Lyapunov Based Trajectory Tracking Dynamic Control for a QBOT-2

Marc Vazquez, Mateusz Ardito-Proulx and Sabiha Wadoo
New York Institute of Technology

Abstract **This paper presents the nonlinear kinematic and dynamic control for a QBOT2, a wheeled mobile robot (WMR). The work presents the design and simulation of a trajectory tracking controller for the system. The kinematic controller is a Lyapunov based nonlinear feedback control. The method of Backstepping is used to design the dynamic controller. The controllers are implemented on the system hardware using QUARC, a MATLAB-Simulink based software.**

Index Terms **-QBOT2, Lyapunov design, Backstepping method, Dynamic control, QUARC**

INTRODUCTION

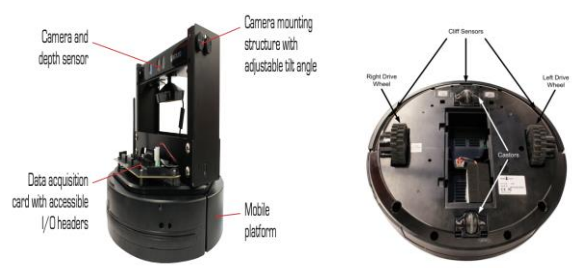
Wheeled mobile robots (WMR) have presented themselves as a highly versatile and complex class of robots. WMR are suitable for a variety of applications in unstructured environments where a high degree of autonomy is required. Consumers, as well as industries, rely on mobile robots due to their low cost and flexibility of use. This desired intelligent behavior as well as growing widespread utilization has motivated an intense research of WMR in the last decade [1]. With many applications germane to the development of autonomous vehicles, the issue of controlling the trajectory and velocity of a mobile robot becomes increasingly pressing. When designing controllers, we may consider controlling the kinematic model or the dynamic model of the robot. The kinematic model of the robot describes the motion of the robot in terms of its linear and angular velocities, while the dynamic model describes the torques that actuate these velocities.

Furthermore, the feedback control of a WMR can be divided into three control objectives: trajectory tracking, path following, and point stabilization [2]. Trajectory tracking refers to the objective of moving a robot along a time parameterized reference movement [3]. Alternatively, path following refers to the objective of moving a robot along a predetermined path, not dictated by variable of time.

The focus of this paper is to design a nonlinear trajectory tracking control for both kinematic and dynamic models of a WMR. The WMR used in this paper is Quanser QBOT2. The QBOT2 is a differential drive WMR, in the sense that the two wheels that steer the vehicle can be actuated independently [2]. When considering the real-world implementations of this controller, the advantage of stability becomes evident, although with it the disadvantage of a slow settling time. In addition, there is a limit on the maximum gain the robot can handle before its transient oscillations become unacceptable.

MATHEMATICAL MODEL

We use QBOT2 as the platform for the implementation of the controllers designed in this paper. QBOT2 is a WMR designed by Quanser [4]. Figure 1 (a) illustrates the QBOT and describes some of its peripherals, while Figure 1 (b) shows the bottom of the QBOT and its drive wheels. The movement of the QBOT is dictated by two independently controlled drive wheels parallel to each other at a distance *l*, which is 0.235 m. Two castor wheels act to support the QBOT. Since these castor wheels do not affect the direction of movement or the magnitude of the driving force, they are not considered during the modeling process. The movement of the robot is expressed in terms of the velocity of both wheels, v_L and v_R.



(a) (b)

Figure 1. Quanser QBOT2 Platform

I. Kinematic Model

The kinematic model of the robot is the relationship of these wheel velocities to the linear velocity v_c and angular velocity w_c of the robot. The forward/backward velocity v_c of the robot chassis center and the angular velocity ω_c are represented as

$$v_c = \frac{v_L + v_R}{2} \tag{1}$$

$$\omega_c = \dot{\theta} = \frac{v_R - v_L}{d} \tag{2}$$

where d is the distance between the left and right wheels, $\dot{\theta}$ is the heading angle of the robot, v_L is the left wheel velocity, and v_R is the right wheel velocity. For a two wheeled differential drive mobile robot, we wish to describe its motion in a Cartesian frame. For specifying the movement of the

978-1-7281-7520-1/20/$31.00 ©2020 IEEE

mobile robot, two coordinate system need to be established. The first one is global coordinate frame, which takes a point in a two-dimensional plane as the origin; the other one is local coordinate frame with the centroid of the robot as the origin, shown in Figure 2.

