www.ijarp.com ISSN 2456-9992 Page: 01-37

International Journal Advanced Research Publications

Research Article

Volume: 02

Issue: 07

A COMPARATIVE DC MOTOR SPEED CONTROL OF AN INTELLIGENT GROUNDNUTS PODS THRESHING MACHINE WITH PID AND FUZZY LOGIC CONTROLLER

B. Bello *[1]., J.D. Jiya[2]

[1,2]Department of Electrical and Electronics Engineering, Abubakar Tafawa Balewa Unicersity Bauchi, 740272, Nigeria.

Article Received: 05 June 2026, Article Revised: 25 June 2026, Published on: 15 July 2026

***Corresponding Author: B. Bello**

Department of Electrical and Electronics Engineering, Abubakar Tafawa Balewa Unicersity Bauchi, 740272, Nigeria.

Doi: https://doi-doi.org/101555/ijarp.2392

ABSTRACT

Groundnut threshing is a critical post-harvest operation that significantly influences grain quality, productivity, and processing efficiency. Intelligent speed regulation of the threshing drum has become essential for minimizing seed damage while maintaining high operational performance under varying crop conditions. However, conventional PI/PID controllers often experience reduced performance under nonlinear operating conditions and fluctuating load disturbances caused by variations in groundnut pod properties and feeding rates, resulting in slower response and reduced adaptability. To address this challenge, this study developed and evaluated an adaptive DC motor speed control system for an intelligent groundnut threshing machine using PI and Singleton Fuzzy Logic Controllers. A mathematical model of a 220 V series DC motor with an armature resistance of 0.5 Ω, armature inductance of 0.0025 H, rotor inertia of 0.0013 kg·m², and load torque of 21 N·m was developed and implemented in MATLAB/Simulink. Controller performance was evaluated at reference speeds of 50, 90, and 150 rpm using rise time, settling time, overshoot, and steady-state error as performance indices. The Singleton Fuzzy Logic Controller achieved superior dynamic performance with a rise time of 0.011 to 0.014s compared with 0.035s for the PI controller and a settling time of 0.017 to 0.049s compared with 0.099 to 0.150s, while both controllers maintained 0% overshoot. The fuzzy controller also achieved a steady-state error of 0.3 to 1.1%, demonstrating accurate speed tracking and robust operation over the tested speed range. The study concludes that the Singleton Fuzzy Logic Controller provides faster response, improved robustness, and more reliable speed regulation than the PI controller, making it a suitable

intelligent control strategy for enhancing the performance and automation of groundnut threshing machines.

KEYWORDS: Groundnut threshing machine, DC motor speed control, Fuzzy Logic Controller, PI controller, MATLAB/Simulink.

1. INTRODUCTION

Groundnut (*Arachis hypogaea* L.) is one of the most important oilseed and food legume crops cultivated worldwide due to its high nutritional value, economic significance, and wide range of industrial applications. The crop serves as a major source of edible oil, protein, vitamins, and minerals, while its by-products are extensively utilized in livestock feed and food processing industries. In many developing countries, particularly in sub-Saharan Africa, groundnut production contributes significantly to food security, poverty reduction, and rural economic development. Nigeria is among the leading producers of groundnut in Africa, where millions of smallholder farmers depend on its cultivation as a major source of livelihood (Lokossou *et al.*, 2022).

Threshing is a critical post-harvest operation that involves separating groundnut pods from the harvested vines or plants. The efficiency of this operation directly influences the quantity and quality of marketable produce. Traditional threshing methods, including manual beating and simple mechanical devices, are generally labour-intensive, time-consuming, and inefficient. They often result in low throughput capacity, high pod losses, and excessive mechanical damage to seeds, inconsistent threshing quality, and increased labour costs (Bello *et al.,* 2021).. These limitations reduce overall processing efficiency and negatively affect the commercial value of the harvested crop. Consequently, the development of intelligent threshing machines capable of achieving high threshing efficiency while minimizing kernel damage has become an important research focus in agricultural mechanization (Sale *et al.,* 2022).

Modern agricultural engineering has adopted automation and intelligent control technologies to improve the performance of agricultural machinery such as groundnut threshing machine. Intelligent threshing machines integrate sensors, electronic control units, and advanced control algorithms to regulate machine operating parameters in real time (Onuoha *et al.,* 2022). Among these parameters, the rotational speed of the threshing drum is one of the most influential factors affecting threshing efficiency, seed damage, cleaning efficiency, energy consumption, and machine productivity. The optimum drum speed varies with several factors, including groundnut variety, pod size, feeding rate. Therefore, maintaining a constant or adaptive drum speed under changing operating conditions is essential for achieving optimal machine performance. Conventional threshing machines generally operate at a fixed drum speed regardless of variations in crop loading conditions (Bello *et al.,* 2019).. During practical field operations, however, the feeding rate fluctuates continuously, causing sudden changes in load torque acting on the driving motor (Widayat *et al.,* 2025). These disturbances lead to undesirable speed fluctuations that reduce threshing efficiency, increase seed breakage, and elevate power consumption. The nonlinear characteristics of the threshing process, coupled with uncertainties arising from variable crop properties, make accurate speed regulation a challenging control problem Lin *et al.,* (2020).

The proportional integral derivative (PID) controller remains one of the most widely applied control techniques in industrial automation due to its simple structure, ease of implementation, and satisfactory performance for linear systems. PID controllers regulate system output by minimizing the error between the desired and measured speeds through proportional, integral, and derivative actions. Since agricultural threshing processes exhibit highly nonlinear behaviour resulting from variations in crop properties and feeding conditions, conventional PID controllers often require repeated tuning and may fail to maintain optimal performance under dynamic operating environments. To overcome this challenges, intelligent control technique is require which is Fuzzy Logic Controller (FLC). . Fuzzy Logic Controllers emulate human reasoning by incorporating expert knowledge into a set of linguistic rules rather than relying on an exact mathematical model of the controlled system. FLCs are particularly suitable for agricultural machinery because they can effectively handle nonlinearities, uncertainties, and parameter variations commonly encountered during field operations. By continuously adjusting the motor control signal according to speed error and its rate of change, fuzzy controllers provide smoother responses, reduced overshoot, and improved robustness compared with conventional PID controllers Saatchi (2024).

However, although numerous studies have investigated PID, fuzzy logic controllers independently for motor speed regulation, relatively few have provided a comprehensive comparison of these two control strategies for intelligent groundnut pod threshing machines operating under realistic and varying loading conditions. Most existing threshing machines continue to rely on constant speed operation or conventional control techniques that cannot adequately compensate for rapid load fluctuations caused by changes in crop feeding rate.

This studies therefore, the adaptive speed control system for an intelligent groundnut pods threshing machine using three different control approaches: a PID controller, a Fuzzy Logic Controller, and an Adaptive Neuro-Fuzzy Inference System. A mathematical model of the threshing machine drive system is developed and implemented in MATLAB/Simulink to evaluate the dynamic performance of the three controllers under identical operating conditions fields Lingxiao and Pang (2020).

2. LITERATURE

The modern agricultural mechanization has increasingly focused on the development of intelligent threshing systems capable of improving productivity, reducing post-harvest losses, and enhancing product quality. Groundnut threshing machines have transformed from manually operated devices to motorized and semi-automated systems designed to increase throughput and reduce labour intensities (Lokossou *et al.*, 2022). Researches on groundnut threshing have shown that machine performance is strongly influenced by operational parameters such as drum speed, concave clearance, feed rate, and moisture content. Excessively high drum speeds generally increase threshing efficiency but also lead to greater seed breakage and mechanical damage, whereas low speeds reduce seed damage but may result in incomplete threshing and lower throughput (Sale *et al.,* 2022). Consequently, maintaining an optimal threshing drum speed under varying groundnut conditions has become a major challenge in the design of intelligent groundnut threshing machines.

