

Building embedded quasi-time-optimal controller for two-wheeled self-balancing robot

Hai N. Phan^{1,*} and Chiem X. Nguyen²

¹Department of Software Engineering, Le Quy Don Technical University, Hanoi, Vietnam

²Automation & Engineering Department, Le Quy Don Technical University, Hanoi, Vietnam

Abstract. In this paper, two-wheeled self-balancing robot problem is introduced, and a quasi-time-optimal approach is applied to synthesize the control law. The result is compared with that of other methods. The process for designing and building the testing model, implementing the synthesized control law is also described.

1 Introduction

Two-wheeled self-balancing robot is a robot with two coaxial wheels. The idea of building such a robot, proposed firstly by Seway Inc., is based on two pillars, one being the way human-being keeping balance on two feet and the other being the movement of two-wheeled vehicle. Nowadays the robot has been developed to commercial product. Fig. 1 shows a typical two-wheeled self-balancing robot.



Fig. 1. Two-wheeled self-balancing robot.

Two-wheeled self-balancing robot has been extensively used as a means of communication thanks to its advantages, such as: small size, environmental protection, simplicity in controlling, flexibility in various topographies, simplicity in transport, installing and.

Qualitatively, the principle for keeping balance in two-wheeled robots is as follows. A robot in the equilibrium position is tilted with an angle φ when subjected to a force. For the robot to regain its equilibrium, a force F is needed to apply to it so that its center of gravity is located in the middle of two wheels as shown in Fig. 2.

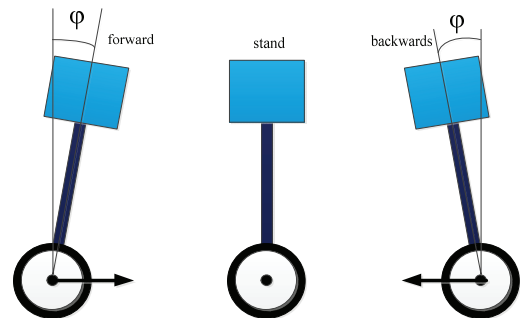


Fig. 2. The qualitatively balancing principle.

2 Mathematical model of the robot

The mathematical model of two-wheeled self-balancing robot is described in equations set (1) [1]:

$$\begin{bmatrix} \ddot{x} \\ \ddot{\varphi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{k_m k_e (M_p l r - I_p - M_p l^2)}{R r^2 \alpha} & \frac{M_p^2 g l^2}{\alpha} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{2 k_m k_e (r \beta - M_p l)}{R r^2 \alpha} & \frac{M_p^2 g l^2 \beta}{\alpha} & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2 k_m (I_p + M_p l^2 - M_p l r)}{R r \alpha} \\ 0 \\ \frac{2 k_m (M_p l - r \beta)}{R r \alpha} \end{bmatrix} U_a \quad (1)$$

Where: x - displacement (m); \dot{x} - velocity (m/s); \ddot{x} - acceleration (m/s^2); φ - angular displacement (rad); $\dot{\varphi}$ - angular velocity (rad/s); $\ddot{\varphi}$ - angular acceleration (rad/s^2); U_a - voltage (V); k_m - rotation moment constant (Nm/A); k_e - reactive power constant (Vs/rad); R - nominal resistance (Ω); l - distance between the weight center of the wheels and the weight center of the robot (m); g - gravity acceleration (m/s^2); M_p - the chassis volume (kg); r - wheel radius (m); I_p - moment of inertia of the chassis (Nm); I_w - moment of inertia of the wheels (Nm); M_w - the wheel weight (kg). In addition:

* Corresponding author: phanhai@mail.ru

$$\beta = (2M_\omega + \frac{2I_\omega}{r^2} + M_p)$$

$$\alpha = (I_p\beta + 2M_p l^2 (M_\omega + \frac{I_\omega}{r^2}))$$

The problem of synthesizing control law for two-wheeled self-balancing robot is to find the force F via the voltage U_a that guarantees the robot to move from the start position $(x(0), \varphi(0))$ to the end position (x_{sp}, θ) and stand still there.

Controlling a self-balancing robot is equivalent to the problem of controlling an inverted pendulum which is a typical system for simulation and implementation of controlling methods [2-7]. The optimal controls for the system were proposed in [2] and [3], in which the law using the signum function was used. This also means that the time optimal problem was not addressed. In this paper, we proposed a method for synthesizing quasi-time optimal nonlinear control law to overcome the above-mentioned problem, and implement it in an embedded board.

3 Using the quasi-time optimization approach

The quasi-time optimization approach for synthesizing control law is developed by Neydorf R. A's group [21, 22]. By using this approach, the problem of synthesizing control law for a wide spectrum of nonlinear objects can be solved. It is reported that the synthesized control law has several advantages, such as: quasi-optimal minimum-time, asymptotic stability and robustness.

