International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 21, Number 3 (2026) pp. 351-372
Research India Publications
https://www.ripublication.com

Performance Evaluation of Kookaburra and Driving Training-Based Optimization for tuning of High-Precision Velocity Tracking Controller in Conveyer Drive Systems

Hyreil Anuar Kasdirin*, and Muhammad Aiman Firdaus Bota
Faculty of Electrical Technology and Engineering,
Universiti Teknikal Malaysia Melaka (UTeM),
76100, Durian Tunggal, Melaka, Malaysia
**Correspondence: hyreil@utem.edu.my*

Abstract

This project addresses the critical challenge of achieving high-precision transient performance and robust velocity tracking in industrial conveyor belt systems driven by permanent magnet DC motors. While Proportional-Integral-Derivative (PID) controllers are widely favoured for their structural simplicity, conventional manual and deterministic tuning techniques frequently fail under conditions of actuator voltage saturation (+24.0V), mechanical load uncertainty, and inherent system nonlinearities. This study presents an intelligent hybrid tuning framework by benchmarking two advanced metaheuristic optimization algorithms: the Kookaburra Optimization Algorithm (KOA) and the Driving Training-Based Optimization (DTBO) method. The control architecture was meticulously modelled and validated within the MATLAB/Simulink environment, utilizing the Integral of Time-multiplied Absolute Error (ITAE) as the primary objective function to penalize transient oscillations and steady-state offsets effectively. Comparative analysis reveals that while KOA suffers from premature convergence due to rapid stochastic diversity loss, the multi-phase DTBO comprising Instructor Training, Pattern Mimicking, and Autonomous Practice consistently navigates the search space to reach an elite global minimum. Quantitative results confirm the superiority of the KOA-tuned PID controller, achieving a dramatic reduction in time near-steady state response of 2.199 s and lower steady state error (0.001 rad/s or 0.10 %), and profound resilience to parameter distortions. This research establishes a highly reliable, model-independent blueprint for the next generation of automated industrial motor control architectures.

Keywords: Conveyer Drive Systems; PID controller; Kookaburra Optimization Algorithm; Driving Training-Based Optimization; Auto-tuning

1. Introduction

Automated systems have evolved from simple mechanical synchronization to highly decentralized, intelligent, and fast-responding network ecosystems in today's industrial environment. Speed-tracking machinery is essential to systems like computerized numerical control (CNC) machining centers, multi-axis robotic arms, robotic assembly plants, material conveyor mechanisms, and electric vehicle (EV) drivetrains. Electric motors, namely Direct Current (DC) motor drives and Brushless Direct Current (BLDC) motors, are at the heart of these automated systems. Because of their remarkable power-to-weight ratios, linear torque-speed curves, simplicity of speed manipulation, and great dependability over long operating lives, these devices are frequently utilized.

The controller architecture must execute fast setpoint tracking while continually suppressing disturbances to properly integrate these motors into a complex industrial framework. The Proportional-Integral-Derivative (PID) control method has been the mainstay of industrial automation for more than 70 years. Its enduring popularity can be attributed to its straightforward structure, low real-time computing burden, cheap deployment cost, and the obvious physical significance of its three tuning parameters (K_p, K_i, and K_d gains). A particular portion of the signal response error is fixed by each parameter. Under steady-state operation, the integral term pushes the tracking offset to zero, the proportional term reduces the initial rising lag, and the derivative action predicts future trends and adds the required damping to stabilize the physical loop.

The tuning stage is the main obstacle to reaching optimal performance, despite its widespread use. Finding the precise set of the proportional (K_p), integral (K_i), and derivative (K_d) is a very non-linear optimization problem. The Ziegler-Nichols (ZN) frequency-response and Cohen-Coon open-loop approaches are two examples of traditional heuristic rules that depend on simplistic plant models. When applied to high-order, non-linear physical plants, these techniques usually result in highly oscillatory responses with excessive overshoots and extended settling periods. In order to get optimal performance, the discipline of control engineering has resorted to artificial intelligence (AI) and soft-computing paradigms, particularly metaheuristic optimization techniques.

In modern industrial logistics, maintaining a strictly constant velocity profile for conveyor belt handling systems is critical to eliminate material spillage, reduce component slippage, and minimize mechanical wear on the primary drive shaft. However, the velocity regulation of a permanent magnet DC motor-driven conveyor system presents significant control vulnerabilities during its initial startup and transient phases due to several coupled engineering challenges.

Therefore, the integration of a multi-phase metaheuristic optimization engine, such as the Driving Training-Based Optimization (DTBO) method, to auto-tune the K_p, K_i, and K_d gains represent a crucial research need to balance global exploration and fine local exploitation. Therefore, in resolving the control limitations outlined in the problem statement, this research project dedicates itself to completing the core objectives by investigating and simulating the optimization convergence capabilities and transient tracking profiles of Driving Training-Based Optimization (DTBO) and

the Kookaburra Optimization Algorithm (KOA) metaheuristic algorithms in achieving stable conveyor reference velocity tracking with minimized overshoot and rapid settling times. The plant used is a conveyer drive system plant model in a continuous-time, closed-loop Proportional-Integral-Derivative (PID) speed control loop under strict $\pm 24.0V$ actuator saturation limits using the MATLAB/Simulink environment.

2. Literature Review

This sections reviews and offers a thorough analysis of earlier studies on motor drive speed regulation in contemporary industrial conveyer drive automation systems. After establishing the crucial connection between automated conveyor belt logistics and high-precision motion drives, it provides a thorough explanation for choosing the Proportional-Integral-Derivative (PID) controller architecture. In order to support the hybrid intelligent autotuning framework suggested in this study, this chapter also critically assesses the intrinsic shortcomings of traditional deterministic tuning metrics, examines cutting-edge, state-of-the-art metaheuristic optimization techniques, and describes the basic mathematical formulas utilized inside the Simulink circuit layout.

2.1 Conveyer Drive Systems

High-efficiency motion drives are essential for optimizing system throughput in the context of contemporary production and smart factory paradigms controlled by Industry 4.0 [14]. An Automated conveyor belt systems move goods between synchronized robotic arms, sorting stations, and assembly hubs, acting as the structural logistical backbone of factory floors [13]. Reducing velocity variations is crucial for these continuous operations in order to prevent line bottlenecks, structural shocks to cargo, or complete system shutdowns [15].



Figure 1: Parameters and constant values of the DC motor model.

These automated systems are powered by electric motors, namely brushless direct current (BLDC) and direct current (DC) motor drives [2, 16]. Because of certain

verified electromechanical properties, the drive systems used industrial automation actuators are often high starting torque to smoothly accelerate big conveyor payloads from a full halt, the maximum electromagnetic torque is generated at zero speed [11, 12].

