International Journal of Science and Research Archive

eISSN: 2582-8185

Cross Ref DOI: 10.30574/ijsra

Journal homepage: https://ijsra.net/

(Research Article)

Adaptive control optimization using NLTA algorithms for mechatronic systems.

Olakunle Abimbola Kumuyi *

Department of System engineering, College of Technology, Architecture and Applied Engineering, Bowling Green, Ohio, United State.

International Journal of Science and Research Archive, 2025, 14(02), 646-659

Publication history: Received on 01 January 2024; revised on 04 February 2025; accepted on 07 February 2025

Article DOI: https://doi.org/10.30574/ijsra.2025.14.2.0416

Abstract

The Non-Linear Threshold Accepting (NLTA) algorithm has been successfully applied in electrical and electronic systems, particularly in optimizing power distribution and voltage regulation. However, its application in mechatronic systems remains largely unexplored. Given that most Unmanned Ground Vehicles (UGVs) and robotic systems used in logistics and industrial environments integrate electrical, electronic, and mechanical subsystems, an advanced adaptive control strategy is essential to ensure optimal performance in dynamic environment be it in engineering/manufacturing or last-mile delivery environments. Traditional Proportional-Integral-Derivative (PID) controllers, while widely adopted, can still be improve for real-time adaptability, leading to improved efficiency in trajectory control, response time, and energy consumption. These possibilities necessitate needs for a more robust control framework with better and improved capabilities of dynamically adjusting to operational uncertainties.

In this research a comparative performance evaluation was conducted using Mathematical models and MATLAB/Simulink simulations, benchmarking the NLTA algorithm against conventional PID controllers and other heuristic optimization techniques such as Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC), and Differential Evolution (DE). Unlike some of the traditional PID controllers that require manual tuning and often fail to adapt to non-linear system variations, NLTA employs an adaptive threshold mechanism to iteratively optimize control gains, ensuring improved operational stability, energy efficiency, and trajectory accuracy. The NLTA algorithm's ability to better self-adjust in real time provides a significant advantage over existing control methods, making it a better and more viable alternative for enhancing the performance and reliability of mechatronics or other systems in need of optimization.

A comparative analysis was conducted using MATLAB/Simulink simulations to benchmark NLTA against conventional PID controllers and other heuristic-based optimization methods, including Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC), and Differential Evolution (DE). The results demonstrate that NLTA outperforms traditional control strategies by achieving faster response times, reduced settling time, enhanced robustness against environmental disturbances, and improved overall system efficiency. While the NLTA model also holds promise for warehouse layout optimization where dynamic reconfiguration could enhance operational efficiency this research focuses solely on evaluating its effectiveness as a control strategy for mechatronic systems compared to existing PID-based approaches.

The findings reinforce the potential of NLTA as an advanced control framework for system and operational optimization also, bridging the gap between electrical, electronic, and mechanical control integration. Future work will explore real-world deployment, AI-driven predictive modeling for enhanced adaptability, and the extension of NLTA's capabilities to logistics facility layout optimization. By validating NLTA's effectiveness against traditional PID controllers, this research is aimed to contribute to the ongoing evolution of intelligent control mechanisms in logistics engineering especially autonomous last-mile delivery systems.

* Corresponding author: Kumuyi Olakunle

Keywords: Adaptive Control; Non-Linear Threshold Accepting (NLTA); Proportional-Integral-Derivative (PID) Control; Adaptive Control Systems; Particle Swarm Optimization (PSO); Artificial Bee Colony (ABC) Algorithm; Differential Evolution (DE) Algorithm; Real-Time Optimization; Threshold Acceptance Mechanism

1. Introduction

Modern logistics environments rely heavily on mechatronic systems, which integrate electronics, electrical, computation, and mechanics in a synergistic manner. These systems, which are essential to supply chain optimization, include robotics, drones, electronic vehicles, and automated industrial processes like automatic loading systems. The efficient transportation of people, products, and services from the point of origin to the final consumer is the fundamental function of logistics, which successfully closes the gap between production and consumption [1]. For instance, robotic systems are used in warehouses for automated picking and packaging, which greatly lowers human error and boosts productivity [2]. In a similar vein, drones and electronic vehicles are transforming delivery procedures by improving last-mile delivery speed and flexibility, which is essential for urban logistics where conventional delivery vans encounter limitations [3]. Also, automated industrial systems, such as automated guided vehicles (AGVs) and conveyor belts, optimize throughput times and lower labor costs by streamlining the movement of commodities throughout facilities, from sorting to storage and retrieval. These mechatronic devices rely on control systems for accurate management and performance. Advanced instruments would not be able to carry out their duties consistently in the absence of efficient control systems. Control systems give operators the ability to direct system operations, track system performance, and establish operational boundaries, guaranteeing that the machinery runs within safe and effective bounds.

In addition to improving operational efficiency, the incorporation of sophisticated control systems into mechatronic devices also makes logistics operations more flexible by enabling real-time reactions to environmental and operational changes. Given the fluctuating demand patterns and ever rising expectations for service delivery in modern logistics, this flexibility is essential. In addition to those generalized points certain main points most be re-emphasize.

- Mechatronic systems in dynamic and complex logistical environments must interact safely and effectively with human operators, necessitating highly responsive and adaptive capabilities. Such systems, including robots and automated guided vehicles (AGVs), operate in conditions with varied lighting, diverse loads, and unpredictable human behavior, requiring advanced sensing and control to navigate these challenges successfully. Real-time data processing and adaptability are crucial, as developments in artificial intelligence and machine learning allow these systems to instantly adjust operations, enhancing both performance and safety [4], show that responsive mechatronic systems can significantly reduce error rates and increase throughput by dynamically adapting to human movements and operational changes. learning. For instance, by examining data patterns, adaptive algorithms can anticipate possible interruptions and enable machines to make proactive adjustments to their operations [5].

1.1. Adaptive Mechatronic Systems in Dynamic Logistic Environments

Mechatronic systems designed for logistics operations are deployed in environments that are inherently complex and constantly evolving. These settings are characterized by high levels of activity and require seamless interactions between automated systems and human operators, each significantly influencing the other's efficiency and safety [6]. For instance, robots and automated guided vehicles (AGVs) must interact closely with human workers, necessitating highly responsive systems that can adapt in real time to avoid accidents and ensure smooth operations. A study on human-robot collaboration in warehouse environments demonstrated that responsive mechatronic systems could significantly reduce error rates and increase throughput by dynamically adjusting to human movements and operational changes [6]. Logistics environments often consist of varying conditions that can change rapidly, such as fluctuating lighting conditions, diverse load types, and unpredictable human behavior. To effectively navigate these complexities, mechatronic systems are equipped with sophisticated sensing and adaptive control capabilities. These advanced technologies enable the systems to accurately perceive their environment and react appropriately, thus playing a crucial role in maintaining operational continuity [7]. The capability to process information swiftly and accurately is critical for the optimal performance of mechatronic systems in logistics. Real-time data processing allows these systems to make immediate adjustments to their operations, which enhances both performance and safety. Recent advancements in machine learning and artificial intelligence have further bolstered the adaptability of these systems. For example, adaptive algorithms are now capable of predicting potential disruptions by analyzing data trends, thus allowing machines to proactively adjust their operations [8].