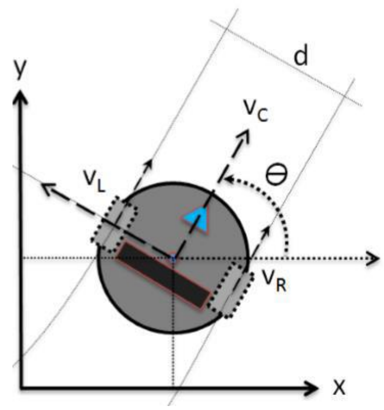


Figure 2. Kinematic Model of a mobile robot

The robot's posture q is defined by the following position vector.

$$q = [x \quad y \quad \theta]^T \tag{3}$$

The robot is bounded by a non-holonomic constraint [. This constraint can be described by

$$\dot{x}\sin(\theta) - \dot{y}\cos(\theta) = 0 \tag{4}$$

In order to describe the linear and angular velocities of the robot, v_c and ω_c, respectively, we take the derivative of posture and map v_c and ω_c to the robots cartesian frame so that the kinematic model can be described as [5, 6].

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} \tag{5}$$

From equation previous stated, it is reasonable to conclude that the trajectory can be controlled by adjusting the linear velocity and angular velocity of the mobile robot.

II. Dynamic Model

The dynamic model of the QBOT 2 describes the motion of the robot in terms of the forces, or torques applied, providing a richer analysis of motion from a more physical perspective. For a nonholonomic mobile robot, the Lagrange formulation to establish the equations of motion is given by [7],[8],[9].

$$M(q)\,\dot{\eta} + C_m(q,\dot{q})\eta + G(q) = B(q)\tau \tag{6}$$

where $\eta = [v, \omega]$ presents a velocity vector, $M(q)$ presents a symmetric positive definite inertia matrix, $C_m(q,\dot{q})$ presents the centripetal and Coriolis matrix, q is generalized coordinates in cartesian coordinates and $\dot{q}$ presents constrained velocities; $G(q)$ presents the unknown bounded disturbance and $B(q)\,\tau$ presents a torque control input vector with

$$B(q) = \frac{1}{r}\begin{bmatrix} 1 & 1 \\ \frac{d}{2} & -\frac{d}{2} \end{bmatrix}$$

and

$$\tau = [\tau_r \quad \tau_l]^T$$

represents the torques of right and left motors. Combining (6) with non-holonomic constraints (4), the dynamic model of non-holonomic mobile robot is given as

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \frac{1}{mr} & \frac{1}{mr} \\ \frac{d}{2rI} & -\frac{d}{2rI} \end{bmatrix} \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix} \tag{7}$$

where m is the mass of the robot, r is the wheel radius, d is the distance between the two wheels and I is the moment of inertia about the robot chassis' axis of rotation.

KINEMATIC CONTROL

The purpose of the trajectory-tracking controller is to implement a control regulation to make the error posture infinitely approach to zero. The kinematic controller is designed using Lyapunov theory to essentially converge the energy of the error signal to zero by choosing controller gains such that the controller is stable. First we define the error model of the robot by defining $q_e = [x_e, y_e, \theta_e]^T$ as the difference between the reference position $q_r = [x_r, y_r, \theta_r]^T$ and the actual position of the mobile robot $q_c = [x_c, y_c, \theta_c]^T$. The relationship equation between the three postures is as following

$$q_e = R(\theta)(q_r - q_c) \tag{8}$$

where $R(\theta)$ is the rotation matrix given by

$$R(\theta) = \begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The derivative of (8) with the substitution of (5) gives the error model of mobile robot as

$$\dot{q_e} = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} cos\theta_e & 0 \\ \sin\theta_e & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} + \begin{bmatrix} -1 & y_e \\ 0 & -x_e \\ 0 & -1 \end{bmatrix} \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} \tag{9}$$

