A Control Strategy for a Tethered Follower Robot for Pulmonary Rehabilitation

Bianchi Luciano, Buniak Esteban Alejandro, Ramele Rodrigo, *Member, IEEE,* and Santos Juan Miguel

***Abstract*—Patients that suffer Chronic Obstructive Pulmonary Disease (COPD) undergo a procedure called Pulmonary Rehabilitation that helps them to improve disease prognosis. Pulmonary Rehabilitation consists of different physical exercises and walking activities conducted at medical facilities under supervision of a physical therapist. In order to perform these procedures, patients require oxygen assistance, but the oxygen tank cannot be carried by the patient due to the musculoskeletal atrophy that characterize this pathology and external assistance is required. The assistance to transport the bulky oxygen tank can be provided by a robotic device that follows the patient while performing the physical activities. This work provides an initial study on the controlling mechanism of a differential tethered robot that implements a leader-follower configuration to carry the oxygen tank for these procedures. Two alternative control strategies are proposed. Results on a simulated and on a real prototype confirms the feasibility of the proposed solution.**

***Index Terms*—COPD, PR, SAR, IoRT, tethered**

I. Introduction

CHRONIC Obstructive Pulmonary Disease (COPD) is an umbrella term that describes several pulmonary affections. They are characterized as a slowly progressive condition marked by airflow limitation, being cigarette smoking the main etiologic factor [1]. This pathology presents a musculoskeletal atrophy [2], [3]. In order to carve these after effects a Pulmonary Rehabilitation procedure is a viable treatment for patients. Pulmonary Rehabilitation procedures consist of controlled walking activities and physical exercises that patients perform under the supervision of a physical therapist. However, COPD patients present a severe low oxygen saturation illness and they require oxygen supply, particularly when performing physical activities [4]. Hence, they are required to carry an oxygen tank for the oxygenotherapy assistance, but their own condition hinders their ability to precisely carry the often bulky external tank. This situation entails to find a pragmatic solution to avoid an additional physical therapist to carry the oxygen tank. The scenario may be aggravated by the fact that this procedure is performed on a rehabilitation gym that could be potentially crowded with several patients, physiologists and physical therapists.

An alternative solution is to use an assitive ground service robot [5] to carry the oxygen tank while following the patient in a leader-follower configuration. There are two reasons that support the initial viability of this idea. First, the rehabilitation gym is a constrained environment where this problem can be tackled by an Unmanned Ground Vehicle (UGV). On the other hand, the range of movements performed by the patient during the Rehabilitation Procedure is highly predictable by the treatment. At the same time, the global robotic research community looks forward for the development of robotic affordable solutions to social and health-related worldwide problems [6].

For the implementation of the leader-follower strategy, several solutions have been proposed, including Simultaneos Localization And Mapping (SLAM) alternatives, vision-based systems or based on electromagnetical beacons [7]. The work presented by [8] explores a differential tethered robotic system to perform camera-based gait analysis of the leader. For COPD Pulmonary Rehabilitaiton procedures, the patient is already umbilicaly linked to the oxygen tank via the breathing cannula. Hence, a robotic solution can exploit this circumstance to perform the *following* mechanism based on a tethered controller. Tethered robots have been extensively researched in robotics [9]. They offer a very simple solution to some common navigation problems, and they can be very effective in robot-to robot interaction, collaborative robotics, or while interacting with humans in Human Robot Interfaces [10], [11], [12].

At the same time, several assistance devices for COPD treatments have been proposed. Particularly relevant are novel telemedicine [13] applications to enhance complementary rehabiliation exercise at home that can track biological markers for patients [14], [3]. The work presented here follows the line established by [15]. Authors studied the use of a single thread tethered follower robot for home oxygen therapy, and compared two different control algorithms and their effectiveness to mimic the leader trajectory and to avoid obstacles. However, their approach focuses on the usage of the device exclusively for home therapy, and not within the context of a Pulmonary Rehabilitation procedure performed by medical personnel on medical facilities.

Hence, this work provides an initial study on the controlling mechanism of a differential tethered robot that implements the leader-follower configuration on a Pulmonary Rehabilitation procedure. To do so, this document unrolls as follows. Section II and III poses the problem and the solution design. Section IV documents the experimental protocol to perform the solution assessment on a simulation and on a real world scenario. Results and discussions are described in Sections V and VI. The clinical assessment performed jointly with medical personnel is tackled in Section VI-A. Finally, conclusions are exposed in the remaining Section VII.

E. A. Buniak, R. Ramele and J.M.Santos are with the CIC Laboratory at the Department of Computer Engineering, Instituto Tecnológico de Buenos Aires (ITBA), Ciudad de Buenos Aires, Argentina e-mail: rramele@itba.edu.ar

Manuscript received May 16, 2020; revised July 6, 2020.

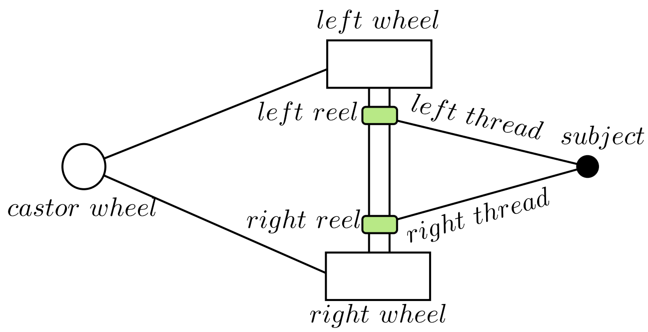

Fig. 1: Components of the robotic vehicle and the tether mechanism

II. Materials and Methods

To be effective, any technological solution for the medical community requires active involvement of key stakeholders: physicians, care-givers, patients and their families [16], [17], [18], [19], [20]. Hence, design methodologies that allow rapid prototyping can bring quick feedback from real users about drawbacks or opportunities for improvements.

Looking forward to achieve this goal, a basic robotic configuration is designed that allows the implementation of the tethered controlling mechanism, while keeping away other necessary features that will be the focus of future iterations. This design is first simulated in a simulation environment, and later, a basic hardware prototype based on Internet of Robotic Things [21] is built to verify the design guidelines and assumptions on a real world scenario.

A. Solution Design

The proposed solution is a Differential-Wheeled Robot (DWR) tethered to the followed subject with two threads ending at a single point attached to the subject waist, back or hand. In the same axis as the two front wheels, the robot has two reels separated by a certain length from which these threads come. As the subject moves away from the robot, the reels release thread so that the patient does not physically drag the vehicle. When the opposite happens, and the vehicle gets closer to the subject, an active spring mechanism driven by electric motors move each reel to retract the thread. The threads need to be tense at all times so that the encoders in each reel can be used to continuously measure the distance between the subject and the reel as devised in Figure 1.