Moreover, the integration of cutting-edge technologies such as the Internet of Things (IoT) and cloud computing into mechatronic systems facilitates enhanced communication and data sharing. This technological integration supports more cohesive system operations and allows for more granular control over individual components, leading to improved system responsiveness and operational efficiency in dynamic logistics settings [9], through these integrated technologies and adaptive strategies, mechatronic systems in logistics not only meet the challenges of their complex environments but also set new standards for efficiency and safety in industrial operations.

1.2. Introduction to the Non-Linear Threshold Accepting (NLTA) Algorithm

An important development in the realm of control systems is the Non-Linear Threshold Accepting (NLTA) algorithm, which provides a fresh method of getting around the drawbacks of conventional Proportional-Integral-Derivative (PID) controllers. The NLTA algorithm was created by Nabil Nahas, Mohammed Abouheaf, Mohamed Noomane Darghouth, and Adel Sharaf. It was first used in electrical systems, where it outperformed traditional PID techniques in controlling voltage systems. As mechanical systems are essential to mechatronics, its efficacy in these applications demonstrated its potential for wider applications in a variety of optimization and control systems [10].

How NLTA Works: The NLTA algorithm diverges from traditional PID control methods by incorporating a structured, iterative process that enhances system response and stability under dynamic conditions. Here’s a detailed breakdown of how the NLTA algorithm optimizes control systems [7]:

- Step 1: Start with initial Ω_0 and relatively high Ω.
- Step 2: Indicate the allowed number of total iteration steps (Υ) and the discount value $\Delta\Omega$.
- Step 3: Pick initial values for the control PID gains R^0_c (i.e., K_p, K_i, and K_d) and let $R_c = R^0_c$
- Step 4: Loop (while the current iteration is < Υ):
  - Choose a neighboring solution $R`_c$ from the feasible space of the current one (R_c).
  - Evaluate the objective functions $\Psi(R_c)$ and $\Psi(R`_c)$.
  - Calculate $H` = \Psi(R`_c) / \Psi(R_c)$ and evaluate $H(\Omega) = 1/ \left(1 + (\Omega/\Omega_0)^2\right)^{1/2}$
  - If ($H` \leq 1/H(\Omega)$) or ($H` \leq 1$) then update $Rc = R`_c$ and accept that solution.
  - To control the convergence speed of the search process, adjust Ω by fixed rate $\Delta\Omega$ such that $\Omega = \Omega - \Delta\Omega$.
- Step 5: Terminate the loop upon reaching the capacity Υ.

1.3. Objectives and Scope

The primary objective of this research is to investigate the efficacy of Non-Linear Threshold Accepting and its application to advanced control systems in dynamic environments by comparing the Non-Linear Threshold Accepting (NLTA) algorithm with already widely used PID algorithms. This involves a thorough analysis of traditional PID controllers in comparison to the (NLTA), highlighting their performances in real-time adaptability, efficiency under varying operational conditions, and overall impact on system performance. The report outlines a structured approach to designing and integrating NLTA and the to be compared algorithms into existing intelligent control systems, developing simulation models for validation, and formulating an implementation plan for real-world logistics operations. Additionally, it evaluates the enhancements in system performance, including improved response precision, reduced maintenance, optimized energy consumption, and lower operational costs. Beyond logistics, the research explores the broader applicability of NLTA in fields such as manufacturing, robotics, automotive, and aerospace, emphasizing its scalability and adaptability. Expected outcomes include increased productivity, reduced error rates, and cost efficiency, establishing NLTA as a viable alternative to traditional PID controllers in various industrial applications. The report also provides a comparative analysis between other well established PID controllers and NLTA, incorporating case studies to highlight performance differences and real-world implications. Key design methodologies, including algorithm tuning and system configuration, are outlined alongside detailed simulation protocols to validate NLTA's effectiveness under diverse operational scenarios. Findings from simulations will be presented with statistical analysis and performance benchmarks, demonstrating efficiency improvements such as reduced energy consumption and lower maintenance needs. The study primarily focuses on mechanical systems within controlled environments, with results derived from simulations and literature reviews, acknowledging potential real-world complexities.

This research contributes significantly to adaptive control systems, with broad industrial implications and future research potential. By showcasing NLTA’s effectiveness in real-time dynamic environments, the study advances control technology, potentially setting new industry benchmarks for performance and efficiency in logistics, manufacturing, and mechanical control systems. The ability of NLTA to dynamically adjust control parameters makes it particularly valuable in manufacturing automation, robotics, and precision-driven industries such as automotive and aerospace, where responsive control strategies are critical. Additionally, the research paves the way for interdisciplinary applications, merging NLTA with artificial intelligence and machine learning to develop autonomous, predictive control systems. Future studies could explore the scalability of NLTA across different platforms, its integration challenges, and applications in layout optimization and price-inventory control analysis. Furthermore, assessing the economic benefits and environmental impact of NLTA-enhanced systems could provide deeper insights into its advantages beyond technical improvements.

2. Literature Review

2.1. Traditional Control Systems

Traditional control systems like Proportional-Integral-Derivative (PID) controllers have been foundational. These systems are renowned for their robustness and simplicity, making them a staple in various automated processes across industrial operations. PID controllers operate by calculating an error value as the difference between a desired setpoint and a measured process variable. They then apply a correction based on proportional, integral, and derivative terms, hence their name [11].

The development of PID controllers' dates to the early 20th century and has evolved significantly to adapt to the increasing complexity of industrial needs. These systems have been instrumental in applications ranging from simple house appliances to complex automated manufacturing lines. Their ability to maintain control over a process without needing detailed insights into the underlying process makes them highly valuable in a fast-paced, output-driven environment like logistics [12].

2.2. Challenges and Limitations

Despite their widespread use, traditional control systems face significant challenges when deployed in dynamic and complex logistic environments. One major limitation is their reliance on predefined settings and parameters, which do not adapt well to the changing conditions typical of modern logistics operations. For instance, in a high-volume warehouse, conditions such as variable package sizes, fluctuating load weights, and unpredictable throughput rates can render traditional PID control less effective. These systems often struggle with non-linear dynamics where feedback errors do not sufficiently inform the necessary adjustments. This can lead to performance issues such as overshoot, where the controller exceeds its target, or prolonged settling times, which delay the stabilization of the system after a disturbance [13]. Traditional control systems are not inherently equipped to learn from past behaviors or anticipate future states, which are capabilities increasingly required in automated and smart logistics systems. As logistics operations become more integrated with real-time data analytics and predictive technologies, the inability of PID controllers to interface effectively with these advancements becomes a notable drawback. The exploration of these challenges in the literature underscores the need for more adaptive and intelligent control systems capable of coping with the complexities of modern logistics environments. This ongoing shift is marked by a growing reliance on systems that can dynamically adjust and optimize in response to real-time operational data, a frontier where advanced algorithms like the NLTA could play a transformative role.

