

Developing low-cost two wheels balancing scooter using proportional derivative controller

Petrus Sutiyasadi¹, Manukid Parnichkun²

¹Mechatronics Department, Faculty of Vocational, Sanata Dharma University, Yogyakarta, Indonesia

²Mechatronics Department, School of Engineering and Technology, Asian Institute of Technology, Pathum Thani, Thailand

Article Info

Article history:

Received Apr 29, 2021

Revised Dec 29, 2021

Accepted Jan 10, 2022

Keywords:

Balancing scooter

Proportional derivative

controller

Segway

Two wheels

ABSTRACT

This study suggests a strategy for creating a low-cost two-wheeled balancing scooter with a simpler system that can work similarly to the commercial ones on the market. A simple system will help people to understand how it works and how to build it. The mechanical parts were made from simple hollow bar iron. Wheelbarrow wheels were attached to two electric bicycle motors, and the controller was using an 8-bit microcontroller running a proportional derivative (PD) controller. PD controller only is not enough to run the scooter with a passenger smoothly. Some strategies were added to overcome some non-linear problems due to the use of low-cost components. Finally, the system is successfully built and can be ride by a 65 kgs weight of rider. The scooter can turn left or right, and even to make a 360-degree spot-rotation.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Petrus Sutiyasadi

Mechatronics Department, Faculty of Vocational, Sanata Dharma University

Yogyakarta, Indonesia

Email: peter@usd.ac.id

1. INTRODUCTION

Two wheels balancing platform is a platform that shows the principles of a dynamic control system. It is one of the implementations of an inverted pendulum [1]. This kind of balancing platform is applied in many applications, including transportation systems in the form of a two-wheeled balancing scooter. The scooter uses several sensors to form a closed-loop control to know its position and orientation [2]. A gyroscope and accelerometer provide information to calculate the orientation of vertical direction [3]. If there is an encoder, it will check the rotation direction of the motor. This type of platform has good maneuverability and requires a tiny space to make a turn [4]. The Segway Company makes a commercial two wheels balancing scooter called Segway personal transporter (Segway PT). This product is quite strong because it uses a brushless motor with neodymium magnets and powered by a 72 volts battery. It uses a digital signal processing board from Delphi electronics and a signal processor from Texas instruments. It uses a redundant system to anticipated system failure and keeps the rider in safety. It has a backup battery in case the main battery power drops. With the backup battery, the rider can bring the Segway to some charging spots available. A different prototype, a mobile inverted pendulum but uses a similar technique to balance the system, is called JOE [5]. JOE is a two-wheeled balancing platform without a passenger.

Some researchers studied this balancing platform as an inverted pendulum model [6]–[9]. Other researchers investigated balancing platforms on a ball instead of on wheels [10], [11]. This balancing platform is a nonlinear system. However, a linear controller such as proportional–integral–derivative (PID) can balance the system [12]–[16]. To get a better response in controlling a nonlinear system using PID, some researchers use hybrid PID [17]–[19]. To enhance the use of PID in controlling the wheeled balancing

platform, fuzzy control is employed together with the PID control [20]. A fuzzy controller is used to calculate the PID gain on the fly [21]. Another approach uses only a fuzzy controller [22] and a double loop fuzzy controller [23] to control a self-balancing robot. Another approach is using optimized PID for a balancing system [24], optimal control [25], [26] and adaptive control [27]. A comparison of an optimized PID and an optimized fuzzy show that optimized fuzzy performs faster and less jerky [28]. Backpropagation neural network is also capable of navigational control of a two-wheeled balancing robot in an unknown terrain [29].

A two-wheeled balancing platform as a personal transporter should provide stability on each of its maneuvers during the riding. The most critical consideration in designing this type of scooter is stability. That is due to the simplification of its modules, including the controller. All of the papers listed above say little on solving weaknesses like backlash, low power motor drivers, or too much noise while using low-cost sensors. These issues emerge as we create a rideable balanced scooter using low-cost parts. The remainder of the paper is organized as follows; section 2 describes the research methodology. Section 3 shows the result and analysis. Section 4 presents the conclusions.

2. RESEARCH METHOD

2.1. Mechanical design

The structure of the scooter is made from square hollow bar iron. The overall mechanical design is made using SolidWork software. The construction consists of two major parts: the handlebar part and the base part. Both of them are combined using bolt and nut at the lower part of the handlebar. Its controller, direct current (DC) drivers, DC motors, sensors, and batteries are put on the base part. The base part is connected to the wheels. The handlebar part is used to mount battery indicator, switches, and steering potentiometer. There is an emergency killing switch as well. The switch is used to cut the power immediately in case the system is not stable.

The center of gravity (CoG) of the system should lie on the center. All of the components, especially those that have more weight should be put evenly. The controller can balance the system dynamically. But if the CoG is lied rather to the front, the scooter will balance itself with the handlebar tilted back to maintain the CoG position on the center. This situation is not comfortable at all for the passenger. Tilting the handlebar forward or backward will move the system forward or backward. So, to ease the works of the controller, the handlebar should be designed to stay vertical during the static balance. The scooter tends to move in the direction where the system is statically unbalanced. The motor has a built-in gearbox with an 88:9 ratio. Another ratio was added using a sprocket system with a 2:1 ratio. Figure 1 shows the mechanical design of the scooter. Figure 2 shows the free body diagram of a two wheels balancing platform. The free body diagram of the left and the right wheels are shown in Figure 2(a). While Figure 2(b) shows the free body diagram of the chassis and the handle bar. The linearized state space equation is [30]:

