Proceedings of the IEEE
International Conference on Automation and Logistics
Shenyang, China August 2009

Design of Linear Quadratic Optimal Controller for Bicycle Robot

Lei Guo, Qizheng Liao, Shimin Wei, Yufeng Zhuang
School of Automation
Beijing University of Posts and Telecommunications
Haidian District, Beijing, 100876, China
guolei@bupt.edu.cn

***Abstract* –The dynamic model of bicycle robot is similar to be linear model when it is moving at a high speed. Aiming at the goal of balancing bicycle robot with high speed, a kind of linear dynamic model of bicycle robot was presented based on a nonlinear dynamic model of bicycle robot. The open–loop stability was analyzed based on the linear dynamic model. And the controllability and observeability of the linear dynamic model was verified. Then the district time model of the bicycle robot was presented based on the continuous time linear model. Then a linear quadratic optimal controller was designed based on linear control theory for the linear dynamic model of bicycle robot. The computer simulation results show the efficiency of the control algorithm. It is practical for the real bicycle robot experiments in future.**

Index Terms - bicycle robot, Linear dynamic model, district time model, linear quadratic optimal control, computer simulation.

I. INTRODUCTION

In the area of robot control, bicycle robot is a hot problem for its statically-unstable and the dynamically-stable feature. And more and more research has been done to study the problem of bicycle robot self balancing. And several kinds of bicycle robot dynamic models were presented by some researchers. A kind of bicycle dynamic model was proposed based on the equilibrium of moment inertia. Then the dynamic model described with a nonlinear differential equation was disposed with approximate linearization method. And a controller was designed with linear control theory in [1]. Neil H.Getz proposed a simple bicycle model an internal equilibrium controller was designed in [2]. And a bicycle robot dynamic model is presented which is described with two nonlinear differential equations and utilized intelligent control theory such as online reinforcement learning in [3]. Based on the linear feature of the bicycle moving with high speed , a linear dynamic model of bicycle robot and the control algorithm was presented in [4] [5]. A kind of nonlinear bicycle robot model was presented in [6], based on Lagrange method. And the nonlinear dynamic model is different and much more accurate to those models presented in [1][2][3][4][5].Based on the nonlinear dynamic bicycle robot model presented in [6], it is sure that the linear dynamic model of the bicycle robot could be presented by the means of approximate linearization at the zero point. And the district time model could be presented based on the linear continues time model. Then a linear quadratic optimal controller was designed to balance the robot. So, we can balance and drive a bicycle moving at a high speed by the means of controlling steering angle.

II. LINEAR DYNAMIC MODEL OF BICYCLE ROBOT

A. Variable Denotation

When the robot is falling down to the ground, in order to keep the robot vertical to the ground, the front wheel of the bicycle should be rotated a relevant angle (steering angle α). Let m_1 denote the mass of wheel. m_2 means the mass of the triangle frame of the bicycle. Let *r* be the radius of the wheel. Let λ be the distance between the COG of front wheel and the rotation axis of the front fork. Let β denote the rolling angle of the COG of the bicycle with respect to the vertical direction. Let α denote the steering angle of the rotation of front fork in order to keep the robot vertical. Let *h* be the height of COG of the bicycle.

B. The Nonlinear Bicycle Dynamic Model

The nonlinear bicycle dynamic model presented in [6] is represented as follows.

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = c_1 cosx_3 \dot{x}_4 + c_2 sinx_3 x_4^2 + c_3 u \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = d_1 cosx_3 \dot{x}_2 + d_2 sinx_3 \end{cases} \tag{1}$$

In which, $x_1 = \alpha$, $x_2 = \dot{\alpha}$, $x_3 = \beta$, $x_4 = \dot{\beta}$,

$c_1 = \dfrac{-(m_1 r + m_2 h\sigma)\lambda}{\frac{1}{2}m_1 r^2 + m_1\lambda^2 + m_2\sigma^2\lambda^2} = -c_2$, $c_3 = \dfrac{1}{\frac{1}{2}m_1 r^2 + m_1\lambda^2 + m_2\sigma^2\lambda^2}$,

$d_1 = \dfrac{-(m_1 r + m_2 h\sigma)\lambda}{3m_1 r^2 + 2m_2 h^2}$, $d_2 = \dfrac{(m_2 h + 2m_1 r)g}{3m_1 r^2 + 2m_2 h^2}$.

