in Proc. of IV Jornadas de Engenharia Electrónica
e Telecomunicações e de Computadores, pp180-185
November 2008, Lisbon, Portugal

CONTROL OF UNICYCLE TYPE ROBOTS
Tracking, Path Following and Point Stabilization

Ricardo Carona, A. Pedro Aguiar, José Gaspar
Instituto de Sistemas e Robótica, Instituto Superior Técnico,
Universidade Técnica de Lisboa, Portugal
ricardo.carona@gmail.com, {pedro,jag}@isr.ist.utl.pt

Keywords: Unicycle, Dynamics, Tracking, Path Following, Point Stabilization

Abstract: This paper considers the motion control problem of unicycle type mobile robots. We present the mathematical model of the mobile robots taken explicitly into account their dynamics and formulate the respectively motion control strategies of tracking and path-following. Two types of controllers presented in the literature are revised and their performance are illustrated through computer simulations. The problem of regulation to a point is also addressed.

1 INTRODUCTION

In the near future personal mobile robots will be providing a better life not only to common people but especially to elderly and impaired. In particular wheeled robots will be expected to provide many convenient and user friendly transport solutions for both people and objects [7, 9].

The importance of the wheeled mobile robots has long been recognized by the robotics research community, as shown by the numerous robotic competitions and research projects run worldwide in the last decades. See e.g. in the *RoboCup* [6] and *Robotica 2008* sites [4] an history of the many past competition events. The importance of the subject motivated and continues motivating many projects. For instance searching the term *control* on the site of the European *FP7* [8] shows various projects starting on the activity of *Intelligent and safe vehicles*.

The class of unicycle type (mobile) robots, i.e. robots having some forward speed but zero instantaneous lateral motion, is frequently selected for designing and modeling robots. For example many of the robotic competition teams of the last decade selected those robots due to their simplicity and good maneuverability, allowing for example to follow complex trajectories [3, 1, 2]. At the same time research was conducted on controllability, feedback-linearization and -stabilization [5].

Control of unicycle type robots continues raising many research and development challenges. Some objectives include following trajectories and parking, while doing collision avoidance and cooperation with other robots, as in convoying or formation control [10]. The variety of objectives imply multiple research foci. For example, complex applications often imply hierarchical designs, which regulate in each level the performance of the lower control level and thus allow in many cases using linear controllers [3, 1]. Other applications, requiring e.g. guarantees of close loop stability, often imply non-linear controller designs based on Lyapunov techniques [5, 10].

Our work targets smooth pre-defined navigation tasks or human interaction with the robot, and therefore we do not require highly complex behaviors to be performed by the robot, as the human can locally regulate the robot navigation task, however we require the robot to filter and react in a stable manner to noisy reference signals.

In this work we present the mathematical model of the mobile robots taking explicitly into account their dynamics to formulate motion control strategies of tracking where the vehicle has to track a time parameterized reference path (set of positions) and path-following in which the objective is to steer the vehicle to it's path at a reference speed. Two types of controllers presented in the literature are revised and their performance are illustrated through computer simulations[10, 5]. The problem of regulation to a point is also addressed.

This paper is structured as follows. Section 2 introduces mathematical models for the kinematics and dynamics of unicycle-robots. Section 3 details the motion controllers. Section 4 describes some simulations and their results. Finally Section 5 contains some analysis of the results and draws some conclusions.

2 UNICYCLE MODEL

A unicycle type robot is in general a robot moving in a 2D world, having some forward speed but zero instantaneous lateral motion. In other words, it is a non-holonomic system. Despite the unicycle name, it describes in general carts or cars having usually two parallel driven wheels, one mounted of each side of their center. This model comprises many known differential drive robots and approximate in many situations even the four wheeled cars.

Modeling unicycle type robots comprises studying their kinematics and dynamics, as usual with most of the physical systems. Kinematics modeling describes the trajectories that the mobile robots follow when subject to commanding speeds. The dynamics modeling complements the kinematics by accounting to the commanding forces and intrinsic frictions, actually defining the commanding speeds.

