SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I2P111
Volume 13 Issue 2, 147-157, February 2026

Original Article

Lagrangian Dynamics and Simulation-Based PID Control of a Two-Joint Lower-Limb Exoskeleton Robot

Thi-Mai-Phuong Dao[1], Van-Nam Vu[2], Kim-An Hoang[2], Tuan-Dat Nguyen[2], Ngoc-Khoat Nguyen[2*], Tien-Loc Le[3], Tien-Dung Nguyen[2], Thi-Kim-Thanh Tran[4]

[1]*School of Electrical and Electronic Engineering (SEEE), Hanoi University of Industry, Hanoi, Vietnam.*
[2]*Faculty of Control and Automation, Electric Power University, Hanoi, Vietnam.*
[3]*Faculty of Postgraduate Studies, Lac Hong University, Dongnai, Vietnam.*
[4]*Center for Practice & Laboratory, Electric Power University, Hanoi, Vietnam.*

[*]*Corresponding Author : khoatnn@epu.edu.vn*

Received: 19 November 2025 Revised: 21 December 2025 Accepted: 22 January 2026 Published: 20 February 2026

Abstract *- Lower-limb exoskeleton robots are used for functional recovery and for supporting people in physical activities. In this study, a PID controller is designed and simulated for a lower-limb exoskeleton system to support the flexion and extension of the hip and knee joints. First, the nonlinear dynamic model of the Exoskeleton is derived using the Lagrangian method. This model is then used to design the controller. The PID parameters are tuned using simulations in MATLAB/Simulink. The performance of the system is evaluated by percent Overshoot (%OS), Settling Time (Ts), and Root Mean Square Error (RMSE) in trajectory tracking. The simulation results show that the PID controller can follow the desired trajectories with acceptable accuracy, and the system remains stable. The steady-state error is small, and the transient response is acceptable for basic practical operation. From the simulation results, PID control can be applied to the Real-Time Control of a 2-DOF Exoskeleton System. This study can also be used as a reference for future studies on Exoskeleton Systems with more degrees of freedom and for applying other strategies such as fuzzy-PID and Sliding Mode Control (SMC).*

Keywords **-** *Lower-limb Exoskeleton, PID controller, Lagrangian model, MATLAB/Simulink simulation, Trajectory tracking.*

1. Introduction

In recent years, Exoskeleton Robots have been attracting a great deal of attention in the areas of Mechatronics and Biomedical Engineering. They are mobility-assisting or function-restoring for people with motor disabilities [1, 2], such as recovering from Injury, Stroke, Neurological Diseases, and so on. Rapid progress in Exoskeleton Technology, along with enhanced quality of life, has introduced new prospects for exoskeleton applications in rehabilitation medicine, military deployment, and heavy industries.

Exoskeleton design has a key challenge in controlling the mechanical joints precisely coordinated with human Biomechanics. Exoskeleton is, by nature, a Nonlinear System, and the elements of Inertia, Friction, varying load, and human-machine interaction greatly affect it [3, 4]. Thus, the choice and the optimization of an adequate control strategy are determinant to guarantee a smooth, stable, and safe tracking of trajectories.

Of the control strategies proposed that are PID (Proportional-Integral-Derivative)-based regulators, it is one of the most used and effective methods due to its simplicity, ease of implementation, and flexibility in tuning. Even though more advanced control techniques such as LQR (Linear Quadratic Regulator), Sliding-Mode Control (SMC), and fuzzy-adaptive have been developed, PID remains a practical choice in early investigations and the simulation due to its simplicity of implementation, robustness , and easy adaptation with mathematical system models.

This study aims to develop the analytical model of a 2-Degree-Of-Freedom (2-DOF) Lower-Limb Exoskeleton with Hip and Knee Flexion/Extension Motion. Applying the Lagrangian approach to establish the dynamic equations, a PID controller is developed and tuned so as to provide an optimal functioning of the system response.

It is demonstrated that with MATLAB/SIMULINK simulation, control performance could be evaluated quantitatively with overshoot, response time, and trajectory-tracking error. Results validate the applicability of PID-based control to lower limb assistive devices and pave the way for future works on multi-DOF exoskeleton models.

2. Control Methodology

To ensure that the Exoskeleton System accurately tracks the desired motion trajectory of human users, a suitable control technique is crucial. By virtue of the nonlinearity of the robot system together with its complicated joint-to-joint couplings, typical control methodology employed at present can be divided into Linear (such as PID/PD/PI) and Nonlinear Controls (LQR, Sliding-Mode Control, Fuzzy control, and Adaptive, etc.). In the context of this early study, the proposed method takes classical PID control to be the basis for most control techniques due to its inherent stability and implemental simplicity as well as its relatively easy correction through parameter identification to fit real-life system dynamics.

2.1. PID Controller

2.1.1. Definition

Among feedback control systems, the PID controller is a popular technique used in automation and mechatronic applications. It is made up of three parallel lines: proportional , integral, and derivative. With the use of all these three terms, the PID controller makes the system stable, precise, and has a fast response with lower overshoot and steady state error.

For the 2-Degree-of-Freedom (2-DOF) Exoskeleton Control, the PID controller controls the angular motions of the Hip and Knee Joints because the real joint trajectories should be very near the Reference joint trajectories based on human biomechanics. For each joint, independent PID controllers are implemented that work independently within the leg movement as a whole.

The control coefficients Kp, Ki, and Kd are tuned to maximize the system's performance metrics, such as overshoot, Settling Time (Ts), and Root Mean Square Error (RMSE). During simulation, parameter estimates are based on the empirical Ziegler-Nichols process and manually refined according to some available gauges (i.e., the average segment length and limb mass of a typical subject). This optimization makes certain that all joints reach their maximum control performance in realistic operational scenarios.

While the PID control is a simple approach, it remains one that offers relatively good performance in terms of basic Exoskeleton Control. This forms a basis for the authors to work on higher-level control algorithms like Fuzzy-PID, LQR, or Sliding Mode that are better adaptive and provide less influence for noise and load change.

2.1.2. Analytical Model of the Controlled System

The considered system in this paper is a 2-DOF lower Limb Exoskeleton, which consists of two main joints , the Hip and Knee, in terms of rotational angles θ1 and θ2. The mechanical structure is described as a link manipulator in a planar configuration, with the movement occurring in the sagittal plane of the human body.