The actuator is also in linear controllability. This is conveyed its designing linear control loops and manipulating speed are made much easier by linear speed-torque curves [4]. High Reliability: Plant output is maximized by exceptional power-to-weight ratios and outstanding reliability during extended operational lives [17]. The industrial actuator is also having field efficiency where its permanent magnet topologies maximize energy efficiency for 24-hour industrial operations by eliminating continual wound-field stator copper losses (I^2R) [15].

However, real-world conveyor operations introduce a dynamic control problem: loading and unloading heavy components shifts the system's payload [13]. In control theory, these unpredictable payload fluctuations alter the physical rotor inertia (J) and viscous damping friction (B) [3, 19]. This dynamic parameter variation, alongside environmental factors like winding temperature accumulation, creates a widening error between target and actual velocity, necessitating a robust, closed-loop control system [5, 16].

2.2 Proportional-Integral-Derivative (PID) Controller

One of the most popular control techniques in industrial automation is the proportional-integral-derivative (PID) controller because of its straightforward design, resilience, and capacity to deliver adequate dynamic performance. Because PID controllers can efficiently minimize steady-state error while retaining acceptable transient performance, they are widely utilized in motor drives, conveyor systems, robotics, process industries, and manufacturing applications. The controller creates a suitable control action to reduce the error by continually calculating the difference between the desired reference signal and the actual system output [21].

Three elements make up a PID controller's control action: derivative, integral, and proportional actions. Faster system reaction is made possible by the proportional term, which produces an output proportionate to the instantaneous mistake. While the derivative term forecasts future error trends and enhances damping properties, the integral term collects prior errors and removes steady state offset. The PID controller balance’s reaction speed, overshoot, and stability by integrating these three activities [22].

PID controllers have shown to operate very well in DC motor speed control applications, sustaining the intended speed in the face of load disturbances and parameter fluctuations. However, the choice of a PID controller's tuning parameters K_p, K_i, and K_d gains have a significant impact on the controller's overall efficacy. Excessive overshoot, oscillations, or a long settling period might result from improper tuning. As a result, many optimization methods have been put forth to automatically identify the ideal PID settings [23].

Despite decades of enhanced control research and the development of high-order algorithms, classic Proportional-Integral-Derivative (PID) architectures still rule most industrial control loops [10, 17] as shown in Figure 2.

Table 1. Comparison of control systems based on their feature

Control Feature	Linear Quadrator Regulator	Fuzzy Logic Control	Model Predictive Control	PID Controller
Model Dependency	High [16]	Low [22]	Extremely High [16]	Low [6], [7],[21]
Real-time Overhead	Moderate [16]	Moderate [22]	Extremely High [15]	Extremely Low [11],[12]
Industrial Viability	Low [16]	Moderate [22]	Low [16]	High [17],[18]

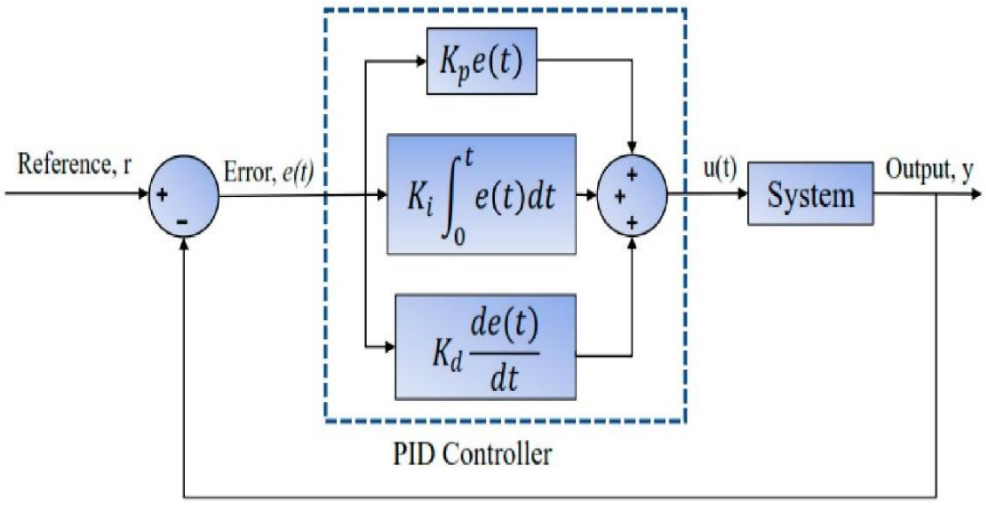


Figure 2: Block diagram of a standard parallel PID controller structure in a closed-loop system.

2.3 Drawbacks of Conventional Heuristic Rules and Tuning Paradigms

Investigating classical industrial loops reveals core drawbacks when conventional deterministic analytical tuning techniques are applied to high-order, non-linear physical plants under operational stress [2, 14] as explained in Table 2

Table 2. Drawbacks of the Conventional Tuning Techniques

Section	Drawback of the Conventional Tuning Techniques
Locally Bound Performance	Conventional tuning processes, such as the Cohen-Coon approaches or the Ziegler-Nichols (ZN) paradigm (1942), depend on basic first-order models with time delay [1, 9]. Only in the vicinity of a tiny, predefined nominal operating point can they function effectively [14].
The Overshoot vs. Speed Dilemma	Rapid acceleration is a major component of traditional heuristic rules [18]. They often produce highly oscillatory responses with severe overshoots and prolonged settling periods when subjected to parameter changes or unanticipated conveyor load disturbances, which results in fragile items on the conveyor belt toppling over [4, 13].
Conditioned Manual Tuning	Both manual and graphical trial-and-error tuning loops often trap the system in sub-optimal gains, require a long time to perform, and significantly rely on operator knowledge [8, 10].

2.4 Evolution of Metaheuristic Algorithms for Parameter Tuning

The field of control engineering has looked to artificial intelligence and soft-computing paradigms to get beyond the strict constraints of classical control theory [2, 16]. The PID tuning procedure is efficiently handled by metaheuristic algorithms as a population-based stochastic search across a restricted three-dimensional parameter workspace [3, 10]. These methods efficiently handle non-linear phenomena such actuator saturation, dead-zones, and friction because they avoid complicated mathematical models of the plant [14].

There is also limitation of such metaheuristic algorithms. For example, of such algorithms such as PSO, the limitation is by monitoring personal best (P_{best}) and global best (G_{best}) locations, PSO simulates the social food-searching behavior of bird flocks [2, 16]. Even if it converges quickly, it stalls early during difficult tuning problems because it has no way out of local minima once its velocity vector falls close to zero [2]. Other than that, for the limitation of GA is shown by monitoring personal best (P_{best}) and global best (G_{best}) locations, PSO simulates the social food-searching behavior of bird flocks [2, 16]. Even if it converges quickly, it stalls early during difficult tuning problems because it has no way out of local minima once its velocity vector falls close to zero [2].

In recent years, researchers have developed advanced algorithms that mimic natural physics, animal behaviours, or human interactions to balance global exploration and local exploitation [15]. Grey Wolf Optimisation or in short, GWO

imitates the pack-hunting tactics and rigid social structure (alpha, beta, delta, and omega levels) of grey wolves in the wild [2]. Although GWO outperforms standard PSO in terms of convergence, it frequently leaves a little steady-state tracking error in the controller settings due to its difficulties with precision refining in the latter phases of optimization [16].