In which, *u* is the steering control torque of the robot.

Equation (1) can be identically transformed as follows.

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = [0.5 \cdot d_2 c_1 sin(2x_3) + c_2 sinx_3 x_4^2 + c_3 u]/(1 - d_1 c_1 cos^2 x_3) \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = [0.5 \cdot d_1 c_2 sin(2x_3) x_4^2 + d_1 c_3 cosx_3 u + d_2 sinx_3]/(1 - d_1 c_1 cos^2 x_3) \end{cases}$$

(Equation (2))

978-1-4244-4795-4/09/$25.00

The output equation is presented as follows. $y = x_3$

Let $C(x_3) = 1 - d_1 c_1 cos^2 x_3$. Given the specific value of the coefficients (c_1, c_2, c_3, d_1, d_2), it can be verified that for any $x_3 \in R$ there is always that $C(x_3) \neq 0$.

C. The Linear Bicycle Dynamic Model

Considering that the bicycle robot moving with high speed, it is easy to understand that the rolling angle $x_3 = \beta$ is relatively small. Then we can have the approximate relations presented as follows.

$sin\, x_3 \approx x_3, cos\, x_3 \approx 1$

Based on those relations, (2) can be represented as follows:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = (d_2 c_1 x_3 + c_2 x_3 x_4^2 + c_3 u)/(1 - d_1 c_1) \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = (d_1 c_2 x_3 x_4^2 + d_2 x_3 + d_1 c_3 u)/(1 - d_1 c_1) \end{cases} \quad (3)$$

For the reason of the rolling angle $x_3 = \beta$ is relatively small at the zero point, we can ignore the high rank small variables. So we can have the approximate relations presented as follows.

$x_3 x_4^2 = \beta \dot{\beta}^2 \approx 0$

Then, (3) can be represented as follows:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = (d_2 c_1 x_3 + c_3 u)/(1 - d_1 c_1) \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = (d_2 x_3 + d_1 c_3 u)/(1 - d_1 c_1) \end{cases} \quad (4)$$

Transfer (4) into matrix forms, (4) can be shown as follows.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{d_2 c_1}{1 - d_1 c_1} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{d_2}{1 - d_1 c_1} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{c_3}{1 - d_1 c_1} \\ 0 \\ \frac{d_1 c_3}{1 - d_1 c_1} \end{bmatrix} u \quad (5)$$

$$y = [1 \quad 0 \quad 0 \quad 0] \cdot [x_1 \quad x_2 \quad x_3 \quad x_4]^T$$

Based on the specific coefficients shown in table I, we can get the actual numerical values in (5). Then (5) could be presented as follows.

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + Du \end{cases}$$

In which, $\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -10.9426 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 5.79882 & 0 \end{bmatrix}$, $\boldsymbol{B} = \begin{bmatrix} 0 \\ 7.06703 \\ 0 \\ -0.0613 \end{bmatrix}$,

$\boldsymbol{C} = [1 \quad 0 \quad 0 \quad 0], D = 0, \boldsymbol{x} = [x_1 \quad x_2 \quad x_3 \quad x_4]^T, \boldsymbol{y} = x_1$.

D. The Analysis of Open-loop System Stability

The characteristic equations of system described with (5) could be shown as follows.

$|\lambda I - \boldsymbol{A}| = 0$

The eigenvalue of the open-loop system can be presented as follows by solving the equation which is shown above.

