Application Of LQR and Fuzzy-LQR Algorithms for Controlling Self-Balancing Bike Model

Vu Nguyen Tran Long
Department of Automatic Control
Ho Chi Minh City University of Technology
Ho Chi Minh City, Vietnam

Phuc Vo Hoang
Department of Automatic Control
Ho Chi Minh City University of Technology
Ho Chi Minh City, Vietnam

Thai-Hoang Huynh
Department of Automatic Control
Ho Chi Minh City University of Technology
Ho Chi Minh City, Vietnam

ABSTRACT
The self-balancing problem is crucial for the future development of self-driving technology, particularly for two-wheeled vehicles. This report investigates and analyzes a self-balancing bike model using a control system based on reaction wheel actuation. The Linear Quadratic Regulator (LQR) and fuzzy controller combined with LQR (Fuzzy-LQR) are applied to stabilize the bike by adjusting the reaction wheel's response. To ensure a comprehensive approach to model development, following a structured methodology is necessary: theoretical analysis, data collection, mathematical modeling and simulation, and real-world experimentation. The results demonstrate that both control methods can effectively stabilize the system. However, balancing performance and energy efficiency must be carefully considered for real-world applications. The Fuzzy-LQR approach performs better than the standalone LQR method, highlighting the advantages of integrating human-inspired intelligent control with traditional control techniques. This finding reinforces the potential of hybrid control strategies in handling nonlinear self-balancing bike models in practical applications.

Keywords
Self-balancing bike, LQR, Fuzzy–LQR

1. INTRODUCTION
A self-balancing bike system is a system that can be controlled in many ways, based mainly on the principle of generating torque to balance the system. The main idea is to get feedback from a sensor measuring the tilt angle of the bike, and then use a mechanism designed to balance the bike at the desired position based on this tilt angle. The self-balancing bike system is a system thathasahigh characteristic of linearity when the tilt angle is smaller than 10 degrees. There are popular methods developed for controlling this system:

- The first one is the control method of the steering angle of the bike to balance the system conducted in research [1], [2], [3]. This method uses a motor to control the handlebar of the bike, hence controlling the steering angle of the system. This method would be energy-saving, but the system will have friction and reaction forces between the wheel and the ground, because this factor will significantly affect the response time of the steering angle change to external factors, and the mathematical model would be complicated.
- The second method is related to usingthe Control Moment Gyroscope (CMG). A CMG consists of a spinning rotor and one or more motorized gimbals that tilt the rotor's angular momentum. As the rotor tilts, the changing angular momentum causes gyroscopic precession torque that balances the bike, this method is conducted inresearch [4], [5], [6]. The advantages of this system are that it can produce a large amount of torque, it has no ground reaction forces, and the system can be stable even when the bike is stationary. The disadvantages are that it consumes more energy and it is physically complex.
- The third approach is to control the system based on controlling the reaction wheel,which is conducted in reports [7], [8]. In comparison to the above approaches, the reaction wheel can dissipate energy suitably with a tilt angle to reduce energy, low cost, simplicity, and no ground force are the advantages of this method. The disadvantages of this method are energy consumption and small torque generation.

Popular algorithms, such as PID and LQR controllers, are applied to control the system with the above-mentioned approaches. Because this system is not perfectly linear and is approximately linearized for most of the controllers applied, algorithms related to PID application perform pretty well to control the system, but when noise happensto make the system out of the linearized range, the model tends to be less stable, and oscillations happen. Besides, with the real-world problems of energy optimization, while controlling is also prioritized, traditional control methods such as PID are less considered. With the problem of optimal control and robustness for the system, the Linear Quadratic Regulator (LQR) is considered one of the most suitable methods thatare applied in studies [7], [9], [10]. Because the LQR controller is developed to balance the controlling effort with the control signal, the energy used problem can be solved well. On the other hand, combining a fuzzy controller with a set of rules from human experiences would be a suitable way to improve the robustness and adaptation of the controlled system.Successful applications can be seen in reports [11], [12], [13]. In this study, the fuzzy controller is combined with the LQR controller by taking the control signal from the LQR controller and processing it into the fuzzy controller to improve the performance of the controlled system.

The rest of this paper is organized as follows: Section 2 introduces the practical bike model and presents the mathematical modeling of the bike system. Section 3 focuses on the design of the LQR controller based on the developed model, while Section 4 introduces a hybrid control approach that combines the fuzzy controller and LQR controllers. Section 5 discusses and compares the simulation and experimental results obtained from both control methods. Finally, Section 6 provides the conclusions of the study.

2. MODELLING OF SELF-BALANCING BIKE
The two-wheel bike model and mathematical modelare shown

in Figures 1 and 2, in which, m_1 is the mass of the bike (including DC motor controlling the reaction wheel), I_1 is the moment of inertia generated by the bike, A is the center of mass of the bike, h_1 is the distance from ground to A, m_2 is the mass of the reaction wheel, I_2 is the moment of inertia generated by reaction wheel, B is the center of mass of the reaction wheel, h_2 is the distance from ground to B, g is the gravitational acceleration.The real parameters of the model are displayed in Table 1.

