ATLANTIS PRESS

Advances in Engineering Research, volume 208

Proceedings of the International Conference on Innovation in Science and Technology (ICIST 2020)

Modelling and Position Controlling of Two Wheeled Mobile Robot

B. Supriyo*, A. Suharjono, K. W. Atmaja
Magister Terapan Teknik Telekomunikasi
Politeknik Negeri Semarang
Semarang, Indonesia
*bsupriyo7763@yahoo.com

***Abstract*—This paper deals with the simulation works for modelling two DC motors and controlling the path position of two wheeled mobile robot based on odometry to track robot movements. The mathematical models of the DC motors which are the main actuators for the robot are obtained by using Matlab System Identification Toolbox. This identification process acquires observed data consisting of input voltage and output speed of each motor obtained during open loop experiment. The open loop response of the resulted mathematical model of the motor is then compared to that of its respective original motor. The position controller of the robot has both inner loop speed feedback and outer loop position feedback. The Proportional, Integral and Derivative (PID) controller is utilized in speed controller part to regulate the speed of the motor, while the Proportional controller is used in position controller part to track the position of the robot. The orientation angle of the robot is calculated using odometry formula. The difference between the orientation angle of the robot and the target angle is called the orientation angle error. This error is used to adjust the speed of each motor such that the robot can be moved straight to the target point.**

***Keywords*—two-wheeled mobile robot, Matlab system identification toolbox, and motor open loop experiment**

I. INTRODUCTION

Wheeled mobile robot (WMR) is a kind of robot in which its movement is based on driver wheels commonly actuated using electrical motors. Recently, wheeled mobile robot has been used in many applications. In dealing with the covid-19 cases, different wheeled mobile robots have been deployed, such as for supporting healthcare staffs for taking care of the covid-19 patients with less contact and helping healthcare delivery [1], avoiding close contact between healthcare staffs with patients when performing oropharyngeal swab (OP swab) [2], monitoring social distancing between humans in crowded area [3], and performing various logistical tasks and disinfection of indoor areas using the UV-C light in both industrial and civil environments [4]. The number of wheels used in mobile robot mainly depends on its robot design, weight and application. The wheeled mobile robots can have two wheels [5-8], three wheels [9-11], four wheels [12-14] and six wheels [15-17].

Mobile robots require a navigation system to guide the robot in moving from one location to another. Robot navigation commonly uses path planning system, where the robot uses odometry-based position sensors and direction sensors to guide the robot orientation, so that the robot knows its position at all times. The common sensor used in the odometry system is an encoder installed on the wheel axle [18].

The type of drive that is commonly used is a two-wheel differential drive equipped with a free wheel (castor) for balancing the robot [19-20]. The movement of this mobile robot is based on two separately driven wheels placed on either side of the robot body. Each wheel is actuated by a DC motor. Its steering motion is mainly performed by controlling the rotation of the two wheels. This mobile robot with a differential drive system is categorized as a robot with a non-holonomic mechanism [7], because the limited movement of the robot which only moves in the same direction as that of the wheel rotation.

To control the movement of the robot, an Arduino Uno microcontroller is needed. This microcontroller board is based on ATMega328. In processing the motor identification system, Arduino Uno is used to obtain experimental data on open loop motor, namely by actuating the motor, reading both the motor input voltage and the motor shaft rotational speed. In this case the Arduino is programmed using the Arduino IDE software. Some of the applications using the Arduino Uno are for teaching mobile robots [21] and indoor position tracking systems [22].

Simulation works can be used as a viable solution to understand the behavior of the robot movements in advance before the robot is tested on the real track. Some simulation works dealing with the movement of the wheeled mobile robot [19]. This paper focuses on the simulation works to control the movement of the robot such that it moves straight to the target position with minimum position error. The simulation works deal with two different DC motors that drive the robot's wheels. The mathematical model of each motor is obtained using Matlab's System Identification Toolbox, by first obtaining both the voltage input and the speed output of the motor during open loop experiments. During this experiment, the two wheels of the mobile robot are attached on a dummy free wheels acting as a roller dynamometer which represents the real track of the robot. The Proportional Integral Derivative (PID) based position controller for each DC motor is designed and implemented through simulation using Matlab/Simuilink. Its position controller performance is observed based on its steady state error of the robot orientation angle.

