Journal of Simulation & Analysis of Novel Technologies in Mechanical Engineering
10 (3) (2017) 0041~0050
HTTP://JSME.IAUKHSH.AC.IR
ISSN: 2008-4927

Trajectory Tracking of Two-Wheeled Mobile Robots, Using LQR Optimal Control Method, Based On Computational Model of KHEPERA IV

Amin Abbasi[1], Ata Jahangir Moshayedi[2,*]

1- Department of Electrical Engineering, Khomeinishahr Branch, Islamic Azad University, Isfahan, 35145-179, Iran

2- Department of Electronic Science, Pune Savitribai Phule Pune University,Pune, 411007, India

**Corresponding Author: moshayedi@electronics.unipune.ac.in*

(Manuscript Received --- 10, 2017; Revised --- 12, 2017; Accepted --- 01, 2018; Online --- 01, 2018)

Abstract

This paper presents a model-based control design for trajectory tracking of two-wheeled mobile robots based on Linear Quadratic Regulator (LQR) optimal control. The model proposed in this article has been implemented on a computational model which is obtained from kinematic and dynamic relations of KHEPERA IV. Along the correct dynamic model for KHEPERA IV plat form which is not elaborated properly in pervious researcher work the purpose of control is to track a predefined reference trajectory with the best possible precision considering the dynamic limits of the robot. Applying several challenging paths to the system showed that the control design is able to track applied reference paths with an acceptable tracking error.

Keywords: KHEPERA IV, Computational Model, LQR Optimal Control, Dynamic model.

1- Introduction

Mobile robots are suitable for many applications. One of the most challenging research problems in robotic systems is to control the motion of a mobile robot in order to track a predefined trajectory with the best possible precision having a structured platform to examine the designed control models is a major task in scientific research studies. KHEPERA IV is one of the most popular mobile robots which is designed and manufactured by K-Team Corporation and it's mostly used for experimental studies in robotics. This robot is applied as benchmark robotic system which can be used for in almost any applications such as navigation, swarm, artificial intelligence, computation, demonstration, etc. [1]. This research paper presents an efficient control method based on Linear Quadratic Regulator (LQR) optimal control for trajectory tracking of a two-wheeled mobile robot. LQR is an optimal control method which provides a systematic way for computing the state feedback control gain matrix. To determine the feedback gain optimally, matrices Q and R are available. where Q is a positive-definite or positive-semidefinite diagonal matrix which is related to state variables, and R is a positive-definite diagonal matrix which is related to input variables [2]. By tuning the elements of Q and R, the optimum performance of the system can be reached.

The presented control method is implemented on computational model of KHEPERA IV and tracking of the applied trajectory is provided by controlling the angular velocity of the wheels[3]. The performance of the control design is evaluated by measuring the error of tracking. This paper is organized as follows. In section 2, the computational model of KHEPERA IV based on kinematic and dynamic equations has been presented. In section 3, an optimal control design based on LQR method is proposed. In section 4, the system has been evaluated by analyzing the simulation results. And section 5 is the conclusion.

2- Computational Model of KHEPERA IV

In this section, a computational model for KHEPERA IV, based on kinematic and dynamic equations of the robot is presented. To create this model, it's assumed that robot moves on a perfectly flat surface with no sliding and no slope.

KHEPERA IV has two independently driven wheels, which rotate around a common axis. Two passive wheels, which rotate in all the directions, ensure the stability of the robot. Fig. 1 shows the kinematic model of KHEPERA IV. Motion of two-wheeled mobile robots is controlled by angular velocity of the wheels ω_L and ω_R. Tangential velocity of the wheels v_L and v_R can be calculated from the following equations:

$$v_R = r\omega_{\mathrm{R}} \tag{1}$$

$$v_L = r\omega_{\mathrm{L}} \tag{2}$$

Where r is radius of the wheels.Angular velocity ω and tangential velocity v of the robot can be obtained from equations (3) and (4) respectively.

$$\omega = \frac{v_R - v_L}{D} \tag{3}$$

$$v = \frac{v_R + v_L}{2} \tag{4}$$

Where D is the distance between the wheels. Position of the robot is given by coordinates x and y and angle θ. To obtain x, y and θ from ω and v, the following relations can be used:

$$\dot{\theta}(t) = \omega \quad \rightarrow \quad \theta(t) = \int \omega(t)dt \tag{5}$$

$$\dot{x}(t) = v_x = v \cdot cos\theta \quad \rightarrow \quad x(t) = \int v \cdot cos\theta\, dt \tag{6}$$

$$\dot{y}(t) = v_y = v \cdot sin\theta \quad \rightarrow \quad y(t) = \int v \cdot sin\theta\, dt \tag{7}$$

According to equations (1) to (7) and the robot model shown in

Fig. 1, the inputs to kinematic model are ω_L and ω_R output variables are x, y and θ.

