Single Landmark Feedback-based Time Optimal Navigation for a Differential Drive Robot

Vladimir Macias[a], Israel Becerra[a,c], Edgar Martinez[b], Rafael Murrieta-Cid[a,*], Hector M. Becerrra[a]

[a] *Centro de Investigación en Matemáticas, Guanajuato*
[b] *Tecnológico Nacional de México / ITS de Guanajuato, Guanajuato*
[c] *CONACYT Research Fellow*

Abstract

In this paper, we propose a feedback-based control approach to execute the time optimal motion trajectories for a differential drive robot. These trajectories are composed of straight lines and rotations in place. We show that the evolution of the position of a single landmark over time, in a local reference frame, makes it possible to track a prescribed time-optimal robot's trajectory, based on feedback of the landmark's position. We also show that the closed-loop system is an exponentially stable one with a nonvanishing perturbation, and that globally uniformly ultimately boundedness of the tracking errors can be achieved. The two main results of this work are: 1) Our approach leverages visual servo control type of methods with tools from optimal control for executing time-optimal trajectories in the state space based on feedback information. 2) The approach is able to work with the minimum number of landmarks–only one–this represents a *necessary and sufficient* condition for landmark-based navigation. Experiments in a physical robot, a nonholonomic differential drive system equipped with an omnidirectional laser sensor, are shown, which validate the proposed theoretical modelling.

Keywords: Time optimality, Feedback-based navigation, Visual Servo

*Corresponding author
Email addresses: maciasandoval@cimat.mx (Vladimir Macias), israelb@cimat.mx (Israel Becerra), emartinez@itesg.edu.mx (Edgar Martinez), murrieta@cimat.mx (Rafael Murrieta-Cid), hector.becerra@cimat.mx (Hector M. Becerrra)

Control, Optimal Control.

1. Introduction

This paper addresses the problem of controlling the trajectory of a differential drive robot (DDR) based on feedback. A DDR is a nonholonomic system [1], it has two wheels with two independent motors which allows the robot to rotate in place. A nonholonomic system cannot move instantaneously in all directions. The nonholonomic property of a system can be formally determined using the Frobenius Theorem [1]. The information used as feedback corresponds to the position of a single landmark in a local reference frame defined by the robot.

The considered trajectories are composed by straight lines segments, and rotations in place, which are the time optimal motion primitives for bounded velocity differential drive robots [2]. This paper is related to the works in [3, 4], in which feedback control laws for a DDR are presented. As a consequence of Brockett's condition [5] that states that no smooth feedback control can stabilize the complete state of the system, the authors of [3, 4] show that pure state feedback stabilization of the vehicle around a given terminal configuration is not possible. However, they show that trajectory tracking is possible, more specifically, the authors show that stabilization of the vehicle's configuration around a virtual vehicle used as reference is possible. Similar to the works in [3, 4], we also propose control laws for trajectory tracking, however, differently to the works in [3, 4], we do not use the configuration of a virtual vehicle as feedback information. Such a feedback scheme basically requires to know the DDR's position and orientation in a global reference frame, which is equivalent to solving a robot localization problem in a global reference frame. Instead, we directly use the location's coordinates, (x_L, y_L), of a single landmark measured in a local reference frame, and an estimation of the robot's orientation, θ, in a reference frame defined by the initial robot's pose. To estimate θ, a geometric method is proposed, based on the evolution of the landmark's position over the local reference frame.

This work is also related to the problem of robot's localization [6, 7]. It is well known that using a single landmark is not possible to totally determine the location and orientation of a robot in a two dimensional plane [7, 8]. However, in this work we show that the evolution of the landmark position over a local reference frame makes it possible to track a prescribed time-optimal robot's trajectory, based on feedback of the landmark's position. Indeed, we show that the accuracy of the robot's orientation provided by the proposed geometric method or observer, depends on the frequency at which the landmark's position over the local reference frame is available.

The present work is also related to landmark-based robot navigation [9, 10, 11] and particularly to controlling a mobile robot using visual servo control [12, 13, 14]. Indeed, in Section 7, a control law is proposed, which is similar to the one that one gets using visual servoing for trajectory tracking [15]; note that, as in [15, 16], it is common to rely on an eye-to-hand setting to perform such task. Our approach is more similar to an eye-in-hand setting, nonetheless, in this work we do not use a video camera and no perspective projection is involved; instead, we determine the pose of the landmark with respect to the robot using a laser range finder that directly provides the distance to the landmark and its direction with respect to the heading of the robot. Moreover, the usage of an on-board range finder allows to avoid camera-related complicating matters such as: estimation of calibration parameters and depth of visual features [17], computation of homographies [18, 21] or trifocal tensors [19], the need of supplementary signals provided by additional sensors such as inertial measurement units [20], etc. It is also worth to mention that we are especially interested in tracking time-optimal trajectories in the state space, which is not usually the case in visual servoing approaches.

Regarding the proposed control laws, the main results are the following. Our closed-loop model corresponds to an exponentially stable system with a nonvanishing perturbation. A main idea along this work is to use as much as possible information directly measured with the sensor on-board the robot as the feedback information to control the system, instead of estimating the com-

plete robot state in a global reference frame, which can be a complex task, in which is hard to rapidly obtain, robust and accurate estimations. Indeed, an important motivation of this work is not to reconstruct the whole state of the robot in a global reference frame, which can be a complex task, in which it is hard to rapidly obtain robust and accurate estimations, instead we propose to use local landmark coordinates (x, y) directly measured in the robots local reference frame with a laser range finder. However, our method needs the absolute θ orientation for the feedback controller. That orientation is estimated in a reference frame defined by the initial robot position and orientation using the geometric estimator presented in Section 6.1.

The ability of navigating based on feedback can be used in several practical applications, since it makes the approach more robust to uncertainty compared with an open loop scheme. For instance, a mobile robot that delivers packages or a mobile agent that carries a camera and sends video to a human user wearing a virtual reality headset for remote telepresence. An important advantage of the approach is that it does not require external sensors to estimate the robots pose, which implies to modify the environment and compromises the robot's autonomy. Furthermore, the capability to move traveling minimum time trajectories has advantages, such as potentially reducing battery or fuel consumption, performing the aforementioned packages delivery task quickly, or saving time to show larger areas to a human user in the telepresence task. Additionally, to the authors knowledge, the problem of leveraging feedback-based control with optimal control tools for executing time-optimal trajectories in the state space has not received enough attention.

The remainder of this paper is organized as follows. First, we present previous related work and the main contributions of this work. In Section 2, the time-optimal trajectories for the DDR, and the corresponding landmark motions in a local reference frame defined attached to the robot, are presented. The problem statement is provided in Section 3. In Section 4, the motion model for the DDR is presented, along with the kinematics of the landmark in the local reference frame. In Section 5, the nonlinear observability of the system

is discussed. In Section 6, a geometric method to estimate the robot's orientation θ is provided. In Section 7, the closed-loop control laws for the systems are obtained. Experiments in a physical robot are presented and discussed in Section 9. Finally, Section 10 concludes the paper and presents possible future work.

Related work

Our work is related to optimal control methods used in robotics, for instance [23, 24, 2, 25, 26, 27, 28], however, those methods typically execute the motion in open loop. Our work proposes a feedback-based method, which uses information directly from a laser range finder.

In [2], the time optimal trajectories for differential drive vehicles in the unobstructed plane are provided. The wheel angular velocities are bounded, but may be discontinuous. The authors prove the existence of optimal controls, derive the structure of optimal trajectories, and develop an algorithm for producing a time optimal trajectory between any two configurations. Every nontrivial optimal trajectory is composed of straight segments alternating with turns about the robots center. In [9], the authors have shown that the shortest distance path for a DDR, in the absence of obstacles, for landmark-based navigation with field-of-view constraints, are composed of three motion primitives: straight-line segments, rotations in place without translation and logarithmic spirals. In that same work, a characterization of the shortest paths for the system based on a partition of the plane into disjoint regions was also provided. This synthesis attempted to obtain the globally optimal paths in the absence of obstacles. Later, in [10], the authors have shown that the synthesis presented in [9] was incomplete, and in [10] the complete partition of the plane and the corresponding globally optimal paths in the absence of obstacles were provided. In [11], the authors have determined the necessary and sufficient conditions for the existence of a path such that the system is able to maintain one landmark visibility in the presence of obstacles, the authors have also extended this result to the problem of planning paths guaranteeing visibility among a set of landmarks,

e.g. to observe a given sequence of landmarks or to observe at each point of the path at least one element of the landmarks set.

In [3, 4], the authors show that pure state feedback stabilization of the vehicle around a given terminal configuration is not possible. However, feedback stabilization of the position of any vehicle's point remains possible. The problem of trajectory tracking in Cartesian space is also considered; the authors show that stabilization of the vehicle configuration around the configuration of a virtual reference vehicle becomes possible as long as the reference vehicle remains moving.

In [6], an efficient method for localizing a mobile robot in an environment with landmarks is presented. It is assumed that the robot can identify these landmarks and measure their bearings relative to each other. Given such a noisy input, the algorithm estimates the robots position and orientation with respect to the map of the environment. The algorithm makes efficient use of a representation of the landmarks by complex numbers. The algorithm runs in linear time with respect to the number of landmarks. In [7], the application of the extended Kalman filter to the problem of mobile robot navigation in a known environment is presented. Observability analysis using nonlinear theory has been done in [29], for the problem of mobile robot localization using landmarks.

In [30], a method is proposed to localize a robot using a single landmark, the robot is equipped with an omnidirectional camera, and the authors assume an orthographic projection model for the camera. Under this setting, the authors propose a landmark model, which is designed to have a three-dimensional, multicolored structure and the projective distortion of the structure encodes the distance and heading of the robot with respect to the landmark. Indeed, in that work the authors assume the landmark has orientation. For that reason, together with the orthographic projection assumption, it is possible for the authors to localize the robot using a single landmark. But this limits the application scenarios, since this represents more required information, namely, the landmark's orientation. In this work, the proposed method does not require the landmark to have an orientation. Besides, another difference with the cur-

rent work is that in [30] the authors do not propose controllers to execute the robot's trajectories based on the landmark location; in this work, we propose such controllers.

In [31], the authors pointed out the equivalence between localizing a non-holonomic DDR and the system observability. Similar to the work proposed in this paper, the authors of [31] propose a method to localize the robot based on a single landmark. However, in that work, the linear and angular velocities of the robot together with the robot's orientation θ are assumed to be measurable; indeed, the authors assume that θ is measured with a compass. In contrast, in this work we do not need to know or measure the robot velocity and orientation θ. Instead, we propose a method to estimate θ, nonetheless, the method assumes that the robot moves in an arc of circle of constant radius, which is a fair assumption for a time interval sufficiently small (see Section 6.1). Furthermore, in [31] the authors neither propose an approach to control the robot trajectories; in this work, we propose such an approach to move the DDR along the shortest time trajectories.

Recent state of the art approaches about robot localization use neural networks as a main tool, for instance [32] and [33]. The work in [32] proposes an approach for robot navigation and localization based on learning with variational neural networks. Regarding the navigation task, a variational neural network is used, raw camera images and an image of a noisy un-routed roadmap are the inputs. The output to be learned is a probability distribution over inverse curvature to navigate at each time instance. Regarding the localization part, the conditional structure of the network updates a posterior belief about the vehicles pose $P(\theta_p|I, M)$ based on the relation between the map and the road topology seen from the vehicle, where θ_p is the pose in the map (position and orientation), I is the visual input and M the map. In the work in [33], the authors propose a network architecture that employs a multitask learning approach to exploit the inter-task relationship between learning semantics, regressing 6-DoF global pose and odometry. The approach simultaneously embeds geometric and semantic knowledge of the world into the pose regression network. Nonethe-

less, as all data-driven approaches, those methods rely on the availability of a training data-set, which is not the case for our proposed method.

In [12], a visual servo tracking controller is developed for a monocular camera system mounted on an underactuated wheeled mobile robot subject to nonholonomic motion constraints. A prerecorded image sequence (e.g., a video) of three target points is used to define a desired trajectory for the robot. Some works, for instance [18], have followed a similar approach where a prerecorded image sequence is utilized to track a predefined trajectory. In our setting, since the time-optimal trajectory to achieve a configuration can be precomputed [2], the evolution of the landmark as seen by the range sensor can also be precomputed, thus, our approach does not need to actually move the robot and prerecord a measurements sequence for later use.

In [13], the authors present a visual servo controller that effects optimal paths in the sense of Euclidian distance for a nonholonomic differential drive robot with field-of-view constraints imposed by the vision system. The control scheme relies on the computation of homographies between current and goal images. The method does not use the homography to compute estimates of pose parameters, instead, the control laws are directly expressed in terms of individual entries in the homography matrix. Individual control laws for the three path classes that define the language of optimal paths: rotations in place, straight-line segments, and logarithmic spirals are developed. None of those works address the particular problem of tracking time-optimal trajectories; their optimality criteria is different from ours. In [14], the authors propose the use of dynamic pose estimation exploiting the 1-D trifocal tensor in the task of driving a mobile robot to a desired location specified by a target image. A position-based visual-servo control drives the robot to a desired position and orientation through smooth velocities by tracking particular parabolic trajectories.

