Simulating passivity for Robotic Walkers via Authority-Sharing

Marco Andreetto, Stefano Divan, Francesco Ferrari, Daniele Fontanelli, Luigi Palopoli and Fabiano Zenatti

Abstract—We consider a robotic walking assistant used to guide senior users across a crowded space. The problem we address is how to guide the user using motorised back wheels. Our strategy aims to simulate a passive behaviour in which the forward velocity is the one imposed by the user, who receives the impression of controlling the motion. The result is obtained leaving the user in control (without any actuation) when she/he follows straight lines, while the motors kick in when the user has to make a turn. We offer extensive theoretical proof of the validity of our strategy. The technique has been validated via extensive experimentation with a large group of older adults.

Index Terms—Service Robots, Authority-Sharing.

I. INTRODUCTION

The ageing of the population of modern countries opens attractive research and market opportunities for robotics experts. In particular, service robots are becoming popular as intelligent mobility aids for older adults, while the reduced mobility of a large number of older adults is the recognised cause and the consequence of a number of physical problems and of the cognitive decline [1], [2]. The reason for the interest toward inexpensive and easy to use robotic mobility aids is motivated by their supposed efficacy in helping their users remain active beyond the walls of their houses.

An interesting example is given by the robotic walker Fri-Walk [3], whose main purpose is to act as a navigation aid and guide the user through the environment along a planned path that satisfies his/her requirements [4], [5]. The device is intended as a navigation aid rather than an autonomous assistive vehicle. For this reason, users should be guided without sacrificing their perceived freedom of movement. In this direction, very inspirational is the paradigm of passive robots [6], which leave the responsibility of the locomotion to the user. Straightforward ways to build a passive robot is either to use actuated steering wheels [7] or to resort to electromagnetic brakes for differential drive [8], [9]. In the latter case, optimal control minimising the braking torques have been recently presented [10]. To modulate the braking action, an appropriate sensing system is needed to estimate the torques applied by the user. In order to reduce the system final cost and its complexity, a bang-bang passive walker solution can be found in [11], in which the vehicle is turned on the left (right) blocking the left (right) rear wheel. The solution is simple (on/off control action) and inexpensive (no need of additional hardware for braking modulation). However, the small set of manoeuvres it produces enables a relatively accurate tracking of the path, although with a questionable user comfort [12]. To summarise, passive robotics is an interesting paradigm for assistive navigation systems, but it needs a combination of front steering wheels and brakes for safety reasons. Possible alternatives are the use complex and expensive hardwares (i.e., force sensors) or bang-bang strategies that diminish the user’s comfort.

The use of an actuation for the back wheels of the vehicle will solve most of the problems, as the robot can generate a large set of “comfortable” manoeuvres. If needed an actuated walker can move autonomously and pick up a user in need in a remote location. Finally, the actuators can be used to generate emergency brakes for safety. Unfortunately, the presence of actuation disrupts the system passivity, with potential safety problems, which can be dealt with using direct or indirect user interfaces [13]. For the former, user commands/intentions are directly communicated to the device through joysticks [14], force sensors [15], [16], [17], turn buttons and voice commands/navigator support [18]. Indirect interfaces recognises 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 [19]; in [20] a depth camera is used to track the user limbs; in [21] current sensors and wheel rotational encoders are used to estimate the user applied forces.

The indirect interface adopted in this work preserves a safe and intuitive behaviour of the system without additional hardware, using simulated passivity with an actuated device following a desired path. A similar approach is in [22], where a motion control solution is applied to an omnidirectional robot equipped with force sensors. In our previous work [23], the passive behaviour is simulated by means of alternating phases when the user is in control and her/his 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). The proposed paper extends the preliminary solution in [23] in four important directions. First, the vehicle controlled velocity is designed on the basis of the intensity of the correction and it is proportionate to the deviation, whereas in [23] a braking action was applied to the robot also to implement small corrections, thus reducing the comfort. This velocity adaptation still preserves the user safety guaranteeing that s/he is never pulled. Second, the path following controller adopted here removes a singularity that was present in [23], thus ensuring global asymptotic stability, a better comfort and the safety. Third, the length of the two phase is adapted to the user behaviour, with a significant improvement on the system’s performance. Finally, the system has been tested with a group of older adults and the qualitative and quantitative analysis of the results is reported in the paper. The users’ evaluation gives a concrete grounding to our notion of comfort, which relates to the number of control actions and the adaptation to the user pace.

