bicchi, Aldo Balestrino, and Giuseppe Casalino. Path tracking control for dubin's cars. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 4, pages 3123–3128. IEEE, 1996.

[17] Yaakov Bar-Shalom, X Rong Li, and Thiagalingam Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[18] Mehdi Benallegue, Jean-Paul Laumond, and Alain Berthoz. A head-neck-system to walk without thinking. 2015.

[19] Paolo Bevilacqua, Marco Frego, Enrico Bertolazzi, Daniele Fontanelli, Luigi Palopoli, and Francesco Biral. Path planning maximising human comfort for

assistive robots. In *Control Applications (CCA), 2016 IEEE Conference on*, pages 1421–1427. IEEE, 2016.

[20] Paolo Bevilacqua, Marco Frego, Daniele Fontanelli, and Luigi Palopoli. Reactive planning for assistive robots. *IEEE Robotics and Automation Letters*, 3(2):1276–1283, 2018.

[21] Antonio Bicchi and Giovanni Tonietti. Fast and" soft-arm" tactics [robot arm design]. *IEEE Robotics & Automation Magazine*, 11(2):22–33, 2004.

[22] Alexander Broad, Jarvis Schultz, Matthew Derry, Todd Murphey, and Brenna Argall. Trust adaptation leads to lower control effort in shared control of crane automation. *IEEE Robotics and Automation Letters*, 2(1):239–246, 2017.

[23] John T Cacioppo and Louise C Hawkley. Perceived social isolation and cognition. *Trends in cognitive sciences*, 13(10):447–454, 2009.

[24] Alec Cameron and Hugh Durrant-Whyte. A bayesian approach to optimal sensor placement. *The International Journal of Robotics Research*, 9(5):70–88, 1990.

[25] Nalin A Chaturvedi, Amit K Sanyal, and N Harris McClamroch. Rigid-body attitude control. *IEEE Control Systems*, 31(3):30–51, 2011.

[26] Oscar Chuy, Yasuhisa Hirata, and Kazuhiro Kosuge. A new control approach for a robotic walking support system in adapting user characteristics. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 36(6):725–733, 2006.

[27] Paulo Coelho and Urbano Nunes. Path-following control of mobile robots in presence of uncertainties. *IEEE Transactions on Robotics*, 21(2):252–261, 2005.

[28] Ulises Cortes, Cristian Barrue, Antonio B Martinez, Cristina Urdiales, Fabio Campana, Roberta Annicchiarico, and Carlo Caltagirone. Assistive technologies for the new generation of senior citizens: the share-it approach. *International Journal of Computers in Healthcare*, 1(1):35–65, 2010.

[29] A. Cosgun, E.A. Sisbot, and H.I. Christensen. Guidance for human navigation using a vibro-tactile belt interface and robot-like motion planning. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 6350–6355, 2014.

[30] C. De Medio, F. Nicolò, and G. Oriolo. *Robot motion planning using vortex fields*, pages 237–244. Birkhäuser, Boston, 1991.

[31] Cosimo Della Santina, Matteo Bianchi, Giorgio Grioli, Franco Angelini, Manuel Catalano, Manolo Garabini, and Antonio Bicchi. Controlling soft robots: balancing feedback and feedforward elements. *IEEE Robotics & Automation Magazine*, 24(3):75–83, 2017.

[32] Francesco Farina, Daniele Fontanelli, Andrea Garulli, Antonio Giannitrapani, and Domenico Prattichizzo. Walking ahead: The headed social force model. *PloS one*, 12(1):e0169734, 2017.

[33] Daniele Fontanelli, Antonio Giannitrapani, Luigi Palopoli, and Domenico Prattichizzo. Unicycle steering by brakes: a passive guidance support for an assistive cart. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pages 2275–2280. IEEE, 2013.

[34] Daniele Fontanelli, Antonio Giannitrapani, Luigi Palopoli, and Domenico Prattichizzo. A passive guidance system for a robotic walking assistant using brakes. In *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*, pages 829–834. IEEE, 2015.