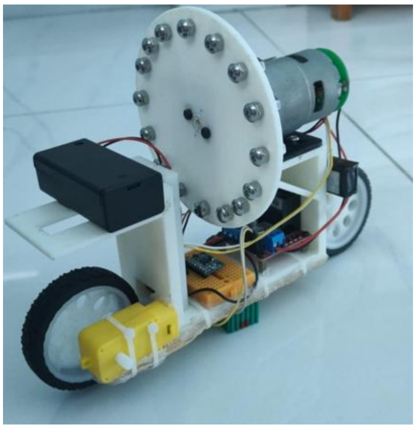

Figure 1: The 3D printed bike prototype for real-world controller application

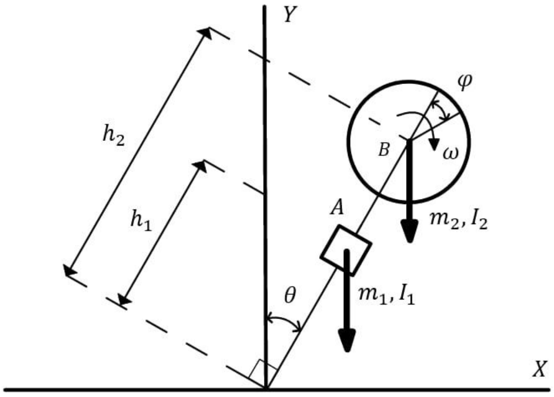

Figure2: Self-balancing bikemathematicalmodeling

From the report [6], the Lagrange equation is used to build the mathematical dynamics of the model:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial q'}\right) - \frac{\partial L}{\partial q} = \tau \#(1)$$

$$L = T - V$$

In which T is the kinetic energy of the system, V is the potential energy of the system, τ is the external torques applied to the system, $q = [\theta\varphi]^T$is the generalized coordinate,θ is the tilt angle of the bike, φ is the position of the reaction wheel.

The kinetic energy of the bike (not includingthe reaction wheel) is:

$$T_1 = \frac{1}{2}I_1\dot{\theta}^2 + \frac{1}{2}m_1\left(h_1\dot{\theta}\right)^2 \#(2)$$

Table 1. Parameters of the real-world model

Parameter	Value
I_1	0.0014295 kg.m^2
I_2	0.000234 kg.m^2
h_1	0.085 m
h_2	0.125 m
m_1	0.8 kg
m_2	0.12 kg
K_e	0.037 Vs/rad
K_m	0.037 Nm/A
R_m	2.9 Ω
a	1
g	9.81 m/s^2

The kinetic energy of the reaction wheel is:

$$T_2 = \frac{1}{2}I_2\left(\dot{\theta} + t\dot{\varphi}\right)^2 + \frac{1}{2}m_2\left(h_2\dot{\theta}\right)^2 \#(3)$$

From (2) and (3), the total kinetic energy of the system is obtainedas:

$$T = \frac{1}{2}I_2\left(\dot{\theta} + t\dot{\varphi}\right)^2 + \frac{1}{2}m_2\left(h_2\dot{\theta}\right)^2 + \frac{1}{2}I_1\dot{\theta}^2 + \frac{1}{2}m_1\left(h_1\dot{\theta}\right)^2 \#\#(4)$$

The potential energy of the system is:

$$V = m_1h_1g\cos\theta + m_2h_2g\cos\theta \ \#(5)$$

From (1), (4), and (5), the expression of L is as follows:

$$L = \frac{1}{2}I_2\left(\dot{\theta} + t\dot{\varphi}\right)^2 + \frac{1}{2}m_2\left(h_2\dot{\theta}\right)^2 + \frac{1}{2}I_1\dot{\theta}^2 + \frac{1}{2}m_1\left(h_1\dot{\theta}\right)^2 \\ -m_1h_1g\cos\theta - m_2h_2g\cos\theta \ \#(6)$$

LetT_m is the shaft torque generated by the motor:

$$T_m = K_t\frac{U - K_e\dot{\varphi}}{R_m} \ \#(7)$$

Replacing $q$in equation (1) by θ and φ, and incorporating equation (7), yields the following expressions:

$$\ddot{\theta} = \frac{mgH}{I - I_2}sin\theta + \frac{K_tK_e\dot{\varphi}}{(I - I_2)R_m} + \frac{K_tu}{(I - I_2)R_m} \ \#(8)$$

$$\ddot{\varphi} = -\frac{mgH}{I - I_2}sin\theta - \frac{K_tK_eI\dot{\varphi}}{(I - I_2)I_2R_m} + \frac{K_tIu}{(I - I_2)I_2R_m} \ \#(9)$$

In which$K_t$is the torque constant of the DC motor, R_m is the internal resistance of the motor, K_e is the back emf constant of the motor, and U is the voltage control applied to the DC motor. The parameters m, H, and I are defined as follows:

$$m = m_1 + m_2;$$

$$H = \frac{m_1h_1 + m_2h_2}{m};$$

$$I = I_1 + I_2 + m_1h_1^2 + m_2h_2^2; \#(10)$$

The bike systemis designed to balance around the vertical direction withinthe range $-8^o \leq \theta \leq 8^o$.In this range, the linearized model of the bike can be obtained by approximating $sin\theta \approx \theta$.

