Scientific Journal of Engineering Research
futuristech
PT. Teknologi Futuristik Indonesia
Received: July 2, 2026; Revised: August 26, 2026; Accepted: August 28, 2026; Published: August 29, 2026; Current version: August 29, 2026. *e-ISSN: 3109-1725*
Digital Object Identifier 10.64539/sjer.v2i4.2026.540

Article

ANFIS-Based PD-Fuzzy Control for Pendubot Stabilization

Van-Long Mach[1], Dang-Khoa Huynh[1], Khanh-Hung Le[1], Vi-Khang Nguyen[1], Van-Hai-Dang Ma[1], Viet-Phi Le[1], Nguyen-Duy-Phuong Huynh[1], Le-Nhat Nguyen[1], Quang-Hoa Le[1], Van-Dong-Hai Nguyen[1,*]

[1] Faculty of Electronics and Electrical Engineering (FEEE), Ho Chi Minh City University of Technology and Engineering (HCM-UTE), Ho Chi Minh City (HCMC), Vietnam; e-mail: hainvd@hcmute.edu.vn (V.-D.-H. Nguyen).
* Correspondence Author

The authors received no financial support for the research, authorship, and/or publication of this article.

Abstract: The Pendubot is a nonlinear underactuated system with two rotational degrees of freedom and a single actuator, making stabilization a challenging problem for intelligent and nonlinear control methods. Reliable balancing control is important for evaluating control strategies under both simulation and practical hardware constraints. However, although fuzzy and intelligent control methods have been investigated for Pendubot systems, the practical implementation and experimental behavior of ANFIS-based PD-Fuzzy control remain insufficiently documented, particularly in comparison with a conventional PD controller under the same laboratory conditions. This study aims to develop and evaluate an ANFIS-based PD-Fuzzy controller for TOP-position stabilization of a Pendubot. The nonlinear dynamics are formulated using the Euler-Lagrange method, while controllability of the linearized model is examined at the TOP and MID equilibrium points. The proposed controller uses ANFIS-based fuzzy blocks to approximate the proportional control actions, while derivative paths remain explicitly implemented. MATLAB/Simulink simulations show that the baseline PD and PD-Fuzzy controllers produce closely matched TOP-balancing responses, with settling times of approximately 2.36 and 2.37 s for link 1, respectively. Experimental evaluation using an STM32F407-based platform demonstrates practical TOP balancing; however, the baseline PD controller provides more favorable behavior than the PD-Fuzzy realization under the reported hardware conditions. These findings indicate that ANFIS-based PD-Fuzzy control is feasible for Pendubot stabilization but does not necessarily provide performance improvement over a conventional PD controller. The study therefore highlights the importance of hardware-aware tuning, sensor quality, and broader ANFIS training data for future improvements.

Keywords: Pendubot; Underactuated system; PD Fuzzy control; ANFIS; STM32F407; Balancing control.

1. Introduction

The Pendubot is a representative underactuated robotic system consisting of two links and one actuated joint [1], [2]. The first link is driven by a motor, whereas the second link rotates freely and is controlled indirectly through the dynamic coupling between the two links [3], [4]. The nonlinear dynamics, unstable equilibrium configurations, and limited actuation make the Pendubot a useful benchmark for investigating balancing control strategies and their implementation on laboratory platforms [5], [6].

Several control approaches have been investigated for the Pendubot and related underactuated control problems. Auto-tuning fuzzy PID control has been applied to a Pendubot system [5]. Fuzzy-based linear quadratic regulator (LQR) approaches have also been investigated for Pendubot balancing [7], [8]. Other studies have considered fuzzy sliding-mode and adaptive sliding-mode strategies [9], [10]. Sliding-mode control for underactuated manipulators and hybrid control strategies for swing-up and balancing have also been reported [11]–[13]. These studies demonstrate the use of conventional, fuzzy, optimal, and nonlinear control strategies to address the challenging dynamics associated with underactuated systems.

A practical dilemma arises when a fuzzy-enhanced controller is implemented on a physical Pendubot. A conventional PD controller has a relatively simple structure and can provide effective balancing [14], whereas incorporating fuzzy blocks into the proportional control paths introduces an additional training and implementation process [15]. Therefore, the use of a fuzzy-enhanced structure should not be assumed to provide better practical performance solely because of its more sophisticated control structure [16], [17]. The difference between simulation and experimental behavior can become significant because physical implementations are affected by factors such as sensor noise, actuator limitations, mechanical tolerances, friction, and other unmodeled system characteristics [18]-[20]. These considerations make a direct comparison between the baseline PD controller and its fuzzy-enhanced realization relevant to the practical evaluation of the implemented control approach.

This study addresses this issue by implementing and evaluating an ANFIS-based PD-Fuzzy controller for Pendubot stabilization. The nonlinear dynamics are formulated using the Euler-Lagrange method, and the controllability of the linearized model is evaluated around the TOP and MID equilibrium configurations. In the implemented controller, ANFIS-based fuzzy blocks approximate the proportional control actions, while the derivative paths remain explicitly implemented. A conventional PD controller is used as the baseline for comparison. The control strategies are evaluated through MATLAB/Simulink simulations and experimental tests on an STM32F407-based Pendubot platform.

The main contributions of this study are: *(i)* formulation of the nonlinear Pendubot model using the reported mechanical parameters; *(ii)* controllability assessment of the linearized model at the TOP and MID equilibrium configurations; *(iii)* implementation and evaluation of a conventional PD controller and an ANFIS-based PD-Fuzzy realization; *(iv)* simulation evaluation of TOP balancing and swing-up behavior; and *(v)* experimental evaluation of TOP balancing using an STM32F407-based embedded platform. The study does not assume that the fuzzy-enhanced controller will outperform the baseline controller; instead, the comparison is used to assess its feasibility and practical behavior under the reported hardware conditions. Experimental claims are limited to TOP balancing, while MID controllability is evaluated analytically and swing-up performance is evaluated through simulation.

The remainder of this paper is organized as follows. Section 2 presents the Pendubot model and equilibrium analysis. Section 3 describes the baseline PD controller, ANFIS-based PD-Fuzzy realization, genetic algorithm tuning, and swing-up module. Section 4 presents the MATLAB/Simulink simulation results. Section 5 describes the experimental platform and real-time implementation. Section 6 presents and discusses the experimental results, reproducibility considerations, limitations, and future development. Finally, Section 7 concludes the paper.

2. Pendubot Model and Equilibrium Analysis

2.1. Mechanical Configuration

The mechanical configuration is shown in Figure 1. The generalized coordinate vector is $q = [q_1 \;\; q_2]^T$, where q_1 is the angle of link 1 with respect to the horizontal axis and q_2 is the relative angle of link 2 with respect to link 1. Since only the first joint is actuated, the generalized torque is $\tau = [\tau_1 \;\; 0]^T$.