Driving Training-Based Optimization DTBO mimics how individuals learn to drive and was initially used in [3] to optimize high-inertia motor speed drives. Phase 1 (instructor teaching) drives large global exploration, whereas Phase 2 (individual practice) focuses on local exploitation around optimum zones [3, 19]. When given complex multi-stage target profiles, it can nevertheless converge slowly even though it beats GWO on tracking criteria and eliminates massive transient oscillations [19]. Inspired by the human learning process at driving schools, Driving Training-Based Optimization (DTBO) is a relatively new population-based metaheuristic algorithm. The system was developed to mimic how trainee drivers and driving instructors interact during the learning process. Throughout the optimization phase, the algorithm may successfully strike a balance between exploration and exploitation because to this learning mechanism [25]. The DTBO algorithm goes through three main stages of operation. By directing candidate solutions toward seasoned teacher solutions, the initial stage known as instructor training promotes global exploration. In the second stage, students copy teachers' effective driving techniques. Lastly, individual solutions are able to carry out local refining around promising locations during the autonomous practice phase. Together, these three phases increase the likelihood of finding the global optimal solution and speed up convergence [25].

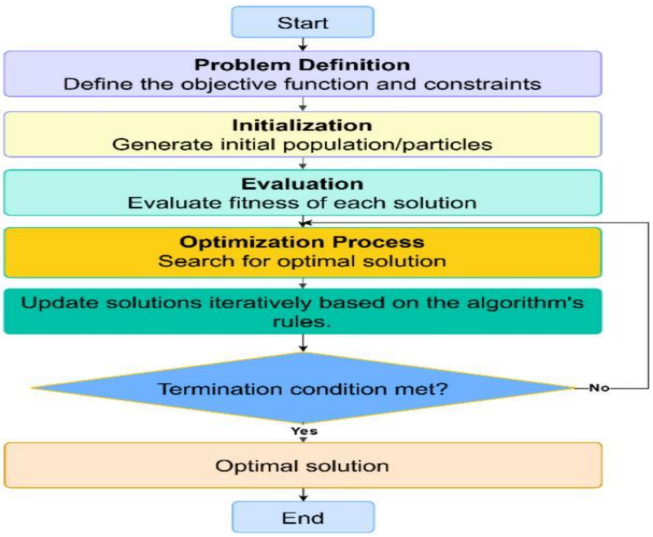


Figure 3: Principle of DTBO algoritm

The efficiency of DTBO for adjusting PID controllers in motor speed control systems has been documented in several research. Compared to some of few traditional optimization methods, the algorithm has shown better tracking performance and greater convergence characteristics. During optimization, DTBO is very useful for

lowering objective function values, eliminating transient oscillations, and cutting settling time [25, 26].

Another metaheuristics algorithm that is used in this project is Kookaburra Optimization Algorithm KOA. Introduced in [1], Kookaburra Optimization Algorithm KOA is a nature-inspired metaheuristic that mimics the collective vocalizations and predatory hunting actions of kookaburras [20]. KOA uses a very dynamic initial population phase that disperses agents broadly throughout the search region to achieve impressive rising times and outperform rivals like the Red Panda Optimization Algorithm (RPOA) under varying loads [1]. A new optimization method inspired by nature, the Kookaburra Optimization Algorithm (KOA) simulates the social and hunting activities of kookaburra birds. The program imitates kookaburra populations' cooperative hunting tactics, territorial communication, and prey-handling behaviours. These biological traits are converted into mathematical operators that direct potential solutions to the best areas in the search space [26].

There are two main phases to KOA. In order to find interesting areas, candidate solutions explore widely throughout the solution space during the initial stage of global exploration. By fine-tuning solutions around the present optimal position, the second step carries out local exploitation. With the help of this dual-stage method, KOA is able to preserve variety in the initial iterations while progressively increasing the accuracy of the solution as the optimization goes on [26]. KOA can effectively tune PID controllers for DC motor applications, according to recent study. When compared to a number of conventional optimization methods, the algorithm has demonstrated improved transient response characteristics and improved speed regulation performance. However, research has also shown that when population variety quickly declines in subsequent iterations, KOA may occasionally encounter premature convergence [26].



Figure 4: Kookaburra Bird

Both DTBO and KOA are population-based metaheuristic optimization methods that have been effectively used to solve PID controller tuning issues. However, there are substantial differences between the two algorithms' basic search methods. While KOA draws inspiration from cooperative hunting behaviour in nature, DTBO depends on human learning behaviour incorporating teacher assistance and independent practice [25]. Due to its organized multi-stage learning process, DTBO often offers higher

exploitation capabilities, according to earlier research. This feature enables DTBO to tweak parameters more precisely in the last phases of optimization. On the other hand, during the first search phase, KOA has a high global exploration capability that allows it to swiftly identify potential places. However, in some optimization settings, the aggressive decrease of search step size may raise the danger of premature convergence [26]. In order to ascertain which algorithm provides better tracking performance, lower ITAE values, and enhanced convergence characteristics under the same operating circumstances, it is crucial to compare DTBO and KOA for PID tuning of a DC motor conveyor system. This kind of comparison offers important information about each optimization strategy's viability for industrial speed control applications.

3. Methodology

The methodology in this project consists of three stages: first, a conveyer systems simulation model platform; second, a control strategy that consist of error comparator combined with ITAE as the algorithm's objective function; and the third, the trajectory of the systems and the searching solution methods from the algorithms.

In the discipline of control engineering, research technique is the primary means of guaranteeing the validity, accuracy, and reproducibility of a scientific study. A well-organized methodological framework enables dynamic analysis of the interplay between digital software (optimization algorithms) and physical hardware (plant) in industrial automation systems. The system design, mathematical modelling based on first-principles modelling, theoretical justifications for controller structure selection, performance cost function computation, and the mathematical foundation for two created metaheuristic algorithms are all covered in detail in this chapter.

The problem of stabilization and angular speed tracking for industrial conveyor belt systems powered by permanent magnet direct current (DC) motors is the primary subject of this study. The early transient phase, nonlinear limitations resulting from drive voltage saturation, and the contribution of plant characteristics (such as friction and inertia) to dynamic errors are the primary obstacles in this speed control system. The Proportional-Integral-Derivative (PID) control structure is used as the primary regulator in order to address this problem. This study incorporates two contemporary metaheuristic optimization algorithms since conventional tuning techniques are limited in their ability to handle nonlinear constraints; Kookaburra Optimization Algorithm (KOA) and Driving Training-Based Optimization (DTBO). A hybrid simulation environment that combines Simulink dynamic block diagrams and MATLAB programming scripts is used to evaluate the complete modelling and optimization system. To make it easier to comprehend the shift from physical circuit parameters to digital algorithm proofs, this chapter is structured in phases.

