Design and Implementation of Two-Wheeled Self-Balancing Vehicle Using Accelerometer and Fuzzy Logic

Sunu S. Babu and Anju S. Pillai

Abstract Two-wheeled self-balancing vehicle commercially known as "Segway" is a promising upcoming mode of transportation in many fields viz. corporate worlds, tourist place, medical field, or for personal use. In this paper, a control strategy and sensor-based control of two-wheeled self-balancing vehicle is proposed. The concept of the stabilizing the vehicle is inspired from the inverse pendulum theory. Based on steady-state space mathematical model, the entire system control is divided into two subsystems: self-balance control system (forward or backward motion balancing) and yaw control system (left or right movement). The control strategy used is fuzzy logic and is applied to both subsystems. A prototype model of the self-balancing vehicle is developed and the proposed mathematical model and control logic are verified by testing on the developed prototype.

Keywords Fuzzy logic controller · Self-balancing · Fuzzy rule base accelerometer · Arduino UNO R3

1 Introduction

Two-wheeled self-balancing vehicle, is a kind of low-end version of Segway, is basically used to ride person or any object from one place to another. Recently it is looked as one of the future modes of transportation as it completely runs on electric power (battery source). The vehicle is the output of mechatronics product embedded with control algorithm. With the help of fast evolving technology, the

S.S. Babu (✉) · A.S. Pillai
Department of Electrical and Electronics Engineering, Amrita Vishwa Vidyapeetham, Amrita School of Engineering, Coimbatore 641112, India
e-mail: sunurocks@gmail.com

A.S. Pillai
e-mail: s_anju@cb.amrita.edu

S.C. Satapathy et al. (eds.), *Proceedings of the Second International Conference on Computer and Communication Technologies*, Advances in Intelligent Systems and Computing 381, DOI 10.1007/978-81-322-2526-3_6

vehicle with essential safety features can be fabricated in low cost so that it can become one of the most successful pollution-free two-wheeled vehicles in future along with satisfying the human requirement. The application of such vehicle can be used in corporate world, tourist place, medical field, personal vehicle, and many more.

Basically, the vehicle consists of a chassis which includes two wheels with a handlebar protruding up from it. The vehicle is driven forward and backward by leaning over the chassis in the, respectively, direction whereas for the steering purpose the handlebar is used. The most challenging feature which gives the vehicle limelight is its self-balancing action. Recently many researches are been carried out for the control of the vehicle. In one of the methods [1], two-wheel electric vehicle balancing was done by PD controller and Kalman Filter along with accelerometer and gyroscope as sensing part. In [2], a kinetic equation for control of two-wheeled balancing robot by Newton Dynamics method is attempted. Two controllers were used for controlling action: pole placement state-feedback controller and fuzzy logic controller. In [3], adaptive sliding-mode control method is employed for self balancing and yaw controlling of human transport vehicle. Here mathematical equation is derived and result is verified by the testing vehicle in different terrains. In [4], Arduino microcontroller board was used for developing self-balancing robot, and PI-PD control design was implemented for control action.

In some approach, two control algorithms are used viz. Proportional Integral Derivative (PID) and Linear Quadratic Regulator (LQR) [5]. A bacterial-based swarm controller is presented for self-balancing two-wheeled vehicle in [6]. The bacterial-based swarm algorithm is simplified in order to adjust the parameters of the fuzzy model.

In the current work, a prototype of two-wheeled electric vehicle is designed and developed. Understanding the system dynamics is carried out by mathematical modeling. For determining the vehicle orientation, accelerometer sensor is used. The vehicle control system is fully controlled by Fuzzy logic. The developed prototype is cost optimized with respect to performance, as the vehicle uses only accelerometer sensor instead of combination of gyroscope and accelerometer sensor and the vehicle performance is properly tuned using Fuzzy logic.

The rest of the paper is organized as follows: Sect. 2 describes the system overview, and in Sect. 3 mathematical modeling of the vehicle is presented. Control strategy for balancing the vehicle is explained in Sect. 4 followed by results in Sect. 5. Finally, the paper is concluded in Sect. 6.