Denoting$x = [x_1x_2x_3x_4]^T = \left[\theta \ \dot{\theta} \ \varphi \ \dot{\varphi}\right]^T$, from (8), (9), and (10), the linearized state space equations of the system is obtained as follows:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = \frac{mgH}{I - I_2}x_1 + \frac{K_t K_e}{(I - I_2)R_m}x_4 + \frac{K_t}{(I - I_2)R_m}u$$
$$\dot{x}_3 = x_4$$
$$\dot{x}_4 = -\frac{mgH}{I - I_2}x_1 - \frac{K_t K_e I}{(I - I_2)I_2 R_m}x_4 + \frac{K_t I}{(I - I_2)I_2 R_m}u \#(11)$$

3. LQR CONTROLLER

The cost function of the LQR controller is as follows:

$$J = \int (xQx^T + uRu^T)\, dt \#(12)$$

Q and R are the State Cost Matrix and Control Cost Matrix, respectively.

Assume that the linearized state equation describing the system dynamics is as follows:

$$\dot{x} = Ax + Bu \#(13)$$

In which,x is the state vector of the system, and u is the control signal applied tothe system.

From the linearized state equations (11) of the bike model, the matrix A and vector B can be determined as:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{mgH}{I - I_2} & 0 & 0 & \frac{K_t K_e}{(I - I_2)R_m} \\ 0 & 0 & 0 & 1 \\ -\frac{mgH}{I - I_2} & 0 & 0 & -\frac{K_t K_e I}{(I - I_2)I_2 R_m} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 89.6285 & 0 & 0 & 0.05196 \\ 0 & 0 & 0 & 1 \\ -89.6285 & 0 & 0 & -2.06935 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ \frac{K_t}{(I - I_2)R_m} \\ 0 \\ \frac{K_t I}{(I - I_2)I_2 R_m} \end{bmatrix} = \begin{bmatrix} 0 \\ 1.4044 \\ 0 \\ 55.9285 \end{bmatrix}$$

The LQR control signal is$u = -Kx$, with

$$K = R^{-1}B^T P \#(14)$$

In (14), P is calculated through the Algebraic Riccati Equation:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \#(15)$$

Denoteq_{ij}as the element at row i and column j in matrix Q, thenq_{11} is the weight state factor of tilt angle in matrix Q.

The most suitable way to choose the values of q_{11} and R is to choose them equal to each other. When choosing q_{11} with higher weight values, the model will have more overshoot and will be less stable. And with pairs of value q_{11} and R smaller, the model also has a higher overshoot. The main reason is $K = R^{-1}B^T P$, when R is smaller, the value of K with direct effects from R^{-1}, the value of K will be higher over the effects of P from Q in the Riccati equation. The demonstrations of the assumptions are shown in Figure 3 and Figure 4.

The schematic of the designed LQR controller is presented in Figure 6. K_1, K_2, K_3, K_4 are the gains of $[\theta\ \dot{\theta}\ \varphi\ \dot{\varphi}]$ respectively.

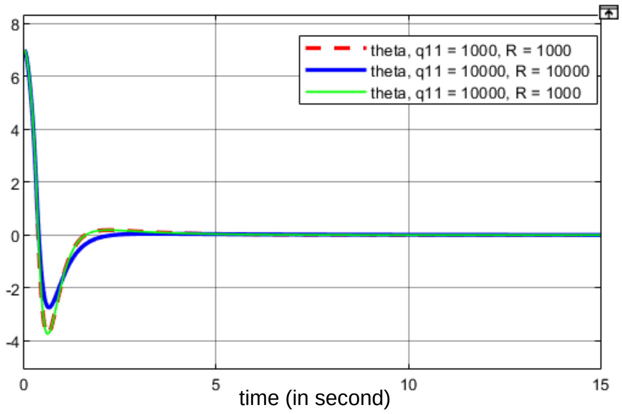

Figure3: Tilt angle(theta) response (in degrees)

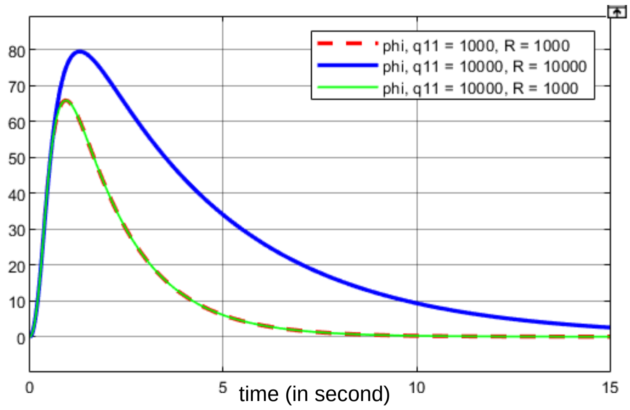

Figure4: Reaction wheel position(phi) response (in radians)

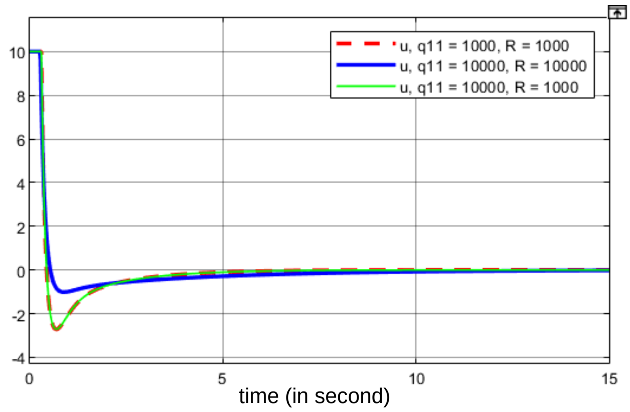

Figure5: Control signal u (in volt)

