Balancing control for single-wheel unicycle Robot using the Sliding mode controller

P L Nguyen¹,²*, T P Nguyen¹ and T M Ngo³

¹Faculty of Engineering, Van Lang University, Ho Chi Minh City, Vietnam
²Institute of Engineering, HUTECH University, Ho Chi Minh City, Vietnam
³Refrigeration Division, Tan Son Nhat International Airport, Ho Chi Minh City, Vietnam

*luan.np@vlu.edu.vn

Abstract. In recent years, sliding mode control has been disseminated in the industry, including nonlinear control, and is considered as one of the simplest methods for nonlinear system control, including uncertainly model-based systems that are affected by external magnetic interference or parameter conversion. This paper shows the method of sliding mode control for a single-wheel unicycle-like mobile Robot. First, the Robot kinematics model was established by using a self-balancing system based on the balancing control principle in two models; flywheel inverted pendulum. In order to simplify the complicated problem with high nonlinearity, this model was balanced in each direction. Then the system model was emulated, showing the result that the separation sliding mode control worked properly with a large angle of deflection (0.17 rad), and the wheel rolled slowly with the relatively low speed (~2.1 rad/s), while the flywheel rotated slowly with high-speed of rod control (~28 rad/s).

1. Introduction

Along with the development of science and technology, today robots have the ability to replace humans working in hazardous environments, in manufacturing, or serving people for entertainment needs or problems in daily life day… Thanks to the continuous development of technology, robots have been built to serve many different purposes. In particular, the development of automatic balancing systems is also very strong. Over the years it has become a familiar topic in the teaching environment of engineering and automation. More and more products apply the theory of balance for fast, compact movement in short distances.

Balancing of a Two-Wheel Unicycle Robot system is an application developed from the self-balancing reverse pendulum system. For a self-balancing single-wheel unicycle Robot system, it is more complicated, it is a problem of combining the balance of the inverted pendulum and the balance of the handwheel, balancing the unicycle Robot in two directions. This is a complex system with high nonlinearity and instability easily. Issues related to this system including controller design, signal-processing filter design to filter measurement signals… are complex problems in the control field. Currently, there are many products applying the equilibrium principle using various algorithms such as PID, Fuzzy fuzzy logic, LQR… [1-4]. In this paper, a self-balancing single-wheel unicycle Robot system using a sliding controller is presented.

The objective of this paper is to build a simulation model for a self-balancing single-wheel Unicycle-like Robot system using the sliding control technique. The mathematical model of the system is divided...
into two models, one in the form of an inverted pendulum (Pitch angle), the other in the form of a flywheel pendulum (Roll angle) from which to design the sliding and forward controller model simulation on Matlab simulink software to check the accuracy as well as see the system's characteristics.

2. Unicycle Robot modeling

The Unicycle Robot system includes a ground contact wheel to move, the robot body, turntable 1 and reaction wheel 2 to change the direction of the robot and keep the robot in balance.

OXYZ is the ground coordinate system; O_{bx}y_{by}z_{bz} is the coordinate system attached to the robot; \( \alpha, \beta, \gamma \) are Pitch, Roll, Yaw angles respectively; \( \theta, \phi, \psi \) are the rotation angle of wheel, balance flywheel and directional swing wheel, respectively.

The self-balancing single-wheel system is a balancing vehicle system based on the principle of balance control with two models: an inverted pendulum and a flywheel pendulum. We can divide the model into three parts: the body, the upper part is a rotating flywheel that can rotate around the shaft to create a moment of inertia to keep the vehicle balanced in a left-right direction according to the flywheel pendulum balance principle, the lower part is the wheel that can move back and forth to keep the vehicle balanced in the front-back direction according to the reverse pendulum balance principle. This is a relatively complex model with high nonlinearity, so to simplify the problem, we divide the system into two small models, that is, we balance the model in each direction according to the above equilibrium principles [5]. And combining the two models after splitting requires that in each model (inverted pendulum and flywheel pendulum), the system must have stable equilibrium in the small region adjacent to the equilibrium point to ensure that the changes of each model have little effect on each other.