These systems aim to automatically adjust machine operating conditions in response to changing crop characteristics. Although such approaches demonstrate the potential of smart agricultural machinery, many existing designs still rely on fixed speed operation or simple manual adjustments, which are inadequate for handling rapid variations in feeding rate and crop properties during practical field operations. This limitation highlights the need for adaptive control strategies capable of regulating drum speed dynamically and maintaining stable machine performance Lin *et al.,* (2020).

DC motors are widely used in agricultural processing equipment because of their simplicity, high starting torque, ease of speed regulation, and compatibility with electronic control systems. The speed of a DC motor can be controlled by varying the armature voltage, field flux, or armature resistance. Numerous studies have modelled DC motor dynamics using electrical and mechanical equations that relate armature voltage, current, torque, angular velocity, and load disturbance (Judy 2025). Because the threshing drum load changes continuously with crop feeding conditions, the driving motor behaves as a nonlinear and time-varying system subjected to external disturbances. Effective speed regulation therefore requires a controller capable of compensating for load fluctuations while maintaining accurate reference tracking (Moghaddas *et al.,* 2011).

2.1 Fuzzy model

A fuzzy model is a way of describing a system that can handle imprecise, vague, or uncertain information, much like how humans make decisions in everyday life. Unlike traditional logic, which classifies things as either true or false, fuzzy models work with degrees of truth, allowing something to be partly true and partly false at the same time Chiu and Peng (2019)

2.1.1 Mamdani Fuzzy Models

The first and most innovative vision of fuzzy models represents the input and output relationships through a collection of IF and THEN rules, where both the antecedents and consequents are expressed using fuzzy values. This basic form of fuzzy rules relies on fuzzy sets in the consequent part, making the models highly intuitive and easy to understand. These models apply fuzzy reasoning and serve as the foundation for qualitative modeling, which expresses input and output mappings in natural language. A well-known example in this category is the Mamdani model. This approach aligns with the ultimate goal of fuzzy logic computing with words and emphasizes model readability over computational cost and accuracy in terms of approximation, classification, or control. However, a drawback of this class of models is that they often become complex, requiring many parameters, which can make them difficult to run, maintain, and fine-tune (Setiawan etal., 2020)

$$R_k : \text{IF } (x \text{ is } A_k) \text{ THEN } (y \text{ is } B_k) \qquad \ldots (1)$$

Where: A_k is a fuzzy set on the input space.

B_k is a fuzzy set on the output space.

2.1.2.Takagi Sugeno Fuzzy Models

This is the second type fuzzy model, where fuzzy rules are defined with fuzzy antecedents but functional consequents, making them a hybrid of fuzzy and non-fuzzy models. These models are capable of representing a wide range of static or dynamic nonlinear systems by combining multiple linear models. The input space is divided into several fuzzy subspaces, and within each subspace, the output is expressed through a linear function.

R_k: IF (x is A_k) THEN (y = h_k (x)) … (2)

Where h_k (x) is the polynomial function of the inputs and A_k represents a local model used to approximate the response of the system Chiu and Peng (2019).

2.2 Membership function

The membership function (MF) defines how each crisp element in the input space is mapped to a membership degree between 0 and 1 within a fuzzy set. The fuzzy essentially expresses how strongly a value belongs to a linguistic category such as low, medium, *or* high. Different shapes of membership functions such as triangular, trapezoidal, Gaussian, bell-shaped, and sigmoidal are used depending on the nature of the problem Tomasiello and Alijani (2021).

2.2.1. Triangular membership Function

The triangular membership function is one of the simplest and most commonly used in fuzzy logic applications. This is defining by the three parameters the left foot A, the peak B, and the right foot C. The function increases linearly from zero at point A to a maximum peak of 1 at point B, and then decreases linearly back to zero at point C as shown in figure 1. Because of its simplicity, triangular functions are computationally efficient and easy to implement. They are particularly suitable for cases where the system requires a quick approximation of linguistic terms such as low, medium, *and* high. For example, in a groundnut threshing machine, the motor speed can be represented by triangular membership functions, where the ranges for low, medium, and high speeds overlap smoothly to allow flexible control decisions (Kabir *et al*., 2024).

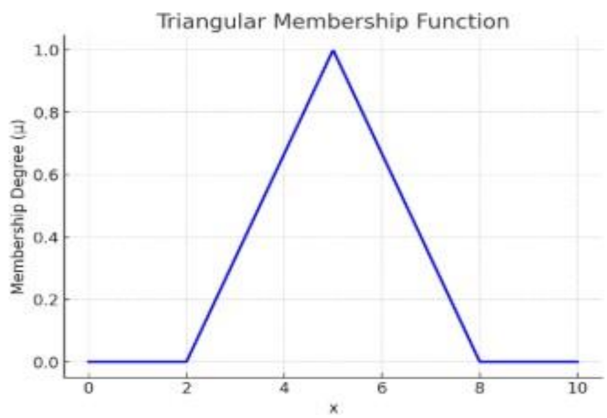

Figure 1: Triangular Membership Function.

The following are the triangular membership function parameters A= 2, B = 5 and C = 8 showing how the degree of membership rises linearly from zero to the peak at 1 and then decreases back to zero.

$$\mu(x;a,b,c)=\begin{cases}0, & x\le a \text{ or } x\ge c\\ \dfrac{x-a}{b-a}, & a<x\le b\\ \dfrac{c-x}{c-b}, & b<x<c\end{cases} \quad \dots (3)$$

2.2.2. Trapezoidal Membership Function

The trapezoidal membership function is a generalization of the triangular function, with a flat top between two middle parameters. It is defined by four parameters points ABCD where the function rises linearly from zero point A to point B, which remains steady to point C and then decreases back to zero at point D. The trapezoidal shape makes it useful when a variable can remain in a certain fuzzy state over a range of values, rather than at a single peak point. For instance, if a concave clearance setting between 8–12 mm is considered fully medium then a trapezoidal membership function can represent this, ensuring stable control without unnecessary switching between fuzzy sets as indicated in figure 2 (Wang *et al.*, 2021).

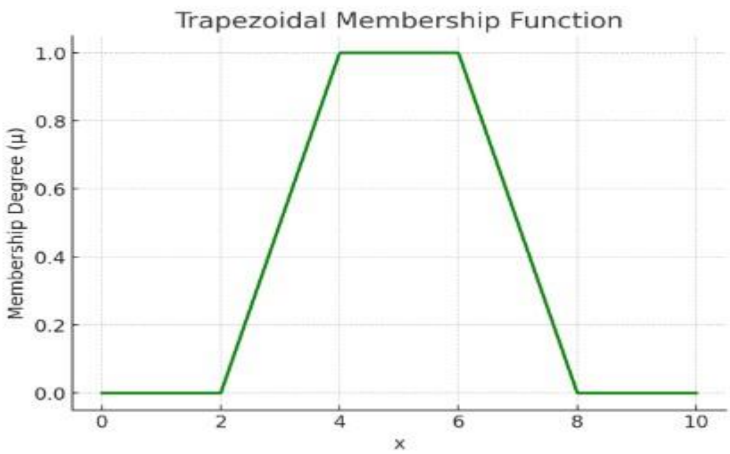

Figure 2: Trapezoidal Membership Function.