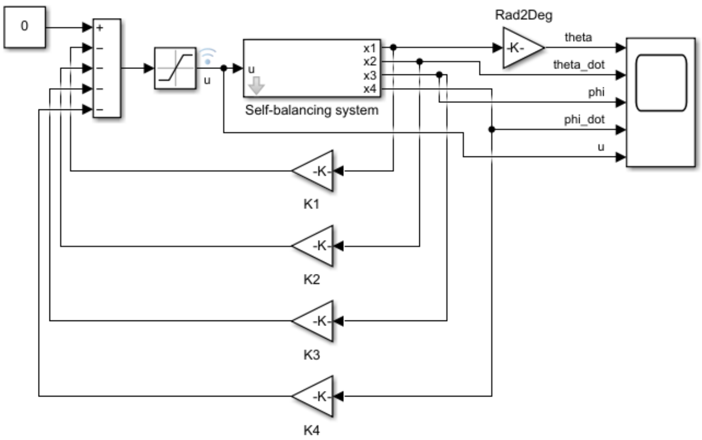

Figure 6: Schematic of the designed LQR controller

The best performance comes from q_{11} = 10000 and R = 10000.Higher values almost have no changes when compared with this case. Although the settling time is a little longer than thetwo left cases, the robustness and stability characteristics of the system are guaranteed.

$$Q = \begin{bmatrix} 10000 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad and \quad R = 10000$$

4. FUZZY–LQR CONTROLLER

The Fuzzy LQR controller has two inputs E and EC, and the output is the control signal U. The inputs E and EC are defined as follows:

$$E = -\frac{K_e}{\sqrt{K_1{}^2 + K_2{}^2 + K_3{}^2 + K_4{}^2}}(K_1\theta + K_3\varphi) \qquad (16)$$

$$EC = -\frac{K_{ec}}{\sqrt{K_1{}^2 + K_2{}^2 + K_3{}^2 + K_4{}^2}}(K_2\dot{\theta} + K_4\dot{\varphi}) \qquad (17)$$

In which K_1, K_2, K_3, K_4 are the state feedback gains of the LQR controller designed in section 3. The constants K_e, K_{ec}, and K_U are the standardization coefficients depending on practical conditions. Each input and output has 7 linguistic values, namely NB, NM, NS, ZE, PS, PM, and PM. Figure 7 and Figure 8 show membership functions defined for two inputs E and EC. Based on experience, the fuzzy rules of the Fuzzy-LQR controller are proposed as shown in Table 2. Figure 9presents the surface relationships between the controller's inputs and output. Figure 10illustrates the schematic of the designed Fuzzy-LQR controller.

Table 2. Rule base applied to Fuzzy-LQR controller

U		***E***						
		NB	NM	NS	ZE	PS	PM	PB
EC	NB	NB	NB	NB	NM	NM	NS	ZE
	NM	NB	NB	NM	NM	NS	ZE	PS
	NS	NB	NM	NM	NS	ZE	PS	PM
	ZE	NM	NM	NS	ZE	PS	PM	PM
	PS	NM	NS	ZE	PS	PM	PM	PB
	PM	NS	ZE	PS	PM	PM	PB	PB