2 System Overview

In this section, the hardware details of the two-wheeled vehicle are presented. The idea for the design of two-wheeled self-balancing vehicle was taken from [1]. The hardware model consists of a broad platform which holds the motor driver and is connected to two wheels. The sensors used are accelerometer and potentiometer for

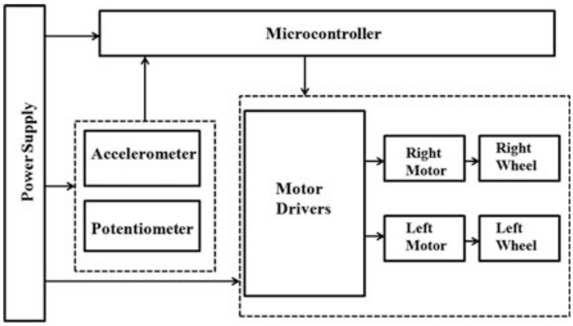


Fig. 1 System block diagram

determining the vehicle orientation. There is an adjustable bar protruding from the platform; used for steering purpose and is connected to potentiometer. Battery provides power supply to microcontroller, sensors, motor driver, and two motors. The platform is designed in such a way that after every component is attached, there is enough space for the rider to stand on it. The block diagram of the vehicle is shown in Fig. 1.

2.1 System Design

The snapshot of the developed prototype of two-wheeled self-balancing vehicle is shown in Sect. 5.2 (Fig. 5).

(i) **Mechanical Components:**

- Platform: The platform is made from plastic material. Its dimension is 22 × 18 cm.
- Motors: Two motors are used for two wheels; left and right. The motor specification is 12 V and 200 rpm.
- Wheels: Two wheels of diameter 5 cm each are used.

(ii) **Electrical and Sensor components:**

- Accelerometer: ADXL335 accelerometer is used. It is an inertial sensor. It can measure acceleration in one, two, or three orthogonal axes.
- Potentiometer: One 5 K pot is used, which is operated at 5 V
- Microcontroller: Arduino UNO R3 controller is chosen. Arduino is an open source electronics platform based on easy to use hardware and software. It is based upon ATmega 328 microcontroller.
- Motor Driver: L293D IC is used for driving the motor. Operating voltage range is from 5 to 36 V and maximum current output is 600 mA per motor.

- Power Supply: 12 V/2 A power supply was used for the entire system. And the distributive circuit was used to provide individual power supply to each component. The range of voltage rating needed was 5–12 V.

3 Modeling

The modeling will help understand the vehicle dynamic nature so as to control it more accurately. The mathematical modeling of the vehicle is studied and surveyed from [1, 2, 7, 8]. The concepts from the references are combined in this paper, to obtain the smooth control of the vehicle. The purpose of modeling is for proper controlling of the vehicle. Before the formulation of mathematical equations, some assumptions are taken under consideration.

- There is only point contact between wheel and ground.
- There is no relative motion between platform and the feet of person, that is, platform and feet are always in contact.
- There is no slipping between two wheels and ground; only rotational motion is possible.

Free body diagram of the both wheel and platform is considered for derivation of mathematical equation which is referred from [7]. The main objective of the vehicle is to balance itself that is there should not be any motion for vehicle when no force is applied to vehicle. Now free body diagram equations are in non-linear form which has to be made linear for getting the appropriate result. This is done by linearizing the equations around the operating point ($X_{\mathrm{RM}} = 0$, $\theta_{\mathrm{p}} = 0$ and $\delta = 0$). Also considering the vehicle parameter and neglecting the other friction force, the resultant state space equations obtained are as follows:

$$\begin{bmatrix} \dot{X}_{RM} \\ \dot{V}_{RM} \\ \dot{\theta}_p \\ \dot{\omega}_p \\ \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & A_{23} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & A_{43} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_{RM} \\ V_{RM} \\ \theta_p \\ \omega_p \\ \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ B_{21} & B_{22} \\ 0 & 0 \\ B_{41} & B_{42} \\ 0 & 0 \\ B_{61} & B_{62} \end{bmatrix} \begin{bmatrix} C_L \\ C_{\mathrm{R}} \end{bmatrix} \tag{1}$$

$$\mathrm{Y} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_{RM} \\ V_{RM} \\ \theta_P \\ \omega_P \\ \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} U \tag{2}$$