Therefore, the trapezoid membership function parameters are A= 2, B = 4, C = 6 and D = 8 showing how the degree of membership rises linearly from zero to a setting range of value before decreases back to zero.

Hence the defined parameter are as these a, b, c, d with a$< b \le c < d$

$$\mu(x;a,b,c,d)=\begin{cases}0, & x\le a \text{ or } x\ge d\\ \dfrac{x-a}{b-a}, & a<x\le b\\ 1, & b<x\le c\\ \dfrac{d-x}{d-c}, & c<x<d\end{cases} \quad \text{... (4)}$$

2.2.3. Gaussian Membership Function

The Gaussian membership function is a smooth, bell-shaped curve defined by two parameters the center c and the standard deviation sigma. The function gradually increases toward the center and decreases symmetrically on either side, creating a continuous and smooth transition between fuzzy sets. This property makes Gaussian functions highly effective in systems that require precision and gradual changes, avoiding abrupt transitions that could destabilize control. In the context of groundnut threshing, Gaussian functions can be used to model uncertain parameters such as kernel moisture content, where smooth variations in input can lead to more stable control actions as shown in figure 3.

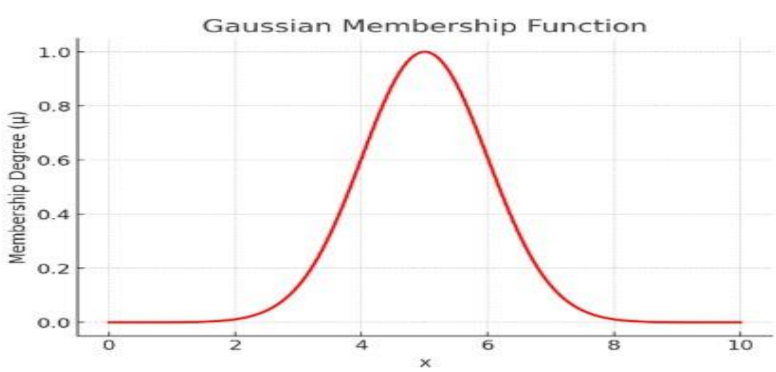

Figure 3: Gaussian Membership Function.

The Gaussian membership function with center C = 5 and width sigma (σ) = 1. The fuzzy set rises with a smooth bell-shaped curve with gradual transitions on both sides. The parameter is defined as center C and the standard deviation is σ.

$$\mu(x;c,\sigma)=\exp\left(-\frac{(x-c)^2}{2\sigma^2}\right) \quad \text{... (5)}$$

2.2.4 Sigmoidal Membership Function

The sigmoidal membership function is an S-shaped curve defined by two parameters the slope

A, and the center C. It is particularly useful for modeling open ended fuzzy sets such as very low or

$$\mu(x; a, c) = \frac{1}{1 + e^{-a(x-c)}} \quad \dots (6)$$

Very high, because the curve does not have a finite boundary but instead asymptotically approaches zero or one. This makes it suitable for cases where the variable of interest continues indefinitely in one direction. In the threshing machine application, the sigmoidal function can be used to represent conditions like excessive load *or* very high speed, where the fuzzy concept extends beyond the normal operating range of the system as indicated in figure 4.

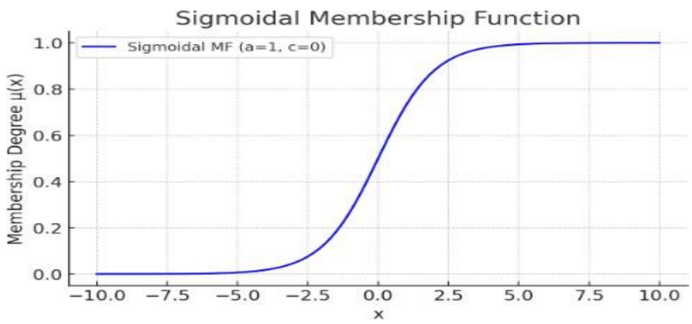

Figure 4: Sigmoidal Membership Function.

2.3. Fuzzy inference system (FIS) architecture

The fuzzy inference system (FIS) it's an architecture that contain the four main components that work together to transform crisp input values into a crisp output through reasoning with fuzzification, fuzzy rules, inference engine and defuzzification (Saatchi 2024).

2.3.1 Fuzzification

Fuzzification is the first and most important step in a fuzzy determination variable, where the crisp numerical inputs are converted into fuzzy values sets that can be processed by the system as indicated in figure 5. In this process, each input variable is mapped into one or more fuzzy sets using predefined membership functions. For example, a crisp input such as motor speed at 450 rpm can be expressed as belonging to partly fuzzy set as low speed, medium speed and partly to high Speed with varying degrees of membership. This transformation allows the system to handle uncertainty, imprecision, and linguistic concepts such as low, medium, or high that are difficult to express in strict mathematical terms.

Fuzzification ensures that real-world inputs, which are often noisy or imprecise, are translated into a form suitable for reasoning and decision-making in fuzzy logic controllers. The choice of membership functions such as triangular, trapezoidal, Gaussian, or sigmoidal directly influences the quality and accuracy of fuzzification Jain and Sharma (2020).

2.3.2. Fuzzy Rule Base

The rule base is the core knowledge representation in a fuzzy logic system. It consists of a collection of fuzzy IF–THEN rules that describe the relationship between input variables and output variables in linguistic terms. These rules capture expert knowledge, human reasoning, or experimental observations in a form that the fuzzy inference engine can use for decision-making (Jia *eta l.,* 2024).

$$R_k: \text{IF } (x_1 \text{ is } A_1) \text{ AND } (x_2 \text{ is } A_2) \text{ THEN } (y \text{ is } B) \qquad \dots (7)$$

Where, x_1, x_2, are input variables A_1, A_2, are fuzzy sets for inputs, y is output variable and B is fuzzy set for output.

2.3.3 Fuzzy Inference Engine

Fuzzy inference is the process of mapping given inputs to corresponding outputs using fuzzy logic, it also refers to the region of a fuzzy set where the membership function has the maximum value of 1 and is responsible for simulating human reasoning by interpreting and processing fuzzy rules as shown in figure 5. It takes the fuzzified input values, which have been converted into degrees of membership, and applies them to the rule base of membership functions, fuzzy logic operators, and if- then rules. The inference engine evaluates these rules using fuzzy logic operators such as (AND, OR, and NOT) to determine how strongly each rule is activated (Lashin *et al.,* 2022).

2.3.4. Defuzzification

This is the final step in a fuzzy inference system where the fuzzy output set, expressed in linguistic and membership function terms, is converted into a single crisp numerical value that can be used for decision-making or control actions as indicated in figure 5. Since real-world systems require precise outputs such as motor speed, temperature setting, or control signal etc. defuzzification bridges the gap between fuzzy reasoning and practical implementation (Pancardo *et al*., 2021).

