DOI: 10.2478/aucts-2019-0004

ACTA UNIVERSITATIS CIBINIENSIS – TECHNICAL SERIES
Vol. 71 2019

POSITION CONTROL OF A MOBILE ROBOT THROUGH PID CONTROLLER

Alexandru BÂRSAN
Faculty of Engineering/ Department of Industrial Machines and Equipment, "Lucian Blaga" University, Sibiu, Romania, alexandru.barsan@ulbsibiu.ro

Abstract: *In this paper, the implementation and tuning steps of the PID controllers were proposed for the position control of a mobile robot with differential steering. The purpose of this robot was the participation in a European amateur robotics contest, the mobile robot being developed in the Engineering Faculty of Sibiu, Department of Industrial Machines and Equipment. After a brief introduction of the mobile entertainment robot, followed by description of the robot's different components and traction principles, several principles of tuning the PID controller for the mobile robot are presented. The paper ends with some general conclusions based upon the results obtained from studying the research and considering all the ideas that were introduced.*

Key words: Differential wheeled robot, Position control, Odometry, PID tuning

1. Introduction

In the past decade researches within mobile robotics have been expanding rapidly, especially due to their numerous advantages over their disadvantages regarding the workplace. It's well known that the mobile robots are nowadays used within different industries and for many purposes. In order to develop the technologies necessary to obtain mobile robots that support or substitute certain operations performed by humans, there is a need to combine several fields such as artificial intelligence, sensors, motor control, signal processing, computing systems, trajectory planning, electronics and computer science.

The robot can successfully perform its tasks depending on the knowledge it has about the initial configuration of the workspace, but also the ones obtained during its evolution [1]. There are certain problems that arise in mobile robots, such as: determining the position and orientation on the environment, avoiding the impact with different objects, planning an optimal movement path. The robot navigation is influenced by several methods, such as measuring the number of rotations made by the motor wheels, using gyros and accelerometers, but usually determines the pose of the robot in relation to a fixed coordinate system.

When developing an autonomous mobile robot, to carry out the specific navigation tasks, the robot must be equipped with a suitable locomotion system. But the mobile robot would be nowhere near as effective, if it were not supported by an adequate control system. For that it is proposed a closed loop control by using a PID controller that allows adjustment of the speed of the brushless DC motors. The reaction system is ensured through two rotary incremental encoders.

2. General aspects of the mobile robot

Two categories of the service robots that are experiencing a technological revolution are scientific and event robots [2]. A further emerging market is that of mobile entertainment robots developed for amateur robotics contests that put an emphasis on an active learning process. The European robotic contest highlights values such as fair-play, knowledge sharing and creativity, leading to the technical challenge to build an autonomous robot [3].

The necessary steps before the construction of the autonomous mobile robot, due to the need to respect the overall dimensions required, are represented in a 3D model of the mechanical design of the robot. Starting from certain specific aspects that the robot has to meet, such as overall dimensions: the undeployed perimeter at the beginning of the match is 1200 [mm], and the height of robot must not exceed 350 [mm], the decisive role in achieving the tasks at hand and a good functioning of the mobile robot is played by the locomotion system. Both the locomotion type (wheel, track, foot, etc.) and the structural aspects (degrees of mobility, orientation, maneuverability) need to be considered in order to achieve the desired final position.

Build on an orthogonal chassis, the robot was developed on several levels, having a total mass of 10 [Kg] and reaching a maximum speed of 2 [m/s]. Power supply is achieved through a 22.2 [V] LiPo battery with a capacity of 7 Ah. A development board Arduino Mega 2560 is used as the controller of the represented robot. Considering the very low power output from the Arduino ports, that are insufficient to spin the DC brushless motors, an ESCON Module 50/5 was used, 4-Q Servo controller for DC/EC motors, 5/15 [A], 10 - 50 [VDC].

Figure 1: Autonomous Mobile Robot CyberTech

2.1 Differential-drive system

The mechanical structure of the robot is based on a differential-drive configuration consisting of two wheels placed on a common horizontal axis and fixed relative to the robot chassis, which are independently controlled by two DC brushless motors, and one-rear-caster wheel (fig. 2). The navigation strategy for this differential drive robot consist in three steps: the movement in a straight line, rotation around a fixed axis, and then moving straight again. In order to make a translation movement, the two traction wheels of the robot must be operated in the same direction. To achieve a rotational motion, the two driven wheels of the robot must be operated in different rotational directions. To establish the momentary position of the robot the navigation method used is odometry [4]. It uses the data (impulses) from two Kübler incremental rotation encoders with 4096 pulses per revolution (PPR) that it translates into linear displacements relative to the surface it is running on.