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

with

$$\beta = \left(2M_w + \frac{2I_w}{r^2} + M_p \right) \dots \alpha = \left[I_p \beta + 2M_p I^2 \left(M_w + \frac{I_w}{r} \right) \right]$$

Variables used on the (1) are listed: k_m is torque constant, k_e is back EMF constant, R is armature resistant, V_a is applied voltage, $I_w I_w$ is inertia of the wheels, r is radius of the wheels, $I_p I_p$ is inertia of the chassis, ϕ is scooter's leaning angle, $\dot{\phi}$ is angular velocity of the scooter's leaning angle, $\ddot{\phi}$ is angular acceleration of the scooter's leaning angle, l is distance between wheel and center of gravity, x is horizontal displacement, \dot{x} is horizontal velocity of the scooter, \ddot{x} is horizontal acceleration of the scooter, M_w is mass of the wheel, M_p is mass of the chassis, and g is gravity. After inputting the system parameters into (1), the state space equation and the transfer function are obtained as,

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -0.06626 & 21.02 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -0.03092 & 15.42 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 0.2721 \\ 0 \\ 0.127 \end{bmatrix} V_a \tag{2}$$

$$G(s) = G_1(s) + G_2(s) \tag{3}$$

$$G_1(s) = \frac{X(s)}{V_a} = \frac{0.2721s^2 + 2.417e-16s - 1.526}{s^4 + 0.06626s^3 - 15.42s^2 - 0.3714s} \tag{4}$$

$$G_2(s) = \frac{\varphi(s)}{V_a} = \frac{0.127s}{s^3 + 0.06626s^2 - 15.42s - 0.3714} \tag{5}$$

A PD controller applicable on a single-input single-output (SISO) system only. Therefore, only scooter inclination transfer function will be considered. Figure 3(a) shows the overall control system block diagram. Since the reference angle is zero, the block diagram can be arranged like in Figure 3(b) to make it easier to analyze. Therefore, the equation become as in (6).

$$\frac{\Phi(s)}{F(s)} = \frac{P(s)}{1 + C(s)P(s)} \tag{6}$$



Figure 1. Mechanical design of the scooter

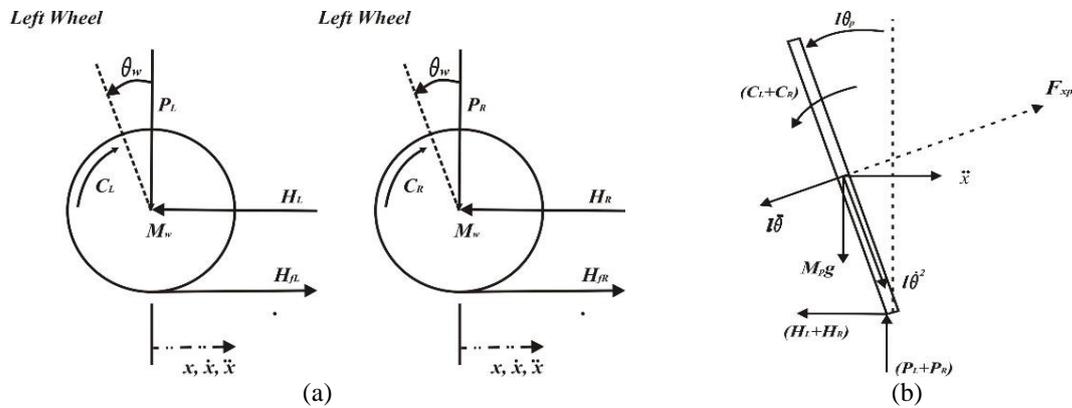


Figure 2. Free body diagram of the balancing scooter (a) free body diagram of the wheels and (b) free body diagram of the chassis and the handle bar

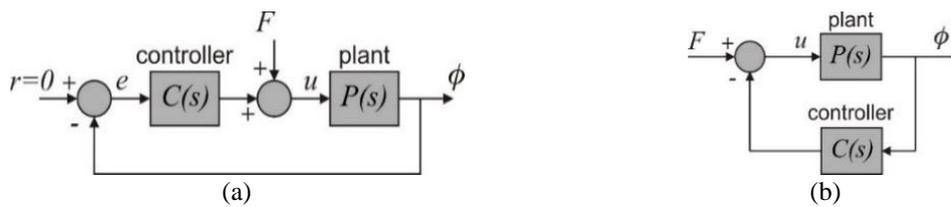


Figure 3. Re-arrange of the scooter closed loop system block diagram to get inclination angle response from an impulse disturbance (a) the closed loop block diagram of the system and (b) rearranged block diagram to make it easier to be analyzed

2.2. Electronic design

The electronic part of the system consists of a controller, DC motor driver, sensor, battery, user interface, and battery indicator. Figure 4 shows the block diagram of the control system. The sensor signals consist of angle and angle rate of the system refer to the vertical direction. A proportional derivative (PD) algorithm was employed to calculate the output signal. The output signal was translated into pulse width modulation (PWM) signal to rotate the DC motors.