The paper is organised as follows. Section II presents the mathematical background, formalises the path following problem and gives an overview of the solution. Section III presents the details for the passivity simulation and the adopted control law, while Section IV describes how to combine the controller simulated passivity in an authority-sharing paradigm. The experimental results with the elderly are reported in Section V, while Section VI concludes the paper with some final remarks.

II. PROBLEM FORMULATION AND SOLUTION OVERVIEW

The vehicle is modelled as a unicycle like robot, having differential kinematics

$$[\dot{x}, \dot{y}, \dot{\theta}]^T = [v \cos \theta, v \sin \theta, \omega]^T,$$

where the coordinates $[x, y]$ define the position of the vehicle reference point, i.e. the mid point $O_m$ of the rear axle, with respect to a fixed world frame $\langle W \rangle = \{O_w, X_w, Y_w, Z_w\}$ and $\theta$ is the orientation of the moving vehicle frame $\{O_m, X_m, Y_m, Z_m\}$ with respect to $\langle W \rangle$. The vehicle linear and angular velocities are denoted by $v$ and $\omega$, respectively. In this work, we explicitly make the assumption that $v > 0$, since it is reasonable to assume that the user usually pushes the rollerator to move in the environment, while whenever the user imposes $v \leq 0$, he/she is performing on spot manoeuvres that do not require any control action. Under the hypothesis of pure wheel rolling motion, the vehicle velocities $v$ and $\omega$ are linked to angular velocity of the rear wheels by

$$v = r \frac{\omega_R + \omega_L}{2}, \quad \omega = \frac{r (\omega_R - \omega_L)}{d},$$

where $\omega_L$ and $\omega_R$ are the rotational velocities of the left and right wheel, respectively, $r$ is the wheel radius and $d$ is the length of the rear axle.

To properly represent the path following problem, we use a Frenet reference frame $\langle F \rangle = \{O_f, X_f, Y_f, Z_f\}$ moving along the path (see Figure 1). Let $s$ be the curvilinear abscissa of the Frenet frame origin $O_f$, $\theta_d$ the desired yaw of the vehicle (i.e. the path orientation with respect to $\langle W \rangle$ in the point defined by $s$), and let $[l_x, l_y]$ be the coordinates of the vehicle reference point $O_m$ in the Frenet frame. Define the vehicle orientation error as $\tilde{\theta} := \theta - \theta_d$. Using this new set of coordinates $[l_x, l_y, \tilde{\theta}]$, the differential kinematics of the vehicle (1) is rewritten as [24]

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

where $c(s) = \frac{\omega_R(s)}{r}$ is the path curvature and the velocity $\dot{s}$ of the Frenet frame is an auxiliary control input. The state of the vehicle can be equivalently represented as $\chi = [l_x, l_y, \tilde{\theta}]^T$ or $\chi = [x, y, \theta]^T$.

Using the coordinates $[l_x, l_y, \tilde{\theta}]$, the path following problem is considered solved if

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

where $t$ denotes the time, and $l_\infty > 0$ and $\tilde{\theta}_\infty > 0$ are positive arbitrary tolerated errors.

A. Solution overview

The path following problem (4) has to be solved using the available actuators, i.e. the rear motors. The motors are controlled to impose the wheel velocities $\omega_R$ and $\omega_L$ to the right and to the left wheel, respectively. According to (2), whenever the wheel velocities $\omega_R$ and $\omega_L$ are chosen, the vehicle velocities $v$ and $\omega$ are defined as well. While the vehicle angular velocity $\omega$ can be chosen to control the yaw $\theta$ 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 $v$ [11], [7], [12]. In this work, instead, the robot is active, hence the motor velocities in (2) are used as input, therefore, as in [23], 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 2:

**Robot in control:** The control authority is given to the robot. The wheel velocities $\omega_R$ and $\omega_L$ are controlled and the forward...
velocity \( v = v^\star \) and the angular velocity \( \omega = \omega^\star \) are imposed to the vehicle as in (2);

**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 behaviour in **Robot in control** mode is here simulated by imposing a controlled velocity \( v^\star \) close (or even equal) to \( v_{\text{user}} \), 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) are satisfied, in the **Robot in control** mode only the angular velocity \( \omega^\star \) can be freely determined.