[35] A. Garrell and A. Sanfeliu. Cooperative social robots to accompany groups of people. *International Journal of Robotics Research*, 31(13):1675–1701, 2012.

[36] Nancy M Gell, Robert B Wallace, Andrea Z Lacroix, Tracy M Mroz, and Kushang V Patel. Mobility device use in older adults and incidence of falls and worry about falling: Findings from the 2011–2012 national health and aging trends study. *Journal of the American Geriatrics Society*, 63(5):853–859, 2015.

[37] F. Gemperle, T. Hirsch, A. Goode, J. Pearce, D. Siewiorek, and A. Smailigic. Wearable vibro-tactile display, 2003. Carnegie Mellon University.

[38] W Gharieb. Intelligent robotic walker design. In *International Conference on Automation, Robotics and Autonomous Systems*. Citeseer, 2006.

[39] R Brent Gillespie, J Edward Colgate, and Michael A Peshkin. A general framework for cobot control. *IEEE Transactions on Robotics and Automation*, 17(4):391–401, 2001.

[40] Rafal Goebel, Ricardo G Sanfelice, and Andrew R Teel. *Hybrid Dynamical Systems: modeling, stability, and robustness*. Princeton University Press, 2012.

[41] Deepak Gopinath, Siddarth Jain, and Brenna D Argall. Human-in-the-loop optimization of shared autonomy in assistive robotics. *IEEE Robotics and Automation Letters*, 2(1):247–254, 2017.

[42] Birgit Graf. An adaptive guidance system for robotic walking aids. *CIT. Journal of Computing and Information Technology*, 17(1):109–120, 2009.

[43] Paul Griffiths and R Brent Gillespie. Shared control between human and machine: Haptic display of automation during manual control of vehicle heading. In *Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2004. HAPTICS'04. Proceedings. 12th International Symposium on*, pages 358–366. IEEE, 2004.

[44] Louise C Hawkley, Ronald A Thisted, and John T Cacioppo. Loneliness predicts reduced physical activity: cross-sectional & longitudinal analyses. *Health Psychology*, 28(3):354, 2009.

[45] S. Hayati and S. T. Venkataraman. Design and implementation of a robot control system with traded and shared control capability. In *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*, pages 1310–1315 vol.3, May 1989.

[46] D. Helbing and P. Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51:4282–4286, 1995.

[47] M. A. Heller and W. Schiff. *The psychology of touch*. Psychology Press, 2013.

[48] Yasuhisa Hirata, Takahiro Baba, and Kazuhiro Kosuge. Motion control of omni-directional type walking support system" walking helper". In *Robot and Human Interactive Communication, 2003. Proceedings. ROMAN 2003. The 12th IEEE International Workshop on*, pages 85–90. IEEE, 2003.

[49] Yasuhisa Hirata, Asami Hara, and Kazuhiro Kosuge. Motion control of passive intelligent walker using servo brakes. *IEEE Transactions on Robotics*, 23(5):981–990, 2007.

[50] Yasuhisa Hirata, Shinji Komatsuda, and Kazuhiro Kosuge. Fall prevention control of passive intelligent walker based on human model. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 1222–1228. IEEE, 2008.

[51] Yasuhisa Hirata, Asami Muraki, and Kazuhiro Kosuge. Motion control of intelligent passive-type walker for fall-prevention function based on estimation of user state. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 3498–3503. IEEE, 2006.

[52] Yasuhisa Hirata, Asami Muraki, and Kazuhiro Kosuge. Motion control of intelligent walker based on renew of estimation parameters for user state. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 1050–1055. IEEE, 2006.

[53] Cunjun Huang, Glenn Wasson, Majd Alwan, Pradip Sheth, and Alexandre Ledoux. Shared navigational control and user intent detection in an intelligent walker. In *AAAI Fall 2005 Symposium*, 2005.