2.3. Advanced Adaptive Control Systems

The idea of proportional control, which is essential to the creation of contemporary control systems, originated with Christiaan Huygens' groundbreaking 17th-century works and was further developed by James Watt in the 19th century [14]. The output of this early control method, which dates to the 1600s, is exactly proportionate to the error signal between the actual value and the intended setpoint. Because of their simplicity, proportional controllers work well in many situations; but, because they cannot eliminate residual error, they usually produce a steady-state error. This intrinsic restriction emphasizes the need for extra control measures in more intricate systems. Throughout the 20th century, the evolution of the standard PID controller was significantly advanced by numerous engineers, who built upon the foundational theories initially introduced by pioneers like Christiaan Huygens, James Watt, and Nicolas Minorsky. This period of enhancement particularly saw the development of specialized variants such as PI and PD controllers during the 1930s and 1950s, aligning with major advancements in automation technology [14] [13] [9].

2.3.1. PID Development Over Time

- Proportional Control: The concept of proportional control, crucial in the development of modern control systems, has its roots in the pioneering works of Christiaan Huygens in the 17th century and was further refined in the 19th century by James Watt. This early form of control, evident from the 1600s onward, provides an output that is directly proportional to the error signal between the desired setpoint and the actual value [15]. Although proportional controllers are effective for many applications due to their simplicity, they typically result in a steady-state error, as they lack the capability to eliminate residual error. This inherent limitation underscores the necessity for additional control actions in more complex systems.

- Integral Controller (I Controller): The integral controller, a key component of PID control, was significantly developed through the early 20th century by engineers including Minorsky. Recognized primarily in the 1920s, the integral controller enhances system control by focusing on the elimination of steady-state errors [16]. It achieves this by integrating the error signal over time, which accounts for the accumulation of past errors. This function is critical for ensuring that control systems not only reach but also maintain the desired setpoint accurately, correcting any ongoing discrepancies between the current output and the setpoint.
- Derivative Controller: The derivative controller, another integral aspect of PID control, was further explored and refined by engineering theorists in the early 20th century, including notable contributions from Minorsky [13] [9]. This control component emerged concurrently with other PID concepts during the 1920s. The function of the derivative controller is to produce an output that reflects the rate of change of the error, which helps in predicting future trends and behaviors of the system. By anticipating rapid changes in error, it aids in dampening oscillations and enhances the overall stability of the system. However, while derivative control is beneficial for improving system responsiveness and preventing overshoot, excessive use of this action can lead to increased noise and potential instability within the control loop.
- Standard PID Controller: The standard PID (Proportional-Integral-Derivative) controller, developed by Russian American engineer Nicolas Minorsky in 1922, effectively combines three control mechanisms: proportional, integral, and derivative [17]. The proportional element ensures the output is proportionate to the current error, the integral element integrates past errors to eliminate residual steady-state errors, and the derivative [17] element predicts and mitigates future errors based on the rate of error change. This tripartite control strategy enables the PID controller to maintain stable and responsive process control, adapting to changes in a gradual and precise manner.
- PI Controller: This model eliminates the derivative component, relying solely on proportional and integral actions to manage system control. It is particularly favored in applications where derivative action might exacerbate system noise, making it less desirable. The integral component helps eliminate steady-state error by integrating the error over time, ensuring the system remains accurate over prolonged periods.
- PD Controller: In contrast, the PD controller excludes the integral action and focuses on the proportional and derivative terms. This configuration is advantageous for processes that demand rapid responses, as the derivative action helps predict and correct future system behavior swiftly. While this model does not address steady-state errors as effectively as the PI controller, its ability to promptly respond to changes makes it invaluable in dynamic environments where speed is critical.

2.3.2. Examples of Digital PID Controllers

- Honeywell's DC1040 Controller Series: These controllers are known for their robustness and flexibility, commonly used in industrial environments where precise temperature, pressure, and humidity control are required. Honeywell's controllers integrate digital technology to provide enhanced accuracy and ease of use, making them popular in manufacturing and processing industries.
- Siemens SIMATIC Controllers: Siemens offers a range of controllers that use advanced PID functions. These controllers are notable for their scalability and integration into larger automation systems, often used in complex industrial operations that require precise motion control and process management.

These digital PID controllers represent a significant leap from their analog predecessors, primarily due to their ability to adapt dynamically to varying process requirements, thanks to embedded software that can analyze and modify control parameters in real time. This adaptability is critical in applications such as automated manufacturing lines and HVAC systems in large buildings, where conditions can change rapidly and require immediate system adjustments to maintain optimal operation [11]. These advancements underscore the evolution of PID technology from static, manually tuned systems to dynamic, automated solutions capable of self-optimization in response to environmental changes, thereby enhancing system performance and efficiency.

2.4. The Non-Linear Threshold Accepting (NLTA) algorithm

2.4.1. Origin and Development of NLTA

The difficulties posed by non-convex optimization problems, which are common in a variety of domains like engineering, logistics, and economics, led to the development of the NLTA algorithm. The metaheuristic optimization technique known as threshold accepting serves as the basis for NLTA. Initially, it was established as a feasible substitute for traditional optimization techniques [10]. Researchers like Nahas and colleagues, who have written extensively on the mechanics and applications of NLTA, have been essential in its development. Through their work, NLTA's adaptability and resilience in dynamic contexts were highlighted, demonstrating its efficacy in handling complicated optimization scenarios like economic dispatch in energy systems. [10]. Applying NLTA to solve non-linear and non-convex optimization problems has been the focus of pivotal research. By carefully addressing problems that were frequently handled poorly by conventional optimization techniques, the algorithm, for instance, has demonstrated great potential in power generation and schedule optimization [10]. The promise of NLTA as an advanced optimization tool is highlighted by these studies taken together.

Mechanics of NLTA

Threshold acceptance is a fundamental tactic in the NLTA's special mechanism of operation. This method enables the algorithm to "accept" alternatives that, mainly if they fall within a specified threshold range, may not be superior to existing solutions [10]. NLTA uses a two-step procedure: in the exploration stage, the algorithm assesses different possible solutions and determines how fit they are using predetermined standards. Solutions that meet or nearly meet the criterion are then accepted, even if they are more expensive than some of the current options. This expands the algorithm's search space by enabling it to avoid local optima [14].

Additionally, the non-linear feature enables the algorithm to dynamically modify the threshold, adjusting to the context of the search and producing more successful search tactics. In comparison to static threshold systems, NLTA's flexibility makes it especially helpful for complex and diverse problem domains, enabling superior optimization outcomes.