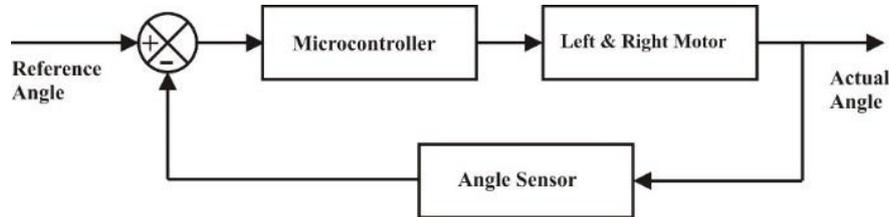


Figure 4. Closed loop diagram block of the scooter

The tilt angle was provided by an accelerometer combined with a gyroscope packed into a single vertical gyro VG400. The accelerometer signal and the gyro signal were fused using Kalman fusion sensor or Kalman filter to get the angle value. Nowadays, a combo inertial measurement unit (IMU) consist of an accelerometer and gyroscope is available at a very cheap price. PD controller calculates the appropriate output for the motor using the angle and the rate angle information. All the calculations were done inside the microcontroller. The scooter uses a PIC16F877 microcontroller. Figure 5 shows the controller box that is consists of a PIC16F877 Microcontroller and a small battery to power the microcontroller, DC driver circuit, and vertical gyro sensor. A 16x2 LCD is put on the box to show the PD parameters such as the proportional gain and the differential gain. Outside the box, there are two potentiometers to set the PD gain and terminal for serial communication. The microcontroller sends the PD output to the DC motor driver circuit. The DC driver circuit uses the MOSFET IRF3205 H-Bridge circuit. The actuators of the scooter are two 250 watt DC motors connected to two 13-inch wheelbarrow wheels. The two DC motors get the power from two 12 volt dry cell batteries. Table 1 shows the list of the scooter's electronic components.

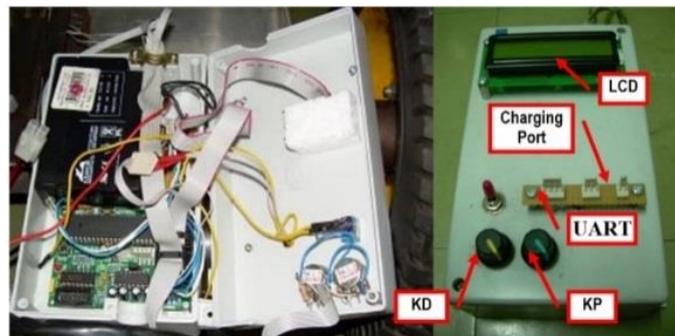


Figure 5. Controller box of the scooter

Table 1. Electronic components specification

No	Function	Components
1	Tilt sensor	combo VG400 vertical gyro
2	Motor Driver	MOSFET IRF3205 H Bridge
3	Actuator	MY1060 250 watt DC Motor
4	Power Source	Dry Cell Battery 12 V

2.3. Software design

The scooter will balance itself by moving the platform toward the leaning direction of the handlebar. Information in which direction the system fall, how much the leaning angle now, and how fast the angle changes between times are important. A proportional and derivative controller is selected to manage all those

information and calculate the torque needed to balance the system. The torque output for the motor is calculated by multiplying the angle and the rate angle with some constants. The PD controller output was converted into PWM value and sent to the DC motor driver. The DC motor will produce torque according to the PWM signal sent to the DC motor driver. Both the left and right motors received the same amount of value of PD output. However, the steering angle command differentiated the final control value sent to each of the DC motors. The algorithm of the PD controller is shown:

$$P = K_p e \quad (7)$$

$$D = K_d \frac{de}{dt} \quad (8)$$

$$PD = P + D \quad (9)$$

P is the value of the proportional controller, D is the derivative controller value, and PD is the overall controller output. The e is the error between the actual angle and the reference angle which is zero degree. It tells how much the tilting angle of the scooter's handlebar is. The $\frac{de}{dt}$ is the derivative of the error. It tells how fast the scooter is handlebar falling backward or forward. The time cycle of the controller is 20 ms. At a very small PWM value, the wheels will not move. It needs to add a constant from the output value. However, the system needs a small dead band angle to improve the stabilization process by eliminating chattering around the equilibrium.

2.3.1. Sensor fusion using Kalman filter

The output generation speed of the accelerometer signal is fast but full of noise. While the output of the gyro signal is disturbed by gyro drift. Kalman Filter was used to fusing both of the signals to get better angle reading. Kalman filter is a linear estimator. It estimates the states in the presence of process noise and measurement noise. The process noise and the measurement noise are modeled as white noise [31]. Kalman filter is popular in autonomous navigation research [32]. The mean squared error between the estimate and the actual data is minimized. Consider the following equations:

State equation:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (10)$$

Output equation:

$$z_k = Hx_k + v_k \quad (11)$$

Variable x is the state variable, and z is the measurement value with the measurement noise. In this case, the state variables are output angle and gyro bias/drift. A and B are the matrices of the states. The process noise is represented by variable w , and the measurement noise is represented by variable v . Both of the parameters are random variables and assumed to have a normal probability distribution. The covariances Q and R also have a normal distribution and independent of each other. The covariance R can be determined based on the real measurement variance. While Q can be determined intuitively. Based on the experiment, if we put more on the Q accelerometer the output response is fast but noisy. If we put more on the Q gyro, the output response is slower but less noisy. The best response for the tilt feedback is Q accelerometer=0.001 and Q gyroscope=0.003. More detail on how to define matrices Q and R can be found in [33]. Kalman algorithm predicts the state estimates. It has two continuous processes which are "priors" and "posteriori" state estimate or "prediction" and "update" stages. The update is done by the sensor reading.