3.1 Conveyer Plant Model as the Model Platform

The ability of the electric motor to maintain a perfectly constant velocity profile is crucial for material handling systems that use conveyor transits in the contemporary manufacturing and logistics industries. Unexpected speed fluctuations generate

significant mechanical stress across the pulley drive shaft, cause material spillage or ripping owing to structural slippage, and jeopardize the product feed rate.

The interrelated physical subsystems make up the conveyor plant represented in this study is by using power regulation unit, permanent magnet DC motor, and mechanical load. Power Regulation Unit (Variable Frequency Drive/DC Chopper) is chosen to activate the motor coils where it converts the digital voltage instruction from the PID block into a bounded analogue armature voltage. Permanent Magnet DC Motor (The Primary Actuator) is chosen due to its fast frequency response to variations in input armature voltage, strong beginning torque capabilities, and linear torque-speed characteristics. Mechanical Load (Belt and Pulley Friction) refers to the total rotational inertia of the driving pulley and motor rotor as well as the viscous friction produced across the sliding belt support bed. Mathematical Modelling of the Physical Plant (First-Principles Derivation.

Insert a highly accurate mathematical model is an absolute prerequisite to guarantee the validity of the control simulation. The physical permanent magnet DC motor plant is derived by coupling Kirchhoff's Voltage Law (electrical domain) with Newton's Second Law of Motion (mechanical domain).

3.2 The Electrical Armature Circuit, Mechanical Rotor and Load Dynamics

An internal resistance (Ra), an armature winding inductance (La), and an induced back-electromotive force voltage (back-EMF, eb) produced by the armature conductors rotating inside the magnetic flux field make up the motor's internal armature circuit. The following is the loop equation that Kirchhoff's Voltage Law governs:

$$u(t) = R_a i_a + L_a \frac{di_a(t)}{dt} + e_b(t) \qquad (Eq\ 3.1)$$

Where ia (t) is the armature current and u(t) is the saturated armature voltage that enters the motor terminals. The motor shaft's rotating angular velocity (ω(t)) scaled by the back-EMF constant (Kb) is directly proportional to the back-EMF voltage (eb):

$$e_b(t) = K_b \omega\ (t) \qquad (Eq\ 3.2)$$

Substituting Equation $(u(t))$ into Equation $(e_b(t))$ and isolating the derivative of the armature current yields the first-order electrical state differential equation:

$$\frac{di_a(t)}{dt} = \frac{1}{La}\left[u(t) - R_a i_a(t) - Kb\ \omega(t)\right] \qquad (Eq\ 3.3)$$

Through the motor torque constant (Kt), the armature current (ia) and the internal electromagnetic torque (Tm) generated by the DC motor are directly proportional:

$$T_m(t) = K_t i_a(t) \qquad (Eq\ 3.4)$$

This electromagnetic torque is used to counteract any external resistive load torque (T_L), overcome the internal mechanical viscous friction coefficient (B), and accelerate

the system's total combined rotational inertia (J). Newton's Second Law governs the mechanical torque balance equation, which has the following structure:

$$T_m(t) = J\frac{d\omega\ (t)}{dt} + B\omega(t) + T_L(t) \qquad (Eq\ 3.5)$$

Substituting Equation ($T_m(t)$) provides the first-order mechanical state differential equation for the conveyor's angular velocity:

$$\frac{d\omega(t)}{dt} = \frac{1}{J}[K_t i_a(t) - B\omega(t) - T_L(t) \qquad (Eq\ 3.6)$$

Assuming a free-running steady state with no external active load step block applied (TL = 0), the continuous-time mechanical differential equation in your particular dcmotor3.slx Simulink block diagram (as shown in Figure 5) is mathematically expanded by dividing the physical constants directly by the inertia factor (J):

$$\frac{d\omega(t)}{dt} = \left(\frac{K_t}{J}\cdot i_a(t)\right) - \left(\frac{B}{J}\cdot \omega(t)\right) \qquad (Eq\ 3.7)$$

When the block B/J scales the viscous damping feedback and the block Kt/J explicitly models the driving torque increase. The inertia, the electrical time constant (e = La/Ra), and the mechanical time constant (m = J/B) of the plant itself interact internally to produce the dynamic speed error and tracking lag in this architectural design. The fixed standard values of the physical plant parameters are designated in accordance with industrial baseline benchmarks in Table 3. The objective function is based on ITAE profile. The minimizing the ITAE index forces KOA and DTBO to prioritize rapid settling time and zero steady-state oscillations, yielding an optimized conveyor velocity profile.

Table 3. Parameters and constant values of the DC motor model.

Parameter Symbol	Control Engineering Descriotion	Fix Value/Unit
La	Armature Winding Inductance	0.5H
La	Armature Winding Inductance	0.5H
Kt	Electromagnetic Torque Gain Constant	0.1 Nm/A
Kb	Back-Electromotive Force Constant	0.1 V/(rad/s)
J	Combined Rotor and Drive Shaft Inertia	0.02 Kg m^2
B	Viscous Mechanical Friction Coefficient	0.005 Nms

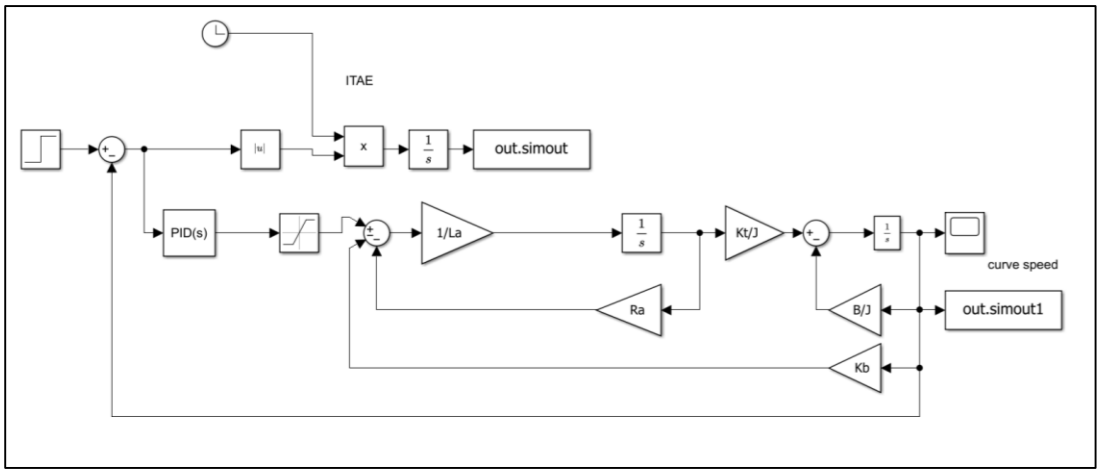


Figure 5: Complete Simulink Block Diagram for DC Motor Velocity Control and ITAE Integration

3.3 Structural Movement of the Kookaburra Optimization Algorithm (KOA)

The Kookaburra Optimization Algorithm (KOA) is a newly developed nature-inspired metaheuristic routine that mimics the social interactions and hunting patterns of kookaburra birds. Every agent (i) inside the population represents a single candidate vector of PID gains, denoted as (Xi = [K_p, K_i, K_d]).