The overall controller implementing simulated passivity via authority-sharing is then composed by two ingredients: (i) a path following control law ensuring (4) and simulating a passive robot, i.e. suitably computing the forward velocity \( v^\star \) given the desired user velocity \( v_{\text{user}} \), to be applied in the **Robot in control** mode; (ii) a switching strategy between the two modes based on the user behaviour and implementing the simulated passivity via authority-sharing paradigm.

### III. Simulating Passivity in **Robot in control** Mode

The path following problem (4) is decoupled in two subproblems. In order to simulate passivity and to improve the user balance, the robot should not pull the assisted person, that is the vehicle should not increase the speed \( v_{\text{user}} \). To this end, we design two alternative strategies to compute \( v^\star \), whose on-line selection is determined as described in Section III-A. Then, we design a control law \( \omega = \omega^\star(\chi) \) that correctly steers the vehicle regardless of the forward velocity \( v \) of the vehicle.

#### A. Forward velocity selection to simulate passivity

1) **Velocity projection:** Whenever the angular velocity \( \omega \neq 0 \), one of the two wheels has a larger velocity than the vehicle reference point velocity \( v \) (according to Equation (2)). For instance, if the vehicle turns right, the left wheel is faster than both the right wheel and \( v \). 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 \( v_{\text{user}} \). In particular, if the requested angular velocity is positive, i.e. \( \omega^\star > 0 \), the vehicle turns left and the right wheel is the fastest, the fastest one, is set to \( \omega_R = v_{\text{user}}/r \). According to (2), we finally get

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

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 3.

2) **Braking actuation:** Since the motors directly command the wheel velocities, projecting the user’s velocity \( v_{\text{user}} \) on the fastest wheel might still generate discomfort if the correction that the robot has to apply is relevant. Consider for example Figure 4 in the **User in control** mode (centre 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 angular the velocities is relevant (right part of Figure 4). 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 4).

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 \( \omega \), 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 \( \omega \)), 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

\[
v^\star = \begin{cases} v_{\text{user}} - \frac{|\omega^\star| d}{2}, & \text{when } |\omega - \omega^\star| \leq \Omega, \\ \alpha v_{\text{user}}, & \text{when } |\omega - \omega^\star| > \Omega, \end{cases}
\]

where \( \Omega > 0 \) and \( \alpha \) are two comfort parameters to be tailored on the specific user’s requirements.
B. Path Following Controller

It is a common practice in path following problems, to define an approaching angle function \( \delta(\cdot) \) defining the manoeuvre that the vehicle has to take to approach and follow the path like in [24], [25]. In the rest of this section we will prove that if \( \delta(\cdot) \) is a function of \( l_y \) in (3) and satisfies some mild assumptions, the path following problem (4) is solved with \( l_\infty = \tilde{\theta}_\infty = 0 \) if \( e_\theta \triangleq \tilde{\theta} - \delta(l_y) \) converges to zero, where \( \tilde{\theta} \) is again reported in (3). The auxiliary control input \( \tilde{s} \) is designed to stabilise the path following error \( l_\gamma \).

**Theorem 1 (Attitude stabilisation):** Consider the vehicle kinematic model (3) and assume that the vehicle in persistent motion, i.e. the forward velocity \( v \geq 0 \) satisfies \( \lim_{t \to \infty} v(t) \neq 0 \), and that the approaching angle \( \delta(l_y) \) is differentiable, strictly monotonic and odd satisfying \( l_y \delta(l_y) \leq 0 \) and \( |\delta(l_y)| \leq \frac{\pi}{2} |l_y| \).