To represent the dynamic behavior of the system, Lagrangian modeling [8] is employed, which is frequently used for multi-degree-of-freedom mechanical systems. The resulting system of equations can be expressed in a general form as follows:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \qquad (1)$$

Where:

$M(q)$ is the system inertia matrix

$C(q, q^{\cdot})$ is the Coriolis and centrifugal force matrix

$G(q)$ denotes the vector of gravitational torques

$\tau = [\tau_1, \tau_2]^T$ is defined as the vector of control torques applied to the joints

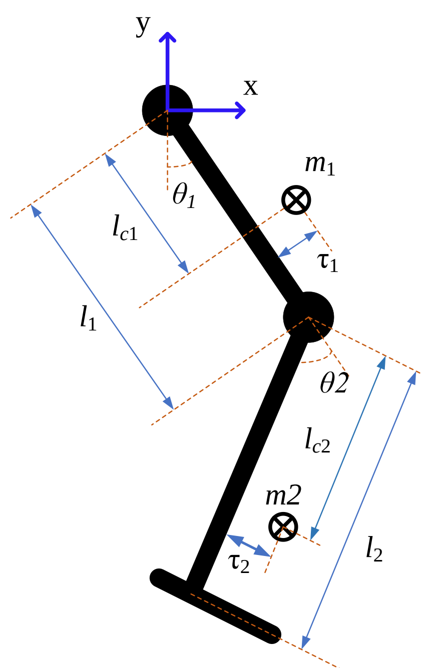

Fig. 1 A 2-DOF lower exoskeleton robot model

The elements of these matrices are determined based on the geometric and mass parameters of each link, as follows:

$$M(q) = \begin{bmatrix} m_1 l_{c1}^2 + I_2 + I_1 + m_2(l_1^2 + l_{c2}^2 + 2\cos(\theta_2) l_1 l_{c2}) & I_2 + (l_{c2}^2 + l_1 l_{c2}\cos\theta_2) m_2 \\ m_2(l_{c2}^2 + l_1 l_{c2}\cos\theta_2) + I_2 & m_2 l_{c2}^2 + I_2 \end{bmatrix} \qquad (2)$$

$$C(q,\dot{q}) = \begin{bmatrix} -m_2 l_1 l_{c2}\sin\theta_2\,\dot{\theta}_2 & -m_2 l_1 l_{c2}\sin\theta_2\,(\dot{\theta}_1 + \dot{\theta}_2) \\ m_2 l_1 l_{c2}\sin\theta_2\,\dot{\theta}_1 & 0 \end{bmatrix} \qquad (3)$$

$$G(q) = \begin{bmatrix} m_2 \cos(\theta_1 + \theta_2) l_{c2} g + \cos(\theta_1)(m_1 l_{c1} + m_2 l_1) g \\ m_2 \cos(\theta_1 + \theta_2) g l_{c2} \end{bmatrix} \tag{4}$$

In the model above, the physical parameters are defined as follows:

- Mass of each link: m_1 = 5kg, m_2 = 3kg.
- Length of the two links: l_1 = 0.5m and l_2 = 0.4m.
- Distances from joints to center of mass: l_{c1} = 0.25m and l_{c2} = 0.2m
- Moment of inertia: I_1 = 0.4167kg· m^2 and I_2 = 0.16kg · m^2
- The acceleration of gravity: g = 9.81 m/s^2

The dynamic equations described above are the analytical model of the 2-DOF Exoskeleton System. In simulation, the control torque $\tau = [\tau 1, \tau 2]^T$ is the input signal and the actual angle positions of each joint θ1(t), θ2(t) are its output.

The model acts as the *plant* or *process under control* for developing and tuning the PID controller in MATLAB/Simulink, so that it can be used to assess how the robot performs in tracking reference trajectories.

2.1.3. Block Diagram of the Controlled System

With the development of the 2-DOF Exoskeleton Model and Mathematical description, this study continues to establish a control closed-loop block diagram in order to simulate and analyze the control process for the tracking accuracy of the trajectory of each joint. The control is formed according to the principle of negative feedback, where the PID controller provides a control torque by using the error between the desired and actual joint angle.

The control architecture consists of the following main functional blocks:

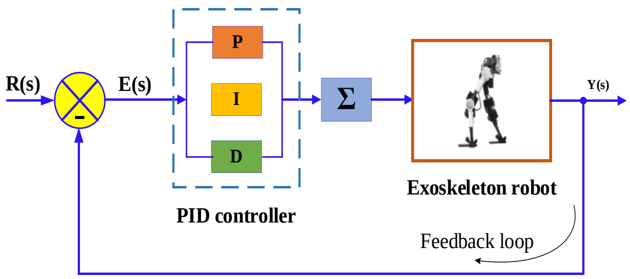

Fig. 2 A PID controller applied for lower exoskeleton robot control

The complete equation of the PID controller is as follows:

$$u(t) = u_i(t) + u_p(t) + u_d(t)$$

$$= K_i \int_0^t e(\tau) d\tau + K_p e(t) + K_d \frac{d}{dt} e(t) \tag{5}$$

2.1.4. PID Controller Design Procedure

The PID controller for the exoskeleton system is designed following a four-stage procedure:

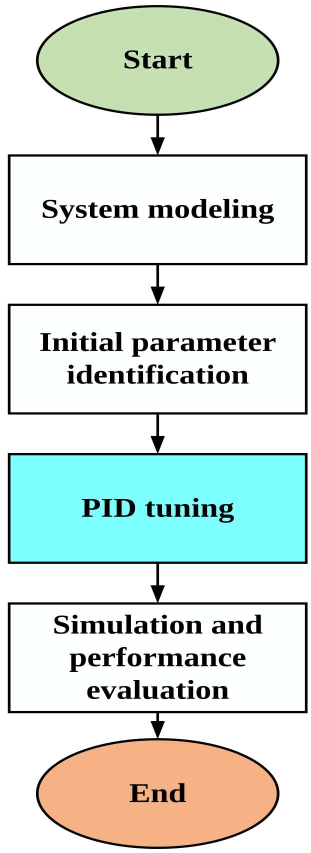

Fig. 3 Design and tuning procedure of the PID controller for the 2-DOF exoskeleton

This approach is applied independently to each joint (hip and Knee) to ensure precise and stable control of the entire lower-limb movement.

Step 1. Modeling the controlled system

First, the 2-DOF exoskeleton system is mathematically modeled based on the Lagrangian approach (described in Section 2.1.2). In this article, the interconnection between the control $\tau = [\tau 1 \ \ \tau 2]^T$, and joint rotation angles $\theta = [\theta 1 \ \ \theta 2]^T$ of each (anatomical) joint is modeled, considering all effects due to inertia, gravity, and dynamic link interaction.