2.1 Kinematic Model

The kinematic model of a unicycle type robot is usually described by a simple non-linear model:

$$\left\{\begin{array}{lcl} \dot{x} & = & v\cos(\theta) \\ \dot{y} & = & v\sin(\theta) \\ \dot{\theta} & = & \omega \end{array}\right. \tag{1}$$

where $P = (x, y, \theta)$ is the robot position and orientation in world reference frame, and the pair (v, ω) is the input control encompassing the linear and angular velocities.

2.2 Dynamic Model

The kinematic model of a unicycle is mainly concerned with a geometric description of the trajectories followed by the robot, disregarding the origins of forces. This means for example that force saturations or dead-zones, internal to the robot or resulting from heavier loads, are not incorporated in the kinematics and thus not considered. Hence, significant errors can occur when one simulates or tests real unicycle robots. In this section we describe a dynamic model for unicycle robots which allows including physical limitations of the robot or forces, binaries or friction.

According to the 2nd Newton's Law, the translational and rotational dynamics of the unicycle robot can be described by $M\dot{v} = F - B_v v$ and $J\dot{\omega} = T - B_\omega \omega$, respectively, where M is the robot's mass, J is the Inertia Moment, F are the forces applied on the system, T is the wheel axis binary, B_v is the translational friction coefficient and B_ω is rotational friction coefficient. We assume that these values are constant for the typical values of velocities of a moving robot.

Considering the forces acting on the left and right wheels, F_L and F_R, one has that these forces are equivalent to the force F applied in the point r_F, referred to the center of the wheel's axis as shown in Fig.1. More precisely, one has $F = F_R + F_L$ and $T = l(F_R - F_L)$, where l represents half the length of the axis between the wheels.

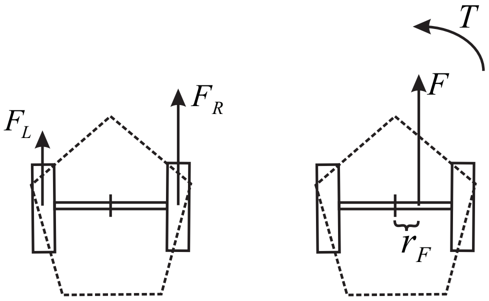


Figure 1: Forces on the wheels (a) and an equivalent representation (b).

Similarly, considering the voltages of the left and right wheel motors, one can define also the mean (average) and differential voltages, e_{am} and e_{ad}, and finally obtain the relationship between voltages and forces, $F = K_m e_{am} - K_v v$ and $T = K_d e_{ad} - K_\omega \omega$, where K_m, K_v, K_d, K_ω are constants. Concluding, we have the following dynamic model for the unicycle:

$$\left\{\begin{array}{lcll} M\frac{dv}{dt} & = & -K_v v & +K_m e_{am} \\ J\frac{d\omega}{dt} & = & -K_\omega \omega & +K_d e_{ad} \end{array}\right. . \tag{2}$$

3 MOTION CONTROL

This section describes the motion control laws for tracking, path-following and point regulation. The controllers proposed exhibit a inner-outer-loop structure (see Fig.3). The inner-loop control law (common to all the motion control laws proposed) is responsible to compute the adequate electrical signals (voltage) that will tackle the wheels's motors to force the robot to move according to a desired linear and angular velocity. These desired velocities are the control signals generated by the outer-loop controller.

3.1 Inner-Loop Control

To accomplish the goal of driving the robot to a desired linear velocity v_d and angular velocity ω_d, a first step is to compute the error between the true velocities and the desired ones. To this effect, let $e_v = v - v_d$ and $e_\omega = \omega - \omega_d$ be respectively the linear and angular velocity errors. Then, from Eq.2 we can conclude that a simple proportional control law of the form

$$e_{am} = -K_{p_1} e_v \quad (3)$$
$$e_{ad} = -K_{p_2} e_\omega \quad (4)$$

drives the errors to a small neighborhood of zero. If the dynamic of the robot has small static gains then the error neighborhood may be significant. In that case one can enforce the steady state error convergence to zero (with a constant input) by adding an integral term, that is,

$$e_{am} = -K_{p_1} e_v - K_{i_1} \int_0^t e_v(\tau)\, d\tau \quad (5)$$
$$e_{ad} = -K_{p_2} e_\omega - K_{i_2} \int_0^t e_\omega(\tau)\, d\tau. \quad (6)$$

3.2 Tracking

In tracking, the goal is to force the robot to track a time parameterized reference position. In this work we consider two different types of controllers for the outer-loop, which we term by *Controller-A* and *Controller-B*. *Controller-A* denotes the class of nonlinear controllers for the Tracking and Path Following described in [5]. *Controller-B* represents an unified controller proposed in [10]. In the following, we describe the motion controllers for Tracking.