II. System Description

In this study, the two wheeled mobile robot is designed and developed to have two driving wheels installed on the right and left sides of the center of the robot body and two free wheels (castors) attached on the front and back sides of the robot body. Each wheel is actuated by a DC motor (MOT-R, MOT-L) equipped with two channel encoders (ENC-R, ENC-L). The DC motor used in this robot has a maximum speed of 100 rpm (12 Volt DC), with a speed reducer gearbox of 74.83 and an encoder of 11 pulse-per-revolutions (pprs), so that the total pulses per one rotation of the motor output shaft is 823.1 pulses. Each of the robot motor is driven by an H-bridge motor driver (DRV-R, DRV-L) which acts as interface unit between the microcontroller and the DC motor. An Arduino Uno microcontroller is used as a main controller for this robot system. The microcontroller reads the encoder sensors, controls the dc motor, and transfers the sensor data to the computer via USB port. The computer is used to program the microcontroller, to log the sensor data from microcontroller, to carry out the motor identification process using Matlab's System Identification toolbox and to run control simulation using Matlab/Simulink. The block diagram of this system can be seen in figure 1.

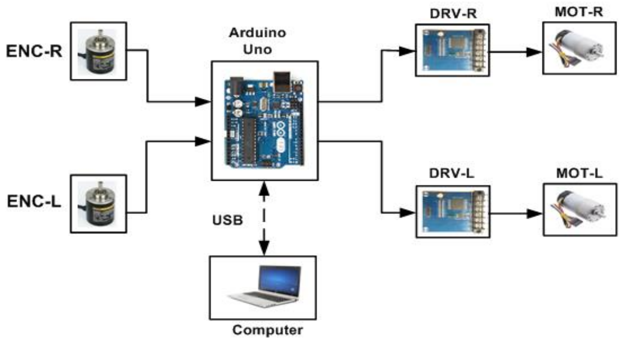


Fig. 1. Block diagram of the electronic system.

The simple test rig for the two wheeled mobile robot is developed. It consists of two free dummy wheels, where the two robot wheels attach on them and transmit the speeds. The test rig also has four linear slide bearings attached on its vertical lead rods which ensure the robot moves freely on vertical axis and the full load robot is fully supported on the wheels' test rig. By assuming that no slip will occur between the robot wheels and the dummy wheels, the speed of the robot will be determined using the encoder sensors attached on the motors. This test rig represents the robot track, so it gives the flexibility for the user to modify and test the program for the robot before it runs on the real track. This test rig can be seen in figure 2.

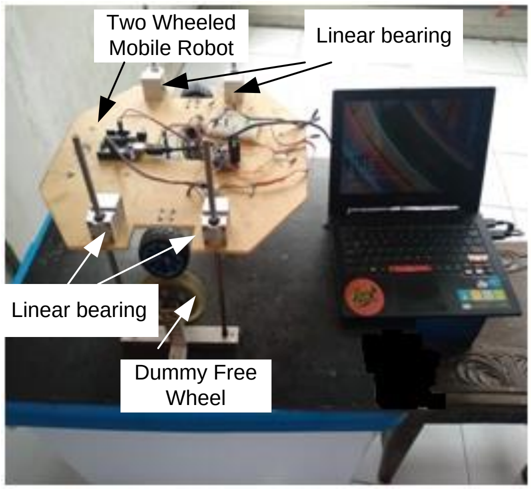


Fig. 2. Two wheeled mobile robot test rig.