1.1. Main contributions

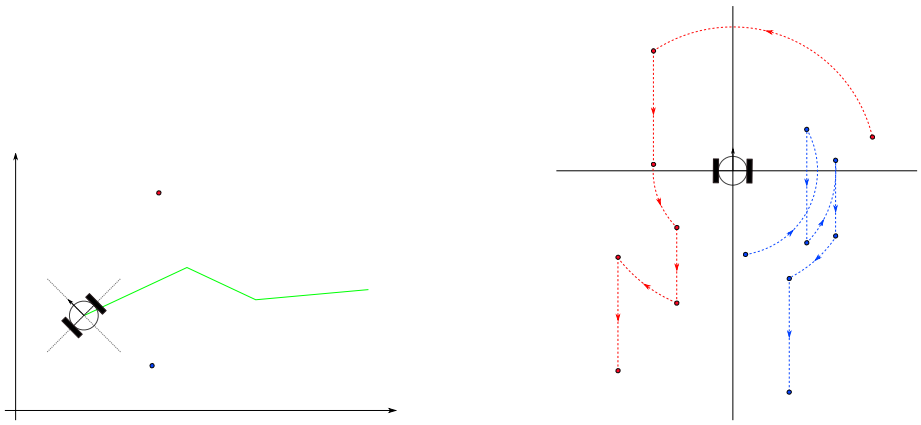
As it was previously mentioned, this work is related to several robotics problems, such as landmarks based navigation, robot localization, and feedback-

based control. In summary, in this broad context, the main contributions of this work are the following ones:

1. We propose a feedback control scheme that seeks to *execute time-optimal primitives in the state space*. The proposed approach is able to work with the minimum number of landmarks–only one, with no orientation–, which represents a *necessary and sufficient condition* for feedback based optimal navigation based on landmarks.
2. A geometric method for computing the robot's orientation θ is proposed, whose accuracy depends on the frequency in which the landmark position over the local reference frame is available.
3. We show that the control system is an exponentially stable system with a non-vanishing perturbation, and that globally uniformly ultimately boundedness of the tracking errors can be achieved.
4. Experiments in a physical robot show the validity of the theoretical analysis.

2. Time-optimal trajectories for the DDR and landmark motions in a local reference frame

The time-optimal trajectories for a DDR in the absence of obstacles [2], are composed by motion primitives that are either rotations in place (clockwise rotation $\curvearrowright$ or counter-clockwise rotations $\curvearrowleft$) or straight line motions (forward motion $\Uparrow$ or backward motion $\Downarrow$). In [2], those time-optimal motion primitives are concatenated and organized within nine possible symmetry classes of optimal trajectories (indexed as A, B, C, ..., H, I), yielding globally time-optimal trajectories between any two DDR poses in an obstacles-free environment. The trajectories are comprised of at most five motion primitives. The authors of [2] already provide an algorithm that determines the time-optimal trajectories between a given start and goal position. That algorithm will be used along the present work to resolve the trajectories to be tracked (we refer the reader to [2] for further details).



(a) Global reference frame.

(b) Local reference frame.

Figure 1: A local reference frame centered in the robot with the y-axis aligned with the robot's heading, is considered. Subfigure (a) shows a trajectory generated by a sequence of time optimal motion primitives. In that subfigure, two landmarks are shown. Their motion in the local reference frame is shown in Subfigure (b).

On the other hand, the sequence of time-optimal motion primitives not only defines a trajectory for the DDR, but also defines a path for the landmark in a local reference frame relative to the DDR, whose y-axis is aligned with the robot's heading. Since the time-optimal motion primitives are either rotations in place or motions in a straight line for the DDR, the paths generated for the landmarks are alternated concatenations of arcs of circle and straight line segments (see Figure 1). If the DDR rotates in place clockwise an amount of α degrees with respect to a global reference frame, then, in the local reference frame the landmark will move counterclockwise on an arc of circle, whose endpoints define an angle of α degrees. If the DDR moves forward in a straight line of length d, then, in the local reference frame, the landmark will move in a straight line of length d parallel to the heading of the DDR, and its y-coordinate will decrease.

3. Problem statement

Let $\mathbb{R}^2$ be the free obstacle environment with a reference frame assigned to it, and which is further referred as the *realistic space*. Such reference frame is assumed to be centered at the initial position of the robot and with the y-axis aligned with the initial robot's heading. The addressed robotic system is a differential drive robot (DDR) that moves within the realistic space, in which there is one landmark. The DDR is equipped with a laser that measures the relative distances from the center of the DDR to the landmark, as well as the orientation of the landmark with respect to the current heading of the DDR. It is also considered that using the methodology from [2], a time-optimal path in the realistic space–comprised of rotations in place and straight line motions, has already been computed and given to the robot. The goal is to control the DDR's execution of the rotations in place, and straight line motions that form the reference time-optimal trajectory, using as feedback information the relative location of the landmark provided by the laser.

We stress the fact that, for our addressed problem, we *neither want nor need to localize the robot in an absolute global reference frame.*

4. Motion model

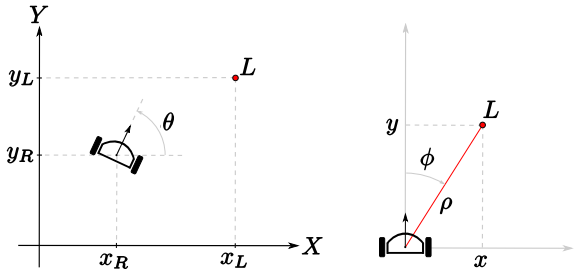
In this section we start by presenting the kinematic model of the robot, that is, a differential drive robot. Next, we proceed by presenting the kinematic model of the landmark with a local reference fixed to the DDR, which will be further referred to as the reduced space.

4.1. DDR kinematic model in the realistic space

The configuration of the DDR is denoted by $(x_R, y_R, \theta) \in \mathbb{R}^2 \times S^1$, where $(x_R, y_R) \in \mathbb{R}^2$ are the coordinates of the center of the DDR in the realistic space, and $\theta \in S^1$ is the orientation of its heading (see Figure 2(a)). The landmark L has coordinates $(x_L, y_L) \in \mathbb{R}^2$.

The kinematics of the DDR is given by the next system of equations

$$\begin{aligned} \dot{x}_R &= v\cos\theta, \\ \dot{y}_R &= v\sin\theta, \\ \dot{\theta} &= \omega. \end{aligned} \tag{1}$$



(a) DDR and landmark in the realistic space.

(b) Reduced Space.

Figure 2: Model of the considered differential drive robot, the robot's configuration and landmark's position in the reference frame of the realistic space, and reduced space.

4.2. Kinematic model of the landmark in the reduced space

To address the problem, we will consider a local reference frame fixed on the DDR, with origin on the center of the DDR and with the y-axis aligned with the current robot's heading. This local reference frame is known as *reduced space* [34]. Working in the reduced space has the advantage that the system can be represented as $\mathbf{x}(t) = (x, y) \in \mathbb{R}^2$.

Given a configuration for the DDR $q = (x_R, y_R; \theta) \in \mathbb{R}^2 \times S^1$, we can associate to it a transformation $T_q : \mathbb{R}^2 \times \{1\} \to \mathbb{R}^2 \times \{1\}$ defined by

$$T_q = \begin{pmatrix} \sin\theta & -\cos\theta & 0 \\ \cos\theta & \sin\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -x_R \\ 0 & 1 & -y_R \\ 0 & 0 & 1 \end{pmatrix}, \tag{2}$$

which is an isometric bijection. Note that $\mathbb{R}^2 \times \{1\}$ can be identified with the plane $\mathbb{R}^2$. So, T_q transforms the plane $\mathbb{R}^2$ into the plane $T_q(\mathbb{R}^2)$. In other words, if x_R is a point on the realistic space with coordinates $\mathbf{x}_R = (x_R, y_R)$, then

$T_q(x_R, y_R, 1)$ are the coordinates of $\mathbf{x}_R$ in the reduced space. Let $L = (x_L, y_L)$ be the landmark's coordinates in the realistic space, then its Cartesian coordinates in the reduced space are given by $T_q(L)$, this is

$$\begin{aligned} x &= (x_L - x_R)\sin(\theta) - (y_L - y_R)\cos(\theta), \\ y &= (x_L - x_R)\cos(\theta) + (y_L - y_R)\sin(\theta). \end{aligned} \tag{3}$$

If (ρ, ϕ) represents the polar coordinates of L in the reduced space, then we will consider the angle ϕ measured with respect to the heading of the DDR, in the positive sense if it is clockwise, and in the negative sense if it is counterclockwise. In this manner, the relation between the Cartesian coordinates and the polar coordinates of L is given by

$$\begin{aligned} x &= \rho\sin\phi \\ y &= \rho\cos\phi \end{aligned} \quad \Leftrightarrow \quad \begin{aligned} \rho &= \sqrt{x^2 + y^2}, \\ \phi &= \tan^{-1}(\tfrac{x}{y}). \end{aligned} \tag{4}$$

It is clear that, given a configuration $p = (x_R, y_R, \theta)$ for the DDR in the realistic space, the transformation T_p defined as in Eq. (3) is unique, besides, this transformation is bijective. So, if (x_L, y_L) are the coordinates of the landmark in the realistic space, then there is a unique pair of coordinates (x, y) in the reduced space, such that $T_p(x_L, y_L) = (x, y)$. However, the reciprocal is not true. Given a point (x, y) in the reduced space, there exist an infinite set $\{(x_R, y_R, \theta)_\alpha\}$ of configurations for the DDR, which defines a set of transformations $\{T_\alpha\}$ such that $T_\alpha(x_L, y_L) = (x, y)$ for all α.

Since the reduced space is a local reference frame fixed on the DDR, the motion of the DDR in the realistic space generates a relative motion of the landmark L in the reduced space. The kinematics of L in the reduced space depends completely on the kinematics of the DDR in the realistic space (refer to Eq. (1)). Substituting $\dot{x}_R$, $\dot{y}_R$ and $\dot{\theta}$ from Eq. (1) in the time-derivatives of x and y, we obtain the kinematic model of L in the reduced space given by the following equations

$$\begin{aligned} \dot{x} &= \omega y, \\ \dot{y} &= -\omega x - v. \end{aligned} \tag{5}$$

Appendix A presents in detail the derivation of Eq. (5).

The kinematics of the landmark L can also be expressed in polar coordinates as follows

$$\begin{aligned} \dot{\rho}(t) &= -v(t)\cos\phi(t), \\ \dot{\phi}(t) &= \omega(t) + \frac{v(t)\sin\phi(t)}{\rho(t)}. \end{aligned} \tag{6}$$

Appendix B presents in detail the derivation of Eq. (6).

5. Non-linear observability

The existence of an infinite set $\{(x_R, y_R, \theta)_\alpha\}$ of configurations for the DDR given a point (x, y) in the reduced space, suggests that the system may not be observable. Next, we introduce a series of concepts to present the definition of non-linear observability given in [35].

Considering a system Σ

$$\begin{aligned} \dot{\mathbf{x}} = &\quad f(\mathbf{x}, u), \\ \mathbf{y} = &\quad g(\mathbf{x}), \end{aligned} \tag{7}$$

where $u \in \Omega$, a subset of $\mathbb{R}^l$, $\mathbf{x} \in \mathbb{R}^m$, $\mathbf{y} \in \mathbb{R}^p$, f and g are C^∞ functions, and assume the trajectories of Σ to satisfy the initial condition $\mathbf{x}(t^0) = \mathbf{x}^0$. The input-output map of the pair $(\Sigma, \mathbf{x}^0)$, the indistinguishable property between states $\mathbf{x}^0$ and $\mathbf{x}^1$, and the observability of a system Σ, are defined as follows.