The center-of-mass coordinates are obtained from the geometry in Figure 1. The active-link center of mass is located at $(l_{c1}\cos q_1\,, l_{c1}\sin q_1)$, whereas the passive-link center of mass is obtained by adding the link-1 endpoint position and the relative motion of link 2. These expressions are used to derive the kinetic and potential energies of the system.

$$x_1 = l_{c1}\cos(q_1),\quad y_1 = l_{c1}\sin(q_1) \tag{1}$$

$$x_2 = l_1\cos(q_1) + l_{c2}\cos(q_1 + q_2) \tag{2}$$

$$y_2 = l_1\sin(q_1) + l_{c2}\sin(q_1 + q_2) \tag{3}$$

The Euler-Lagrange equation is then used to derive the nonlinear model. Let L denote the Lagrangian, K the kinetic energy, and V the potential energy. The governing relationship and the compact dynamic model are written as follows.

$$L = K - V \tag{4}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = \tau_i \tag{5}$$

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \tag{6}$$

Using the lumped parameters β_1–β_5 defined above, the inertia matrix, Coriolis and centrifugal matrix, and gravity vector are expressed as follows:

$$D(q) = [[\beta^1 + \beta^2 + 2\beta^3 cos(q^2), \beta^2 + \beta^3 cos(q^2)], [\beta_2 + \beta_3\, cos(q_2), \beta_2]] \tag{7}$$

$$C(q,\dot{q}) = [[-\beta_3\dot{q}_2 sin(q_2), -\beta_3(\dot{q}_1 + \dot{q}_2)sin(q_2)], [\beta_3\dot{q}_1 sin(q_2), 0]] \tag{8}$$

$$G(q) = [\beta_4 g\, cos(q_1) + \beta_5 g\, cos(q_1 + q_2), \beta_5 g\, cos(q_1 + q_2)]^{\mathrm{T}} \tag{9}$$

The lumped parameters follow $\beta_1 = m_1 l_{c1}^2 + m_2 l_1^2 + I_1$, $\beta_2 = m_2 l_{c2}^2 + I_2$, $\beta_3 = m_2 l_1 l_{c2}$, $\beta_4 = m_1 l_{c1} + m_2 l_1$ and $\beta_5 = m_2 l_{c2}$. The numerical values are summarized in Table 1.

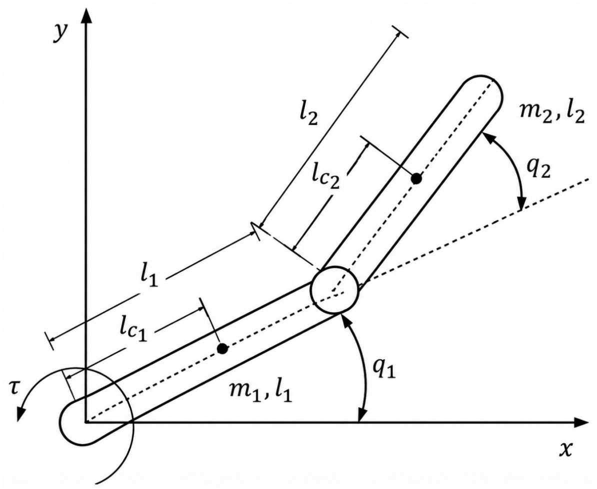


Figure 1. Coordinate definition and mechanical parameters of the two-link Pendubot [21].

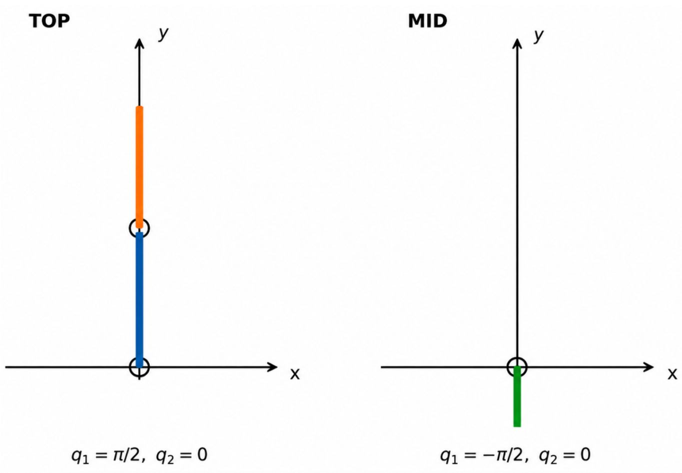


Figure 2. TOP and MID equilibrium configurations considered in the local controllability study.

Table 1. Mechanical and derived parameters of the Pendubot.

Parameter	Value	Unit
l_1	0.15	m
l_2	0.25	m
I_1	0.0189	kg·m²
I_2	0.000409	kg·m²
l_{c1}	0.055	m
l_{c2}	0.149	m
m_1	0.25	kg
m_2	0.07	kg
g	9.81	m/s²
β_1	0.0212	-
β_2	0.0020	-
β_3	0.0016	-
β_4	0.0243	-
β_5	0.0104	-

The first-link and second-link center-of-mass positions are written from the geometric configuration, differentiated to obtain the velocities, and substituted into the total kinetic energy. The resulting compact expressions are:

$$K = \frac{1}{2}\beta_1\dot{q}_1^2 + \frac{1}{2}\beta_2(\dot{q}_1 + \dot{q}_2)^2 + \beta_3\cos(q_2)\dot{q}_1(\dot{q}_1 + \dot{q}_2) \tag{10}$$

$$V = \beta_4 g\sin(q_1) + \beta_5 g\sin(q_1 + q_2) \tag{11}$$

These energy expressions lead directly to the matrices in (4)-(6). The representation is convenient for simulation because the coupling between the active and passive joints remains explicit. It also clarifies why balancing is difficult: the passive-joint response is generated through the inertial, Coriolis, and gravitational coupling terms rather than by a dedicated actuator.

2.2. State Representation and Equilibria

The first-order state vector is $x = [q_1 \quad \dot{q}_1 \quad q_2 \quad \dot{q}_2]^T$. The nonlinear accelerations are obtained from $\ddot{q} = D(q)^{-1}[\tau - C(q,\dot{q})\dot{q} - G(q)]$. Linearization around an equilibrium gives the local state model $\dot{x} = Ax + Bu$, where A and B are the corresponding Jacobian matrices.

$$x = [q_1 \quad \dot{q}_1 \quad q_2 \quad \dot{q}_2]^T, \qquad \dot{x} = f(x,u) \tag{12}$$

$$A = \left.\frac{\partial f}{\partial x}\right|_{x_e, u_e}, \qquad B = \left.\frac{\partial f}{\partial u}\right|_{x_e, u_e} \tag{13}$$

$$C = [B \quad AB \quad A^2B \quad A^3B], \qquad rank(C) = 4 \tag{14}$$

Three representative equilibrium configurations are considered. TOP corresponds to $q_1 = \pi/2$ and $q_2 = 0$. MID corresponds to $q_1 = -\pi/2$ and $q_2 = 0$. A third OX-Top configuration is also examined. The TOP and MID equilibrium configurations are illustrated in Figure 2. The controllability rank is four at TOP and MID, equal to the number of states. The OX-Top configuration has rank two and therefore does not provide full local controllability. The equilibrium configurations and their corresponding controllability ranks are summarized in Table 2.

3. Controller Design

3.1. Baseline PID and Implemented PD Structure

The standard PID control law is given by [22], [23]:

$$u(t) = K_p e(t) + K_i\int e(t)\,dt + K_d\frac{de(t)}{dt} \tag{15}$$

The proportional term reacts directly to the present error, the integral term is intended to remove persistent offset, and the derivative term damps rapid variations. Among the considered control structures, a PD realization is selected for the implemented balancing model because of its practical suitability for the available plant and hardware.

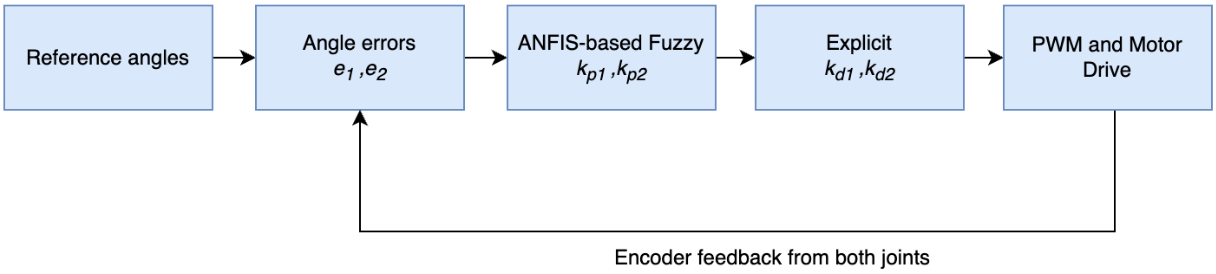


Figure 3. Implemented ANFIS-based PD-Fuzzy structure: fuzzy proportional paths and explicit derivative paths.