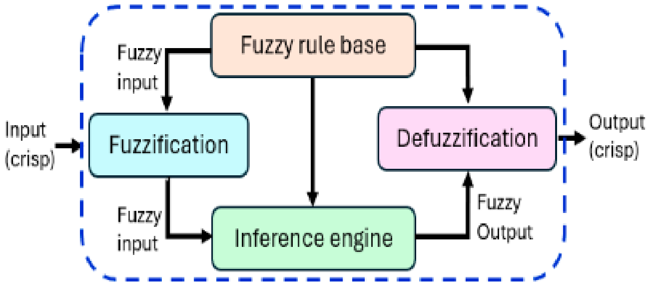

Figure 5: Structure of Fuzzy Inference System **Saatchi. (2024)**

2.4 DC Motor

A DC motor is an electromechanical device that converts electrical energy from a direct current source into mechanical rotational motion through the interaction of magnetic fields and current carrying conductors. DC motors are widely used in control applications due to their simple construction, high starting torque, ease of speed regulation, and fast dynamic response. Compared to AC motors, DC motors offer superior controllability, making them suitable for advanced control techniques such as fuzzy logic and PID controllers, particularly in applications requiring precise speed and torque control. Recent studies highlight that improvements in DC motor design and control strategies have further enhanced their efficiency, reliability, and performance in industrial and intelligent automation systems (Ibrahim and Baballe, 2024).

2.4.1 DC motor control

There are several methods available for controlling DC motors, which are generally categorized into three main groups:

1. Classical methods such as PID and PI controllers.
2. Modern approaches method, such as adaptive and optimal control techniques.
3. Artificial intelligence based methods including neural networks and fuzzy logic.

In linear control design, the method is typically based on the main application across a wide frequency range. However, linear controllers have limited effectiveness, as they cannot fully compensate for the nonlinear effects of the system. Direct current (DC) motors exist in different types, and various methods have been developed for controlling their speed. In this study, a DC motor was selected for speed control, where its speed was regulated by adjusting the supply voltage, particularly at speeds lower than the nominal value (Moghaddas *et al.,* 2011).

DC machines are known for their versatility. Through different configurations of shunt, eries, and separately excited field windings, they can be designed to exhibit a wide range of volt-ampere and speed-torque characteristics in both dynamic and steady-state conditions. Due to their ease of control, DC machines have been widely applied in systems that require a broad range of motor speeds and precise output regulation.

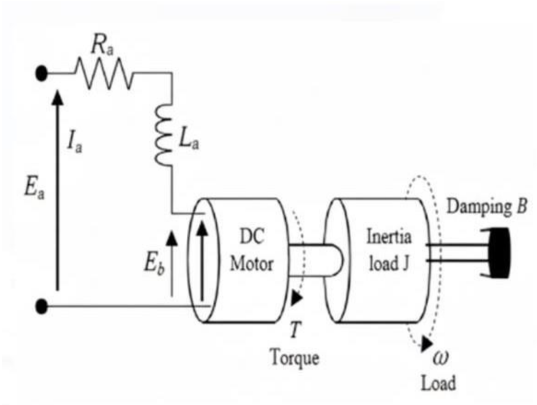

Figure 6: Structure of DC Motor.

$$V_t = R_a I_a + L_a = \frac{dI_a}{dt} + E_a \quad \text{... (8)}$$

$$T = J\frac{dw}{dt} + Bw - T_i \quad \text{... (9)}$$

$$T = K_T L_a \quad \text{... (10)}$$

$$E_a = K_a\omega \text{ While } \quad \varphi = \frac{d\omega}{dt} \quad \text{... (11)}$$

However the speed control equation is express in equation (10)

$$\omega = \frac{V - I_a R_a}{K\phi} \quad \text{... (12)}$$

Where

V_a = Armature voltage I_a = Armature current

R_a = Armature resistance T = Torque

L_a = Armature inductance E_b = Back EMF

K = EMF (motor) constant

□ = Motor angular speed

3.0 METHODOLOGY

The development of an adaptive speed control system requires an accurate mathematical representation of the intelligent groundnut pods threshing machine. The mathematical model establishes the dynamic relationship between the electrical input supplied to the drive motor and the rotational speed of the threshing drum while accounting for disturbances introduced by varying crop feeding rates. In this study, the threshing machine is modelled as an electromechanical system consisting of a separately excited DC motor, a power transmission mechanism, and a threshing drum subjected to variable load torque. The mathematical model serves as the basis for designing and evaluating the PID, and Fuzzy Logic controllers in MATLAB/Simulink.

3.1 Mathematical Model of the DC Motor

The DC motor (Series DC motor) provides the necessary torque and voltage speed regulation to adjust and ensure stable operation despite disturbances, thereby minimizing the groundnut pods breakage and maximizing threshing efficiency under varying operating conditions. DC motor is selected as the prime mover because of its high starting torque, wide speed regulation range, rapid dynamic response, and ease of electronic control. The motor converts electrical energy into mechanical rotational motion that drives the threshing drum. The dynamic behaviour of the motor is described by electrical and mechanical equations derived from Kirchhoff's voltage law and Newton's second law for rotational systems.

Applying Kirchhoff's Voltage Law as the electrical armature equation as adopted from Kuczmann (2024), Molina-Santana *et al* (2025).

$$V_a(t) = E_b(t) + I_a(t)R_a + L_a \frac{dI_a(t)}{dt} \qquad \ldots (13)$$

The electrical circuit of the armature consists of an armature resistance (R_a), armature inductance (L_a), applied armature voltage (V_a), armature current (I_a), and back electromotive force (E_b).

The EMF is proportional to the motor angular speed which is express as

$$E_b(t) = K_b\omega(t) \qquad \ldots (14)$$

K_b = EMF constant

□ = Motor angular speed

Substituting the EMF into the voltage equation (15)

$$V_a(t) = K_b\omega(t) + I_a(t)R_a + L_a\frac{dI_a(t)}{dt} \quad \ldots (15)$$

Hence, the mechanical torque/speed equation given by the dynamics governing the motor speed using Newton second law for rotation. as adopted from Kuczmann (2024), Molina-Santana *et al* (2025).

$$J\frac{d\omega(t)}{dt} + B\omega(t) = K_t I_a(t) - T_L \quad \ldots (16)$$

J = is the rotor inertial

B = is the viscous friction coefficient K_t = is the torque constant

T_L = Torque load (N/M)

Haven equation (15 and 16) as both Electrical and Mechanical Equation for the DC motor, now taken the Laplace Transform of both the two equation

$$V_a(s) = K_b\omega(s) + I_a(s)R_a + L_aI_a(s) \quad \ldots (17)$$

$$(J(s) + B)\omega(s) = K_tI_a(s) - T_L(s) \quad \ldots (18)$$

From equation (18) making $I_a(s)$ the subject and ignoring the load torque for voltage-speed by making it zero

$$I_a(s) = \frac{(J(s) + B)\omega(s)}{K_t} \quad \ldots (19)$$

Substitute equation (19) into equation (17) Recall

$$V_a(s) = K_b\omega(s) + (R_a + L_a)I_a(s)$$

$$\text{Therefore, } V_a(s) = K_b\omega(s) + (R_a + L_a)\frac{(J(s) + B)\omega(s)}{K_t} \quad \ldots (20)$$

Taken K_t as denominator and multiple it across

$$V_a(s)K_t = K_b K_t \omega(s) + (R_a + L_a)(J(s) + B)\omega(s) \quad \ldots (21)$$

Dividing both side by the coefficient of both K_t and ω

Therefore, the D.C motor voltage and speed equation is express as (22)

$$\frac{\omega(s)}{V_a(s)} = \frac{K_t}{(J_s + B)(L_a s + R_a) + K_t K_e} \quad \ldots (22)$$

The electrical and the mechanical equation are implemented in the Simulink environment to evaluate the performance of PI and Singleton fuzzy logic controller which further be express the equation as adopted from Molina-Santana *et al* (2025) as follows.