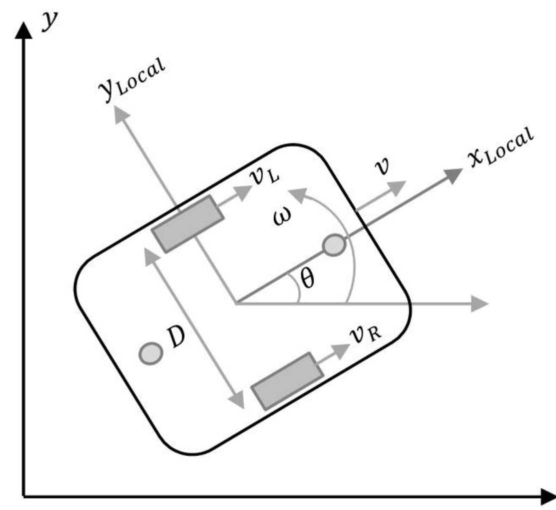

Fig. 1: Kinematic Model of KHEPERA IV

2-1- Dynamic Model of KHEPERA IV:

Some parameters of the system such as friction force and mass of the robot are dynamic specifications, and have not mentioned in kinematic model. Therefore, the computational model can be improved using mathematical equations of dynamic model. Dynamic model of KHEPERA IV has shown in FIG. 2 and shape of the mathematical equations are as follows:

$$F_L + F_R = m \cdot a \tag{8}$$

$$\frac{(F_R - F_L)D}{2} = J \cdot \varepsilon \tag{9}$$

Where F_L and F_R are the forces applied to left and right wheel respectively. m(Mass of the robot), a (tangential acceleration) J (moment of inertia) and ε (angular acceleration) are dynamic parameters of the robot. Now the state space model of the system can be defined using kinematic and dynamic specifications of KHEPERA IV. Inputs, outputs and state variables of the model have chosen as follows:

$$x(t) = [x_1(t), x_2(t), x_3(t), x_4(t)] = [v(t), \omega(t), \omega_L(t), \omega_R(t)] \tag{10}$$

$$u(t) = [u_1(t), u_2(t), u_3(t), u_4(t)] = [F_L, F_R, U_L, U_R] \tag{11}$$

$$y(t) = [y_1(t), y_2(t)] = [x_3(t), x_4(t)] = [\omega_L(t), \omega_R(t)] \tag{12}$$

Where U_L and U_R are voltages applied to DC motors which drive the wheels of the robot. From equation (10) we have

$$x_1(t) = v(t). \tag{13}$$

After taking derivation from the sides of the equation we have $\dot{x}_1(t) = a(t)$ and using equation (8) the following relation will be obtained:

$$\dot{x}_1(t) = \frac{F_L + F_R}{m} = \frac{1}{m}F_L + \frac{1}{m}F_R \tag{14}$$

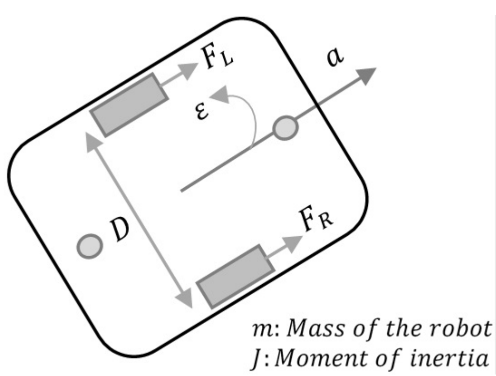

FIG. 2: DYNAMIC MODEL OF KHEPERA IV

From equation (9), $u_1(t) = F_L$ and $u_2(t) = F_R$. Therefore, the equation (13) can be written as:

$$\dot{x}_1(t) = \frac{1}{m}u_1(t) + \frac{1}{m}u_2(t) \tag{14}$$

As it's given from equation (10) $x_2(t) = \omega(t)$. By taking derivation from the equation we have$\dot{x}_2(t) = \varepsilon(t)$. By applying equation (9) the second state equation will be given as:

$$\dot{x}_2(t) = \frac{-D}{2J}u_1(t) + \frac{D}{2J}u_2(t) \tag{15}$$

U_L and U_R have applied to generate the angular velocities ω_L and ω_R respectively. Voltage values of the motors are described by the following equations [4]:

$$J\varepsilon_L(t) + F\omega_L(t) + F_L r = U_L \tag{16}$$

$$J\varepsilon_R(t) + F\omega_R(t) + F_R r = U_R \tag{17}$$

Where ε_L and ε_R are angular accelerations of the wheels and F is the friction force. By choosing third state variable as $x_3(t) = \omega_L$ and fourth state variable as $x_4(t) = \omega_R$ and also by considering U_L and U_R as third and fourth input variables respectively, following state equations will be obtained:

$$\dot{x}_3(t) = -\frac{F}{J}x_3(t) - \frac{r}{J}u_1(t) + \frac{1}{J}u_3(t) \tag{18}$$

$$\dot{x}_4(t) = -\frac{F}{J}x_4(t) - \frac{r}{J}u_2(t) + \frac{1}{J}u_4(t) \tag{19}$$

Therefore, there are four state equations which can be represented in form of state space matrix as below[4]:

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{F}{J} & 0 \\ 0 & 0 & 0 & -\frac{F}{J} \end{bmatrix} x(t) + \begin{bmatrix} \frac{1}{m} & \frac{1}{m} & 0 & 0 \\ \frac{-D}{2J} & \frac{D}{2J} & 0 & 0 \\ -\frac{r}{J} & 0 & \frac{1}{J} & 0 \\ 0 & -\frac{r}{J} & 0 & \frac{1}{J} \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(t) \end{cases} \tag{20}$$

To have the realistic results, real specification of KHEPERA IV has been applied for simulation of the model. The constant parameter values are considered as following[5]:

$m = 0 \cdot 565\, kg, D = 0 \cdot 145\, m$

$r = 0 \cdot 021\, m, J = 0 \cdot 1\, kgm^2$

$F = 0 \cdot 001\, N.$

3- Presenting a Control Model for Trajectory Tracking of KHEPERA IV:

In this section a control method for trajectory tracking of the robot is presented. Control model of the system has two parts: in the first part, coordinates and angel of the robot will be obtained from input angular velocity of the wheels, and in the second part, angular velocities required for the first part of the system with the aim of the best possible tracking of reference trajectory will be generated. For the first part of the system the proposed model in [4] has been applied.