Tracking with *Controller-A* - Consider a virtual unicycle type robot governed by

$$\left\{ \begin{array}{lcl} \dot{x}_r & = & v_r \cos(\theta_r) \\ \dot{y}_r & = & v_r \sin(\theta_r) \\ \dot{\theta}_r & = & \omega_r \end{array} \right. \quad (7)$$

where $p_r(t) = \big(x_r(t), y_r(t)\big)$ is the desired reference position at time t. The objective is to derive an outer-loop control system responsible to generate the adequate desired linear and angular velocities (v_d, ω_d) to force the position of the robot $p = (x, y)$ to converge to the reference position $p_r = (x_r, y_r)$. To achieve this, consider first the position error expressed in the body referential

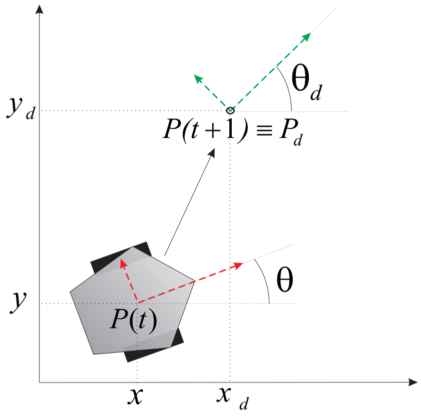


Figure 2: Tracking

$$e = R(p_r - p), \quad R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \quad (8)$$

Clearly, if e goes to zero then p converges to p_r. Using the results in [5], the control law is given by

$$\begin{bmatrix} v_d \\ \omega_d \end{bmatrix} = C \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} - \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (9)$$

where $C = diag\{\cos(\theta_r - \theta), 1\}$, $u_1 = -k_1(x_r - x)$, $u_2 = k_2 v_r sinc(\theta_r - \theta)(y_r - y) - k_3(\theta_r - \theta)$ and k_i, $i = 1, 2, 3$ are positive constant gains.

Tracking with *Controller-B* - The unicycle belongs to a subclass of the systems presented in [10], so *Controller-B* is adapted to this subclass model. In this case, the desired reference path, p_r is parametrized by a new variable γ:

$$p_r(\gamma) = [x_r(\gamma)\; y_r(\gamma)]^T \quad (10)$$

The new variable, γ assumes different roles depending on the strategy we want to use, *Tracking* or *Path Following*. In the *Tracking* strategy, we just have to set $\gamma = t$, $\gamma(0) = 0$.

As before, the error e is the difference between the desired position and the instant position of the robot, see Eq.(8), and thus depends on γ. Defining $\delta = [d\; 0]^T$ allows the specification of a constant distance, d between the robot and the target or desired position. Following the control strategy in [10], the control law is defined as:

$$\begin{bmatrix} v_d \\ \omega_d \end{bmatrix} = \Delta^{-1}(-K \tanh(e(\gamma) - \delta) + R^{\mathrm{T}} \dot{p}_r(\gamma)) \quad (11)$$

where $\Delta = diag\{1, -d\}$ and $K = diag\{K_1, K_2\}$ is a constant gains matrix. In [10] it is shown that with this feedback law, the error e converges to δ.