From equation (15) the armature voltage equation from the left hand side

$$V_a(t) - E_b(t) = I_a(t)R_a + L_a \frac{dI_a(t)}{dt} \quad \ldots (23)$$

Taken the Laplace Transform

$$V_a(s) - E_b(s) = I_a(s)R_a + \tau_a I_a(s) \quad \ldots (24)$$

Making $I_a(s)$ the subject

dt

$$I_a(s) = \frac{1}{R_a}\frac{1}{\tau_a s + 1}[V_a(s) - E_b(s)] \quad \ldots (25)$$

Where $\tau_a = \frac{L_a}{R_a}$ ... (26)

To match the model Simulink block $\frac{1}{R_a(\tau_a s + 1)}$

Therefore the Electrical model block for Simulink is given as $= \frac{1/R}{L_a s + 1}$ ... (27)

While for electromagnetic torque block equation

$$T_e(t) = K_t I_a(t) \quad \ldots (28)$$

The mechanical dynamic from the right hand side taken from equation (21)

$$J\frac{d\omega(t)}{dt} + B\omega(t) = T_e(t) - T_L(t) \quad \ldots (29)$$

Taken Laplace Transform and making the ω subject of the formula

In summary $\omega(s) = \frac{1}{B} - \frac{1}{\tau_m s + 1}\left[T_e(s) - T_L(s)\right]$ ... (30)

$\tau_m = \frac{J}{B}$ ... (31)

To match the model Simulink block $\frac{1}{B(\tau_m s + 1)}$ ... (32)

Furthermore, the mechanical model block for Simulink is given as = $\frac{1/B}{Js + 1}$ ... (33)

Therefore, equations (23 and 33) were used for the evaluation of the performance of fuzzy logic controller in the Simulink model, and the table 1 expresses the DC motor (Series DC motor) parameter for MATLAB/Simulink for purpose simulation.

Table 1: DC Motor (Series DC motor) Parameters

PARAMETER	DESCRIPTION	VALUE
V_a	Armature Voltage	220 V
L_a	Armature Inductance	0.0025 H
R_a	Armature Resistance	0.5 Ω
J	Mechanical Inertia	0.0013 kg·m²
B	Viscous Damping Coefficient	0.001 N·m·s/rad
T	Load Torque	21 N·m
□	Rate Speed	1800 rpm

3.2 Controller

A DC motor is control by a controller, such singleton fuzzy controller, is used to regulate the speed and torque of a DC motor by adjusting the applied voltage to ensure stable and accurate operation. In an intelligent groundnut threshing machine, the controller ensures that the motor maintains a steady and precise rotational speed despite variations in load caused by different groundnut varieties such as texture (hardness), size, (clearance). The controller receives a reference speed, often determined by the optimal operating speed for each groundnut variety, and continuously compares it with the actual measured speed. Based on this comparison, the controller automatically increases or decreases the motor's input power to compensate for disturbances and maintain smooth, reliable operation. By providing precise speed regulation, the DC motor enhances threshing efficiency, reduces groundnut seed damage, and ensures consistent machine performance under diverse working conditions.

3.2.1 PID Controller

The PID controller is used to regulate the motor voltage that match the optimal threshing speed require by each groundnut variety. These varieties are dependent to reference speeds that are generated by a fuzzy logic system and the PID controller continuously compares this reference speed with the actual motor speed and compute the error (e) and the change in error (Δe) using proportional, integral, and derivative actions, the controller rapidly corrects speed deviations, eliminates steady-state errors, and stabilizes the system during sudden load variations caused by differences in groundnut texture, size, hardness, and feeding rate. By adjusting motor voltage and torque in real time, the PID controller ensures smooth operation between varieties, maintains consistent speed. However, the reference signal enters a classical PI controller whose transfer function is

$$.PI(s) = K_P + \frac{K_i}{s} \quad \ldots (34)$$

The PI controller receives error and generates the control action

$$.\ Error\ e(t) = r(t) - y(t) \quad \ldots (35)$$

$$U(t) = K_P e(t) + K_i \int e(t)dt \quad \ldots (36)$$

Therefore, the control signal passes through a first-order dynamic armature voltage plant which is been recall from equation (43) as electrical model block for Simulink evaluation.

$$.\ G_1(s) = \frac{1/R}{L_a s + 1} \quad \ldots (37)$$

The output of this stage is further modified by another first-order system of mechanical dynamic of torque which is also recall from equation (33) as mechanical model block for Simulink used

$$G_2(s) = \frac{1/B}{Js + 1} \quad \ldots (38)$$

$$\text{Hence,}\quad G_1(s) = \frac{1/0.5}{0.0025s + 1} \text{ and } G_2(s) = \frac{1/0.001}{0.0013s + 1}$$

The figure (7) ensures that any deviation from the desired output automatically adjusts the motor speed, the integrated fuzzy and PI plant system provides an intelligent adaptive controller where fuzzy logic selects the appropriate operating rules condition based on groundnut variety properties, while the PI controller system stabilize the system to achieve the desired output voltage for efficient performance.

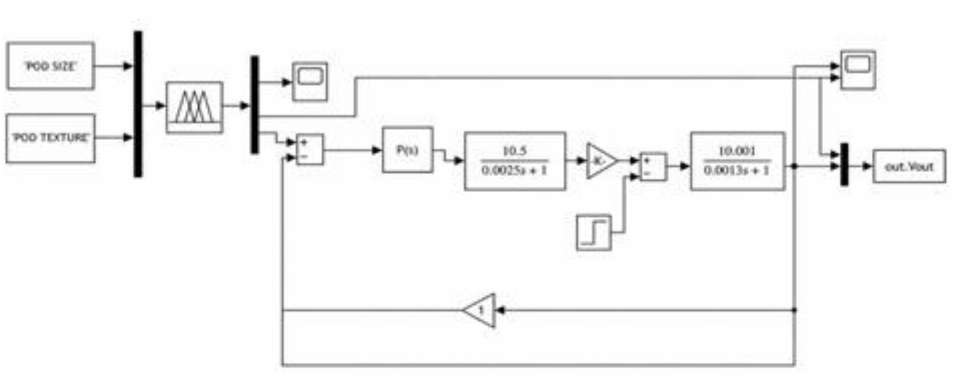

Figure 7: Simulink Diagram of PI Controller.

3.2.2 Fuzzy logic DC motor speed control

The Fuzzy Logic Control (FLC 2) regulates the DC motor speed to a high precision regulator within a closed-loop feedback system. This also compares the reference speed signal provided by system against the actual speed feedback from the DC Motor to identify the current error and the rate at which change error. By processing these two variables through a fuzzy inference system, it can intelligently adjust the voltage output to compensate for load variations. The Fuzzy logic approach allows for smoother and fast stability.

3.2.2.1 Fuzzification

The fuzzification process in Fuzzy Logic Controller (FLC) transforms the crisp input signals related to groundnut pod texture and speed control error into linguistic fuzzy variables that guide the motor speed adjustment. In this control structure, the texture characteristics of the groundnut pods, which may range from soft to hard shell, influence the reference speed generated by the system. These pods texture properties affect the resistance during threshing, where harder textured pods require higher impact force i.e high motor speed, while soft pods require lower speed to minimize groundnut seed damage. The output reference speed from the system is compared with the actual motor speed to generate an error signal (e) and change in error (Δe), which serves as the inputs to FLC. During fuzzification, these crisp error signals are mapped into fuzzy linguistic sets such as Negative Big (NB), Negative Medium

(NM), Negative Small (NS), Zero (Z), Positive Big (PB), Positive Medium (PM), and Positive Small (PS). Using predefined membership functions. This mapping allows the controller to interpret variations in pod texture indirectly through speed deviations and dynamically adjust the control signal sent to the DC motor.

