Control Engineering Practice 18 (2010) 318–329

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

A linear-interpolation-based controller design for trajectory tracking of mobile robots

Gustavo Scaglia [a,b,*], Andrés Rosales [b], Lucia Quintero [b], Vicente Mut [b], Ravi Agarwal [c]

[a] Instituto de Ingeniería Química, Universidad Nacional de San Juan, Argentina
[b] Instituto de Automática, Universidad Nacional de San Juan, Argentina
[c] Department of Mathematics, Florida Institute of Technology, USA

ARTICLE INFO

Article history:
Received 26 August 2008
Accepted 20 November 2009
Available online 23 December 2009

Keywords:
Control system design
Nonlinear model
Tracking trajectory control
Mobile robot
Numerical methods

ABSTRACT

This work presents a novel linear interpolation based methodology to design control algorithms for the trajectory tracking of mobile robotic systems. Particularly, a typical nonlinear multivariable system—a mobile robot—is analysed. The methodology is simple and can be applied to the design of a large class of control systems. Simulation and experimental results are presented and discussed, demonstrating the good performance of the proposed methodology.

1. Introduction

One of the main problems found in mobile robot control is trajectory tracking. In general, the objective is that the mobile robot can reach a prescribed Cartesian position (x, y) with a pre-established orientation θ for each sampling period. These combined actions result in tracking some desired trajectory of the mobile robot. In order to achieve this objective, commonly two control variables are available, namely: robot's linear velocity V and the angular velocity W.

The use of trajectory tracking for a navigation system is justified in structured workspaces as well as in partially structured workspaces, where unexpected obstacles can be found during the navigation. In the first case, the reference trajectory can be set from a global trajectory planner. In the second case, the algorithms used to avoid obstacles usually re-plan the trajectory in order to avoid a collision, generating a new reference trajectory from this point on. Besides, there exist algorithms that express the reference trajectory of the mobile robot as function of a descriptor called r (Del Rio, Jiménez, Sevillano, Amaya, & Balcells, 2002) or s (called "*virtual time*") (Lee & Park, 2003) whose derivative is a function of the tracking error and the time t. For example, if the tracking error is large, the reference trajectory should wait for the mobile robot; otherwise, if the tracking error is small, then the reference trajectory must tend to the original trajectory calculated by the global planner. In this way, the module of trajectory tracking will use the original path or the on-line recalculated path as reference to obtain the smallest error when the mobile robot follows the path (Normey-Rico, Gomez-Ortega, & Camacho, 1999). Therefore, the trajectory tracking is always important independently from whether the reference trajectory has been generated by a trajectory global planner or a local one.

Various control strategies have been proposed for trajectory tracking, some of which are based on either kinematic or dynamic models of the mobile robot (Lee, Song, Lee, & Teng, 2001), depending on the operative speed and the precision of the dynamic model, respectively (Do & Pan, 2006). Different structures to control these systems have been developed as well. In Tsuji, Morasso, and Kaneko (1995), the authors used a time-varying feedback gain whose evolution can be modified through parameters that determine the convergence time and the behaviour of the system. In Fierro and Lewis (1995), the controller proposed by Kanayama, Kimura, Miyazaki, and Noguchi (1990) is used to generate the inputs to a velocity controller, making the position error asymptotically stable. So, a controller to force the velocity of the mobile robot to follow the reference velocity is designed. The work of Fukao, Nakagawa, and Adachi (2000), extends the design proposed by Fierro and Lewis (1995) and considers that the model parameters are unknown. In Kim, Shin, and Lee (2000), an adaptive controller that takes into account the parametric uncertainties and the robot external perturbations to guarantee perfect velocity tracking is proposed. The reference

* Corresponding author at: Instituto de Ingeniería Química, Universidad Nacional de San Juan, Argentina. Tel.: +54 264 421 3303; fax: +54 264 421 3672.
E-mail addresses: gscaglia@unsj.edu.ar (G. Scaglia), arosales@inaut.unsj.edu.ar (A. Rosales), olquinte@inaut.unsj.edu.ar (L. Quintero), vmut@inaut.unsj.edu.ar (V. Mut), agarwal@fit.edu (R. Agarwal).

0967-0661/$ - see front matter
doi:10.1016/j.conengprac.2009.11.011

velocity is obtained by using the controller proposed by Kanayama et al. (1990). In Chwa (2004) two controllers are designed; they are called position controller and heading controller. The former ensures position tracking and the latter is activated when tracking error is little enough and tracking reference does not change its position. This reduces the error over the mobile robot orientation at the end of the path. In Shim and Sung (2004) the posture controller is designed in function of posture error and, in this way, the reference velocities are generated on the basis of a specification set as: (i) if the distance to a reference posture is relatively large, then the robot movement is quick, and the speed is reduced as the robot approaches to the target; (ii) the robot should take the shortest amount of time to reach the desired posture. Later, the reference velocities input a PID controller that generates the torque needed by the desired speed. In Sun and Cui (2004), a controller for trajectory tracking is designed using the kinematic model of the mobile robot and a transformation matrix. Such matrix is singular if the linear velocity of the mobile robot is zero. Therefore, the effectiveness of that controller is ensured only if the velocity is different from zero. Simulation results using linear velocity different from zero as initial condition are shown in that paper. In Sun (2005) a controller based on the error model of Kanayama et al. (1990) is proposed. This controller is formed by two equations which are switched depending on the value of the angular velocity of the mobile robot and the prescribed tolerance of it. In Martins, Celeste, Carelli, Sarcinelli-Filho, and Bastos-Filho (2008) an adaptive controller used to guide a mobile robot during trajectory tracking is proposed. Initially, the desired values of the linear and angular velocities are generated, considering only the kinematic model of the robot. Next, such values are processed to compensate the robot dynamics, thus generating the commands of linear and angular velocities delivered to the robot actuators.