Encoders in each reel measure the difference in length of each thread compared to its initial position. These differences in length are the input for the control algorithm. Using the encoder, the difference in length for each thread can be measured with Equations 1.

$$\begin{aligned} l_L &= pulses_l \, \tfrac{2\pi r}{ppr} \\ l_R &= pulses_r \, \tfrac{2\pi r}{ppr} \end{aligned} \tag{1}$$

where $pulses_l$ and $pulses_r$ are the pulses obtained for the left and right encoder. Pulses refer to natural discrete numbers that represents the circular movement of the encoder shaft. The variable r is the radius of the reel and ppr is the pulses per revolution (resolution) of the encoder. These equations provide the estimated values for the left l_L and right l_R, which are the length of thread released from each reel. The initial position of the threads is configurable.

B. Hardware

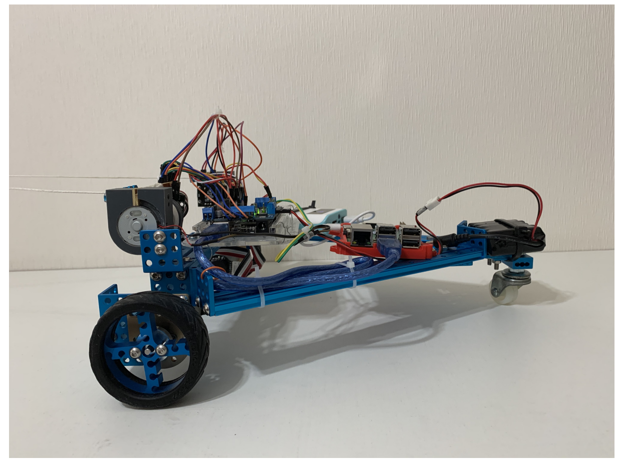

Fig. 2: Side-view of the robot prototype. A differential wheel, the reel and the rear free castor wheel can be observed from the picture.

Frames are constructed from aluminum extrusions produced by Makeblock (Shenzhen, China). The prototype is a three-wheeled robot with two frontal differential drive wheels and a free castor wheel as a third point of contact on the back, as can be seen on Figures 2 and 3.

Two motors Makeblock Optical Encoder Motor-25 9V/86 rpm are used on in-wheel configuration providing optical encoding. A microcontroller Arduino (Arduino LLC, Italy) Mega 2560 is used to implement the control loop and to provide encoder processing. On top of it an Adafruit (Adafruit, New York City) Motor Shield v2 bridge is used to drive the four DC motors, one for each wheel and one on each reel. The Arduino board is also connected to a Single Board Computer (SBC) Raspberry Pi (Raspberry Pi Foundation, United Kingdom) 3B+ through serial connection on one of the USB port.

The SBC connects to a WiFi network and can receive remote commands to control the robot. It also broadcasts telemetry data to any listening devices on the network. The control algorithms run in this board, which continuously communicates with the Arduino board to receive sensor data and to issue commands to move the motors or retract the reels. The algorithms are programmed in Python 3.7, which allows the exact same code to run both the simulated and real world prototype.

The two reels are designed from PVC extrusions and are shown on Figure 3. They are attached to regular FA-12350 DC motors scavenged from old compact discs. Each reel is axially locked to inexpensive Ky-040 rotary encoder [22] which provides around 20 pulses per revolution.

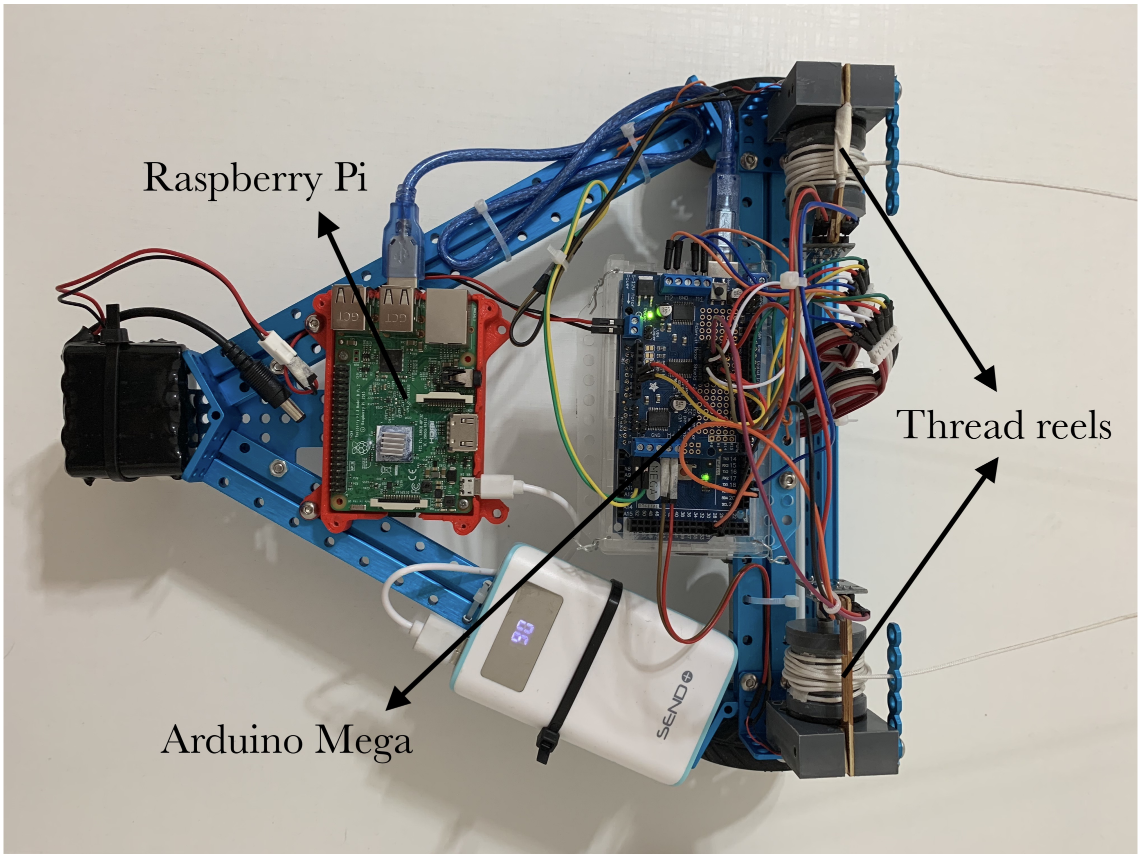

Fig. 3: Top view of the robot prototype. The SBC is shown on the center, alongside the Arduino Mega board. Both reels can be seen on the same vertical plane of the wheel axis. A power bank (white) is located on one side of the robot, and on the rear part, the motor power battery (black) is located.

