

HYBRID PD AND ADAPTIVE BACKSTEPPING CONTROL FOR SELF-BALANCING TWO-WHEEL ELECTRIC SCOOTER

NGUYEN NGOC SON¹, HO PHAM HUY ANH²

¹*Faculty of Electronic Engineering, Industrial University of Ho Chi Minh City, Vietnam;
nguyennhocson@iuh.edu.vn*

²*Faculty of Electrical and Electronic Engineering,
Ho Chi Minh City University of Technology, Vietnam;
hphanh@hcmut.edu.vn*

Abstract. This paper proposes a combination of adaptive self-balancing controller and the left and right turning PD controller for self-balancing two-wheel electric scooter (eScooter). An adaptive self-balancing controller is synthesized by the backstepping approach and the Lyapunov stability theory. The proposed adaptive controller allows the design of a feedback control that stabilizes self-balancing control of eScooter in the presence of uncertainty and perturbation. Additionally, the sensor signals are treated by Kalman filters and the CAN networks are applied to communication among modules of eScooter. Simulation and experiment results are shown to analyze and validate the performance of proposed controller.

Keywords. Adaptive backstepping control, Kalman filter, self-balancing two-wheel robot, CAN networks, embedded system.

1. INTRODUCTION

In control theory, the backstepping control is a technique developed in 1990 by Petar V.Kokotovic and others [5,6,9] for designing stable control applied to a special class of nonlinear dynamic systems. Backstepping control method, based on the Lyapunov design approach, is efficiently applied when higher derivative appearance in presence of uncertainty and perturbation. The key idea of adaptive backstepping technique is to drive the error equation to zero by designing Lyapunov stability approach, by using the recursive structure to seek the controlled function. Hence the adaptive backstepping method induces a feedback control rule that ensures to efficiently control the nonlinearity of the plant.

The eScooter based on inverted pendulum model is a highly nonlinear system with uncertain parameters, which is very difficult to control with six variable state parameters. The eScooter is composed of two coaxial wheels which are mounted parallel to each other and are operated by two brushless DC electric motors (BLDC motors). Accelerometer and gyro sensor permit to determine the pitch angle. In addition, potentiometer is used to measure the yaw angle of eScooter. Furthermore, CAN networks are applied to communicate between control module and display module implemented on the eScooter. By this way it can carry the human load up to 85 Kg. The main characteristic of proposed eScooter is self-balancing capability. This feature helps the eScooter always in equilibrium, despite eScooter equipped only one axis with two wheels. The driver commands an eScooter to go forward by shifting their body forward on the platform, and go backward by shifting their body

backward, respectively. Furthermore, in order to turn, the driver needs to guide the handlebar to the left or the right.

Up to now, some research results published on the world about a self-balancing two-wheel robot (a small and compact robot model, can't transport people) focused on the following issues. The modeling and identification of a self-balancing two-wheel robot is investigated in [4, 7, 10, 11, 14]. The control problem of a self-balancing two-wheel robot, based on the linear control methods, is presented in [7, 11]. Nonlinear intelligent control of a self-balancing two-wheel robot is introduced in [8, 14]. Backstepping control method of a self-balancing two-wheel robot is investigated in [1, 3, 12–15]. Kalman filter applied to the filter of the sensor noise is introduced in [2]. The main drawback of these researches is focused only in a small and compact self-balancing robot model which can't transport people. To overcome this drawback, this paper introduces the adaptive backstepping control to design a novel controller for eScooter which can transport people up to 85 kg.

The paper is organized as follows: Section 2 describes the mathematical model of proposed eScooter. Section 3 introduces the proposed controller design and then presents simulation results. Section 4 introduces the hardware set up, particularly focused in the sensor selection, the associated algorithms and the communicating CAN networks. The verification of the proposed controller applied to real-time eScooter implementation is experimented. Finally, conclusion is presented in Section 5.

2. MATHEMATICAL MODEL OF ESCOOTER

In this section, Newton method is applied to determining the mathematical model of eScooter, [7, 11]. Figure 1 shows the coordinate system of eScooter.

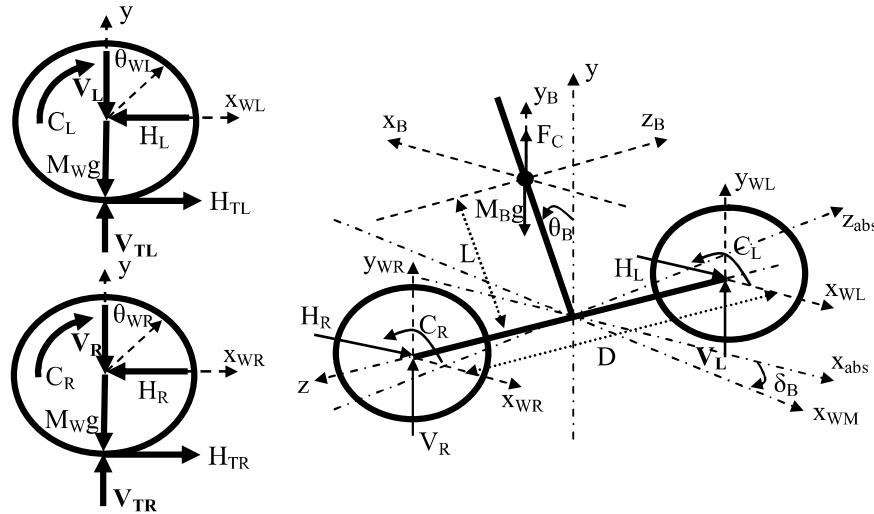


Figure 1: Coordinate system of the eScooter

For the left wheel of eScooter (same as the right wheel)