The kinematic control model (9) has forward velocity v_c and the angular velocity w_c as the control inputs and the error in position as the output. The goal of the kinematic controller is to drive the error to zero. The nonlinear kinematic controller is designed using a Lyapunov based method where a positive definite Lyapunov function is chosen such that the system dynamics are stable along the function's time change. Defining the following Lyapunov function for the system

$$V_0 = \frac{1}{2}(x_e^2 + y_e^2) + \left(\frac{1 - cos\theta_e}{k_2}\right) \tag{10}$$

The time derivative of the Lyapunov function V_0 is

$$\dot{V}_0 = \dot{x}_e x_e + \dot{y}_e y_e + \dot{\theta}_e \frac{sin\theta_e}{k_2} \tag{11}$$

Combining (11 with the error dynamics (9) yields

$$\dot{V}_0 = (-v_c + v_r cos\theta_e)x_e + v_r sin\theta_e y_e + \cdots$$
$$\ldots \ldots + \left(-\frac{\omega_c}{k_2} + \frac{\omega_r}{k_2}\right) sin\theta_e \tag{12}$$

Choosing the following nonlinear control law,

$$\begin{bmatrix} v_c \\ \omega_c \end{bmatrix} = \begin{bmatrix} v_r cos\theta_e + k_1 x_e \\ \omega_r + k_2 v_r y_e + k_3 sin\theta_e \end{bmatrix} \quad (13)$$

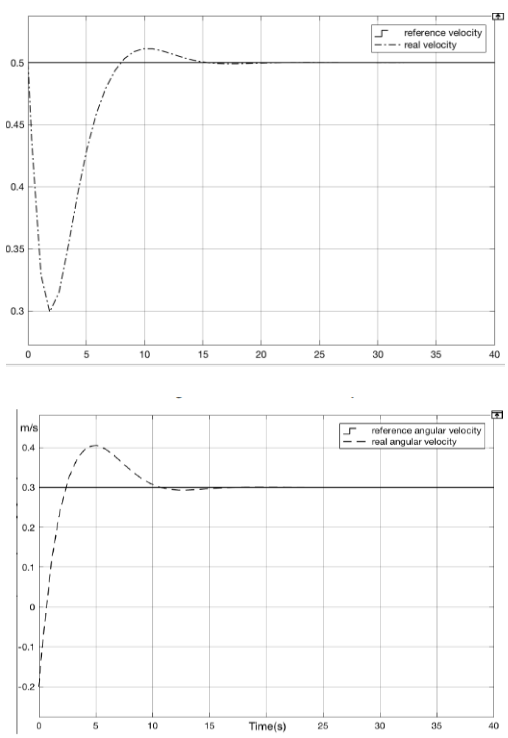
the time derivate of Lyapunov function is

$$\dot{V}_0 = -k_1 x_e^2 - \frac{k_3}{k_2} sin^2\theta_e$$

With the control gains k_1, k_2, k_3 chosen to be positive the stability condition ($\dot{V}_0 \leq 0$) is satisfied.

KINEMATIC CONTROL IMPLEMENTATION

The simulation for the kinematic controller implementation is obtained using QUARC which is a Simulink based software. The robot was first run in real time with Simulink and QUARC with a set initial velocity values for the right and left wheels. The initial position of QBOT 2 is (-1,1,0), we set reference initial position as (0,0,0), reference linear velocity as 0.5 m/s and angular velocity as 0.1 rad/s. The control gains were selected as $k_1 = 0.7$, $k_2 = 1$, $k_3 = 0.7$. The implementation setup in Simulink using kinematic control is shown in Figure 3. and implementation results are shown in Figure 4.



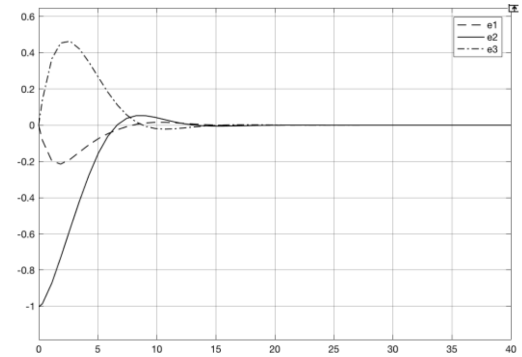


Figure 4. Kinematic Control Trajectory Tracking Implementation Results

DYNAMIC CONTROL

The dynamic control design is achieved using Backstepping technique of nonlinear control. Backstepping is a technique developed circa 1990 by Petar V. Kokotovic and others for designing stabilizing controls for a special class of nonlinear dynamical systems [10]. This method is used for the design of the torque control based on the system dynamics and to make the velocity of the mobile robot converges to the generated desired velocity [11]. The diagram of the trajectory tracking dynamic control is shown in Figure 5.