	PB	ZE	PS	PM	PM	PB	PB	PB

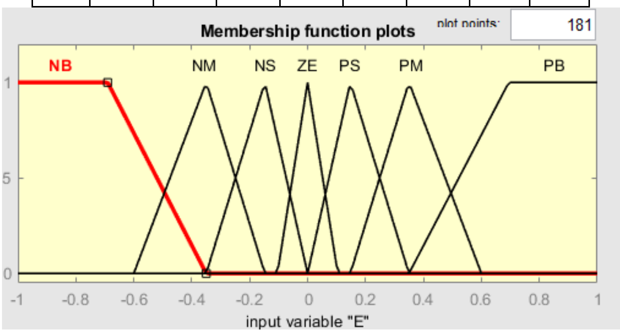

Figure7: Membership functions of input E

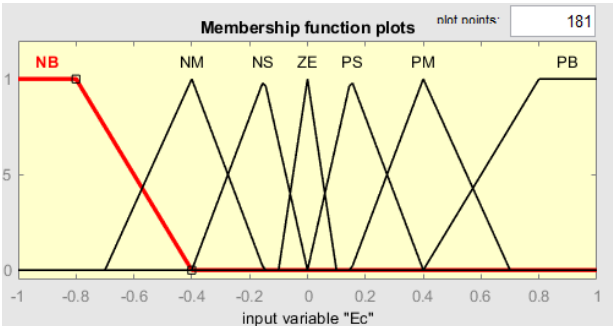

Figure8: Membership functions of input EC

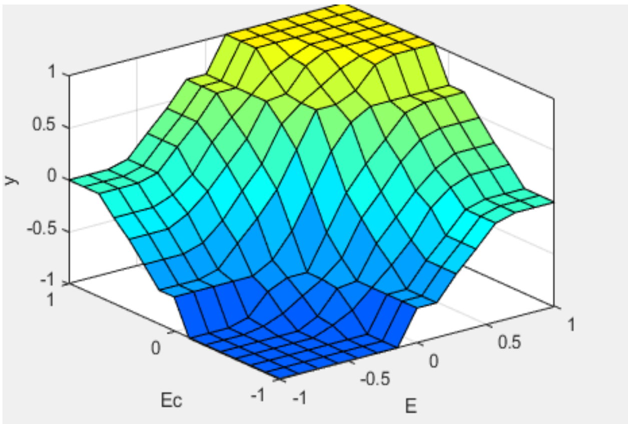

Figure9:Fuzzy surface relationship between inputs and output

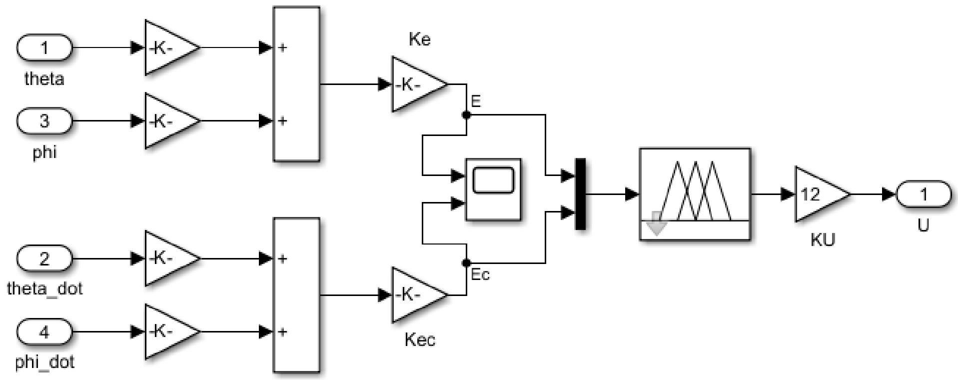

Figure10: Schematic of the Fuzzy–LQR controller

Analysis ofthe approach of the FLC controller combined with the LQR controller:

Step 1: With factors $(K_1\theta + K_3\varphi)$ and $(K_2\dot{\theta} + K_4\dot{\varphi})$ in E and EC, respectively, the control signal of the LQR controller is generated again. In other words, the control signal from the LQR controller is used to process further in the fuzzy controller.

Step 2: Multiply these two inputs by the gain of$\frac{1}{\sqrt{K_1{}^2+K_2{}^2+K_3{}^2+K_4{}^2}}$, the inputs are normalized into the range between −1 and 1.

From this procedure, the control signal of the LQR controller is processed into the fuzzy controller. The gain vector $K$of the LQR controller is computed initially and remains fixed, even when the system dynamics change, such as when the system operates outside the linearized range. As a result, while the LQR controller can perform well in the linearized range, its control signal lacks adaptability to varying model conditions.From this less-adaptive characteristic, the energy used and the ability to deal with complicated situations will not be optimized. The adaptive and well-adjusted characteristic of the fuzzy controller from the rule base will improve the control performance by overcoming the weakness of the LQR controller

5. SIMULATION AND EXPERIMENT RESULTS

5.1 Simulation setting

MATLAB is used to write code and use Simulink blocks to build the real-world mathematical model of the system for simulation purposes.The feedback linearization and finding the state feedback gain vector$K$ through the Algebraic Riccati equation of the LQR controller are conducted through a script code file in MATLAB.

In Simulink, the fixed-step solver “ode5 (Dormand-Prince)” is usedwith a sampling time of 0.001s.

For the fuzzy controller, the Fuzzy toolbox available in MATLABexports the necessary files for the Simulink simulation.

The state feedback vector$K$ of the LQR controller from the best performance of Q and R in section 3 is used, using the MATLAB command$K = lqr(A, B, Q, R)$to find K:

$$K = [-159.9917\ -16.9432\ -0.01\ -0.08196]$$

The standardization coefficients of the Fuzzy–LQR controller are set as follows: *Ke*=8, *Kec*=8, and *KU*=12.

5.2 Simulation results

Figure 11 and Figure 12 show the simulation results of the Fuzzy-LQR with the LQR controllers when setting the initial tilt angle for both methods are 7 degrees, and 8 degrees, respectively.