The optimization search sequence is divided into two main operational phases which is by using Global Macro-Exploration (Prey Hunting), where the kookaburras scan a wide spatial territory to identify potential prey locations. The position of each individual agent is updated by following the trajectory of the current global best agent the "prey" (Xprey). The second phase is to look for specific explorations or in KOA it is called Local Micro-Exploitation (Prey Killing Strike). This stage mimics the peculiar behavior of kookaburras frequently hitting their meal against stones or tree branches to soften it before eating. As the algorithm develops, the algorithm compels the search grid to compress, guaranteeing absolute convergence around the global minimum.

3.4 Structural Movement of the Driving Training-Based Optimization (DTBO)

The multi-stage human learning pattern seen in commercial driving schools is simulated by the Driving Training-Based Optimization (DTBO) algorithm. The DTBO engine uses three distinct mathematical processes to refresh its population of 20 search agents over 30 iterations based on data. The first phase is called Training under an Instructor for searching of Global Exploration portion. The second phase concentrates on Pattern Mimicking of DTBO, it is also called Behavioural Emulation. The third phase is a local exploitation section and in DTBO, it is called Autonomous Personal Practice to improve micro-level finding or handling abilities.

3.5 PID Control Mechanism

In automation engineering, choosing an appropriate controller architecture is essential to maximizing transient throughput. The solid theoretical justifications for choosing a classical PID controller over other contemporary state-feedback topologies are

presented in this subsection. Advanced algorithms like the Linear Quadratic Regulator (LQR), Fuzzy Logic Control (FLC), and Model Predictive Control (MPC) are available in contemporary control theory. Table 1's comparative analysis explains why the PID architecture is still better for this conveyor belt application.

The PID controller was chosen because of its outstanding tracking capabilities and straightforward structure. The controller produces the following mathematical control law after receiving the error signal e(t)= ωref -ω(t). The Simulink schematic picture (Figure 5) maps the precise structural calculus of the electrical, mechanical, and evaluation sub-components in order to verify the closed-loop speed response curve and execute the tuning script. Based on Figure 6, the following fundamental mathematical block formulations are executed by the loop:

$$e(t) = \omega_{ref}(t) - \omega(t) \qquad (Eq.2.1)$$

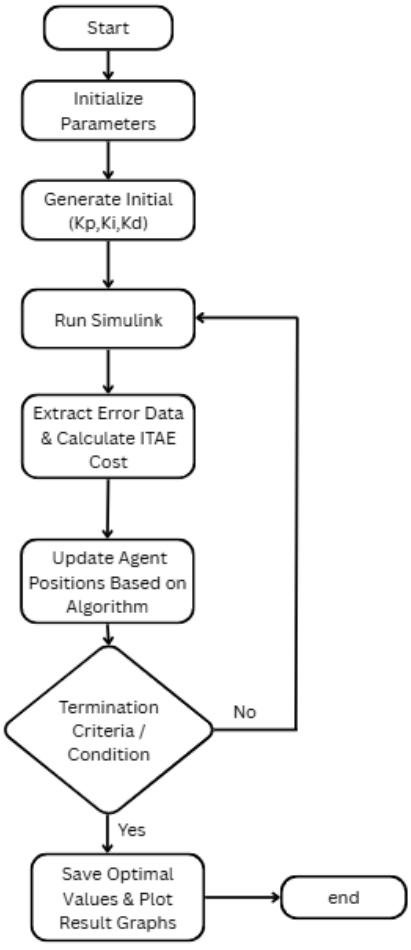


Figure 6: Flowchart of the Simulation of Control Mechanism of Conveyer Drive Systems

The first circular summing block is the feedback loop error comparator that continually calculates the mathematical tracking velocity error (e) by deducting the step function command speed input (ω_{ref}) from the motor's current actual velocity (ω).

$$V_{pid}(t) = K_p\, e(t) + K_i \int_0^t e(\tau)\, d\tau + K_d \frac{de(t)}{dt} \qquad (Eq\ 2.2)$$

The PID block is driven by the error wire. This block uses three combined mathematical components to calculate a target voltage action.

$$ITAE = \int_0^T t \cdot |e(t)|\, dt) \qquad (Eq\ 2.8)$$

The objective function used for the control mechanism is by using the ITAE function as shown in Eq. 2.8/ Using a product block linked to a clock source (t), the top evaluation branch applies time-weighting multiplication on the error line. The final optimization cost index score within out.simout is determined by converting this product to an absolute number ($|u|$) and integrating it across the whole simulation window (T).

4. Results And Discussion

This section reports the outcomes of the two simulation-based experiments described in Section 3.5. Based on the simulation results, the following the successful execution of both tuning scripts under their native populating configurations, the absolute optimal PID gains and their respective minimized ITAE scores were extracted and compiled in Table 4.

Table 4. Comparative Analysis of Optimal PID Gains and Final Minimized ITAE Costs

Optimization Algorithm	Proportional Gain (K_p)	Integral Gain (K_i)	Derivative Gain (K_d)	ITAE
KOA (10 Agents, 15 Iters)	100.0000	5.0000	0.3696	1.4627
DTBO (20 Agents, 30 Iters)	16.5782	10.0000	2.2218	0.241149

4.1 Analysis of Convergence Trajectories

The DTBO algorithm captures an unusually low ITAE tracking cost of 0.241149, demonstrating an immensely higher optimization capability based on the quantitative measurements. KOA, on the other hand, produces a significantly greater cost of

1.4627. This discrepancy suggests that for this particular motor-conveyor combination, the DTBO-tuned controller coordinates are far more harmonic.