The model is developed in MATLAB/Simulink by using the ODE45 numerical integration algorithm to solve ordinary differential equations (simulated code using a MATLAB function block).

Step 2. Determination of initial parameters

The initial PID controller coefficients are calculated by the empirical Ziegler-Nichols technique with consideration of individual joint position. This method provides an estimation of the three initial gains given the step response of the system, which is utilized prior to performing optimal control.

Step 3. Parameter tuning (PID tuning)

Once the initial settings are provided, it is necessary to perform tuning with the following two parallel strategies:

- Experimental simulation method (Trial-and-Error): Change each parameter one at a time and search for how the implant performance (overshoot/travel distance/settling time) can be improved with less tube movement by keeping the overall stability.
- Performance optimization method: This method employs quantitative evaluation metrics such as Root Mean Square Error (RMSE), settling time, and overshoot to evaluate the quality of each set of parameters.

The parameters are selected when the system achieves the following criteria: OS < 10%, ts < 1.0s, and RMSE < 5.0.

Step 4. Simulation and Performance Evaluation

Simulation is conducted in MATLAB/Simulink, using the reference trajectory θ_{ref} (t) derived from actual human walking data, where the hip angle varies approximately 30-40°, and the knee angle 50-60° within a single gait cycle.

3. Simulation and Result Evaluation

After designing the PID controller, I conducted simulations in MATLAB/Simulink to illustrate the experimental setup, results, and performance analysis of a 2-DOF Exoskeleton System under PID control.

3.1. PID Controller Design for a Two-Degree-of-Freedom Robot

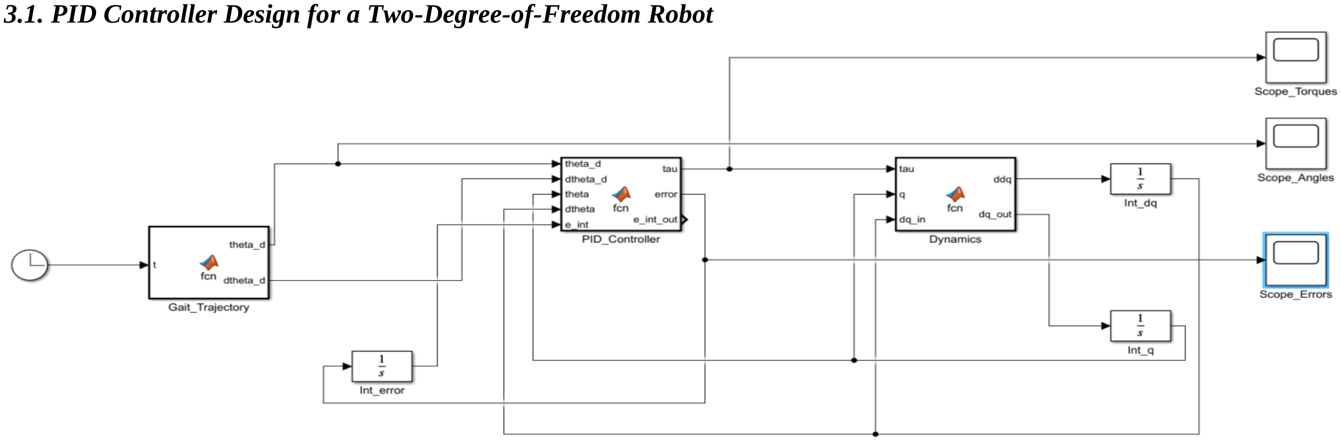

Fig. 4 Design of the MATLAB/Simulink simulation block diagram

3.1.1. Overview of the Structure

The system consists of three main blocks:

1. Gait_Trajectory block

Function:

- Generate the desired trajectory $\theta_d(t)$ $\dot{\theta}_d(t)$ (reference inputs).

Significance:

- This block simulates the human leg motion trajectory, serving as the "target" for the Exoskeleton Robot to track.

2. PID_Controller block

Function:

- Receives the desired and actual signals, computes the error, and generates the control torque τ.

Signal significance:

- The error signal is sent to the Scope_Errors block to monitor the tracking error over time.
- The torque τ is sent to the Dynamics block to simulate the physical response of the system.

Why are e_int_in and e_int_out needed?

They store the continuous integral of the error across simulation cycles, preventing it from resetting at each iteration. The external Int_error block is used to compute:

$$e_{int}(t) = \int e(t)dt \qquad (6)$$

Moreover, feed it back to the PID_Controller.

3. Dynamics block

Function:

- Simulates the physical model of the Exoskeleton (Lagrange equations), receives the control torque τ, and outputs the actual joint angles and velocities.

Mathematical Significance:

The joint accelerations are calculated as:

$$q^{\cdot\cdot} = M - [\tau - q^{\cdot}C(q^{\cdot}, q) - G(q)]1(q) \qquad (7)$$

The acceleration values ddq are integrated twice to obtain:

- Dq out→ joint angular velocity (1^{st} integration)
- q→ joint angle (2^{nd} integration)

Additionally, there are auxiliary blocks such as:

Integrator blocks: simulate actual kinematics, converting torque → acceleration → velocity → angle.

Scope blocks: display results for observation and analysis.

3.1.2. Simulation Setup

The simulation process will be performed on MATLAB/Simulink [9, 10] when the dynamic model of the 2-degree-of-freedom Exoskeleton is created in Chapter 2, Section 2.1.2, using the Lagrange equations. Two separate PID controllers are developed for the Knee and Hip Joints.

The simulation parameters of the system presented for simulation are presented in Table 1.

Table 1. Physical parameters

Symbol	Quantity Names	Value	Unit
m1	Mass of the thigh	5	Kg
m2	Mass of the shank	3	Kg
l1	Length of the thigh	0.5	m
l2	Length of the shank	0.4	m
lc1	Thigh center of mass position	0.25	m
lc2	Shank center of mass position	0.2	m
g	Acceleration of gravity	9.81	m/s^2
j1	Thigh moment of inertia	0.4167	$Kg \cdot m^2$
j2	Shank moment (inertia)	0.16	$Kg \cdot m^2$
b1	Hip joint friction coefficient	0.02	N·m·s/rad
b2	Knee joint friction coefficient	0.01	N·m·s/rad

Table 2. Simulation parameters of the PID controller

Joint	**Kp**	**Ki**	**Kd**
Hip	1207.47	4.49181	29.2485
Knee	1999.83	619.287	3.94485

Table 3. Walking trajectory parameters

Symbol	**Meaning**	**Value**	**Unit**
f_gait	Walking step frequency	0.83	Hz
hip_amp	Hip joint oscillation amplitude	0.35	rad
knee_amp	Knee joint oscillation amplitude	0.567	rad
hip_offset	Average hip joint angle	0.175	rad

Table 4. Simulation settings

Parameter	**Value**	**Notes**
Simulation time	0-6s	Approximately 5 gait cycles
Initial state	0	Initial joint angles, velocities, and error integral
Integration method	ODE45	- Absolute value of the error: 1e-6; - Relative value of the error: 1e-8.