$$M_W \ddot{x}_{WL} = H_{TL} - H_L \quad (1)$$

$$M_W \ddot{y}_{WL} = V_{TL} - V_L - M_W g \quad (2)$$

$$J_{WL}\ddot{\theta}_{WL} = C_L - H_{TL}R \quad (3)$$

$$x_{WL} = \theta_{WL}R \quad (4)$$

$$J_{WL} = \frac{1}{2}M_{WL}R^2 \quad (5)$$

$$\delta = \frac{x_{WL} - x_{WR}}{D} \quad (6)$$

For the body of eScooter

$$M_B\ddot{x}_B = H_L + H_R \quad (7)$$

$$M_B\ddot{y}_B = V_L + V_R - M_Bg + \frac{C_L + C_R}{L} \sin \theta_B \quad (8)$$

$$J_B\ddot{\theta}_B = (V_L + V_R)L \sin \theta_B - (H_L + H_R)L \cos \theta_B - (C_L + C_R) \quad (9)$$

$$x_B = L \sin \theta_B + \frac{x_{WL} + x_{WR}}{2} \quad (10)$$

$$y_B = -L(1 - \cos \theta_B) \quad (11)$$

$$J_B = \frac{1}{3}M_B L^2 \quad (12)$$

$$\theta = \theta_B = \theta_W = \theta_{WL} = \theta_{WR} \quad (13)$$

$$x_{WM} = \frac{x_{WL} + x_{WR}}{2} \quad (14)$$

$$J_\delta \ddot{\delta} = \frac{D}{2} (H_L - H_R) \quad (15)$$

where H_{TL} , H_{TR} , H_L , H_R , V_{TL} , V_{TR} , V_L , V_R represents reaction forces between the different free bodies. The symbols and definitions of all eScooter's parameters are tabulated in Table 1.

Symbol	Value [Unit]	Parameter
θ	[rad]	Pitch angle
δ	[rad]	Yaw angle
M_w	7[kg]	Mass of wheel
M_B	[kg]	Mass of body
R	0.2[m]	Radius of wheel
L	[m]	Distance between the z axis and the gravity center of eScooter
D	0.6[m]	Distance between the contact patches of the wheels
g	9.8[m/s ²]	Gravity constant
C_L, C_R	[N.m]	Input torques of the right and left wheels
H_{TL}, H_{TR}	[N]	Friction between the ground and the right and left wheels
H_L, H_R	[N]	Reaction forces impact on the right and left wheels
J_{TL}, J_{TR}	[N.m]	Inertial moment of the rotating masses with respect to the z axis
θ_{WL}, θ_{WR}	[rad]	Pitch angle of the right and left wheels
J_B	[N.m]	Inertial moment of the chassis with respect to the z axis

Table 1: Parameters of eScooter are used in simulation and experiment

Substituting (7), (8) and (13) into (9), results in

$$J_B \ddot{\theta} = M_B (\ddot{y}_B \sin \theta - \ddot{x}_B \cos \theta) + M_B g L \sin \theta - (C_L + C_R)(1 + \sin^2 \theta) \quad (16)$$

From (10), (11) and (14), we infer

$$\ddot{y}_B \sin \theta - \ddot{x}_B \cos \theta = -L \ddot{\theta} - \ddot{x}_{WM} \cos \theta \quad (17)$$

Substituting (17) and (12) into (16), yields

$$\frac{4}{3} M_B L^2 \ddot{\theta} + M_B L \cos \theta \ddot{x}_{WM} = M_B g L \sin \theta - (1 + \sin^2 \theta) C_\theta \quad (18)$$

where $C_\theta = C_L + C_R$. From (1), we infer

$$M_W (\ddot{x}_{WL} + \ddot{x}_{WR}) = -(H_L + H_R) + (H_{TL} + H_{TR}) \quad (19)$$

Substituting (3) and (7) into (19), results in

$$M_W (\ddot{x}_{WL} + \ddot{x}_{WR}) = -M_B \ddot{x}_B + \frac{C_L + C_R - (J_{WL} \ddot{\theta}_{WL} + J_{WR} \ddot{\theta}_{WR})}{R} \quad (20)$$

or $2M_W \ddot{x}_{WM} = -M_B \ddot{x}_B + \frac{C_\theta}{R} - 2 \frac{J_W \ddot{\theta}}{R}$

From (10) and (14), we derive

$$\ddot{x}_B = \ddot{\theta} L \cos \theta - \dot{\theta} L \sin \theta + \ddot{x}_{WL} \quad (21)$$

Substituting (21) and (5) into (20), yields

$$(M_B L \cos \theta + M_W R) \ddot{\theta} + (2M_W + M_B) \ddot{x}_{WM} = \dot{\theta}^2 M_B L \sin \theta + \frac{C_\theta}{R} \quad (22)$$