$\lambda_1 = \lambda_2 = 0, \lambda_3 = 2.4081, \lambda_4 = -2.4081$

It is easy to find that the open-loop system have four poles. The system has tow poles are at the zero point, have a pole in the left plane and another pole in the right plane. Then we can get the conclusion that the open-loop system is unstable.

E. The Analysis of Open-loop System Controllability

Theorem The states equation of a single input linear system is presented as follows $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}$ .In which, $\boldsymbol{A}$ denotes a $n \times n$ dimensional matrix, $\boldsymbol{B}$ denotes a $n \times r$ dimensional matrix, u denotes a $r \times 1$ dimensional vector field. When the rank of matrix $\boldsymbol{M} = [\boldsymbol{B} \quad \boldsymbol{AB} \quad \boldsymbol{A}^2\boldsymbol{B} \quad \cdots \quad \boldsymbol{A}^{n-1}\boldsymbol{B}]$ is n, the system is controllable.The matrix $\boldsymbol{M}$ of system described with (5) could be shown as follows.

$$\boldsymbol{M} = \begin{bmatrix} 0 & 7.067 & 0 & 0.671 \\ 7.067 & 0 & 0.671 & 0 \\ 0 & -0.0613 & 0 & -0.3556 \\ -0.0613 & 0 & -0.3556 & 0 \end{bmatrix}$$

It is obvious that the system described with (5) is controllable.

F. The Analysis of Open-loop System Controllability

Theorem The states equation of a single input linear system is presented as follows. $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}$, $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}$ .In which, $\boldsymbol{A}$ denotes a $n \times n$ dimensional matrix, $\boldsymbol{B}$ denotes a $n \times r$ dimensional matrix, u denotes a $r \times 1$ dimensional vector field, $\boldsymbol{C}$ denotes a $m \times n$ dimensional matrix. The observable matrix $\boldsymbol{N}$ is presented as follows.

$\boldsymbol{N} = [\boldsymbol{C} \quad \boldsymbol{CA} \quad \cdots \quad \boldsymbol{CA}^{n-1}]^T$

The system is observable in condition of the rank of matrix $\boldsymbol{N}$ is n .The matrix $\boldsymbol{N}$ of system which is described with (5) could be shown as follows.

$$\boldsymbol{N} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -10.9426 & 0 \\ 0 & 0 & 0 & -10.9426 \end{bmatrix}$$

The rank of matrix $\boldsymbol{N}$ is presented as follows.

$rank(N) = n = 4$

It is obvious that the system which is described with (5) is observable.

III. Design of Linear Quadratic Optimal Controller

The continuous-time system described with (5) can be represented into discrete-time system based on Discrete-time system theory as follows.

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{Gx}(k) + \boldsymbol{H}u(k) \\ \mathrm{y}(k) = \boldsymbol{Cx}(k) + Du(k) \end{cases}$$

$\boldsymbol{G}(T) = e^{AT}$, $\boldsymbol{H}(T) = (\int_0^T e^{AT} d\lambda)\boldsymbol{B}$, $\boldsymbol{C} = [1 \quad 0 \quad 0 \quad 0]$, $D = 0$.

The sampling period of the discrete-time system is set to be $T = 1/15s$ to satisfy Shannon sampling theorem.

Then the matrix $\boldsymbol{G}$ could be represented as follows.

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0.1 & -0.055 & -0.0018 \\ 0 & 1 & -1.1049 & -0.055 \\ 0 & 0 & 1.0291 & 0.101 \\ 0 & 0 & 0.5885 & 1.0291 \end{bmatrix}, \boldsymbol{H} = \begin{bmatrix} 0.0353 \\ 0.7068 \\ -0.0003 \\ -0.0062 \end{bmatrix}$$

The control system diagram of the discrete-time system is presented as follows.