Table 5. Ziegler-nichols method results (closed-loop)

Joint	**Kp**	**Ki**	**Kd**
Hip	120	160	9.38
Knee	96	147.7	7.69

3.2. Simulation Results of 2-DOF Joint Trajectories (Hip + Knee) Under PID Control

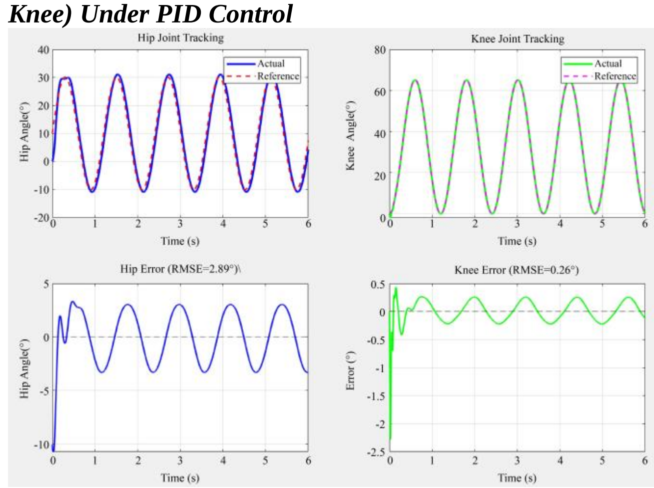

Fig. 5 Time response simulation of the 2-DOF hip and knee joints

3.2.1. Hip Joint Tracking

Blue: Actual Joint Angle

Red dashed: Reference Trajectory

- The hip joint trajectory oscillates between approximately −10° and 30°, simulating the natural flexion-extension motion of the lower limb during a gait cycle.
- The Actual Trajectory (blue) closely follows the Reference (red), indicating practical PID controller tuning.
- Phase lag is minimal (0.05-0.1 s), demonstrating fast system response and reduced delay.
- The actual amplitude closely matches the desired amplitude, confirming that the control torque is sufficient and the Kp: Ki ratio is well-tuned, allowing the system to remain stable without significant overshoot.

3.2.2. Knee Joint Tracking

Green: Actual angle.

Brick Pink: Reference Trajectory.

- The knee joint oscillates between 0° and 65°, correctly representing the physiological flexion-extension movement of the Knee during a gait cycle.
- The actual and Reference paths are almost perfectly superimposed, with near-zero phase and amplitude deviation.

3.2.3. Hip Error

- This is the hip joint's tracking error, in degrees (°).
- The tracking error at the hip joint oscillates between −10° and +5°, averaging around 0°, showing the controller's good balancing capability.
- The error is periodic, with its amplitude decreasing over time, indicating the system is converging to steady-state stability.
- The small RMSE value (2.89°) proves that the Root Mean Square Error (RMSE) is low compared to the motion angle amplitude, indicating high accuracy.

3.2.4. Knee Error

- The knee joint tracking error is tiny, oscillating within ±0.5°, demonstrating near-perfect trajectory tracking capability.
- The error line is stable, has a regular harmonic form, and shows no high-frequency noise or error drift over time.
- The RMSE value = 0.26° indicates the controller achieves very high accuracy, ensuring a smooth dynamic response.

Explanation: The hip joint is in front of the kinematic chain while the Exoskeleton Robot moves it above all joints and needs to pull the whole system by producing torque. Thus, it is primarily influenced by inertia and load torque for the whole leg. Thus, the amplitude and phase delay of hip errors are larger than those at the Knee.

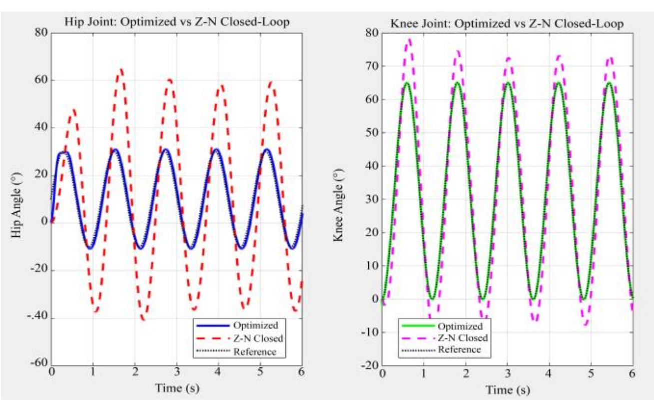

Fig. 6 Comparison results between the two methods

Hip Joint: Optimized vs Z-N Closed
Optimized (Blue): This is the controller tuned and optimized using fminsearch.
Z-N Closed (Brick Red): Actual angle with the PID controller using the Ziegler-Nichols (Z-N) method.
Reference (Black Dotted): Desired Trajectory.

- The hip joint trajectory oscillates between −20° and 30°, corresponding to flexion-extension motion in a gait cycle.
- The Optimized Control Signal (blue) closely tracks the reference trajectory, with amplitude and phase nearly matching, showing high control performance and good stability.
- The Z-N Controller (red) shows excessive amplitude, significant oscillation, and apparent phase deviation, proving the system was not tuned accurately enough for the nonlinear model.

3.2.5. Knee Joint: Optimized vs Z-N Closed

Optimized (Green): Tuned controller using fminsearch.
Reference (Black Dotted): Desired knee trajectory (0° to 65°).
Z-N Closed (Brick Purple): Actual angle with the Z-N PID controller.

- The knee joint oscillates between 0° and 70°, simulating the main flexion-extension movement of the leg.
- The Optimized signal almost perfectly matches the reference trajectory, with slight deviation and high stability.
- The Z-N signal exhibits overshoot and phase lag, showing that parameters purely from Z-N rules are not optimal for a system with complex dynamic interactions.

Conclusions drawn based on the simulation results: It is evident that the optimal PID controller is very good compared with Z-N in a nonlinear model of interconnection. The performance of the hip joint is substantially improved in both reduction of oscillation amplitude and phase lag, with the Knee nearly perfectly modeled.

This result proves the efficacy of the automatic PID optimization method, instead of conventional heuristic rules such as Z-N.