2.4.2. Comparative Studies

Significant performance disparities are seen when NLTA is compared to other adaptive algorithms and conventional PID controllers, especially when optimization tasks are involved. In one study [10], NLTA was assessed in conjunction with PID control techniques in a dynamic system that needed energy dispatch solutions. The findings showed that by successfully handling non-convex optimization problems, NLTA outperformed conventional PID controllers, resulting in quicker convergence times and increased system efficiency overall. The benefits of NLTA over other adaptive algorithms, like genetic algorithms and swarm intelligence techniques, were illustrated by another set of comparison assessments. According to research, NLTA continuously produced better results in terms of both quality and computing efficiency, particularly when dealing with situations that had nonlinear limitations and highly changing needs [10]. Additionally, NLTA's adaptive characteristics enable it to learn and exploit issue structures more effectively, resulting in optimal or near-optimal solutions in complex situations, according to a performance analysis comparing NLTA with more traditional methods [10]. Together, these investigations support NLTA's wider relevance, particularly in contemporary adaptive control systems.

Through its threshold acceptance mechanism, the Non-Linear Threshold Accepting (NLTA) algorithm employs a distinctive and calculated optimization strategy. Accepting solutions that fall within a certain threshold range but may not be better than the best available is essential to NLTA. Because it allows the algorithm to avoid local optima, which frequently impede more conventional optimization strategies, this methodology is essential. The two primary stages of NLTA's operations are acceptance and exploration. The program evaluates a range of alternative solutions during the discovery phase, determining their efficacy using predetermined standards. After that, it accepts solutions that reach or come close to the predetermined threshold, even if they are more expensive than some of the ones that are currently available. Because it broadens the algorithm's search capabilities beyond local answers to more globally optimal solutions, this procedure is very beneficial [10]The unique feature of NLTA is that it dynamically modifies the acceptance threshold in response to the non-linear characteristics of the problem being addressed. Because of its flexibility, NLTA may adjust its search approach to the particulars of the issue, producing more effective and efficient optimization results. In complicated and varied issue situations, NLTA's adaptability makes it incredibly successful, providing an optimization advantage over algorithms with fixed threshold settings.

2.5. Comparison of NLTA with Other Optimization Methods for PID Gain Optimization

The Non-Linear Threshold Accepting Algorithm (NLTA) is an optimization technique used for PID gain tuning, alongside methods like Particle Swarm Optimization (PSO), Differential Evolution (DE), and Artificial Bee Colony (ABC) optimization. PSO excels in simplicity and speed, balancing exploration, and exploitation [18], but is prone to local optima and parameter sensitivity. DE offers strong global optimization capabilities and simplicity, though it may converge slower and requires careful parameter tuning [11]. ABC is effective for exploration and robust against local optima but can also have a slower convergence rate and parameter sensitivity.

NLTA distinguishes itself through its unique thresholding technique, which enables adaptive control over search strategies, making it particularly effective in dynamic and complex environments. While PSO and DE may struggle with local optima and require meticulous parameter adjustments, and ABC may converge slowly, NLTA's thresholding technique addresses non-linear optimization challenges efficiently [11] [10] [9]. The choice of method depends on the specific optimization task's criteria, including the problem space's characteristics and the desired balance between exploration efficiency and solution accuracy.

3. Material and methods

To modify PID controller parameters and enhance system response, this project intends to optimize mechanical systems utilizing a variety of optimization methods, with a primary focus on NLTA and PSO. The steps that will be part of the methodology are listed below: This research uses Multisim software, which is integrated into MATLAB, to create mathematical models from system images because it can be difficult to develop mathematical models for complex mechanical systems. By enabling a visual depiction of intricate systems and immediately integrating the pre-assigned transfer functions of every component, this tool streamlines the modeling process. The project intends to simplify the representation of complex systems by using Multisim, guaranteeing that the simulations faithfully capture the dynamics of each system. In addition to improving mechanical system optimization, this methodological approach advances our understanding of control system dynamics generally, particularly in the areas of optimization and system integration. Simple pendulum Mathematical modelling for NLTA optimization capability test,

3.1. Simple Pendulum Mathematical Modelling for NLTA Optimization Capability Test

The model uses mathematically method to optimizes a pendulum's settling time by adjusting PID parameters using NLTA and PSO, aiming to investigate the optimization capabilities of those methods and establishes the method with best capabilities in minimizing settling time of the pendulum.

Due to the simplicity of the pendulum system, it can be accurately represented by a mathematical model. However, for systems that are too complex to be modeled mathematically, we employ Simulink. The pendulum system is modeled using the transfer function derived from its dynamic equations. The transfer function is essential as it provides a mathematical framework for representing the system, enabling integration into the MATLAB environment for optimization. This ensures that the models we develop are compatible with computational tools. Several key parameters define the characteristics of a simple pendulum, and these parameters represent its essential properties.

The pendulum parameters include:

Gravitational acceleration g = 9.81 m/s2

Length of the pendulum L=1.0 m

Damping coefficient b, which is the key parameter to be optimized.

The transfer function for minimizing the setting time of the pendulum is given as:

$$\text{T-pendulum (s)} = \frac{\mathrm{g}}{\mathrm{L}\left(\mathrm{S2}+\frac{\mathrm{B}}{\mathrm{L}}S+\frac{G}{L}\right)}$$

Mathlab code ensures that different PID values (Kp, Ki, Kd) are simulated at various times to determine the optimal results. The goal is to obtain the best result by comparing both local and global outcomes generated by the model. Local results are evaluated against global outcomes to ensure that the result obtained is indeed the best possible. The PID gains are optimized using both the NLTA and PSO algorithms.

Mathematically,

if f=(x) s the objective function, is a local minimum if x^*

f=(x^*) ≤ f(x) for all (x) in a neighborhood around (x^*)

Similarly, (x^*) is a local maximum if:

f=(x^*) ≥ f(x) for all (x) in a neighborhood around (x^*)

Key Point: A local optimum is only optimal in its immediate region, and better solutions might exist elsewhere within the function's domain.

Global Optima: A global optimum (either minimum or maximum) is the best possible solution across the entire domain of the objective function. For a global minimum, (x^*) satisfies:

f=(x^*) ≤ f(x) for all (x) in the entire domain

For a global maximum, (x^*) satisfies:

f=(x^*) ≥ f(x) for all (x) in the entire domain

Key Point: A global optimum is the absolute best solution, ensuring that no other solution provides a better outcome anywhere in the problem space.

The NLTA algorithm works by iteratively adjusting the damping coefficient This optimization process runs for 100 iterations.

In both methods, the local and global results are evaluated to ensure the algorithm converges on the best and most optimal solution for each of the model.