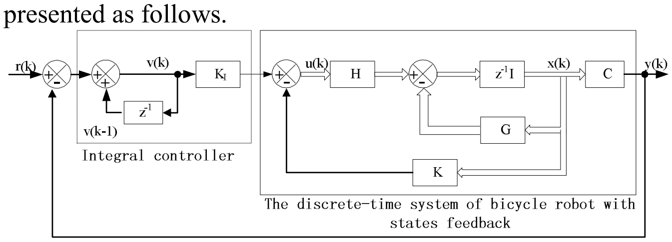


Fig. 1 Control system diagram of the discrete-time system

The discrete-time states equations of the control system the is presented as follows.

$$\boldsymbol{x}(k+1) = \boldsymbol{Gx}(k) + \boldsymbol{H}u(k) \tag{6}$$

$$y(k) = \boldsymbol{Cx}(k) \tag{7}$$

$$v(k) = v(k-1) + r(k) - y(k) \tag{8}$$

$$u(k) = -\boldsymbol{Kx}(k) + K_I v(k) \tag{9}$$

In which, $\boldsymbol{K} = [k_1 \quad k_2 \quad k_3 \quad k_4]$.

Equation (8) could be represented as follows.

$$v(k+1) = v(k) + r(k+1) - y(k+1) \tag{10}$$

Equation (7) could be represented as follows.

$$y(k+1) = \boldsymbol{Cx}(k+1) \tag{11}$$

Let (6) substitute to (11),we get,

$$y(k+1) = \boldsymbol{C}[\boldsymbol{Gx}(k) + \boldsymbol{H}u(k)] \tag{12}$$

Let (12) substitute to (10),we get,

$$\begin{aligned} v(k+1) &= v(k) + r(k+1) - \boldsymbol{C}[\boldsymbol{Gx}(k) + \boldsymbol{H}u(k)] = \\ &- \boldsymbol{CGx}(k) + v(k) - \boldsymbol{CH}u(k) + r(k+1) \end{aligned} \tag{13}$$

Transferring (6) and (13) into matrix form, we get,

$$\begin{bmatrix} \boldsymbol{x}(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{G} & 0 \\ -\boldsymbol{CG} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(k) \\ v(k) \end{bmatrix} + \begin{bmatrix} \boldsymbol{H} \\ -\boldsymbol{CH} \end{bmatrix} u(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(k+1) \tag{14}$$

Letting input signal be step signal, we get,

$r(k) = r(k+1) = r(\infty) = 1$

In condition of $k \to \infty$, (14) could be represented as follows.

$$\begin{bmatrix} \boldsymbol{x}(\infty) \\ v(\infty) \end{bmatrix} = \begin{bmatrix} \boldsymbol{G} & 0 \\ -\boldsymbol{CG} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(\infty) \\ v(\infty) \end{bmatrix} + \begin{bmatrix} \boldsymbol{H} \\ -\boldsymbol{CH} \end{bmatrix} u(\infty) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(\infty) \tag{15}$$

The system error variables could be defined as follows.

$$\boldsymbol{x}_e(k) = \boldsymbol{x}(k) - x(\infty),\ v_e(k) = v(k) - v(\infty),$$
$$u_e(k) = u(k) - u(\infty) \tag{16}$$

Let (14) subtract (15),we get,

$$\begin{bmatrix} \boldsymbol{x}(k+1) - \boldsymbol{x}(\infty) \\ v(k+1) - v(\infty) \end{bmatrix} = \begin{bmatrix} \boldsymbol{G} & 0 \\ -\boldsymbol{CG} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(k) - \boldsymbol{x}(\infty) \\ v(k) - v(\infty) \end{bmatrix} + \begin{bmatrix} \boldsymbol{H} \\ -\boldsymbol{CH} \end{bmatrix} [u(k) - u(\infty)]$$
$$+ \begin{bmatrix} 0 \\ 1 \end{bmatrix} [r(k+1) - r(\infty)]$$

Let (16) substitute to equation which is shown above, we get,

$$\begin{bmatrix} \boldsymbol{x}_e(k+1) \\ v_e(k+1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{G} & 0 \\ -\boldsymbol{CG} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_e(k) \\ v_e(k) \end{bmatrix} + \begin{bmatrix} \boldsymbol{H} \\ -\boldsymbol{CH} \end{bmatrix} u_e(k) \tag{17}$$