To simplify model (1), we introduce the following variables:

$$a = \frac{2km.ke(M_p lr - I_p - M_p l^2)}{Rr^2\alpha};$$

$$b = \frac{M_p^2 gl^2}{\alpha}; c = \frac{2km(I_p + M_p l^2 - M_p lr)}{Rr\alpha};$$

$$a_1 = \frac{2km.ke(r\beta - M_p l)}{Rr^2\alpha}; b_1 = \frac{M_p gl\beta}{\alpha};$$

$$c_1 = \frac{2km(M_p l - r\beta)}{Rr\alpha}; u = U_a;$$

It can be seen that $c_1 a = a_1 c$, or $a - a_1 c / c_1 = 0$.

Substituting these values into equation set (1), we get:

$$\begin{cases} \dot{x} = \dot{x} \\ \ddot{x} = a\dot{x} + b\varphi + cu \\ \dot{\varphi} = \dot{\varphi} \\ \ddot{\varphi} = a_1\dot{x} + b_1\varphi + c_1u \end{cases} \quad (2)$$

Changing state variables as follows:

$$y_1 = x - \frac{c}{c_1}\varphi; y_2 = \dot{x} - \frac{c}{c_1}\dot{\varphi}; \quad (3)$$

$$y_3 = \varphi; y_4 = \dot{\varphi}$$

It is easy to verify that Jacobian matrix of transformation (3) is not degraded:

$$\det \left[\frac{\partial y}{\partial X} \right] = \begin{vmatrix} 1 & 0 & -c/c_1 & 0 \\ 0 & 1 & 0 & -c/c_1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{vmatrix} = \frac{c}{c_1} \neq 0$$

Where: $X = (x \quad \dot{x} \quad \varphi \quad \dot{\varphi})^T$.

Equations by $Y = (y_1, y_2, y_3, y_4)^T$:

$$\begin{cases} \dot{y}_1 = y_2 \\ \dot{y}_2 = (b - \frac{b_1 c}{c_1})y_3 + (a - \frac{a_1 c}{c_1})y_4 \\ \dot{y}_3 = -\frac{c_1}{c}y_2 + \frac{c_1}{c}y_4 \\ \dot{y}_4 = by_3 + ay_4 + cu \end{cases} \quad (4)$$

Equations (4) is in a controllable Jordan form [22]. Virtual system (4), which has asymptotic stability and quasi-time optimum, is used to synthesize the control law u with $z_1 = y_1$; u is found to introduce the system (4) into the virtual system (5) [22]. The obtained control law is a differentiable function of state variables, of which the formula is very long and therefore not given here.

$$\begin{cases} \dot{z}_1 = -v \frac{z_1 - x_{sp}}{\sqrt{(z_1 - x_{sp})^2 + \varepsilon_1^2}} + z_2 \\ \dot{z}_2 = -\frac{z_2}{\varepsilon_2} + z_3 \\ \dot{z}_3 = -\frac{z_3}{\varepsilon_3} + z_4 \\ \dot{z}_4 = -\frac{z_4}{\varepsilon_4} \end{cases} \quad (5)$$

Where $h_0(z, x_{sp})$ is quasi-time optimal function ($h_0(z, x_{sp}) \approx \text{sign}(z - x_{sp})$) [21].

Fig. 3 shows the simulation result of robot movement using the synthesized control law. The values set for simulation are: $k_m = 0.022$, $R = 1.7$, $r = 0.035$, $M_p = 1$, $M_w = 0.03$, $I_p = 0.0012$, $I_w = 0.000016$, $l = 1$, $g = 9.81$, $v = 1$, $\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = \varepsilon_4 = 0.5$ $m_l = 2$, $m_p = 0.2$, $y(0) = x(0) = 0$, $x_{sp} = 1$, $\dot{x}(0) = 0$, $\varphi(0) = 0$, $\dot{\varphi}(0) = 0$.

Fig. 3 demonstrates the synthesized quasi-time optimal control. As compared to fuzzy control shown in Fig. 4, the synthesized quasi-time optimal control has several advantages. First, signal u is smooth. Second,

there is no oscillation of control variable x . And finally, the speed of control is faster.

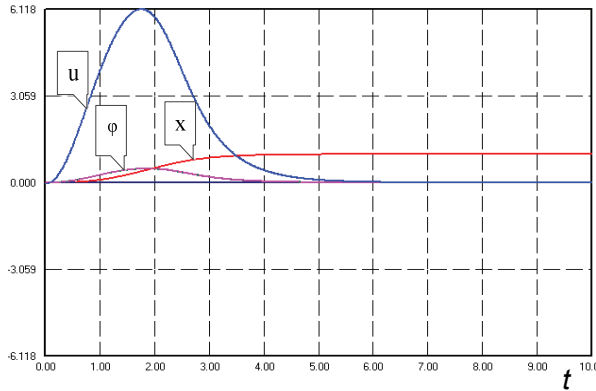


Fig. 3. Transition process with quasi-time-optimal control law.