Solving the system of equations (18) and (22), results in

$$A \ddot{\theta} = B_1 \dot{\theta}^2 + C_1 C_\theta \quad (23)$$

$$A \ddot{x}_{WM} = B_2 \dot{\theta}^2 - C_2 C_\theta \quad (24)$$

On the other hand, from (1), (3) and (4) we have

$$H_L = \frac{C_L}{R} - \ddot{x}_{WL} \left(M_W + \frac{J_{WL}}{R^2} \right) \quad (25)$$

From (6), we get

$$\ddot{\delta} = \frac{\ddot{x}_{WL} - \ddot{x}_{WR}}{D} \quad (26)$$

From (25) and (26), we have

$$H_L - H_R = \frac{C_L - C_R}{R} - D \ddot{\delta} \left(M_W + \frac{J_W}{R^2} \right) \quad (27)$$

Substituting (27) into (15), yields

$$\left(J_{\delta} + \frac{1}{2} D^2 \left(M_W + \frac{J_W}{R^2} \right) \right) \ddot{\delta} = \frac{1}{2} D \frac{C_L - C_R}{2} \quad (28)$$

and,

$$J_W = \frac{1}{2} M_W R^2 \text{ and } J_{\delta} = \frac{1}{3} M_B \left(\frac{D}{2} \right)^2 = \frac{1}{12} M_B D^2. \quad (29)$$

Substituting (29) into (28), results in

$$\ddot{\delta} = C_3 C_{\delta} \quad (30)$$

In summary, the state-space equations of eScooter are described in (23), (24) and (30). Where,

$$\begin{cases} C_{\theta} = C_L + C_R \\ C_{\delta} = C_L - C_R \\ C_2 = \frac{0.75(M_W R + M_B L \cos \theta)(1 + \sin^2 \theta)}{M_B L^2} + \frac{1}{R} \\ C_3 = \frac{6}{(9M_W + M_B)RD} \end{cases}$$

and

$$\begin{cases} A = 2M_W + M_B - \frac{0.75(M_W R + M_B L \cos \theta) \cos \theta}{L} \\ B_1 = \frac{0.75g(2M_W + M_B) \sin \theta}{L} - \frac{0.75M_B L \sin \theta \cos \theta}{L} \dot{\theta}^2 \\ C_1 = - \left(\frac{0.75(1 + \sin^2 \theta)(2M_W + M_B)}{M_B L^2} + \frac{0.75 \cos \theta}{RL} \right) \\ B_2 = \frac{-0.75g(M_W R + M_B L \cos \theta) \sin \theta}{L} + M_B L \sin \theta \dot{\theta}^2 \end{cases}$$

3. PROPOSED CONTROLLER DESIGN

The proposed controller for eScooter system is combined an adaptive self-balancing controller with the left and right turning PD controller. The self-balancing controller based on the adaptive backstepping method is used for controlling eScooter in equilibrium with pitch angle $\theta = 0^\circ$. The PD controller is used for controlling eScooter in turning left and right.

3.1. Left and Right Turning Controller

The general structure of the left turn and right turn PD controller is depicted in Figure 2.

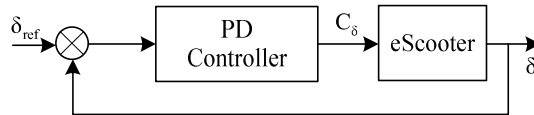


Figure 2: The turning left and right PD controller for eScooter

The main features of the left turn and right turn controller are described as follows

- The reference signal $\delta_{\text{ref}} = 0$
- The PD controller is depicted as $G_c(s) = K_p + K_d s$

Using (30), the transfer function of eScooter is determined

$$G_{\text{yaw}}(s) = \frac{\delta(s)}{C_\delta(s)} = \frac{1}{s^2} C_3 \quad (31)$$

The transfer function describes the overall system

$$G_{\text{overall}}(s) = \frac{(K_p + K_d s)C_3}{s^2 + K_d C_3 s + K_p C_3} \quad (32)$$

For closed-loop system stability, it needs

$$s^2 + K_d C_3 s + K_p C_3 = s^2 + 2\varepsilon\omega_n s + \omega_n^2 \quad (33)$$

where ε is the damping ratio and ω_n represents the natural frequency.

From (33), we derive $K_p = \frac{\omega_n^2}{C_3}$ and $K_d = \frac{2\varepsilon\omega_n}{C_3}$. Thus, we obtain the control signal input function C_δ which is designed as follows

$$C_\delta = -(K_p \delta(t) + K_d \dot{\delta}(t)) \quad (34)$$

3.2. Adaptive Self-balancing Controller

The general structure of the self-balancing controller based on adaptive backstepping method is depicted in Figure 3.

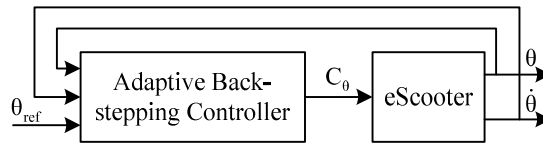


Figure 3: The self-balancing controller for eScooter

First the state variables of the system are defined as: $x_1 = \theta$, $x_2 = \dot{\theta}$. From equation (23), the system's state space equations can be rewritten in the following form:

$$\begin{cases} \dot{x}_1 = x_2 & (a) \\ g(x_1)\dot{x}_2 = C_\theta - h(x_1, x_2) & (b) \end{cases} \quad (35)$$

where $\begin{cases} g(x_1) = \frac{A}{C_1} < 0 \\ h(x_1, x_2) = -\frac{B_1}{C_1} \end{cases}$. The error equation is defined as:

$$e_1 = x_{1ref} - x_1 \quad (36)$$

where θ_{ref} is the referential value of signal θ , which is equal zero for proposed eScooter.