III. Methods

Each DC motor of the robot is tested in open loop experiment to obtain the observed data for system identification process. In this case the microcontroller triggers the input of the DC motor with a 12-Volt DC voltage, reads the speed of DC motor via encoder and transfers the data to computer via USB serial port. The mathematical model of the DC motor is then performed by utilizing System Identification Tool box in Matlab, as seen in figure 3, by selecting the second order transfer function as its output result. This process imports the observed data file consisting of time, input voltage of the DC motor and output speed of the dc motor shaft. This resulted transfer function represents the mathematical model of its respective motor. In simulink environment, this transfer function is then tested in open loop mode by using the 0-to-12V unit step input. The output speed response of this test is compared to that of the real dc motor open loop experiment.

Based on this DC motor transfer function, the PID based speed controller is designed and implemented in Matlab/Simulink with the speed set point of 100 rpms. The PID parameters of this speed controller are finely tuned such that its output speed response results in no overshoot and no steady state error. The Proportional (P) position controller is then constructed based on this PID speed controller, in which the output of P position controller is used as set point for the speed controller. There are two feedbacks implemented on this position controller. The inner feedback is the speed feedback for the speed controller, while the outer feedback is a position feedback for the position controller. The output of position controller is a distance travelled. The motor output speed is a rotational speed (ω) in rpm. If the radius of the robot wheel (r) is known, then its linear distance travelled (d) for time (t) can be determined using equation (1). This distance information is used in position feedback for position controller. The P parameter of this position controller is finely tuned such that its output position response results in no overshoot and no steady state error. The various set points of 100 cm. 300 cm and 500 cm are selected to test the performance of the P position controller for both right and left motors.

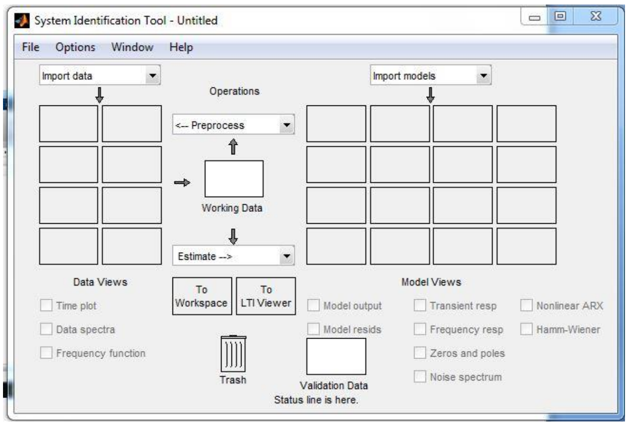


Fig. 3. System identification tool window.

$$d = (\pi/30)\,\omega\, r\, t \qquad (1)$$

Where d [cm], ω [rpm], r [cm] and t [second]. The independent movements of both right and left motors might cause an error in robot orientation angle, if the distance travelled performed by each motor is not the same for every sampling time. Figure 4 shows the illustration of the robot's orientation angle (α) during movement from position P_0 to P_1 with the average distance travelled of d_T, which can be calculated using equation (2) and the orientation angle can be determined using equation (3). The center position of the robot is initially at P_0. The global horizontal axis is X_G and vertical axis is Y_G. During moving, the robot horizontal axis is X_R and vertical axis is Y_R.

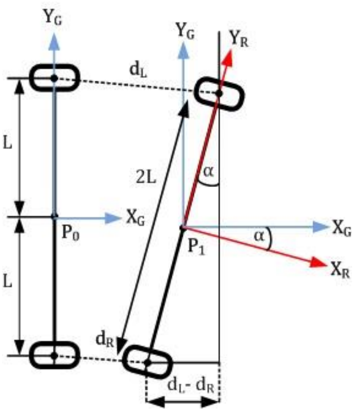


Fig. 4. Angle of orientation.

$$d_T = (d_L + d_R)/2 \qquad (2)$$

$$\alpha = (d_L - d_R)/2L \qquad (3)$$

Where d_L and d_R are distance travelled by left motor and right motor, respectively, and L is the distance between the motor and the center point of the robot.