The prototype can be seen on Figure 2 and 3. It has two separate batteries, one powering the Raspberry Pi, the Arduino board and the encoder electronics, and the other powering the drive and reel motors. The first battery is a commercial power bank with a capacity of 10000 mA·h, and an output of 3.1A (over two USB ports) at 5V. The motor battery is a set of 10 AAA nickel–metal hydride batteries (1.2V each), with a total output of 12V.

C. Active Reel Spring

As previously mentioned, an active spring mechanism is also put in place to keep the threads tense. However, in order to extend the useful life of the reel motors, and to save battery, an algorithm to activate and deactivate the motors was developed.

The algorithm works as follows:

1) While wheels are moving, retract reels.
2) If wheels stop moving, wait for *reel wait time* seconds, then retract reels.
3) Retract reels until the reel encoders values have not changed during *reel retract time* seconds.
4) If wheels started moving or the encoder values have changed while retracting, start the *reel retract time* countdown again.

D. Control Strategy

Two simple algorithmic control strategies are proposed and evaluated. The first one is called *Follow the thread* and the second strategy is *Rotate and go*.

1) Follow the Thread: This control strategy is similar to the one presented in [8]. It is based on the idea that both the relative angle between the subject and the vehicle orientation and, the relative distance between the robot and the subject, can be computed from the length of the left and right threads. They are described by Equations 2, 3 and 4.

$$V_t = c_v(\frac{l_L + l_R}{2} - l_D) \quad (2)$$

$$\omega_L = V_t + c_\alpha(l_L - l_R) \quad (3)$$

$$\omega_R = V_t - c_\alpha(l_L - l_R) \quad (4)$$

where c_v and c_α are constant coefficients used for calibration whose units are expressed in $[\frac{1}{s}]$, and l_D is a constant offset in $[m]$ that is used to customize the desired length of the thread where the robot does not move. As shown on Figure 4, l_L and l_R are changes in the length of thread that was released from each reel in $[m]$ obtained from encoder information from Equation 1, V_t is the estimated forward velocity for the target

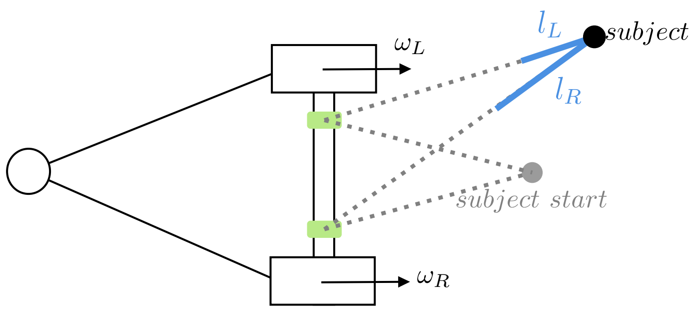

Fig. 4: Following mechanism parameters

and finally ω_L are ω_R are the velocity values that are directly used to drive each wheel motor.

To stop the vehicle completely when it is close to its expected position, an additional condition is added:

$$\textit{if } \frac{l_L + l_R}{2} < l_D \textit{ then } \omega_L = \omega_R = 0. \tag{5}$$

2) Rotate and Go: The *Rotate and go* algorithm divides the vehicle movement in two steps:

- Rotating the vehicle around the center point of the axis that connects its front wheels in order to aim at the subject.
- Go forward in a straight line until the vehicle is at the expected distance to the subject.

The procedure is detailed in Algorithm 1. The variable V_r is the speed at which the vehicle will rotate on its axis, and V_f is the speed at which the vehicle will move forward once it can move on the subject's direction. The algorithm requires three parameters, c_v and c_r that regulates the coefficients for the forward and rotation movement, and an additional parameter Dt_{off} that regulates the sensibility of the rotation movement. The constant parameter Dm_{off} is similar to l_D, and is used to customize the length of the thread where the robot does not move at all. Temporary variables Dt and Dm are used in the algorithm to calculate the difference and the average of the changes in released thread. Lastly, $base_{vr}$, is another constant used to determine the minimum rotational velocity of the vehicle.

Define $Dt \leftarrow l_L - l_R$
Define $Dm \leftarrow \frac{l_R + l_L}{2}$
$V_r \leftarrow c_r * (abs(Dt) - Dt_{off}) + base_{vr}$
$V_f \leftarrow c_v * (Dm - Dm_{off})$
if $abs(Dt) > Dt_{off}$ **then**
  if $l_L > l_R$ **then**
    $\omega_L \leftarrow V_r$
    $\omega_R \leftarrow -V_r$
  else
    $\omega_L \leftarrow -V_r$
    $\omega_R \leftarrow V_r$
  end if
else
  if $D_m > Dm_{off}$ **then**
    $\omega_R \leftarrow V_f$
    $\omega_L \leftarrow V_f$
  else
    $\omega_R \leftarrow 0$
    $\omega_L \leftarrow 0$
  end if
end if
Return ω_R and ω_L.

Algorithm 1: Rotate and go algorithm

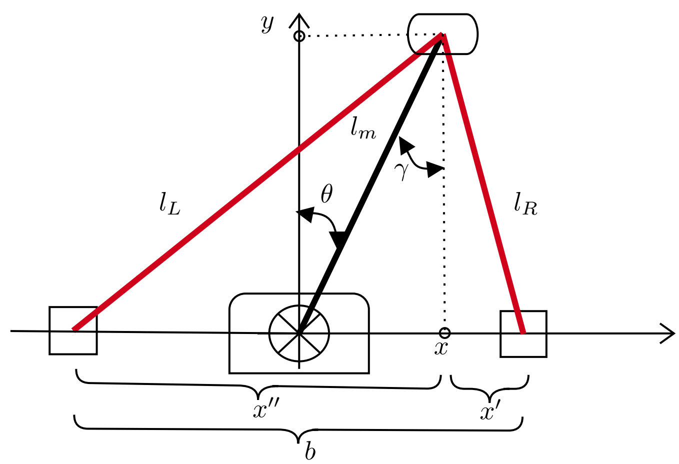

Fig. 5: The center of coordinates is the midpoint of the wheel axle, where the y positive axis points in the same direction as the robot direction.