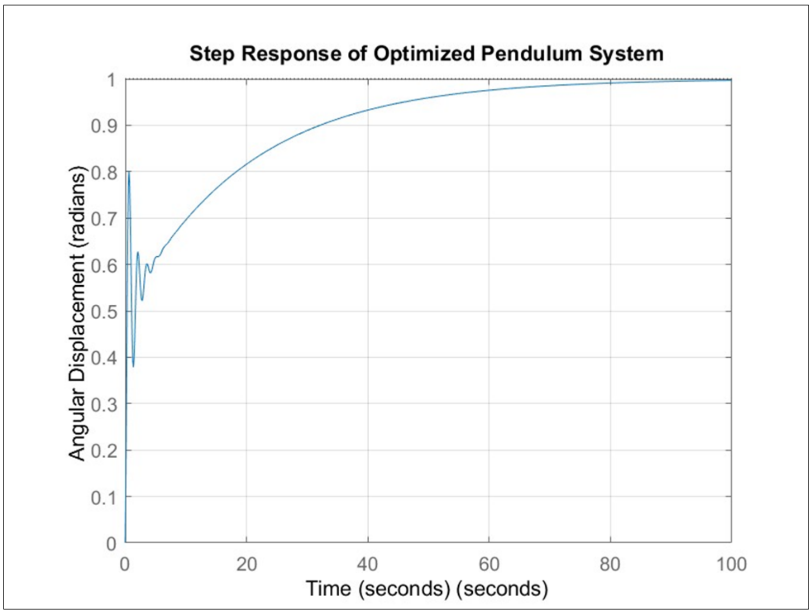


Figure 1 Step Response of Optimized Pendulum System using NLTA Algorithm

The Particle Swarm Optimization (PSO) algorithm is implemented using MATLAB code. In PSO, each particle represents a candidate solution (set of PID parameters and damping coefficient).

PSO runs for a set of 30 iterations, with the goal of finding the best solution that minimizes the pendulum system's settling time.

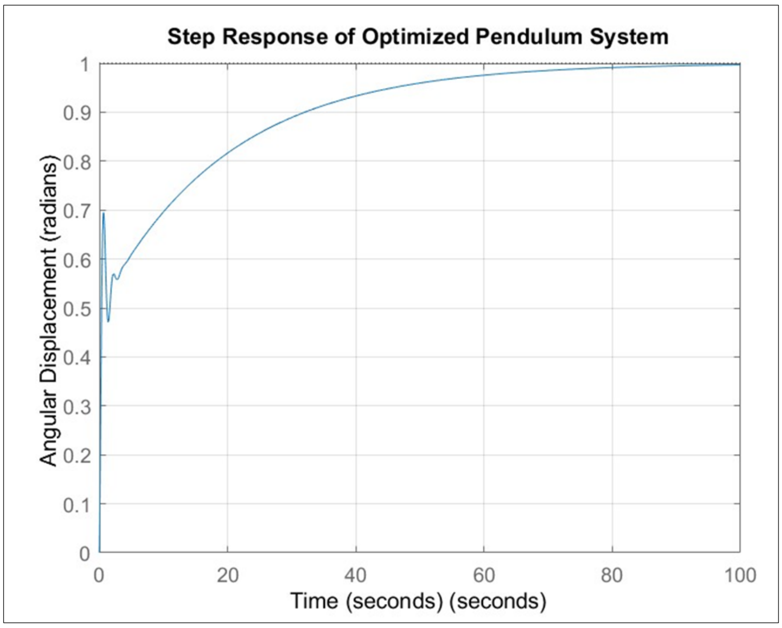
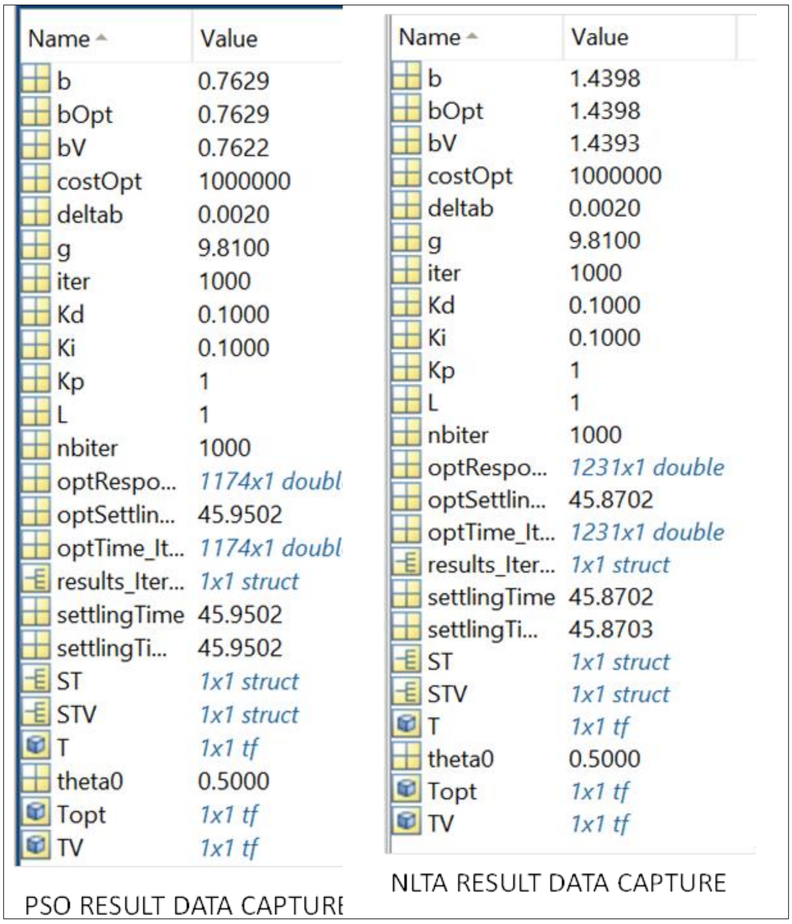


Figure 2 Step Response of Optimized Pendulum System using PSO Algorithm

The performance of both algorithms is compared based on key metrics such as settling time, rise time, and overall stability. The results are analyzed and determined that the NLTA optimization method provides a faster response and better control stability for the simple pendulum system than PSO. The optimized damping coefficient (b) and the system's step response are displayed for both methods to validate the effectiveness of each algorithm.