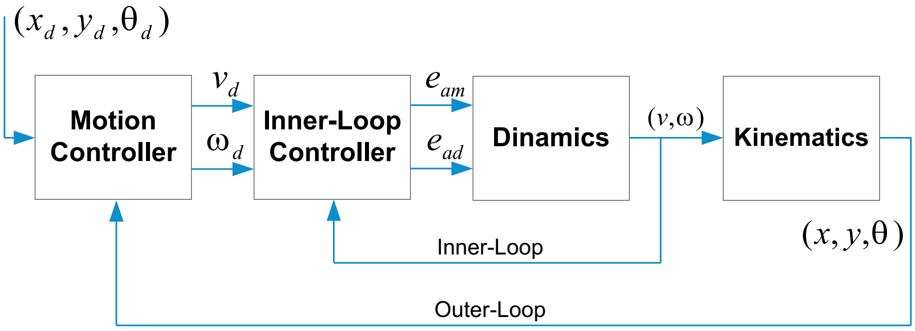


Figure 3: Model of the mobile-robot including kinematics, dynamics and the controllers. The inner loop controls the linear and angular velocities, while the outer loop regulates the motion in order to perform Tracking or Path Following.

3.3 Path Following

In path-following, the vehicle is required to converge and follow a geometric path, without a pre-defined traking law assigned to it.

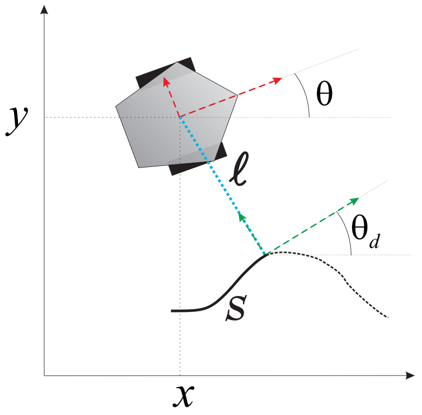


Figure 4: Illustration of the Path Following approach

Path Following with *Controller-A* - Consider Fig. 4 that illustrates the path-following strategy used by this type of controller. In this case, the linear velocity v is set to a nonzero constant value. By computing the nearest path point to the robot and the tangent vector of the path (in that point), two error variables are defined: the distance to the closest path point l and error angle $\tilde{\theta} = \theta - \theta_d$. Clearly, if l and $\tilde{\theta}$ are driven to zero, then the vehicle will follow the path. With these new variables and introducing the distance along the path s, the system motion can be described by

$$\begin{cases} \dot{s} = \frac{v\cos(\theta)}{1-c(s)l} \\ \dot{l} = v\sin(\theta) \\ \dot{\tilde{\theta}} = \omega - \frac{v\cos(\tilde{\theta})c(s)}{1-c(s)l} \end{cases} \tag{12}$$

where c denotes the curvature of the path. To simplify the notation, we introduce the auxiliary control variable u such that $\dot{\tilde{\theta}} = u$. According to [5], the feedback law is defined as $u = -k_a v l sinc\tilde{\theta} - k_b\tilde{\theta}$ where k_a and k_b are constant gains to be selected considering the linearization system's closed loop poles (a and ξ are constants) $k_a = a^2$, $k_b = 2\xi a$. The desired linear and angular velocities to drive the inner loop are thus given by

$$\begin{bmatrix} v_d \\ \omega_d \end{bmatrix} = \begin{bmatrix} K_s \\ u + \frac{v\cos\tilde{\theta}c(s)}{1-c(s)l} \end{bmatrix} \tag{13}$$

