Simulation Strategy for Dual Wheel Mobile Robot Control

Submitted: 22nd May 2014; accepted: 7th July 2014

Zulfiqar Ali Soomro

DOI: 10.14313/JAMRIS_4-2014/33

Abstract:
This paper deals with modeling and control strategies of the motion of wheeled mobile robots. The model of the vehicle has two driving wheels and the angular velocities of the two wheels are independently controlled. To illustrate how the kinematics model of the system can be used to design P controller and a PI controller is applied. The nonlinear and linearised models are simulated in MATLAB® and simulink. The results are drawn here to evaluate the performance of the proposed controllers with concerned graphics.

Keywords: *mobile robot, P controller, PI controller, Simulink simulation*

1. Introduction

Wheeled type mobile robots (WMR) have great importance in industry and factory processing operations due to its effectiveness. The robotic designer must be a compendium of all basic skills and concerned engineering fields. The assistive technology is one of the researched areas in the formation of wheelchairs and development of wheel structured mobile robot.

There is large literature about the motion control of mobile robots. The researched areas of WMR are theoretical and practical in nature of usage. Past research work on automatic mobile robotic control consists on path planning and path tracking control in engineering line. It is divided into four major categories: nonlinear, linear, geometrical, and intelligent aspects. The instantaneous velocities of system components are limited to non-holonomic system in limiting the local movement of the system for mobile robots.

The kinematic type model of the mobile robot is selected due to its dealing with the geometry of robot motion with respect to a fixed reference coordinate frame as a function of time, without considering forces and moments that create the motion. Here PI controller is proposed for motion and uncertainties of WMR as it reduces time rise, steady error with rise of time settling, and the overshooting.

2. Formulation of Mathematical Model

The proposed model is a dual wheeled mobile robot whose position is determined by three co-ordinates in a plane: showing the positions x and y, and robot heading angle θ in an absolute frame. In the configuration space, the robots can move along the path that originates from the current posture (x, y, θ). In below figures, C is the center of the motion of the two wheeled mobile robots is shown. Where (Vx, Vy) represents linear speed or tangential velocity, and w is the angular velocity and center of gravity of the platform is at the origin (o).

Fig. 1-A, represents the active driving wheel in two dimensions where r denotes the radius of the wheels, and D the azimuth length between the two wheels and C as the center of motion of the mobile robot. Fig. 1-B shows the (Vx, Vy) absolute (Cartesian) in coordinate system of the linear speed with the center of the platform at the origin (o), w is the angular velocity and theta (θ) is the heading angle of the turn in radians. This system is shown in x and y coordinates and orientation change due to time. Based on its speed and orientation at any instant, the x the y coordinates of the robot's center point are also changing. Although the states change with respect to time, the physical laws that rule the behavior of the mobile robot do not change with time, which reflects the system as the time invariant.

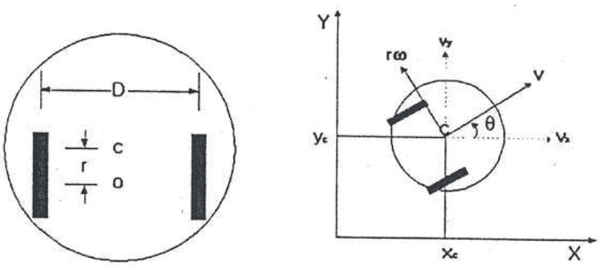

Fig. 1. Dynamics of driving wheel in two dimensions

The mathematical equations behind our system model as represented in Fig. 4 are as follows:

$$V=\sqrt{V^2x+V^2y}\ , \tag{1}$$

where

$$Vx=V_c\cos\theta\ ,Vy=V_c\sin\theta\ ,$$

while

$$\dot{X}_C=Vx\text{, and }\dot{Y}_C=Vy\ ,$$

$$\theta c=tan^{-1}\left[\frac{Vy}{Vx}\right], \tag{2}$$

$$W_C = \frac{r_w}{D(W_R - W_L)}, \tag{3}$$

where W_C – angular velocity, or steering velocity, W_R – angular velocity for right wheel, W_L – angular velocity for left wheel, D – azimuth length between the wheels, r_w – radius of the wheels.

$$V_C = \frac{r_w}{2(W_R - W_L)}, \tag{4}$$

where V_C = tangential velocity, or linear velocity.

$$\dot{X} = V_c \cos\theta = \frac{r_w}{2(W_R - W_L)} \cos\theta, \tag{5}$$

$$\dot{Y}_C = V_c \sin\theta = \frac{r_w}{2(W_R - W_L)} \sin\theta, \tag{6}$$

$$\theta = Wc = \frac{r_w}{D(W_R - W_L)}. \tag{7}$$

Here control inputs are (W_R, W_L) and our control outputs are (*X, Y,* θ*)*. Therefore, desired trajectory is *X(t), Y(t),* θ*(t).*

This kind of drive is used to establish our equations as a differential drive, and the desired simplest selection form, where V_L and V_R are equal. In this case the radius *D* is infinite and the robot moves in a straight line, but it should be noted that the robot does not move in a straight line, except in a curved trajectory about a point on distance *D* away from the center of the robot, altering both the robot's position and orientation. The differential drive system for Vehicle is very sensitive to the relative velocity of wheels to use castor wheels for balance of the WMR.