3.2.6. Hip Control Torque (Left)

- Initial torque peak > 200 N·m: This is the startup control pulse as the robot begins to move (from rest state q = 0). The PID requires a large force to overcome initial inertia and gravity.
- In the initial phase, there is a short torque spike due to the PID controller's reaction at startup, compensating for the significant initial error between the actual and desired trajectories.
- After about 0.5s, the torque quickly stabilizes and oscillates harmonically around 0, showing the controller has reached a dynamic equilibrium state.
- The torque maintains a periodic oscillation, in phase with the joint movement, indicating a smooth, stable control response and mechanical energy savings.
- This torque value is within the feasible limits of hip robot actuators, showing high control efficiency and avoiding actuator saturation.

3.2.7. Knee Control Torque (Right)

- The initial peak ~80 N·m is much smaller than the hip. This is because the Knee only lifts the shank, which has less inertia.
- There is an initial torque pulse (~80 N·m) at startup, as the PID reacts to the initial trajectory error, then it quickly reduces to a stable oscillation.
- From t = 0.5s onwards, the torque signal becomes harmonic and regular, with a small amplitude, proving the control system has achieved steady-state stability and good trajectory tracking.
- The knee torque waveform is almost smooth, without spikes, showing coordinated action between the two joints and minimizing mechanical vibrations.

Conclusion

Both joints have cyclically controlled torque, well-damped, suitable for walking dynamics. The hip joint bears the main load and inertial load of the entire leg; thus, it experiences greater torque, while the knee joint moves a shorter distance and therefore reacts faster with a smaller amplitude. The sudden surge and initial strong response, followed by gradual stabilization, demonstrate that the optimized PID controller can achieve fast response along with long-term stability. Experimental results show relatively high performance, the control algorithm does not generate tremendous controlled torque, and there are no saturation conditions on the controller, consistent with the practical design of the robot actuator.

From the results, it is proven that the torque oscillation in the experiment is a combination of three forces: Anti-gravity force + Acceleration/deceleration force + Error correction force. All three forces vary cyclically, so the total torque must also fluctuate.

3.2.8. Explanation of Motion Dynamics

This image is captured at t = 6.00s, the end of a simulation cycle.

Throughout the process, the two joint angles θ1(hip) and θ2 (Knee) vary cyclically according to the sin-cos functions defined in the Gait Trajectory:

$$\theta_1(t) = 0.175 + 0.35\, sin(2\pi f t)$$
$$\theta_2(t) = 0.567(1 - cos(2\pi f t)) \quad (8)$$

This creates a hip-swinging + knee-flexing - extending motion, similar to actual human walking.

Analysis of the state in the simulation image:

- At the end of the cycle, the thigh is slightly inclined forward 10.03° (equivalent to 0.175 rad) relative to the vertical.
- The shank is extended forward, creating a large opening angle between the thigh and shank (~ 64.97°).

Conclusion

The 2D simulation accurately depicts the leg movements during the gait phases. The Hip and Knee Joints coordinate well to form a stable, closed, and symmetrical ankle trajectory, demonstrating the powerful capabilities of the gravity-compensated PID controller in tracking the trajectory. The mechanism operates smoothly and without vibration, simulating natural step-like movements.

X-axis: Position (Joint angle, in degrees).
Y-axis: Velocity (Joint angular velocity, in degrees/second).

Both plots share key characteristics:
1. Startup phase

- The path (blue or green) starts from the origin (0, 0).
- (0, 0) is the start of the simulation (state params.x0 = zeros(6,1)), where the leg is stationary (Angle = 0, Velocity = 0).
- The "tail" spiraling outwards very quickly is the transient response. This is the initial moment (first ~0.5 seconds) when the PID controller gives a strong "kick" (as seen in the torque spike) to bring the system from rest into motion.

2. Closed loop

- This is the most important part. After the short startup phase, both trajectories converge to a stable, closed loop (an ellipse).
- This is called a Limit Cycle.
- Significance: This proves your system is highly stable.
  - It does not spiral outwards (a sign of instability - like the Z-N controller).
  - It does not spiral inwards and stop at a point (a sign of being overdamped and unable to maintain motion).
  - Instead, it enters a periodic, repeating trajectory. This is precisely the desired behavior of a gait.

Hip Phase Portrait

The closed loop oscillates between −10° and +30°, with a peak velocity over 200°/s. The initial kick (the line shooting up) is apparent.

Knee Phase Portrait

The closed loop oscillates between 0° and 65°, with a more symmetrical velocity (around ±160°/s).

Conclusion: These plots are the strongest visual evidence that your controller has succeeded in creating a stable and periodic walking motion.