The model parameters are shown in Table 1.

| Notation | Unit          | Interpretation                                                                 |
|----------|---------------|-------------------------------------------------------------------------------|
| \( \alpha \) | rad           | Pitch, Roll, Yaw angle deviation respectively.                                |
| \( \beta \) | rad           | Rotation angle position of wheel, balance flywheel, turntable.               |
| \( \gamma \) | rad           | Rotational angle of wheel, balance flywheel, directional swing wheel.        |
| \( m_w \) | kg            | Respectively the mass of the wheel, robot frame, balance flywheel, and directional flywheel. |
| \( m_b \) | kg            |                                                                                   |
| \( m_d \) | kg            |                                                                                   |
| \( m_f \) | kg            |                                                                                   |
| \( \tau_w \) | N.m          | In turn is the controlling force acting on the wheel, balance flywheel, and directional flywheel. |
| \( \tau_d \) | N.m          |                                                                                   |
| \( \tau_f \) | N.m          |                                                                                   |
| \( I_{wx} \) | Kg.m²        | The inertial moment of the wheel on the three axis x, y, z axis respectively. |
| \( I_{wy} \) | Kg.m²        |                                                                                   |
| \( I_{wz} \) | Kg.m²        |                                                                                   |
| \( I_{bx} \) | Kg.m²        | The inertial moment of the robot frame follows the x, y, z axis respectively.    |
| \( I_{by} \) | Kg.m²        |                                                                                   |
| \( I_{bz} \) | Kg.m²        |                                                                                   |
| \( I_{dx} \) | Kg.m²        | The balance flywheel moment of inertia according to the x, y, and z axis respectively. |
| \( I_{dy} \) | Kg.m²        |                                                                                   |
| \( I_{dz} \) | Kg.m²        |                                                                                   |
| \( I_{fx} \) | Kg.m²        | The handwheel moment of the directional flywheel in the direction of the x, y, and z axis respectively. |
| \( I_{fy} \) | Kg.m²        |                                                                                   |
| \( I_{fz} \) | Kg.m²        |                                                                                   |
| \( l \) | m            | Distance from wheel to rotating flywheel                                        |
| \( l_r \) | m            | Wheel radius.                                                                  |
The dynamic equation of the unicycle built from the Euler-Lagrange equation is as follows:

\[ \frac{d}{dt} \left[ \frac{\partial L}{\partial q} \right] - \frac{\partial L}{\partial \dot{q}} = Q \]  

(1)

In which:

- \( T \) is the kinetic energy of the whole system,
- \( V \) is the potential energy of the system,

\[ q = [\alpha \beta \gamma \phi \theta \psi]^T, \quad Q = [0 \quad 0 \quad \tau_d \quad \tau_w \quad \tau_f] \]

We have the kinetic equations of the self-balancing unicycle on the Pitch, Roll and Yaw axis.

a. Axis Pitch.

\[ l_g \dot{\alpha} + \left( I_{by} + I_{by} + \frac{l_w^2}{4} (m_y + 4m_r) + l_w^2 m_d + \frac{l_w^2 + lm_r}{2} l_y \right) \dot{\alpha} \]

\[ + \left( I_{wd} m_d + \frac{l_w^2 + lm_r}{2} l_y \right) \alpha \cos \alpha - \left( l_y m_d + \frac{m_y}{2} + m_j \right) \alpha^2 \sin \alpha + \left( I_{wy} + l^2 (m_y + m_j + m_w) \right) \ddot{\alpha} = \tau_w \]

(2)

b. Axis Roll.

\[ I_{dx} \dot{\phi} + \left( I_{dx} + I_{dx} + I_{dx} + l_x^2 m_d + l^2 m_j + \frac{m_y}{2} (l + 2l_y) + 2l_y (l_y m_d + lm_r) + l_y^2 (m_d + m_j + m_w) \right) \ddot{\phi} \]

\[- \left( I_y (m_y + m_j + m_w) + \frac{m_y}{2} (l + 2l_y) + l_y^2 m_d + lm_r \right) g \sin \beta = 0 \]

(3)

\[ I_{dx} \ddot{\phi} + I_{dx} \ddot{\beta} = \tau_d \]

(4)

c. Axis Yaw.

\[ I_{fx} \dot{\psi} + \left( I_{wc} + I_{dx} + I_{dx} + I_{fc} \right) \ddot{\psi} = 0 \]

\[ I_{fx} \ddot{\psi} + I_{fx} \ddot{\psi} = \tau_f \]

(5)

(6)

(7)

3. Unicycle Robot controller design using the Sliding mode controller

A diagram of the sliding control system block is shown in figure 2.
\[ s_1(x) = \dot{e}_1 + \lambda_4 e_1 = x_2 - x_{2d} + \lambda_4 (x_1 - x_{1d} - z) = 0 \]
\[ s_2(x) = \dot{e}_2 + \lambda_4 e_2 = x_4 - x_{4d} + \lambda_2 (x_3 - x_{3d}) = 0 \] (8)