[54] Markus Huber, Yi-Huang Su, Melanie Krüger, Katrin Faschian, Stefan Glasauer, and Joachim Hermsdörfer. Adjustments of speed and path when avoiding collisions with another pedestrian. *PloS one*, 9(2):e89589, 2014.

[55] Sin-Yi Jiang, Chen-Yang Lin, Ko-Tung Huang, and Kai-Tai Song. Shared control design of a walking-assistant robot. *IEEE Transactions on Control Systems Technology*, 25(6):2143–2150, 2017.

[56] I. Karuei, K.E. MacLean, Z. Foley-Fisher, R. MacKenzie, S. Koch, and M. El-Zohairy. Detecting vibrations across the body in mobile contexts. In *Proc. ACM Int. Conf. on Human Factors in Computing Systems*, pages 3267–3276, 2011.

[57] K.-T. Khaw, N. Wareham, S. Bingham, A. Welch, R. Luben, and N. Day. Combined impact of health behaviours and mortality in men and women: The EPIC-Norfolk prospective population study. *PLOS Medicine*, 5(1):e12, 2008.

[58] Dong-Han Kim and Jong-Hwan Kim. A real-time limit-cycle navigation method for fast mobile robots and its application to robot soccer. *Robotics and Autonomous Systems*, 42(1):17–30, 2003.

[59] W. S. Kim, B. Hannaford, and A. K. Fejczy. Force-reflection and shared compliant control in operating telemanipulators with time delay. *IEEE Transactions on Robotics and Automation*, 8(2):176–185, Apr 1992.

[60] R. L. Koslover, B. T. Gleeson, J. T. de Bever, and W. R. Provancher. Mobile Navigation Using Haptic, Audio, and Visual Direction Cues with a Handheld Test Platform. *IEEE Transactions on Haptics*, 5(1):33–38, 2012.

[61] Ramu Krishnan. *Permanent magnet synchronous and brushless DC motor drives*. CRC Press/Taylor & Francis, 2010.

[62] Miroslav Krstic, Ioannis Kanellakopoulos, and Peter V Kokotovic. *Nonlinear and adaptive control design*. Wiley, 1995.

[63] Vladimir Kulyukin, Aliasgar Kutiyanawala, Edmund LoPresti, Judith Matthews, and Richard Simpson. iwalker: Toward a rollator-mounted wayfinding system for the elderly. In *RFID, 2008 IEEE International Conference on*, pages 303–311. IEEE, 2008.

[64] Jean-Paul Laumond. *Robot motion planning and control. Lectures Notes in Control and Information Sciences 229*, volume 3. 1998.

[65] Geunho Lee, Takanori Ohnuma, and Nak Young Chong. Design and control of jaist active robotic walker. *Intelligent Service Robotics*, 3(3):125–135, 2010.

[66] Unghui Lee, Jiwon Jung, Seunghak Shin, Yongseop Jeong, Kibaek Park, David Hyunchul Shim, and In-so Kweon. Eurecar turbo: a self-driving car that can handle adverse weather conditions. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 2301–2306. IEEE, 2016.

[67] Werner Leonhard. *Control of electrical drives*. Springer Science & Business Media, 2012.

[68] J. Lieberman and C. Breazeal. Tikl: Development of a wearable vibrotactile feedback suit for improved human motor learning. 23(5):919–926, 2007.

[69] R.W. Lindeman, J.L. Sibert, R. Mendez-Mendez, S. Patil, and D. Phifer. Effectiveness of directional vibrotactile cuing on a building-clearing task. In *Proc. ACM Int. Conf. on Human factors in computing systems*, pages 271–280. ACM, 2005.

[70] R.W. Lindeman, Y. Yanagida, H. Noma, and K. Hosaka. Wearable vibrotactile systems for virtual contact and information display. In *Virtual Reality*, volume 9, pages 203–213. Springer, 2006.

[71] Cheng-Kai Lu, Yi-Che Huang, and Cheng-Jung Lee. Adaptive guidance system design for the assistive robotic walker. *Neurocomputing*, 170:152–160, 2015.