As shown in Fig. 3, there are four input variables applied to state space model of the robot. F_L and F_R are tangential forces which generate the acceleration of the motion by acting on left and right wheel respectively. Third and fourth input variables U_L and U_R are voltage values of DC motors which drive left and right wheel respectively. These voltages have given from PID controllers which control the angular velocity of the wheels. It's been assumed that, the frictional forces applied to the wheels are equal and take constant values. By considering that, the friction forces are the only forces applied on wheels, these inputs can be modeled as Step functions with similar Final Values and similar Step Times. With this consideration, the only variables, which determine the speed of the wheels, are U_L and U_R controlled by PID controllers.

Parameter values for both PIDs have chosen as: $k_p = 10, k_i = 10, k_d = 1$.

These parameters have obtained by observing the tracking quality of input signals and trajectory curve. In the second part of the control model a reference trajectory (x_{ref}, y_{ref} and θ_{ref}) is given to the system and comparing to the output trajectory (x, y and θ) the error of trajectory tracking will be determined. In the next step, the angular velocities ω_L and ω_R have to be generated due to minimizing the error signal. Therefore, input variables to this part of the control model are x_{ref}, y_{ref} and θ_{ref}, and output variables are ω_L and ω_R. To implement LQR optimal control on the system we need to define a mathematical model to determine ω_L and ω_R from x_{ref}, y_{ref} and θ_{ref}. According to equations (5 to 7), the motion equation in matrix form is as following.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} cos\,\theta & 0 \\ sin\,\theta & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{21}$$

By determining v and ω, tangential velocities of the wheels can be calculated from the following equations:

$$v_L = v - \frac{\omega \cdot D}{2} \tag{22}$$

$$v_R = v + \frac{\omega \cdot D}{2} \tag{23}$$

From the given reference trajectory (x_{ref}, y_{ref}), the angle θ_{ref} can be obtained by equation (24):

$$\theta_{ref} = \mathrm{atan}\frac{\dot{y}_{ref}}{\dot{x}_{ref}} + k\pi \quad , \quad k = 0,1 \tag{24}$$

Where $k = 0$ is for forward drive direction and $k = 1$ is for reverse drive direction.

The reference angular velocity ω_{ref} and tangential velocity v_{ref} of the robot can be calculated from the following equations:

$$\omega_{ref} = \frac{\dot{x}_{ref} \cdot \ddot{y}_{ref} - \dot{y}_{ref} \cdot \ddot{x}_{ref}}{(\dot{x}_{ref})^2 + (\dot{y}_{ref})^2} \tag{25}$$

$$v_{ref} = \pm\sqrt{(\dot{x}_{ref})^2 + (\dot{y}_{ref})^2} \tag{26}$$

For calculating tangential velocity, $(+)$ is for forward direction $(\dot{x}_{ref} \geq 0)$, and $(-)$ is for reverse direction $(\dot{x}_{ref} < 0)$.

It's clear that the tangential velocity of robot should be non-zero $(v_{ref} \neq 0)$ because for $v_{ref} = 0$ equation (25) goes to infinity and also θ_{ref} cannot be calculated from equation (24).

When the robot tracks a reference trajectory, several tracking errors will appear in x, y and θ. these errors can be expressed as:

$$\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} x_{ref} - x \\ y_{ref} - y \\ \theta_{ref} - \theta \end{bmatrix} \tag{27}$$

where e_x is error of x position, e_y is error of y position, and e_θ is error of the angle. These errors are in base frame coordinate system. Therefore, to transform the error matrix to the robot coordinate, a rotation matrix has been applied as below:

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} \tag{28}$$

As shown in FIG. 3 a control design based on LQR method is used for determining ω_L and ω_R. Linearized state space form of the system around the operating point O.P (O.P: $e_1 = e_2 = e_3 = 0$), has found from [6] as below:

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} = \begin{bmatrix} 0 & \omega_{ref} & 0 \\ -\omega_{ref} & 0 & v_{ref} \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_{cl} \\ \omega_{cl} \end{bmatrix} \tag{29}$$

Where v_{cl} and ω_{cl} are inputs from the closed-loop. The closed-loop system has three state variablese_1, e_2 and e_3, and two inputs v_{cl} and ω_{cl}. In order to determine inputs of the closed-loop system the LQR optimal control is used. According to the LQR definitions, for the system state space model$\dot{x} = Ax + Bu$, the inputs can be obtained from$u = -K.x$, where K is the gain matrix determined by LQR controller optimally. Therefore, to obtain the inputs of the closed-loop, the equation (30) is available:

$$\begin{bmatrix} v_{cl} \\ \omega_{cl} \end{bmatrix} = -\begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \end{bmatrix} \cdot \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \tag{30}$$