III. Comparison with Alternative Methods

A. Equivalence between the single and double tethered system

In [15] authors propose a similar design based on only one thread. They propose two control strategies based on two input parameters that are obtained from a linear and circular potentiometer that determines the length of the thread l_m and the orientation angle θ. We show here that they are equivalent to the approach presented based on l_L and l_R.

Based on a frame reference with the robot on the center of coordinates as shown on Figure 5, from the input values l_m and θ we can derive the position of the leader as

$$\begin{aligned} y &= l_m \cos(\theta) \\ x &= l_m \sin(\theta) \end{aligned}$$

where x is the follower distance on the horizontal direction, and y the follower distance on the vertical direction.

In this frame of reference, the lengths of the threads, l_R and l_L can be determined as

$$\begin{aligned} l_R &= \sqrt[2]{y^2 + x'^2} \\ l_L &= \sqrt[2]{y^2 + x''^2} \end{aligned} \tag{6}$$

where x' and x'' are the distances from the leader horizontal position to the right and left wheel respectively.

From Figure 5 we can see that $x'' + x' = b$, with b the axle length. Combining this with Equation 6, we can form a system of equations to determine x', x'' and y:

$$\begin{aligned} x'' + x' &= b && \text{(a)} \\ {x'}^2 + y^2 &= {l_R}^2 && \text{(b)} \\ {x''}^2 + y^2 &= {l_L}^2 && \text{(c).} \end{aligned}$$

From the first Equation (a), rearranging and squaring both terms leads to

$$\begin{aligned} {x'}^2 &= (b - x'')^2 \\ &= b^2 - 2bx'' + {x''}^2. \end{aligned} \tag{7}$$

By subtracting (b) and (c), it can be obtained

$${x'}^2 - {x''}^2 = {l_R}^2 - {l_L}^2$$

and replacing ${x'}^2$ from Equation 7 in it

$$b^2 - 2bx'' = {l_R}^2 - {l_L}^2.$$

From this equation, x'' can be determined as

$$x'' = \frac{({l_R}^2 - {l_L}^2 - b^2)}{-2\ b}.$$

From Figure 5 it can also be seen that $\frac{b}{2}$ is equals to $x'' - x$. Hence this can be used to finally determine x and y values based on the threads length l_L and l_R and the axle length b. The y value comes from Equation 6.

$$\begin{aligned} y &= \sqrt[2]{{l_L}^2 - {x''}^2} \\ x &= x'' - \frac{b}{2} \end{aligned}$$

Finally, from trigonometry, it can be seen that

$$\sin\gamma = \frac{x}{l_m}$$

and

$$\cos\gamma = \frac{y}{l_m}$$

and as $\theta = \gamma$, we can obtain the values of θ and l_m:

$$\begin{aligned} \theta &= \arctan\frac{x}{y} \\ l_m &= \frac{y}{\cos\theta}. \end{aligned}$$

There is no loss of information and both systems are equivalent.

B. Design comparison

In the scheme proposed by [15] a single thread is recovered mechanically by means of a circular flat spring. This device works intensively when the robot is following the patient and, therefore, the spring is exposed to wear. Additionally with a circular flat spring, the spring tension depends on the released thread length, hence the recovery force will vary accordingly. Instead, the double thread tethered design implemented with an active reel spring allows a more controlled situation and depends exclusively on the software that controls the reel motor, which can be regulated.

Regarding the control algorithm, authors in [15] introduced two methods. The first of them computes the angular velocity as a function of the difference between the measured thread length l_m and the desired distance to patient l_D. In this way, if the patient moves around the robot with a l_m equals to l_D, the robot will not adjust its direction until the patient stops turning and restarts moving forward. Hence, the robot must correct its direction but the correction angle could be large, which leads the robot to deviate off the desired trajectory. Additionally, the second method proposed is based on dead-reckoning to estimate the patient position. This strategy requires very precise measurements to reduce the cumulative error, which is characteristic of this estimation procedure [23].

IV. Experimental Protocol

This section describes the experimental protocol used to evaluate the performance of the proposed solution. The Pulmonary Rehabilitation procedure consists on a series of walking activities aimed to promote patient muscular recovery and well being [3]. They are slow pace motions following a specific trajectory on a rehabilitation gym.

In order to standardize the procedure [24], the *Lemniscate of Gerono* is used as desired trajectory, a curve shaped like an ∞ symbol, described by the Equations 8:

$$\begin{aligned} &x(\phi) = a\ cos(\phi) \\ &y(\phi) = a\ cos(\phi)\ sin(\phi) \\ &where\ \phi \in \{-\pi, \pi\} \end{aligned} \tag{8}$$

where ϕ is the free parameter, a is the limit length of the arc of the curve, and x and y are the parametric functions that determine the shape of the trajectory on the navigation plane.

The reason this shape was chosen is because it combines different kinds of trajectories where the vehicle can be tested: long straight segments, sharp and soft curves, all in one single shape. Similar curves are also used in other proposed experiments in [5], [15].

Regarding metrics, four are proposed to evaluate the performance. They are:

- *Normal trajectory deviation, n.t.d.*: the subject trajectory is divided into small segments and then the normal distance to the robot trajectory is calculated for each of those segments. Trajectory deviation curve is relevant to evaluate how closely the robot mimics the leader trajectory, which is the ultimate goal of the robotic vehicle.

- *Robot-leader distance, r.l.d.*: The euclidean distance between the robot and the leader, at any point in time. This curve is particularly important since the robot has a limited amount of thread available, so if the leader uses all the available thread, it will start dragging the robot and damaging the following mechanism, and overall it may rise the possibility of disconnecting the breathing oxygen cannula. This is a scenario that must not happen under any circumstance, as it can also be dangerous for a potential patient using the device.
- *Total trajectory deviation, t.t.d.*: The area under the curve resulting from the the *normal trajectory deviation* over the length traveled by the leader.
- *Maximum trajectory deviation, m.t.d.*: the maximum *normal trajectory deviation* registered during an experiment.

In this work, a *following behavior* is considered satisfactory if its maximum trajectory deviation is less than 0.75 m and the robot-leader distance never exceeds 1.5 m [25].

First the simulation is described and later the evaluation on the robotic prototype is detailed.

A. Simulation

A model of the proposed design was first built on Webots [26] simulator. The threading mechanism was implemented using virtual threads [10]. The leader traveled according to a predefined trajectory with constant velocity, following the lemniscate trajectory.