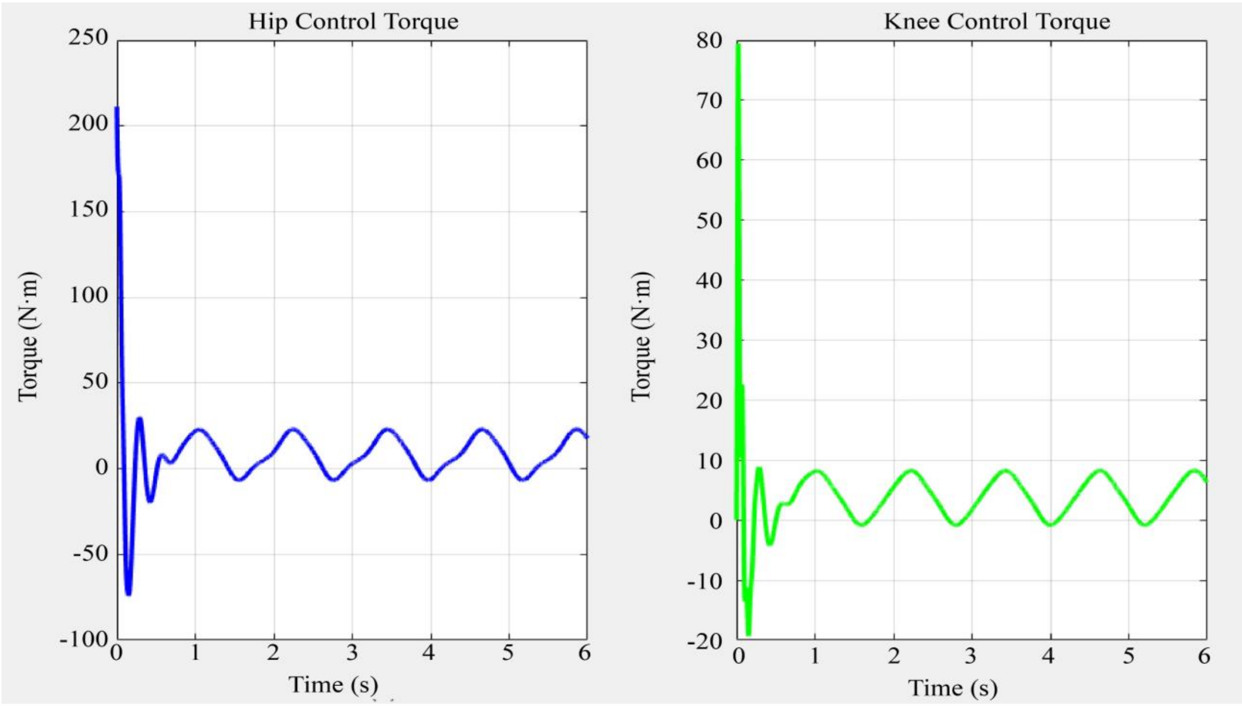

Fig. 7 Control torque graph of the 2 joints

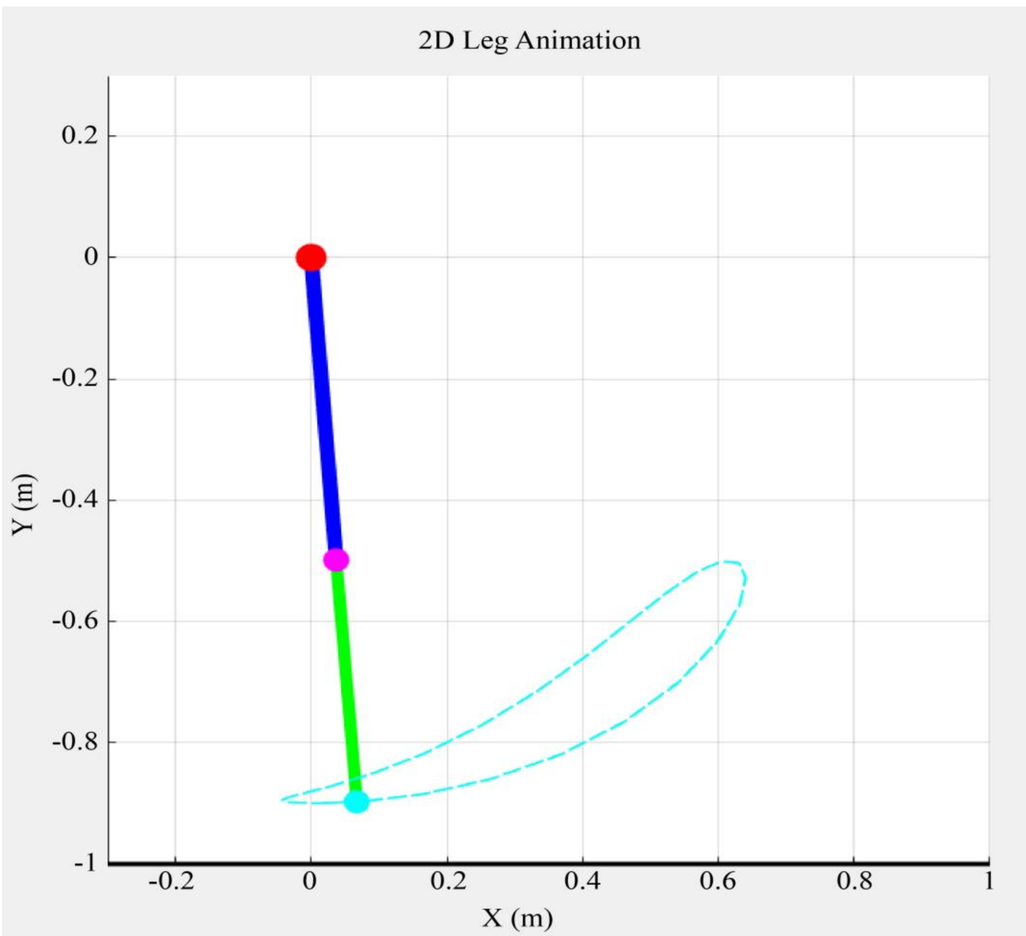

Fig. 8 2D Simulation results of the movement trajectory

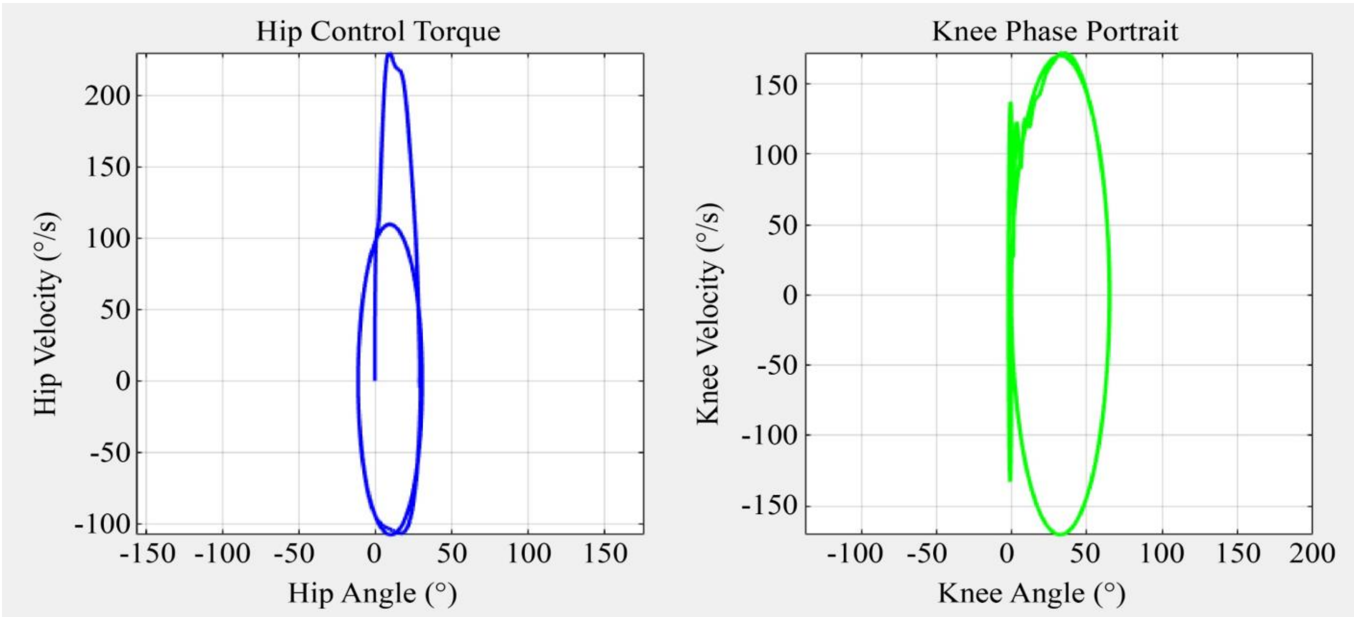

Fig. 9 Phase portrait graph

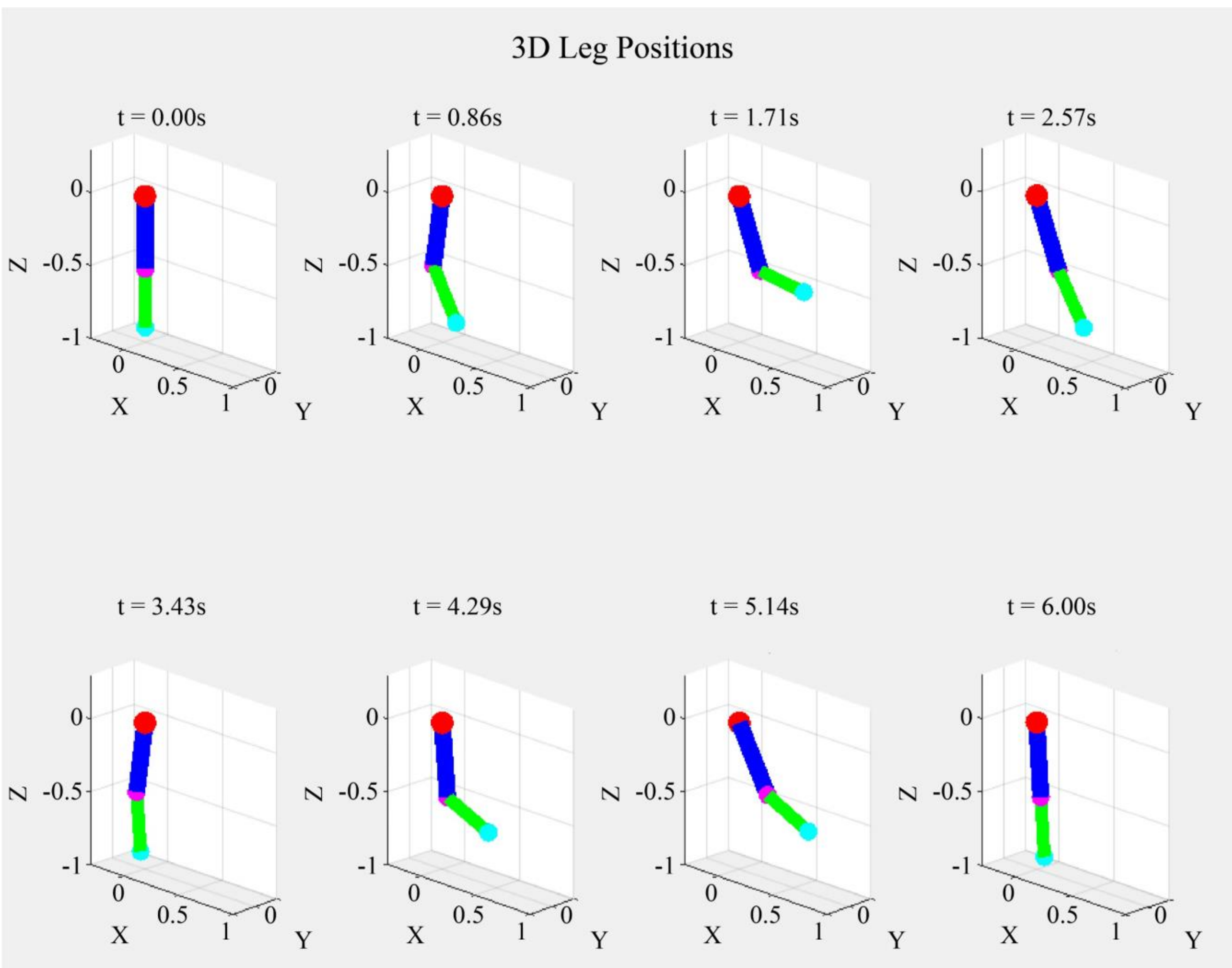

Fig. 10 3D Simulation results of the movement trajectory

3.2.9. Explanation of Motion through time Markers

t = 0.00s (Start): Leg at rest, fully extended (Hip angle = 0°, Knee angle = 0°), corresponding to state x0.

t = 0.86s (Swing): Thigh swings firmly forward (positive hip angle) and knee flexes (positive knee angle) to lift the foot off the ground. t = 1.71s (Peak swing): Thigh reaches its furthest forward swing position. The Knee begins to extend.

t = 2.57s (Mid-cycle): Leg swings to a near-vertical position, preparing for the backward swing phase.

t = 3.43s (Swing back): Thigh swings backward (negative hip angle) to complete the cycle.

t = 4.29s & t = 5.14s (Next cycle): Leg repeats the forward swing and knee flexion, similar to t=0.86s and t=1.71s.

t = 6.00s (End): After nearly 5 cycles, the leg returns to the extended position, ready for a new cycle.

Conclusion

The 3D simulation of the Exoskeleton leg walking has been demonstrated, which can perform a full-gait cycle, with good coordination between the Knee and Hip Joints. The flexion-extension-contact stages pass continuously, then repeat stably over the gait cycles. These results validate that the gravity-compensated PID is very effective and capable of producing motions close to biological gaits.