[72] Valerio Magnago, Marco Andreetto, Stefano Divan, Daniele Fontanelli, and Luigi Palopoli. Ruling the control authority of a service robot based on information precision. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7204–7210. IEEE, 2018.

[73] Sonia MartíNez and Francesco Bullo. Optimal sensor placement and motion coordination for target tracking. *Automatica*, 42(4):661–668, 2006.

[74] Maria Martins, Cristina Santos, and Anselmo Frizera. Online control of a mobility assistance smart walker. In *Bioengineering (ENBENG), 2012 IEEE 2nd Portuguese Meeting in*, pages 1–6. IEEE, 2012.

[75] Maria Martins, Cristina Santos, Anselmo Frizera, and Ramón Ceres. Real time control of the asbgo walker through a physical human–robot interface. *Measurement*, 48:77–86, 2014.

[76] Maria Martins, Cristina Santos, Eurico Seabra, Luis Basílio, and Anselmo Frizera. A new integrated device to read user intentions when walking with a smart walker. In *Industrial Informatics (INDIN), 2013 11th IEEE International Conference on*, pages 299–304. IEEE, 2013.

[77] Maria M Martins, Cristina P Santos, Anselmo Frizera-Neto, and Ramón Ceres. Assistive mobility devices focusing on smart walkers: Classification and review. *Robotics and Autonomous Systems*, 60(4):548–562, 2012.

[78] Christopher G Mayhew, Ricardo G Sanfelice, and Andrew R Teel. Quaternion-based hybrid control for robust global attitude tracking. *IEEE Transactions on Automatic control*, 56(11):2555–2566, 2011.

[79] Skye McLachlan, Jaimal Arblaster, OK Liu, Jaime Valls Miro, and L Chenoweth. A multi-stage shared control method for an intelligent mobility assistant. In *9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005.*, pages 426–429. IEEE, 2005.

[80] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 935–942, June 2009.

[81] Alain Micaelli and Claude Samson. *Trajectory tracking for unicycle-type and two-steering-wheels mobile robots*. PhD thesis, INRIA, 1993.

[82] Jaime Valls Miró, Vivien Osswald, Mitesh Patel, and Gamini Dissanayake. Robotic assistance with attitude: A mobility agent for motor function rehabilitation and ambulation support. In *Rehabilitation Robotics, 2009. ICORR 2009. IEEE International Conference on*, pages 529–534. IEEE, 2009.

[83] N. Moshtagh and A. Jadbabaie. Distributed geodesic control laws for flocking of nonholonomic agents. *IEEE Transactions on Automatic Control*, 52(4):681–686, 2007.

[84] Mark Mulder and David A Abbink. Sharing control with elderly drivers: Haptic guidance during curve negotiation. *IFAC Proceedings Volumes*, 43(13):310–315, 2010.

[85] Mark Mulder, David A Abbink, and Erwin R Boer. The effect of haptic guidance on curve negotiation behavior of young, experienced drivers. In *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on*, pages 804–809. IEEE, 2008.

[86] United Nations. World population ageing: 1950-2050. *New York: Department of Economic and Social Affairs*, 2002.

[87] P. Nazemzadeh, D. Fontanelli, and D. Macii. Optimal Placement of Landmarks for Indoor Localization using Sensors with a Limited Range. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, Madrid, Spain, Oct. 2016. IEEE.

[88] Payam Nazemzadeh, Federico Moro, Daniele Fontanelli, David Macii, and Luigi Palopoli. Indoor Positioning of a Robotic Walking Assistant for Large Public Environments. *IEEE Trans. on Instrumentation and Measurement*, 64(11):2965–2976, Nov 2015.

[89] R. Olfati-Saber. Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.

[90] Anne-Hélène Olivier, Antoine Marin, Armel Crétual, Alain Berthoz, and Julien Pettré. Collision avoidance between two walkers: Role-dependent strategies. *Gait & posture*, 38(4):751–756, 2013.