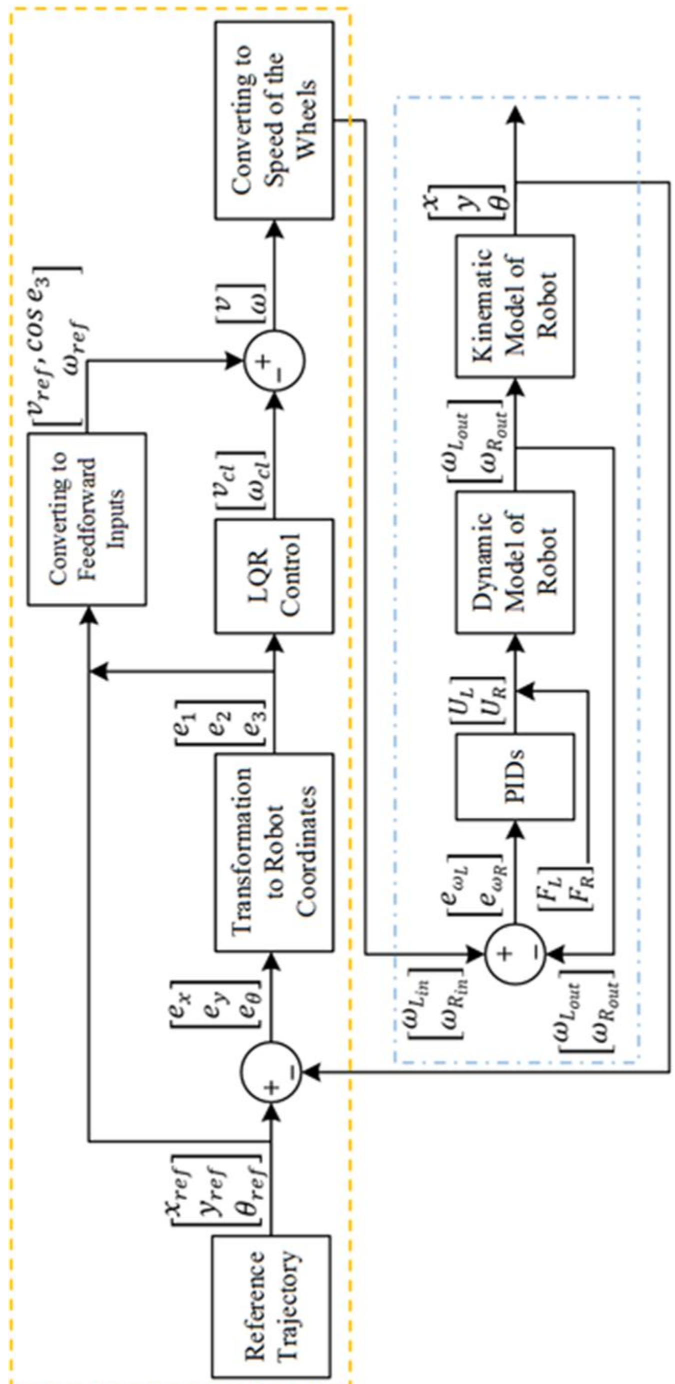

Fig. 3: The Schematic Diagram of Optimal Control System

As discussed in section 1, the LQR controller can be tuned optimally by adjusting the elements of matrices Q and R. For simulating the performance of tracking system, Q and R have taken the following values:

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 10 \end{bmatrix} \quad , \quad R = 0 \cdot 0001 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Commonly used method for obtaining the values of matrices Q and R is trial and error method[7], and this method is applied in this paper. Three elements of the main diagonal of matrix Q belong to the state variablese_1, e_2 and e_3. Therefore, by changing these elements, the sensitivity of the system to the state variables can be adjusted. The elements of the main diagonal of matrix$R$, belong to the control inputs v_{cl} and ω_{cl}.

In this case, increasing the value of the matrix elements, leads to less trajectory tracking quality and by reducing these values rapid changes in input signals will appear, which can lead to instability of the system. After determining v_{cl} and ω_{cl} from equation (30), inputs to robot model, v and ω are obtained from the following equations[6]:

$$v = v_{ref} \cdot \cos e_3 - v_{cl} \tag{31}$$

$$\omega = \omega_{ref} - \omega_{cl} \tag{32}$$

Now by applying equations (22) and (23), tangential speeds of the wheels v_L and v_R are calculated from v and ω, and angular velocities $\omega_L = \frac{v_L}{r}$ and $\omega_R = \frac{v_R}{r}$ are available easily. To evaluate the performance of the system, trajectory tracking error is considered. Tracking error of the system e_t is as expressed in **Error! Reference source not found.**, and it's calculated from equation (33).