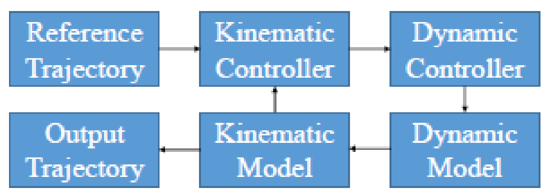


Figure 5. Trajectory Tracking Dynamic Control Block Diagram

To design the dynamic control law, the model given by equation (7) is reformulated in terms of tracking errors. The output of kinematic closed loop dynamics (13) are taken as reference for the velocity, denoted as $[v_r, w_r]^T$, then the velocity error should be

$$\begin{bmatrix} v_e \\ \omega_e \end{bmatrix} = \begin{bmatrix} v_r - v_c \\ \omega_r - \omega_c \end{bmatrix} \quad (14)$$

To design a controller that acts upon the torques of the robot which in turn actuate the wheels and induce motion, we apply a similar method by first defining a Lyapunov candidate, and attempting to converge the error of the robots linear and angular velocities to zero. We define the following Lyapunov function for the dynamic model in terms of the Lyapunov function V_0 of kinematic model

$$V_1 = V_0 + \frac{1}{2}(v_e^2 + \omega_e^2) \geq 0 \quad (15)$$

The derivative of the Lyapunov function V_1 yields

$$\dot{V}_1 = \dot{V}_0 + \dot{v}_e v_e + \dot{\omega}_e \omega_e \quad (16)$$

Choosing the following control law, we can satisfy the condition of convergence of Lyapunov function V_1.

$$\begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix} = \begin{bmatrix} \frac{mr}{2} & \frac{rI}{d} \\ \frac{mr}{2} & -\frac{rI}{d} \end{bmatrix} \begin{bmatrix} \dot{v}_r \\ \dot{\omega}_r \end{bmatrix} + \begin{bmatrix} \frac{k_4 mr}{2} & \frac{k_5 rI}{d} \\ \frac{k_4 mr}{2} & -\frac{k_5 rI}{d} \end{bmatrix} \begin{bmatrix} v_e \\ \omega_e \end{bmatrix} \quad (17)$$

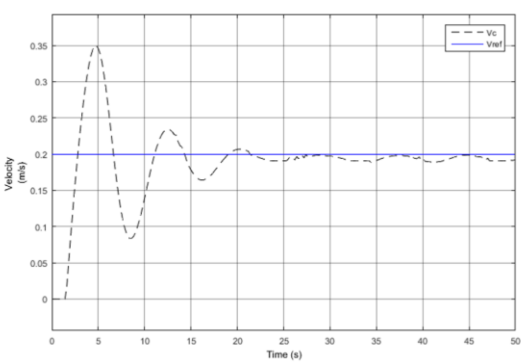
Substituting (17) in velocity error dynamics and using (12), the Lyapunov function derivative (16) yields

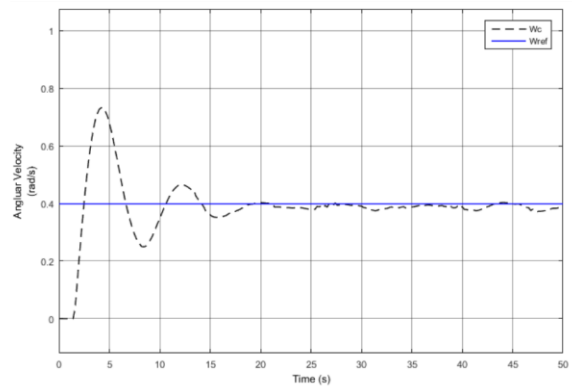
$$\dot{V}_1 = \dot{V}_0 - k_4 {v_e}^2 - k_5 {\omega_e}^2 \quad (18)$$