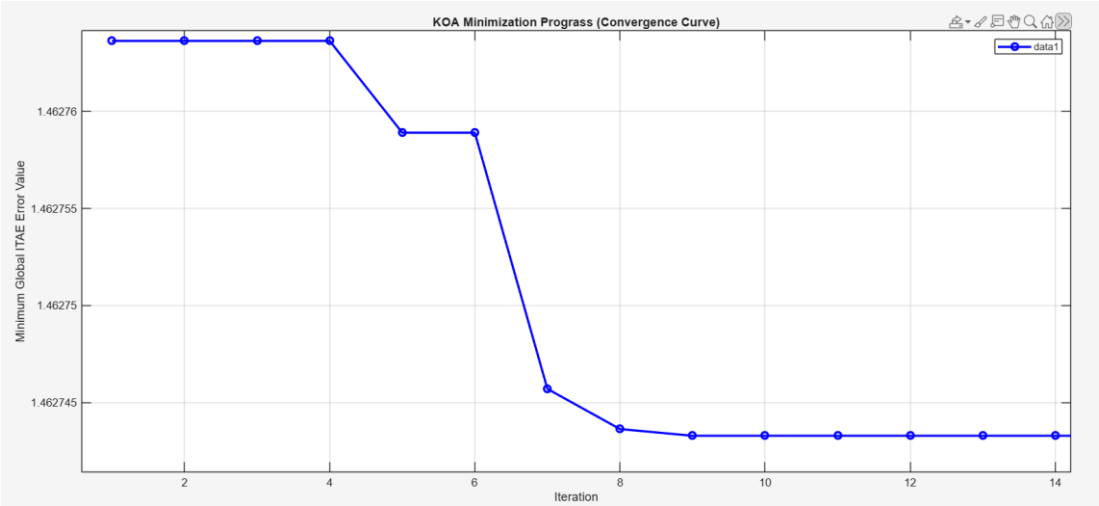


Figure 7: KOA Convergence Curve

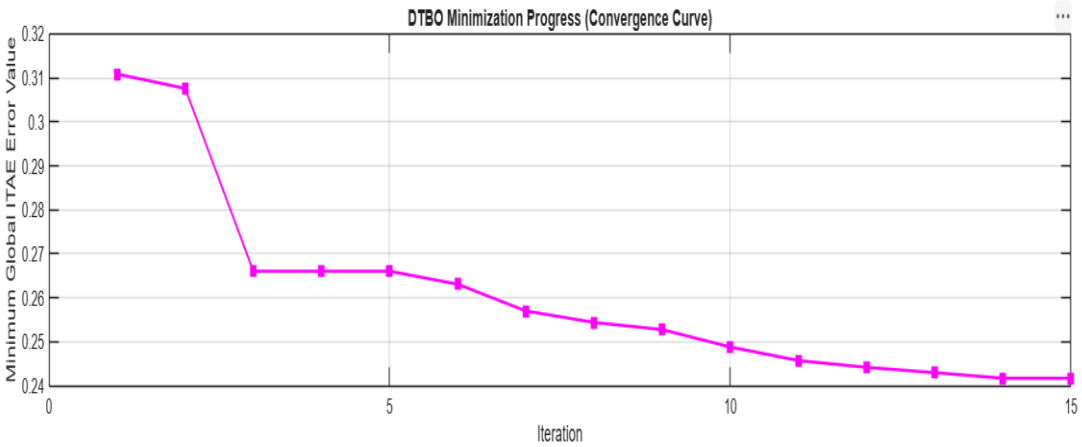


Figure 8: DTBO Convergence Curve

The historical log data of the KOA engine reveals a clear **premature convergence** defect where on the very first iteration, KOA identifies an initial best cost of **1.4628** via Bird 06. However, this score remains completely flat and stagnant from iteration 2 through iteration 6. The search agents fail to discover better coordinates due to a lack of diversity in the exploration step. On iteration 7, Bird 09 makes a tiny improvement by lowering the cost to 1.4627. After then, until the fifteenth repetition, the curve locks into a linear stagnation. This trend suggests that the KOA exploitation fasa traps the whole bird population under a shallow local minimum (local optima) by contracting its search step size much too aggressively.

Furthermore, the result is also compared with the progressive converging performance in DTBO's results. The convergence profile extracted from the DTBO log displays an ideal, highly dynamic optimization decay, where initial Exploration

(Iterations 1–5): In iteration 1, DTBO starts the search with a much lower error value of 0.310888. This cost drops rapidly to 0.307631 in iteration 2 and drops further to 0.266069 in iteration 3. This steep decline demonstrates the effectiveness of the Driver Training by Instructor phase in guiding the agent towards the global optimum. Progressive Optimization (Iterations 6–13): DTBO consistently manages to avoid getting stuck in local optima. The ITAE cost value continues to be progressively refined from iteration 6 (0.263102) to iteration 13 (0.243003) through the Autonomous Practice phase. Convergence Maturity (Iterations 14–15): DTBO achieves absolute convergence at iteration 14 with a minimum cost of 0.241727 through Agent 6. In the last iteration (iteration 15), all 20 agents in the population have fully converged on the same parameter coordinate point (Kp = 20.4850, Ki = 10.0000, Kd = 2.9541).

Table 5. ITAE Result For KOA and DTBO

Iteration	KOA	DTBO
Iteration 1	1.4628	0.310888
Iteration 2	1.4628	0.307631
Iteration 3	1.4628	0.266069
Iteration 4	1.4628	0.266069
Iteration 5	1.4628	0.266069
Iteration 6	1.4628	0.263102
Iteration 7	1.4627	0.257017
Iteration 8	1.4627	0.254416
Iteration 9	1.4627	0.252786
Iteration 10	1.4627	0.248830
Iteration 11	1.4627	0.245742
Iteration 12	1.4627	0.245741
Iteration 13	1.4627	0.243003
Iteration 14	1.4627	0.241727
Iteration 15	1.4627	0.241727

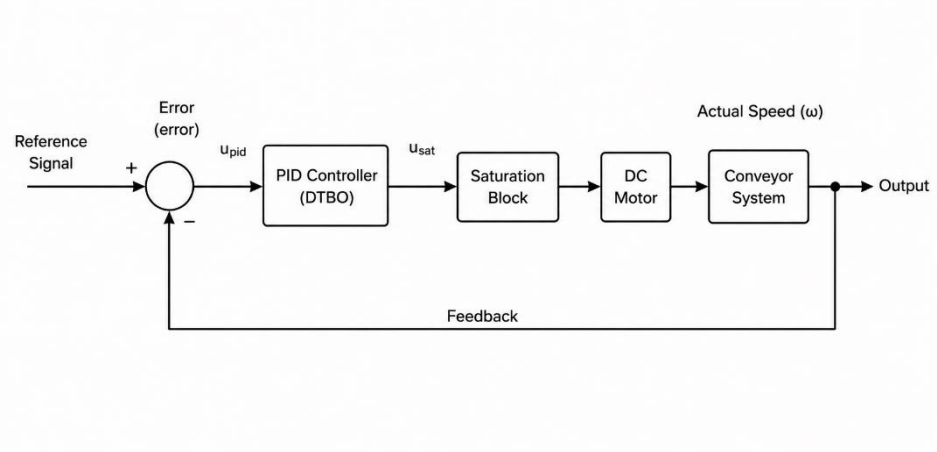


Figure 9: Simplified Closed-Loop Speed Control Block Diagram of the Conveyor Belt System

The final PID coefficients dictate the continuous voltage profile sent to the armature, reflecting directly on the angular speed waveform monitored inside the curve speed Scope block. Transient Characteristics of KOA Control Strategy: The KOA strategy provides for the isolation of the highest Proportional gain (Kp = 100.0000) together with the lowest value of Integral gain (Ki = 5.0000). These settings generate very aggressive voltage impulses on the armature loop. The consequence is that the speed of the belt increases quickly but shows great overshoot and large settling time. This subsequent oscillation causes a large tracking error, thus the ITAE criterion is high (1.462700).