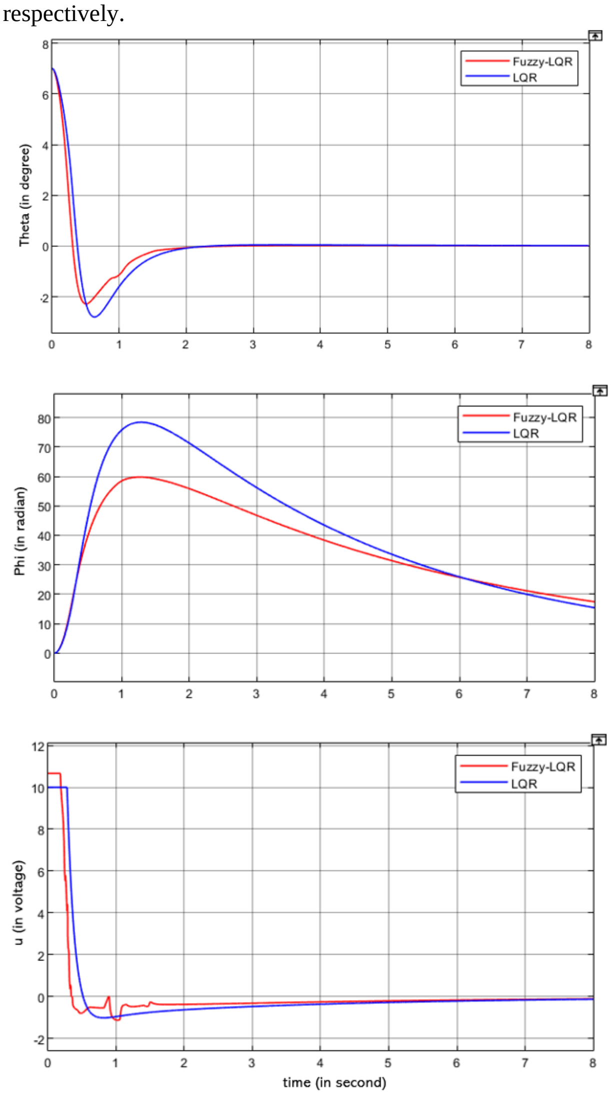

Figure 11: Bike model responses when setting the initial tilt angle equal to 7°

Table 3. Performances of the model when setting the initial tilt angle equal to 7°

Criteria	LQR	Fuzzy-LQR
Maximum peak overshoot (degrees)	0.042	0.02
Settling time (s)	7	4.7
Steady-state error	0	0

When the initial tilt angle of the bike is 7^o, the maximum peak overshoot when applying the LQR controller is 0.042 degrees and the Fuzzy-LQR controller is 0.02 degrees, the settling time when applying the LQR controller is 7s and the Fuzzy-LQR controller is 4.7s. This simulation result demonstrates that the performance of the Fuzzy-LQR controller is better than that of the LQR controller standaloneinterms of time response and stability when controlling the system.

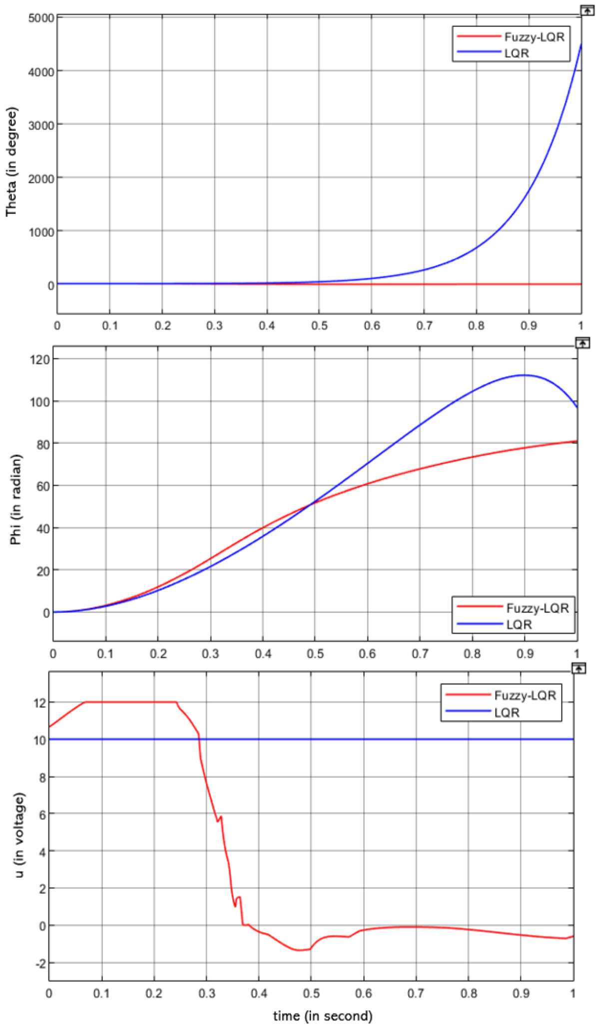

Figure 12: Bike model responses when setting the initial tilt angle equal to 8^o

Table 3. Performances of the model when setting the initial tilt angle equal to 8^o

Criteria	LQR	Fuzzy-LQR
Maximum peak overshoot (degrees)	N/A	0.035
Settling time (s)	N/A	6.5
Steady-state error	N/A	0

When the initial tilt angle of the bike is 8^o, the maximum peak overshoot when applying the Fuzzy-LQR controller is 0.035 degrees, and the settling time is 6.5s. Meanwhile, applying the LQR controller in this case, the tilt angle θgoes to a very large value, demonstrating that the LQR controller cannot control the system in this case, making the bike fall in a short time. This simulation result demonstrates the robustness and adaptation characteristics of the Fuzzy-LQR controller when the working range of tilt angle is near the out-of-linearized-range.

5.3 Experiment results

The microcontroller STM32 is used to implement the code of the LQR controller and the Fuzzy-LQR controller to control the real-world model. The DC775 motor is used as an actuator to control the reaction wheel when receiving the control signal from the microcontroller.

To get the data related to the tilt angle of the bike, an MPU 6050 sensor is used, and an encoder of 100 pulses is used to get the data related to the reaction wheel speed and position through the DC775 motor.APython file is used to visualize the data received from the microcontroller through the UART (Universal Asynchronous Receiver-Transmitter) connection between the STM32 and the computer. Figure 13illustrates the block diagram of the implemented hardware.