3.2.2.2 Decision Making Logic (Fuzzy rule and Inference)

The fuzzy logic controller (FLC) functions as the real-time motor speed regulator that ensures accurate tracking of the reference speed that was generated by the motor for better groundnut threshing.

Each mapping describes how the fuzzy controller regulates output voltage behaviour, which in turn determines the selected DC motor speed. Therefore the voltage has a linguistics variable as follows; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z) i.e maintain the previous speed, Positive Big (PB), Positive Medium (PM), and Positive Small (PS) as showed in table 2

Table 2: Mapping Rules for FLC2.

E(t)⬇ (Δe)➡	**NB**	**NM**	**NS**	**Z**	**PS**	**PM**	**PB**
NB	NB	NB	NM	NM	NS	Z	PS
NM	NB	NM	NM	NS	Z	PS	PM
NS	NM	NM	NS	Z	PS	PM	PM
Z	NM	NS	Z	Z	Z	PS	PM
PS	NS	Z	PS	PM	PM	PM	PB
PM	Z	PS	PM	PM	PM	PB	PB
PB	PS	PM	PM	PB	PB	PB	PB

The matrix rule based, were used to developed the FLC 2 rules using equation (39) which is also expressed in the form of *IF, AND, THEN* statements

$$R = IF\ (x_1\ is\ A_1\ AND\ xi_2\ is\ Ai_2)\ THEN\ (y\ is\ B\ AND\ yi_2\ is\ Bi_2) \quad \ldots (39)$$

R1: IF Error is NB AND ΔError is NB THEN Voltage is Negative Big

R2: IF Error is NB AND ΔError is NM THEN Voltage is Negative Big

R3: IF Error is NB AND ΔError is NS THEN Voltage is Negative Medium

R4: IF Error is NB AND ΔError is Z THEN Voltage is Negative Medium

R5: IF Error is NB AND ΔError is PS THEN Voltage is Negative Small

R6: IF Error is NB AND ΔError is PM THEN Voltage is Zero **(maintained previous speed)**

R7: IF Error is NB AND ΔError is PB THEN Voltage is Positive Small R8: IF Error is NM AND ΔError is NB THEN Voltage is Negative Big

R9: IF Error is NM AND ΔError is NM THEN Voltage is Negative Medium R10: IF Error is NM AND ΔError is NS THEN Voltage is Negative Medium R11: IF Error is NM AND ΔError is Z THEN Voltage is Negative Small

R12: IF Error is NM AND ΔError is PS THEN Voltage is Zero **(maintained previous speed)**

R13: IF Error is NM AND ΔError is PM THEN Voltage is Positive Small R14: IF Error is NM AND ΔError is PB THEN Voltage is Positive Medium R15: IF Error is NS AND ΔError is NB THEN Voltage is Negative Medium R16: IF Error is NS AND ΔError is NM THEN Voltage is Negative Medium R17: IF Error is NS AND ΔError is NS THEN Voltage is Negative Small

R18: IF Error is NS AND ΔError is Z THEN Voltage is Zero **(maintained previous speed)**

R19: IF Error is NS AND ΔError is PS THEN Voltage is Positive Small R20: IF Error is NS AND ΔError is PM THEN Voltage is Positive Medium R21: IF Error is NS AND ΔError is PB THEN Voltage is Positive Medium R22: IF Error is Z AND ΔError is NB THEN Voltage is Negative Medium R23: IF Error is Z AND ΔError is NM THEN Voltage is Negative Small

R24: IF Error is Z AND ΔError is NS THEN Voltage is Zero **(maintained previous speed)**

R25: IF Error is Z AND ΔError is Z THEN Voltage is Zero **(maintained previous speed)**

R26: IF Error is Z AND ΔError is PS THEN Voltage is Zero **(maintained previous speed)**

R27: IF Error is Z AND ΔError is PM THEN Voltage is Positive Small R28: IF Error is Z AND ΔError is PB THEN Voltage is Positive Medium R29: IF Error is PS AND ΔError is NB THEN Voltage is Negative Small

R30: IF Error is PS AND ΔError is NM THEN Voltage is Zero **(maintained previous speed)**

R31: IF Error is PS AND ΔError is NS THEN Voltage is Positive Small R32: IF Error is PS AND ΔError is Z THEN Voltage is Positive Medium R33: IF Error is PS AND ΔError is PS THEN Voltage is Positive Medium R34: IF Error is PS AND ΔError is PM THEN Voltage is Positive Medium R35: IF Error is PS AND ΔError is PB THEN Voltage is Positive Big

R36: IF Error is PM AND ΔError is NB THEN Voltage is Zero **(maintained previous**

speed)

R37: IF Error is PM AND ΔError is NM THEN Voltage is Positive Small R38: IF Error is PM AND ΔError is NS THEN Voltage is Positive Medium R39: IF Error is PM AND ΔError is Z THEN Voltage is Positive Medium R40: IF Error is PM AND ΔError is PS THEN Voltage is Positive Medium R41: IF Error is PM AND ΔError is PM THEN Voltage is Positive Big R42: IF Error is PM AND ΔError is PB THEN Voltage is Positive Big R43: IF Error is PB AND ΔError is NB THEN Voltage is Positive Small R44: IF Error is PB AND ΔError is NM THEN Voltage is Positive Medium R45: IF Error is PB AND ΔError is NS THEN ΔError is Positive Medium R46: IF Error is PB AND ΔError is Z THEN Voltage is Positive Big

R47: IF Error is PB AND ΔError is PS THEN Voltage is Positive Big

R48: IF Error is PB AND ΔError is PM THEN Voltage is Positive Big R49: IF Error is PB AND ΔError is PB THEN Voltage is Positive Big

3.2.2.3 Defuzzification

However, in FLC, haven its inference system process the speed error and change in error and produces a fuzzy control signal that is defuzzified into a continuous motor armature voltage. This regulated voltage directly controls the motor drive, increasing the voltage for hard textured pods that require higher impact force and reducing the motor voltage for soft pods better threshing. Furthermore, the FLC simplifies its defuzzification stage using membership function where the output is represented by a set of crisp constant values, such as NB, NM, NS, Z, PS, PM, and PB that corresponds to a single fixed numerical output such as -200, -150, -100, 0, 100 and 150. Hence, Therefore, the control signal passes through a first-order dynamic armature voltage plant which is been recall from equation (40) as electrical model block for Simulink evaluation.

$$. G_1(s) = \frac{1/R}{L_a s + 1} \qquad \dots (40)$$

The output of this stage is further modified by another first-order system of mechanical dynamic of torque which is also recall from equation (33) as mechanical model block for Simulink used

$$G_2(s) = \frac{1/B}{Js+1} \qquad \ldots (41)$$

Hence, $G_1(s) = \dfrac{1/0.5}{0.0025s+1}$ and $G_2(s) = \dfrac{1/0.001}{0.0013s+1}$