Combining (9) and (16), we get,

$$u_e(k) = -\boldsymbol{Kx}_e(k) + K_I v_e(k) = [-\boldsymbol{K} \quad K_I] \begin{bmatrix} \boldsymbol{x}_e(k) \\ v_e(k) \end{bmatrix} \tag{18}$$

Define variables as follows. $w(k) = u_e(k)$

$\overset{*}{\boldsymbol{G}} = \begin{bmatrix} \boldsymbol{G} & 0 \\ -\boldsymbol{CG} & 1 \end{bmatrix}$, $\overset{*}{\boldsymbol{H}} = \begin{bmatrix} \boldsymbol{H} \\ -\boldsymbol{CH} \end{bmatrix}$, $\overset{*}{K} = [\boldsymbol{K} \quad -K_I]$,

$$\boldsymbol{\xi}(k) = \begin{bmatrix} \boldsymbol{x}_e(k) \\ v_e(k) \end{bmatrix} = [x_{1e}(k) \quad x_{2e}(k) \quad x_{3e}(k) \quad x_{4e}(k) \quad x_{5e}(k)]^T.$$

Based on the definition above and (17), (18), the close-loop system could be presented as follows.

$$\boldsymbol{\xi}(k+1) = \overset{*}{\boldsymbol{G}}\boldsymbol{\xi}(k) + \overset{*}{\boldsymbol{H}} w(k),\ w(k) = -\overset{*}{\boldsymbol{K}}\boldsymbol{\xi}(k)$$

The indicators of linear quadratic could be presented as follows.

$$J = \frac{1}{2}\sum_{k=0}^{\infty} [\boldsymbol{\xi}^T \boldsymbol{Q} \boldsymbol{\xi} + w\boldsymbol{R}w]$$

To realize the goal of controlling the variables x_{1e} and x_{3e} effectively, the matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$ are presented as follows.

$$\boldsymbol{Q} = \begin{bmatrix} 15 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 90 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \boldsymbol{R} = 1.$$

Based on computation, the matrix $\overset{*}{\boldsymbol{K}}$ which can make the value of J to be minimum is presented as follows.

$$\overset{*}{\boldsymbol{K}} = [1.085 \quad 0.2176 \quad -273.521 \quad -98.519 \quad -0.2141]$$

IV. Computer Simulation

Let (9) substitute to (6), we get,

$$\boldsymbol{x}(k+1) = (\boldsymbol{G} - \boldsymbol{HK})\boldsymbol{x}(k) + \boldsymbol{H}K_I v(k) \tag{19}$$

Let (9) substitute to (13), we get,

$$v(k+1) = (-\boldsymbol{CG} + \boldsymbol{CHK})\boldsymbol{x}(k) + (1 - \boldsymbol{CH}K_I)v(k) + r(k+1) \tag{20}$$

Combining (19) with (20) and transferring them into matrix form, we get,

$$\begin{bmatrix} \boldsymbol{x}(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{G} - \boldsymbol{HK} & \boldsymbol{H}K_I \\ -\boldsymbol{CG} + \boldsymbol{CHK} & 1 - \boldsymbol{CH}K_I \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(k) \\ v(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(k+1) \tag{21}$$

Computer simulation is achieved based on the simulink toolbox of Matlab.

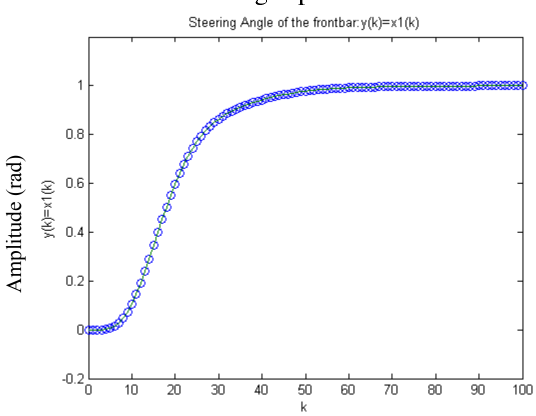
The input signal is step signal. The sampling period of the discrete-time system is set to be $T = 1/15s$.