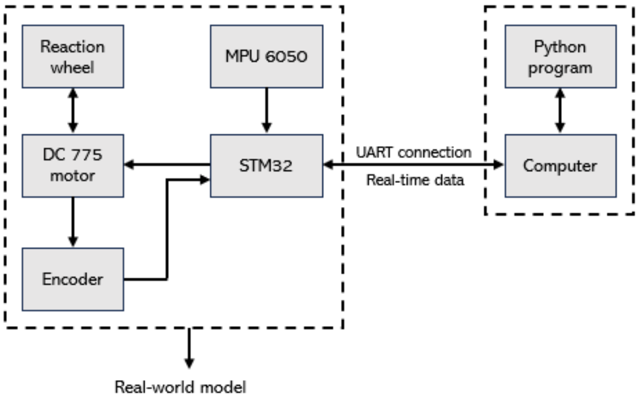

Figure 13: Block diagram of the implemented hardware

Figure 14and Figure 15 show the experiment results on therealistic model of the self-balancing bike. The Fuzzy-LQR controller has a little higher overshoot, –3^{o}compared with –2.5^o,but stabilizesfaster in comparison to the LQR controller standalone. This effect may be caused by the difference in ideal characteristics between simulation and experiment hardware,the overall performance of the Fuzzy–LQR controller is also better than the LQR controller.

The simulation and experiment results prove the advancement of the Fuzzy-LQR controller in comparison to the LQR controller, in both ideal conditions of simulation and non-ideal conditions of experiment. With the simulation results, the fast response and stability of the Fuzzy-LQR controller are demonstrated through settling time and overshoot value, however, the control signal of the Fuzzy-LQR controller displayed has a little chattering effect and not smooth like that of the LQR controller, this phenomenon can be explained by the neighboring feature, rules defined, and fast rate of change of the input in the fuzzy controller, which processed through the membership functions of input E and EC, make the output – control signal change immediately.