Definition 1. *Let $(u(t), [t^0, t^1])$ be the admissible input that gives rise to a solution $(\mathbf{x}(t), [t^0, t^1])$ of $\dot{\mathbf{x}} = f(\mathbf{x}, u(t))$ satisfying the initial condition. This, in turn, defines an output $(\mathbf{y}(t), [t^0, t^1])$ by $\mathbf{y}(t) = g(\mathbf{x}(t))$. Then, the* input-output map of Σ at $\mathbf{x}^0$ *is denoted by*

$$\Sigma_{\mathbf{x}^0} : (u(t), [t^0, t^1]) \mapsto (\mathbf{y}(t), [t^0, t^1]). \tag{8}$$

Definition 2. *A pair of states $\mathbf{x}^0$ and $\mathbf{x}^1$ are* indistinguishable*, denoted $\mathbf{x}^0 I \mathbf{x}^1$, if $(\Sigma, \mathbf{x}^0)$ and $(\Sigma, \mathbf{x}^1)$ realize the same input-output map, i.e., for every admissible input $(u(t), [t^0, t^1])$*

$$\Sigma_{\mathbf{x}^0} : (u(t), [t^0, t^1]) = \Sigma_{\mathbf{x}^1} : (u(t), [t^0, t^1]). \tag{9}$$

Definition 3. Σ *is said to* be observable at $\mathbf{x}^0$ *if* $I(\mathbf{x}^0) = \{\mathbf{x}^0\}$ *and* Σ *is* observable, *if* $I(\mathbf{x}) = \{\mathbf{x}\}$ *for every* $\mathbf{x} \in M$. $I(.)$ *returns the set of indistinguishable elements of the argument.*

Through Theorem 1, we show that the system studied in this paper is not observable in the sense of Definition 3 from [35]. See Appendix C for its proof.

Theorem 1. *The system* Σ*:*

$$\begin{aligned} \dot{x}_R &= v\cos\theta, \\ \dot{y}_R &= v\sin\theta, \\ \dot{\theta} &= \omega, \\ g(q) &= T_q(L), \end{aligned} \tag{10}$$

is not observable.

Remark 1. *The system* Σ *from Eq. (10) is not observable, however, using the methodology from Section 6.1, the true orientation angle* θ *of the robot measured in a global reference frame can be approximated, provided that the used reference frame is centered at the origin of the initial robot's position and with its* y*-axis aligned with the initial DDR's heading. Furthermore, the time optimal trajectories of the DDR can be executed based on feedback information corresponding to the landmark location in the reduced space (see Section 7).*

Remark 2. *Even though in the initial configuration, the reference frame is assumed to be centered at the origin of the initial robot's position and with its* y*-axis aligned with the initial DDR's heading, the method proposed in Section 6.1 is able to keep estimating the correct robot's orientation as time elapses and the robot moves changing its pose.*

Note that the fact that the system is not observable in continuous time does not prevent one to propose an estimator in the discrete case, see for instance [14].

Considering the tracking task given to the DDR, the optimal trajectories are defined in the realistic space and consist of rotations in place and straight

line motions. Concerning rotations in place, the angle of the landmark pose in polar coordinates in the reduced space suffices to perform that motion. In regard to straight line motions, recall that the motion of the robot is perceived by the on-board laser range finder as a motion of the landmark in the reduced space, in other words, the straight line trajectory in the realistic space maps to a trajectory of the landmark in the reduced space. However, there are infinitely many trajectories in the realistic space map to the same landmark trajectory in the reduced space, for instance, see the cyan trajectories in Figure 3. Thus, to resolve the correct trajectory that the DDR needs to track, the orientation of the robot in the realistic space is utilized. In that manner, by means of the position (x, y) of the landmark in the reduced space and the robots orientation θ in the realistic space, the DDR is able to identify and track the correct straight line reference. However, notice that θ is unknown, hence, in the next section, we provide a geometric methodology to estimate it.

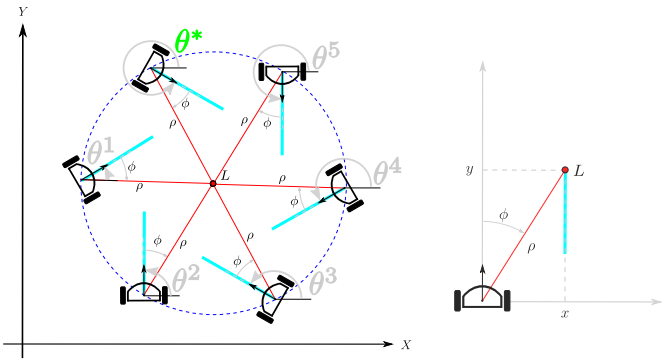


Figure 3: A series of straight line trajectories in the realistic space (cyan trajectories in the left figure) map to the same landmark trajectory in the reduced space (cyan trajectory in the right figure). Nonetheless, the trajectories in the realistic space can be uniquely identified through the robots orientation θ that is required to track each of them. The desired trajectory is the one labeled as θ^*.

6. A method to estimate the robot's orientation

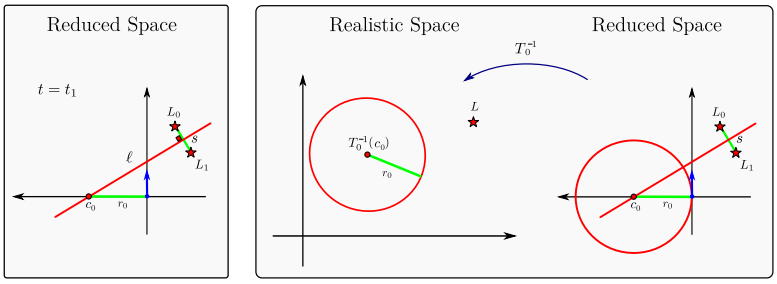
Below, in Section 7, we show that it is possible to execute the optimal motion primitives based on the location (x, y) of the landmark in the reduced space, plus the orientation θ of the DDR with respect to the initial robot pose, namely, in the realistic space. The landmark coordinates in the reduced space are directly measurable by the robot's sensor (a laser range finder). Unfortunately, this is not the case for the variable θ in the realistic space. As shown in Figure 3, there is an indistinguishable space (dotted blue circle) given a reading of the landmark location in the reduced space. The method is based on geometric reasoning and is capable of estimating the angle θ in the reference frame defined by the initial robot's position and orientation. To do that, we need to consider some properties of the transformation defined in Eq. (2). For that, we give the next definition.

Definition 4. *Let $\mathcal{C}$ be a circumference in the realistic space. We say that the configuration $q = (x_R, y_R, \theta)$ is* tangent *to $\mathcal{C}$ if the straight line determined by the point (x_R, y_R) and slope $\tan(\theta)$ is tangent to the circumference $\mathcal{C}$ at the point (x_R, y_R). We say that $\mathcal{C}$* is on the left (right) side *of the configuration q, if $\mathcal{C}$ is on the left (right) side of the DDR at the configuration q.*

6.1. Methodology to calculate θ

The methodology to estimate θ approximates the robot's motion between consecutive sensor's measurements, making use of the robot's movements that are either straight lines or arcs of circles. More precisely, based on two consecutive position readings of the landmark in the reduced space, the methodology fits either a straight line motion or an arc motion of the robot, such that the fit motion would generate the respective landmark positions in the reduced space. Then, the fit motion is then used to estimate the resulting robot's orientation. Below, the detailed methodology to compute θ is presented.

First of all, for $t = 0$, we have $q_0 = (x_{R0}, y_{R0}, \theta_0)$, the initial configuration of the DDR, which is assumed to coincide with the origin of the used reference



(a) Line segment bisector

(b) Circumference $\mathcal{C}_0$

Figure 4: Line segment bisector and circumference $\mathcal{C}_0$

frame and with the DDR's heading aligned with the positive y-axis, namely, $q_0 = (0, 0, \frac{\pi}{2})$. Its transformation is given as T_{q_0}, and $L_0 = (x_0, y_0) := T_{q_0}(L)$ as the landmark's pose in the reduced space. Analogously, for $t = t_1$, we have $L_1 = (x_1, y_1)$ as the landmark's pose in the reduced space. As it was mentioned before, there are infinitely many configurations $\{q_\alpha\}_{\alpha \in A}$ such that their associated transformations, $\{T_q\}_{\alpha \in A}$, satisfy $L_1 = T_{q_\alpha}(L) = T_{q_\beta}(L)\ \forall\ \alpha, \beta \in A$, with $\alpha \neq \beta$. Therefore, the correct configuration needs to be determined. To do that, we proceed to study the following two cases:

Case I: ***The DDR has moved on a straight line.***

In this case, at t_1 the configuration of the DDR is $q_1 = (x_{R1}, y_{R1}, \theta_1)$, where $\theta_1 = \theta_0$.

Case II: ***The DDR has moved on an arc of circle.***

To determine θ_1 for the DDR, we present the next methodology.

1. ***Determine the circumference in the realistic space on which the DDR moved.***

   Since the DDR is assumed to have moved on a circle arc, then the DDR is on a circumference $\mathcal{C}_0$ in the realistic space. $T_{q_0}(\mathcal{C}_0)$ is a circumference in the reduced space with center on the x-axis. Thus, L_0 and L_1 are on a circumference concentric to the circumference $T_{q_0}(\mathcal{C}_0)$. Let s be the line

segment in the reduced space whose end points are L_0 and L_1, and let ℓ be the line segment bisector of s, then L_0 and L_1 are on the circumference with center $c_0 = \ell \cap x$-axis (see Figure 4(a)). Since T_{q_0} is an isometric bijection, then the circumference $\mathcal{C}_0$ has its center on the point $T_{q_0}^{-1}(c_0)$ and radius $r_0 = \|c_0\|$. Therefore, the circumference $\mathcal{C}_0$ is determined (see Figure 4(b)).

2. ***Determine all possible poses for the DDR in the realistic space.*** As previously mentioned, $L_1 = (x_1, y_1)$ is the landmark in the reduced space at the time $t = t_1$, or in polar coordinates, $L_1 = (\rho_1, \phi_1)$. Consequently, in the realistic space the distance between the center of the DDR and the landmark L is ρ_1. Therefore, the pose of the DDR is on the circumference $\bar{\mathcal{C}}_1$ in the realistic space, which has center L and radius ρ_1 (see Figure 5). Since the DDR is on $\mathcal{C}_0$ but also on $\bar{\mathcal{C}}_1$, then $\mathcal{C}_0 \cap \bar{\mathcal{C}}_1 \neq \emptyset$. Notice that $\mathcal{C}_0 \cap \bar{\mathcal{C}}_1$ has one or two points. All the possible poses for the DDR are the points in the set $\mathcal{C}_0 \cap \bar{\mathcal{C}}_1$.

3. ***Determine the current configuration of the DDR in the realistic space.***
Recall that the DDR is assumed to have moved tangentially on the circumference $\mathcal{C}_0$. Let $\{p_a, p_b\} = \mathcal{C}_0 \cap \bar{\mathcal{C}}_1$ in the realistic space. The Cartesian and Polar coordinates of p_a are denoted as (x_a, y_a) and (ρ_a, θ_a), respectively. Also, let u_a be the unit vector tangent to $\mathcal{C}_0$ at the point p_a, and let v_a the vector from the point p_a to the landmark L. Then, the angle α_a between u_a and v_a can be calculated by

$$\alpha_a = \frac{\arccos(u_a \cdot v_a)}{\|v_a\|}.$$

Equivalent definitions are considered for p_b, α_b, u_b and v_b. Once angles α_a and α_b have been computed, they need to be compared with the angle ϕ_1 from landmark $L_1 = (\rho_1, \phi_1)$ in the reduced space, see Figure 5. If $\alpha_a = \phi_1$, then $\theta_1 = \theta_a + \frac{\pi}{2}$ and $q_1 = (x_a, y_a, \theta_1)$, and if $\alpha_b = \phi_1$, then

$\theta_1 = \theta_b + \frac{\pi}{2}$ and $q_1 = (x_b, y_b, \theta_1)$. Finally, compute T_{q_1} accordingly.

4. ***Repeat the process.***

   These steps are repeated to compute θ_2, θ_3, etc., in which the computation of θ_i makes use of the consecutive position readings of the landmark in the reduced space, L_i and L_{i-1}, and the estimation of $T_{q_{i-1}}$.

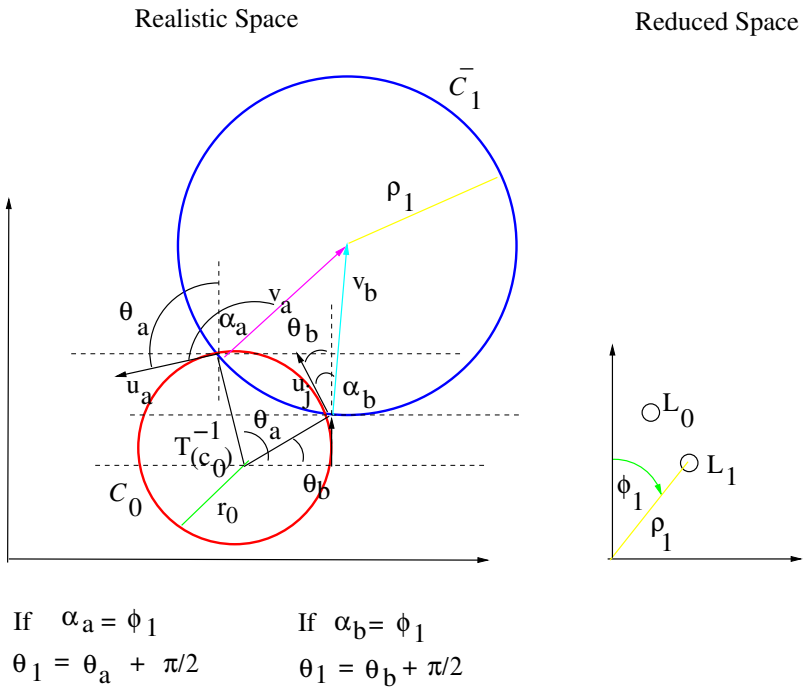


Figure 5: Circumference $\mathcal{C}_0$ shown in red and circumference $\bar{\mathcal{C}}_1$ is shown in blue.