Then the control law

\[
\omega = \omega^*(\chi) = v \left( \gamma(\chi) - \kappa (\tilde{\theta} - \delta(l_y)) \right),
\]

where \( \kappa \) is a constant gain such that \( v\kappa > 0 \) and

\[
\gamma(\chi) = c(s) \tilde{\xi} + (-c(s) \tilde{\xi} l_x + \sin \tilde{\theta}) \frac{d\delta}{dl_y}(l_y),
\]

\[
\tilde{s} = v \tilde{\xi},
\]

\[
\tilde{\xi} = \cos(\tilde{\theta}) + \kappa_x l_x,
\]

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

**Assumption 2:** Consider the Lyapunov function \( V = \frac{1}{2} e_\theta^2 \), whose derivative along the solutions is

\[
\dot{V} = e_\theta \dot{e}_a = e_\theta \left( \omega - c(s) \tilde{s} - l_y \frac{d\delta}{dl_y}(l_y) \right) = e_\theta (\omega - v \gamma(\chi)).
\]

By imposing \( \omega = \omega^*(\chi) \) in (6), we get

\[
\dot{V} = -ve_\theta e_\theta < 0 \quad \forall e_\theta \neq 0,
\]

that ensures the global convergence of \( e_\theta \) to zero.

IV. SIMULATED PASSIVITY VIA AUTHORITY-SHARING

The overall vehicle passive behaviour 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 2. The switching strategy is synthesised with a synergistic use of two different ideas: a time based and a behavioural based approaches. More precisely, in the *User in control* mode, the vehicle behaves passively and estimates the user’s velocity \( v_{\text{user}} \) for a time window \( T_\delta \). In the *Robot in control* mode, the motors impose the velocities \( v = v^* \) and \( \omega = \omega^*(\chi) \) for a maximum time window of \( T_\omega \), unless the user autonomously and approximately follows the path, where the control authority is given back to the user. This complex switching strategy is formalised using hybrid system theory [26].

A. Behavioural 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 \( \omega = \omega^*(\chi) \). 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 III. 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 \( |e_\theta| \).

![Image](image-url)

Because of the monotonicity of \( \delta(\cdot) \), \( |l_y| > l_y^* = \delta^{-1}(\theta_{q_2}) \) implies \( |\delta(l_y)| > \theta_{q_2} \). Therefore, inside region \( |e_\theta| \leq \theta_{q_2} \), Equation (11) holds \( \forall |l_y| > l_y^* \).

\[
V_2 < 0 \quad \forall |l_y| > l_y^*.
\]
Condition (12) ensures that \(|y|\) enters in finite time and remains inside the region \(|y| \leq \varphi_y \). Inside this region, the monotonicity of \(\delta(\cdot)\) implies that \(|\delta(\varphi_y)| \leq \varphi_y \). This bound on \(\delta(\varphi_y)\) inside the region \(\varphi_y \leq \varphi_{y_2}\), implies \(\|\dot{\varphi}_y\| \leq \varphi_{y_2}\) (indeed, \(\varphi_y\) is the analytic solution of the inequality \(|\varphi_y - \varphi(\varphi_y)| \leq \varphi_{y_2}\). Furthermore, once the solution enters the region \(|y| \leq \varphi_y \), we have \(\dot{V}_2 = v\left(-\kappa_x l_y^2 + l_y \sin \hat{\varphi}\right) \leq v\left(-\kappa_x l_y^2 + l_y^2\right) \leq 0\), i.e.
\[
\dot{V}_2 < 0, \quad \forall l_x > \sqrt{\frac{\varphi_y}{\kappa_x}} = \sqrt{\frac{\varphi_y}{\kappa_x}},
\]
which implies that the solution enters in finite time and remains inside the region \(|x| \leq \varphi_y \). The proof follows by noticing that \(\lim_{\varphi_{y_2} \to 0} l_y = \lim_{\varphi_{y_2} \to 0} \varphi_y = \lim_{\varphi_{y_2} \to 0} \hat{\varphi} = 0\), hence, given the tolerated errors \(l_\infty\) and \(\hat{\varphi}_\infty\), one can always find an upper hysteresis threshold \(\varphi_y\) such that the path following conditions (4) hold.

**Remark 1 (Nonsingular controller):** In our previous work [23], a singularity was generated by the position of the Frenet on the point of the path closest to the vehicle. Intuitively, this happens when the vehicle is on the centre of the path curvature since the closest point on the path is not well defined. Inspired by [24], we have added a dynamic to the Frenet frame, which is modelled with \(\dot{s}\) in (7). As consequence, the control law (6) is nonsingular and globally convergent at the price of three coordinates to express the path following, i.e. \(l_x, y\) and \(\dot{\varphi}\) in (3), instead of just two as in [23]. This improvement is not only technical, but it is extremely beneficial and safer for the particular class of end-users of the system: indeed, close to the singularity of [23], the commanded angular velocity may be very large, hence yielding a sudden motion of the vehicle that may potentially harm the user’s equilibrium.