Based on (21), the steering angle response x_1 of the linear dynamic model with linear quadratic optimal controller is shown in Fig.2. We can learn from Fig.2 that the response x_1 of robot system will be in steady-state in about 2 s.

The relationship of the rolling angle response x_3 of the linear dynamic model with linear quadratic optimal controller between discrete-time variable k is shown in Fig.3. We can learn from Fig.3 that the response of x_3 will reduce to zero in about 2 s.

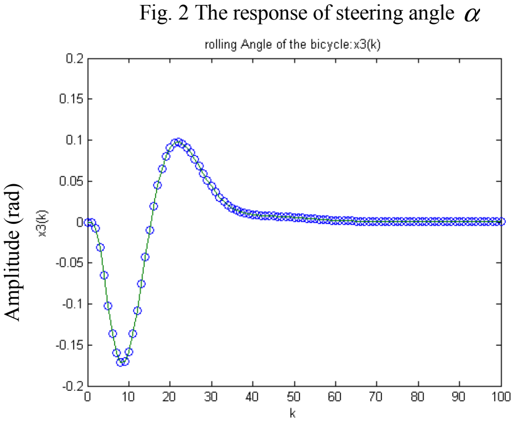
From the result of simulation we can get conclusion that the linear quadratic optimal controller can stabilize the robot system. Not only the system state variable x_1 is steady but also x_3 is steady with the linear quadratic optimal controller.

The simulation experiments of the nonlinear system (2) with the linear quadratic optimal controller were been done based on Matlab software. And the nonlinear system response of rolling angle β with linear quadratic optimal controller is shown in Fig.4. We can learn from Fig.4 that the robot system will fall on the ground in about five seconds. That result is not very perfect. The essence reason might be the natural nonlinear characters of the bicycle robot. However, the nonlinear system will be similar with the linear system when bicycle robot moves with high speed.



time (1/15 s)

Fig. 2 The response of steering angle α



time (1/15 s)

Fig .3 The response of rolling angle β

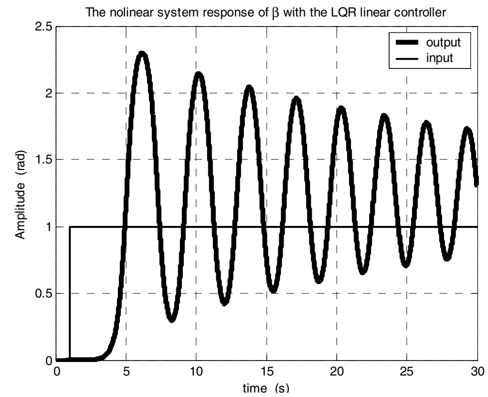


Fig. 4 The nonlinear system response of rolling angle β

V. DESIGN OF THE BICYCLE ROBOT

A real bicycle robot which is built by us is shown in Fig.5. A step motor is fixed on the top of the triangle frame of the bicycle to realize the goal of steering control. Another DC motor is fixed on the bicycle to drive the rear wheel. A tilt sensor is fixed on the bicycle to measure the rolling angle β and the roll angle velocity. And a potentiometer was fixed on the bicycle to measure the steering angle. A control circuit based on the Philips LPC2292 MCU is fixed on the robot to fulfill the task of controlling the steering motor and the driving motor based on the relevant control algorithm. To fulfill the several tasks such as sensor data sampling and motor control & control algorithm computation, the real time embedded operation system $\mu C/OSII$ is transplanted to the control circuit.