Remark 3. *This procedure can be interpreted as determining a relative orientation with respect to the initial robot's configuration. This is enough to solve the stated problem. That is, the goal is to control the execution of rotations in place and straight line motions using as feedback information the location of a single landmark, such that the resulting trajectories track the time-optimal reference trajectories.*

Recall that it is assumed that the sensor that returns the landmark position in the reduced space, gets a measurement each Δ_t units of time. Let n be the

sampling rate of the sensor, and let N be an upper bound to it. Theorem 2 presents a result determining the accuracy of the methodology to calculate θ. To obtain such result, it is assumed that for a sufficiently small Δ_t, the DDR motion from a configuration q_{i-1} to a configuration q_i, can be described by an arc of a circle, which is a reasonable assumption as proved in Proposition 1.

Proposition 1. *Assume that the DDR executes a trajectory during the time interval $[0, \tau]$, and that the angular speed of the DDR is a Lipschitz function. Let $\Delta_t = \frac{\tau}{n}$ and $t_i = t_{i-1} + \Delta_t$ for $i = 1, ..., n$. If $n \to \infty$, then the DDR motion tends to be an arc of circle in the interval $[t_i, t_i + \Delta_t]$ for $i = 1, ..., n$.*

Proof. Since ω is a Lipschitz function, there exists $L_\omega > 0$ such that

$$|\omega(t + \Delta_t) - \omega(t)| \leq L_\omega |(t + \Delta_t) - t| = L_\omega \Delta_t.$$

If $n \to \infty$, then $\Delta_t \to 0$. Thus, $|\omega(t + \Delta_t) - \omega(t)| \to 0$, namely, the angular speed tends to remain constant. Therefore, the DDR will move on an arc of a circle during the time interval $[t_i, t_i + \Delta_t]$ for $i = 1, ..., n$. □

Theorem 2. *Let the robot's linear and angular velocities, v and ω, be Lipschitz functions. There exists an N such that if $n \geq N$, then the error of the methodology to estimate the robot's orientation θ in the realistic space, is equal to zero.*

Proof. Let $q = (x_R, y_R, \theta)$ be an initial configuration for the DDR, and let $(u(t), [\sigma, \tau])$ be an admissible input. Since v and ω are Lipschitz functions, the admissible input $(u(t), [\sigma, \tau])$ generates a smooth trajectory $\mathcal{T} = \{(x_R(t), y_R(t))\}_{t \in [\sigma, \tau]}$ in the plane where the robot is moving. For each $t \in [\sigma, \tau]$ we can get the osculating circle $\mathcal{C}_t$ at the point $(x_R(t), y_R(t))$. The circle $\mathcal{C}_t$ has the property that it matches the curve in a neighborhood V_t of the point $(x_R(t), y_R(t))$. Since x_R and y_R are continuous, there exist t_r such that $(x_R(s), y_R(s)) \in V_t$ for all $s \in [t - t_r, t + t_r]$. Since the robot trajectory $\mathcal{T}$ is compact, we can define $t_{min} = \min\{t_r : t \in [\sigma, \tau]\}$. For the Archimedean property[1], there exist $N \in \mathbb{N}$

[1]In general, the Archimedean property applies to a given quantity if for any two values A

such that $\frac{|\tau-\sigma|}{N} < t_{min}$. Let $\Delta_t = \frac{|\tau-\sigma|}{N}$. Thus, we have the partition of the interval

$$[\sigma, \tau] = [t_0 = \sigma, t_1] \cup [t_1, t_2] \cup \cdots \cup [t_N, t_{N+1} = \tau].$$

In this way, the trajectory $\mathcal{T}$ can be built with N arcs of circle, one for each subinterval $[t_i, t_{i+1}]$. Since, by construction the geometric estimator methodology presented in Section 6.1 used to estimate θ, moves the DDR on arcs of circle, hence it is possible to adjust the robot trajectory using a concatenation of arcs of circle. Therefore, if $n \geq N$, then each element that constitutes the robot trajectory is exactly on a sector of an arc of circle and consequently the error of the methodology is equal to zero. □

If $n < N$, or in the case that the robot's trajectory is not comprised of arcs of circle and line segments, then an error will result in the estimation of θ. Nonetheless, according to Proposition 1, the latter case can be alleviated by increasing n, yielding a decrease on the error of the estimation of θ. Section 8.1 presents simulations that provide insight on how the accuracy of the estimation of θ is affected when $n < N$, thus, the DDR has not moved following an arc between samples of the geometric estimator.

In the next section, we propose control laws for both cases when there is no error on the estimation of the robot orientation θ, and when this error exists.

7. Control laws

In this section, the proposed control laws are presented. We start by elaborating on the case of executing straight line motions. Subsequently, the case corresponding to the execution of rotations in place is presented.

7.1. Straight line motion primitive

In this case, the angle θ is needed to correct the robot motion. Let us assume that we want the DDR to move forward in a straight line to travel a distance d in

and B of this quantity such that $A < B$ it is always possible to find an integer m such that $Am > B$.

the time interval $I_\tau = [0, \tau]$. First, we assume that the condition of Theorem 2 is fulfilled, namely, that the error of the estimation of θ is zero; in this case our result regarding the proposed controller is Theorem 3. Later, we assume that there is a remaining error in the estimation of θ; in this second case our result regarding the proposed controller is Theorem 4.

Let $x(0) := x_0$, $y(0) := y_0$ and $y(\tau) := y_\tau$. Then, we must keep $x(t) = x_0$ for all $t \in I$, while y changes from y_0 to $y_\tau := y_0 - d$.

In this way, the desired values for x, y and θ are

$$\begin{aligned} x^*(t) &= x_0, \\ y^*(t) &= \begin{cases} \frac{d}{2}[1 + \cos(\frac{\pi t}{\tau})] + y_\tau & t \leq \tau, \\ y_\tau & t > \tau, \end{cases} \\ \theta^*(t) &= \theta_0. \end{aligned} \tag{11}$$

Therefore, the tracking errors in each coordinate are given by

$$\begin{aligned} e_x(t) &= x(t) - x_0, \\ e_y(t) &= y(t) - y^*(t), \\ e_\theta(t) &= \theta(t) - \theta_0. \end{aligned} \tag{12}$$

and in vector form $\mathbf{e} = [e_x, e_y, e_\theta]^T$.

Substituting $\dot{x}$ and $\dot{y}$ from Eq. (5), the time-derivatives of the errors expressed in vector form are

$$\begin{pmatrix} \dot{e}_x(t) \\ \dot{e}_y(t) \\ \dot{e}_\theta(t) \end{pmatrix} = \begin{pmatrix} 0 & y(t) \\ -1 & -x(t) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} 0 \\ -\dot{y}^*(t) \\ 0 \end{pmatrix}. \tag{13}$$

with

$$\dot{y}^*(t) = \begin{cases} -\frac{d\pi}{2\tau} \sin(\frac{\pi t}{\tau}) & t \leq \tau, \\ 0 & t > \tau. \end{cases} \tag{14}$$

Or in its simplified form

$$\dot{\mathbf{e}} = \mathbf{L}\mathbf{V} + \mathbf{R}, \tag{15}$$

then, we can solve for

$$\mathbf{V} = \mathbf{L}^{+}\left[\dot{\mathbf{e}} - \mathbf{R}\right]. \tag{16}$$

In Eqs. (15) and (16), the matrix $\mathbf{L}$ is analogous to the so called interaction matrix commonly present in visual servo controllers [36], $\mathbf{R}$ is a known perturbation depending on the trajectory to track, $\mathbf{e}$ is the error that one wants to drive to $\mathbf{0}$, and $\mathbf{L}^{+}$ is the pseudo-inverse matrix of the matrix $\mathbf{L}$ defined as

$$\mathbf{L}^{+} = \frac{1}{y^2(t)+1}\begin{pmatrix} -x(t)y(t) & -y^2(t)-1 & -x(t) \\ y(t) & 0 & 1 \end{pmatrix}. \tag{17}$$

Since we want to have closed-loop error dynamics with the form

$$\dot{\mathbf{e}} = -\mathbf{K}\mathbf{e}, \tag{18}$$

where $\mathbf{K} = diag(k_1, k_2, k_3)$. Substituting Eq. (18) into Eq. (16) yields

$$\mathbf{V} = \mathbf{L}^{+}\left[-\mathbf{K}\mathbf{e} - \mathbf{R}\right]. \tag{19}$$

Expanding Eq, (19) results in

$$\mathbf{V} = \begin{pmatrix} v(t) \\ \omega(t) \end{pmatrix} = \frac{1}{y^2(t)+1}\begin{pmatrix} k_1 e_x(t)x(t)y(t) \\ +\left[k_2 e_y(t) - \dot{y}^*(t)\right]\left[y^2(t)+1\right] \\ +k_3 e_\theta(t)x(t) \\ -k_1 e_x(t)y(t) - k_3 e_\theta(t) \end{pmatrix}, \tag{20}$$

for all $t \in [0, \tau]$. Next, we proceed to prove the stability of controller (20) by means of Proposition 2 (with its Corollary 1) and Lemma 1, whose proofs are presented in Appendix D.

Proposition 2. *Given matrix* $\mathbf{L}\mathbf{L}^{+}$, $\mathbf{x}^T\mathbf{L}\mathbf{L}^{+}\mathbf{x} > 0$ *for all* $\mathbf{x} = (a,\, b,\, c)^T \in \mathbb{R}^3$ *with* $b \neq 0$.

Corollary 1. *Let* $\mathbf{x}$ *be the error vector* $\mathbf{e} = (x - x_0,\, y - y^*,\, \theta - \theta_0)^T$. *Condition* $\mathbf{e}^T\mathbf{L}\mathbf{L}^{+}\mathbf{e} \geq 0$ *holds for all* $\mathbf{e}$. *Additionally,* $\mathbf{e}^T\mathbf{L}\mathbf{L}^{+}\mathbf{e} > 0$ *for all* $\mathbf{e}$ *with* $y - y^* \neq 0$.

Due to disturbances of diverse nature affecting the whole system (noise in actuation, noise in measurements, varying environmental conditions, etc.), the tuple (x, y, θ) can be modeled as a random variable. Therefore, consider the sample space Ω of outcomes that map to tuples (x, y, θ).

Lemma 1. *Consider the experiment in which the DDR is controlled to follow a straight line motion using controller (20). Let $A_i \subset \Omega$ denote the event of getting a tuple (x, y, θ) in which $y - y^* = 0$, and let $\bar{A}_i$ be the complement of A_i, i.e., $\bar{A}_i = \{y - y^* \neq 0\}$. Then, if the experiment keeps running indefinitely, $P[\bar{A}_i \quad i.o.] = 1$.*

Theorem 3. *There exists a matrix of control gains for the controller (20) that guarantees that the tracking error vector $\mathbf{e}$, as defined in (12), converge globally asymptotically to the origin.*

Proof. Consider the following candidate Lyapunov function

$$\mathcal{V} = \frac{1}{2}||\mathbf{e}||^2 = \frac{1}{2}\mathbf{e}^T\mathbf{e}, \tag{21}$$

which accomplishes $\mathcal{V}(0) = 0$ and is positive definite. Substituting the control law $\mathbf{V}$ in its simplified form from (19) into Eq. (15), yields the closed-loop system dynamics

$$\dot{\mathbf{e}} = \mathbf{L}\mathbf{L}^+(-\mathbf{K}\mathbf{e}) + [-\mathbf{L}\mathbf{L}^+ + \mathbf{I}]\mathbf{R}. \tag{22}$$

Then, the time-derivative of the candidate Lyapunov function is given by:

$$\dot{\mathcal{V}} = -\mathbf{e}^T\mathbf{K}\mathbf{L}\mathbf{L}^+\mathbf{e} + \mathbf{e}^T[-\mathbf{L}\mathbf{L}^+ + \mathbf{I}]\mathbf{R}. \tag{23}$$