$$e_t = \sqrt{(e_x)^2 + (e_y)^2} \tag{33}$$

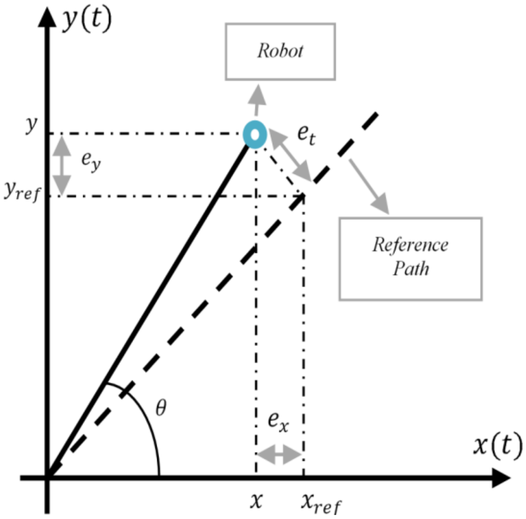

Fig. 4: Trajectory Tracking Error of the System

To examine the system capabilities, several challenging paths are implemented to the control design as reference trajectory. Applied paths are described in Table 1.

TABLE 1 –APPLIED PATHS TO THE ROBOT

No.	Path Name	Specification
1	Circle Path	$x = 0 \cdot 3\, sin(0 \cdot 3t)$ $y = 0 \cdot 3\, cos(0 \cdot 3t)$
2	Ellipse Path	$x = 0 \cdot 5\, sin(0 \cdot 3t)$ $y = 0 \cdot 3\, cos(0 \cdot 3t)$
3	Spiral Path	$x = 0 \cdot 13t\, cos(0.3t)$ $y = 0 \cdot 13t\, sin(0 \cdot 3t)$
4	Eight-Shape Path	$x = 0 \cdot 5\, sin(0 \cdot 3t)$ $y = 0 \cdot 5\, sin(0 \cdot 3t)\, cos(0 \cdot 3t)$
5	Multi-Direction Path	Acute Point: $x = \{0,0 \cdot 2,0 \cdot 1,-0 \cdot 3,0 \cdot 3\}$ $y = \{0,0 \cdot 4,-0 \cdot 2,0,0 \cdot 3\}$

4- Simulation Results:

Simulation has performed with sample time 0.1 second, and for all the paths mentioned in **Error! Reference source not found.**, the initial position of the robot is located in start point of reference trajectory. This provides the linearity condition of the linearized model of equation (29). Figures 5 to 9 show the trajectory tracking (part A), error signal (part B) and angular speed of right and left wheel (part C and D respectively) in input and output of the robot model for each path.