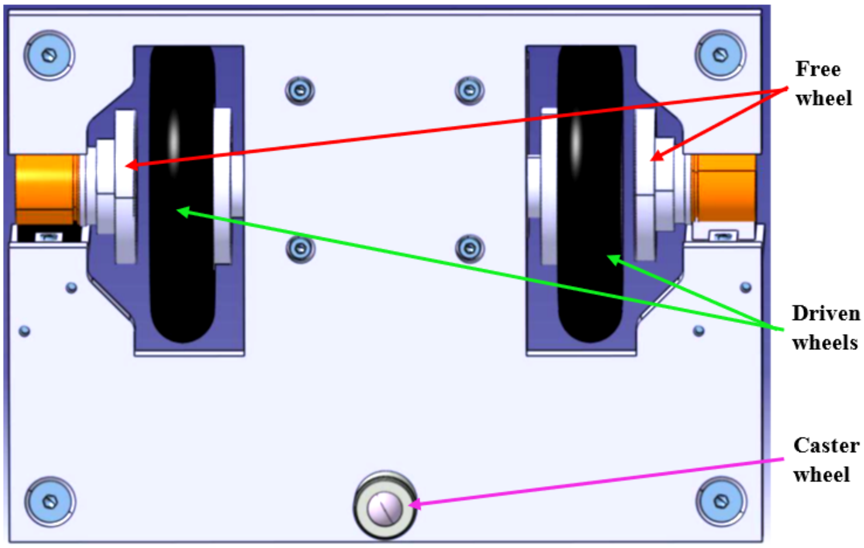

Figure 2: Mobile robot differential-drive system

The schematic representation of the differential-drive is shown in figure 3. The position of the mobile robot in the global frame {X1, O1, Y1} can be defined according to 'A' which signifies the vehicle's center of mass, and 'ϕ' which is the orientation.

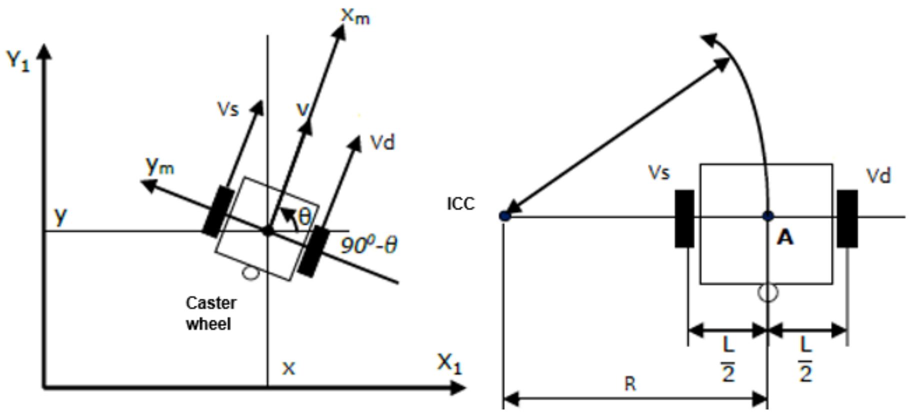

Figure 3: Free-body diagram of the differential-drive

The speeds of the two wheels must respect the following relationships, where:

- v represents robots linear speed;
- ω represents robots angular speed;
- r represents the traction wheel radius;
- $L/2$ represents the distance between either driving wheel and the axis of symmetry;
- ω_r, and ω_l are the angular speeds of the right and left wheels;
- v_r and v_l are the linear speeds of the right and left wheels;
- R is the signed distance from the Instantaneous Center of Curvature (*ICC*) to the midpoint between the wheels.

$$\begin{cases} \omega\left(R+\frac{L}{2}\right)=v_r \\ \omega\left(R-\frac{L}{2}\right)=v_l \end{cases} \quad (1)$$

$$\omega=\frac{v_r-v_l}{L} \quad (2)$$

$$R=\frac{l(v_r+v_l)}{2(v_r-v_l)} \quad (3)$$

From equation (3) the significance of the two cases of movement can be determined as:

- Linear motion in a straight line: the radius R tends to infinity, and the speeds are equal ($v_l = v_r$) and have the same meaning.
- Rotational motion (curve) - radius R has finite value. The pivoting motion around point A is realized if R tends to zero, and the speeds are equal and have different direction.