Table 2. Equilibrium configurations and local controllability.

Configuration	q1	q2	Rank	Interpretation
TOP	pi/2	0	4	Locally controllable
MID	-pi/2	0	4	Locally controllable
OX-Top	Reported in thesis	Reported in thesis	2	Not fully controllable

Table 3. Controller parameter reported in this study.

Stage	K_{p1}	K_{d1}	K_{p2}	K_{d2}
Simulation PD	91.5	9	71.3	5.5
Experimental baseline	9.81	0.35	145.125	7

Table 4. Genetic algorithm settings extracted from the appendix.

Setting	Value
Maximum generations	20000
Population size	20
Crossover probability	0.6
Mutation probability	0.4
Encoding	parameter vector

The implemented balancing model uses two PD channels associated with the two joint errors. After tuning, the simulation gains are $K_{p1} = 91.5$, $K_{d1} = 9$, $K_{p2} = 71.3$, and $K_{d2} = 5.5$. The real-time platform requires empirical retuning and uses $K_{p1} = 9.81$, $K_{d1} = 0.35$, $K_{p2} = 145.125$, and $K_{d2} = 7$. These two parameter sets are kept separate in the paper because they belong to different implementation stages. The controller parameters used in the simulation and experimental stages are summarized in Table 3.

$$t = \left(K_{p1}e_1 + K_{d1}\dot{e}_1\right) - \left(K_{p2}e_2 + K_{d2}\dot{e}_2\right) \tag{16}$$

3.2. ANFIS-Based PD-Fuzzy Realization

The fuzzy-enhanced design retains the derivative paths and replaces the two proportional gain paths with trained fuzzy blocks. The resulting ANFIS-based PD-Fuzzy control structure is illustrated in Figure 3. Input-output data are generated from the baseline proportional channels and used to train fuzzy approximations of K_{p1} and K_{p2} using the MATLAB ANFIS editor.

Input-output data are generated from the baseline proportional channels and uses the MATLAB ANFIS editor to construct fuzzy approximations for K_{p1} and K_{p2}. Consequently, the hybrid controller is not a fully independent fuzzy controller; it is a data-driven realization of the proportional paths within the broader PD-Fuzzy framework.

The implemented ANFIS-based fuzzy block uses three triangular membership functions with constant output terms. The fuzzy models are trained using input-output data generated from the corresponding proportional control paths and subsequently incorporated into the PD-Fuzzy controller.

The ANFIS workflow is carried out in four steps. First, a random excitation is generated and passed through the selected proportional gain. Second, the paired input and output signals are stored as a training matrix. Third, the MATLAB ANFIS editor creates a Sugeno-type fuzzy approximation. Finally, the trained fuzzy block replaces the corresponding proportional gain in the closed-loop model. The same procedure is repeated for the second proportional path.

The source introduces ANFIS as an adaptive neuro-fuzzy inference system in which the membership-function parameters are adjusted from training data. The generated fuzzy model uses a Sugeno form, constant or linear consequent functions, and a shared rule structure. In the reported workflow, the fuzzy block learns the baseline proportional relation rather than constructing a large manually designed rule base.

3.3. Genetic Algorithm Tuning

The simulation gains are obtained using a genetic algorithm. The optimization process involves population initialization, fitness evaluation, selection, crossover, mutation, and termination. The overall genetic algorithm optimization workflow is illustrated in Figure 4. The reported code uses a maximum of 20000 generations, a population size of 20, crossover probability 0.6, and mutation probability 0.4. These values are summarized in Table 4.

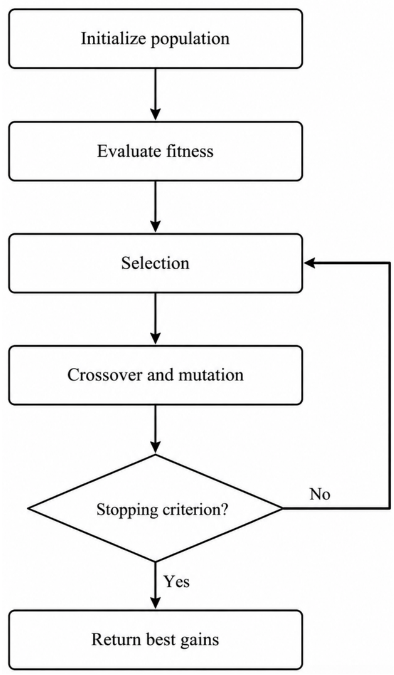


Figure 4. Genetic algorithm workflow used for controller-parameter search.

Table 5. Swing-up simulation summary.

Quantity	Reported result
Link 1 settling time	approximately 2.5 s
Link 2 settling time	approximately 2.4 s
Requested voltage	up to 24 V
Experimental validation	not completed in the source

3.4. Swing-Up Simulation Module

This study also derives a swing-up module by reorganizing the second-link dynamics and applying an outer PD tracking loop for link 1. This part is validated in simulation only. The reduced dynamics are expressed with a virtual input v and an outer PD law.

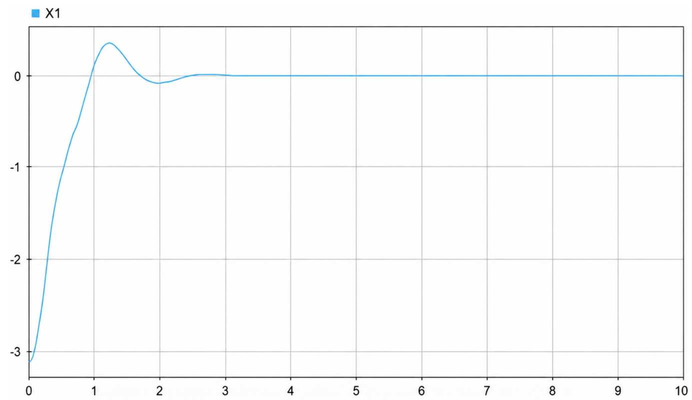
$$v = K_D(\dot{q}_1 d - \dot{q}_1) + K_P(q_1 d - q_1) \quad (17)$$

$$u = swing - up\ term + balancing\ correction \quad (18)$$

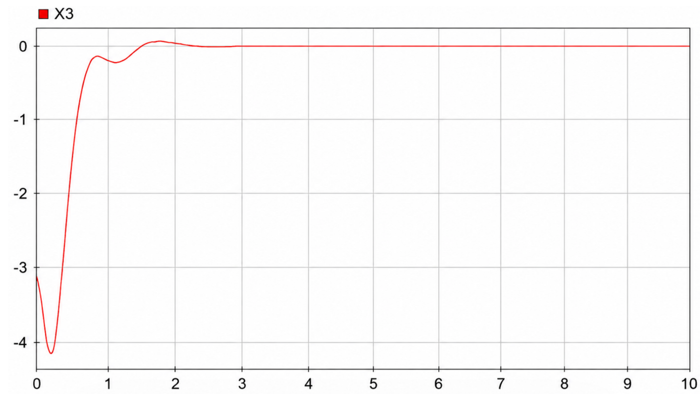
The simulated response reaches the balancing region after approximately 2.5 s for link 1 and 2.4 s for link 2, while the requested drive voltage reaches 24 V. The simulated swing-up and balancing responses are shown in Figure 5. The hardware prototype did not reproduce this swing-up task because the available power stage and encoder signals were insufficient for reliable operation. The main reported swing-up simulation results are summarized in Table 5.