[91] D. A. Paley, N. E. Leonard, R. Sepulchre, D. Grunbaum, and J. K. Parrish. Oscillator models and collective motion. *IEEE Control Systems*, 27(4):89–105, 2007.

[92] Xanthi S Papageorgiou, Costas S Tzafestas, Petros Maragos, Georgios Pavlakos, Georgia Chalvatzaki, George Moustris, Iasonas Kokkinos, Angelika Peer, Bartlomiej Stanczyk, Evita-Stavroula Fotinea, et al. Advances in intelligent mobility assistance robot integrating multimodal sensory processing. In *International Conference on Universal Access in Human-Computer Interaction*, pages 692–703. Springer, 2014.

[93] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision (ICCV)*, volume 00, pages 261–268, Sept. 2010.

[94] Michael A Peshkin, J Edward Colgate, Wit Wannasuphoprasit, Carl A Moore, R Brent Gillespie, and Prasad Akella. Cobot architecture. *IEEE Transactions on Robotics and Automation*, 17(4):377–390, 2001.

[95] Andrew J Rentschler, Rory A Cooper, Bruce Blasch, and Michael L Boninger. Intelligent walkers for the elderly: Performance and safety testing of va-pamaid robotic walker. *Journal of rehabilitation research and development*, 40(5):423, 2003.

[96] A. Riener. *Sensor Actuator Supported Implicit Interaction in Driver Assistance Systems*. Springer, 2010.

[97] Masao Saida, Yasuhisa Hirata, and Kazuhiro Kosuge. Development of passive type double wheel caster unit based on analysis of feasible braking force and moment set. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 311–317. IEEE, 2011.

[98] Claude Samson. Control of chained systems application to path following and time-varying point-stabilization of mobile robots. *IEEE transactions on Automatic Control*, 40(1):64–77, 1995.

[99] Stefano Scheggi, Marco Aggravi, Fabio Morbidi, and Domenico Prattichizzo. Cooperative human-robot haptic navigation. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 2693–2698, 2014.

[100] Stefano Scheggi, Marco Aggravi, and Domenico Prattichizzo. Cooperative navigation for mixed human–robot teams using haptic feedback. *IEEE Transactions on Human-Machine Systems*, 47(4):462–473, 2017.

[101] Stephen Se, David Lowe, and Jim Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *The international Journal of robotics Research*, 21(8):735–758, 2002.

[102] Fei Shi, Qixin Cao, Chuntao Leng, and Hongbing Tan. Based on force sensing-controlled human-machine interaction system for walking assistant robot. In *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, pages 6528–6533. IEEE, 2010.

[103] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics. Modelling, Planning and Control*. Springer, 2009.

[104] Richard C Simpson and Simon P Levine. Adaptive shared control of a smart wheelchair operated by voice control. In *Intelligent Robots and Systems, 1997. IROS'97., Proceedings of the 1997 IEEE/RSJ International Conference on*, volume 2, pages 622–626. IEEE, 1997.

[105] Bruno Sinopoli, Luca Schenato, Massimo Franceschetti, Kameshwar Poolla, Michael I Jordan, and Shankar S Sastry. Kalman filtering with intermittent observations. *IEEE transactions on Automatic Control*, 49(9):1453–1464, 2004.

[106] D Soetanto, L Lapierre, and A Pascoal. Adaptive, non-singular path-following control of dynamic wheeled robots. In *IEEE Conf. on Decision and Control*, volume 2, pages 1765–1770. IEEE, 2003.

[107] Kai-Tai Song and Sin-Yi Jiang. Force-cooperative guidance design of an omni-directional walking assistive robot. In *Mechatronics and Automation (ICMA), 2011 International Conference on*, pages 1258–1263. IEEE, 2011.

[108] Kai-Tai Song and Chen-Yang Lin. A new compliant motion control design of a walking-help robot based on motor current and speed measurement. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 4493–4498. IEEE, 2009.