Initial error covariance and Initial estimated state must be described in the priory state estimate. Other parameters that should be put in the Kalman filter loop are the noise covariance R_0 and the process noise covariance Q_0 . The predictor equations are as (12), (13):

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (12)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (13)$$

The (12) calculates state estimate of the state variable, and error covariance estimate is calculated in (13). Error covariance is the error between the estimate of x and the true state. The following are the update equations:

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (14)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H \hat{x}_k^-) \quad (15)$$

$$P_k = (I - K_k H) P_k^- \quad (16)$$

The error covariance P will be minimized by gain K, H and R are the measurement matrix and the measurement covariance matrix respectively. The “a posteriori” state estimate is calculated in (15). P_k is the covariance matrix of the “a posteriori”.

Both gyroscope and accelerometer have their drawbacks. The gyroscope signal tends to drift along the time, and the accelerometer has significant measurement noise. To take only their advantages needs a little bit of effort. Kalman fusion sensor is one of a good way for that. By combining some information, the Kalman filter gets a good estimate of the actual tilt angle. Kalman filter was applied with angle and gyro bias as the state variables. The steps of the Kalman Filter cycle is explained:

Step 1. Variable initialization

- a) Determine the frequency of the gyro rate measurement to update the state.
- b) Determine R (measurement covariance noise)
- c) Determine R (process covariance noise)

Step 2. Routine updated every “d_t”

- a) Determine update of the covariance matrix

$$P_{k+1} = P_{k+1} + \dot{P} d_t$$

$$\dot{P} = A P_k + P_k A' + Q$$

- b) Determine update of the estimated state

$$\theta += g d_t$$

$$g = gyro_measurement - gyro_bias$$

Step 3. When the accelerometer reading is ready, the routine is updated.

- a) Determine the Kalman gain $K = P_{k+1} H^T (H P_{k+1} H^T + R)^{-1}$
- b) Update the covariance matrix $P_{k+1} = (I - KH) P_{k+1}$
- c) Update the state estimate $X += K (accelerometer_measurement_angle - \theta)$

2.3.2. Custom strategies for controlling the two wheels balancing scooter

A balancing scooter is a highly non-linear system. On the other hand, a PD controller is a linear controller. Therefore, there are some non-linear problems on the system that a PD controller cannot handle. We need to overcome those problems by adding some strategies to the controller. The biggest problem is when the scooter cannot pull back the handlebar when the passenger sets its leaning angle too big by pushing it too far. Secondly, a low-cost DC motor with a gearbox usually has a big backlash. A sudden change in the direction of the motor rotation will cause a hard impact on the gearbox or the chain sprocket system. The system's response may oscillate and lead to instability. Figure 6(a) shows the main loop of the controller. Besides the PD controller, there are exceeding angle routine, backlash routine, and steering routine. Figure 6(b) shows the steering routine. Exceeding angle routine is used to limit the forward tilting angle. The forward velocity is proportional to the leaning forward angle. The bigger the angle the faster the scooter will run. The speed is limited by limiting the forward tilting angle to prevent over-speed. If the leaning angle is too big, the system will push the handlebar backward. This also prevents the rider from falling forward if the scooter cannot reach the required speed to tilt back the handlebar. The routine flowchart is shown in Figure 6(c).

The scooter uses a sprocket and chain of a bicycle. It has a significant backlash that causes a significant impact when the rotation direction of the motor is changed. If it happened near the equilibrium position, it generates hardware vibration. It makes the mechanical part easy to be worn out. It also gives significant disturbance to the balancing process. The routine of the anti-backlash is shown in Figure 6(d) it is done by rotating the sprocket fast in just several milliseconds. Enough to fill the backlash gap and then continue to follow the PD routine until the direction of the motor is changed again.