The second term can be seen as a vanishing perturbation, since by (14) $\lim_{t\to\infty}\mathbf{R} = 0$ and the only concern is to verify the negative definiteness of the first term. The product of matrices $\mathbf{L}\mathbf{L}^+$ is not an identity matrix; however, from Corollary 1, $\mathbf{e}^T\mathbf{L}\mathbf{L}^+\mathbf{e} \geq 0$ for all $\mathbf{e}$, and $\mathbf{e}^T\mathbf{L}\mathbf{L}^+\mathbf{e} > 0$ for all $\mathbf{e}$ with $y - y^* \neq 0$. Furthermore, from Lemma 1, condition $y - y^* \neq 0$ happens infinitely often. Therefore, there exists a matrix of control gains $\mathbf{K} > 0$ sufficiently large, which allows the controller to manage the couplings between errors introduced by the

product $\mathbf{L}\mathbf{L}^{+}$, such that global asymptotic convergence of the tracking errors toward $\mathbf{0}$ can be ensured. □

Remark 4. *The fact that coming across a tuple (x, y, θ) with $y - y^{*} = 0$ has zero probability, does not mean that such a scenario is impossible. Nonetheless, it suggests that it occurs so infrequently that we are unlikely to find it in practice. Indeed, during testing, that scenario has not been encountered.*

It is important to mention that it is necessary to consider the error induced by the inaccuracy of the method used to estimate the value of the robot orientation θ in the realistic space. Indeed, such a method acts as an observer over the robot orientation θ (Section 6.1). Thus, the estimation error over θ, refer to it as ϵ_θ, will be modelled through the error vector $\hat{\mathbf{e}}$ given by

$$\hat{\mathbf{e}} = \begin{pmatrix} x(t) - x_0 \\ y(t) - y^{*}(t) \\ \theta(t) - \theta_0 + \epsilon_\theta \end{pmatrix}, \tag{24}$$

or in its simplified form,

$$\hat{\mathbf{e}} = \mathbf{e} + \epsilon, \tag{25}$$

with

$$\epsilon = (0\ 0\ \epsilon_\theta)^{T}, \quad \text{where} \quad \epsilon_\theta = \hat{\theta}(t) - \theta(t), \tag{26}$$

being $\hat{\theta}(t)$ the estimation done by the proposed geometric method and $\theta(t)$ the real robot orientation. Obtaining the derivative of Eq. (25) with respect to time, results in the next error dynamics

$$\dot{\hat{\mathbf{e}}} = \mathbf{L}\mathbf{V} + \mathbf{R} + \dot{\epsilon}. \tag{27}$$

Since $\hat{\mathbf{e}}$ is measured instead of $\mathbf{e}$, the aforementioned controller acquires the form

$$\mathbf{V} = \mathbf{L}^{+}\left[-\mathbf{K}\hat{\mathbf{e}} - \mathbf{R}\right]. \tag{28}$$

Theorem 4. *There exist a matrix of control gains for the controller (28) that guarantees that the tracking error vector $\hat{\mathbf{e}}$, as defined in Eq. (24), is globally uniformly ultimately bounded (GUUB).*

Proof. Consider the following candidate Lyapunov function:

$$\mathcal{V}(\hat{\mathbf{e}}) = \frac{1}{2}||\hat{\mathbf{e}}||^2 = \frac{1}{2}\hat{\mathbf{e}}^T\hat{\mathbf{e}} \tag{29}$$

which accomplishes $\mathcal{V}(0) = 0$ and is positive definite. Substituting $\mathbf{V}$ from Eq. (28) into Eq. (27), yields the closed-loop system dynamics

$$\dot{\hat{\mathbf{e}}} = \mathbf{L}\mathbf{L}^+(-\mathbf{K}\hat{\mathbf{e}}) + [-\mathbf{L}\mathbf{L}^+ + \mathbf{I}]\mathbf{R} + \dot{\epsilon}. \tag{30}$$

The second term can be seen as a vanishing perturbation, since by Eq. (14) $\lim_{t\to\infty}\mathbf{R} = 0$. Therefore, the closed-loop dynamics can be simplified as

$$\dot{\hat{\mathbf{e}}} = -\mathbf{K}\mathbf{L}\mathbf{L}^+\hat{\mathbf{e}} + \dot{\epsilon}. \tag{31}$$

This results in a system with a nonvanishing perturbation $\dot{\epsilon}$. Given the definition of ϵ in Eq. (26), $\dot{\epsilon}$ depends on the robot trajectory, the velocities at which it is traveling, and the estimated orientation $\hat{\theta}$. Since the method that computes $\hat{\theta}$ assumes that the robot moves in arcs of circles and the ending orientation of an arc is the starting orientation of the next arc, then, the time derivative of the orientation of a DDR that moves on a circle is bounded in a bounded domain. Besides, regarding the robot trajectory, it is assumed that the robot linear velocity v and the robot angular velocity ω are Lipschitz functions, hence, the robot angular velocity $\dot{\theta} = \omega$ is also bounded in a bounded domain as well as the variation of the estimation error ϵ. Therefore, we have that the perturbation is bounded, i.e $\|\dot{\epsilon}\| \leq \varepsilon$. Then, the time-derivative of the candidate Lyapunov function is given by

$$\begin{aligned} \dot{\mathcal{V}}(\hat{\mathbf{e}}) &= -\hat{\mathbf{e}}^T\mathbf{K}\mathbf{L}\mathbf{L}^+\hat{\mathbf{e}} + \hat{\mathbf{e}}^T\dot{\epsilon} \\ &\leq -\hat{\mathbf{e}}^T\mathbf{K}\mathbf{L}\mathbf{L}^+\hat{\mathbf{e}} + \varepsilon\|\hat{\mathbf{e}}\| \end{aligned} \tag{32}$$

Although that matrix $\mathbf{L}\mathbf{L}^+$ is positive definite almost everywhere according to Corollary 1, the right-hand side of the inequality in (32) is not negative definite because, near the origin, the positive linear term $\varepsilon\|\hat{\mathbf{e}}\|$ dominates the negative quadratic term. However, there exists a sufficiently large matrix gain $\mathbf{K}$ which shall yield globally uniformly ultimately boundedness of the tracking error vector $\hat{\mathbf{e}}$ [22]. □

Remark 5. *It is plausible that if the system gets close to a state that produces an error* $\hat{\mathbf{e}}$ *such that* $\hat{\mathbf{e}}^T\mathbf{L}\mathbf{L}^+\hat{\mathbf{e}} = 0$*, which requires* $y - y^* = 0$*, the disturbances could dominate no matter how large the gain is. However, in practice, it has been observed that this does not appear to be an issue. This could be due the fact that* y^* *varies as a function of time, thereby helping to keep the state of the system outside of the neighbourhood where the disturbances dominate.*

7.2. Rotation in place

In this case, the feedback information needed to control the robot is retrievable in the reduced space. For this case, we will use the kinematics of the landmark given in polar coordinates (refer to Eq. (6)). The steps to get the controller for a rotation in place are completely analogous to those of the previous case of a straight line motion. Let us assume that we want the DDR to rotate in place an angle α clockwise (counterclockwise), in a time interval $I = [0, \tau]$. Then, the landmark must move counterclockwise (clockwise) on an arc of circle in the reduced space. The final points of this arc in conjunction with the origin, form an angle whose value is α.

Let $\phi(0) := \phi_0$ and $\phi(\tau) := \phi_\tau = \phi_0 - \alpha$. Then, we want ϕ to change from ϕ_0 to ϕ_τ. The desired value for ϕ is

$$\phi^*(t) = \frac{\alpha}{2}[1 + \cos(\frac{\pi t}{\tau})] + \phi_\tau. \tag{33}$$

The error for ϕ is given by

$$e_\phi(t) = \phi(t) - \phi^*(t), \tag{34}$$

then, the time-derivative for the error is

$$\dot{e}_\phi(t) = \dot{\phi}(t) - \dot{\phi}^*(t), \tag{35}$$

with

$$\dot{\phi}^*(t) = -\frac{\alpha\pi}{2\tau}\sin\left(\frac{\pi t}{\tau}\right). \tag{36}$$

Since the objective is to execute a pure rotation in place without any translation, the linear velocity $v(t)$ is fixed to 0. Considering the expression for $\dot{\phi}(t)$

given by Eq. (6) with $v(t) = 0$, the derivative of angle $\phi(t)$ simplifies to

$$\dot{\phi}(t) = \omega(t). \tag{37}$$

Substituting $\dot{\phi}^*(t)$ and $\dot{\phi}(t)$ from Eqs. (36) and (37) into Eq. (35), we get

$$\dot{e}_\phi(t) = \omega(t) + \frac{\alpha\pi}{2\tau} \sin\left(\frac{\pi t}{\tau}\right). \tag{38}$$

Equating $\dot{e}_\phi(t) = -k_3 e_\phi(t)$, yields

$$\omega(t) = -k_3 e_\phi(t) - \frac{\alpha\pi}{2\tau} \sin\left(\frac{\pi t}{\tau}\right), \tag{39}$$

therefore, our controller for executing rotations in place is

$$\mathbf{V} = \begin{pmatrix} v(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} 0 \\ -k_3 e_\phi(t) - \frac{\alpha\pi}{2\tau} \sin\left(\frac{\pi t}{\tau}\right) \end{pmatrix}. \tag{40}$$

Theorem 5. *The controller (40) with $k_3 > 0$ guarantees that the error $e_\phi(t)$ dynamics is exponentially stable.*

Proof. Substituting $\omega(t)$ from Eq. (39) into Eq. (38), results in the closed-loop system dynamics

$$\dot{e}_\phi(t) = -k_3 e_\phi(t). \tag{41}$$

Eq. (41) corresponds to a first-order and exponentially stable system, whose solution is $e_\phi(t) = e_{\phi(0)}\, \mathrm{e}^{-k_3 t}$. The result follows. □

8. Simulations

In this section, simulations are presented in which we evaluate the performance of the geometric method to estimate the robot's orientation and the performance of the proposed control law to perform straight lines motion primitives. Additionally, simulations are provided to illustrate the benefits of performing time-optimal trajectories.

8.1. Evaluation of the methodology to estimate θ

To evaluate the estimation of θ of our geometric method, a disturbed straight line trajectory of the DDR was simulated for 5 secs. The simulation was computed at 20 Hz in which noise was added to the DDR inputs. At each iteration of the simulation loop, the DDR inputs were kept constant, hence, the DDR performed an arc of circle at each iteration. Considering the simulation duration of 5 secs running at 20 Hz results into $N = 100$ (see Section 6.1). Figure 6 presents some results about the error on the estimation θ, ϵ_θ, which is defined as the difference between the estimated value and the real θ. Figure 6 presents three estimations in which the geometric method is executed at different frequencies: One at 20 Hz, corresponding to $n = N$. Another at 4 Hz, thus, $n < N$ as $n = 20$. And a third one at 1 Hz, again with $n < N$ since $n = 5$. The respective mean squared errors (MSE) are 3.85e-07, 7.3e-4, and 7.8e-3 respectively; hence, from the MSE and Figure 6, it can be observed that accuracy of the proposed geometric method increases as its sampling frequency increases. Even more, in all cases, ϵ_θ was kept within ± 0.2 radians (± 11.5 degs), see Figure 6.

8.2. Straight line controller evaluation

The present section provides a performance analysis of our straight line controller in the presence of disturbances. Table 1 shows that analysis. The results are organized in two main groups: open loop and closed loop. The open loop results refer to the DDR behaviour when input with the proper velocities to produce a straight line motion, more precisely, the velocities are set as $v = 1\frac{m}{s}$ and $w = 0$. Such results provide a baseline on the deviations from the reference trajectory, induced by the considered disturbances. The closed loop results correspond to the application of our feedback control law, which is fundamentally described by Eq. (20). In both cases, the task given to the DDR is to track a 3m straight line, thus, in the reference $y^*(t)$, a value $d = 3$ is set with $\tau = 3$.

The shown statistics correspond to the mean squared error (**MSE**) – along the executed trajectory – for the landmark coordinates (x, y) in the reduced space and the DDR heading θ in the realistic space, each of which are the

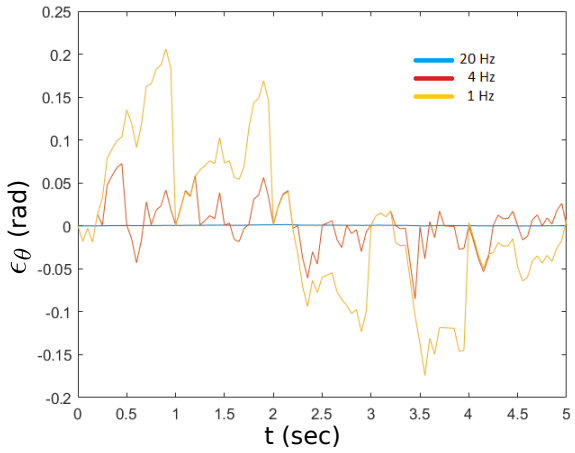


Figure 6: Errors on the estimation of θ performed by the geometric method from Section 6.1. The method is run at different sampling frequencies to depict how the estimation improves as the frequency increases. (The estimated θ remains the same as the previous estimated value until a new landmark reading is available for the geometric estimator to be executed.)

errors that our controller seeks to drive towards 0. In a similar vein, equivalent statistics are presented for the DDR position in the realistic space (x_R, y_R), that is, considering the error between the robot's position and the straight line reference, which evaluates the actual tracking task that the robot must fulfil. Additionally, errors for the terminal positions (**TE**) are also reported. The latter is defined as the Euclidean distance at time $t = 3$ between the reference position and the actual robot position.