Case 1: Assume that the functions $g(x_1)$ and $h(x_1, x_2)$ are determined. Using the integral backstepping technique to design a self-balancing controller as follows:

Step 1: A virtual control equation α is designed such is that $\lim_{t \rightarrow \infty} e_1(t) = 0$. Virtual control equation is defined as:

$$\alpha = k_1 e_1 + c_1 z_1 + x'' \quad (37)$$

With k_1, c_1 are positive constants and $z_1 = \int_1^e (\tau) d\tau$ is the integral function. By using this equation we can ensure that tracking error converge to zero.

The first Lyapunov function is declared and defined as:

$$V_1 = \frac{c_1}{2} z_1^2 + \frac{1}{2} e_1^2 \quad (38)$$

Take the derivation of V_1 , yields:

$$\dot{V}_1 = c_1 z_1 \dot{z}_1 + e_1 \dot{e}_1 = e_1 (c_1 z_1 + \dot{e}_1) \quad (39)$$

Step 2: Using (35b), we design input value C_θ such is that $\lim_{t \rightarrow \infty} (\alpha - x_2) = 0$. The second error equation is defined as:

$$e_2 = \alpha - x_2 \quad (40)$$

Rewrite equation (35b) in the form of:

$$\dot{x}_1 = \alpha - e_2 = k_1 e_1 + c_1 z_1 + \dot{x}_{1ref} - e_2 \quad (41)$$

From equation (36), it can deduce that:

$$\dot{e}_1 = \dot{x}_{1ref} - \dot{x}_1 = -k_1 e_1 - c_1 z_1 + e_2 \quad (42)$$

By using derivative of e_2 to ensure the desired dynamic feature for the velocity tracking error:

$$\begin{aligned} g \dot{e}_2 &= g \dot{\alpha} - g \dot{x}_2 \\ &= g (k_1 \dot{e}_1 + c_1 \dot{e}_1 + \ddot{x}_{1ref}) - (C_\theta - h(x_1, x_2)) \\ &= g ((c_1 - k_1^2) e_1 - k_1 c_1 z_1 + k_1 e_2 + \ddot{x}_{1ref}) - (C_\theta - h(x_1, x_2)) \end{aligned} \quad (43)$$

Substituting (42) into (40), results in:

$$\dot{V}_1 = e_1 (c_1 z_1 - k_1 e_1 - c_1 z_1 + e_2) = -k_1 e_1^2 + e_1 e_2 \quad (44)$$

Continually, the second Lyapunov function is declared and defined as:

$$V_2 = V_1 + \frac{1}{2} e_2^2 \quad (45)$$

From the derivative of V_2 , we obtain:

$$\dot{V}_2 = \dot{V}_1 + e_2 \dot{e}_2 = -k_1 e_1^2 + e_1 e_2 + e_2 \dot{e}_2 \quad (46)$$

For making \dot{V}_2 , we choose \dot{e}_2 as follows:

$$\dot{e}_2 = -k_2 e_2 - e_1 \text{ with } k_2 \geq 0 \quad (47)$$

Then:

$$g \dot{e}_2 = -g k_2 e_2 - g e_1 \quad (48)$$

Substituting (48) into (46), yields:

$$\dot{V}_2 = -k_1 e_1^2 - k_2 e_2^2 < 0 \quad (49)$$

Finally the control signal input function C_θ is determined as:

$$C_\theta = g((1 + c_1 - k_1^2)e_1 - k_1 c_1 z_1 + (k_1 + k_2)e_2 + \ddot{x}_{1ref}) + h \quad (50)$$

By defining:

$$\begin{cases} \varepsilon = (1 + c_1 - k_1^2)e_1 - k_1 c_1 z_1 + (k_1 + k_2)e_2 + \ddot{x}_{1ref} \\ \tau = \frac{h}{g} \end{cases}$$

Then, the equation (50) is rewritten as

$$C_\theta = g(\varepsilon + \tau)s \quad (51)$$

Case 2: In fact, we can not determine the functions $g(x_1)$ and $h(x_1, x_2)$ because these functions depend on uncertain parameters of eScooter such as: Distance between the z axis and the gravity center of eScooter (L), Mass of body (M_B). Thus the functions $\hat{g}(x_1)$ and $\hat{h}(x_1, x_2)$ are estimated to design a adaptive self-balancing controller. Now, the control signal of adaptive backstepping technique is determined as follows:

$$C_{\theta a} = \hat{g}(\varepsilon + \hat{\tau}) \quad (52)$$

In this case, the second error value e_2 is determined:

$$\begin{aligned} \dot{e}_2 &= (c_1 - k_1^2)e_1 - k_1 c_1 z_1 + k_1 e_2 + \ddot{x}_{1ref} - \frac{1}{g}(C_{\theta a} - h) \\ &= \varepsilon - (e_1 + k_2 e_2) - \frac{1}{g}(C_{\theta a} - h) \end{aligned} \quad (53)$$