The simulation is also useful to study the effects of the different constants on the robot movement for the different strategies. The leader starts at the midpoint of the trajectory and completes a full circuit getting back to the initial position, while the robot follows its track. Four different configuration sets of c_v and c_α were tested for *Follow the thread*, while 6 different configurations were tested for *Rotate and go*.

B. Real world

A real world experiment was performed, pegging to the same conditions implemented on the simulation environment. A motion capture system is used to track the movement of a human leader along a predetermined trajectory. The motion tracking system consists of an array of 16 OptiTrack (NaturalPoint Inc, Oregon, US) Flex 3 cameras, which measure the position of reflective markers with an accuracy of ± 1 cm at sampling rate of 100 Hz. The calibration and data collection was made using the Motive motion capture software.

As shown on Figure 6, a tracking marker was placed on each side of the robot (on top of each thread reel). The human leader used his hand to grab the tip at which the two tethers were tied together. A third marker was placed in his hand, using a glove. The lemniscate of Gerono, used in the simulation, was marked on the floor, and the human leader tried to move his hand following this shape as close as possible, with stable speed. The shape was marked according to the shape described in Equation 2, using $a = 2$ m.

The three markers allowed to measure the trajectory of both the robot and the leader, and then obtain the same metrics calculated in the simulation. Four experiments were performed for each set of parameter configurations. In this case, only two set of configuration were tested for each strategy.

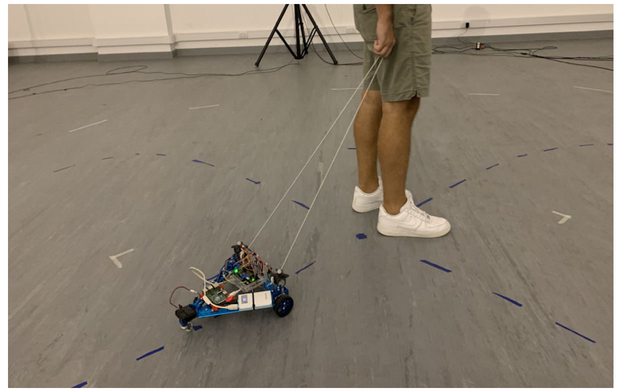

Fig. 6: Hardware prototype on the motion capture system and a testing subject holding the threads. On top of the device, two markers are placed and an additional marker is on a glove that the user is wearing (not shown on the picture). The lemniscate of Gerono was marked on the floor. The subject follows this track on the performed experiments.

V. Results

Simulation results for both control strategies are shown on Figure 7. Subfigures (a) and (b) expound the trajectories of the leader and the follower for each strategy, while (c) and (d) describe their speed profiles. Subfigure (e) show the distance between the robot and the patient for both strategies. Results metrics for the simulations are shown on Table I for the *Follow the thread* strategy, whereas metrics for *Rotate and go* are shown on Table II.

c_v	c_α	m.t.d.	t.t.d.
10	15	0.3614	2.0651
15	5	0.4325	2.055
15	**10**	**0.2188**	**1.0902**
15	15	0.2891	1.5059
5	20	0.5733	3.7289

TABLE I: Maximum trajectory deviation m.t.d. (m) and Total trajectory deviation t.t.d. for different *Follow the thread* constants.

c_v	c_r	Dt_{off}	m.t.d.	t.t.d.
10	20	0.1	0.4310	1.6380
20	20	0.05	0.7775	3.1139
20	**20**	**0.1**	**0.4123**	**0.9872**
20	35	0.1	0.4143	1.4820
20	5	0.05	0.7815	3.0892
35	20	0.1	0.6337	1.6190

TABLE II: Maximum trajectory deviation m.t.d. (m) and total trajectory deviation t.t.d. for different *Rotate and go* constants.

Results for the real world experiment can be seen on Figure 8. Table III shows the metrics of the *Follow the thread*

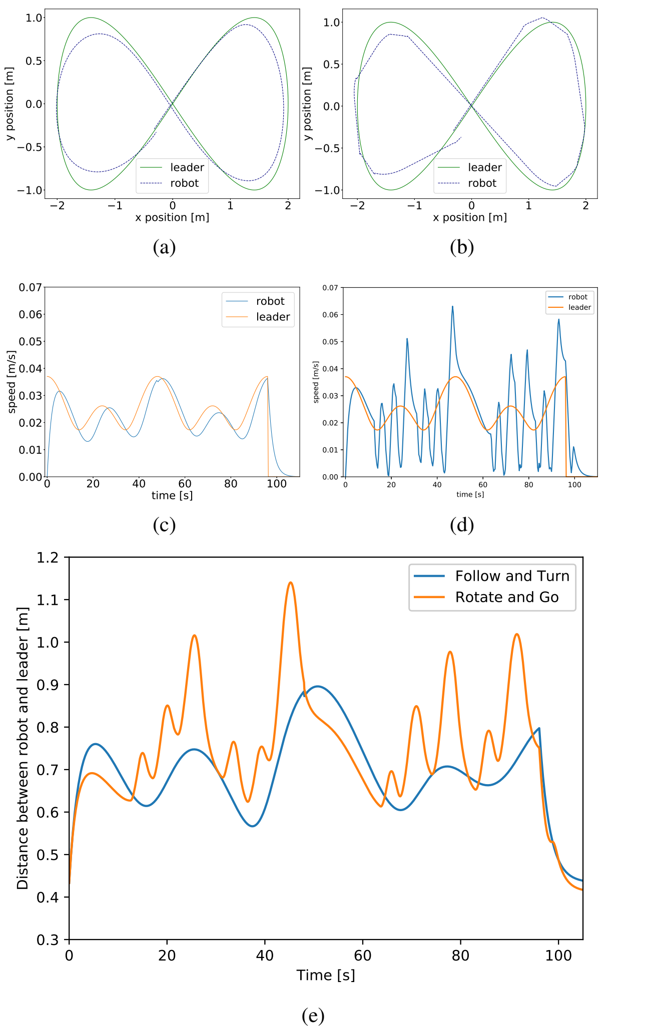

Fig. 7: Simulation Results: Trajectories of the leader and follower for *Follow the thread* (a) and *Rotate and go* (b) . Speed profiles of the leader and follower for *Follow the thread* (c) and *Rotate and go* (d). (e) Separation distance between robot and leader for both strategies (m).