3. Control Strategy for WMR

The dual wheel mobile robot control is difficult as the control inputs are produced to satisfy the constraints while the robot is moving. A mathematical model is never a perfect reflection of physical system of the robot. The actual facing challenge in motion planning is to invent a planning scheme integrated with a control system being able to detect and recognize the unexpected incidents on the basis of sensory information, to modify and adjust the basic plan at a high rate to cope with time and location variations in the happening of events without re-planning and coping.

3.1. Simulation of PI Control Strategy

Here PI controller is proposed for this dual wheel robot (DWR). The P controller is less of its tolerance for high gains in primary shortcoming, which acts to add an integral gain KI to its controller. The integral gain will draws out the signal drop and usually the system reacts better with a PI controller than a P controller due to grow of integral gain ever larger even within small errors. Integral gain provides stiffness to the signal, which means when the error happens, the integral gain will move to correct it. If the gain is higher, it will the faster the correction.

The Design of a PI controller is more complicated than a P controller as shown in Figs. 2 and 3. To avoid saturation and the "wind-up" problems, more gains is added. Tuning the PI controller is very sensitive for high K_p gains. The best results are obtained when *Kp* is 0.1, and when the gain is larger, then controller performs poorly.

The mathematical formulae described above are represented in Fig. 2 and finally merged in the Fig. 3, in shape of subsystem. In Fig. 2, there are three inputs, two outputs and four gains representing as proportional (K_p) and integral proportional controllers (k_I).

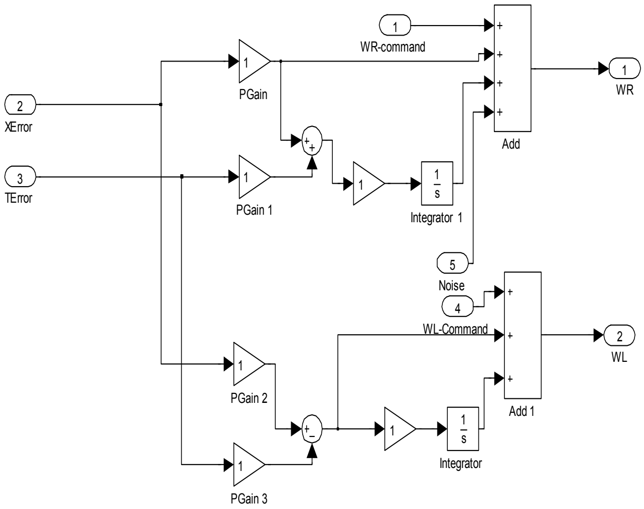

Fig. 2. Simulink diagram for the PI controller

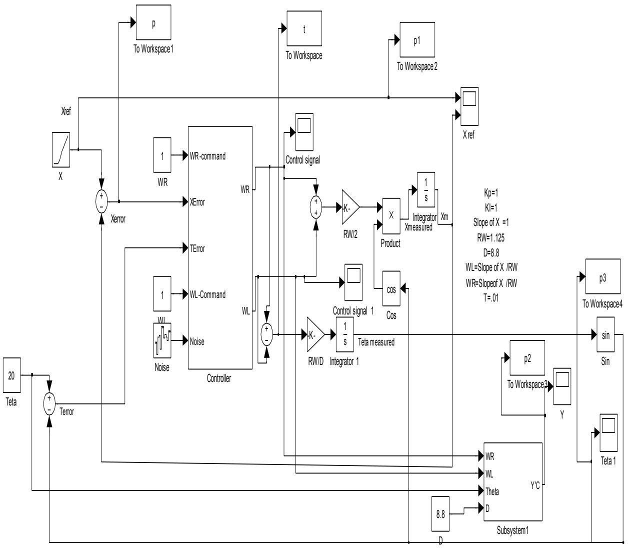

Fig. 3. Simulink Diagram for PI Controller

In Fig. 3, the controller in subsystem represented the Simulink block diagram of Fig. 2 is inserted and subsystem-1 represents the right and left wheels along with their orientation and distance between them.

Some Simulink blocks are denoted by 'scopes' to represent different control parameters for having their graphics too.