The considered disturbances are as follows. First, equal disturbances on the DDR's input velocities, v and w, are contemplated. The analyzed disturbances are: zero mean Gaussian noise (G) with a standard deviation of 0.5, a periodic disturbance (P) defined as $0.5\sin(2\pi t)$, and a constant bias (C) of 1 unit. Second, disturbances on the measured variables are also added to emulate noise in the system sensors (NM). More specifically, Gaussian noise with mean of 0.1 (to emulate a bias) and a standard deviation of 0.1 is added to (x, y) and θ; subsequently, such perturbed signals are used to compute the controller inputs.

	Open loop			Closed loop						
	G	P	C	ND	G	P	C	G + NM	P + NM	C + NM
MSE e_x	0.345	0.820	2.100	0	0.021	0.172	0.027	0.031	0.181	0.035
MSE e_y	0.064	0.176	0.380	2.7e-05	0.001	0.022	0.004	0.013	0.032	0.015
TE (x, y)	0.680	0.273	2.640	0.008	0.086	0.474	0.158	0.175	0.473	0.180
MSE e_{x_R}	0.006	0.019	0.0461	0	1.6e-4	1.1e-4	0.001	6.2e-4	3.6e-4	1.6e-4
MSE e_{y_R}	0.054	0.061	0.047	2.7e-5	3e-4	0.002	7.9e-4	0.015	0.020	0.022
TE (x_R, y_R)	0.160	0.243	0.514	0.008	0.027	0.042	0.067	0.137	0.085	0.165
MSE e_θ	0.004	0.009	0.029	0	2.8e-4	0.002	2.6e-4	0.018	0.021	0.020

Table 1: Straight line controller performance.

As a note, the Gaussian statistics correspond to the average on 30 trials.

From the open loop columns, it can be observed that the constant bias was the disturbance that alter the most the system's behaviour, followed by the periodic disturbance and the Gaussian noise, with the exception of the terminal position error where the Gaussian noise was worse than the periodic disturbance. In the No Disturbances (ND) column of the closed loop statistics, it can be observed that the proposed straight line controller is able to track the reference in the absence of disturbances, achieving errors on the order of 10^{-3} or bellow – a gain $\mathbf{K} = \mathbf{I}$ was employed. The next three columns also show that despite the presence of disturbances, the controller was able to diminish the MSE and TE significantly compared to the open loop results; this was achieved with $\mathbf{K} = 5 \cdot \mathbf{I}$. The only statistic that has greater magnitude than its open loop counterpart is the TE (x, y) for the periodic disturbance; however, notice that the TE (x_R, y_R) in the realistic space also diminished, which is more relevant for the actual task. The same tendency, of reducing the MSE and the TE while applying the controller compared with the open loop statistics, is still observed in the case where noise in the measurements (+NM) is also added on top of the motion disturbances. As conclusion, the proposed straight line controller is capable of handling a considerable level of disturbances, both in the movement of the system and in its sensors.

8.3. Benefits of performing time-optimal trajectories

In the present section, we present series of simulations to assess the execution of optimal versus non-optimal trajectories. Since the time-optimal trajectories for the DDR are composed of rotations in place and straight line motions, we propose to make the DDR to travel two types of curves: a straight line and an arc of circle. The former is part of time-optimal trajectories, while the latter is not. Both the straight line and the arc of circle trajectories have the same start and end positions (start point (0,0) and end point (0,2), where the arc is half circle connecting those points). During the execution of both types of trajectories, uniform noise is added to the DDR inputs to simulate disturbances in the robots motion.

A first set of 30 trials were executed. The results show that, since the arc of circle is longer, the noise acts during a longer period of time, which in turn leaves the DDR in a farther position from the desired goal than the straight line. (In average, the perturbed straight line motion ended 0.055 m away from the end point (0,2), while the perturbed arc of circle ended 0.078 m from the same point.) Such phenomenon of the disturbances deviating longer trajectories could spread to other processes such as estimations of the state, as shown below.

In a second set of 30 trials, two estimators for the orientation of the final pose in the trajectories are tested: the proposed geometric estimator and orientation estimation through numerical simulation of odometry. The obtained results are presented in Table 2, which show the mean absolute error (**MAE**) between the real final orientation and the angle estimations performed with the two aforementioned methods. As expected, the estimation errors are smaller when the motion corresponds to a straight line, that is, to a time-optimal motion primitive. To summarize, the execution of time-optimal trajectories allows disturbances to occur during shorter periods of time, improving the performance of other processes that might be applied during the execution of the trajectories.

	Straight line	Arc of circle
MAE Odometry	2.691 °	3.251 °
MAE Geometric estimator	0.032 °	2.897 °

Table 2: Average errors between the real final orientation and the estimated through odometry, and between the real final orientation and the estimated through the geometric method. Errors are presented for both the straight line and the arc or circle trajectories.

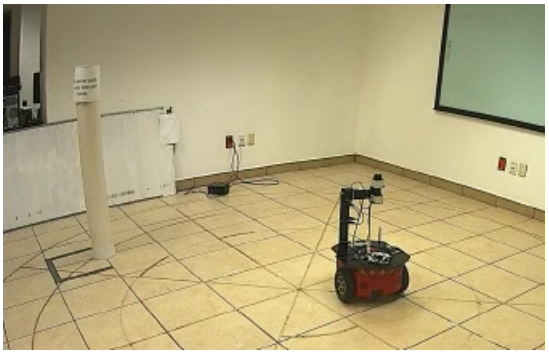
9. Experiments

In the experiments, we have used a Pioneer P3-DX robot, which is a differential drive system. In our implementation, all the algorithms run directly on the robot's computer, which is a Pentium M at 1.8 Ghz with 1 GB of RAM. The operating system is Linux using ROS. The control cycle runs at 12.5 Hz. The software is programmed in C++. An omnidirectional sensor was implemented using two laser range finders Hokuyo model URG-04LX, which were mounted on the robot in opposite directions, see Figure 7(a). The landmark is the cylinder shown at the left of Figure 7(b).

Our implementation of the landmark's detector is simple; it uses the raw data obtained with the lasers to detect two consecutive angular measurements with a difference in distance larger than a given threshold. If this jump in the measurements occurs, then a gap is detected. If two gaps occur separated by a distance close to the landmark diameter, then the landmark is detected. This simple landmark detector was coded to test the whole method: the estimator of the robot orientation θ (see Section 6), and controllers (see Section 7) in the real robot.

In the present experiments, the DDR is commanded to execute a time-optimal trajectory-previously computed using the algorithm provided in [2]-, which corresponds to a rotation in place, followed by a straight line motion, and ends with a second rotation in place. We compare the resulting trajectories based on whether the estimation of the robot orientation θ, is obtained with the robot's odometry, or is obtained using the method presented in Section 6.

(a) The robot and the omnidirectional sensor.



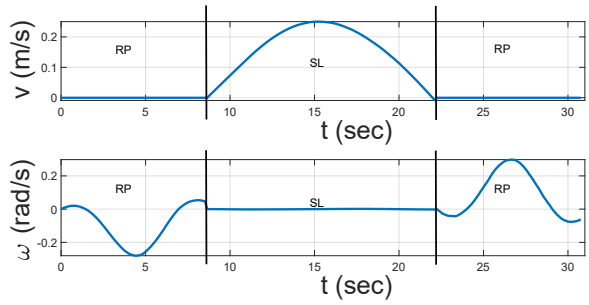
(b) The robot and the landmark, a white cylinder

Figure 7: The robot is a Pioneer P3-DX. The used omnidirectional sensor is two Hokuyo URG-04LX LIDARs mounted in opposite directions. The landmark is the white cylinder shown at the left part of Figure 7(b).

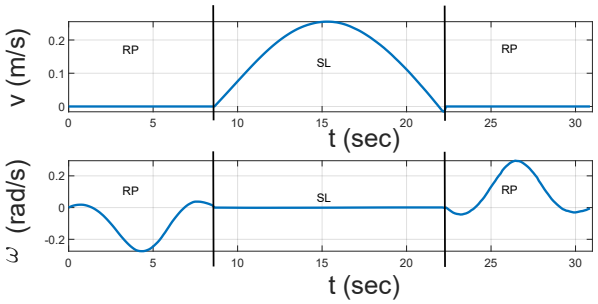
In both cases, the controllers presented in Section 7 are used to execute the trajectory, based on the single landmark location on the local reference frame and the robot orientation on a reference frame defined by the initial robot pose.

Figure 8(a) shows the control signals, v (linear velocity) and ω (angular velocity), in the case when the odometry is used to obtain the robot's orientation θ. In that figure, we label as SL the portion of the chart that corresponds to the execution of the controller that performs the straight line motion. Similarly, RP corresponds to the chart's portion when the rotations in place controller is active. Figure 8(b) shows the same control signals in the case when the method presented in Section 6 is used to estimate the robot's orientation θ. Again, we label as RP the chart's portion related to the rotations in place controller, and as SL the portion related to the straight line motion controller. Figure 8(a) and 8(b) show that in both cases, the resulting control signals generated with the controllers presented in Section 7 are continuous.

For the case when the odometry is used to obtain the robot's orientation θ, Figure 9(a) shows the errors in the x, y coordinates in the reduced space,



(a) Odometry.

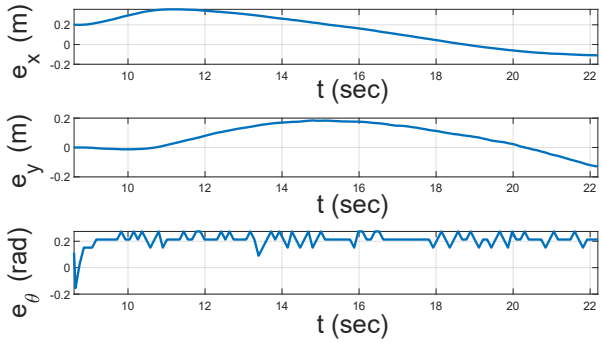


(b) Geometric estimator from Section 6.

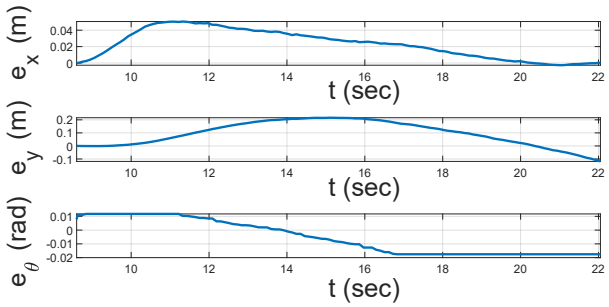
Figure 8: Linear v (m/sec) and angular ω (rad/sec) velocities, when the odometry or the method presented in Section 6 are used to estimate θ.

and orientation θ in the realistic space, resulting from tracking the desired reference in the straight line motion scenario. The errors in the coordinates are shown in meters and the error in the orientation θ is given in radians. Similarly, Figure 9(b) also shows the errors in the x, y coordinates, and the robot's orientation θ, while executing the straight line motion, but for the case when the method presented in Section 6 is used to estimate θ. Again, the errors in the x, y coordinates are in meters and the error in θ is given in radians. The errors in the x, y coordinates are smaller (in the order of few centimeters) when the geometric estimator from Section 6 is used, particularly for the x coordinate. The error in θ is also smaller (about one order of magnitude less) for the geometric estimator. In addition, the error in θ obtained with the odometry is more noisy. Finally, keep in mind that the design of the controller (28), from

690 a theoretical point of view, only guarantees bounded remaining errors but not zero-error. This explains the remaining errors observed in Figure 9.



(a) Odometry.



(b) Geometric estimator from Section 6.

Figure 9: Errors resulting from tracking the desired reference in the straight line motion scenario, in terms of the x, y coordinates in the reduced space, and the robot orientation θ in the realistic space. Errors are presented when either the odometry or the geometric estimator from Section 6 are used to estimate θ.

In the case of a rotation in place, the feedback information needed to control the robot is directly measurable in the reduced space. Figure 10 shows the tracking errors for the rotation in place scenario, in terms of the distance to
695 the landmark ρ and the landmark orientation ϕ relative to the robot's heading. The errors are in the order of a few centimeters for ρ and some centesimal of radians for ϕ.