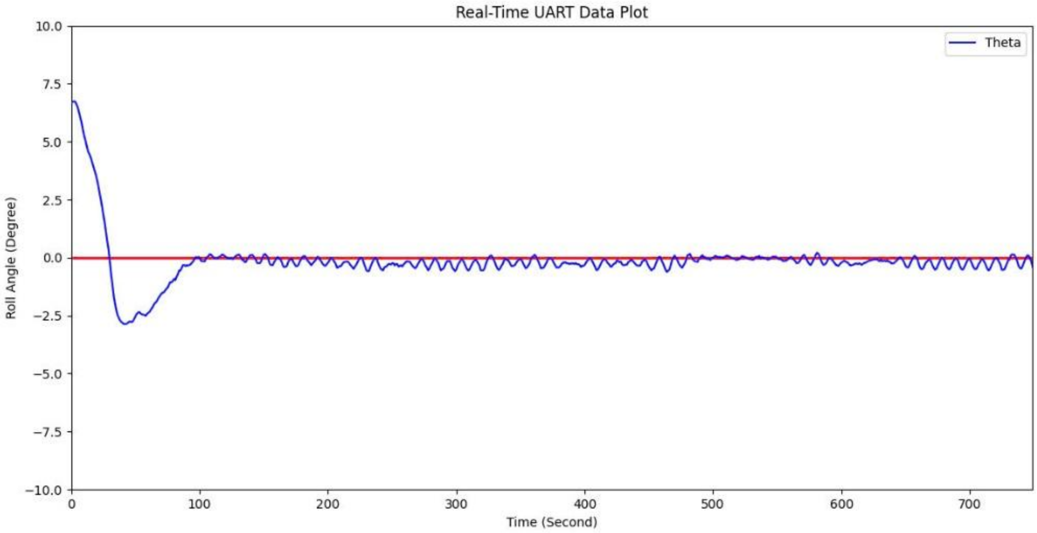

Figure 14: Experiment result of the Fuzzy–LQR controller

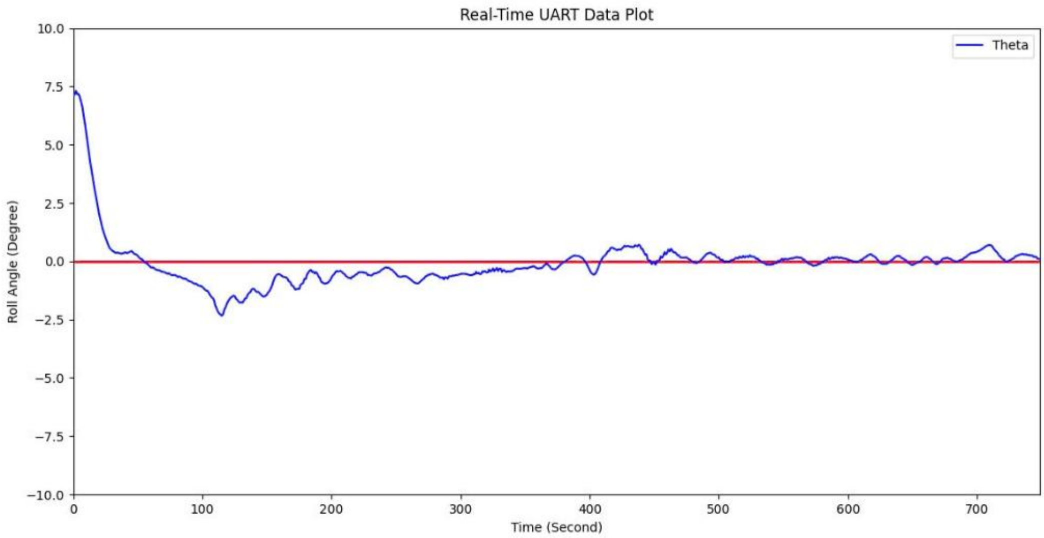

Figure 15: Experiment result of the LQR controller

The fuzzy controller is implemented on hardware devices that require a lot of computational effort, the singleton output membership function is used to save energy. Instead of using the traditional triangular shape of membership function, singleton calculates only one value at a membership function without scanning the whole range, butthe precision is similar between the two methods. This also highlights the practical application of the fuzzy controller when applied to solve real-world problems, not just in ideal conditions like a simulation platform.

6. CONCLUSION

The mathematical model of the system was first developed and used for algorithm analysis and application. The report is mainly based on the LQR control method to develop independent applications and combine them with a fuzzy controller to improve the control performance

The LQR control method can independently perform well in controlling the bike system aroundthe equilibrium point. However, fuzzy combined with LQR to improve the precision and the adaptation of the system in each general case makesthe performance of the model better in terms of overshoot and settling time. Besides, linearization is necessary for the LQR controller, the Fuzzy-LQR controller improves the performance of the model when the working range is near the out-of-linearized-range. Simulation and experiment results even show apparent differences in the performance of Fuzzy-LQR with LQR, which strengthens the advances of intelligent methods in controlling systems.

The use of the reaction wheel combined with the Fuzzy-LQR controller can control the bike well in aenough small range, which can be linearizedwith high robustness and stability.For real-world applications, considering actuators with higher torque generation is necessary, or combining methods to save energy is worth applying. For example, with a small range of the tilt angle, the reaction wheel method – a lower torque but saving more energy can be applied; with a large range of the tilt angle, the CMG method – a higher torque but using more energy can be applied. Besides, the adjustment of the fuzzy controller has a high characteristic of relativity; the performance of the model can also be improved through a better application of the parameters of the fuzzy controller.

7. REFERENCES

[1] T. J. Yeh, Hao-Tien Lu and Po-Hsuan Tseng2024. Balancing Control of a Self-driving Bike.

[2] Jaka P. Sembiring, Ahmad Amirudin, Novia Utami Putri and Danur Wijayanto, “Steering control of a self-balancing bike”, International conference on engineering and computer science (ICECS), 2022.

[3] Yongchao Yu and Mingguo Zhao, “Steering Control for Autonomously Balancing Bike at Low Speed”, 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO).

[4] Ning He, Qianhao Hu, Jiaxuan Chen and Kai Zhang, “Self-Balancing Bike Based on Control Moment Gyroscope” 2023 42nd Chinese Control Conference (CCC).

[5] Nitheesh kumar G, Navneeth S, Suraj A, and Pramod Sreedharan, “Design and Analysis of a Self-balancing Bike Model”, 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1132 012014

[6] Pom Yuan Lam, “Design and development of a self-balancing bike using control moment gyro”, Master Thesis, National University of Singapore, 2012.

[7] Kiattisin Kanjanawanishkul, “LQR and MPC Controller Design and Comparison for a Stationary Self-Balancing Bike Robot with a Reaction Wheel” Kybernetika, Vol. 51 (2015), No. 1, 173–191

[8] Patil, O., Jadhav, S., Ramakrishnan, R. 2021. Development of Reaction Wheel Controlled Self-Balancing Bike for Improving Vehicle Stability Control. In: Nalim, M.R., Vasudevan, R., Rahatekar, S. (eds) Advances in Automotive Technologies. Lecture Notes in Mechanical Engineering. Springer, Singapore. https://doi.org/10.1007/978-981-15-5947-1_15

[9] Pornchanok Vanich, “Self-Balancing Bike Using a Hybrid of Reaction Wheel and Steering Control”, Master Thesis, Asian Institute of Technology, 2022.

[10] Dag Christian Annestad, “Autonomous Bike- The First Self Balanced Ride”, Master Thesis, Norwegian University of Science and Technology, 2011.

[11] Zied Ben Hazem, Mohammad Javad Fotuhi, Zafer Bingül, “Development of a Fuzzy-LQR and Fuzzy-LQG stability control for a double link rotary inverted pendulum”, Journal of the Franklin Institute 357 (2020) 10529–10556, 2020.

[12] Jumi Bharali and Mrinal Buragohain, “Design and Performance Analysis of Fuzzy LQR, Fuzzy PID and LQR Controller for Active SuspensionSystem using 3 Degree of Freedom Quarter car model” 2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES).

[13] Y. I. Kudinov, F. F. Pashchenko, A. Y. Kelina, D. I. Vasutin, E. S. Duvanov, and A. F. Pashchenko, “Analysis of control system models with conventional LQR and fuzzy LQR controller,” 2018 Proc. 13th Int. Symp. “Intelligent Systems” (INTELS’18).