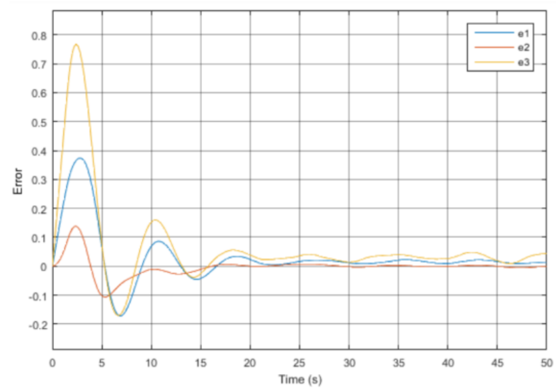
The control gains k_4 and k_5 must be positive constants to ensure Lyapunov stability.

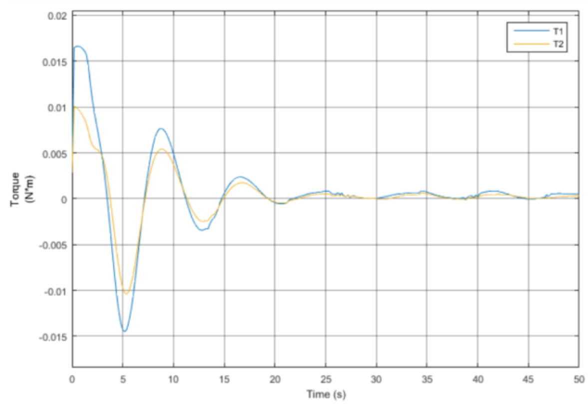
DYNAMIC CONTROL IMPLEMENTATION

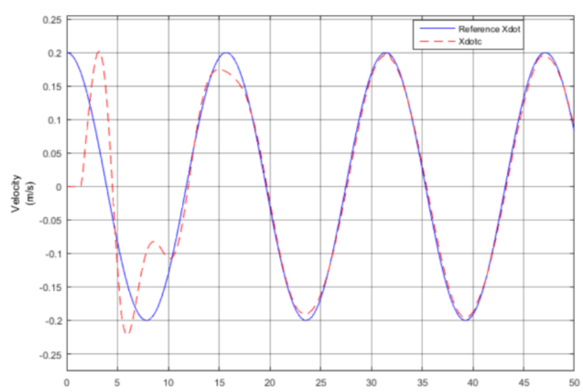
Below, we examine the performance of these controllers with respect to linear and angular velocity tracking, circular trajectory tracking, and position error tracking. We examined the performance both in a simulated environment as well as the actual implementation on the QBOT 2. All simulations as well as the protocol for implementing the controllers on the QBOT 2 are performed in MATLAB/Simulink. The simulation setup for the dynamic controller in Simulink is shown in Figure 6. and control results are shown in Figure 7. The initial position of QBOT2 is (-1,1,0), set reference initial position as (0,0,0), initial linear velocity is 0.2 m/s and angular velocity is 0.4 rad/s. The control gains are selected as $k1$ = 0.7, $k2$ = 1, $k3$ = 0.7, $k4$ = 0.28, and $k5$ = 0.28. After adding the kinematic and dynamic control laws we have designed, we obtained the results to compare between desired velocity(linear/angular) and actual velocity(linear/angular), desired torque (linear/angular) and actual torque (linear/angular), the results are shown in Figure 7.