The terms A_{23}, A_{43}, B_{21}, $B_{21,}$ B_{22}, B_{41}, B_{42}, B_{61}, and B_{62} present in the above equation are function of vehicle's parameters. The above equation is final state space equation of the vehicle. Controlling of the system is divided into two subsystems:

1. Controlling forward and backward motion of vehicle:

$$\begin{bmatrix} \dot{X}_{RM} \\ \dot{V}_{RM} \\ \dot{\theta}_P \\ \dot{\omega}_P \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & A_{23} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & A_{43} & 0 \end{bmatrix} \begin{bmatrix} X_{RM} \\ V_{RM} \\ \theta_P \\ \omega_P \end{bmatrix} + \begin{bmatrix} 0 \\ B_{21} \\ 0 \\ B_{41} \end{bmatrix} \tag{3}$$

2. Controlling Yaw motion (Left or right):

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 \\ B_{61} \end{bmatrix} [C_\theta] \tag{4}$$

The (1), (3), and (4) have been referred from [2]. From above equation, we can get the state space equation of vehicle of any dimension and it can be simulated in MATLAB. For the prototype made, the values are as follows:

M_P = 100 g, M_R = 150 gm, L = 15 cm, D = 20 cm, R = 3.5 cm,
X = 0.0154, Y = 15.46, $A_{23} = -2.927$, $A_{43} = 63.636$,
$B_2 = 134.11$, $B_4 = -1003.896$, $B_6 = 591.133$.

The meaning of each parameter is given in Table 1.

Table 1 Various symbols used

θ_P	Angle describing pitch movement
C_L, C_R	Torque applied to control left and right wheel
Θ_d	Disturbance angle due to controlling action
δ	Rotation angle in vertical axis (Yaw)
R	Radius of wheel
M_p	Mass of platform
D	Lateral distance between contact patches of the wheel
L	Distance between z axis and the CG of the platform
X_{RM}	Linear position of the chassis
V_{RM}	Linear speed of the chassis

4 Control Strategy

After the mathematical modeling of the vehicle, next step is to formulate control strategy for balancing the vehicle. Already in past, many strategies have been used like PID controller, Fuzzy logic, Neural Network, State feedback controller, and many more. In the current work, Fuzzy-based control strategy is chosen as in [2] as it is one of the better control strategies and can give accurate results with proper tuning. The output from the fuzzy logic controller will be produced in a very less elapsed time.

In this application, the fuzzy-based controller will have two input values. One input will be from the accelerometer which gives the orientation of vehicle and second input is the reading of potentiometer which is used for the steering purpose. From these two input values, the fuzzy controller has to generate logic to control both wheels for getting required movement of the vehicle. A fuzzy logic controller uses fuzzy logic to determine the actions to be taken. The procedure involved in generation of the fuzzy logic is as follows:

1. Input scaling and shifting,
2. Fuzzification,
3. Fuzzy inference, and
4. Defuzzification.

In input scaling and shifting step, the inputs are in crisp domain which then processed and scaled. Then in fuzzification, the inputs whose values are crisp in nature are converted into fuzzy domain. In fuzzy inference, a proper control action is determined, that is the output value for the given input value is obtained with the help of rule table. At last, the fuzzy-based linguistic term is converted to deterministic output in the defuzzification step. MATLAB is used for formulating the fuzzy controller.

In MATLAB, there is a fuzzy tool box which is used to formulate fuzzy rule table. Here the fuzzy logic controller has two inputs and two outputs. From the input, we get the value of the vehicle orientation; based on that output value is generated which is used to control the left and right wheel. The control in left and right wheel is done by using PWM pulse.