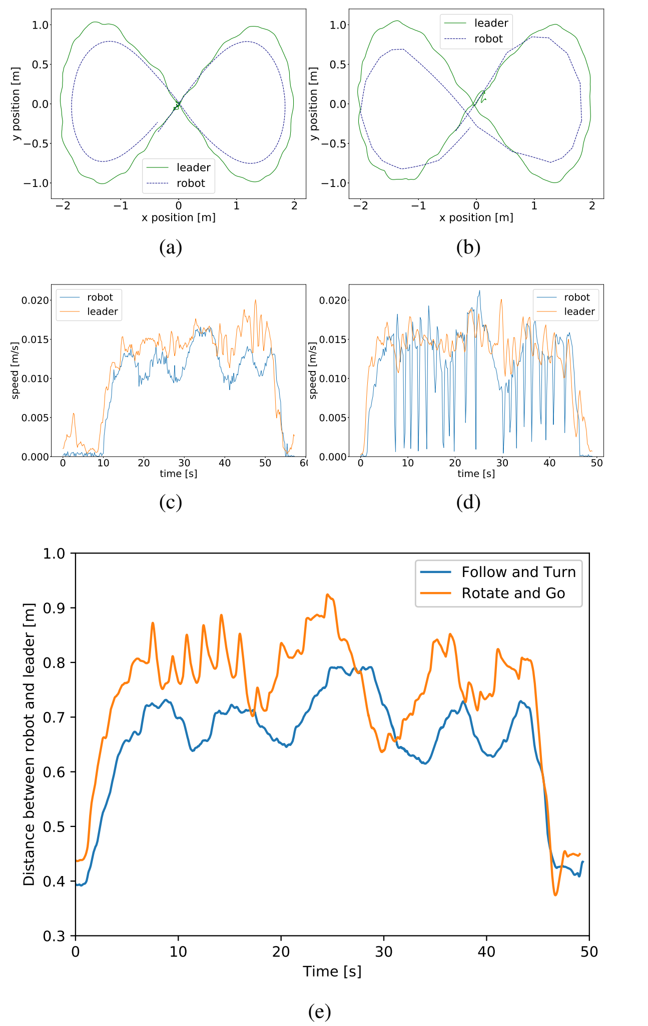

Fig. 8: Experimentation Results: Trajectories of the leader and follower for *Follow the thread* (a) and *Rotate and go* (b). Speed profiles of the leader and follower for *Follow the thread* (c) and *Rotate and go* (d). (e) Separation distance between robot and leader for both strategies (m).

strategy, whereas Table IV provides the metrics for the *Rotate and go* approach.

c_v	c_α	m.t.d. (m)	t.t.d.
25	**20**	**0.3876**	**1.9761**
25	35	0.4672	2.3528

TABLE III: Maximum trajectory deviation m.t.d. and area under normal trajectory deviation t.t.d. in motion capture experiments using *Follow the thread*.

c_v	c_r	Dt_{off}	m.t.d. (m)	t.t.d.
30	35	0.04	0.4116	2.8309
30	**35**	**0.08**	**0.3739**	**2.0367**

TABLE IV: Maximum trajectory deviation m.t.d. and area under normal trajectory deviation t.t.d. in motion capture experiments using *Rotate and go*.

A. Validation carrying the oxygen tank weight

We performed an additional simulated experiment to validate whether or not the selected algorithm could be extended to handle weight of the real oxygen tank. A common type of tank used in this type of therapies is a 5 kg oxygen tank.

We added on the Webots simulation a 5 kg mass on top of our model, and adjusted the coefficients c_v and c_α for the *Follow the thread* control strategy. Results are consistent with the outcomes shown previously on Figures 7 and 8. In Figure 9 it can be seen that the shape of the trajectory follows the patient smoothly, the speed profile also follows stringently the leader speed, and finally the distance between the patient and the robot is maintained within the safe boundaries. The obtained metrics are 0.2223 for m.t.d. and 1.1068 for t.t.d. which shows that the following behavior is achieved and

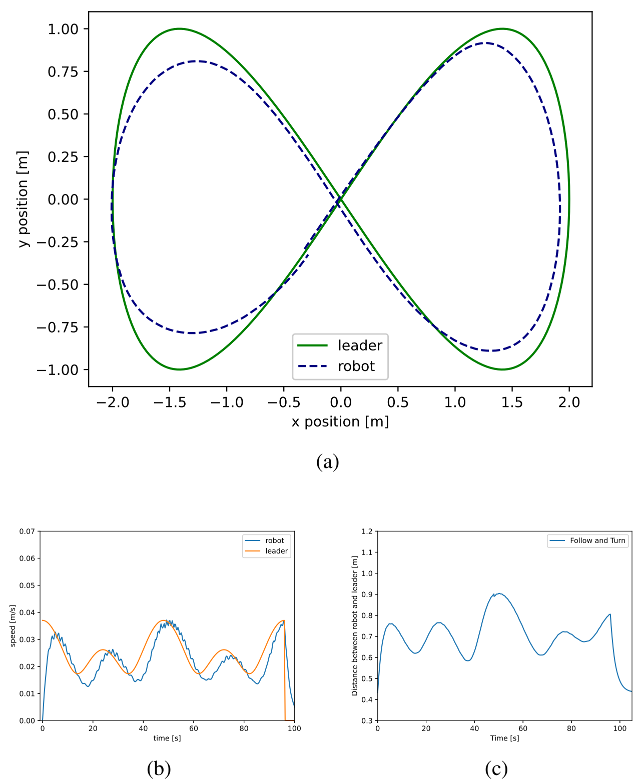

Fig. 9: Simulation results with the oxygen tank: the lemniscate trajectory is shown on (a) with the selected strategy *Follow the thread*. (b) Speed profiles of the patient-leader and the robot-follower. (c) Distance between the patient and the robot while traversing the trajectory.

verifies, from the simulation perspective, the initial feasibility of the control strategy considering the addition of the oxygen tank.

VI. Discussion

From the leader and robot trajectories in Figure 7(a to d), both strategies exhibits basic following behavior. However, *Rotate and go* on Figure 7(b,d) generates a more irregular trajectory, due to the two stage movement algorithm.

Regarding algorithms parameters, the simulation shows that for *Follow the thread*, a low c_α means that the robot is slow to turn and makes wider turns, providing a smoother trajectory. On the other hand, for low c_v values, the robot tends to lag behind the leader when it is going in a straight line. The highlighted values on Table II show the best configuration found.

For the *Rotate and go* configuration, the parameter c_r affects the dynamic behavior of the robot which needs to be adjusted accordingly. Lastly, increasing the Dt_{off} from 0.05 to 0.10 made the vehicle less prone to fall behind and produces a better following profile.

In line with the simulated results, a similar behavior was found on the experiments performed inside the Motion Capture Lab, and the robot exhibits following behavior for both control strategies as shown on Figure 8. Although the parameter values obtained from the simulation had to be readjusted for the real world scenario, the relative relation between them was maintained and that helped to narrow the parameter search space.