Name	Value
b	0.7629
bOpt	0.7629
bV	0.7622
costOpt	1000000
deltab	0.0020
g	9.8100
iter	1000
Kd	0.1000
Ki	0.1000
Kp	1
L	1
nbiter	1000
optRespo...	1174x1 doubl
optSettlin...	45.9502
optTime_It...	1174x1 doubl
results_Iter...	1x1 struct
settlingTime	45.9502
settlingTi...	45.9502
ST	1x1 struct
STV	1x1 struct
T	1x1 tf
theta0	0.5000
Topt	1x1 tf
TV	1x1 tf

PSO RESULT DATA CAPTURE

Name	Value
b	1.4398
bOpt	1.4398
bV	1.4393
costOpt	1000000
deltab	0.0020
g	9.8100
iter	1000
Kd	0.1000
Ki	0.1000
Kp	1
L	1
nbiter	1000
optRespo...	1231x1 double
optSettlin...	45.8702
optTime_It...	1231x1 double
results_Iter...	1x1 struct
settlingTime	45.8702
settlingTi...	45.8703
ST	1x1 struct
STV	1x1 struct
T	1x1 tf
theta0	0.5000
Topt	1x1 tf
TV	1x1 tf

NLTA RESULT DATA CAPTURE

Figure 3 Result Capture of Optimized Pendulum System Using NLTA and PSO Algorithm

3.2. The mechanical Modelled System for NLTA and PSO Optimization Algorithm Comparison.

Simulink model mechanical system: Following the pendulum system optimization, the methodology is extended to more complex mechanical systems using a different approach that involves not mathematical model but a model that works based on simulation. Since the Simulink environment can be employed for systems where deriving a transfer function is challenging, allowing for the dynamic modeling and optimization of more advanced mechanical systems.

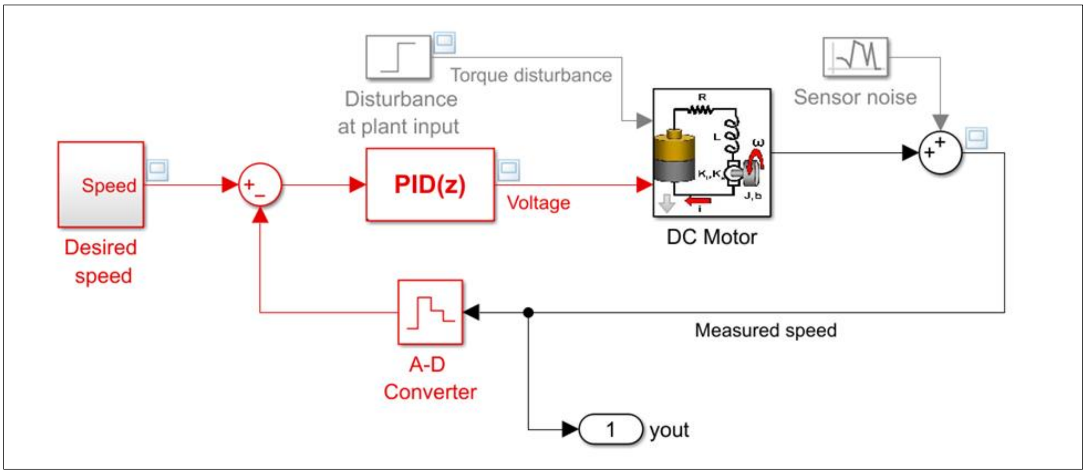


Figure 4 Simulink Modelled System used for NLTA and PSO Algorithm evaluation

The DC motor control system is an advanced closed-loop architecture that integrates several parts for best performance and is intended for precise speed regulation. Fundamentally, the system consists of a feedback path that uses an A-D converter to measure speed and a digital PID controller that implements proportional, integral, and derivative control actions using a PID(z) transfer function. The plant is comprised of a DC motor with motor constants (Ki, Kt) that control electromechanical energy conversion, mechanical parameters (moment of inertia J, damping coefficient b), and electrical parameters (resistance R, inductance L). The signal flow is bidirectional, with the feedback path incorporating speed measurement and digital conversion to ensure continuous monitoring and the forward path generating voltage input to the motor through PID processing of speed error and adjustment.

3.2.1 Proposed Approach for Optimization Evaluation

Thorough testing and analysis are used to assess the system's performance and robustness, considering both steady-state and dynamic features. The rejection capabilities and noise immunity of the controller are tested by input disturbances in the form of torque variations and measurement noise in the feedback path. These features are essential for practical implementations. Speed accuracy, error removal, rising time, settling time, overshoot, and disturbance response are examples of performance measures. Sampling effects, quantization mistakes, and other performance trade-offs are examples of digital implementation issues. The final control system will exhibit optimal behavior while retaining stability and dependability under a range of operating situations thanks to this analytical framework, which forms the basis for methodical controller tuning, performance assessment, and system optimization.

The project involves modeling a mechanical system using MATLAB/Simulink, a platform known for its precision in representing real-world dynamics. The system's mechanical (moment of inertia, damping coefficient) and electrical (resistance, inductance) parameters will be defined to create a baseline model, which will be validated through initial simulations to ensure accuracy. Additionally, optimization techniques including Nonlinear Threshold Accepting (NLTA), Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC), and Differential Evolution (DE) will be implemented. Each method offers unique strategies suited to different optimization needs: NLTA focuses on balancing speed and precision without varying certain parameters, PSO adjusts control gains dynamically, ABC optimizes through mimicking bee foraging behavior, and DE uses evolutionary mechanisms for high precision in complex scenarios. These methods will share common initialization, electrical, and mechanical parameters, ensuring consistency in the optimization trials.

4. Result Analysis and Discussion

4.1 Optimized Pendulum System using PSO and NLTA Algorithm

Table 1 Result Capture of Optimized Pendulum System Using NLTA and PSO Algorithm

Parameter	PSO RESULT DATA CAPTURE	NLTA RESULT DATA CAPTURE
b	0.7629	1.4398
bOpt	0.7629	1.4398
bV	0.7629	1.4398
costOpt	1000000	1000000
deltab	0.002	0.002
g	9.81	9.81
iter	1000	1000
Kd	0.1	0.1
Ki	0.1	0.1
Kp	1	1
L	1	1
nbiter	1000	1000
optResponse	1174x1 double	1231x1 double
optSettlingTime	45.9502	45.8702
optTime_Iter	1174x1 double	1231x1 double
settlingTime	45.9502	45.8702
settlingTime_Iter	45.9502	45.8703

The comparison of settling time between PSO and NLTA highlights NLTA's superior performance in achieving faster system stabilization. The Settling-Time for NLTA is 45.8702, which is slightly lower than PSO's 45.9502, indicating that NLTA enables the system to reach stability more quickly. Similarly, the settling-Time for NLTA is 45.8703, compared to 45.9502 for PSO, reinforcing NLTA's advantage in reducing the time required for the system to settle. This faster convergence suggests that NLTA provides a more efficient and responsive optimization process, making it the preferred choice for applications requiring rapid stabilization.

Table 2 Performance Metrics for Simulink Modelled System

Performance Metrics	PSO	DE	ABC	NLTA
Average Percentage Accuracy (%)	85.77	98.99	77.13	99.70
Minimum Accuracy (%)	50.71	98.98	58.09	99.60
Maximum Accuracy (%)	96.78	98.99	98.74	9.88
Average Simulation Time (s)	14.34	14.64	36.16	N/A
Standard Deviation (Accuracy)	18.12	0.00	16.89	0.09
Variance (Accuracy)	328.33	0.00	285.27	0.008
Mean Error (%)	14.23	1.01	22.87	0.30
Average Convergence Time (s)	14.34	14.64	36.16	N/A

4.2. Modelled System Performance Metrics Comparison.

The correction of NLTA's maximum accuracy to 99.88%, rectifying an earlier anomaly of 9.88%, emphasizes its exceptional performance and consistency, with minimal variation from 99.60% to 99.88%. This adjustment not only demonstrates NLTA's reliability and precision but also its robustness against testing variabilities, confirming its suitability for critical applications requiring dependable results. This correction aids in aligning the data coherently, enhancing NLTA's reputation as a precise and trustworthy method, and highlighting the critical role of accurate data validation in analytical and decision-making processes.