4. Simulation Results

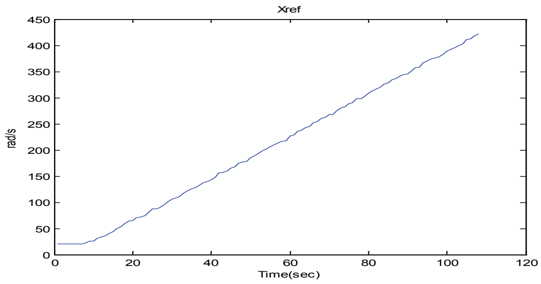

Fig. 4. Xref-position for PI Controller

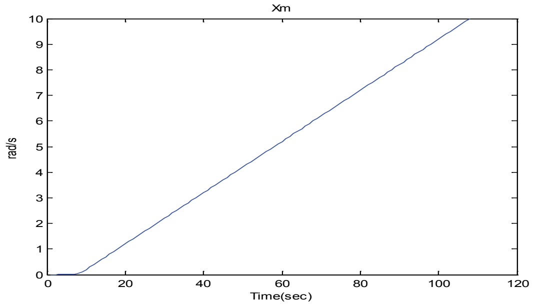

Fig. 5 Xm-position for PI Controller

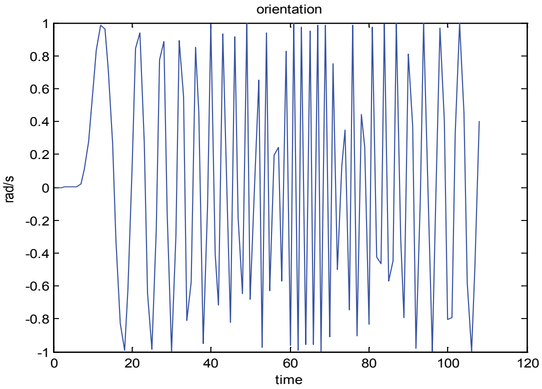

Fig. 6. Theta PI controller the Kp gain is 50 and KI gain is 1

Fig. 4 and 5 show the PI Simulink controller simulation outcomes. In Fig. 4, the output of X-position, which is a ramp input follows the X-reference input, is shown. Fig. 5 presents the PI controller Y-position under, where K p gain is 10 and KI is 0.1. In Fig. 5 the PI controller drives the system better than the P controller as it denotes drives the Y-position for stability and lower oscillation.

Kp gain is 50 and the KI gain is 1 is shown as theta for the mobile robot system in Fig. 6 where the mobile robot system is so sensitive for the KI gain changing with the K p changing.

The Y-position for the mobile system is shown in Fig. 7. The slow system shows as if it does not move at all with tuning, but its gains above it show a good response from the system and the controller drive the system for stability.

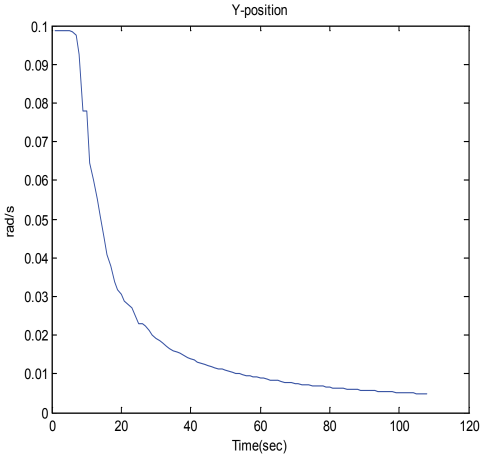

Fig. 7. Y-position under PI controller

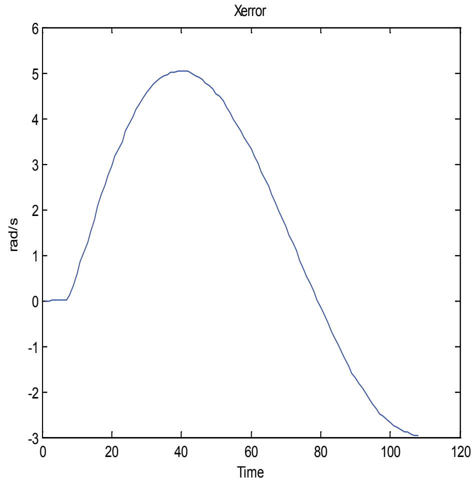

Fig. 8. Xerror in PI Controller

Above Fig. 8 shows the rise in curve conceives that unit step response is over damped giving overshoot. This shows the error of angular velocity of wheels in x-position.