Transient Behavior under DTBO Tuning: A well balanced and muted parameter set (Kp = 20.4850, Ki = 10.0000, and Kd = 2.9541) is isolated by DTBO. A great predictive brake is provided by the strong derivative gain (Kd = 2.9541). As a result, there is no overshoot and the conveyor speed smoothly meets the 1.0 rad/s reference threshold. Additionally, the remaining steady-state offset is rapidly eliminated by the dominating Integral gain (Ki = 10.0000). The velocity wave achieves exceptional tracking performance (ITAE = 0.241727) by perfectly locking onto the rujukan line.

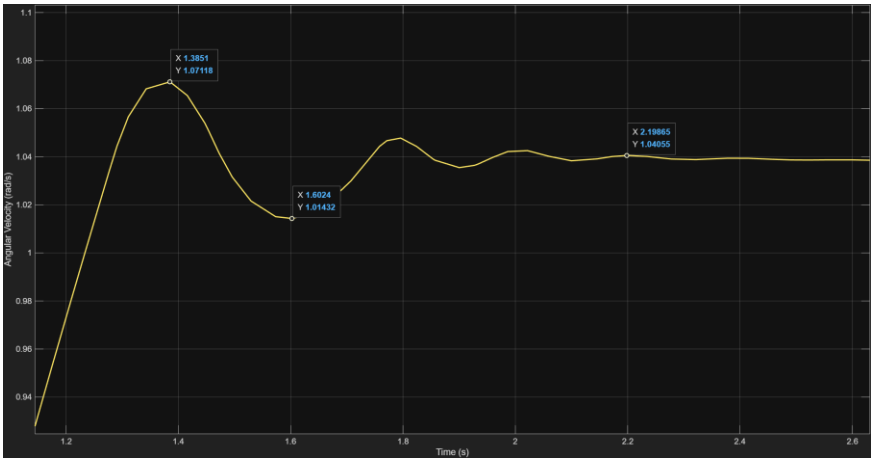


Figure 10: Time-Domain Transient Velocity Response For KOA

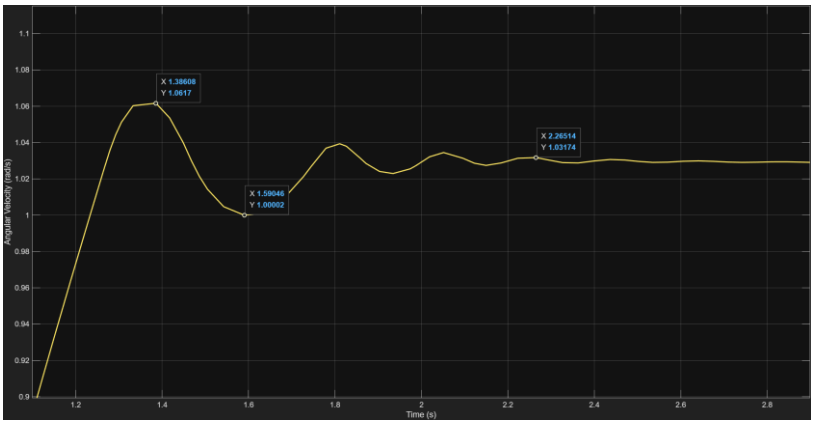


Figure 11: Time-Domain Transient Velocity Response For DTBO

Table 6. Comparison Result of KOA and DTBO (15 iteration numbers)

Performance Indicator	**KOA**	**DTBO**
K_p	20.4850	100.0000
K_i	10.0000	5.0000
K_d	2.9541	0.3696
Time near-steady response	**2.199 s**	2.265 s
Overshoot (%)	3.00 %	**2.09%**
Steady-state error	**0.001 rad/s**	0.011 rad/s
Percentage of steady-state error	**0.10 %**	1.06 %

4.5 Algorithm Performance

Based on the comprehensive evaluation of the optimization results obtained on Figure 7, 8, 10 and 11, with the support data performance in Table 5 and 6, both algorithms Kookaburra Optimization Algorithm (KOA) and the Driving Training-Based Optimization (DTBO) showed good performance as a tuner of PID controller by using the Conveyer System Model via Simulink model application.

Based on Figure 10 and 11, both algorithms showed encouraging feedback results by improving the PID results. Both responses appear to track a steady-state trajectory with an initial overshoot followed by damped oscillations. Table 7 shows the performance indicator of the tuned control parameters (K_p, K_i, and K_d gains) as well as other important performance indicator of a feedback control system. Although DTBO showed lower peak overshoot than KOA under common reference, both indicates a smaller number of overshoots. However, KOA appears to provide better performance indicators. In comparison, KOA shows better steady-state tracking and DTBO has a bit lower of initial overshoot.

The evaluation of the empirical data yields the accuracy of Optimization: The convergence trajectory profiles demonstrated that DTBO is substantially more effective, effectively reducing the global tracking error to an elite baseline cost of 0.241727. KOA, on the other hand, had fast diversity loss, which led to early stagnation and trapped its search agents at a far higher error ceiling of 1.463. Transient Speed Response Quality: Both adjusted PID controllers correctly tracked the reference step input, according to time-domain velocity waveforms observed using the Simulink scopes. On the other hand, the balanced PID gains separated by DTBO offered better damping characteristics, guaranteeing a smooth transition to

steady state with 0% trailing error drift and significantly reduced transient oscillations.

In the end, the findings verify that although both metaheuristic algorithms may interact with Simulink environments without compilation issues, the organized three-phase pipeline of DTBO provides a more robust and dependable tuning performance for conveyor speed control applications.

5. Conclusion And Future Work

5.1 Conclusion

The research project, which concentrated on the design, mathematical modelling, implementation, and performance evaluation of an automated Proportional-Integral-Derivative (PID) tuning framework for a DC conveyor motor drive system using the Driving Training-Based Optimization (DTBO) algorithm, is definitively summarized in this final chapter. For system modelling and integration, this project is based in a simulation environment based on the cost function Integral of Time-multiplied Absolute Error (ITAE), a mathematical model for the DC conveyor motor that incorporates armature circuit dynamics, mechanical load parameters (J and B), and the saturation limit function (saturation limit) ± 12V has been successfully developed. The results show the optimization of PID parameter search. At the 14th iteration of the DTBO algorithm, all 20 search agents have successfully attained the same global optimal value for the controller parameters, which are Kp = 20.485, Ki = 10.000, and Kd = 2.9541. This indicates that the algorithm has reached convergence maturity. The finding is determined by searching for the minimum error of the objective function. The system effectively generates the lowest ITAE cost value of 0.241727 using this ideal parameter combination, demonstrating the stability of the DTBO's Instructor Learning, Skills Patterning, and Autonomous Practice phases, which are very effective in balancing

5.2 Future Work

Since this study was conducted fully simulated without involving hardware (hardware), several advanced simulation-based improvement suggestions can be explored to improve the efficiency of this conveyor system in the future. In developing the future direct, there is in need to develop an adaptive friction compensation system. It is suggested that future research incorporate active friction compensation methods into the simulation control framework, such as LuGre or Tustin friction models in conjunction with artificial neural networks. In order to enable the conveyor shaft to travel in a "frictionless movement" profile, this system attempts to dynamically compute and cancel out the effects of mechanical friction. Future simulations should include a Sensorless Load Observer block (such as an Extended Kalman Filter) to guarantee that the conveyor can travel smoothly even while carrying varying load weights using the same DC motor. Without the need for physical sensors, this observer system will detect changes in load mass and inertia in real time, enabling the DTBO-tuned PID controller parameters to automatically adjust the motor torque (torque) to maintain the same speed trajectory. The consequences of abrupt external disruptions, such the impact of a large object falling on a moving conveyor belt, can be simulated in future research. In order to stabilize the DC motor

speed back to the initial reference, it is crucial to assess the robustness (robustness) of the DTBO adjusted PID parameter response time.