Fig.5 The picture of bicycle robot

Several experiments in which the bicycle robot was controlled with the linear quadratic optimal controller were been done. The experiment result is shown in Fig.6.The left subgraph is the response curve of the steering angle sampled by the potentiometer. The right subgraph in Fig.6 is the response curve of the rolling angle sampled by the tilt sensor. We can learn from Fig.6 that the bicycle robot can only be steady for a while. Then it will fall on the ground. Even some little noise from experiment environment such as the disturbance aroused by wind will make it unstable. And the sensor noise will affect the robot adversely. What's more, we can not let the bicycle robot run at a speed which is high enough without efficacious protection. Although we have done some work, we still have a long way to go.

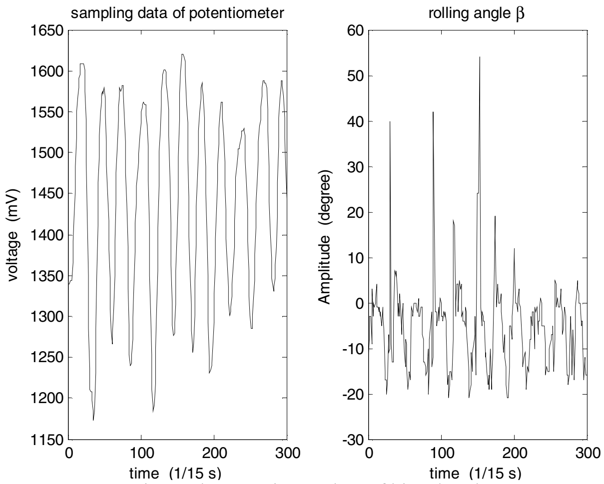


Fig. 6 The experiment data of bicycle robot

TABLE I
THE NUMERICAL VALUE OF THE ROBOT COEFFICIENT

Name	Physical sense	Value
m_1	Mass of the wheel	2.5kg
m_2	Mass of the triangle frame	18kg
m_3	Mass of the balance device	5kg
r	Radius of the wheel	0.33m
λ	A constant distance of steering device	0.04m
h	Height of COG	0.92m
k_1	The distance between A and COG	0.7m
k_2	The distance between A and C	1.1m
σ	$(k_2 - k_1)/k_2$	0.36

VI. Conclusion

Computer simulation testifies that the algorithm which is presented can keep the robot steady in the range of $[-9.7^{\circ}, 5.7^{\circ}]$. The simulation result is accordant to the physical sense of the actual bicycle robot which runs at a high speed. And the linear quadratic optimal controller has the feature of global stability. The controller can stabilize both the state variable α and β as well.

Acknowledgment

The author wishes to thank the IEEE for providing this template and all colleagues who previously provided technical support.

References

[1] Yasuhito Tanaka, Toshiyuki Murakami. "Self sustaining bicycle robot with steering controller," *the 8th IEEE International Workshop on Advanced Motion Control*, *Kawasaki International Center, Kawasaki, Japan,* March, 2004. pp.193-197.

[2] Neil H. G. "Internal equilibrium control of a bicycle," *Proceedings of the 34th Conference on Decision & Control, New Orieans, IEEE*. NH Getz and JE Marsden.1995.pp. 4285-4287.

[3] Jette Randløv, Preben Alstrøm. "Learning to drive a bicycle using reinforcement learning and shaping," *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998)*, Madison, Wisconson, USA, July, 1998. pp.463-471

[4] Shashikanth Suryanrayanan, Masayoshi Tomizula and Matt Weaver. "System dynamics and control of bicycles at high speeds," *Proceedings of the American Control Conference*. Anchorage, AK, 2002:845-850.

[5] *Karl J. Åström, Richard E. Klein, Anders Lennartsson. "Bicycle dynamics and control ," IEEE Control Systems Magazine, 2005, 25(4).pp.26-47*

[6] Lei Guo, Qizheng Liao,Shimin Wei. "Design of motion controller for bicycle robot based on DFL nonlinear control method," *Proceedings of the 6th World Congress on Intelligent Control and Automation (WCICA 06),Vol 11,DaLian,China,June, 2006. pp. 9047-9011*