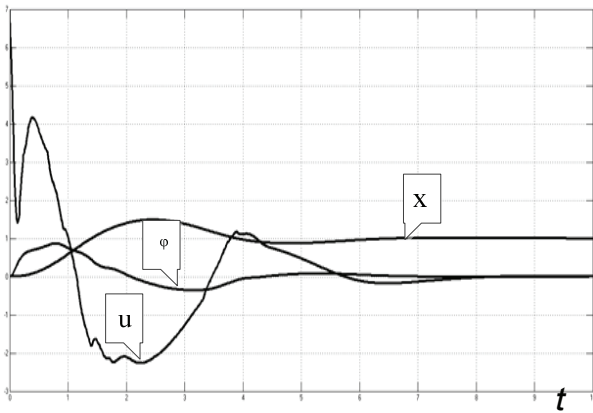


Fig. 4. Transition process with fuzzy control law

4 Designing and building embedded control system

To prove the feasibility of the simulation result, the testing model was build using Arduino microcontroller with a 16MHz crystal frequency to implement the control law as shown in Fig. 5.

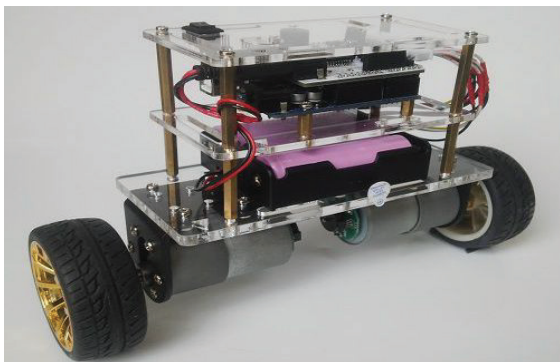


Fig. 5. Two-wheeled self-balancing robot model

The most challenging issue in controlling a self-balancing robot is to keep the angle stable to guarantee the balance of the robot. The testing model is designed to keep the angle stable.

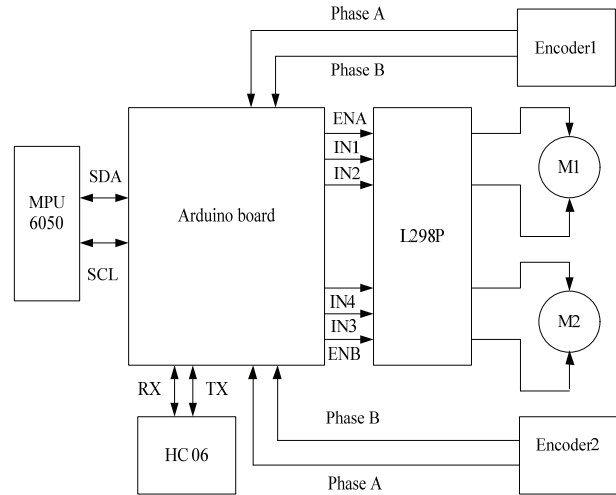


Fig.6. The schematic of two-wheeled self-balancing robot hardware

Where: M1 and M2 are the motor controlling the two wheels, L298P is the power block, and MPU 6050 is the sensing block.

The most difficult problem in implementing quasi - time optimal control law in embedded system is that the control law is quite complex and cumbersome because it contains 4th order derivative of the quasi - time optimal function. This results to long time for computation, making it difficult for a system to meet the real-time requirement. To simplify the calculation process, the intermediate functions, which are the high order derivative of sub-optimal function, were used in the embedded program. Equations (6) and (7) are the quasi - time optimal function (5) and its 1st order derivative. The high order derivatives are not shown here due to the fact that they are very cumbersome.

$$\text{float } h0(\text{float } z, \text{float } zs) \{$$

$$\text{return } ((z-zs)/\text{sqrt}((z-zs)*(z-zs)+f1*f1)); \quad (6)$$

$$\}$$

$$\text{float } h1(\text{float } z, \text{float } zs) \{$$

$$\text{return } (f1*f1/\text{sqrt}(((z-zs)*(z-zs)+$$

$$f1*f1)*((z-zs)*(z-zs)+$$

$$f1*f1)*((z-zs)*(z-zs)+f1*f1))); \quad (7)$$

$$\}$$

The angular displacement for balancing the robot is shown in Fig. 7. In general, the system has stabilized well but is still oscillating due to the following reasons. First, the actual model was built manually with relatively high measurement error. The actual model slips as the system is stable at zero angles. Second, the epsilons in the law are approximate parameters, which are not optimized since it is very hard to estimate them accurately in the simulation. Third, the measurement of feed-back values is with relatively high error. Fourth,

time parameter is discretized in the embedded system. Improving the quality of the control part is the goal of our future work.