Table 6. Controller quality evaluation criteria table

Metric	Hip Joint	Knee Joint	Notes
Controller	Optimized PID gravity compensation	Optimized PID gravity compensation	Auto-tuned via fmin search
Kp, Ki, Kd	1207.47 - 4.49181 - 29.2485	1999.83 - 619.287 - 3.94485	Auto-tuned via fmin search
RMSE (rad / °)	0.0504 rad (2.89°)	0.0045 rad (0.26°)	Low error
Max Error (rad / °)	~0.07 rad (~4.0°)	0.044 rad (~2.5°)	Maximum error
Mean |e| (rad / °)	0.1714 rad (9.82°)	0.2160 rad (12.38°)	Mean absolute error
Overshoot (%)	~ +5.0	~ +1.0	Low os
Settling Time (s)	0.12s	~0.00	Swift response
Control Effort ($N^2 \cdot m^2 \cdot s$)	2975.92	309.81	Energy consumption
Overall evaluation	Tracks well, low error	Very accurate & stable	Suitable for 2DOF gait

3.3. Analysis and Discussion

The PID controller achieved results that ensured good feedback characteristics for trajectory stability and tracking. The achieved parameters showed a good balance between sensitivity and stability of the system:

- Increasing Kp leads to a faster system response and is more susceptible to oscillations.
- Increasing Ki helps reduce the steady-state error.
- And this Kd component helps combat interference during malfunctions, thus minimizing it and improving stability.