Inputs that used to calculate the vehicle's orientation and steering are *fwd-bkd* and *left-right,* respectively. Outputs are *left_pwm* and *right_pwm*. Membership functions that are used for *fwd-bkd* are stop, slow, medium, and fast. These all membership functions are of trapezoidal shape. Membership functions that are used for left-right are *steady*, *soft-left, hard-left, soft-right*, and *hard-right*. These all membership functions are also of trapezoidal shape. Output membership functions used here are *stop, slow, medium*, and *fast*. With each membership function has weights of 0, 60, 150, and 255. Fuzzy inference system used here is Sugeno method [9] which is based on weighted average method. The detail of fuzzy controller implementation in fuzzy tool box is given in Figs. 2, 3 and 4.

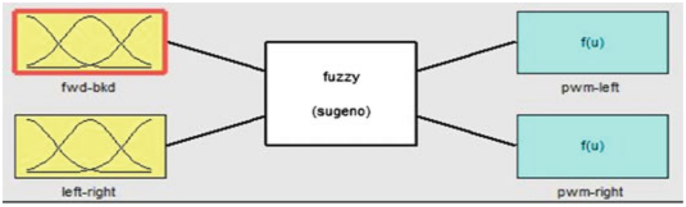


Fig. 2 Fuzzy controller in fuzzy tool box

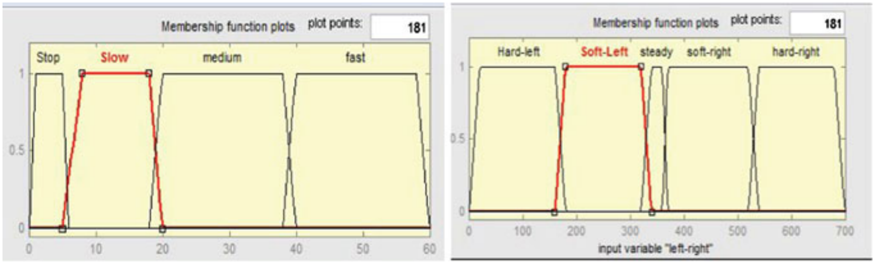
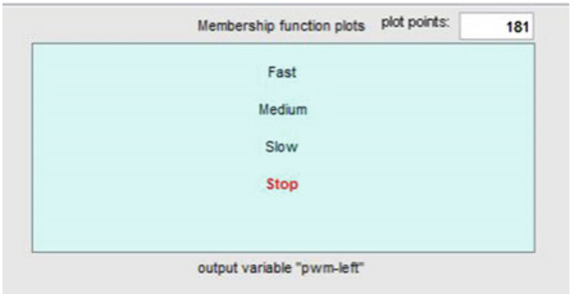


Fig. 3 Input membership function for accelerometer and steering

Fig. 4 Output membership function



Membership Function details:
Stop: Vehicle should be stopped, fast: Vehicle should have maximum speed,
Slow: Vehicle should have slow speed, medium: Vehicle should have medium speed,
Hard_left: Have sharp left turn, Soft_left: Have soft-left turn,
steady: Vehicle going straight, hard_right: Have sharp left turn,
soft_right: Have soft-left turn
Membership Function details:
Stop: Motor pwm value is 0, Slow: Motor pwm value is 60,
Medium: Motor pwm is 150, Fast: Motor pwm is 255 (maximum).
There are two inputs, one with 4 membership functions and another with 5 membership functions; so considering all the possible combinations, the fuzzy

rule base table comprises of 20 rules. Based on each combination corresponding output membership function has been mapped. The rules taken are as follows:-

1. If input is Slow and steady then output should be Slow for both wheels.
2. If input is medium and hard_right then output should be Stop for right wheel and medium for left wheel.

Similarly the rest of rule table is formed.

5 Result Analysis

5.1 Software Results

The fuzzy logic formulated for the vehicle is tested on the basis of fuzzy rule base table. The logic was tested in fuzzy tool box which was available in MATLAB. For various combinations of input values, output value is noted. The obtained results are found to be accurate. For testing purpose, the input value was given as *fwd-bkd* = 30 and *left-right* = 350, that means vehicle has to go in slow speed in straight direction (according to rule base) and the obtained output is *pwm-left* = 150 & *pwm-right* = 150, which is the required output. Similarly for every possible input combinations, output is checked and the controller is tuned according to it for getting better result.