As expected, the *Rotate and go* strategy performs a *stop and go* movement, since the vehicle completely stops when rotating to face the leader. This is shown on the speed profiles in Figure 7(c,d) as well as on the real experiment on Figure 8(c,d). The smoother movement of the *Follow the thread* algorithm is an additional desired goal, since it can be perceived as less violent or unexpected.

Finally, regarding the Robot-leader distance r.l.d., the *Rotate and go* strategy results in a less stable distance (higher standard deviation), with higher maximum values, both on the simulated (Figure 7(e)) and on the real world scenario (Figure 8(e)). This can be specially problematic if we consider the cannula connecting the oxygen tank to the patient, as the cannula has a limited length.

A. Clinical Assessment

No amount of metrics are enough to evaluate if the robot is a viable solution for this problem or not, without the feedback and the evaluation of the people that are going to physically make use of it.

ALPI is a non-profit civil association located in Buenos Aires, Argentina, that provides neuromotor rehabilitation for pediatric and adult patients. It was founded in 1943 with the main focus of treating children with poliomyelitis, and has since expanded to deal with all kinds of related diseases.

Four professional care-givers from ALPI were invited to test and evaluate the controlling strategy on the prototype. A live demonstration of the robot working and following a moving person was performed.

In the demonstration, the robot design was outlined and an explanation was given on how the robot worked, how it was built and how to operate it. Both control strategies were explained, along with the main superficial differences between them.

Afterwards, health care professionals were invited to use the robot themselves, simulating they were the patient being followed. They were allowed to switch between the two control strategies to evaluate both of them, and made different tests, one walking along standardized trajectory marked on the floor and another one walking freely along the available space of the Motion Capture Lab. They used the robot freely to get a general idea of how it behaved, and how it could be used in the rehabilitation process. After all evaluations were finished, various aspects of the vehicle prototype were discussed and then a survey was handed out to document their experience with the robot, get their expert opinion on how the two strategies compared against each other, and what other improvements were needed in order to deliver a fully usable product. Survey questions and their averaged numerical evaluations are provided in Table V.

According to their answers, and the discussion we had after testing the robot, the general opinion was that the *Follow the*

Question	Avg. answer
How would you qualify, from 1 to 5, your overall experience with *Follow the thread*? (1:Bad, 5:Excellent)	5.0
How safe would a patient be, from 1 to 5, being followed by the robot using the *Follow the thread* strategy? (1:Very unsafe, 5:Very safe)	4.25
How would you qualify, from 1 to 5, your overall experience with *Rotate and go*? (1:Bad, 5:Excellent)	3.5
How safe would a patient be, from 1 to 5, being followed by the robot using the *Rotate and go* strategy? (1:Very unsafe, 5:Very safe)	3.25

TABLE V: Answers to survey questions.

thread strategy was safer and more convenient for the task. In the survey, when asked *Which of the two strategies is more effective at following the patient in a rehabilitation exercise?*, all 4 people responded that *Follow the thread* is ”much better”.

The main concern with the *Rotate and go* strategy was that having to wait for the robot to rotate before moving forward might be unsafe, as the patient could move away from it and compromise the cannula connecting him or her to the oxygen tank. This issue was identified during our own tests, and was not mentioned when explaining the following mechanism to the doctors, to avoid skewing them. They independently identified this problem, and emphasized that it could be a great source of discomfort for the patient.

Another aspect that was remarked from the *Follow the thread* strategy is that since it had a smoother movement, with no sudden stops or accelerations, it was favorable for the stability of the robot in order to carry the heavy oxygen tank.

Two needed security measures were also brought up by the ALPI team. Firstly, the need to add some mechanism for obstacle avoidance. They mentioned the need to have sensors to detect if the robot was about to hit something (specially the patient), and stop immediately, apart from what the control strategy indicated. Secondly, they recognized that some patients have very weak stability, and might fall down or take a step back, towards the robot, so it should be able to automatically move away from the patient, in order not to become another obstacle for him or her.

In order to have more information for the next steps in the development of the robot, we asked for their advise to design the mechanism to attach the threads to the patient being followed. Two ideas were proposed: a belt strapped to the patient waist, or a clasp tied to the clothes of the patient, also near its waistline. The waist is a good attachment point, since it is relatively more stable when the patient moves, compared to its hands or legs, that may make sudden movements and confuse the robot sensors.

VII. Conclusion and Future Work

From the practical experiments, it is verified that both algorithms ensuing the *following behavior* in the task of tracking the patient along a lemniscate-shaped trajectory. This following behavior is accomplished with a simple mechanism, a characteristic that significantly keep the price of the device low, putting it within reach of many medical institutions on developing nations.

Each control strategy has its advantages, but according to various metrics described in this work, the *Follow the thread* strategy had a more desirable behavior, as it tended to follow the leader from a closer distance at all times, while moving in a smooth and predictable way.

Insightful feedback is gathered from healthcare professionals from ALPI, who provided invaluable data to evaluate the solution. Over all, they highlight the *Follow the thread* strategy as being the safer and more effective one. Most importantly, they also validated the research and were enthusiast about the direction of the project. They proposed a series of improvements and next steps after seeing the prototype in action.

As described in the beginning, it is essential to involve stakeholders such as patients, doctors, nurses, and any other professionals involved in the rehabilitation process early in the design roadmap. They are the ones who understand the problem better than anyone else, and will be the end users of any developed product, as long as it is useful for them.

A. Future Work

The next steps for this project is to scale and iterate the design towards the desired solution, using the data obtained from this experiments and the feedback from care-givers.

- Redesign the active spring control mechanism in order to hold the motor temperature in its operational range.
- An easy and safe interaction between the patient, the operator and the robot. How to communicate the state of the robot to the operator, how to control and manipulate the robot in an effective and user-friendly way.
- Safety measures to keep the patient and the care-giver safe when using the robot. Not only safe from the robot movement, but also from its electronic components.
- An obstacle avoidance subsystem. This necessity is emphasized by the personnel from ALPI. The robot should have mechanisms in place to deal with emergency situations, and under no circumstance it can hit the patient or the doctor operating it.
- Achieve a battery autonomy that makes the robot useful throughout a complete pulmonary rehabilitation exercise. It is crucial for its usefulness to be able to hold a charge for this period of time, along with the ability to quickly swap batteries if the vehicle will be continually used with different patients.