Based on the simulation analysis conclusions, it can be concluded that the PID controller still provides some good performance [11, 12] for controlling the Exoskeleton in the simulation research phase and provides a basis for deploying various advanced control techniques such as Fuzzy-PID, LQR, and Sliding Mode Controller in the future.

4. Conclusions and Future Work

4.1. Conclusion

The present paper reports the design, simulation, and testing of a PID controller for a 2-DOF Lower Limb Exoskeleton. The paper utilizes the Lagrange formulation to derive a dynamic model of the system, which accounts for the mechanical properties and control feedback precisely. For the control section, the modifications were done in MATLAB/Simulink environment for simulation. Emphasis is mainly laid on the choice of the parameters so that system stability, fast response, and low tracking errors are simultaneously balanced and optimized.

After obtaining the simulation results, it was demonstrated that the Exoskeleton system has stability, low error [12-14], and accurately tracks the trajectory. In conclusion, the study confirmed that the classical PID architecture is a cost-effective and straightforward method that can be used as a first-line control architecture as a foundation for developing intelligent control techniques, in addition to offering several additional degrees of freedom for future applications.

4.2. Future Work

Although the PID controller has demonstrated excellent control performance of the 2-DOF Exoskeleton Robot, there are still some limitations in this study that need to be further investigated in the future. There are four main focuses for future development:

- Exploring 4-DOF models with different degrees of freedom.
- Applying optimization methods (e.g., GA, PSO) to automatically adjust PID parameters, thereby reducing design time and improving control performance.
- Implementing higher-order control strategies such as Fuzzy-PID, Sliding Mode Control (SMC), and LQR [15-17] to improve adaptability and overcome the effects of disturbances or user load changes.
- Testing and evaluation on physical devices.

References

[1] Jun Min Cha et al., "Wearable Robots for Rehabilitation and Assistance of Gait: A Narrative Review," *Annals of Rehabilitation Medicine*, vol. 49, no. 4, pp. 187-195, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[2] Lei Sun et al., "A Soft Exoskeleton for Hip Extension and Flexion Assistance based on Reinforcement Learning Control," *Scientific Reports*, vol. 15, pp. 1-18, 5435, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[3] Zhenlei Chen et al., "Control and Implementation of 2-DOF Lower Limb Exoskeleton Experiment Platform," *Chinese Journal of Mechanical Engineering*, vol. 34, pp. 1-17, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Sergei V. Sarkisian, Lukas Gabert, and Tommaso Lenzi, "Series-Elastic Actuator with Two Degree-of-Freedom PID Control Improves Torque Control in a Powered Knee Exoskeleton," *Wearable Technologies*, vol. 4, pp. 1-14, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Rabé Andersson, Mikael Cronhjort, and José Chilo, "Wireless PID-Based Control for a Single-Legged Rehabilitation Exoskeleton," *Machines*, vol. 12, no. 11, pp. 1-26, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Mohammad Soleimani Amiri et al., "Adaptive Particle Swarm Optimization of PID Gain Tuning for Lower-Limb Human Exoskeleton in Virtual Environment," *Mathematics*, vol. 8, no. 11, pp. 1-16, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] J. Annisa et al., "Optimizing PID-Based Controller Utilizing Hybrid Evolutionary Algorithmn in Electric Motor-Driven Exoskeletons for Therapeutic Locomotion of Stroke Patients," *2024 IEEE 10th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*, Bandung, Indonesia, pp. 207-211, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] Umidjon Abdumutalipovich Takabaev, and Zafar Botirovich Juraev, "Development and Analysis of the Kinematic-Dynamic Model of the Rehabilitation Exoskeleton System," *Technical Science and Innovation*, vol. 2025, no. 1, pp. 49-54, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[9] Mehmet Işcan et al., "Design and Control of an Exoskeleton Robot: A MATLAB Simscape Application," *4th International Conference On Advances In Mechanical Engineering*, Istanbul, Turkey, pp. 1867-1877, 2018. [Google Scholar]

[10] Rajan Prasad et al., "Bi-Planar Trajectory Tracking with a Novel 3DOF Cable Driven Lower Limb Rehabilitation Exoskeleton (C-LREX)," *Sensors*, vol. 23, no. 3, pp. 1-15, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Jiaqi Liu, Hongbin Fang, and Jian Xu, "Online Adaptive PID Control for a Multi-Joint Lower Extremity Exoskeleton System Using Improved Particle Swarm Optimization," *Machines*, vol. 10, no. 1, pp. 1-15, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] Wei Li et al., "Investigation of Trajectory Tracking Control in Hip Joints of Lower-Limb Exoskeletons Using SSA-Fuzzy PID Optimization," *Sensors*, vol. 25, no. 5, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[13] Pengfei Zhang, and Xueshan Gao, "Fuzzy Adaptive Sliding Mode Control of Lower Limb Exoskeleton Rehabilitation Robot," *Proceedings of the Romanian Academy, Series A*, vol. 23, pp. 379-387, 2022. [Google Scholar] [Publisher Link]

[14] N.A. Alawad, A.J. Humaidi, and A.S. Alaraji, "Sliding Mode-Based Active Disturbance Rejection," *Annals of the Brazilian Academy of Sciences*, vol. 95, no. 2, pp. 1-10, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[15] Hafte Tkue, Kassahun Berisha, and Fisseha Teklay, "Designing a Fuzzy Sliding Mode Controller for 2-DOF Lower Limb Exoskeleton Rehabilitation Robot," *Research Square*, pp. 1-16, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Al Asady Mohanad, and Takhakh Ayad Murad, "Eight Degree of Freedom Upper Limb Rehabilitation Robot Optimization for Proportional-Integral Derivative Controller," *Library of Progress-Library Science, Information Technology & Computer*, vol. 44, no. 2s, 2024. [Google Scholar]

[17] Yan Shi et al., "Model Parameters Identification and Backstepping Control of Lower Limb Exoskeleton Based on Enhanced Whale Algorithm," *Chinese Journal of Mechanical Engineering*, vol. 37, pp. 1-15, 2024. [CrossRef] [Google Scholar] [Publisher Link]