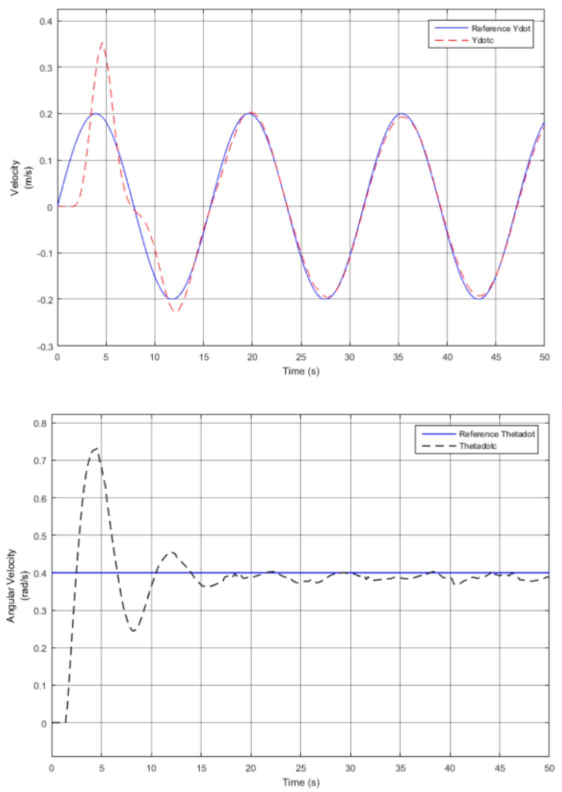


Figure 7. Dynamic Control Trajectory Tracking Implementation Results

CONCLUSION

The paper presented the design and implementation of a nonlinear dynamic control on a QBOT2. The goal of this research was to design and implement a kinematic and dynamic controller to converge the trajectory of a differential-drive wheeled mobile robot to its reference trajectory. Much of the controller design was formulated using the Lyapunov and Backstepping methods. However, upon implementing this controller on the QBOT 2, the control gains had to be kept low to avoid unacceptable oscillations in the velocities of the robot. The settling time for both the linear and angular velocity is clearly quite slow, but such is the tradeoff for minimal oscillations. However, the steady state response yields impressive trajectory tracking, after the first circle is completed which is affected by the slow rise time of the system.