IV. Results and Discussion

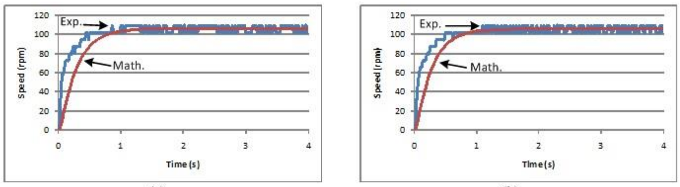
The mathematical models of both left and right DC motors for this two wheeled mobile robot, in the form of transfer functions, which are obtained from system identification process using System Identification Toolbox in Matlab are represented in equation (4) and (5), respectively.

$$y_1(s) = \frac{843.54}{s^2 + 28.8\,s + 95}\, u_1(s) \qquad (4)$$

$$y_2(s) = \frac{849}{s^2 + 28.1\,s + 96.2}\, u_2(s) \qquad (5)$$

Where y_1 and y_2 represent speed output of the left and right motor respectively, u_1 and u_2 represent voltage input of the left and right motor respectively.

The open loop responses of both the experimental and mathematical model of each DC motor are shown in figure 5. In figure 5(a), for the left motor, the average error percentage of the mathematical model compared to that of the experimental one is about 5.35%. The maximum speed of the original left motor is 109.34 rpm, while the maximum speed of the left motor mathematical model is 106.6 rpm. In figure 5(b), for the right motor, the average error percentage of the mathematical model compared to that of the experimental one is about 4.75%. The maximum speed of the original right motor is 109.34 rpm, while the maximum speed of the right motor mathematical model is 105.9 rpm.



(a) (b)

Fig. 5. Open loop responses (a) left motor (b) right motor.

The simulink program of the Position Controllers of both left and right motors of the robot can be seen in figure 6. Both position controllers incorporate the PID speed controller with the proportional gain (Kp_s) of 0.8, integral gain (Ki_s) of 0.45, and derivative gain (Kd_s) of 0.05, while the proportional gain for P position controller (Kp_p1) is 0.4. The output responses of the position controller for both left and right motors for set points of 100 cm, 300 cm and 500 cm can be seen in figure 6. The results, as seen in figure 7, show that all the responses show good results with no overshoots and no steady state errors.

The orientation angle of the robot during traveling in horizontal direction straight from position 0cm to 500 cm is shown in figure 8. From figure 8, it can be seen that the orientation angle is ranging from -0.033º (counter clock wise) to 0.069º (clock wise), with its steady state value of about -0.01º. It can be said that the orientation angle of the robot due to the proposed P position controller is almost 0º.

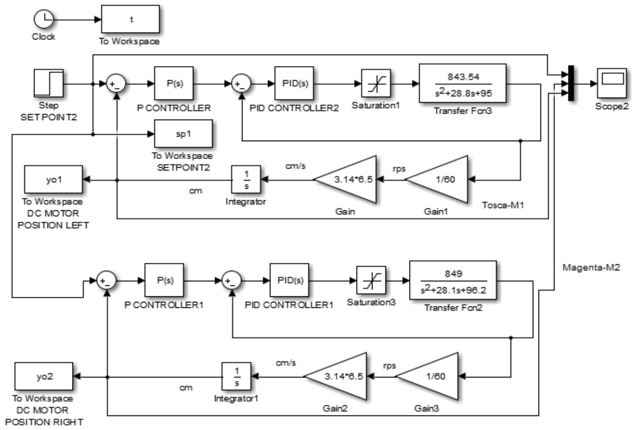
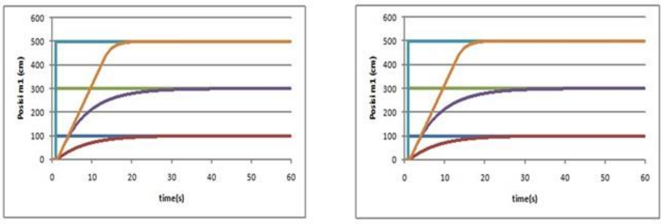


Fig. 6. Position controller.