By choosing the speeds v_l and v_r you can reach a multitude of different points. The kinematic equation (4) describes the rotational motion of the robot at a distance R about its ICC with an angular velocity of ω.

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} \cos(\omega\delta t) & -\sin(\omega\delta t) & 0 \\ \sin(\omega\delta t) & \cos(\omega\delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - x_{ICC} \\ y - y_{ICC} \\ \theta \end{bmatrix} + \begin{bmatrix} x_{ICC} \\ y_{ICC} \\ \theta\delta t \end{bmatrix} \qquad (4)$$

Thus, by integrating the above equation, starting from a set of initial conditions (x_0, y_0, θ_0), the position of the robot at time t can be determined, based on the control parameters v_r (t) and v_l (t). The inverse kinematic equations of the two-wheeled mobile robot are:

$$x(t) = \frac{1}{2}\int_0^t [v_d(t) + v_s(t)] \cdot \cos[\theta(t)]dt \qquad (5)$$

$$y(t) = \frac{1}{2}\int_0^t [v_d(t) + v_s(t)] \cdot \sin[\theta(t)]dt \qquad (6)$$

$$\theta(t) = \frac{1}{2}\int_0^t [v_d(t) - v_s(t)]\, dt \qquad (7)$$

Since solving the above system of equations is cumbersome, two special cases of robot movement with differential locomotion system are proposed.

In the first case, if v_l is equated with $v_r = v$, the above equation becomes:

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x + v\cos(\theta)\delta t \\ y + v\sin(\theta)\delta t \\ \theta \end{pmatrix} \qquad (8)$$

In the second case, if $v_l = - v_r = v$, the equation becomes:

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta + 2\frac{v\delta t}{l} \end{pmatrix} \qquad (9)$$

3. PID controller implementation

Since when it was desired to move the robot to a certain point on the playing surface, from the moment when the desired position was introduced and until it was changed by the closed loop adjustment, a certain latency occurred. This latency was materialized by the fact that the robot did not stop at the desired location and continued to move for a short time. Because of this, there were oscillations around the desired location, manifested by moving the robot forward and backward to the desired reference point. Therefore, to eliminate this latency, as well as to fine-tune the output quantities (speed, torque) of the brushless DC motor, the PID (proportional-integrator-derivative) controller was used. The simple structure and implementation, overall good control performance and robust design are a several advantages of the PID control [5, 6].

The PID algorithm consists of three tuning parameters that can be interpreted, with respect to time, in the following way:

- P – proportional term, is used in calculating current errors;
- I – integral term, provides information about the amount of previous errors;
- D – derivative term, provides a prediction of future errors, based on the current size fluctuation rate.

In figure 4 is shown the adjustment scheme of the locomotion system using the PID controller.

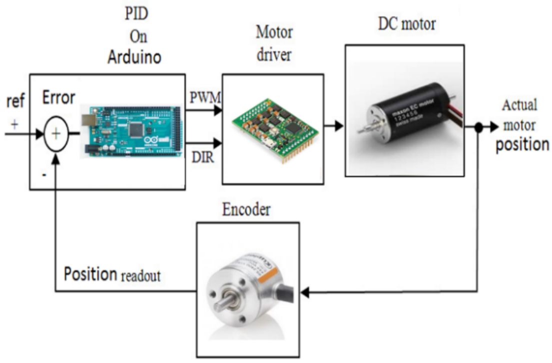

Figure 4: Closed loop of PID implementation

The PID controller implementation allows the speed adjustment of DC brushless motors and is done using the digital signal processor of the Arduino board. The reaction of the system is ensured by the two Kübler encoders, which provide at the output of 4096 PPR. Because the robot was programmed to reach a certain point on the playing surface by entering the position value, it was necessary to use two sets of PID constants, one set for movement (translation) and one for orientation (rotation).