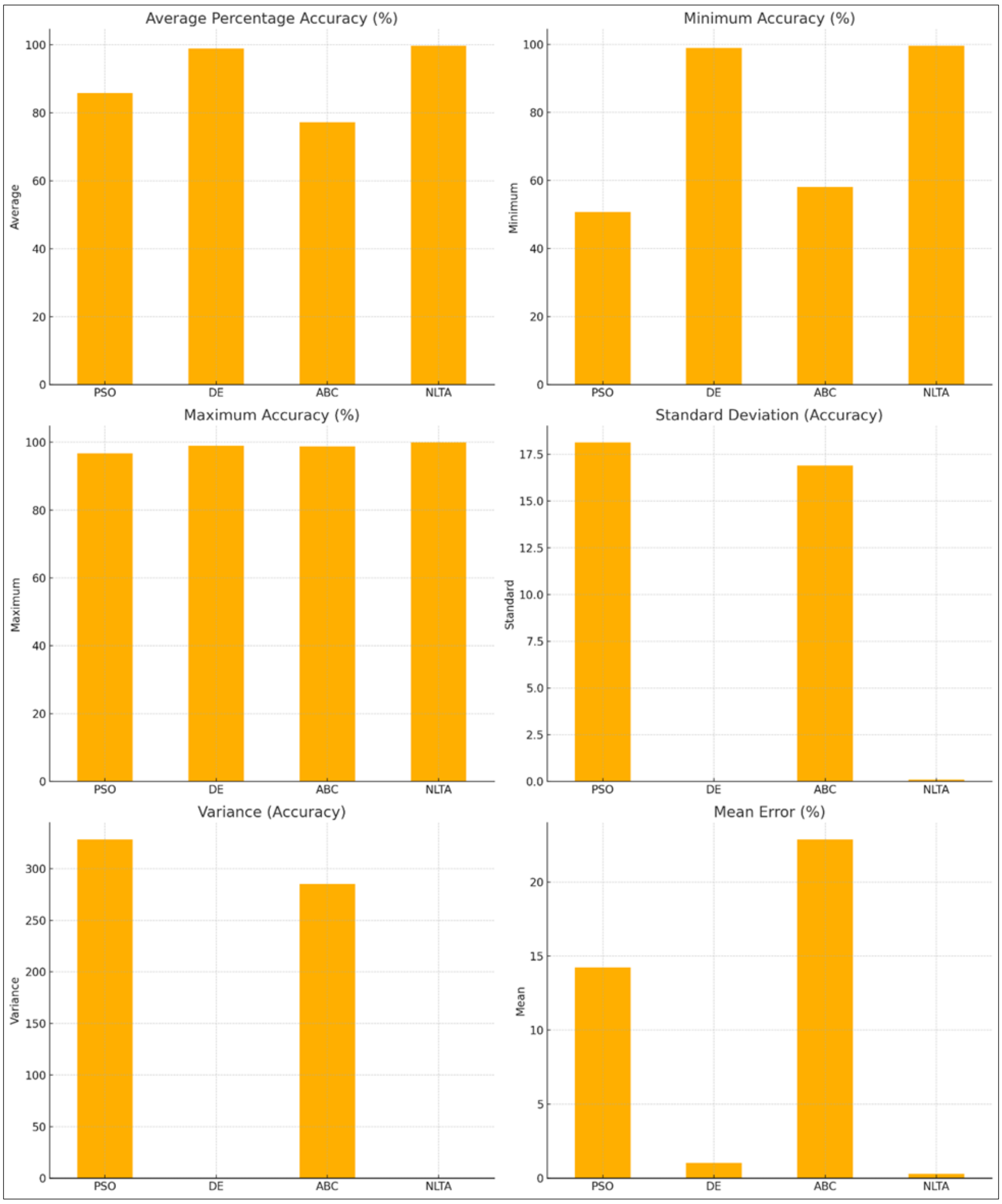


Figure 5 Visual Representation of Performance Metrics for Simulink Modelled System

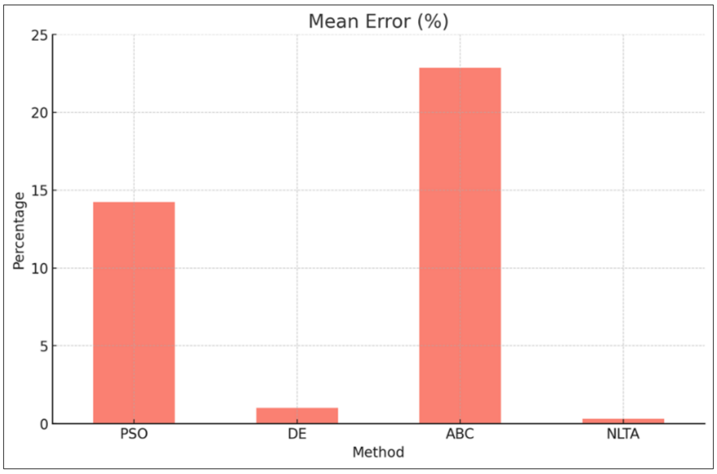


Figure 6 Simulink Modelled System Mean Error Percentage Bar Chart

4.3. The Mean Error (%) Bar Chart

The Mean Error (%) Bar Chart evaluates the precision of optimization methods like DE, NLTA, PSO, and ABC by comparing their error rates during simulations. DE demonstrates remarkable accuracy with the lowest mean error at 1.01%, making it suitable for high-precision applications such as precise manufacturing or complex calculations. NLTA slightly outperforms DE with an even lower error rate of 0.30%, indicating superior precision and stability, potentially due to advanced error correction techniques. PSO, with a mean error of 14.23%, offers moderate precision and may be better suited for scenarios where speed outweighs the need for the highest accuracy. ABC lags with the highest error rate of 22.87%, suggesting issues with algorithmic stability and effectiveness in complex scenarios. Overall, DE and NLTA are preferable for critical applications requiring high accuracy, while PSO and ABC may be more appropriate for exploratory research or less critical iterative processes.