On other hand, we have:

$$\begin{aligned} \frac{1}{g}(C_{\theta a} - h) &= \frac{1}{g}(C_\theta - h) + \frac{1}{g}(C_{\theta a} - C_\theta) = \frac{1}{g}(g(\varepsilon + \tau) - h) + \frac{1}{g}(\hat{g}(\varepsilon + \hat{\tau}) - g(\varepsilon + \tau)) \\ &= \varepsilon + \frac{1}{g}(\varepsilon(\hat{g} - g) + \hat{g}\hat{\tau} - g\tau) \end{aligned} \quad (54)$$

By define $\begin{cases} \tilde{g} = g - \hat{g} \\ \tilde{\tau} = \tau - \hat{\tau} \end{cases}$ then

$$\begin{aligned} \frac{1}{g}(C_{\theta a} - h) &= \varepsilon + \frac{1}{g}(\varepsilon(\hat{g} - g) + \hat{g}\hat{\tau} - g(\hat{\tau} + \tilde{\tau})) \\ &= \varepsilon - \frac{\tilde{g}}{g}\varepsilon - \frac{\tilde{g}}{g}\hat{\tau} - \tilde{\tau} = \varepsilon - \frac{\tilde{g}}{g}(\varepsilon + \hat{\tau}) - \tilde{\tau} \end{aligned} \quad (55)$$

Substituting (55) into (53), yields:

$$\dot{e}_2 = \varepsilon - (e_1 + k_2 e_2) - (\varepsilon - \frac{\tilde{g}}{g}(\varepsilon + \hat{\tau}) - \tilde{\tau}) = -(e_1 + k_2 e_2) + \frac{\tilde{g}}{g}(\varepsilon + \hat{\tau}) + \tilde{\tau} \quad (56)$$

Now the adaptive rules can be constructed by using Lyapunov energy function V_3 which is defined as:

$$V_3 = V_2 + \frac{1}{2k_3} \tilde{g}^2 + \frac{1}{2k_4} \tilde{\tau}^2 \quad (57)$$

From the derivation of V_3 , we obtain:

$$\begin{aligned} \dot{V}_3 &= \dot{V}_2 + \frac{1}{k_3} \tilde{g} \dot{g} + \frac{1}{k_4} \tilde{\tau} \dot{\tau} = -k_1 e_1^2 + e_1 e_2 + e_2 \dot{e}_2 + \frac{1}{k_3} \tilde{g} \dot{g} + \frac{1}{k_4} \tilde{\tau} \dot{\tau} \\ &= -k_1 e_1^2 - k_2 e_2^2 + \tilde{g} \left(\frac{1}{g} (\varepsilon + \tilde{\tau}) e_2 + \frac{1}{k_3} \dot{\tilde{g}} \right) + \tilde{\tau} \left(e_2 + \frac{1}{k_4} \dot{\tilde{\tau}} \right) \end{aligned} \quad (58)$$

where k_3, k_4 are positive constant values. The adaptive laws are implemented as:

$$\begin{cases} \dot{\tilde{\tau}} = -k_4 e_2 \\ \dot{\tilde{g}} = -k_3 \frac{1}{g} (\varepsilon + \tilde{\tau}) e_2 \end{cases} \quad (59)$$

Then, the equation (58) is rewritten:

$$\dot{V}_3 = -k_1 e_1^2 - k_2 e_2^2 < 0 \text{ with } \dot{g} = -\dot{\tilde{g}}; \dot{\tau} = -\dot{\tilde{\tau}} \quad (60)$$

Finally the control signal input function $C_{\theta a}$ is determined as:

$$C_{\theta a} = \tilde{g}(\varepsilon + \tilde{\tau}) \text{ with } \begin{cases} \dot{\tilde{\tau}} = k_4 e_2 \\ \dot{\tilde{g}} = k_5 (\varepsilon + \tilde{\tau}) e_2 \\ k_4 > 0, k_5 < 0 \end{cases} \quad (61)$$

In summary, the adaptive backstepping controller has been successfully designed. Based on the equation (61), we see that the control signal input $C_{\theta a}$ doesn't depend on uncertain parameters of eScooter and perturbation.

3.3. Decoupling the input torque for eScooter

By combining the control signal $C_{\theta a}$ (or C_{θ}) of self-balancing controller and the control signal C_{δ} of left and right turning controller, the input torques are applied to eScooter right and left wheels. This scheme is shown as Figure 4.

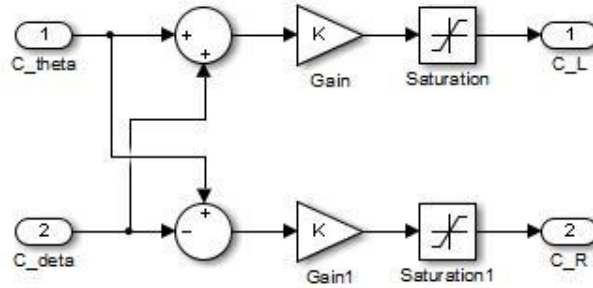


Figure 4: Decoupling of two control signals applied to the two-wheel of eScooter

3.4. Simulation Results

In this section, we analyse and validate the performance of proposed controller. All of the simulations were performed by Matlab version 2009a on an Intel Core i3 computer with a clock rate of 2.53GHz and 2.00GB of RAM. Parameters of eScooter tabulated in Table 1 are used in simulation. The parameters of adaptive backstepping controller are selected as $k_1 = 118$, $k_2 = 26.4$, $c_1 = 0.001$, $c_4 = 0.001$, $c_5 = -0.95$ and the parameters of PD controller are selected as $\varepsilon = 1$, $\omega_n = 10$. Figure 5 illustrates the block diagram of the proposed controller for eScooter.