Since the most important requirement for engines was to reach the commanded position in a relatively short time, the stationary error of angular velocity must be less than 1%. Of course, this is not enough, because the engines must accelerate to the stationary speed as fast as possible, almost simultaneously with their power on. However, an angular velocity much higher than the nominal one can cause engine failure, so we imposed an over-regulation of less than 5%.

Firstly, for an imposed reference position of 1000 mm on a forward direction, the PID controller was implemented and tuned through MATLAB software (fig. 5). After creation of the Simulink model based on the CAD model of the robot created in Catia V5, in order to reproduce the dynamic model of the robot, it was observed that the mobile robot presented a settling time of 4 [s]. Not being satisfied with this result, a second attempt was tried using a different method.

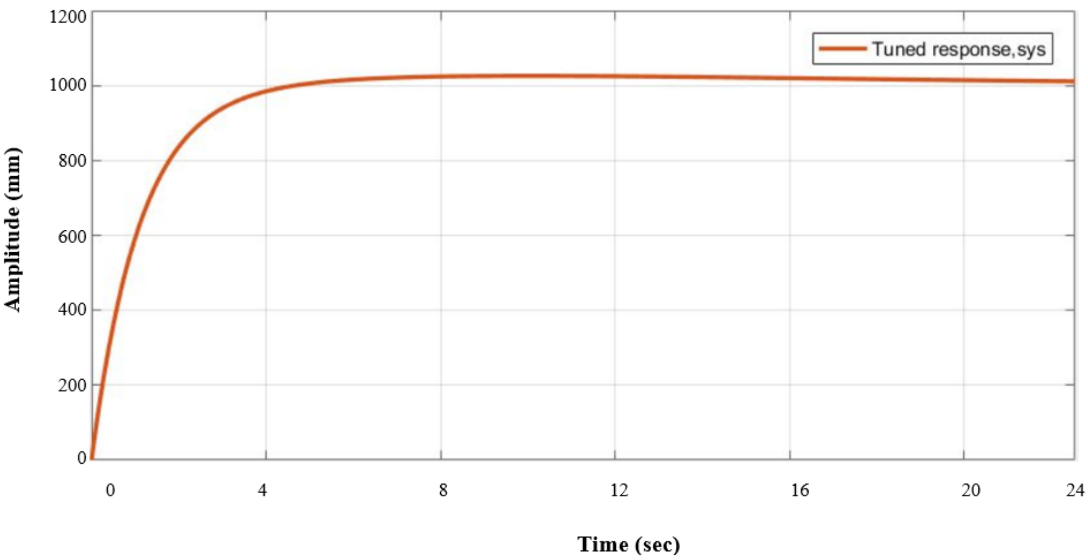

Figure 5: Results after PID implementation in MATLAB $K_P = 1.2$, $K_D = 0.15$

In order to improve the optimal values for the desired control response the Ziegler-Nichols heuristic method was implemented, which implies the initial equalization of the K_I and K_D coefficients with 0, and then the gradually increase the K_P coefficient value until it attains the ultimate gain K_u before the robot oscillates with a constant amplitude [7, 8]. Therefore, when trying to adjust the robot, the K_P value was increased until the robot started to go back and forth to the desired reference point. At this point it has reached the point where the robot does not behave properly. Once the K_P value was established, K_u and the oscillation period T_u were used to set the PID gains as shown:

Table 1: PID constants

Type	K_P	K_I	K_D
Movement	1.2	0	0.15

Orientation	4	0	0.32

4. Conclusions

The differential-drive was used because it offers a great possibility of changing the direction in restricted spaces as well as a reduction in overall dimensions of the robot. However, the main drawback is that the differential locomotive system introduces robot positioning and orientation errors due to the different diameters of the traction wheels, but also because of the imprecise robot assembly. To eliminate this disadvantage, the odometry as a navigation method was used. A pair of free wheels with incremental rotation encoders and a systematic calibration to reduce systematic errors of odometry was implemented.

In order to improve the control over the locomotion system, mainly in the closed loop, the implementation of a PID controller was necessary in order to manage and fine-tune the outputs of the brushless DC motors. After several experimental trials, result have shown that the second method yielded better results, where the implementation does not need an integral factor. The application requires only two parameters K_P, K_D, to provide the appropriate control over the system since the value of K_I is 0. Therefore, the PID will be referred as PD and for most of the experiments the parameters used are: K_P=1.2, K_D=0.15 for movement and K_P=4, K_D=0.32 for orientation, as a result the overshoot of the robot came down to 3 [mm] in a settling time of 3 [s].