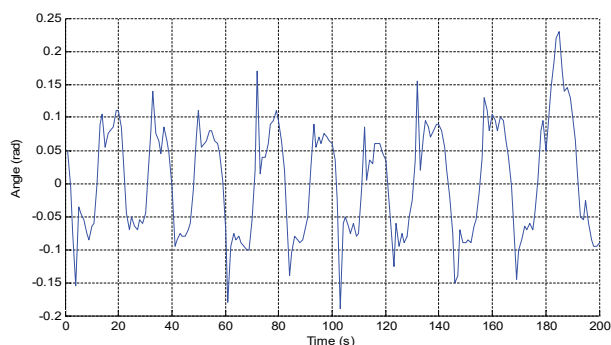


Fig. 7. The change of balance angle with time

5 Conclusions

In this paper, the problem designing and building two-wheeled self-balancing robot has been addressed comprehensively. First, the problem of synthesizing control law was addressed theoretically. By using the quasi – time optimal approach, the problem of synthesizing control law was solved. Then the robot and embedded board were designed to implement the synthesized control law.

References

1. Y.C. Ong, M.B. Shukri, Z. Abidin, *Proceedings of the IEEE 7th. Research and Development (SCoReD 2006)* (2006)
2. L.B. Prasad, B. Tyagi, H.O. Gupta, *International Journal of Automation and Computing*, **11**(6), 661-670 (2014)
3. S.C. Lin, C.C. Tsai, H.C. Huang, *Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics*, 1965-1970 (2009)
4. S.C. Lin, C.C. Tsai, W.L. Luo, *In proc. 33rd Annu. IEEE IECON*, 868-873 (2007)
5. H. Ohara, T. Murakami, *IEEE Trans. Ind. Electron.*, **55**, 1277-1285 (2008)
6. K.E. Vinodh, J. Jovitha, *Procedia Engineering*, **64**, 169-178 (2013).
7. R. Mellah, F. Lahouazi, S. Djennoune, S. Guermah, R. Toumi, *International Journal of Control, Automation and Systems*, **1** (2012)
8. O. Boubaker, *International Journal of Advanced Robotic Systems*, 1-9 (2013)
9. R.P.M. Chan, K.A. Stol, and C.R. Halkyard, *Annual Reviews in Control*, 89-103 (2013)
10. A. Filipescu and V. Minzu, *Proc. Int. Conf. System Theory, Control, and Computing*, 1-6 (2011)
11. E.J. Hwang, H.S. Kang, C.H. Hyun, and M. Park, *International Journal of Advanced Robotic Systems*, **10**, 1-8 (2013)
12. B.S. Park, S.J. Yoo, J.B. Park, and Y.H. Choi, *IEEE Transactions on Control Systems Technology*, 207-214 (2009)
13. R. Neydorf, *SAE Technical Paper*, 2015-01-2481, 8, doi: 10.4271.2015-01-2481 (2015)
14. R. Neydorf, Y. Sigida, V. Voloshin and Y. Chen, *SAE Technical Paper* 2013-01-2111, 10, doi:10.4271/2013-01-2111 (2013)
15. A. Filipescu and V. Minzu, *Proc. Int. Conf. System Theory, Control, and Computing*, 1-6 (2011)
16. J. Wu, Y. Liang, and Z. Wang, *6th international forum on strategic technology (IFOST)*, 1031–1035 (2011)
17. I. Baturone, F.J. Moreno-Velo, V. Blanco, and J. Ferruz, *IEEE Trans. Ind. Electron.*, **55**, no. 2, 928-936, (2008)
18. F. Grasser, A. D. Arrigo, and S. Colombi, *IEEE Trans. Ind. Electron.*, **49**, no. 1, 107-114 (2002)
19. Xiaojun, X.Z. Gao, and Xianlin, *In proc. 2004 IEEE International Conference on Systems, Man and Cybernetics*, **3**, 2286 – 2291 (2004)
20. P.V. Kokotovic, *Control Systems Magazine, IEEE*, **12** (3), 7–17 (2008)
21. R.A. Neudorf, Vestnik DSTU. Management and diagnostics in dynamic systems, 13-21 (1999) [in Russian]
22. N.H. Fan, Metody raspoznavanija upravljajemyh zhordanovyh form dinam., Cand. tech. sciences. Diss., Rostov-on-Don, 193 (2008) [in Russian]