Compared with the classic slide controller, we find that the control target has been changed.
\[ x_1 = x_{1d} + z \]
\[ x_2 = x_{2d} \]
\[ x_3 = x_{3d} \]
\[ x_4 = x_{4d} \] (9)

Similar to the classic sliding controller we have the following control signals.
\[ u = u_1 = \hat{u}_1 - K_1 \text{sat} \left( s_1(x) b_1(x) / \phi_1 \right); K_1, \phi_1 > 0 \] (10)

With
\[ \hat{u}_1 = -b_1^{-1} (x) \left( f_1(x) + \lambda_4 \dot{e}_1 \right) \] (11)

Output \( z \) is the value
\[ z = K_2 \text{sat} \left( s_2(x) / \phi_2 \right); K_2, \phi_2 > 0 \] (12)

The controller works in the following sequence:

When \( s_2 \neq 0 \) therefore \( z \neq 0 \) The requirement for the controller is generate signals \( u \) reduce deviation \( s_2 \) with value 0. when the \( s_2 \to 0 \), follow \( z \to 0 \), and \( s_1 \to 0 \) thus the system control target can be achieved.

4. Results and discussion

From the mathematical model describing the system found above, the author of the paper has designed the model simulating it on Matlab Simulink with the actual parameters and system models as shown in Figures 3 and 4.

![Figure 3. Schematic simulation of Unicycle Robot system with sliding mode controller](image-url)
4.1. Simulate math model without control effect

Response results:

Figure 4. The actual configuration of the system

Figure 5. Robot corner speed and position response without control signal

Figure 6. Response and PITCH angular velocity without control signal
From the above graphs, we see: when there is no signal to control and ignore the friction at the rotation axis as well as the air friction, if the system deviates from the initial angle 0 it will vibrate like a pendulum conditioning around the 180º equilibrium point.

4.2. System simulation with decoupled sliding mode control

Output response when using decoupled sliding mode control

With the following initial conditions:

\[
\begin{bmatrix}
\alpha & \beta & \gamma & \dot{\alpha} & \dot{\beta} \\
\theta & \phi & \psi & \dot{\theta} & \dot{\phi}
\end{bmatrix} = [0.17 \ 0.17 \ 0.17 \ 0 \ 0]
\]

\[
\begin{bmatrix}
\theta & \phi & \psi & \dot{\theta} & \dot{\phi}
\end{bmatrix} = [0 \ 0 \ 0 \ 0 \ 0]
\]
Figure 10. Response to wheel position and speed with decoupled sliding mode control.

Figure 11. Response of deflection angle and ROLL angle velocity with decoupled sliding mode control.

Figure 12. Flywheel position and speed response with decoupled sliding mode control

Figure 13. Response to YAW rotation angle with a given value
The simulation results show that the decoupled sliding mode control works well with a large initial deviation angle (0.17rad):

- The pitch and roll angle responds to 0 quite quickly (~ 1s)
- The wheel spins quite a bit and the wheel speed is relatively low (~ 2.1 rad / s).
- The flywheel is also quite low and control speed should be high (~ 28 rad / s).

The paper explored the single-wheel robot, modeled the system, and designed a decoupled sliding mode controller for the balance holding system. The simulation results on Matlab Simulink show the efficiency of the decoupled sliding mode control

References
[1] Jae-Won An, Min-Gyu Kim and Jangmyung Lee, “Control of a Unicycle Robot using a Non-model based Controller”, Journal of Institute of Control, Robotics and Systems (2014).
[2] Seong I. Han and Jang M. Lee “Balancing and Velocity Control of a Unicycle Robot Based on the Dynamic Model”, IEEE Transactions on Industrial Electronics January 2015.
[3] Xiao-gang Ruan and Yu-feng Wang, “The Modelling and Control of Flywheel Inverted Pendulum System”, Artificial Intelligence and Robot Institute Beijing University of Technology Beijing, 100124, China.
[4] Zhao jie and Ren Sijing, “Sliding Mode Control of Inverted Pendulum Based on State Observer”, 2016.
[5] M.J. Mahmoodabadi, S. Momennejad, A. Bagher, “Online optimal decoupled sliding mode control based on moving least squares and particle swarm optimization”, 2014.