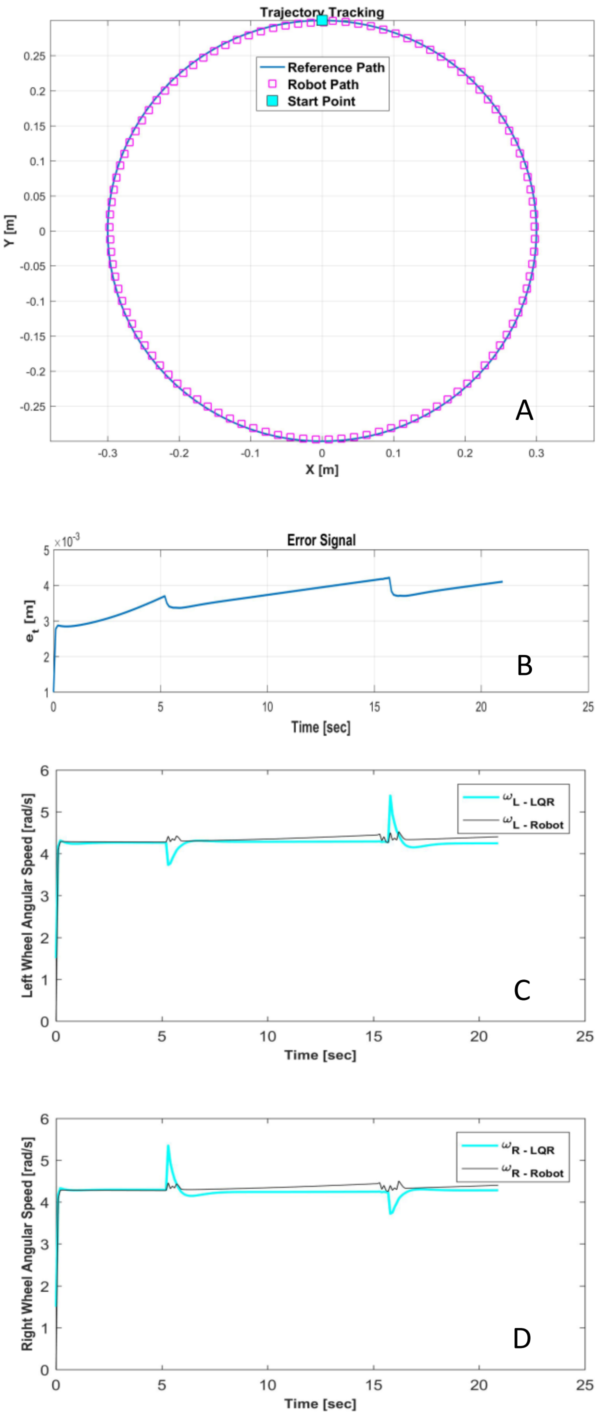

Fig. 5: Circle Path Simulation Results

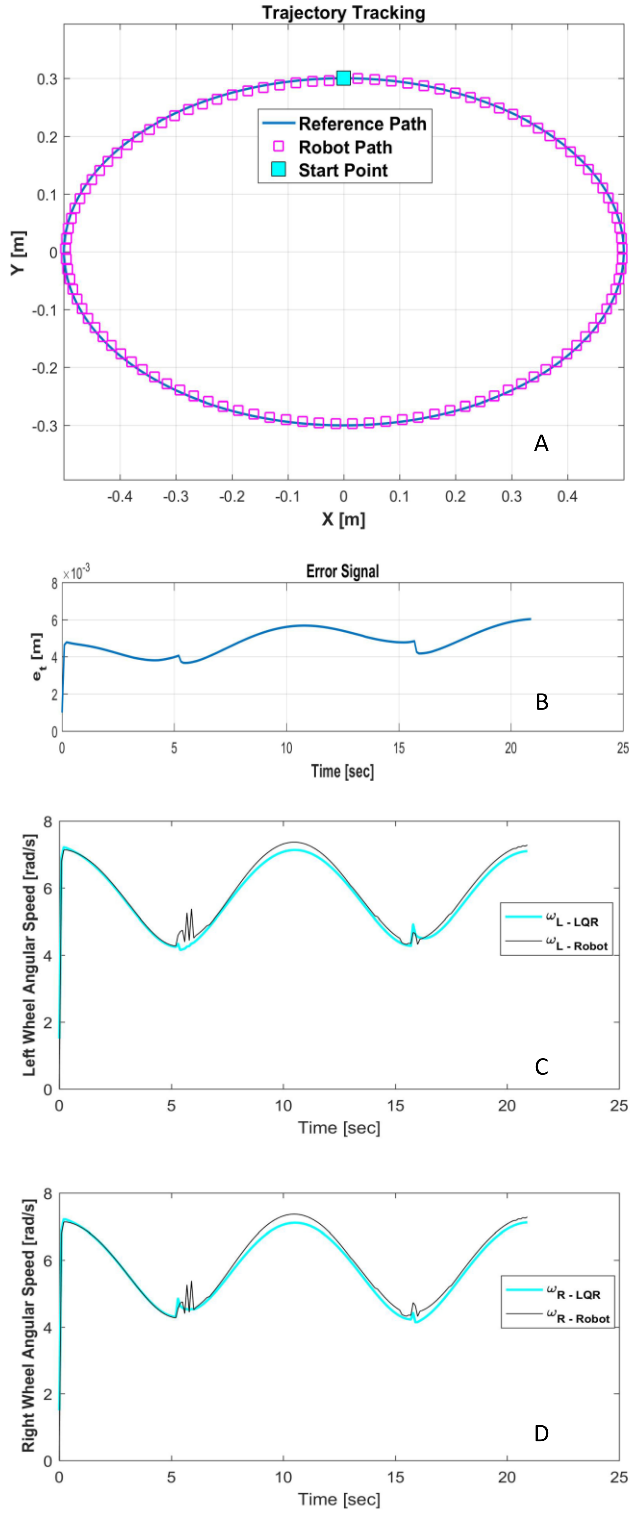

FIG. 6: ELLIPSE PATH SIMULATION RESULTS

Fig. 5 shows the circle path with 1.88 meters of length. In part (B) the error signal has two peak points. These points are related to changing the direction of tangential velocity of the robot which was appeared in equation (26).

In parts (C) and (D) both wheels were able to track input control signal with good performance.

In Fig. 6 the length of the ellipse path was 2.51 meters. Similar to circle path two peak points in error signal were appeared and the system was able to track the applied angular speeds as input control signals.