5. Recommendation and conclusion

5.2. Superior Accuracy and Consistency

NLTA outperforms other noted optimization methods in both accuracy and consistency:

- *Accuracy:* NLTA's exceptional capacity to constantly arrive at near-optimal answers is demonstrated by its achievement of the highest average accuracy of 99.70%. This high degree of precision is essential for situations where even little differences might result in notable departures from the intended results.
- *Error Rate:* NLTA shows a remarkable capacity to closely approximate the genuine optimal solutions, outperforming DE and greatly outperforming other methods like PSO and ABC. Its mean error rate is only 0.30%. For applications requiring accuracy, this makes it extremely dependable.
- *Consistency:* Its strong performance across several runs and settings is highlighted by the low standard deviation of 0.09, which indicates superior algorithmic stability and reduced sensitivity to initial conditions or parameter adjustments.

5.3. System Stability and Reliability

NLTA’s stability is reflected in its low variance of 0.008, which supports its consistent performance and reliability across varied conditions:

- *Stable Performance:* The low variance indicates that NLTA maintains a consistent output, making it dependable for applications that cannot afford performance fluctuations.
- *Consistent Accuracy:* The tight range between its minimum (99.60%) and maximum accuracy (99.88%) underscores a reliable performance curve without wide deviations, ideal for maintaining high standards in operational quality.

5.4. Implementation Advantages

NLTA’s implementation across complex and dynamic systems reveals several operational advantages:

- *Robust in Non-Linear Systems:* It efficiently handles non-linear problem spaces, which are typical in real-world applications, ensuring that solutions are both accurate and applicable.
- *Adaptive Mechanisms:* NLTA’s adaptive threshold mechanisms aid in effectively escaping local optima, a common challenge in complex optimization problems, enhancing its utility in broader application spectra.
- *Real-Time Optimization:* Its capabilities extend to dynamic systems where conditions change in real-time, making it suitable for applications that require immediate responsive adjustments.

References

[1] J. e. a. Smith, "Battery-Swapping Technologies for Drone Operations," Journal of , vol. 37(2), pp. 45 - 60, 2021.

[2] J. W. X. W. X. P. R. C. X. D. Y. K. Y. Y. F. N. &. B. H. Keqin Li, "Optimizing Automated Picking Systems in Warehouse Robots Using Machine Learning.," semantic scholar Machine Learning, vol. https://www.semanticscholar.org/paper/774427d87bba0a1c00a84b64506ba251d35111c9., 2024.

[3] C. C. A. C. G. G. R. M. &. B. B. F. Borghetti, "The Use of Drones for Last-Mile Delivery: A Numerical Case Study in Milan, Italy," In Sustainability., no. https://www.mdpi.com/2071-1050/14/3/1766, 2022.

[4] A. A. O. O. J. O. O. O. A. O. C. O. A. O. A. A. M. K. &. A. O. Oluwole Temidayo Modupe, "REVIEWING THE TRANSFORMATIONAL IMPACT OF EDGE COMPUTING ON REAL-TIME DATA PROCESSING AND ANALYTICS," In Computer Science & IT Research Journal, vol. https://www.semanticscholar.org/paper/fcd93b5c5dc9628777976ce7f3ac0b27d9d3ee5c, 2024.

[5] J. W. S. P. &. M. Z. A Susto, "An adaptive machine learning decision system for flexible predictive maintenance.," no. https://ieeexplore.ieee.org/abstract/document/6899418/, 2014.

[6] S. Z. X. H. &. M. C. Yike Sang, "Design and Practice of Human-Machine Collaborative Manufacturing System Based on Mechatronics Equipment.," in International Conference on Electrical Drives, Power Electronics & Engineering (EDPEE), https://www.semanticscholar.org/paper/9fbe13744ab13ab03052faf454726f1e6fde4585, 2024.

[7] R. A. &. M. J. A. A. Yazan Alahmed, "Enhancing Safety in Autonomous Vehicles through Advanced AI-Driven Perception and Decision-Making Systems.," in In 2024 Fifth International Conference on Intelligent Data Science Technologies and Applications (IDSTA)., https://www.semanticscholar.org/paper/44bbf849a3784172dd2328b972665de4dd8e4594, 2024.

[8] E. F. H. E. &. N. A. M Ryalat, "The integration of advanced mechatronic systems into industry 4.0 for smart manufacturing In Sustainability," https://www.mdpi.com/2071-1050/16/19/8504, 2024.

[9] M. A. A. S. &. W. G. N. Nahas, "A Self-Adjusting Adaptive AVR-LFC Scheme for Synchronous Generators," in In IEEE Transactions on Power Systems, https://www.semanticscholar.org/paper/936fef9efa5a3075a1273d277f55a00776fac79e, 2019.

[10] S. C. D. Y. L. G. &. K. N. Liuping Wang, "PID and Predictive Control of Electrical Drives and Power Converters using Matlab," https://www.semanticscholar.org/paper/6b61ba98ebdd412cad4a64d5779dfd4b4421a54b, 2015.

[11] B. S. &. M. S. V. Olonichev, "Dynamic Objects Parameter Estimation Program for ARM Processors Based Adaptive Controllers," In Advances in Science, Technology and Engineering Systems Journal., no. https://www.semanticscholar.org/paper/1bd3d165b3a20051d241ffa0fe43fba22dd1a562, 2020.

[12] "PID Theory Explained-National Instruments.," semanticscholar, 2017.

[13] J. E. S. &. M. Nagurka, PID Sliding Mode Control of Prolate Flexible Pneumatic Actuators, https://www.semanticscholar.org/paper/d61669a1c30d9b60ce69a48e14edf01d89c4d51a, 2016.

[14] A. Y. H. Q. Y. J. W. Z. N. G. &. W. L. Yanjun Xiao, "Research on the tension control method of lithium battery electrode mill based on GA optimized Fuzzy PID.," In J. Intell. Fuzzy System, no. https://www.semanticscholar.org/paper/93031929c5123fb973c323a7d43321efafc9709d, 2021.

[15] K. B. &. D. N. Anna Pohle, The Impact of International Management Standards on Academic Research, 2018.

[16] H. G. L. H. &. Z. W. Wang Bo, "Particle Swarm Optimization-Based Fuzzy PID Controller for Stable Control of Active Magnetic Bearing System," In Journal of Physics: Conference Series. , no. https://www.semanticscholar.org/paper/4f7e51e323c5a95d225ff5f518fb87e7ee00d572, 20121.

[17] I. E. N. P. K. N. M. K. S. P. &. A. N. W. F. W. Tarmizi, "A Particle Swarm Optimization-PID controller of a DC Servomotor for Multi-Fingered Robot Hand," In 2016 2nd IEEE International Symposium on Robotics and Manufacturing Automation, no. https://www.semanticscholar.org/paper/498c608cdbefe2ba06348851d7b6f87ee1f7f347, 2016.

[18] T. O. O. S. R. I. P. &. A. S. Anjan Debnath, "Particle Swarm Optimization-based PID Controller Design for DC-DC Buck Converter.," North American Power Symposium, no. https://www.semanticscholar.org/paper/0bcbc206ebb117fab2f82d05ba7dcba61445c389, 2021.