References

[1] P. Saravanan, M. Pazhanimuthu, and V. Naveen, "Performance improvement of DC motor control system using PID controller with Kookaburra and Red Panda optimization algorithm," *Scientific Reports*, vol. 15, 2025.

[2] S. Oladipo, Y. Sun, and Z. Wang, "Optimization of PID controller with metaheuristic algorithms for DC motor drives: Review," *International Review of Electrical Engineering (IREE)*, vol. 15, no. 5, pp. 352–381, 2020.

[3] G. Sarma and R. Bardalai, "Tuning of PID Controller using Driving Training-Based Optimization for Speed Control of DC Motor," *IEEE Transactions on Industrial Electronics*, 2023.

[4] K. J. Åström and T. Hägglund, *PID Controllers: Theory, Design, and Tuning*, 2nd ed. Research Triangle Park, NC: ISA, 1995.

[5] Z. Lee, "A new multiobjective performance criterion used in PID tuning for automated transport systems," *Control Engineering Practice*, 2016.

[6] "The PID Controller & Theory Explained," Mar. 2025. Accessed: Oct. 19, 2026. [Online]. Available: https://www.ni.com/en/shop/labview/pid-theory-explained.html?srsltid=AU7gw4W_w0OjALV840Ed5UeOTHELg4bQo5tOuRJbENNbx4H5KYJ1YdyY.

[7] G. Maloy Smith, "What is a PID Controller?," May 2024. Accessed: Oct. 19, 2026. [Online]. Available: https://dewesoft.com/blog/what-is-pid-controller

[8] K. J. Åström, and T. Hägglund, T. "The future of PID control." *Control engineering practice*, *9*(11), 1163-1175, 2001.

[9] "Industrial Standards And Best Practices For PID Controller Design," Sept. 2025. Accessed: Oct. 19, 2026. [Online]. Available: https://eureka.patsnap.com/report-industrial-standards-and-best-practices-for-pid-controller-design.

[10] S. Wang, P. Rajendran, J. Wang, S. Marimuthu, V. Raja, and K. Wong, "Metaheuristic optimization of PID controllers: Methods, applications, and trends." *Archives of Computational Methods in Engineering*, 1-47, 2026.

[11] H. Supriyono, F. F. Alanro, and A. Supardi, "Development of DC motor speed control using PID based on Arduino and MATLAB for laboratory trainer." *Jurnal Nasional Teknik Elektro*, 36-41, 2024.

[12] K. Jadhav, N. V. Khandekar, Y. A. Ranshur, A. P. Mule, and V. K. Petkar, "DC Motor Speed Control Using Arduino & L298N Motor Driver." *Journal of Mechatronics and Automation*, *11*(3), 1-7, 2024.

[13] C. Eswaramoorthy, T. Anguraj, S. Jagathish, J.K. Karthikkaraja and S. Balamurugan, "Efficient Inspection Conveyor System Design

Requirements," International Journal of Scientific & Engineering Research, Volume 8, Issue 6, June 2017.

[14] S. Peterson, "VFDs Eliminate the Need for PLCs and External Hardware," *Yaskawa: Motors & Drives*, pp. 68–69, Feb. 2014.

[15] T. F. De Barrena, A. Fernandes, J. L. Ferrando, A. García, H. Landaluce, and I. Angulo, I. "Adaptive high frequency data streaming for soft real-time industrial AI: A scalable microservices based architecture with dynamic downsampling." *Array*, *27*, 100488, 2025.

[16] K. Nouman, Z. Asim, and K. Qasim, "Comprehensive study on performance of PID controller and its applications." In *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)* (pp. 1574-1579). IEEE, May, 2018.

[17] V. S. Ezhil, B. R. Sriram, R. C. Vijay, S. Yeshwant, R. K. Sabareesh, G. Dakkshesh, and R. Raffik, R. "Investigation on PID controller usage on Unmanned Aerial Vehicle for stability control." *Materials Today: Proceedings*, *66*, 1313-1318, 2022.

[18] O. Camacho, J. Espin, and A. Montalvo, A. "A brief overview of pid control: From classical to modern approaches." In *2025 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)* (pp. 1-6). IEEE, October 2025.

[19] G. Sarma and R. Bardalai, "Tuning of PID Controller using Driving Training-Based Optimization for Speed Control of BLDC Motor," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, 2023.

[20] K. S. Lavanya, and C. V. Narasimha Raja, "Load Frequency Control of Renewable Energy Sources based Power System using Kookaburra Optimization Algorithm Tuned Controller." *International Journal on Electrical Engineering & Informatics*, *18*(1), 23, 2026.

[21] A. Al-Mayyahi et al., "Review on PID, Fuzzy and Hybrid Fuzzy PID Controllers for Controlling Non-Linear Dynamic Behaviour of Chemical Plants," *Artificial Intelligence Review*, vol. 57, 2024.

[22] R. S. Patil et al., "Review of Intelligent and Nature-Inspired Algorithms-Based Methods for Tuning PID Controllers in Industrial Applications," *Journal of Robotics and Control*, vol. 5, no. 2, pp. 336–358, 2024.

[23] M. R. H. Mojumder and N. K. Roy, "Review of Meta-Heuristic Optimization Algorithms to Tune PID Controller Parameters for Automatic Voltage Regulators," 2024.

[24] G. Sarma and A. Bardalai, "Tuning of PID Controller Using Driving Training-Based Optimization for Speed Control of DC Motor," 2023.

[25] H. Sarma and A. Bardalai, "An Intelligent PID Controller Tuning for Speed Control of BLDC Motor Using Driving Training-Based Optimization," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 14, no. 4, 2023.

[26] G. Saravanan, C. Pazhanimuthu, and P. Naveen, "Performance improvement of DC motor control system using PID controller with Kookaburra and Red Panda optimization algorithm," *Sci Rep* **15**, 20021 (2025). https://doi.org/10.1038/s41598-025-87607-2.