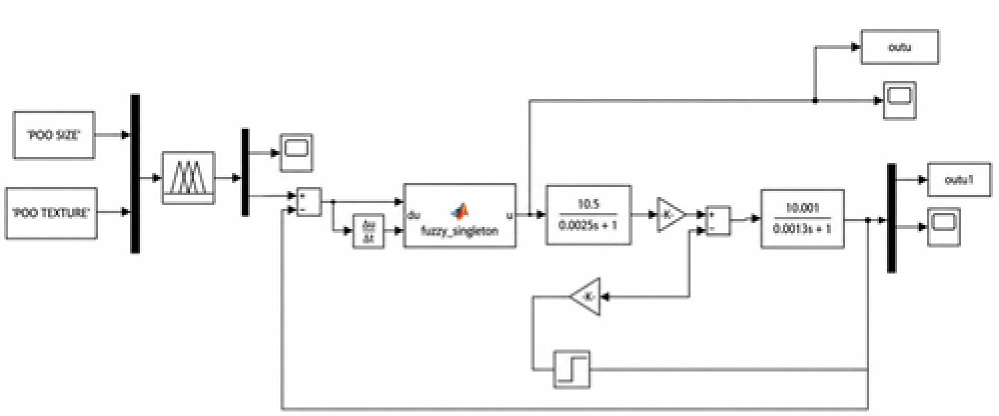

Figure 8: Simulink Diagram of Fuzzy Controller

Comparison between PID and Fuzzy Controller Model

However, the performance of the intelligent groundnut threshing machine was compared by developing two independent control strategies in MATLAB/Simulink of PID controller and a fuzzy logic controller with output functions as indicated in figure 9. The reference motor speed was compared with the measured speed to obtain the current speed error e and the change of error de. These two signals served as the inputs to the fuzzy controller implemented through the fuzzy singleton function. The fuzzy controller employed triangular membership functions for input variables and constant singleton values for the output. A complete 7×7 matrix rule base was evaluated, and the final control action was obtained.

Furthermore, a conventional PID controller was developed the PID gains were tuned to achieve acceptable stability and dynamic performance for the DC motor plant. Both controllers were applied to the same DC motor model, represented by two input sequential capturing the electrical and mechanical dynamics of the motor. Hence, the overall Simulink model was configured such that the fuzzy logic controller and the PID controller operated in separate but identical feedback loops. Both controllers received the same reference input and motor feedback signal, allowing direct comparison under identical simulation conditions. The control outputs from both controllers were then fed into identical motor models to observe their dynamic responses, including rise time, settling time, overshoot, and steady-state error.

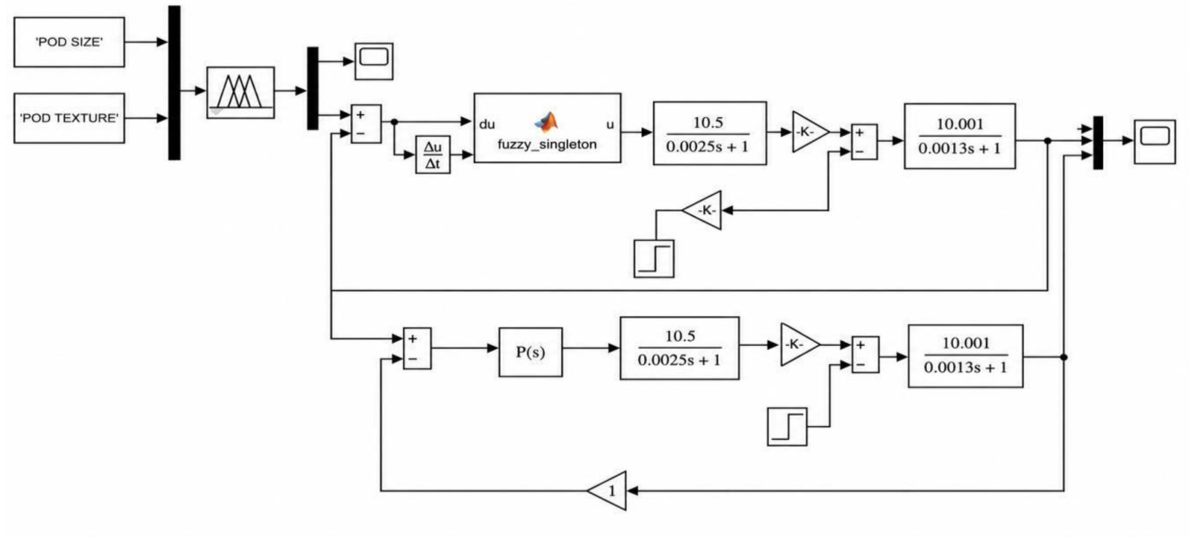

Figure 9: Combined Simulink Diagram of the Controllers.

4.0 RESULT

4.1 PID Controller Responses of DC Motor Speed

Controlled Motor Speed Tracking Performance at 50 rpm Reference

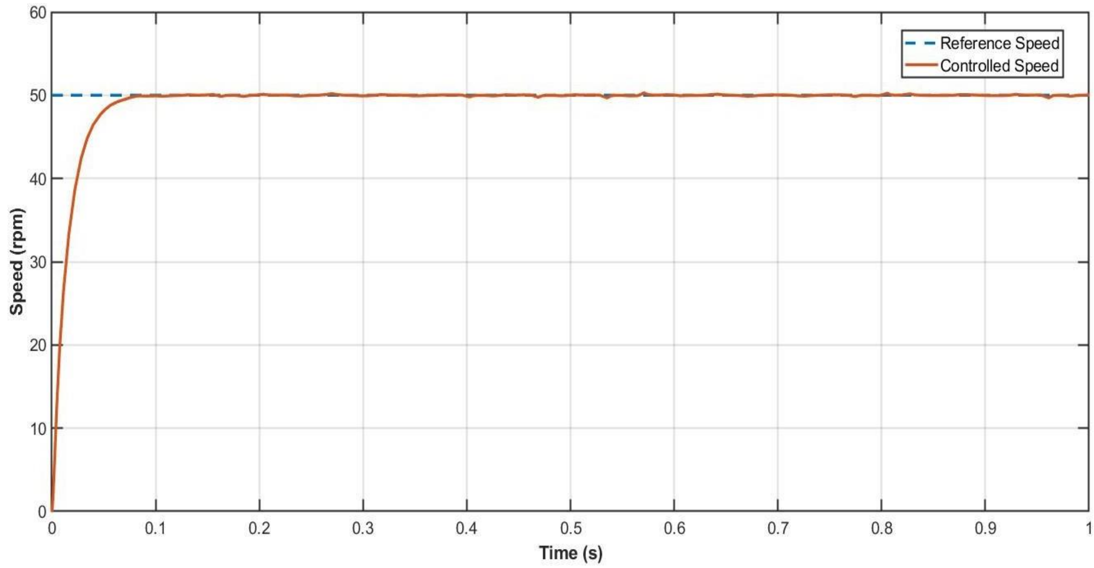

Figure 10: PID Output Controlled Speed Response at 50rpm.