The best results after the PID controller tuning are presented in Figure 6.

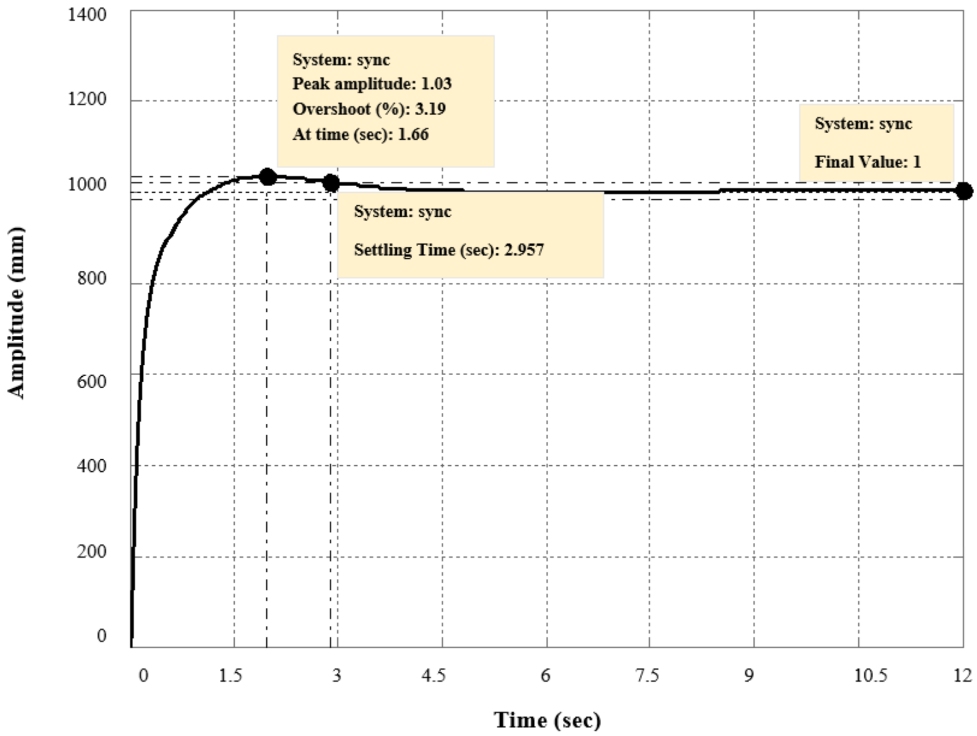

Figure 6: Result obtained for K_P=1.2, K_I=0, K_D=0.15

5. References

1. Fukao, T., Nakagawa, H., Adachi, N., Adaptive tracking control of a nonholonomic mobile robot, *IEEE transactions on Robotics and Automation*, Vol. 16, No. 5, pp. 609-615, (2000).
2. Ciupe, V., Maniu, I., New trends in service robotics, In *New trends in medical and service robots, Springer*, Cham, pp. 57-74, (2014).
3. Crenganis, M., Bologa, O., Developing a mobile platform for EUROBOT 2015, *Buletinul AGIR. Supliment*, Vol. 1, pp. 230-235, (2015).
4. Borenstein, J., Feng, L., Measurement and correction of systematic odometry errors in mobile robots, *IEEE Transactions on robotics and automation*, Vol. 12, No. 6, pp. 869-880, (1996).
5. Tan, K. K., Wang, Q. G., Hang, C. C., *Advances in PID control*, Springer Science & Business Media, (2012).

6. Crenganis, M., Bologa, O., IMPLEMENTING PID CONTROLLER FOR A MOBILE PLATFORM, (2015).
7. Crenganis, M., Breaz, R. E., Racz, G., Bologa, O., Zigler-Nicols PID Tuning Method for Position Control of a Mobile Robot, *Applied Mechanics and Materials*, *Trans Tech Publications*, Vol. 841, pp. 221-226, (2016).
8. Ang, K. H., Chomg, G. C., PID Control System Analysis and Design. Problems, remedies and future directions. *IEEE Control Systems Magazine*, pp. 32-41, (2006).