(a) (b)

Fig. 7. Position controller response (a) left motor (b) right motor.

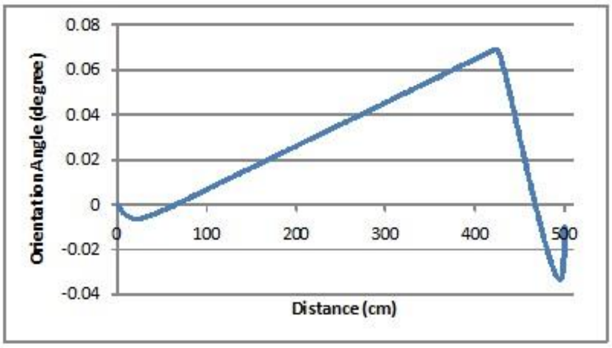


Fig. 8. Angle orientation of robot.

V. Conclusion

The simulation work of P position controller which incorporates PID speed controller for both DC motors of the robot is implemented successfully. This work carries out a system identification process for each motor to obtain its mathematical model in the form of a second order transfer function by utilizing System Identification Tool box in Matlab. This resulted model is then tested and compared to that of the real dc motor open loop experiment. The P position controller which incorporates the PID speed controller performs adequately for set point of 100 cm, 300 cm and 500 cm, with no overshoot and no steady state error. The orientation angle of the robot during traveling straight from its origin point 0 cm to 500 cm due to the proposed P position controller is almost 0º.

References

[1] M. Tavakoli, J. Carriere, and A. Torabi, "Robotics, Smart Wearable Technologies, and Autonomous Intelligent Systems for Healthcare During the COVID-19 Pandemic: An Analysis of the State of the Art and Future Vision," in Adv. Intell. Syst., vol. 2, no. (1), pp. 1-7, 2020.

[2] S.Q. Li, W.L. Guo, H. Liu, T. Wang, Y.Y. Zhou, T. Yu, ... and S.Y. Li, "Clinical application of an intelligent oropharyngeal swab robot: implication for the COVID-19 pandemic," European Respiratory Journal, vol. 56, no. (2), pp. 1-5, 2020.

[3] A.J. Sathyamoorthy, U. Patel, Y.A.S. Savle, M. Paul, and D. Manocha, "COVID -Robot: Monitoring Social Distancing Constraints in Crowded Scenarios," arXiv:2008.06585v2[cs.RO], pp. 1-11, 2020.

[4] J. Baˇcík, P. Tkáˇc, L. Hric, S. Stanislav Alexoviˇc, K. Kyslan, R. Olexa, and D. Perduková, "Phollower—The Universal Autonomous Mobile Robot for Industry and Civil Environments with COVID-19 Germicide Addon Meeting Safety Requirements," Appl. Sci., vol. 10, no. (1), pp. 1-16, 2020.

[5] W. Chen and T. Zhang, "An indoor mobile robot navigation technique using odometry and electronic compass," International Journal of Advanced Robotic Systems, vol. 14, no. (3), pp. 1-15, 2017.

[6] H. Mirzaeinejad and A.M. Shafei, "Modeling and trajectory tracking control of a two-wheeled mobile robot: Gibbs–Appell and prediction-based approaches," Robotica, vol. 36, no. (10), pp. 1-20, 2018.

[7] Y. Tian, N. Sidek, and N. Sarkar, "Modeling and control of a nonholonomic Wheeled Mobile Robot with wheel slip dynamics," IEEE Symposium on Computational Intelligence in Control and Automation, pp. 7-14, 2009.