4. MATLAB and Simulink Results

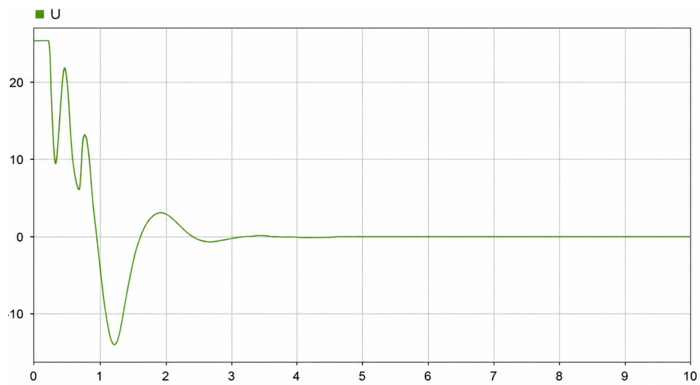
The TOP-position simulations compare the baseline PD controller and the PD-Fuzzy realization. For the baseline PD case, link 1 settles at approximately 1.571 rad after 2.36 s and remains within 1.566-1.598 rad. Link 2 settles around 0 rad after 1.3 s and remains within -0.045 to 0.007 rad. The corresponding TOP-balancing responses are shown in Figure 6.



(a)



(b)

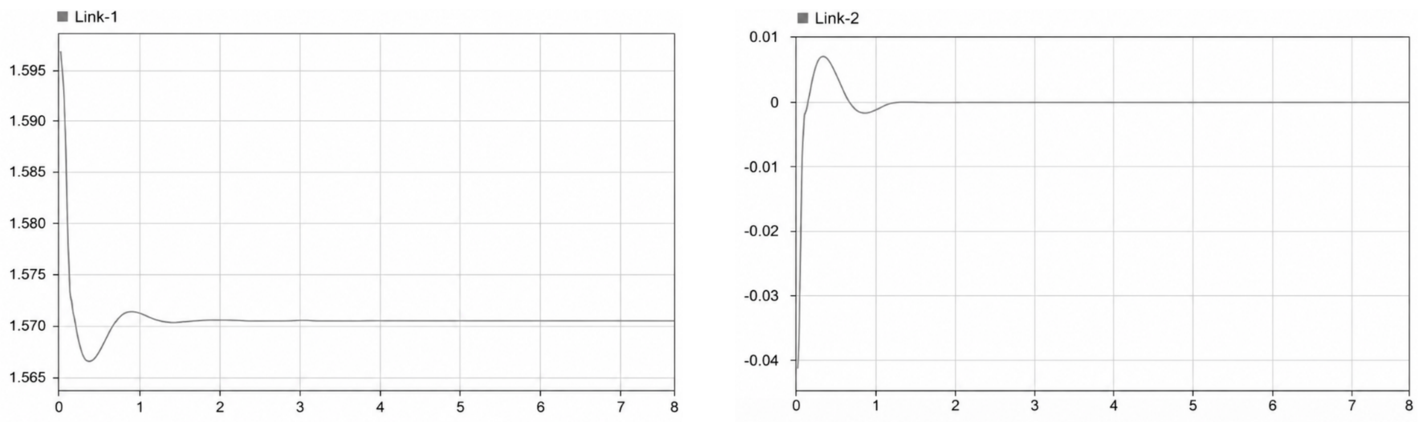


(c)

Figure 5. Simulated swing-up and balancing responses: (a) link 1 angle, (b) link 2 angle, and (c) requested drive voltage.

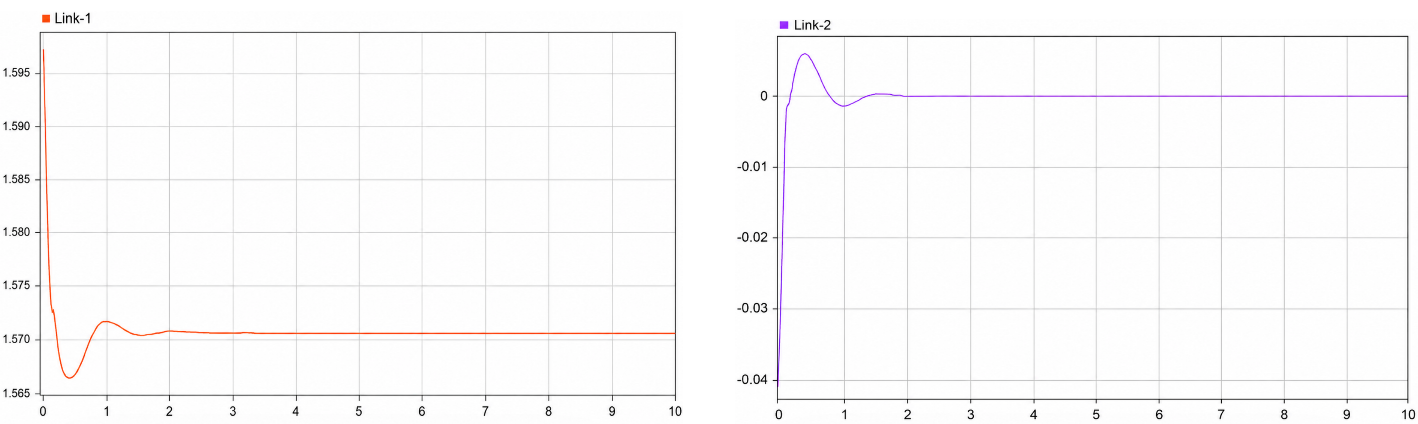
For the PD-Fuzzy case, link 1 settles at approximately 1.571 rad after 2.37 s with the same reported response interval of 1.566-1.598 rad. Link 2 again settles after approximately 1.3 s with a reported range of -0.045 to 0.007 rad. The response shapes in Figure 7 are therefore closely matched to the baseline result. A summary of the reported TOP-position simulation results for both controllers is provided in Table 6.

The reported link-1 range implies an absolute upper deviation of approximately 0.027 rad from the final value. This value is a direct post-processing result from the interval given in the source.



(a) (b)

Figure 6. Simulated TOP-balancing responses of the baseline PD controller: (a) link 1 and (b) link 2.



(a) (b)

Figure 7. Simulated TOP-balancing responses of the PD-Fuzzy realization: (a) link 1 and (b) link 2.

Table 6. Top-position simulation summary.

Controller	Signal	Target	Settling time	Reported range
PD	q1	1.571 rad	2.36 s	1.566 to 1.598 rad
PD	q2	0 rad	1.30 s	-0.045 to 0.007 rad
PD-Fuzzy	q1	1.571 rad	2.37 s	1.566 to 1.598 rad
PD-Fuzzy	q2	0 rad	1.30 s	-0.045 to 0.007 rad

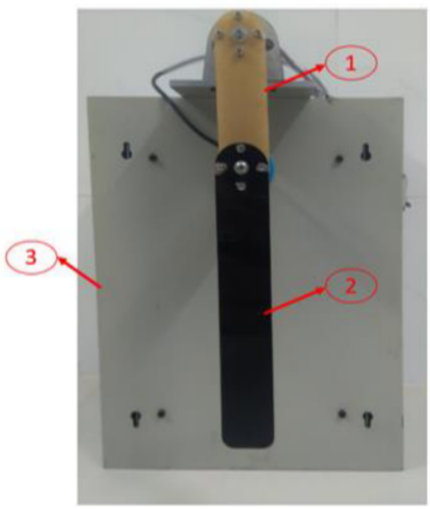


Figure 8. Completed laboratory Pendubot prototype. (1) Link 1, (2) Link 2, (3) Cabinet.