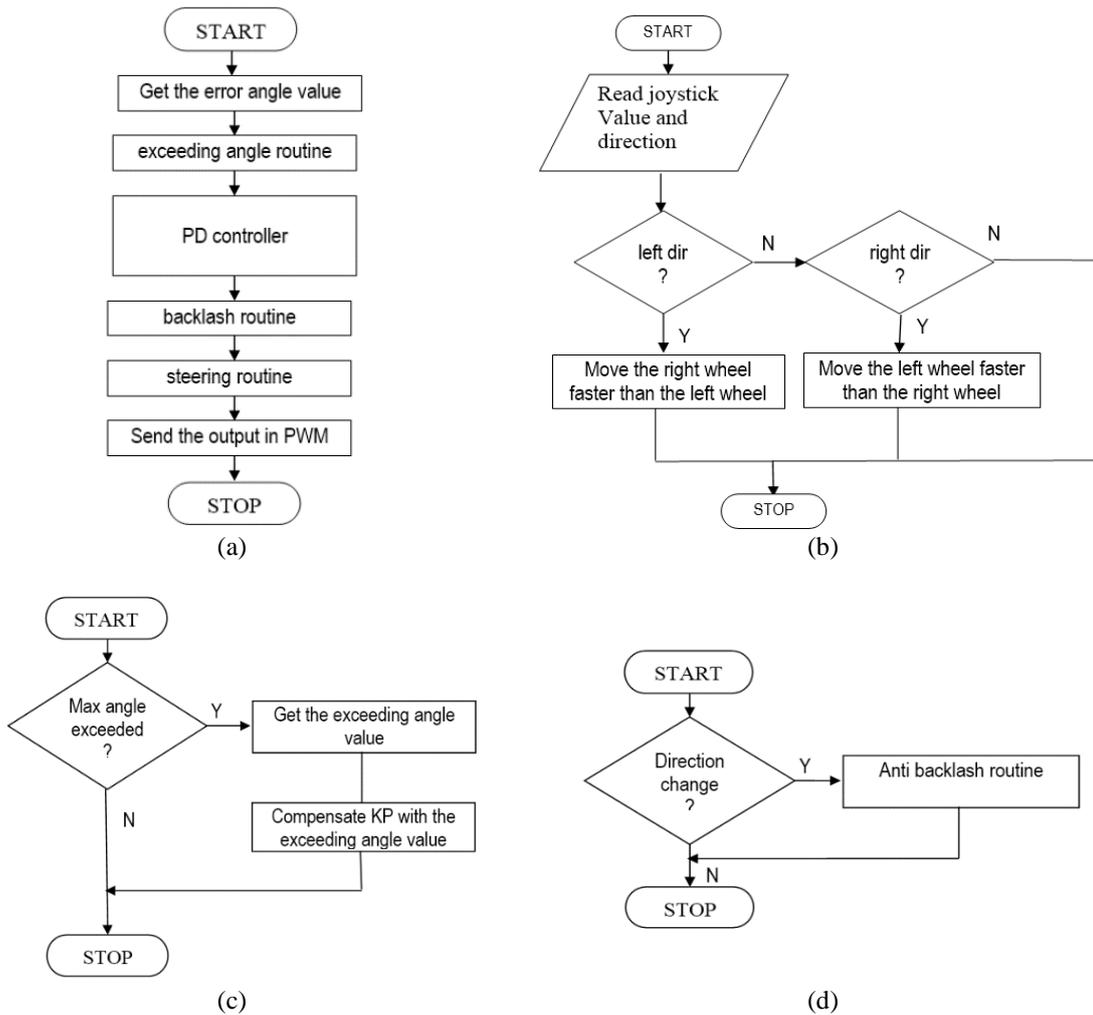


Figure 6. Flowchart of the scooter control algorithm (a) main controller loop, (b) steering algorithm, (c) push handlebar back if exceeding max angle, and (d) anti backlash algorithm

3. RESULTS AND ANALYSIS

One method to tune the PID constant is Ziegler Nichols tuning method [34]. By using only proportional controller, the scooter’s system response was plotted. The proportional gain was gradually increased until the system response oscillates with a constant amplitude. The gain on that stage then called as the ultimate gain, and the period of the oscillation called as the ultimate period. By using Ziegler Nicholes table, the PID gain can be calculated. The ultimate oscillation due to the ultimate gain from the experiment is shown in Figure 7. The achieved ultimate gain and the ultimate period are 6 and 1.088 second respectively. Based on the Ziegler Nichols formula, the proportional gain is 3.6 and the derivative gain is 0.49.

Simulation result of the closed loop system from (6) is shown in Figure 8. The result shows that using PD controller the scooter system can be balanced. Even when there is exist an impulse force on the system, it can balance itself and move the scooter handlebar back in zero degree again. The scooter balancing performance using the result from the Ziegler Nichols method is shown in Figure 9. The scooter can stay balanced around the equilibrium or zero degree. But it oscillates with the amplitude in the average of ± 2 degree.

The Ziegler Nichols, however, can give good starting value of the PD gain. But it needs manual fine tuning to make the system more stable. By adding fine manual tuning, the best PD gains achieved are $K_P=3.6$ and $K_D=5.39$. K_D is slightly higher because it is intended to reduce the overshoot. This combination of PD gains satisfies the stability requirement of this system. The system response in Figure 10 shows an acceptable result as the oscillation does not go beyond 1.5 degrees. The oscillation mostly happens at a negative angle. It means that the handlebar a little bit leans in the forward direction. The negative angle of the system indicates that the transporter leans to the front and makes an angle against the vertical direction. If the system

tends to lean forward or backward, it can be corrected by adjusting the weight distribution of the parts or components inside the system. If the angle is not so big, it can be corrected by changing the reference angle. Changing the reference angle can fix the relative vertical position of the system. However, the platform might be slightly inclined.

Figure 11 and Figure 12 show the inclination angle and the controller output signal to correct the position without and with passenger respectively. The output response of the controller follows the algorithm properly. On the application, the system limits the angle to 20 degrees. If the system angle is more than 20 degrees, the controller will not send any output to the motor. This is done to protect the rider so that the system will not run too fast when the angle is too big.

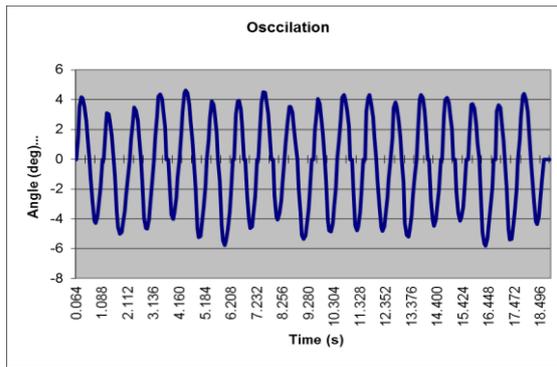


Figure 7. The ultimate oscillation result from the Ziegler Nichols PID tuning method