5.2 Hardware Implementation Result

After formulating the fuzzy logic and simulating it, the next phase is to build a prototype. The prototype is built as shown in Fig. 5. Upon developing the prototype, the next step is to dump the fuzzy logic concept inside the Arduino board. After dumping the fuzzy logic code, the prototype was again tested. After proper

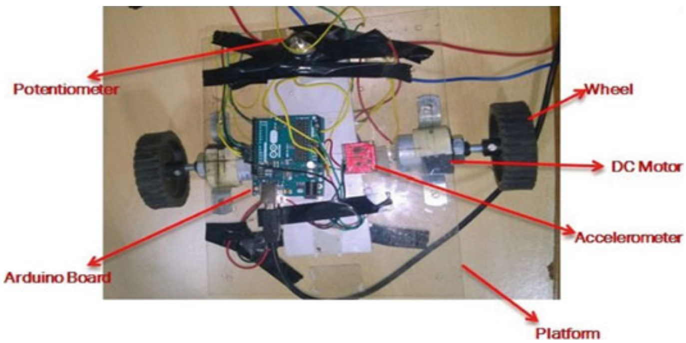


Fig. 5 Prototype of two-wheeled self-balancing vehicle

tuning of fuzzy logic code, balancing of vehicle was improved. The proper working of the prototype implied that the formulated fuzzy logic is correct and the vehicle can be balance with the help of accelerometer. The hardware prototype is shown in below figure.

6 Conclusion and Future Scope

This paper proposes the working and implementation of two-wheeled self balancing vehicle based on fuzzy logic control. Knowledge of inverted pendulum concept is essential for proper mathematical modeling of the vehicle, which is dealt initially. For accurate and fast controlling of the vehicle, a fuzzy logic controller was chosen. Fuzzy logic is implemented successfully in MATLAB tool box and desired outputs were obtained. A miniature prototype of the vehicle is build with the fuzzy logic embedded on it. The working and performance of the prototype was tested with the developed control strategy and acceptable results were obtained. Still there are possibilities of more effective control and error reduction in the current scheme. For secured access of the vehicle, finger print sensor can be attached, which may be the directions of future work.

References

1. Mohtasib, A.M., Shawar, M.H.: Self-balancing two wheel electric vehicle. In: 9th International Symposium on Mechatronics and Its Applications (ISMA13), Amman, Jordan, 9–11 April 2013
2. Wu, J., Zhang, W.: Design of fuzzy logic controller for two wheeled self balancing robot. In: 6th International Forum on Strategic Technology (2011)
3. Lin, S.-C., Tsai, C.-C., Huang, H.-C.: Nonlinear adaptive sliding-mode control design for two-wheeled human transportation vehicle. In: Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, USA, October 2009
4. Juang, H.-S., Lum, K.-Y.: Design and control of a two-wheel self-balancing robot using the Arduino microcontroller board. In: 10th IEEE International Conference on control and Automation(ICCA), Hangzhou, China, 12–14 June, 2013
5. An, W., Li, Y.: Simulation and control of a two wheeled self-balancing robot. In: IEEE International Conference on Robotics and Biomimetics (ROBIO) Shenzhen, China, December 2013
6. Chuang, J.-K., Lu, P.-Y., Leu, Y.-G., Lin, Y.-S.: Bacterial-based control of two wheeled vehicles. In: 11th IEEE International Conference on Control & Automation (ICCA), Taichung, Taiwan, 18–20 June, 2014
7. Grasser, F., Colombi, S., Rufer, A.C.: JOE : A mobile, inverted pendulum. IEEE Trans. Ind. Electron. **49**(1) (2002)
8. Tsai, C.-C., Huang, H.-C., Lin, S.-C.: Adaptive neural network control of a self balancing two wheeled scooter. IEEE Trans. Ind. Electron. **57**(4) (2010)
9. Cai, Z., Su, C.-Y.: Output tracking control of Takagi-Sugeno fuzzy systems with application to an underactuated robots. In: Proceedings of the 2002 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'02 (2002)