The trajectory tracking for mobile robots is characteristically a nonlinear problem. Diverse model-based classic techniques, which propose controllers with a zero-error tracking, have been applied to solve this problem. However, these classic approaches involve an online matrix inversion (e.g. Klanèar & Škrjank, 2007; Vougioukas, 2007), which represents a drawback in the implementation of the aforementioned methods. In this paper, the proposed algorithm does not involve online matrix inversion problems. Most surveys do not present a final expression for the control signals of their controllers (e.g. Liu, Jing, Ding, & Li, 2008; Tsai, Wang, Chang, & Wu, 2004; Wang & Tsai, 2004), because the computation of these control variables must be made by using demanding computer operations. On the other hand, some current straightforward methods present just simulations (Dong & Guo, 2005; Liu, Zhang, Yang, & Yu, 2004; Zhang, Dai, & Zeng, 2007). In this paper, the design of the proposed control law by using linear algebra tools and furthermore the final expression for the control signals, which will be directly implemented on the mobile robot, are presented.

In this paper, a simple approach to track trajectories is proposed. To achieve this goal, it is assumed that the evolution of the system can be approximated by a linear interpolation in each sampling time. Under this assumption, and knowing the desired state, a value for the control action needed to force the system to go from its current state to a desired one can be obtained. As it is a linear approximation, clearly, the tracking errors can be reduced by decreasing the sampling time. The main contribution of this work is that the proposed methodology is based upon easily understandable concepts, and that there is no need for complex calculations to attain the control signal. In this work the control schemes presented in Fierro and Lewis (1995), Fukao, Nakagawa, and Adachi (2000), Kim et al. (2000), Shim and Sung (2004), Cruz, Mcclintock, Perteet, Orqueda, and Cao (2007) will be employed. Accordingly, a kinematic controller is designed first, which generates the reference velocity in order to reach the desired goal. Additionally, this reference is employed to input a velocity controller in the scheme. In our work a PID controller is used as a velocity controller. The implemented controller will be placed on board an existing mobile robot in order to maintain its translational and rotational speeds at desired values (Cruz et al., 2007; Shim & Sung, 2004). In general, most market-available robots have low level PID velocity controllers to track input reference velocities and do not allow the motor voltage to be driven directly. Therefore, it is useful to express the mobile robot model in a suitable way by considering rotational and translational reference velocities as control signals. Furthermore, the velocity controller follows asymptotically its references.

Besides, in this work it is not necessary to switch the controller as in Chwa (2004) in cases when position reference does not change and tracking error is small. The purpose of this paper is that when this situation is detected, the desired orientation changes, calculating the control signal by using the same expression. Additionally, this approach does not suffer from the disadvantage of the controller by Sun & Cui (2004), where a linear velocity different from zero is necessary for good working. Furthermore, this controller does not need to change the control expression when the angular velocity is lower than a pre-established value in contrast to Sun (2005). In addition, Scaglia, Quintero, Mut, and di Sciascio (2008) introduce a numerical methods based controller, where the control law depends on the chosen numerical approximation. In the current work, very-low tracking errors are obtained, taking into account that linear and angular velocities are not considered during the controller design. Besides, a proof of the zero-convergence of the tracking error is also included.

The paper is organized as follows: Section 2 presents the methodology for the controller design using a linear interpolation method and describes the application of the methodology to a multi-variable nonlinear system. This is illustrated in a case study involving a mobile robot PIONEER 2DX. Afterwards, the formulation of the proposed control algorithm is obtained. In Section 3 the results and a further discussion about the tracking, positioning and the movement of the mobile robot into a real environment are presented. Conclusions are detailed in Section 4 and finally, Appendix A refers the demonstration of the error tendency to zero.

2. Methodology for controller design for a mobile robot

Linear interpolation is the basis for many numerical methods. Let us consider Fig. 1 and a smooth nonlinear function $y=g(x)$ in the interval $[a, b]$ then, a linear interpolation is valid:

$$P(x) = \frac{g(b)-g(a)}{b-a}(x-a)+g(a) \tag{1}$$

The interpolation error $e(x)=g(x)-P(x)$ is bounded and it is valid

$$|e(x)| \leq \tfrac{1}{2}|(x-a)(x-b)| \max_{a \leq x \leq b} |g''(x)| \tag{2}$$

Based on the continuity and smoothness of $g(x)$ and assuming a sufficiently small interval $[a, b]$, one gets that the maximum of $g''(x)$ occurs approximately at the middle point $x_m = 0.5(a+b)$, and if the second derivative remains approximately constant in the interval, then it can be approximated by $g''(x_m)$. In this paper, these results will be used to obtain the control for reference tracking. The methodology used will be described next. Let us