VIII. Acknowledgments

This project is part of a joint collaboration between ALPI organization and the ITBA University. Authors would like to thank thoughtfully for the initiative and support given by ALPI, and to the Director of Rehabilitation Technology Dra. Mercedes Molinuevo. Additionally, authors would also like to offer tremendous gratitude to Natalia Nerina Meda, Soledad Suriá, Eduardo Etcheverry and Sergio Carlos Franco, for the idea of this project and for their invaluable help and enthusiasm to move this project forward.

Conflict of Interest Statement

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

[1] W. MacNee, "Pathogenesis of chronic obstructive pulmonary disease," *Proceedings of the American Thoracic Society*, vol. 2, no. 4, pp. 258–266, 2005.

[2] O. Kocsis, M. Vasilopoulou, A. Tsopanoglou, A. Papaioannou, and I. Vogiatzis, "Telemonitoring system for home rehabilitation of patients with COPD," in *2015 E-Health and Bioengineering Conference, EHB 2015*. IEEE, nov 2016, pp. 1–4.

[3] Ming-Feng Wu and Chih-Yu Wen, "A novel shuttle walking model using networked sensing and control for chronic obstructive pulmonary disease: A preliminary study," in *2012 6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops*, may 2012, pp. 147–150.

[4] B. R. Celli, "Pathophysiology of chronic obstructive pulmonary disease," in *Mechanics of Breathing: New Insights from New Technologies: Second Edition*. Milano: Springer Milan, 2014, vol. 9788847056473, pp. 339–352.

[5] A. F. Neto, A. Elias, C. Cifuentes, C. Rodriguez, T. Bastos, and R. Carelli, "Smart walkers: Advanced robotic human walking-aid systems," in *Springer Tracts in Advanced Robotics*. Springer, Cham, 2015, vol. 106, pp. 103–131.

[6] A. Khamis, H. Li, E. Prestes, and T. Haidegger, "AI: A Key Enabler for Sustainable Development Goals: Part 2 [Industry Activities]," *IEEE Robotics and Automation Magazine*, vol. 26, no. 4, pp. 122–127, dec 2019.

[7] M. J. Islam, J. Hong, and J. Sattar, "Person-following by autonomous robots: A categorical overview," *The International Journal of Robotics Research*, vol. 38, no. 14, pp. 1581–1618, 2019.

[8] A. Ortlieb, J. Olivier, M. Bouri, and H. Bleuler, "A robotic platform for lower limb optical motion tracking in open space," in *Mechanisms and Machine Science*. Springer, Cham, 2016, vol. 38, pp. 93–105.

[9] H.-S. Ahn, S.-I. Nah, Y.-C. Lee, and W. Yu, "A Controller Design of a Tethered-Robot Guiding System," pp. 43–46, 2006.

[10] I. Rekleitis, R. Sim, G. Dudek, and E. Milios, "Collaborative exploration for map construction," in *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation, CIRA*, vol. 2001-January. IEEE, 2001, pp. 296–301.

[11] Y. Hirata, Z. Wang, K. Fukaya, and K. Kosuge, "Transporting an object by a passive mobile robot with servo brakes in cooperation with a human," *Advanced Robotics*, vol. 23, no. 4, pp. 387–404, jan 2009.

[12] J. L. Ferrin, B. Thayn, and M. Hornberger, "Follower vehicle control system and method for forward and reverse convoy movement," *US Patent App. 12/238,733*, jan 2010.

[13] A. Banerjee, C. Chakraborty, A. Kumar, and D. Biswas, "Emerging trends in IoT and big data analytics for biomedical and health care technologies," *Handbook of Data Science Approaches for Biomedical Engineering*, pp. 121–152, jan 2020.

[14] G. Yang, C. Kong, and Q. Xu, "A home rehabilitation comprehensive care system for patients with COPD based on comprehensive care pathway," in *Proceedings - IEEE 4th International Conference on Big Data Computing Service and Applications, BigDataService 2018*. IEEE, mar 2018, pp. 161–168.

[15] G. Endo, B. Allan, Y. Iemura, E. F. Fukushima, M. Iribe, T. Takubo, and M. Ohira, "Mobile follower robot as an assistive device for home oxygen therapy - evaluation of tether control algorithms," *ROBOMECH Journal*, vol. 2, no. 1, p. 6, dec 2015.

[16] A. Gaggioli, A. Meneghini, F. Morganti, M. Alcaniz, and G. Riva, "A strategy for computer-assisted mental practice in stroke rehabilitation," *Neurorehabilitation and Neural Repair*, vol. 20, no. 4, pp. 503–507, dec 2006.

[17] J. Fasola and M. J. Matarić, "Using socially assistive human-robot interaction to motivate physical exercise for older adults," *Proceedings of the IEEE*, vol. 100, no. 8, pp. 2512–2526, aug 2012.

[18] A. Cherubini, G. Oriolo, F. MacRí, F. Aloise, F. Cincotti, and D. Mattia, "A multimode navigation system for an assistive robotics project," *Autonomous Robots*, vol. 25, no. 4, pp. 383–404, nov 2008.

[19] J. R. Wolpaw, "Brain-computer interfaces: progress, problems, and possibilities," in *IHI '12*, 2012.

[20] P. Salvini, "On ethical, legal and social issues of care robots," in *Springer Tracts in Advanced Robotics*. Springer, Cham, 2015, vol. 106, pp. 431–445.

[21] P. Simoens, M. Dragone, and A. Saffiotti, "The Internet of Robotic Things: A review of concept, added value and applications," *International Journal of Advanced Robotic Systems*, vol. 15, no. 1, p. 10, jan 2018.

[22] N. T. Sugahara Jun, Ono Koji, "Rotatively-operated electronic component with push switch and rotary encoder," jul 1998. [Online]. Available: https://patents.google.com/patent/US5847335A/en

[23] H. Durrant-Whyte, "Where am I? A tutorial on mobile vehicle localization," *Industrial Robot: An International Journal*, 1994.

[24] C. Sprunk, J. Röwekämper, G. Parent, L. Spinello, G. D. Tipaldi, W. Burgard, and M. Jalobeanu, "An experimental protocol for benchmarking robotic indoor navigation," in *Springer Tracts in Advanced Robotics*. Springer, Cham, 2016, vol. 109, pp. 487–504.

[25] N. D. Munoz Ceballos, J. Alejandro, and N. Londono, "Quantitative Performance Metrics for Mobile Robots Navigation," in *Mobile Robots Navigation*. InTech, mar 2010.

[26] O. Michel, "Cyberbotics Ltd. webots™: Professional mobile robot simulation," *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, pp. 39–42, mar 2004.