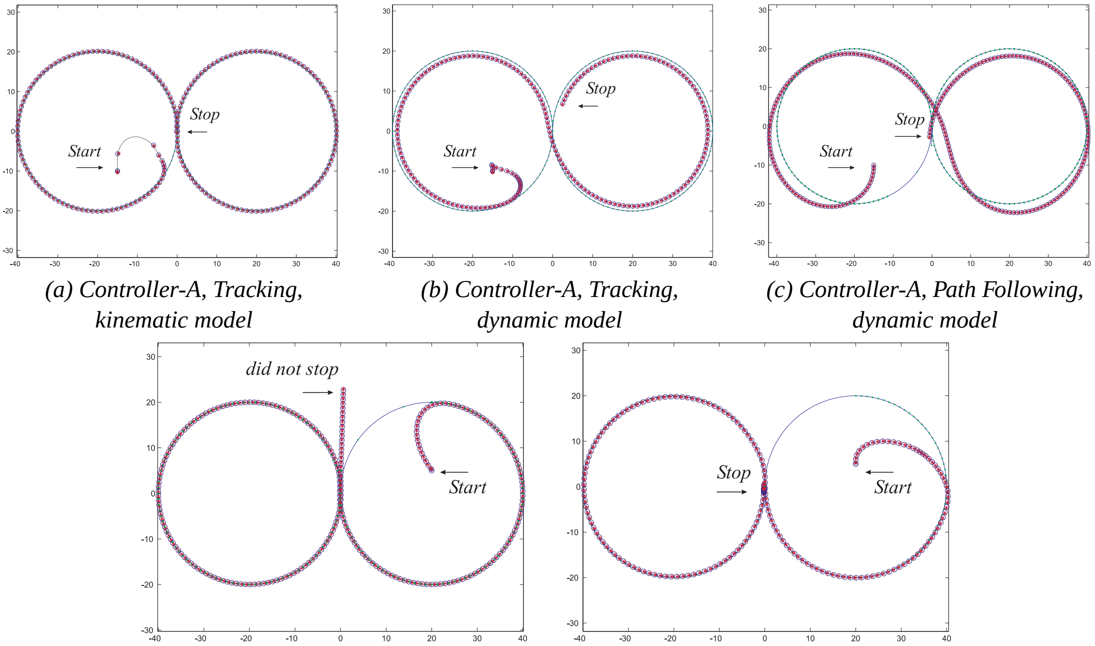
Path Following with *Controller-B* - In this case, the path-parameterization variable γ introduced in Eq.(10) is viewed as another control input. A simple option is to set $\dot{\gamma} = V_d$, where V_d is a desired constant velocity and select the initial point $\gamma(0) = \gamma_0$ to the corresponding nearest path point to the robot. In this case, $\gamma = \int_0^t V_d d\tau + \gamma_0$. Hence we obtain

$$p_r(\gamma) = p_r(V_d t + \gamma_0) \tag{14}$$

and then we use the same feedback law of Eq.(11).

4 SIMULATION AND RESULTS

To illustrate the performance of the control strategies we select a eight shaped path on a horizontal plan, centered at (0,0). Considering different initial conditions - initial robot position and orientation - some experiences were made and are represented at figure 5. Path is represented by a line and the robot is also represented by a circle with a small triangle pointing the robot's heading. The path, after defined, is discretized in order to obtain a reasonable number of points for the robot to follow. These points are obtained considering that the maximum unicycle



(a) Controller-A, Tracking, kinematic model

(b) Controller-A, Tracking, dynamic model

(c) Controller-A, Path Following, dynamic model

(d) Controller-A, Path Following, dynamic model, motion stop

(e) Controller-B, Path Following, dynamic model, motion stop

Figure 5: Kinematics Vs Dynamics in a Tracking task with Controller-A (top-row) - initial point: p_r=(-15, -10), $\theta_r = -\pi/2$). Testing the Motion Stop in a Path Following task (bottom row) - initial point: p_r=(20, 5), $\theta_r = \pi/2$).

translational velocity is 1m/s. This means that the discretization may have at least the same number of points of the path distance in meters if we want to be sure that the robot is able to follow the path.

4.1 Kinematics / Dynamics

This experience allows us to see some robot limitations.
Considering only the kinematics model, we did some simulations with a Tracking strategy using *Controller-A* - figure 5(a). We can see that the robot easily follow the path until the end. In the beginning, because the kinematics model does not impose limits at the wheels's velocities, the vehicle increases its linear and angular velocities with no boundaries. Because it is too far from the desired point - $p_r = (0,0)$, the robot increases its linear velocity and at the next iteration it is already near the path. After two more over speed iterations, the robot is tracking the path. Then, the robot follows the correct trajectory until complete the eight shape - until reaching the final point - again $p_r = (0,0)$).
With the inclusion of the dynamics model, velocities are no longer unlimited. The vehicle physical limitations are represented and no longer the robot moves with over speed. As we can see in figures 5 (b) and (c), the approximation to desired path is much slower in the beginning and also because of the limitations, is not possible to follow the path correctly, no matter what control strategy we use.