Figure 11 shows, for an executed straight line, the value of θ (in radians) in

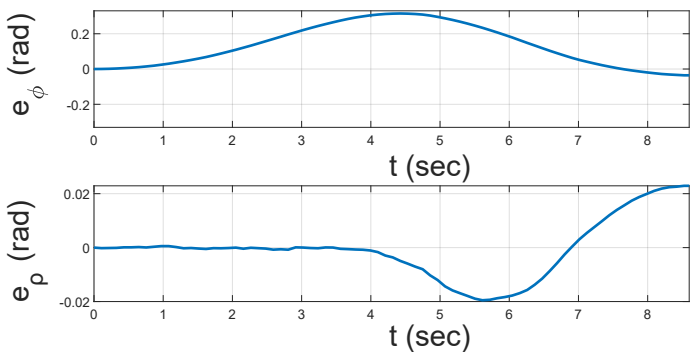


Figure 10: Errors resulting from tracking the desired reference in the rotation in place case, in terms of ρ and ϕ

the realistic space, when that orientation is obtained with odometry and when it is estimated using the method from Section 6. The red curve corresponds to the value obtained with the odomety and the blue curve to the geometric estimator. Both values are similar, but the orientation θ obtained with the odomety is a bit more noisy.

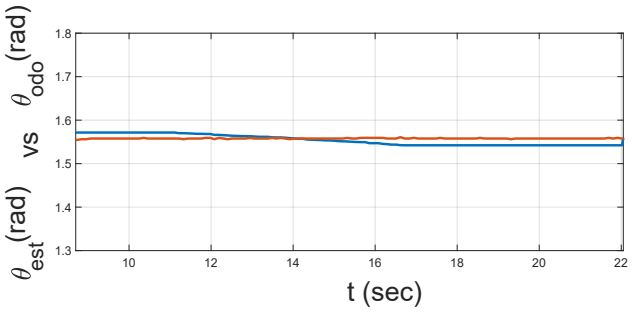


Figure 11: Comparison of the value of the robot orientation θ in the realistic space when the odometry is used (red curve) and when the geometric estimator proposed in Section 6 is used (blue curve).

In the experiments over the robot, we do not know exactly how fast the

dynamics of the robot is, furthermore, since a real-time implementation is not used, there might be arbitrary operating system interruptions. Therefore, we cannot determine whether or not there exist the condition to have zero error for the geometric estimator. However, we have a control method able to deal with errors in the estimation. As we can see, in the results of the experiments, the measured errors are small.

In the paper's *multimedia material*, we have included a video ("video1.mp4") showing the execution of a trajectory, for the case when the odometry is used to obtain θ, and for the case when the geometric estimator is used instead. Qualitatively speaking, both trajectories are comparable, but quantitatively, they present the differences that were just depicted along the whole present section.

As a final note, the utility of the proposed geometric estimator is that all the required sensing is onboard of the robot, it does not require external sensors, such as a motion capture device, to estimate the robot pose, which requires to modify the environment and compromises the robot's autonomy. Another possibility is to use the odometry as feedback information, but as we have seen, the results are better with the proposed method to estimate θ. Furthermore, depending on the robot and the type of environment (for example, on slippery ground), odometry might provide even worse results; the proposed method is independent of such factors.

10. Conclusions and future work

In this paper, we have proposed a feedback based control approach to execute the time-optimal motion trajectories for a differential drive robot. These trajectories are composed of straight lines and rotations in place. The location of a single landmark in a local reference frame defined by the robot is used as feedback information. We have shown that the evolution of a single landmark position over such a local reference frame makes it possible to track a prescribed time-optimal robot's trajectory. We have proposed a geometric method for com-

puting the robot's orientation θ, whose accuracy depends on the frequency in which the landmark position over the local reference frame is available. It has been shown that if the sampling rate n of the sensor is equal or larger than a bound N over that sampling rate, then the error of the methodology to estimate the robot's orientation θ is zero. We have designed a controller to track a reference trajectory using as feedback the location coordinates of a single landmark over a local reference frame. It has been shown that the control system is an exponentially stable system with a nonvanishing perturbation, and we can guarantee that globally uniformly ultimately boundedness of the tracking errors can be achieved. The proposed control scheme for execution of the motion primitives is similar to the one obtained using position-based visual servo control. However, note that it is straight forward to leverage visual servo control with methods from optimal control for yielding optimal trajectories; our approach is able to do that and execute *time-optimal primitives in the state space*. Furthermore, the approach is able to work with the minimum number of landmarks–only one– this represents a *necessary and sufficient* condition for feedback-based optimal navigation based on landmarks. It is also important to mention that the use of an omnidirectional laser range finder simplifies the implementation, avoiding further complicating matters resulting from using a camera. The direct use case of the proposed method is landmark-based navigation, which is a relevant problem in robotics. An important technical contribution is that the approach does not require to reconstruct the whole state of the robot in a global reference frame, which can be a complex task, in which it is hard to rapidly obtain robust and accurate estimations. A drawback of the method is that it requires an accurate sensor to measure the distance from the robot to the landmark and the direction of the landmark with respect to the robot. However, in practice laser range finders enjoy a large accuracy.

Finally, experiments in a physical robot, a nonholonomic differential drive system equipped with an omnidirectional laser sensor, have been presented, which validate the proposed theoretical modelling.

As a future work, we would like to determine some quantitative bounds on

the error of the estimation of the robot's orientation θ when a robot trajectory cannot be decomposed into arcs of circles, according to some bounds on the motion error.

Appendix A. Kinematics of the landmark in Cartesian coordinates

In this section, we deduce the landmark velocities in Cartesian coordinates.

$$\begin{aligned}
\frac{d}{dt}x(t) = & \frac{d}{dt}\left[(x_L - x_R(t))\sin(\theta(t)) - (y_L - y_R(t))\cos(\theta(t))\right] \\
& +(y_L - y_R(t))\sin(\theta(t))\dot{\theta}(t) + \cos(\theta(t))\dot{y}_R(t) \\
= & (x_L - x_R(t))\cos(\theta(t))\omega(t) - v(t)\sin(\theta(t))\cos(\theta(t)) \\
& +(y_L - y_R(t))\sin(\theta(t))\omega(t) + v(t)\cos(\theta(t))\sin(\theta(t)) \\
= & \omega(t)((x_L - x_R(t))\cos(\theta(t)) + (y_L - y_R(t))\sin(\theta(t))) \\
= & \omega(t)y(t).
\end{aligned}$$

$$\begin{aligned}
\frac{d}{dt}y(t) = & \frac{d}{dt}\left[(x_L - x_R(t))\cos(\theta(t)) + (y_L - y_R(t))\sin(\theta(t))\right] \\
& +(y_L - y_R(t))\cos(\theta(t))\dot{\theta}(t) - \sin(\theta(t))\dot{y}_R(t) \\
= & -(x_L - x_R(t))\sin(\theta(t))\omega(t) - \cos(\theta(t))v(t)\cos(\theta(t)) \\
& +(y_L - y_R(t))\cos(\theta(t))\omega(t) - \sin(\theta(t))v(t)\sin(\theta(t)) \\
= & -\omega(t)[(x_L - x_R(t))\sin(\theta(t)) - (y_L - y_R(t))\cos(\theta(t))] \\
& -v(t)[\cos^2(\theta(t)) + \sin^2(\theta(t)] \\
= & -\omega(t)x(t) - v(t).
\end{aligned}$$

Appendix B. Kinematics of the landmark in polar coordinates

Now, we present the landmark velocities in polar coordinates. We start by obtaining an expression for $\dot{\rho}$.

$$\begin{aligned}
\frac{d}{dt}\rho(t) &= \frac{d}{dt}[\sqrt{x^2 + y^2}] \\
&= \frac{\omega(t)x(t)y(t) + y(t)(-\omega(t)x(t) - v(t))}{\rho(t)} \\
&= \frac{-v(t)y(t)}{\rho(t)}.
\end{aligned}$$

Substituting y from Eq. (4) in the equation above, we get

$$\frac{d}{dt}\rho(t) = -v(t)\cos\phi(t).$$

We proceed to obtain an expression for $\dot{\phi}$ as

$$\begin{aligned}\frac{d}{dt}\phi(t) &= \frac{d}{dt}\left[\tan^{-1}\left(\tfrac{x}{y}\right)\right] \\ &= \frac{y(t)\dot{x}(t) - x(t)\dot{y}(t)}{y^2(t) + x^2(t)}.\end{aligned}$$

Substituting Eq. (5) in the equation above, we get

$$\frac{d}{dt}\phi(t) = \frac{\omega(t)\left[x^2(t) + y^2(t)\right] + v(t)x(t)}{x^2(t) + y^2(t)}.$$

Substituting x and y from Eq. (4) in the equation above, we get

$$\frac{d}{dt}\phi(t) = \omega(t) + \frac{v\sin\phi(t)}{\rho(t)}.$$

Appendix C. Non-observability

Definition 5. *Given a configuration $q = (x_R, y_R, \theta) \in \mathbb{R}^2 \times S^1$, we define the* transversal reflection *of q as the configuration*

$$\tilde{q} = (2x_L - x_R, 2y_L - y_R, \theta + \pi). \tag{C.1}$$

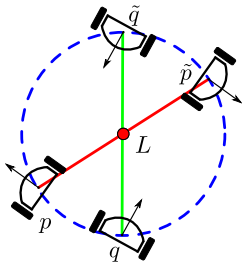


Figure C.12: DDR Transversal Reflection.

Notice that the poses (x_R, y_R) and $(2x_L - x_R, 2y_L - y_R)$ are antipodal points on the circumference in the realistic space with center on the landmark $L = (x_L, y_L)$ and radius $\|(x_R - x_L, y_R - y_L)\|$. See Fig C.12.

Lemma 2. *Let $q = (x_R, y_R, \theta)$ be a robot configuration and $\tilde{q} = (x'_R, y'_R, \theta')$ its transversal reflection. Considering the observation function $g(q) = T_q(L)$, then $g(q) = g(\tilde{q})$.*

Proof. Let $g(q) = (x, y)$ and $g(\tilde{q}) = (x', y')$. The coordinates of the landmark in the reduced space with respect to the configuration q are

$$\begin{aligned} x &= (x_L - x_R)\sin(\theta) - (y_L - y_R)\cos(\theta), \\ y &= (x_L - x_R)\cos(\theta) + (y_L - y_R)\sin(\theta), \end{aligned} \tag{C.2}$$

while for the configuration $\tilde{q}$ we have

$$\begin{aligned} x' &= (x_L - x'_R)\sin(\theta') - (y_L - y'_R)\cos(\theta'), \\ y' &= (x_L - x'_R)\cos(\theta') + (y_L - y'_R)\sin(\theta'). \end{aligned} \tag{C.3}$$

Since $\tilde{q}$ is the transversal reflection of q, substituting $x'_R = 2x_L - x_R$, $y'_R = 2y_L - y_R$ and $\theta' = \theta + \pi$ in Eq. (C.3), yields

$$\begin{aligned} x' &= (x_R - x_L)\sin(\theta + \pi) - (y_R - y_L)\cos(\theta + \pi), \\ y' &= (x_R - x_L)\cos(\theta + \pi) + (y_R - y_L)\sin(\theta + \pi). \end{aligned} \tag{C.4}$$

Moreover

$$\sin(\theta + \pi) = \sin(-\theta) = -\sin(\theta),\ -\cos(\theta + \pi) = \cos(-\theta) = \cos(\theta),$$

then

$$\begin{aligned} x' &= (x_L - x_R)\sin(\theta) - (y_L - y_R)\cos(\theta), \\ y' &= (x_L - x_R)\cos(\theta) + (y_L - y_R)\sin(\theta). \end{aligned} \tag{C.5}$$

Thus, from Eqs. (C.2) and (C.5), $(x, y) = (x', y')$. Therefore, $g(q) = g(\tilde{q})$. □

Proof of Theorem 1.