Table 1. Different Parameters used in Simulink

Parameter	Xerror	Xref	Y	θ	Twist
Minimum	–2.973	0.00	0.0048	–0.9975	20.328
Maximum	5.0514	10	0.0987	1.000	422.13

Table 2. PI controller simulation results for wheel mobile robot system

Kp	KI	X-position	Y-position	Theta	Control Signals LW	Control Signals RW
1	0.1	0.0002123	0.0002022	0.12150	0.1158	0.1159
3	0.1	0.0008265	0.0008176	0.15090	0.1538	0.1509
10	0.1	7.865e-005	8.968e-005	0.08599	0.1857	0.1857
20	0.1	6.032e-005	5.073e-005	0.08039	0.2789	0.2792
40	0.1	2.238e-005	2.246e-005	0.05327	0.2605	0.2607
50	0.1	1.865e-005	1.258e-005	0.03526	0.2835	0.2538

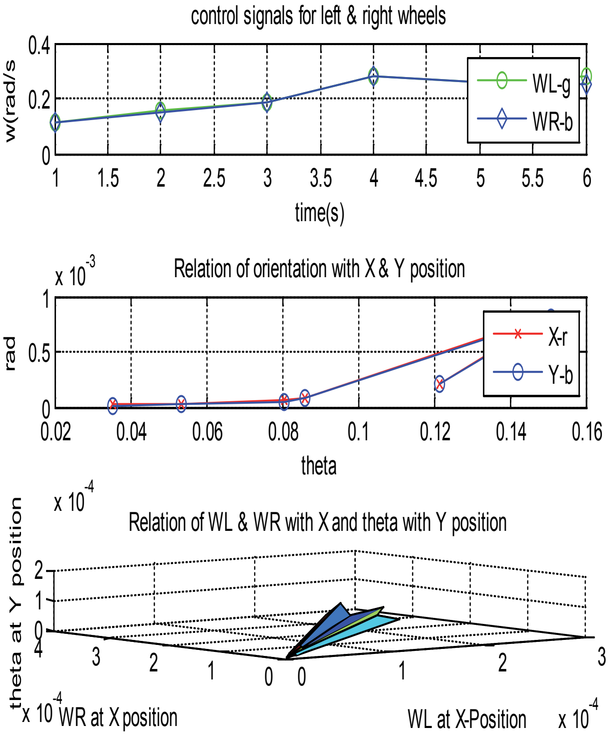

Fig. 9. Correlation of Simulink parameters with each other

In 1st part of above Fig. 9, the varying relation of left and right wheels is shown w.r.t times, which are almost on same line. Similarly in 2nd part, the X-Y positions with respect to orientation are lying in same line. While in the 3rd part, X-position of wheels w.r.t orientation at Y-position is shown in 3-D format.

5. Conclusion

The various simulation results were obtained by using the strategy to apply K_p and K_I controllers to control the noise and uncertainties created by WMR. The mathematical model was framed for motion study. These mathematical formulas were developed in Simulink block diagram later on to observe achievable objective. The Matlab tool was used receive concerned tables and graphics.

There are a number of standardized methodologies for tuning, which can be used to simplify the process but often tuning is relies on trial-and-error as well as the experience of the engineer. The steady state error can be eliminated by introducing some elements of integral control (K_I).

AUTHOR

Zulfiqar Ali Soomro – Directorate of Post-graduate Studies, Mehran University of Engineering & Technology, Jamshoro Sindh 76052, Pakistan.

References

[1] Gao Zhiqiang, "From linear to nonlinear control means: a practical Progression," *ISA Transactions*, vol. 41, no. 2 , April 2002, 177–89. DOI: http://dx.doi.org/10.1016/S0019-0578(07)60077-9.

[2] Miller Merl K., Winkless Nelson, and Bosworth Joe, *The Personal Robot Navigator*, Robot Press, 2000.

[3] Nadiu Desineni, *Optimal Control Systems*, Dover Publications, 2000.

[4] Nehmzow Ulrich, *Mobile Robotics*, Springer, 2003.

[5] Dudek Gregory, Jenkin Michael, Computational Principles of Mobile Robotics," Cambridge University. 2000.

[6] Martin Fred. "Robotic Explorations," Prentice-Hall. 2001.

[7] Branwyn Gareth, *Absolute Beginner's Guide to Building Robots*, Que Publishing, 2004.

[8] Barzamini R., Afshar A., *Dynamic adaptive tracking control for wheeled mobile robots*, AmirKabir University of Technology, Tehran, Iran, 2006.

[9] Gao Zhiqiang, Huang Yi, Han Jingqing, "An alternative Paradigm For Control System Design." In: *Proc. of the 40th IEEE Conference on Decision and Control*, Orlando, Florida, December 2001, 4578–4585.

[10] Ellis George, "*Control Systems: Design Guide*," Academic Press, 2000.

[11] Mc Comb Gordan, *The Robot Builder's Bonanza*, McGraw-Hill, 2001.

[12] Wise Edwin, *Applied Robotics*, PROMPT 2000.