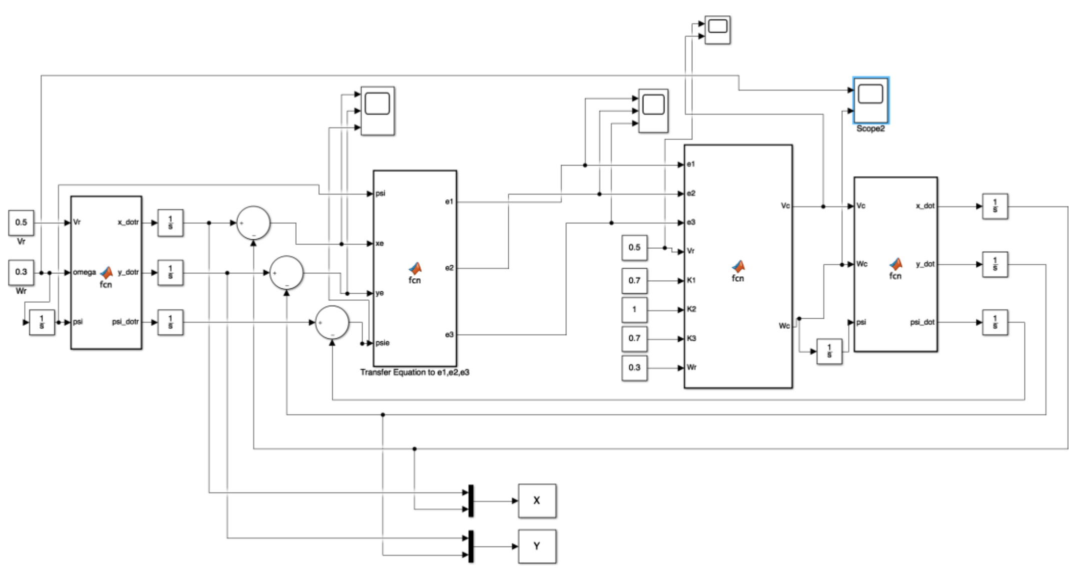


Figure 4. Kinematic Control Implementation Simulink Model

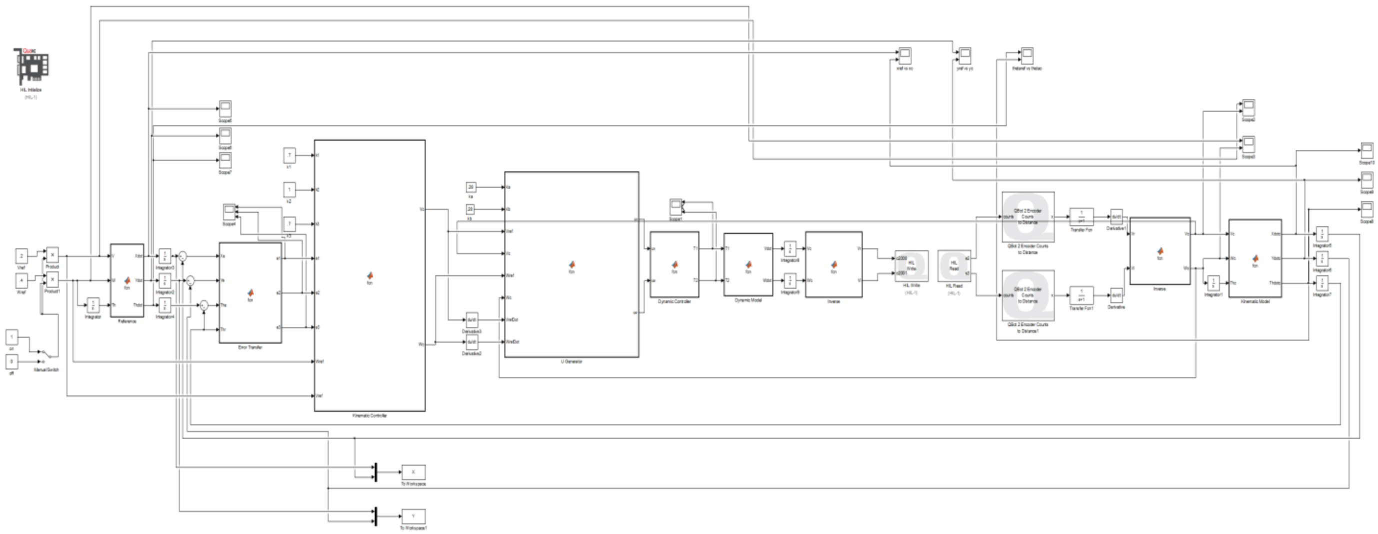


Figure 6. Dynamic Control Implementation Simulink Model

References

[1] R.Fierro and F.L. Lewis, “Control of a Nonholonomic Mobile Robot: Backstepping Kinematics into Dynamics”, Proceedings of the 34th Conference on Decision & Control, 1995.

[2] Siti Nurmaini, K Dewi, B Tutuko, "Differential-Drive Mobile Robot Control Design based-on Linear Feedback Control Law", IAES International Conference on Electrical Engineering, 190, 2017.

[3] A.Shukla, H. Goyal, S.Agarwal, S.Nema, P.K.Padhy, "Path Control of Mobile Robot Using Cuckoo-PID", IEEE International Conference on Computer, Communication and Control, 2015.

[4] Quanser, “Quanser QBOT 2 for Quarc Instructors Workbook” 2017.

[5] Dhaouadi, Rached, and Ahmad Abu Hatab, "Dynamic modelling of differential-drive mobile robots using Lagrange and newton-euler methodologies: A unified framework.", Advances in Robotics Automation. 2013.

[6] Y. Yamamoto and X. Yun. "Coordinating Locomotion and Manipulation of a Mobile Manipulator.", World Scientific, 1993.

[7] G. Zidani, S. Drid, L. Chrifi-Alaoui, A. Benmakhlouf, and S. Chaouch, “Backstepping controller for a wheeled mobile robot,” in 2015 4th International Conference on Systems and Control (ICSC), 2015, pp. 443–448.

[8] C. S. Shijin and K. Udayakumar, “Speed control of wheeled mobile robots using PID with dynamic and kinematic modelling,” in 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 2017, pp. 1–7.

[9] Gholipour, Ali, and M. J. Yazdanpanah. ”Dynamic tracking control of nonholonomic mobile robot with model reference adaptation for uncertain parameters.” European Control Conference (ECC), 2003. IEEE, 2003.

[10] Richa Watiasih, Muhammad Rivai,"Path Planning Mobile Robot Using Waypoint For Gas Level Mapping ", 2017 International Seminar on Intelligent Technology and Its Application.

[11] Emna Baklouti, Nader Ben Amor,"Particle filter localization and real time obstacle avoidance control based on 3D perception for wheelchair mobile robot ", 2016 IEEE 8th International Conference on Intelligent Systems.2016.