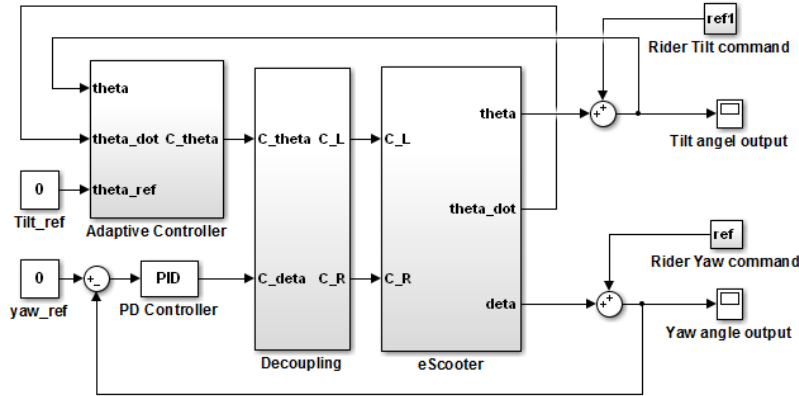


Figure 5: Block diagram of proposed controller for eScooter

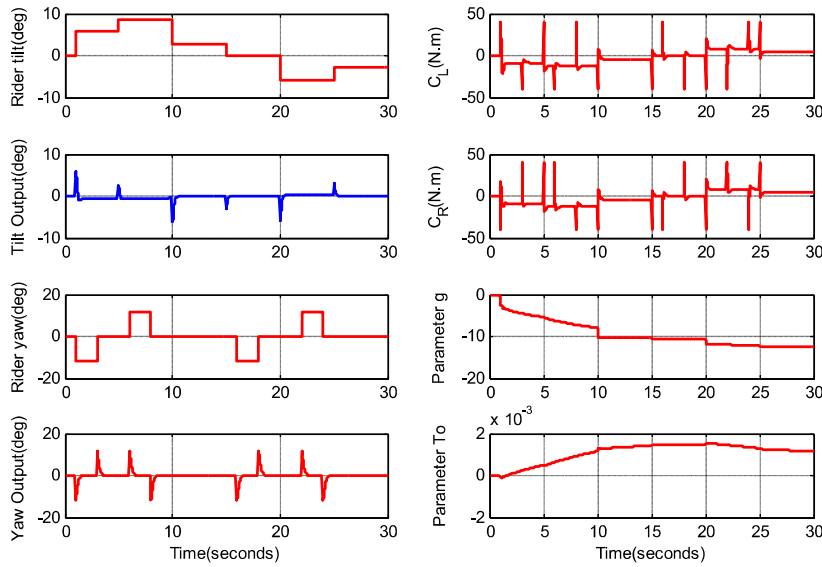


Figure 6: The performance of proposed controller for eScooter

Figure 6 shows the simulation result of eScooter with tilt angle and yaw angle responses, input torque applied to right and left wheel (C_L and C_R), the convergence of unknown $\hat{g}(x_1)$ and $\hat{h}(x_1, x_2)$ parameters. Based on Figure 4, input torque C_L and C_R are determined and saturated to suitable for characteristics of BLDC motor. We see that eScooter is efficiently controlled through rider's tilt angle and rider's yaw angle.

Based on these results, the proposed combination of the adaptive backstepping control based on Lyapunov theory and the PD control effectively shows robustness in the presence of uncertainty parameters and disturbances for tracking problems.

4. EXPERIMENTAL RESULTS

4.1. Hardware Configuration

The investigated eScooter is made of two coaxial wheels which are mounted parallel to each other and are driven by two BLDC motors. Figure 7 shows the experimental photograph of eScooter. Figure 8 illustrates the block diagram of the eScooter control architecture.

Sensors are equipped with accelerometer and gyro for measuring the pitch angle value and potentiometer for measuring the yaw angle. These signals are read by ADC part of master module. Master module is implemented on an embedded dsPIC board. All sensor signals are filtered by Kalman filter and then provided for the proposed controller.

The controller output signals are sent to the slave modules via CAN networks. Slave modules 1, 2 and 3 are implemented on an embedded dsPIC board. Where, slave module 1 and 2 control the left and right wheel of eScooter, respectively. Slave module 3 (HMI) displays the eScooter speed through LCD graphic screen.

CAN networks is applied to communication between master module with slave modules as illustrates in Figure 9. Due to advantages such as: transfer speed up to 1MB, high reliability and good flexibility. CAN networks helps us control the eScooter hierarchy and meets the requirements of real-time operation of eScooter.