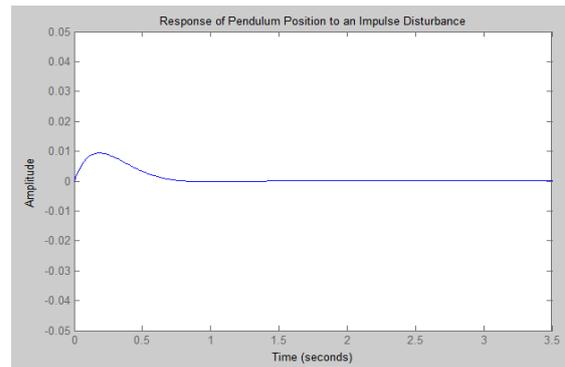


Figure 8. Simulation result of a scooter response to an impulse input

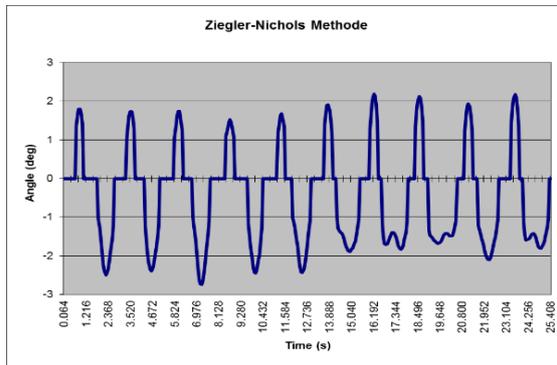


Figure 9. The scooter balancing performance with Ziegler Nichols PID gain tuning

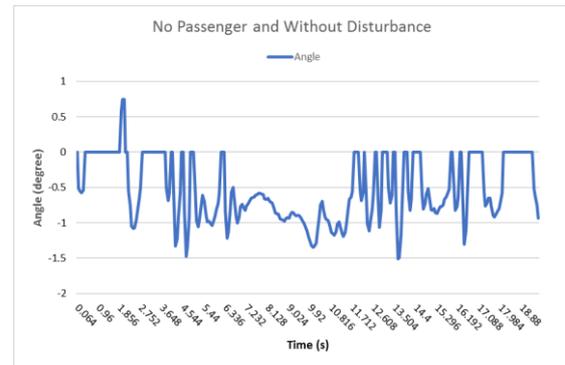


Figure 10. The scooter balancing performance with manual PD gain tuning.

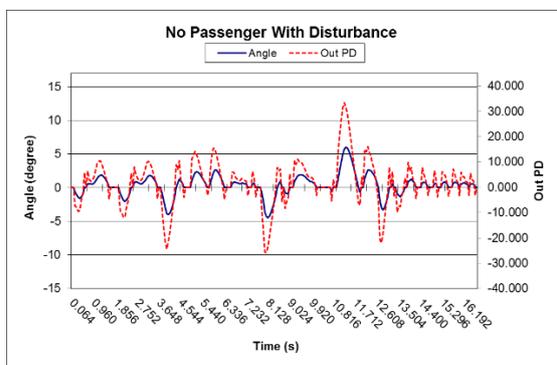


Figure 11. Scooter without passenger balancing performance

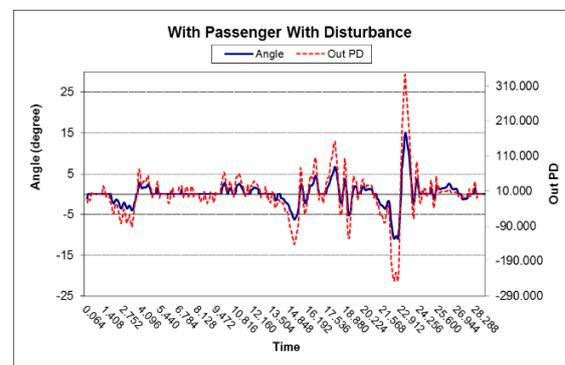


Figure 12. Scooter with passenger balancing performance

Figure 13 shows the scooter testing on a tennis court. Figure 13(a) shows when the scooter at rest and Figure 13(b) shows when the scooter has a rider on it. The scooter moved forward when the handlebar was pushed forward. On the opposite, it runs backward when the handlebar is pulled backward. The bigger the angle the faster the system will move.

The faster the motor is rotated, the lesser the torque is produced. This will affect the stability of the system. Motor torque is reduced because the voltage from the battery is reduced by back EMF generated by the motor itself. Moreover, if the scooter moves on a rough surface, it will experience larger disturbance. The future recommendation is to study the variation of the passenger weight to the stability of the system. Different inertia needs a different set of PID gain. Gain scheduling or PID adaptive can be a good candidate for future investigation. Another approach is using a robust controller. H_∞ robust controller with a simple structure such as PID structure is also a good controller for a small embedded system [35]. With a robust controller, the system is stable even in the presence of uncertainties including a variation of passengers.

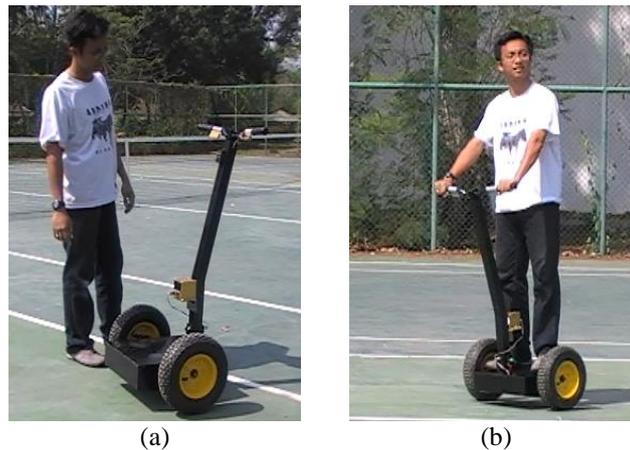


Figure 13. Hardware experiment on a tennis court (a) Scooter at rest position without passenger and (b) Scooter cruises with passenger

4. CONCLUSION

The personal transporter or the balancing scooter is successfully developed. The scooter can be ridden and controlled through the leaning handlebar and the steering potentiometer. A rider with maximum 65 kgs of weight successfully rides the scooter and does some maneuvers with it. Limitation of 65 kgs merely due to the mechanical reason. The chassis that support the passenger is made from sheet metal. It starts to deform when someone more than 65 kgs stands over it. The scooter has good maneuverability in limited space. It can rotate 360 degrees in a spot. The system also has a good balance in the reverse direction. The oscillation around the equilibrium takes less than plus-minus 1.5 degrees which is neglectable by the rider since they will not realize it on the handlebar. When it has a rider on it, the oscillation is reduced due to the damping effect from the rider's hand.