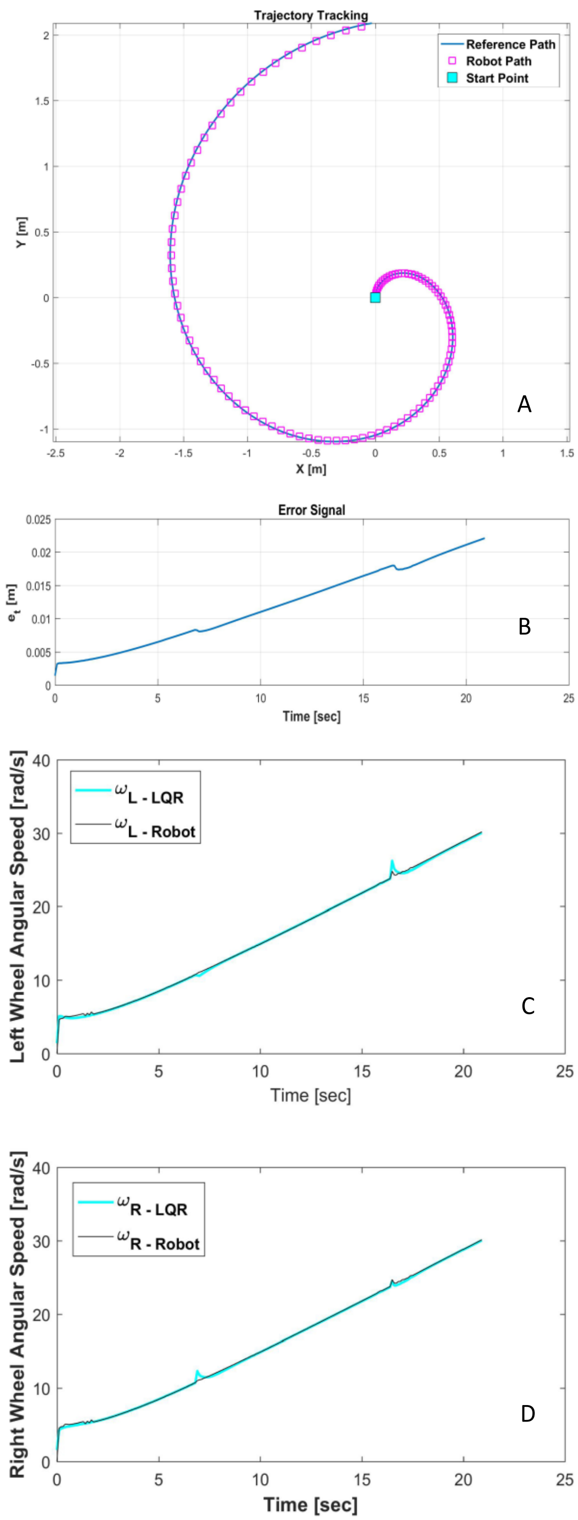

Fig. 7: Ellipse Path Simulation Results

In Fig. 7 a spiral path with the length of 7.05 meters has been applied to the system. The error signal has increasing behavior because the radius of the rotation in at the first of the path is very small and by increasing the speed, physical specification of the robot did not allow it to follow the path curve precisely. Besides, in part (C) and (D) the speed of the wheels showed good tracking accuracy, but the value of the angular speeds are not logical.

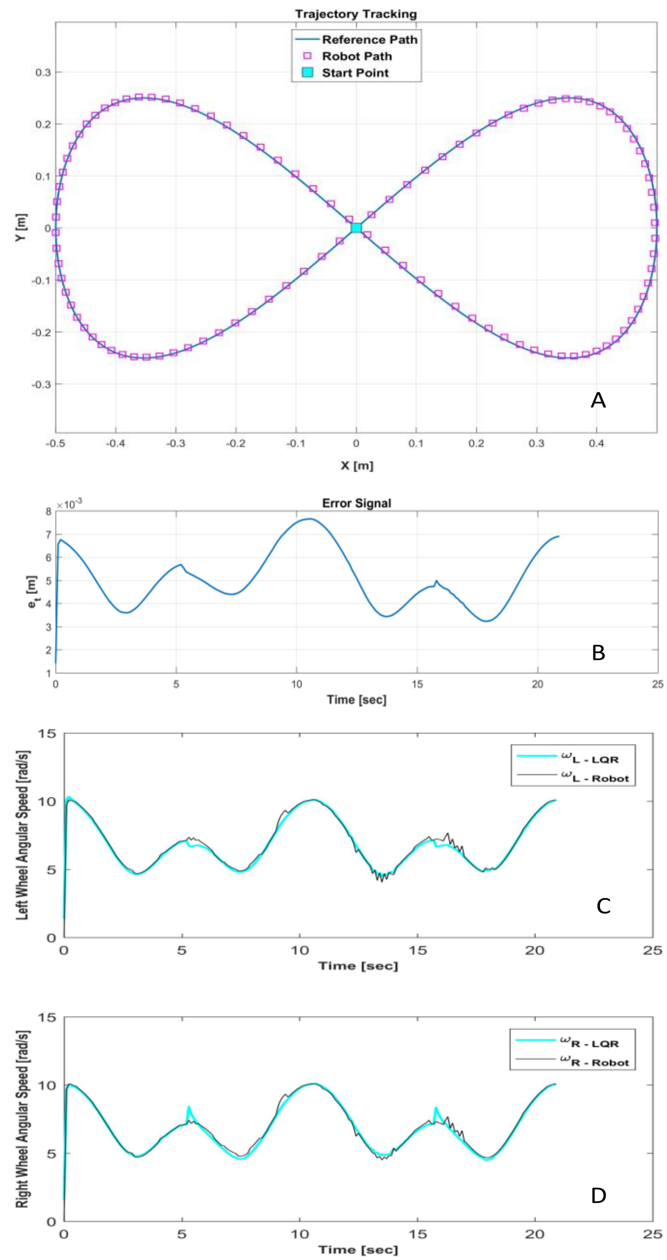

Fig. 8: Ellipse Path Simulation Results

An eight-shape path in Fig. 8 is applied to the system. The length of path is 3.05 meters. In part (B) the error signal has three peak points which indicates to the number of changing the direction of tangential velocity vector of the robot. In parts

(C) and (D) though the input signal has a little oscillation, but the system shows a good tracking behavior.