4.2 Motion Stop

One important issue about the robot motion is the motion stop. In a simple way, it's important that the robot stops when it's imposed. Using the dynamic's model for this approach, the purpose here is to test if both controllers are immune to the motion stop problem. In this case, a Path Following strategy is used. The test itself is for the robot to follow the eight shape path and stop at the last point - which is again $P_d = (0,0)$. In this case, dynamics is included and controllers are tuned to have a good response to the limitations.

Using *Controller-A*, according to[5] we were not expecting this controller to stop after the motion. And as it can be seen in figure5(d), it did not stop. Using *Controller-B*, as it can be seen at figure5(e), the robot begins to stop around the desired point, stabilizing and finally stopping near the point.

5 CONCLUSIONS

This paper revised tracking and path following strategies based on an inner-outer structure. These strategies were formulated over mathematical models for the unicycle kinematics and dynamics. The control laws designed for the outer loop were based on two controllers already presented in the literature and motion stop around a point was also considered.
Concerning only kinematics, simulations shown that both strategies had good performance tracking / following the path. With the inclusion of dynamics different behaviors were observed. The results were not so good but tunning the control gains from the control laws yield to a good performance also with the inclusion of forces, binaries and friction limitations.

While both controllers had good performances in the motion aspect, the motion-stop is not explicitly solved by *Controller-A*. *Controller-B*, after tracking / following the path, is able to stop near the desired point.

This simulator is a tool that allow us to improve control strategies: different approaches can be tested now in order to improve wheeled vehicles behavior. In the future work, tests will be performed on real robots, using proprioceptive sensing (odometry) and exteroceptive sensing (off-board cameras estimating the pose of the robot). Applications of the control strategies will involve remotely operating, parking and chaining mobile robots. In addition, the novel control strategies will enable building more intuitive and automatic human-robot interactions.

6 Acknowledgements

Research partly funded by the Portuguese FCT Programa Operacional Sociedade de Informação (POSI) in the frame of QCA III, the Portuguese FCT/ISR/IST plurianual funding through the POS Conhecimento Program that includes FEDER funds and the European Project FP6-EU-IST-045062 - URUS.

REFERENCES

[1] L Almeida. Modelização de pequenos robots autónomos: um exemplo. In *Revista do DETUA, Vol. 2, N^o 1*, pages 133–140, Aveiro, Portugal, Set 1997.

[2] Luís Almeida, Alexandre Mota, and Pedro Fonseca. Comparing control strategies for line-tracking robots. In *Proceedings of AMC'98, IEEE 5th Int. Workshop on Advanced Motion Control*, pages 542 – 547, Coimbra, Portugal, Jun/july 1998.

[3] Pedro Aparício, João Ferreira, Pedro Raposo, and Pedro Lima. Barbaneta - a modular autonomous vehicle. In *3rd IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, Portugal, 1998.

[4] Univ. Aveiro. Robotica2008 8th portuguese robotics open. `http://robotica.ua.pt/robotica2008/index-en.htm`, 2008.

[5] C. Canudas de Wit. , nonlinear control design for mobile robots. In *The International Journal of Robotics Research*, pages 121–156, France, Set 1997.

[6] RoboCup federation. Robocup official site. `http://www.robocup.org`, 2008.

[7] iBOT® Mobility System. `http://www.ibotnow.com/`.

[8] EU publications office. Community research and development information service, seventh research framework programme (fp7), search projects page. `http://cordis.europa.eu/fp7/projects_en.html`, 2008.

[9] Alberto Sanfeliu and Juan Andrade-Cetto. Ubiquitous networking robotics in urban settings. In *Proc. IEEE/RSJ IROS Workshop on Network Robot Systems*, pages 14–18, Beijing, China, 2006.

[10] Francesco Vanni, A. Pedro Aguiar, and António M. Pascoal. Cooperative path-following of underactuated autonomous marine vehicle with logic-based communication. In *Proc. of NGCUV'08 - IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles*, pages 1–6, Lisbon, Portugal, Apr 2008.