Figure 7: Photograph of eScooter

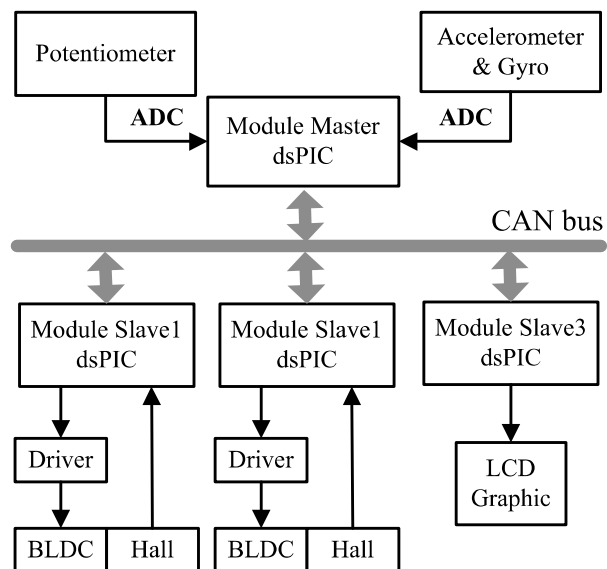


Figure 8: Block diagram of the eScooter architecture

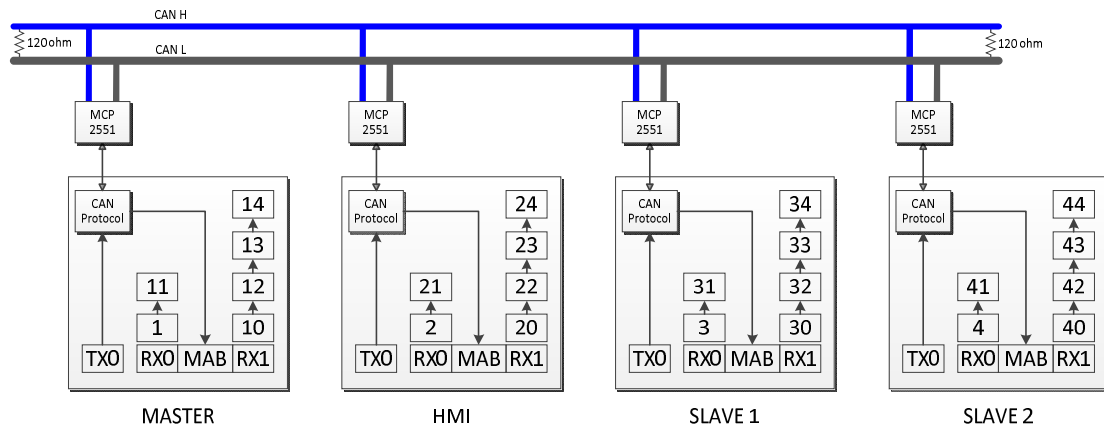


Figure 9: CAN networks architecture in eScooter

4.2. Experimental Results

After embedding the signals processing and control algorithm into eScooter’s hardware, a very good performance is obtained not only in backward and forward movement but also in left turn and right turn operations as well. Figure 10 shows that the tilt angle response oscillates around the equilibrium 0^0 value from 0 to 1.8s when the eScooter has no effect of the outside force. And then, the tilt angle response is swaying only about 0.7s and then returns stable around the equilibrium 0^0 value when the eScooter has effect from outside force. Finally, Fig.11 represents the good tilt angle response when eScooter runs backward and runs forward.

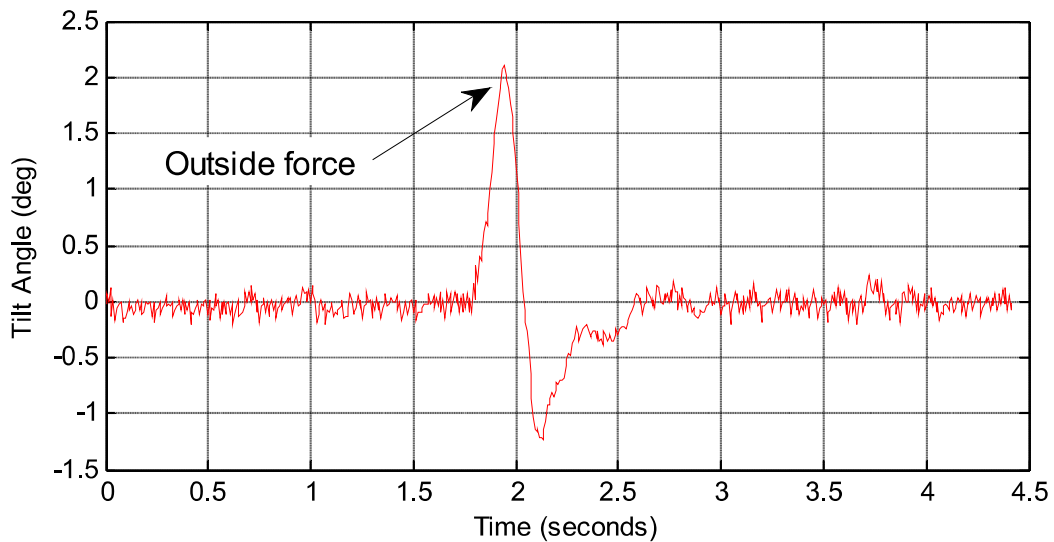


Figure 10: The eScooter tilt angle response during receiving effect from outside force