REFERENCES

- [1] H. J. T. Smith and J. A. Blackburn, "Experimental study of an inverted pendulum," *American Journal of Physics*, vol. 60, no. 10, pp. 909–911, Oct. 1992, doi: 10.1119/1.17012.
- [2] O. Mayr, "The origins of feedback control," *Scientific American*, vol. 223, no. 4, pp. 110–118, Oct. 1970, doi: 10.1038/scientificamerican1070-110.
- [3] N. Yazdi, F. Ayazi, and K. Najafi, "Micromachined inertial sensors," in *Proceedings of the IEEE*, 1998, vol. 86, no. 8, pp. 1640–1659, doi: 10.1109/5.704269.
- [4] M. R. M. Romlay, M. I. Azhar, S. F. Toha, and M. M. Rashid, "Two-wheel balancing robot; review on control methods and experiments," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 7, no. 6S, pp. 106–112, 2019.
- [5] F. Grasser, A. D'Arrigo, S. Colombi, and A. C. Rufer, "JOE: a mobile, inverted pendulum," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 1, pp. 107–114, 2002, doi: 10.1109/41.982254.
- [6] M. U. Draz, M. S. Ali, M. Majeed, U. Ejaz, and U. Izhar, "Segway electric vehicle," in *2012 International Conference of Robotics and Artificial Intelligence*, Oct. 2012, pp. 34–39, doi: 10.1109/ICRAI.2012.6413423.
- [7] T.-P. A. Perdicoulis and P. L. Dos Santos, "The secrets of segway revealed to students: revisiting the inverted pendulum," in *2018 13th APCA International Conference on Control and Soft Computing (CONTROLO)*, Jun. 2018, pp. 43–48, doi: 10.1109/CONTROLO.2018.8514296.