[109] Philippe Soueres, Andrea Balluchi, and Antonio Bicchi. Optimal feedback control for route tracking with a bounded-curvature vehicle. *International Journal of Control*, 74(10):1009–1019, 2001.

[110] Micah Steele and R Brent Gillespie. Shared control between human and machine: Using a haptic steering wheel to aid in land vehicle guidance. In *Proceedings of the human factors and ergonomics society annual meeting*, volume 45, pages 1671–1675. SAGE Publications, 2001.

[111] Martin F Stoelen, Alberto Jardón Huete, Virginia Fernández, Carlos Balaguer, and Fabio Bonsignorio. An information-theoretic approach to modeling and quantifying assistive robotics hri. In *Proceedings of the 6th international conference on Human-robot interaction*, pages 257–258. ACM, 2011.

[112] Sajjad Taghvaei, Yasuhisa Hirata, and Kazuhiro Kosuge. Control of a passive walker using a depth sensor for user state estimation. In *Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on*, pages 1639–1645. IEEE, 2011.

[113] Renpeng Tan, Shuoyu Wang, and Yinlai Jiang. Path tracking control considering center of gravity shift and load change for an omni-directional walker. In *Information and Automation (ICIA), 2010 IEEE International Conference on*, pages 672–675. IEEE, 2010.

[114] Renpeng Tan, Shuoyu Wang, Yinlai Jiang, Kenji Ishida, and Masanori Nagano. Adaptive controller for motion control of an omni-directional walker. In *Mechatronics and Automation (ICMA), 2010 International Conference on*, pages 156–161. IEEE, 2010.

[115] H. G. Tanner, A. Jadbabaie, and G. J. Pappas. Flocking in fixed and switching networks. *IEEE Transactions on Automatic Control*, 52(5):863–868, 2007.

[116] Jan BF Van Erp, Hendrik AHC Van Veen, Chris Jansen, and Trevor Dobbins. Waypoint navigation with a vibrotactile waist belt. *ACM Trans. on Applied Perception (TAP)*, 2(2):106–117, 2005.

[117] Bram Vanderborght, Alin Albu-Schäffer, Antonio Bicchi, Etienne Burdet, Darwin G Caldwell, Raffaella Carloni, MG Catalano, Oliver Eiberger, Werner Friedl, Ganesh Ganesh, et al. Variable impedance actuators: A review. *Robotics and autonomous systems*, 61(12):1601–1614, 2013.

[118] Peter Vas. *Sensorless vector and direct torque control*. Oxford Univ. Press, 1998.

[119] Andreas Wachaja, Pratik Agarwal, Mathias Zink, Miguel Reyes Adame, Knut Möller, and Wolfram Burgard. Navigating blind people with a smart walker. In

Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on, pages 6014–6019. IEEE, 2015.

[120] JM Hostalet Wandosell and Birgit Graf. Non-holonomic navigation system of a walking-aid robot. In *Robot and Human Interactive Communication, 2002. Proceedings. 11th IEEE International Workshop on*, pages 518–523. IEEE, 2002.

[121] Yina Wang and Shuoyu Wang. A new directional-intent recognition method for walking training using an omnidirectional robot. *Journal of Intelligent & Robotic Systems*, 87(2):231–246, 2017.

[122] Glenn Wasson, Pradip Sheth, Majd Alwan, Kevin Granata, Alexandre Ledoux, and Cunjun Huang. User intent in a shared control framework for pedestrian mobility aids. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 3, pages 2962–2967. IEEE, 2003.

[123] Haoyong Yu, Matthew Spenko, and Steven Dubowsky. An adaptive shared control system for an intelligent mobility aid for the elderly. *Autonomous Robots*, 15(1):53–66, 2003.

[124] Fabiano Zenatti, Daniele Fontanelli, Luigi Palopoli, David Macii, and Payam Nazemzadeh. Optimal placement of passive sensors for robot localisation. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 4586–4593. IEEE, 2016.