[8] Y. Wang, S. Wang, R. Tan, and Y. Jiang, "Motion control of a wheeled mobile robot using digital acceleration control method," International Journal of Innovative Computing, Information and Control, vol. 9, no. (1), pp. 387-396, 2013.

[9] A. Pandey, S. Jha, and D. Chakravarty, "Modeling and Control of an Autonomous Three Wheeled Mobile Robot with Front Steer," First IEEE International Conference on Robotic Computing (IRC), pp. 136-142, 2017.

[10] A. Tharakeshwar and A. Ghosal, "Modeling and Simulation of a Three-wheeled Mobile Robot on Uneven Terrains with Two-degree-of-freedom Suspension Mechanisms," Mechanics Based Design of Structures and Machines, vol. 1, no. (1), pp. 1-24, 2015.

[11] N. Hacene and M. Boubekeur, "Motion Analysis and Control of Three-Wheeled Omnidirectional Mobile Robot," Journal of Control, Automation and Electrical Systems, vol. 30, no. (2), pp. 1-20, 2018.

[12] M. Trojnacki, "Dynamics Model of a Four-Wheeled Mobile Robot for Control Applications," Advances in Intelligent Systems and Computing, vol. 323, no. (1), pp. 99-116, 2015.

[13] L. Miková, M. Kelemen, and D. Koniar, "Mathematical model of four wheeled mobile robot and its experimental verification," Applied Mechanics and Materials, vol. 611, no. (1), pp. 130-136, 2014.

[14] R. Oftadeh, M.M. Aref, R. Ghabcheloo, and J. Mattila, "Mechatronic Design of a Four Wheel Steering Mobile Robot with Fault-Tolerant Odometry Feedback," 6th IFAC Symposium on Mechatronic Systems The International Federation of Automatic Control, pp. 663-669, 2013.

[15] T. Jilek, F. Burian, and V. Kriz, "Kinematic Models for Odometry of a Six-Wheeled Mobile Robot," IFAC-Papers OnLine, vol. 49, no. (25), pp. 305–310, 2016.

[16] W. Li, Z. Li, Y. Liu, L. Ding, J. Wang, H. Gao, and Z. Deng, "Semi-autonomous bilateral teleoperation of six-wheeled mobile robot on soft terrains," Mechanical Systems and Signal Processing, vol. 133, no. (1), pp. 1-17, 2019.

[17] C.J. Ostafew, A.P. Schoellig, and T.D. Barfoot, "Visual teach and repeat, repeat, repeat: Iterative Learning Control to improve mobile robot path tracking in challenging outdoor environments," IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 176-181, 2013.

[18] S. Maldonado-Bascón, R.J. López-Sastre, F.J. Acevedo-Rodríguez, and P. Gil-Jiménez, "On-Board Correction of Systematic Odometry Errors in Differential Robots," Journal of Sensors, vol. 2019, no. (1), pp. 1-8, 2019.

[19] S.K. Malu and J. Majumdar, "Kinematics, Localization and Control of Differential Drive Mobile Robot," Global Journal of Researches in Engineering (H): Robotics & Nano-Tech, vol. 14, no. (1), pp. 1-9, 2014.

[20] H. Yang, X. Fan, P. Shi, and C. Hua, "Nonlinear Control for Tracking and Obstacle Avoidance of a Wheeled Mobile Robot With Nonholonomic Constraint," IEEE Transactions on Control Systems Technology, vol. 24, no. (2), pp. 741-746, 2016.

[21] A. Araújo, D. Portugal, M.S. Couceiro, and R.P. Rocha, "Integrating Arduino-based educational mobile robots in ROS," 13th International Conference on Autonomous Robot Systems, Lisbon, pp. 1-6, 2013.

[22] L.C. Png, L. Chen, S. Liu, and W.K. Peh, "An Arduino-based indoor positioning system (IPS) using visible light communication and ultrasound," 2014 IEEE International Conference on Consumer Electronics – Taiwan, pp. 217-218, 2014.