5. Experimental Platform and Real-time Implementation

5.1. Hardware configuration

The laboratory platform consists of a two-link Pendubot, an STM32F407 Discovery board, a CP2102 USB-UART interface, encoder feedback for both joints, an L298 H-bridge, and a NISCA NF5475 DC motor. Link 1 is 0.15 m long and link 2 is 0.25 m long. The links are fabricated from mica sheets, and the electronics are mounted inside a cabinet. The completed prototype is shown in Figure 8.

The mechanical design uses a 10 mm mica sheet for link 1 and a 5 mm mica sheet for link 2. The first link is 150 mm long and 40 mm wide. The second link is 250 mm long and 40 mm wide. The joints are assembled with encoder couplings, and the enclosure provides a compact mounting location for the controller, power supply, and motor-drive electronics. The reported mechanical construction

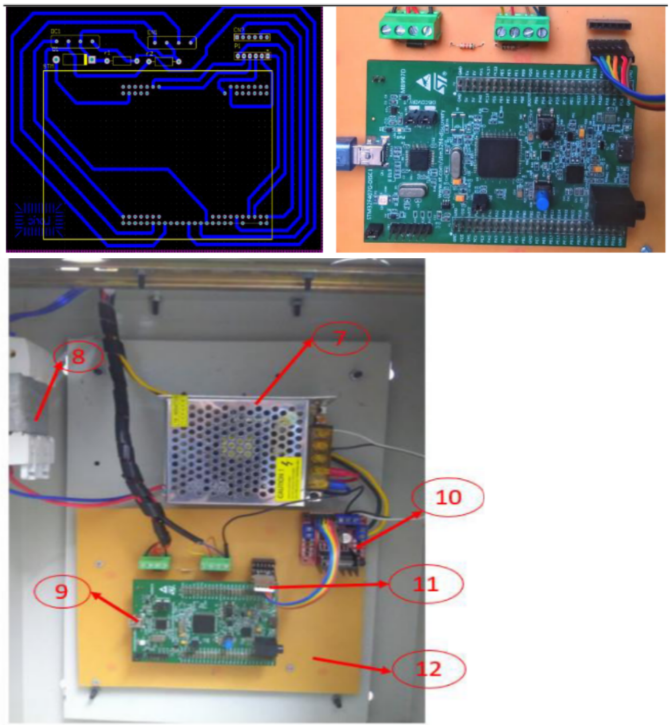


Figure 9. Hardware integration details: (a) PCB and STM32F407 board and (b) electronics enclosure interior. (8) NISCA NF5475 brushed DC motor, (9) STM32F407, (10) L298N H-bridge motor driver, (11) CP2102 USB-UART, (12) PCB.

Table 7. Reported mechanical construction details.

Part	Material or dimension
Link 1	mica, 150 mm × 40 mm × 10 mm
Link 2	mica, 250 mm × 40 mm × 5 mm
Cabinet	approximately 43 cm × 35 cm × 15 cm
Joint sensing	encoder coupling at both joints

Table 8. Main experimental component.

Component	Reported specifications or role
Controller board	STM32F407 Discovery, ARM Cortex-M4F
Motor	NISCA NF5475 brushed DC motor, 12-24 V
Motor power	32 W
Encoder resolution	200 ppr; 800-count processing used in software
Motor driver	L298 H-bridge
Computer interface	CP2102 USB-UART
Base sampling time	0.01 s

Table 9. Motor specifications reported in this study.

Motor parameter	Value
Supply voltage	12-24 V DC
Rated power	32 W
Load current	1.49 A
Starting current	8.264 A
Starting torque	464 mN·m
Load torque	71.07 mN·m
Speed	4500 rpm

Table 10. H-bridge comparison from this study.

Item	L298 module	IR2184 H-bridge
Input voltage	5-30 V DC	12-36 V DC
Maximum power	25 W per bridge	150 W
Control method	PWM	PWM
Maximum current	2 A per bridge	30 A

details are summarized in Table 7. The main components and their reported specifications or roles are summarized in Table 8.

The STM32F407 platform is selected because it provides an ARM Cortex-M4F processor, 1 MB Flash memory, 192 KB RAM, and integrated debugging support. This study uses the Waijung blockset to deploy the MATLAB/Simulink model to the target without an additional code-generation workflow. This arrangement makes the platform suitable for rapid laboratory iteration.

The CP2102 USB-UART interface provides the communication path between the embedded controller and the computer. The documented pins include TXD, RXD, GND, 5 V, DTR, and 3.3 V. In the implemented setup, the UART link is used for transmitting measured values and supporting the experimental monitoring process. The integration of the controller, motor driver, communication interface, and PCB within the experimental enclosure is illustrated in Figure 9.

The NISCA NF5475 motor is reported with a 12-24 V supply range, 32 W rated power, 1.49 A load current, 8.264 A starting current, 464 mN·m starting torque, 71.07 mN·m load torque, and 4500 rpm nominal speed. The integrated encoder uses A and B pulse channels and a 5 V supply. The reported motor specifications are summarized in Table 9.

This study compares the L298 module with an IR2184-based H-bridge. The higher-power IR2184 option is rated for a larger current, whereas the L298 module is selected for the developed prototype because it is simple, inexpensive, and adequate for the intended laboratory tests. The reported specifications of the two H-bridge options are compared in Table 10.

5.2. Embedded Control Chain

The embedded chain reads pulses from the two encoders, converts them to angular measurements, computes the control command, applies a rotation-direction and safety-limit block, and generates PWM for the motor driver. The computer interface is used for communication and monitoring. The thesis reports a protection range of -90 to 90 degrees for joint 1 and -30 to 30 degrees for joint 2. The overall real-time control chain implemented on the STM32F407 platform is shown in Figure 10.

The Waijung blockset is used to connect the MATLAB/Simulink model to the STM32F4 target [24], [25]. Timer-based blocks implement encoder reading and PWM generation. The angle-conversion logic accounts for

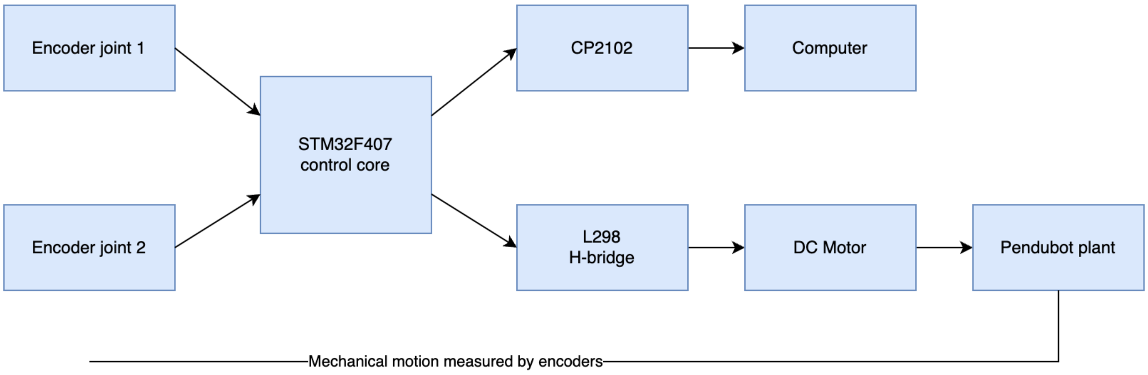


Figure 10. Real-time control chain derived from the implemented STM32F407 platform.