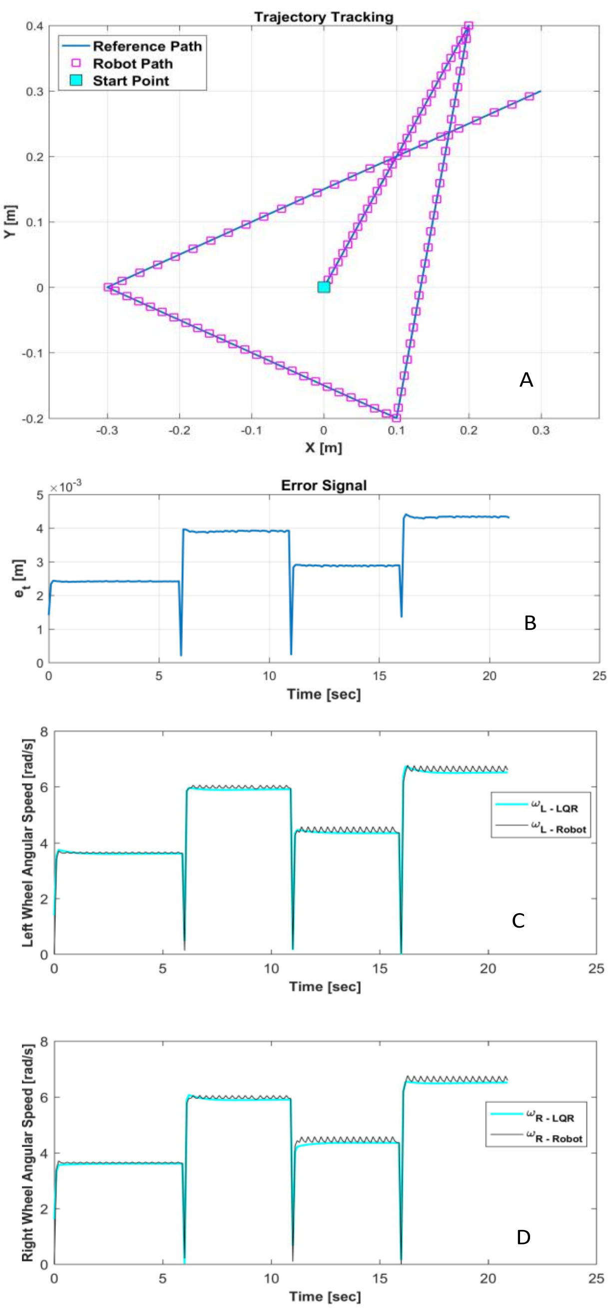

FIG. 9: ELLIPSE PATH SIMULATION RESULTS

Fig. 9 shows a multi-directional path with sharp edges in the path curve and 2.3 meters of length. The system was able to track all the acute points in the path curve and the tracking performance was almost perfect. The result of system tracking error in five different paths is shown in Table 2.

TABLE 2 –MAXIMUM AND MINIMUM OF TRACKING ERROR

Path Name	Maximum Tracking Error (mm)	Minimum Tracking Error (mm)
Circle	4.2	0.5
Ellipse	6	1
Spiral	25	0.4
Eight-Shape	7.8	3.2
Multi-Direction	4.5	0.2

Table 2 shows the minimum and maximum of tracking error signal of each applied path. Considering the information from the table, the best tracking quality based on error signal belongs to the Circle path and the Spiral path shows the worst result.

5- Conclusion

In a previous work on KHEPERA platform [8], a model-free control design for determining the angular velocity of the wheels based on reference trajectory inputs is proposed. In the mentioned paper, determining several parameters such as Gain values and Saturation limits which define the boundary of output angular speeds was difficult. Therefore, in this article, a new method with more elaboration is presented. The proposed control method was based on LQR optimal control and simulation results in figures 5 to 9 showed that the presented model was able to track applied reference trajectories with the satisfying tracking precision and system performance. According to the information from Table 2, the maximum tracking error belongs to Spiral path and the minimum one is related to Circle and Ellipse paths. As seen in Figures 5 to 9 in all of the paths the input signals ω_L and ω_R have some oscillation. It seems that it's because of the sensitivity of the LQR controller to control the error signal. By adjusting the LQR matrices these oscillations can be reduced but it leads to less trajectory tracking quality. Besides, changing the sign of tangential velocity signal leads to several peak points in error signal. It seems that it's because of the overshot of input signals ω_L and ω_R in mentioned sample times. This problem is more sensible in the parts (C) and (D) of Fig. 9 and it may to cause the uncertainties in the real experimental performance of the robot, but seems that the tracking quality would be satisfying.

6- References

[1] k-tream "http://www.k-team.com/mobile-robotics-products/khepera-iv/introduction," kteam mobile robotics, 2016. .

[2] Ogata, K, Yang, and Y, Modern control engineering. 1970.

[3] P. Suster and A. Jadlovská, "Tracking trajectory of the mobile robot Khepera II using approaches of artificial intelligence," Acta Electrotech. Inform., vol. 11, no. 1, p. 38, 2011.

[4] P. Suster and A. Jadlovská, "NEURAL TRACKING TRAJECTORY OF THE MOBILE ROBOT KHEPERA II IN INTERNAL MODEL CONTROL STRUCTURE," Acta Electrotech. Inform., vol. 11, no. 1, p. 38, 2010.

[5] K-Team, KHEPERAIV User manual, Revision 1. K-Team, 2016.

[6] G. Klancar, D. Matko, and S. Blazic, "Mobile robot control on a reference path," in Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, 2005., 2005, pp. 1343–1348.

[7] R. Bhushan, K. Chatterjee, and R. Shankar, "Comparison between GA-based LQR and conventional LQR control method of DFIG wind energy system," in Recent Advances in Information Technology (RAIT), 2016 3rd International Conference on, 2016, pp. 214–219.