- [8] K. Zakova and M. Hok, "Interactive three dimensional presentation of Segway laboratory model," in *2016 International Conference on Emerging eLearning Technologies and Applications (ICETA)*, Nov. 2016, pp. 377–380, doi: 10.1109/ICETA.2016.7802064.
- [9] T. Kuwata, M. Tanaka, M. Wada, T. Umetani, and M. Ito, "Localization of segway RMP," in *SICE Annual Conference 2011*, 2011, pp. 1675–1680.
- [10] P. D. Ba, S.-G. Lee, S. Back, J. Kim, and M. K. Lee, "Balancing and translation control of a ball segway that a human can ride," in *2016 16th International Conference on Control, Automation and Systems (ICCAS)*, Oct. 2016, pp. 477–480, doi: 10.1109/ICCAS.2016.7832362.
- [11] S. Back and S.-G. Lee, "Modeling and control system design of a ball segway," in *2016 16th International Conference on Control, Automation and Systems (ICCAS)*, Oct. 2016, pp. 481–483, doi: 10.1109/ICCAS.2016.7832363.
- [12] Y. G. Rashid and A. M. A. Hussain, "Implementing optimization of PID controller for DC motor speed control," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 23, no. 2, pp. 657–664, Aug. 2021, doi: 10.11591/ijeecs.v23.i2.pp657-664.
- [13] M. Abdulelah Alawan and O. J. Mohammed Al-Furaiji, "Numerous speeds-loads controller for DC-shunt motor based on PID controller with on-line parameters tuning supported by genetic algorithm," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 21, no. 1, pp. 64–73, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp64-73.
- [14] S. N. Al-Bargothi, G. M. Qaryouti, and Q. M. Jaber, "Speed control of DC motor using conventional and adaptive PID controllers," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 16, no. 3, pp. 1221–1228, Dec. 2019, doi: 10.11591/ijeecs.v16.i3.pp1221-1228.
- [15] O. A. Trujillo, N. Toro-García, and F. E. Hoyos, "PID controller using rapid control prototyping techniques," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 3, pp. 1645–1655, Jun. 2019, doi: 10.11591/ijece.v9i3.pp1645-1655.
- [16] M. H. Salehpour, H. D. Taghirad, and H. Moradi, "Two PID-based controllers for a tethered segway on dome shaped structures," in *2018 6th RSI International Conference on Robotics and Mechatronics (ICRoM)*, Oct. 2018, pp. 577–582, doi: 10.1109/ICRoM.2018.8657643.
- [17] Y. Ye, C.-B. Yin, Y. Gong, and J. Zhou, "Position control of nonlinear hydraulic system using an improved PSO based PID controller," *Mechanical Systems and Signal Processing*, vol. 83, pp. 241–259, Jan. 2017, doi: 10.1016/j.ymssp.2016.06.010.
- [18] A. Kaba, "Improved PID rate control of a quadrotor with a convexity-based surrogated model," *Aircraft Engineering and Aerospace Technology*, vol. 93, no. 8, pp. 1287–1301, Sep. 2021, doi: 10.1108/AEAT-03-2021-0093.
- [19] M. Rahman, Z. C. Ong, W. T. Chong, and S. Julai, "Smart semi-active PID-ACO control strategy for tower vibration reduction in wind turbines with MR damper," *Earthquake Engineering and Engineering Vibration*, vol. 18, no. 4, pp. 887–902, Oct. 2019, doi: 10.1007/s11803-019-0541-6.
- [20] B. Sumantri, E. H. Binugroho, I. M. Putra, and R. Rokhana, "Fuzzy-PID controller for an energy efficient personal vehicle: Two-wheel electric skateboard," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 6, pp. 5312–5320, Dec. 2019, doi: 10.11591/ijece.v9i6.pp5312-5320.
- [21] T. Qi, C. Li, W. Kai, and L. Yan, "Research on motion control of mobile robot with fuzzy PID arithmetic," in *2009 9th International Conference on Electronic Measurement and Instruments*, Aug. 2009, pp. 3–366, doi: 10.1109/ICEMI.2009.5274284.
- [22] Á. Odry, R. Fullér, I. J. Rudas, and P. Odry, "Fuzzy control of self-balancing robots: A control laboratory project," *Computer Applications in Engineering Education*, vol. 28, no. 3, pp. 512–535, May 2020, doi: 10.1002/cae.22219.
- [23] C.-F. Hsu and W.-F. Kao, "Double-loop fuzzy motion control with CoG supervisor for two-wheeled self-balancing assistant robots," *International Journal of Dynamics and Control*, vol. 8, no. 3, pp. 851–866, Sep. 2020, doi: 10.1007/s40435-020-00617-y.
- [24] P. Sutyasadi and M. Parnichkun, "Push recovery control of quadraped robot using particle swarm optimization based structure specified mixed sensitivity H_2 / H_∞ control," *Industrial Robot: the international journal of robotics research and application*, vol. 47, no. 3, pp. 423–434, Mar. 2020, doi: 10.1108/IR-06-2019-0135.
- [25] S. Chantarachit, "Development and control segway by LQR adjustable gain," in *2019 International Conference on Information and Communications Technology (ICOIACT)*, Jul. 2019, pp. 649–653, doi: 10.1109/ICOIACT46704.2019.8938489.
- [26] R. Babazadeh, A. G. Khiabani, and H. Azmi, "Optimal control of Segway personal transporter," in *2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)*, Jan. 2016, pp. 18–22, doi: 10.1109/ICCIAutom.2016.7483129.
- [27] W. Netto, R. Lakhani, and S. M. Sundaram, "Design and performance comparison of different adaptive control schemes for pitch angle control in a twin-rotor-MIMO-system," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 5, p. 4114, Oct. 2019, doi: 10.11591/ijece.v9i5.pp4114-4129.
- [28] A. Odry and R. Fuller, "Comparison of optimized pid and fuzzy control strategies on a mobile pendulum robot," in *2018 IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, May 2018, pp. 207–212, doi: 10.1109/SACI.2018.8440947.
- [29] A. Chhotray and D. R. Parhi, "Navigational control analysis of two-wheeled self-balancing robot in an unknown terrain using back-propagation neural network integrated modified DAYANI approach," *Robotica*, vol. 37, no. 8, pp. 1346–1362, Aug. 2019, doi: 10.1017/S0263574718001558.
- [30] R. C. Ooi, "Balancing a two-wheeled autonomous robot," University of Western Australia School of Mechanical Engineering, 2003.
- [31] O. Bayasli and H. Salhi, "The cubic root unscented kalman filter to estimate the position and orientation of mobile robot trajectory," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 5, pp. 5243–5250, Oct. 2020, doi: 10.11591/ijece.v10i5.pp5243-5250.
- [32] L. Lasmadi, A. I. Cahyadi, S. Herdjunto, and R. Hidayat, "Inertial navigation for quadrotor using Kalman filter with drift compensation," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 5, pp. 2596–2604, Oct. 2017, doi: 10.11591/ijece.v7i5.pp2596-2604.
- [33] Y. Zahraoui and M. Akherraz, "Kalman filtering applied to induction motor state estimation," in *Dynamic Data Assimilation-Beating the Uncertainties*, IntechOpen, 2020.
- [34] M. Nafea, A. R. M. Ali, J. Baliah, and M. S. Mohamed Ali, "Metamodel-based optimization of a PID controller parameters for a coupled-tank system," *Telecommunication Computing Electronics and Control (TELKOMNIKA)*, vol. 16, no. 4, pp. 1590–1596, Aug. 2018, doi: 10.12928/telkomnika.v16i4.9069.
- [35] P. Sutyasadi and M. B. Wicaksono, "Joint control of a robotic arm using particle swarm optimization based H_2/H_∞ robust control on arduino," *Telecommunication Computing Electronics and Control (TELKOMNIKA)*, vol. 18, no. 2, pp. 1021–1029, Apr. 2020, doi: 10.12928/telkomnika.v18i2.14749.

BIOGRAPHIES OF AUTHORS

Petrus Sutiyasadi     earned his master degree in mechatronics in 2008 and doctoral degree in mechatronics in 2016 from Asian Institute of Technology, Thailand. Currently he is a lecturer at Mechatronics Department in Sanata Dharma University, Yogyakarta, Indonesia. His research interest includes robust control, robotic, and frugal innovation in mechatronic engineering. He can be contacted at email: peter@usd.ac.id.



Manukid Parnichkun     earned his master degree in Precision Machinery Engineering in 1993 and doctoral degree in mechatronics in 1996 from The University of Tokyo, Japan. Currently he is a full professor at Industrial System Engineering Program in The Asian Institute of Technology, Bangkok, Thailand. His research interest includes new robot mechanism, novel control algorithm, and innovative measurement concept. He can be contacted at email: manukid@ait.asia.