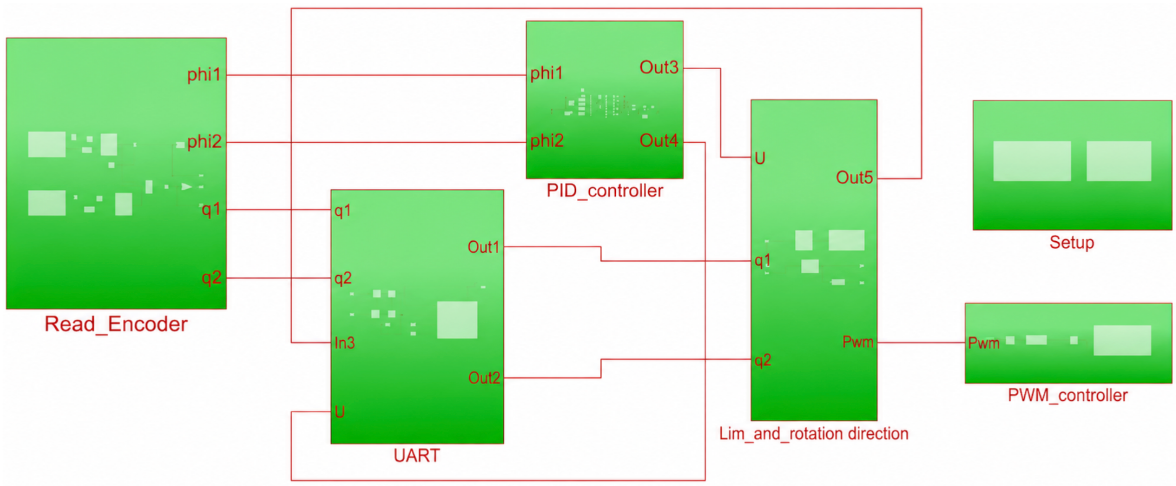


Figure 11. High-level Simulink organization of the embedded PID-Fuzzy experiment.

Table 11. Real-time software module

Module	Function
Read_Encoder	Read pulse counts and convert them to joint angles
UART	Transmit measured values through CP2102
PID_controller	Compute the baseline or fuzzy-enhanced control command
Lim_and_rotation_direction	Apply safety limits and set motor direction
Setup	Configure STM32F4 and UART parameters

pulse counting and scaling. The main real-time software modules and their functions are summarized in Table 11. The source contains some inconsistent encoder descriptions; therefore, this article reports the sensor specification as 200 ppr and separately notes the 800-count processing used in software.

The real-time Simulink implementation is organized as a closed loop with encoder reading, UART communication, controller computation, safety limiting, and PWM actuation. Figure 11 reproduces the high-level block organization of the fuzzy-enhanced experiment.

5.3. Encoder Processing and Safety Logic

The embedded model reads the two encoder channels using timer blocks and converts pulse counts into angles. The source notes that one mechanical revolution is processed as 800 counts after the selected pulse scaling. For link 1, the conversion is referenced to the vertical position. For link 2, the relative angle is obtained from the difference between the two measured joint motions.

$$q_1 = 360° \left(\frac{count_1}{800}\right) - 90° \tag{19}$$

$$q_2 = 360° \left(\frac{count_2}{800}\right) - q_1 \tag{20}$$

Before the PWM command is issued, a direction block selects the motor polarity from the sign of the control signal. A limit block disables actuation when the measured angles leave the permitted balancing region. This protection is useful during laboratory testing because the underactuated plant can rotate rapidly when the upright condition is lost.

5.4. Encoder Interface and Real-Time Execution Notes

Two encoder interfaces are used in the experimental platform. The encoder integrated with the NISCA NF5475

Table 12. Encoder and interface details.

Item	Reported detail
Integrated motor encoder	5 VDC; A/B pulse channels; 200 ppr; 20 kHz response
Auxiliary joint encoder	5 VDC; A/B pulse channels; 200 ppr; 6 mm shaft; 45 mm outer diameter
CP2102 interface	TXD, RXD, GND, 5 V, DTR, and 3.3 V pins
PCB integration	STM32F4, CP2102, two encoder inputs, and L298 output
Software angular scaling	800 processed counts per revolution in the implemented model

Table 13. Experimental top-balancing comparison.

Control label	Record length	Joint 1 observation	Joint 2 observation	Overall assessment
PD baseline	800 samples ≈ 8 s	about -15° to 20°	about -1° to 2.45°	More favorable
PD-Fuzzy	700 samples ≈ 7 s	offset around -11.5°	about -4° to 4°	Operational, but less favorable

Table 14. Source-based reproducibility checklist.

Category	Reported setting
Plant geometry	l1 = 0.15 m; l2 = 0.25 m
Simulation gains	K_{p1} = 91.5; K_{d1} = 9; K_{p2} = 71.3; K_{d2} = 5.5
Experimental gains	K_{p1} = 9.81; K_{d1} = 0.35; K_{p2} = 145.125; K_{d2} = 7
Controller target	TOP balancing; MID analyzed locally
Sampling and PWM	base sample time 0.01 s; PWM period 0.001 s
Angle limits	joint 1: -90° to 90°; joint 2: -30° to 30°
Encoder processing	200 ppr sensor; 800-count software processing
Swing-up status	simulation only; requested voltage reaches 24 V

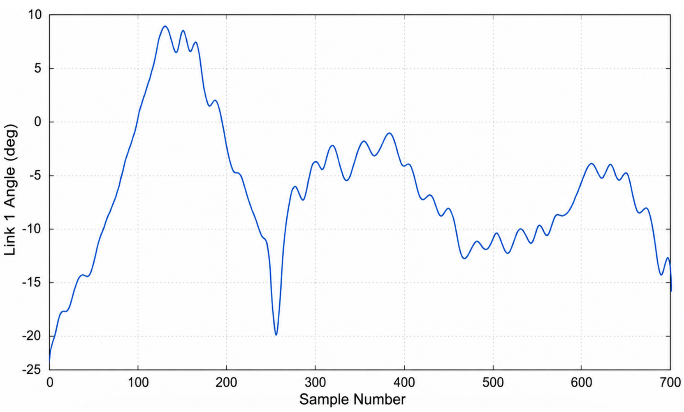


Figure 12. Experimental TOP-balancing responses of the baseline controller: joint 1 and joint 2 angles versus sample index.

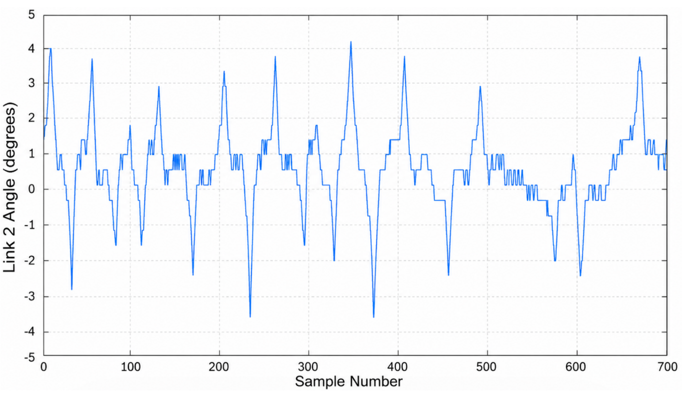


Figure 13. Experimental TOP-balancing responses of the PD-Fuzzy realization: joint 1 and joint 2 angles versus sample index.

motor is supplied at 5 VDC and provides two pulse channels, A and B. Its reported resolution is 200 ppr and its documented response frequency is 20 kHz. The auxiliary encoder used at the second joint is also supplied at 5 VDC, provides A and B channels, and is reported with a 200 ppr resolution. The source additionally lists an approximately 100 g mass, a 6 mm shaft diameter, and a 45 mm outer diameter for this auxiliary sensing unit. The reported encoder and interface specifications are shown in Table 12.