### B. 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 behaviour logic variable \(q\), described previously. Roughly speaking, \(p\) determines if the **authority** can be given to the robot, while \(q\) determines if the **authority** is given to the robot.

To model the activation time of the two control modes, we introduce two additional hybrid states \(\tau_U\) and \(\tau_R\) acting as timers. The timer of the **User in control mode** has hybrid dynamics
\[
\begin{aligned}
\begin{cases}
\dot{\tau}_U &= 1 - p, \\
\dot{\rho} &= 0, \\
\dot{h} &= 0,
\end{cases} & \quad \tau_U \leq T_{U}, \quad p \in \{0, 1\}, \\
\begin{cases}
\tau^+_U &= 0, \\
p^+ &= \min(q, 1 - p), \\
h^+ &= \max(\sin((\Omega - \omega - \omega^*)) \cdot 0),
\end{cases} & \quad \tau_U = T_{U}, \\
p \in \{0, 1\}, \\
h \in \{0, 1\},
\end{aligned}
\]
where the logic variable \(h \in \{0, 1\}\) determines how the forward velocity should be computed, according to (5), i.e.
\[
v^* = h \left( v_\text{user} - |\omega|^2 \cdot \frac{d}{2} \right) + (1 - h) \alpha v_\text{user}.
\]

The discrete dynamics \(p^+ = \min(q, 1 - p)\) in (14) 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
\[
\begin{aligned}
\begin{cases}
\dot{\tau}_R &= p, \\
\dot{\rho} &= 0, \\
\dot{h} &= 0,
\end{cases} & \quad \tau_R \leq T_{R}, \\
\begin{cases}
\tau^+_R &= 0, \\
p^+ &= 1 - p, \\
h^+ &= \max(\sin((\Omega - \omega - \omega^*)) \cdot 0),
\end{cases} & \quad \tau_R = T_{R}, \\
p \in \{0, 1\},
\end{aligned}
\]
Notice that, when \(p = 1\), the continuous dynamics in (16) is design to increment the timer \(\tau_R\) up to \(\tau_R = T_{R}\). Then \(\tau_R\) is reset to zero by the discrete dynamics in (17) and \(p\) jumps to 0 activating the dynamics (13) and (14). Finally, to ensure that the **User in control mode** immediately restarts when \(q\) jumps from 1 to 0, the discrete dynamics (8) is modified as
\[
\begin{aligned}
\begin{cases}
\dot{q} &= 0, \\
\dot{\tau}_R &= p, \\
\dot{\rho} &= 0, \\
\dot{h} &= 0,
\end{cases} & \quad [\epsilon_\theta, q]^T \in \mathbb{C}, \\
\begin{cases}
\tau^+_R &= 0, \\
p^+ &= 1 - q, \\
h^+ &= \max(\sin((\Omega - \omega - \omega^*)) \cdot 0),
\end{cases} & \quad [\tau_R, p]^T \in \mathbb{C}, \\
P^+ &= (1 - q)p, \\
\end{aligned}
\]

The result of the hybrid controller described in (13), (14), (16), (17), (18) and (19) is succinctly
\[
v = \begin{cases}
v_\text{user} & q = 0 \lor p = 0, \\
v^* & \text{otherwise},
\end{cases} \quad \omega = \begin{cases}
\omega_\text{user} & q = 0 \lor p = 0, \\
\omega^* & \text{otherwise}.
\end{cases}
\]

### C. Activation time tuning

The lengths of the activation times \(T_R\) and \(T_U\) clearly influence the user 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 precautionary upper bounded by a constant \(T_{R_{\text{max}}} > 0\). Conversely, the larger \(T_U\), the larger is the user comfort. However, the path following error is larger with uncooperative users. As a consequence, to ensure that the path following requirements (4) hold even with an uncooperative user, the **User in control mode** should last for a small amount of time. According to Theorem 3, the performance is guaranteed as long as \(|\epsilon_\theta| < \theta_{y_2}\) remains valid. The perturbation induced by the user can be quantified as
\[
|\epsilon_\theta^\text{user}| \leq |\omega_\text{user}| + |v_\text{user}^*| \gamma(\chi)|, \quad h_0
\]
hence the maximum increment of attitude error that the user can generate is
\[
\Delta \epsilon_\theta = \int_{t_0}^{t_0 + T_U} |\epsilon_\theta^\text{user}| d\tau, \quad \text{where} \ t_0 \ \text{is the}
\]
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 \( \Delta e_\theta \). Therefore, an adaptive discrete dynamics is added to (17) and (19) (i.e. at the end of each User in control mode), reported next