Proof. Let $q = (x_R, y_R, \theta)$ be a configuration for the DDR and let $\tilde{q} = (x'_R, y'_R, \theta')$ be its transversal reflection. Let $(u(t), [\sigma, \tau])$ be an admissible control, hence, $\Sigma_q(u(t), [\sigma, \tau])$ and $\Sigma_{\tilde{q}}(u(t), [\sigma, \tau])$ are the input-output maps given $(u(t), [\sigma, \tau])$, starting from the configurations q and $\tilde{q}$, respectively. Assume that $0 \leq \Delta_t \leq |\tau - \sigma|$ and let $t = \sigma + \Delta_t$. The evolution of the system starting from the configuration $q(\sigma)$ is dictated by

$$\begin{aligned} \dot{x}_R &= v\cos\theta, \\ \dot{y}_R &= v\sin\theta, \\ \dot{\theta} &= \omega, \end{aligned} \tag{C.6}$$

while for configuration $\tilde{q}(\sigma)$ we have

$$\begin{aligned} \dot{x}'_R &= v\cos\theta', \\ \dot{y}'_R &= v\sin\theta', \\ \dot{\theta}' &= \omega. \end{aligned} \tag{C.7}$$

By substituting $\theta' = \theta + \pi$ in Eq. (C.7) and simplifying, we have

$$\begin{aligned} \dot{x}'_R &= -\dot{x}_R, \\ \dot{y}'_R &= -\dot{y}_R, \\ \dot{\theta}' &= \dot{\theta}. \end{aligned} \tag{C.8}$$

In this way, using a forward Euler approximation to solve the differential equations for the configuration $\tilde{q}(t)$, starting from $\tilde{q}(\sigma) = (2x_L - x_R(\sigma), 2y_L - y_R(\sigma), \theta(\sigma) + \pi)$, and considering Eqs. (C.8) and (C.6), we have

$$\begin{aligned} x'_R(t) &= (2x_L - x_R(\sigma)) - v(\sigma)\cos(\theta(\sigma))\Delta_t, \\ y'_R(t) &= (2y_L - y_R(\sigma)) - v(\sigma)\sin(\theta(\sigma))\Delta_t, \\ \theta'(t) &= (\theta(\sigma) + \pi) + \omega(\sigma)\Delta_t. \end{aligned} \tag{C.9}$$

By simplifying the system in Eq. (C.9) we get

$$\begin{aligned} x'_R(t) &= 2x_L - x_R(t), \\ y'_R(t) &= 2y_L - y_R(t)), \\ \theta'(t) &= \theta(t) + \pi. \end{aligned} \tag{C.10}$$

Thus, $\tilde{q}(t)$ is the transversal reflection of $q(t)$ for all $t \in [\sigma, \tau]$. From Lemma 2, $g(q) = g(\tilde{q})$, then, $\Sigma_q(u(t), [\sigma, \tau]) = \Sigma_{\tilde{q}}(u(t), [\sigma, \tau])$ for all admissible control $(u(t), [\sigma, \tau])$. This means that the configurations q and $\tilde{q}$ are indistinguishable, therefore, the system Σ is not observable. □

Appendix D. Closed loop stability

Proposition 2. *Given matrix* $\mathbf{L}\mathbf{L}^+$, $\mathbf{x}^T\mathbf{L}\mathbf{L}^+\mathbf{x} > 0$ *for all* $\mathbf{x} = (a,\ b,\ c)^T \in \mathbb{R}^3$ *with* $b \neq 0$.

Proof. First, we have

$$\mathbf{L}\mathbf{L}^{+} = \frac{1}{y^2+1}\begin{pmatrix} y^2 & 0 & y \\ 0 & y^2+1 & 0 \\ y & 0 & 1 \end{pmatrix}. \tag{D.1}$$

Then

$$\begin{aligned} \mathbf{x}^T\mathbf{L}\mathbf{L}^{+}\mathbf{x} &= \begin{pmatrix} a & b & c \end{pmatrix}\begin{pmatrix} \frac{y^2}{y^2+1} & 0 & \frac{y}{y^2+1} \\ 0 & 1 & 0 \\ \frac{y}{y^2+1} & 0 & \frac{1}{y^2+1} \end{pmatrix}\begin{pmatrix} a \\ b \\ c \end{pmatrix} \\ &= \frac{(ay+c)^2}{y^2+1} + b^2. \end{aligned} \tag{D.2}$$

Both adders in the last equation are non negative, therefore, $\mathbf{x}^T\mathbf{L}\mathbf{L}^{+}\mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathbb{R}^3$. Moreover, $b \neq 0$ suffices for $\mathbf{x}^T\mathbf{L}\mathbf{L}^{+}\mathbf{x} > 0$. The result follows. □

Proof of Lemma 1.

Proof. The condition $y - y^* = 0$ implies the specific requirement of $y = y^*$; thus, A_i is related to a zero-measure set (see Figure D.13). Consequently, $P[A_i] = 0$, which in turn implies $P[\bar{A}_i] = 1$. Additionally, as the experiment keeps running indefinitely, contemplate the sequence of events $\{\bar{A}_i\}_{i=1}^{\infty}$. Since $P[\bar{A}_i] = 1$, $\sum_{i=1}^{\infty} P[\bar{A}_i] = \infty$, then, by the Borel-Cantelli lemma, $P[\bar{A}_i \quad i.o.] = 1$. □

[1] J.-P. Laumond, Robot Motion Planning and Control, *Lecture Notes in Control and Information Sciences* Vol. 29, Springer, 1998.

[2] D. J. Balkcom, M. T. Mason, Time Optimal Trajectories for Bounded Velocity Differential Drive Vehicles. *I. J. Robotics Res.* 21(3): 199-218, 2002.

[3] C. Samson, K. Ait-Abderrahim, Feedback Control of a Nonholonomic Wheeled Cart in Cartesian Space. *IEEE International Conference on Robotics and Automation*, ICRA, 1991, pages 1242-1247.

[4] C. Samson and K. Ait-Abderrahim, Mobile robot control. Part 1 : Feedback control of nonholonomic wheeled cart in Cartesian space, *[Research Report] RR-1288, INRIA*. 1990.

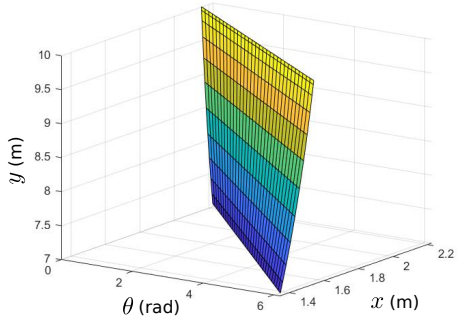


Figure D.13: The shown surface corresponds to an example of a set of tuples (x, y, θ) that produce $\mathbf{e}^T \mathbf{L}\mathbf{L}^+ \mathbf{e} = 0$, where $\mathbf{e} = (x - x_0,\ y - y^*,\ \theta - \theta_0)^T$. The set is a zero-measure set.

[5] R. W. Brockett. Asymptotic stability and feedback stabilization. In R. W. Brockett, R. S. Millman, and H. J. Sussmann, editors, *Differential Geometric Control Theory*. Birkhauser, Boston, MA, 1983.

[6] M. Betke and L. Gurvits, Mobile robot localization using landmarks. *IEEE Trans. Robotics and Automation* 13(2): 251-263, 1997.

[7] J. J. Leonard, H. F. Durrant-Whyte, Mobile robot localization by tracking geometric beacons. *IEEE Trans. Robotics and Automation* 7(3): 376-382, 1991.

[8] S. M. LaValle, Sensing and Filtering: A Fresh Perspective Based on Preimages and Information Spaces. *Foundations and Trends in Robotics*. 1(4): 253-372, 2012.

[9] S. Bhattacharya, R. Murrieta-Cid, and S. Hutchinson, Optimal paths for landmark-based navigation by differential-drive vehicles with field-of-view constraints. *IEEE Trans. Robot.*, vol. 23, no. 1, pp. 4759, 2007

[10] P. Salaris, D. Fontanelli, L. Pallottino, A. Bicchi, Shortest paths for a robot with nonholonomic and field-of-view constraints, *IEEE Trans. Robot.* 26, 269281, 2010.

[11] J.-B. Hayet, H. Carlos, C. Esteves, R. Murrieta-Cid, Motion planning for maintaining landmarks visibility with a differential drive robot. *Robot. Auton. Syst.* 4(62), 456473, 2014.

[12] J. Chen, W. E. Dixon, D. M. Dawson, and M. McIntyre, Homography-based visual servo tracking control of a wheeled mobile robot, *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 407416, Apr. 2006.

[13] G. Lopez-Nicolas, N. R. Gans, S. Bhattacharya, C. Sagues, J. J. Guerrero, and S. Hutchinson, Homography-based control scheme for mobile robots with nonholonomic and field-of-view constraints. *IEEE Trans. Syst., Man, Cybern. B*, vol. 40, no. 4, pp. 11151127, 2010.

[14] H. M. Becerra and C. Sagues, Exploiting the trifocal tensor in dynamic pose estimation for visual control. *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 5, pp. 19311939, 2013.

[15] X. Liang, H. Wang, W. Chen, D. Guo and T. Liu, Adaptive Image-Based Trajectory Tracking Control of Wheeled Mobile Robots With an Uncalibrated Fixed Camera *IEEE Transactions on Control Systems Technology*, vol. 23, no. 6, pp. 2266-2282, 2015.

[16] X. Liang, H. Wang, Y. Liu, B. You, Z. Liu and W. Chen. Calibration-Free Image-Based Trajectory Tracking Control of Mobile Robots With an Overhead Camera, *IEEE Transactions on Automation Science and Engineering*, Vol. 17, No. 2, pp. 933-946, 2020.

[17] K. Wang and Y. Liu and L. Li. Visual Servoing Trajectory Tracking of Nonholonomic Mobile Robots Without Direct Position Measurement, *IEEE Transactions on Robotics*, Vol. 30, No. 4, pp. 1026-1035, 2014.

[18] B. Li and Y. Fang and G. Hu and X. Zhang. Model-Free Unified Tracking and Regulation Visual Servoing of Wheeled Mobile Robots, *IEEE Transactions on Control Systems Technology*, Vol. 24, No. 4, pp. 1328-1339, 2016.

[19] J. Chen and B. Jia and K. Zhang. Trifocal Tensor-Based Adaptive Visual Trajectory Tracking Control of Mobile Robots, *IEEE Transactions on Cybernetics*, Vol. 47, No. 11, pp. 3784-3798, 2017.

[20] L. Li and Y. Liu and T. Jiang and K. Wang and M. Fang. Adaptive Trajectory Tracking of Nonholonomic Mobile Robots Using Vision-Based Position and Velocity Estimation, *IEEE Transactions on Cybernetics*, Vol. 48, No. 2, pp. 571-582, 2018.

[21] K. Zhang and J. Chen and Y. Li and Y. Gao. Unified Visual Servoing Tracking and Regulation of Wheeled Mobile Robots With an Uncalibrated Camera, *IEEE/ASME Transactions on Mechatronics*, Vol. 23, No. 4, pp. 1728-1739, 2018.

[22] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, New Jersey: Prentice Hall, 2002.

[23] P. Soueres and J. -. Laumond, Shortest paths synthesis for a car-like robot, in *IEEE Transactions on Automatic Control*, vol. 41, no. 5, pp. 672-688, May 1996, doi: 10.1109/9.489204.

[24] H. Wang, Y. Chen and P. Soueres. A geometric algorithm to compute time-optimal trajectories for a bidirectional steered robot. *IEEE Transactions on Robotics*, Vol. 25, No. 2, pp. 399-413, 2009.

[25] D. J. Balkcom, P. A. Kavathekar, M. T. Mason, Time-optimal Trajectories for an Omni-directional Vehicle. *I. J. Robotics Res.* 25(10): 985-999, 2006.

[26] H. R. Chitsaz, S. M. LaValle, D. J. Balkcom, M. T. Mason, Minimum Wheel-Rotation Paths for Differential-Drive Mobile Robots. *I. J. Robotics Res.* 28(1): 66-80, 2009

[27] A. A. Furtuna, D. J. Balkcom, Generalizing Dubins Curves: Minimum-time Sequences of Body-fixed Rotations and Translations in the Plane. *I. J. Robotics Res.* 29(6): 703-726, 2010.

[28] V. Macias, I. Becerra, R. Murrieta-Cid, H. M. Becerra, S. Hutchinson, Image Feedback based Optimal Control and the Value of Information in a Differential Game, *Automatica*, Vol. 90. pp 271-285, 2018.

[29] A. Martinelli and R. Siegwart, Observability analysis for mobile robot localization, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1471-1476, Aug. 2005.

[30] G. Jang, S. Kim, J. Kim, I. Kweon, Metric Localization Using a Single Artificial Landmark for Indoor Mobile Robots. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, IROS, 2005, pages 2857-2862.

[31] H. Sert, A. Kokosy and W. Perruquetti, A single landmark based localization algorithm for non-holonomic mobile robots. *IEEE International Conference on Robotics and Automation*, ICRA, 2011, pages 293-298.

[32] Alexander Amini, Guy Rosman, Sertac Karaman and Daniela Rus, Variational End-to-End Navigation and Localization. *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8958-8964, Montreal, Canada, 2019.

[33] N. Radwan, A. Valada and W. Burgard, "VLocNet++: Deep Multitask Learning for Semantic Visual Localization and Odometry," in IEEE Robotics and Automation Letters, vol. 3, no. 4, pp. 4407-4414, Oct. 2018, doi: 10.1109/LRA.2018.2869640.

[34] R. Isaacs. Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization.*John Wiley and Sons, Inc.*, New York, 1965.

[35] R. Hermann and A. J. Krener, Nonlinear Controllability and Observability, *IEEE Trans. Automatic Control*, Vol. 22.No 5. pp 728-740, 1977.

[36] F. Chaumette and S. Hutchinson. Visual servo control. I. Basic approaches, *IEEE Robot. Autom. Mag.*, 13(4):82–90, 2006.