The sensing principle described in this study is optical pulse generation. A coded rotating disc interrupts or transmits light from an LED source toward receiving elements. The resulting square-wave signals allow the processor to estimate position and rotational speed. In the embedded implementation, the Read_Encoder subsystem converts the measured counts to angular values before the feedback signals are passed to the controller.

This study also presents a dedicated interface board designed to connect the STM32F4 controller, the two encoder signals, the L298 driver, and the CP2102 communication module. This hardware arrangement supports a closed-loop execution cycle consisting of pulse acquisition, angular conversion, controller evaluation, direction and safety checks, and PWM generation. The cycle is repeated continuously during the balancing test.

The timer settings distinguish the control update from the PWM actuation. The source reports a 0.01 s base sampling time for the model and a 0.001 s PWM period. These values are retained exactly as reported so that a future replication can rebuild the timing configuration without assuming unreported implementation details.

6. Experimental Results and Discussion

6.1. Experimental Procedure

The experimental tests begin with the Pendubot manually positioned near the upright TOP equilibrium. The embedded controller then reads the two encoder signals, calculates the control command, checks the permitted angle region, and applies PWM to the motor driver. The logged angle traces are reported against the sample index.

Two balancing tests are compared. The first uses the baseline controller labelled PD in the source. The second uses the fuzzy-enhanced controller labelled PD-Fuzzy. As explained earlier, both implemented structures are interpreted conservatively as PD-like realizations because the visible blocks retain proportional and derivative paths without a clearly deployed integral channel.

6.2. Baseline Balancing Test

In the baseline experiment labelled PD in the source, 800 samples are recorded with a sampling time of 0.01 s, corresponding to approximately 8 s. Joint 1 remains oscillatory, with an observed range of approximately -15 to 20 degrees. Joint 2 is more tightly regulated, with a reported range of approximately -1 to 2.45 degrees. The recorded traces are shown in Figure 12.

6.3. PD-Fuzzy Balancing Test

The fuzzy-enhanced experiment records approximately 700 samples, corresponding to about 7 s. Joint 1 is balanced with a negative offset around -11.5 degrees, while joint 2 fluctuates within approximately -4 to 4 degrees. The traces are shown in Figure 13. The source therefore demonstrates that the PD-Fuzzy implementation is operational on the real system, but it does not demonstrate a performance improvement over the baseline controller. The reported observations from the two experimental balancing tests are compared in Table 13.

6.4. Discussion

The simulation study shows that the PD-Fuzzy realization reproduces the baseline response around TOP with nearly identical settling times. The hardware study is more demanding: encoder noise, wiring effects, power-stage constraints, mechanical tolerances, and the limited training coverage of the fuzzy blocks influence the observed performance. Under these conditions, the baseline controller remains the more favorable choice in the reported experiment.

The results should be interpreted conservatively. TOP balancing is demonstrated in both simulation and experiment, whereas MID is supported only by local controllability analysis. Swing-up is demonstrated only in simulation. These distinctions clarify which results are experimentally validated and which are supported only by analysis or simulation.

The experimental traces also indicate that joint 2 is more tightly regulated than joint 1 in the baseline test. This difference reflects the distinct roles of the active and passive joints in the underactuated system. Further improvements may include a higher-capacity motor driver, improved encoder wiring, and a broader ANFIS training dataset.

6.5. Reproducibility Notes

The reported information provides sufficient detail to reconstruct the main balancing experiments. The plant dimensions, mechanical parameters, controller gains, sampling time, PWM period, safety limits, encoder processing, motor type, and motor-driver configuration are specified. Table 14 summarizes the parameters relevant to future replication and includes only values explicitly reported in the source.

The simulation and experimental parameter sets are deliberately listed separately. The source does not claim that one parameter set can be transferred unchanged from MATLAB/Simulink to the physical prototype. Mechanical tolerances, signal noise, actuator saturation, and driver limitations require practical retuning. Reporting both sets avoids ambiguity and improves reproducibility.

6.6. Limitations and Future Development

Several limitations are preserved from the source rather than hidden during manuscript preparation. First, the experimental platform does not complete the automatic swing-up task. Second, the encoder wiring introduces noise and the experimental response remains less favorable than the simulation response. Third, this study demonstrates TOP balancing but does not provide experimental MID-balancing data. Fourth, the fuzzy blocks approximate the proportional gains learned from the baseline controller and therefore do not yet exploit a broader operating envelope.

This study proposes four directions for further work: improve the time-varying error of the baseline controller, upgrade the hardware so that swing-up can be tested experimentally, extend the two-link Pendubot to a three-link configuration, and compare the resulting system with other control approaches such as LQR and backstepping. These directions are retained as the future-work agenda of the article.

From a publication perspective, the most important future improvement is a controlled experimental campaign with identical initial conditions for the baseline and fuzzy-enhanced designs. Repeating each test several times and storing the raw encoder and PWM signals would make it possible to compare settling time, overshoot, RMSE, control effort, and robustness under disturbance. The current paper provides the implementation baseline for that next stage.

7. Conclusion

This paper has presented a nonlinear Euler-Lagrange model and an ANFIS-based PD-Fuzzy control for the Pendubot system using a measured parameter set. Local controllability was verified for both TOP and MID equilibria, while the OX-Top configuration was considered as an additional case for comparing controllability. While numerical simulations at the TOP configuration indicated comparable dynamics between the baseline PD and the PD-Fuzzy realizations, experiments on an STM32F407-based hardware platform successfully demonstrated TOP balancing. With current hardware limitations and encoder noise conditions, the baseline PD controller performed more effectively than the fuzzy-enhanced variant. In the future, our work will focus on upgrading equipment, reducing sensor noise, realizing swing-up experimentally, extending the model to three links, and comparing the approach with LQR and backstepping methods.

8. Declarations

8.1. Author Contributions

Van-Long Mach: Data Curation, Software, Validation, Formal analysis, Writing - Review & Editing; **Dang-Khoa Huynh:** Writing - Original Draft, Formal analysis, Validation, Writing - Review & Editing; **Khanh-Hung Le:** Supervision, Project administration, Formal analysis, Writing - Review & Editing; **Vi-Khang Nguyen:** Conceptualization, Formal analysis, Validation, Writing - Review & Editing; **Van-Hai-Dang Ma:** Visualization, Formal analysis, Writing - Review & Editing; **Viet-Phi Le:** Investigation, Formal analysis, Writing - Review & Editing; **Nguyen-Duy-Phuong Huynh:** Formal analysis, Validation, Writing - Review & Editing; **Le-Nhat Nguyen:** Formal analysis, Validation, Writing - Review & Editing; **Quang-Hoa Le:** Methodology, Formal analysis, Validation, Writing - Review & Editing; **Van-Dong-Hai Nguyen:** Conceptualization (Lead), Formal analysis, Writing - Review & Editing.

8.2. Institutional Review Board Statement

Not applicable.

8.3. Informed Consent Statement

Not applicable.

8.4. Data Availability Statement

The data used in this study is available on request from the corresponding author.

8.5. Acknowledgment

Not applicable.

8.6. Conflicts of Interest

The authors declare no conflict of interest.

9. References

[1] Z. Yan,, X. Lai, Q. Meng, M. Wu, J. She, M. Iwasaki, “Modeling, analysis, and adaptive neural modified-backstepping control of an uncertain horizontal pendubot with double flexible joints,” *Control Engineering Practice*, vol. 139, art. No. 105647, 2023. https://doi.org/10.1016/j.conengprac.2023.105647.