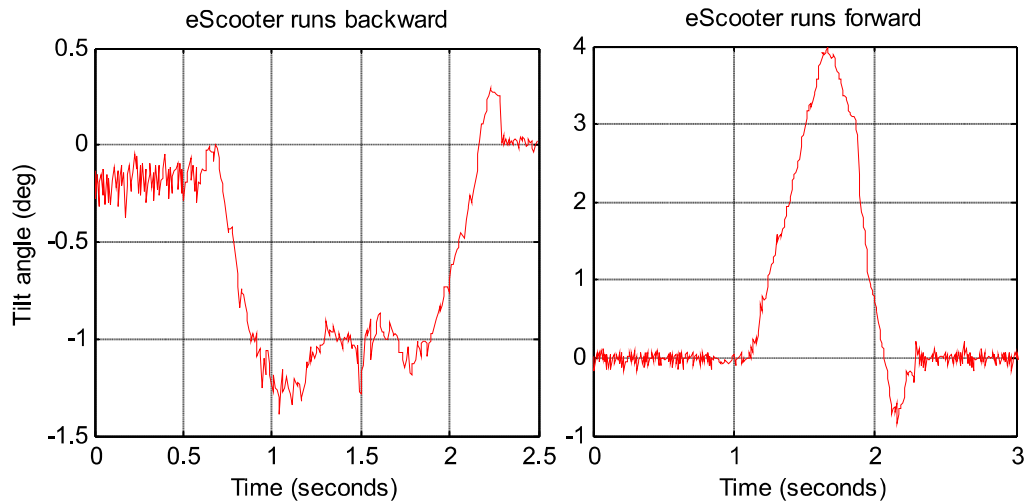


Figure 11: The tilt angle response of eScooter in backward and forward operations

5. CONCLUSIONS

In this paper, the proposed controller combining the adaptive self-balancing backstepping controller and the left and right turning PD controller is applied to the eScooter system. The eScooter terms such as modeling, signal processing, hardware configuration and control scheme are under discussion. Simulation and experimental results prove that the eScooter can be in stable operation and good performance.

6. ACKNOWLEDGMENT

This research was funded by the Vietnam National Foundation for Science and Technology Development (NAFOSTED) and by the Industrial University of HCM city, Vietnam.

REFERENCES

- [1] O. T. Altinöz, *International Journal of Mechanical, Aerospace, Industrial and Mechatronics Engineering*, vol. 1, no. 5, pp. 274 – 277, 2007.
- [2] G. Bishop and G. Welch, “An introduction to the kalman filter,” *Proc of SIGGRAPH, Course*, vol. 8, no. 27599-23175, p. 41, 2001.
- [3] A. Ebrahim and G. V. Murphy, “Adaptive backstepping controller design of an inverted pendulum,” in *System Theory, 2005. SSST’05. Proceedings of the Thirty-Seventh Southeastern Symposium on*. IEEE, 2005, pp. 172–174.
- [4] F. Grasser, A. D’Arrigo, S. Colombi, and A. C. Rufer, “Joe: a mobile, inverted pendulum,” *Industrial Electronics, IEEE Transactions on*, vol. 49, no. 1, pp. 107–114, 2002.
- [5] P. V. Kokotovic, “The joy of feedback: nonlinear and adaptive,” *IEEE Control Systems Magazine*, vol. 12, no. 3, pp. 7–17, 1992.

- [6] M. Krstić, I. Kanellakopoulos, and P. Kokotović, "Adaptive nonlinear control without over-parametrization," *Systems & Control Letters*, vol. 19, no. 3, pp. 177–185, 1992.
- [7] S.-C. Lin and C.-C. Tsai, "Development of a self-balancing human transportation vehicle for the teaching of feedback control," *Education, IEEE Transactions on*, vol. 52, no. 1, pp. 157–168, 2009.
- [8] S.-C. Lin, C.-C. Tsai, and H.-C. Huang, "Adaptive robust self-balancing and steering of a two-wheeled human transportation vehicle," *Journal of Intelligent & Robotic Systems*, vol. 62, no. 1, pp. 103–123, 2011.
- [9] R. Lozano and B. Brogliato, "Adaptive control of robot manipulators with flexible joints," *IEEE Transactions on Automatic Control*, vol. 37, no. 2, pp. 174–181, 1992.
- [10] S. W. Nawawi, M. N. Ahmad, and J. H. S. Osman, "Development of a two-wheeled inverted pendulum mobile robot," in *Research and Development, 2007. SCOReD 2007. 5th Student Conference on*. IEEE, 2007, pp. 1–5.
- [11] S. Nawawi, M. Ahmad, and J. Osman, "Real-time control of a two-wheeled inverted pendulum mobile robot," *World Academy of Science, Engineering and Technology*, vol. 39, pp. 214–220, 2008.
- [12] T. Nomura, Y. Kitsuka, H. Suemitsu, and T. Matsuo, "Adaptive backstepping control for a two-wheeled autonomous robot," in *ICCAS-SICE, 2009*. IEEE, 2009, pp. 4687–4692.
- [13] P. Ranger and A. Desbiens, "Improved backstepping-based adaptive pid control," in *Control and Automation, 2003. ICCA'03. Proceedings. 4th International Conference on*. IEEE, 2003, pp. 123–127.
- [14] X. Ruan and J. Cai, "Fuzzy backstepping controllers for two-wheeled self-balancing robot," in *Informatics in Control, Automation and Robotics, 2009. CAR'09. International Asia Conference on*. IEEE, 2009, pp. 166–169.
- [15] Y. Tan, J. Hu, J. Chang, and H. Tan, "Adaptive integral backstepping motion control and experiment implementation," in *Industry Applications Conference, 2000. Conference Record of the 2000 IEEE*, vol. 2. IEEE, 2000, pp. 1081–1088.

Received on April 02 - 2014

Revised on December 15 - 2014