\[
T_U^+ = \min \left( \frac{a_1}{a_2|\dot{v}| + a_3|\dot{\omega}|}, T_U^{\text{max}} \right), \ T_U \in \mathbb{R}. \tag{20}
\]

\( \dot{v} \) and \( \dot{\omega} \) are the measured vehicle velocities while \( a_1, a_2, \) and \( a_3 \) are constant parameters to be tuned on the specific user. \( T_U^{\text{max}} > 0 \) 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. \( \dot{v} = \dot{\omega} = 0 \)). 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 V. Therefore, those parameters have been set via experiments with young testers (omitted for brevity) to guarantee an average behaviour between aggressive and loose control.

V. EXPERIMENTS

In this section we present the experimental results of the proposed approach. Two studies were conducted in which the older adult participants completed different paths using the FriWalk in one laboratory of the University of Trento. In the first study (with 4 males, 10 females, ageing 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, ageing 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 centre “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 [3] 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 showed to the participant the path to follow and explained the features of the robotic walker and its motion modality. All participants completed a first trial (which was common for everybody) to take confidence with the robot 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 5).

A. Quantitative analysis

We first present the quantitative analysis of the experimental results. The controller parameters adopted in the experiments are: \( \alpha = 0.37 \) and \( \Omega = 0.08 \) [rad/s] for (5), the hysteresis thresholds are set to \( \theta_{q_1} = 8^\circ \) and \( \theta_{q_2} = 15^\circ \), the maximum time for the Robot in control mode is \( T_R = 2 \) [s], while the parameters for the User in control mode in (20) are \( T_U^{\text{max}} = 4 \) [s], \( a_1 = 2, \ a_2 = 2 \) and \( a_3 = 1 \). Finally, we select \( \delta(t_q) = -\pi/2 \tanh(t_q) \). Four sample trajectories along a randomly selected path are reported in Figure 5. The localisation is provided with an EKF [27] fusing the encoder data and the QR codes, positioned on the floor using the deployment [28] and read by the available front camera pointing downwards. It may happens that a QR code reading is missed, hence a localisation jump can be detected in the estimated trajectory (see the dashed trajectory of Exp3 in Figure 5). Nonetheless, the controller is able to correctly steer the user towards the desired path. Figure 6 and Figure 7 reports the time evolution of the error \( e_\theta \) and of the right \( \omega_R \) angular velocity, respectively, for the last 30 seconds of the Exp1 in Figure 5. 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 where \( q \neq 0 \) and \( p = 0 \)). This is the essence of the time based authority sharing presented in Section IV-B, whereas
the behavioural authority sharing (Section IV-A) takes place whenever \( q = p = 0 \). Finally, it has to be noted how the controller is very effective in controlling the error \( e_\theta \) in the region of the hysteresis (Figure 6).

For the actuation, it is evident from Figure 7 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.

**B. 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, 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 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.

**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 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.
In this paper, we have introduced a novel guidance solution 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 motorised back wheels. We offer ample theoretical analysis revealing the asymptotic stability of the solution and its correct behaviour in presence of singularities. The system has been validated with a large base of senior users. We report both quantitative analysis and users evaluation of the FriWalk. From a technical perspective, future works will concentrate in both quantitative analysis and users evaluation of the FriWalk. From a technical perspective, future works will concentrate in both quantitative analysis and users evaluation of the FriWalk.

VI. CONCLUSIONS

In this paper, we have introduced a novel guidance solution 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 motorised back wheels. We offer ample theoretical analysis revealing the asymptotic stability of the solution and its correct behaviour in presence of singularities. The system has been validated with a large base of senior users. We report both quantitative analysis and users evaluation of the FriWalk. From a technical perspective, future works will concentrate in both quantitative analysis and users evaluation of the FriWalk.