[2] P. Parulski, P. Bartkowiak, and D. Pazderski, “Evaluation of Linearization Methods for Control of the Pendubot,” *Applied Sciences*, vol. 11, no. 16, art. No. 7615, 2021. https://doi.org/10.3390/app11167615.

[3] J. G. Romero, and B. Yi, “Immersion and invariance orbital stabilization of underactuated mechanical systems with collocated pre-feedback,” *International Journal of Robust and Nonlinear Control*, vol. 34, no. 14, pp. 9421-9437, 2024. https://doi.org/10.1002/rnc.7467.

[4] V.-T. Nguyen, Q.-C. Nguyen, M. Van, V.-T. Nguyen, D.-H. Pham, D.-C. Nguyen, “Novel adaptive fuzzy control for pendubot with actuator faults and uncertainties: Design and experiments,” *Results in Engineering*, vol. 24, art. No. 102916, 2024. https://doi.org/10.1016/j.rineng.2024.102916.

[5] C. -J. Wu, T. -L. Lee, Y. -Y. Fu and L. -C. Lai, "Auto-Tuning Fuzzy PID Control of a Pendubot System," in *2007 IEEE International Conference on Mechatronics*, Kumamoto, Japan, pp. 1-6, 2007. https://doi.org/10.1109/ICMECH.2007.4280045.

[6] F. Wiebe, S. Kumar, L. J. Shala, S. Vyas, M. Javadi, F. Kirchner, "Open Source Dual-Purpose Acrobot and Pendubot Platform: Benchmarking Control Algorithms for Underactuated Robotics," *IEEE Robotics & Automation Magazine*, vol. 31, no. 2, pp. 113 – 124, 2024. https://doi.org/10.1109/MRA.2023.3341257.

[7] T. N. Nguyen, P. L. Nguyen, and V. D. H. Nguyen, "Pendubot controls using linear quadratic regula-tor combined fuzzy," *Dong Thap University Journal of Science*, no. 38, pp. 89-93, 2019. https://doi.org/10.52714/dthu.38.6.2019.704.

[8] D. A. Q. Nguyen *et al*., "Adaptive Evaluation of LQR Control using Particle Swarm Optimization for Pendubot", *Journal of Fuzzy Systems and Control*, vol. 2, no. 2, pp. 58–66, May 2024. https://doi.org/10.59247/jfsc.v2i2.203.

[9] M. D. Tran, M. T Vo, V. D. H Nguyen. "A comparative study between sliding mode and fuzzy sliding mode control for pendubot", *Serbian Journal of Electrical Engineering*, vol 20, no 3, pp. 361-390, 2023. https://doi.org/10.2298/SJEE2303361T.

[10] C. V. Kien, N. N. Son, H. P. Huy Anh, "Swing Up and Balancing Implementation for the Pendubot Using Advanced Sliding Mode Control," in *Proceedings of the 2015 International Conference on Electrical, Automation and Mechanical Engineering*, 2015. https://doi.org/10.2991/eame-15.2015.108.

[11] W. Wang, J.-Q. Yi, D. B. Zhao, X. J. Liu, "Adaptive sliding mode controller for an underactuated manipulator," in *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.04EX826)*, Shanghai, China, pp. 882-887, 2004. https://doi.org/10.1109/ICMLC.2004.1382310.

[12] N. N. Son, C. V. Kien, H. P. H. Anh, "Uncertain nonlinear system control using hybrid fuzzy LQR-sliding mode technique optimized with evolutionary algorithm". *Engineering Computations*, Vol. 36 No. 6 pp. 1893–1912, 2019. https://doi.org/10.1108/EC-08-2018-0356.

[13] V. T. Tran, T. H. Tran, and V. D. Tran, "Hybrid control for swing up and balancing Pendubot system: An experimental result," in *2017 International Conference on System Science and Engineering (ICSSE)*, 2017. https://doi.org/10.1109/ICSSE.2017.8030915.

[14] C. Lei, R. Li, and Q. Zhu, "Design and stability analysis of semi-implicit cascaded proportional-derivative controller for underactuated cart-pole inverted pendulum system," *Robotica*, vol. 42 , no. 1 , pp. 87 – 117, 2024. https://doi.org/10.1017/S0263574723001352.

[15] I. S. Leal, C. Abeykoon, and Y. S. Perera, "Design, Simulation, Analysis and Optimization of PID and Fuzzy Based Control Systems for a Quadcopter," *Electronics*, vol. 10, no. 18, art. No. 2218, 2021. https://doi.org/10.3390/electronics10182218.

[16] C.-T. Chao, N. Sutarna, J.-S. Chiou, and C.-J. Wang, "Equivalence between Fuzzy PID Controllers and Conventional PID Controllers," *Applied Sciences*, vol. 7, no. 6, art. No. 513, 2017. https://doi.org/10.3390/app7060513.

[17] R.-E. Precup, A.-T. Nguyen, and S. Blažič, "A survey on fuzzy control for mechatronics applications," *International Journal of Systems Science*, vol. 55, no. 4, pp. 771-813, 2024. https://doi.org/10.1080/00207721.2023.2293486.

[18] C. Wei, T. Chai, X. Xin, X. Chen, L. Wang, and Y. H. Chen, "A Signal Compensation-Based Robust Swing-Up and Balance Control Method for the Pendubot," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 3, pp. 3007–3016, 2022, https://doi.org/10.1109/TIE.2021.3065621.

[19] D. Pazderski, P. Parulski, P. Bartkowiak, and P. Herman, "Sub-Optimal Stabilizers of the Pendubot Using Various State Space Representations," *Energies*, vol. 15, no. 14, art. No. 5146, 2022. https://doi.org/10.3390/en15145146.

[20] M. Gulan, M. Salaj, and B. Rohaľ-Ilkiv, "Tracking Control of Unforced and Forced Equilibrium Positions of the Pendubot System: A Nonlinear MHE and MPC Approach," *Actuators*, vol. 12, no. 9, art. No. 343, 2023. https://doi.org/10.3390/act12090343.

[21] V.-T. Nguyen, and Q.-C. Nguyen, "Design of an Adaptive Fuzzy Hierarchical Sliding-Mode Control for Uncertainties Pendubot," *Intelligent Systems and Networks*, pp 568–573, 2022. https://doi.org/10.1007/978-981-19-3394-3_65.

[22] A. Ghosh, A. K. Ray, Md. Nurujjaman, and M. Jamshidi, "Voltage and frequency control in conventional and PV integrated power systems by a particle swarm optimized Ziegler–Nichols based PID controller," *SN Applied Sciences*, vol. 3, art. No. 314, 2021. https://doi.org/10.1007/s42452-021-04327-8.

[23] W. Li, Xi. Wei, D. Sun, S. Zong, and Z. Yue, "Investigation of Trajectory Tracking Control in Hip Joints of Lower-Limb Exoskeletons Using SSA-Fuzzy PID Optimization," *Sensors*, vol. 25, no. 5, art. No. 1335, 2025. https://doi.org/10.3390/s25051335.

[24] A. Hmidet and O. Hasnaoui, "Waijung Blockset-STM32F4 Environment for Real Time Induction Motor Speed Control," in *IEEE 5th International Congress on Information Science and Technology (CiSt)*, 2018. https://doi.org/10.1109/CIST.2018.8596416.

[25] Y. Çakar, Y. Orman, S. Kizir, "Design of a Model Reference Adaptive PID Controller for DC Motor Position Control: Compared with PID and Fuzzy Controllers," *Mugla Journal of Science and Technology*, vol. 6, pp. 25-35. 2020. https://doi.org/10.22531/muglajsci.668552.