Controlled Motor Speed Tracking Performance at 90 rpm Reference

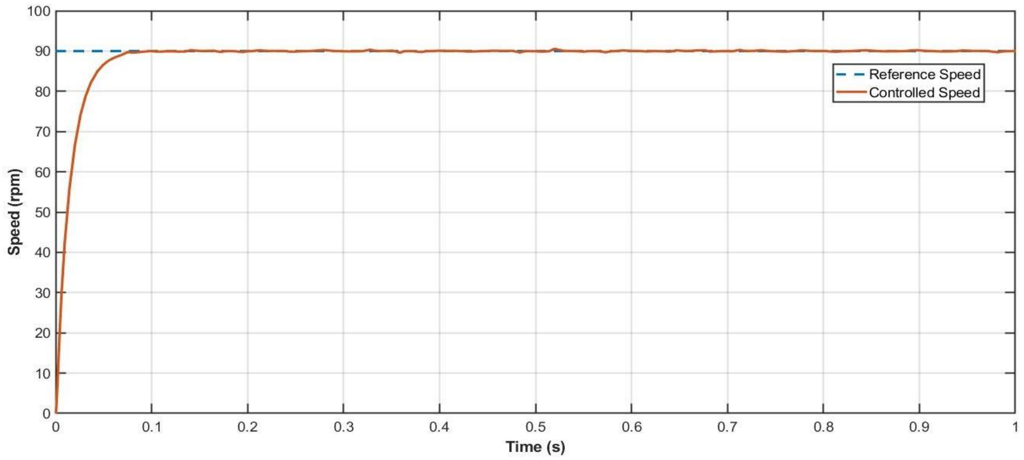

Figure 11: PID Output Controlled Speed Response at 90rpm

Controlled Motor Speed Tracking Performance at 150 rpm Reference

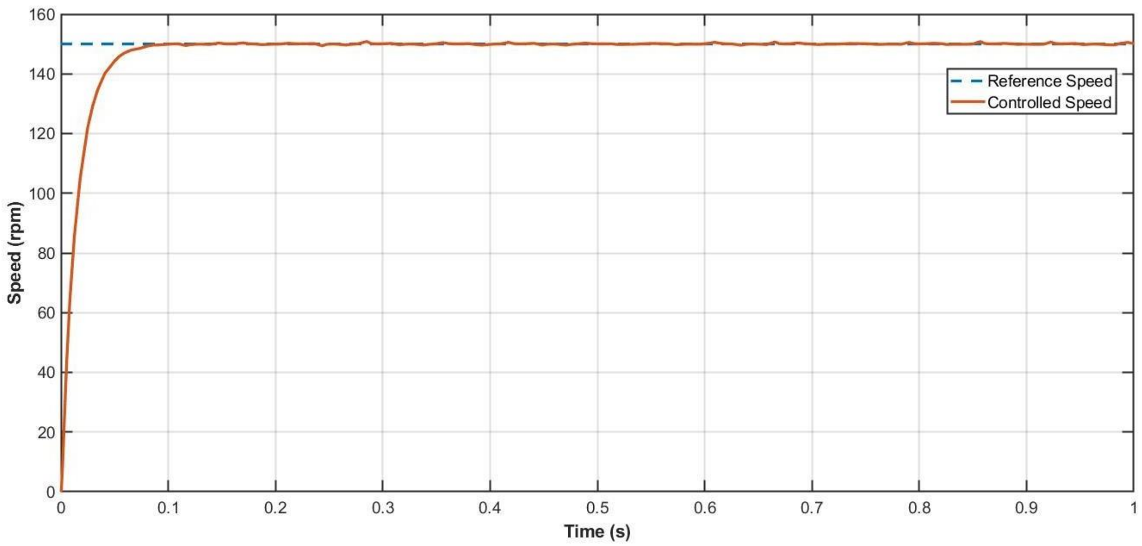

Figure 12: PID Output Controlled Speed Response at 150rpm.

4.2 Fuzzy Controller Response of DC Motor Speed

Controlled Motor Speed Tracking Performance at 50 rpm Reference

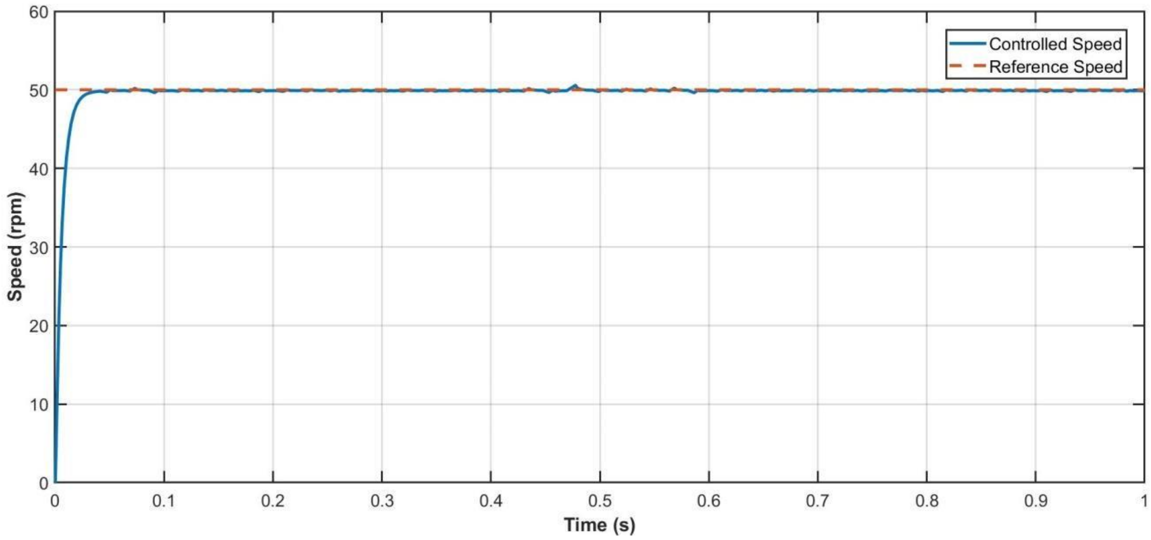

Figure 13: Fuzzy Output Controlled Speed Response at 50rpm

Controlled Motor Speed Tracking Performance at 90 rpm Reference

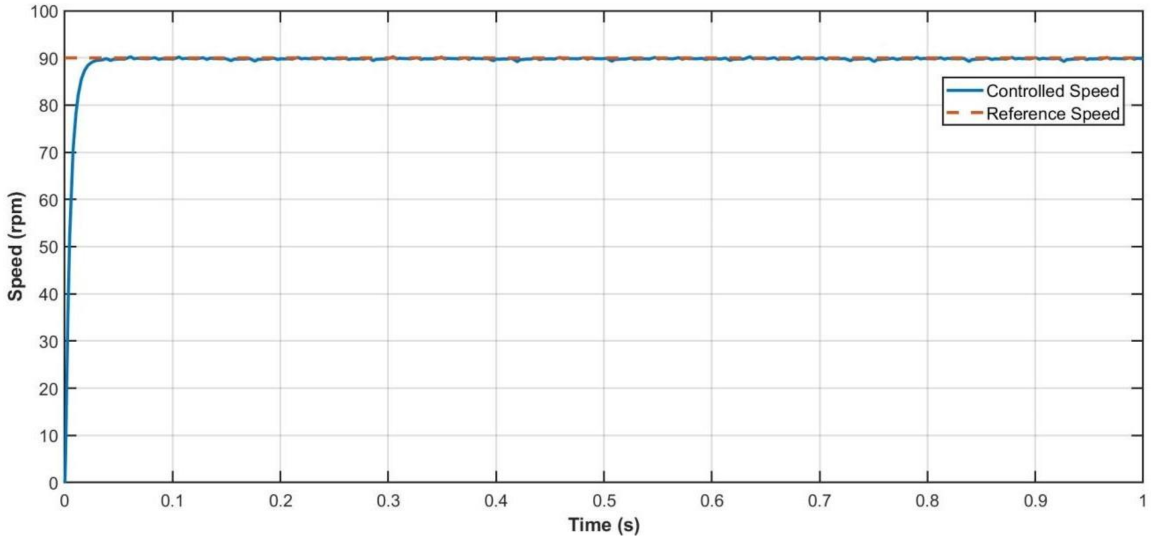

Figure 14: Fuzzy Output Controlled Speed Response at 90rpm

Controlled Motor Speed Tracking Performance at 150 rpm Reference

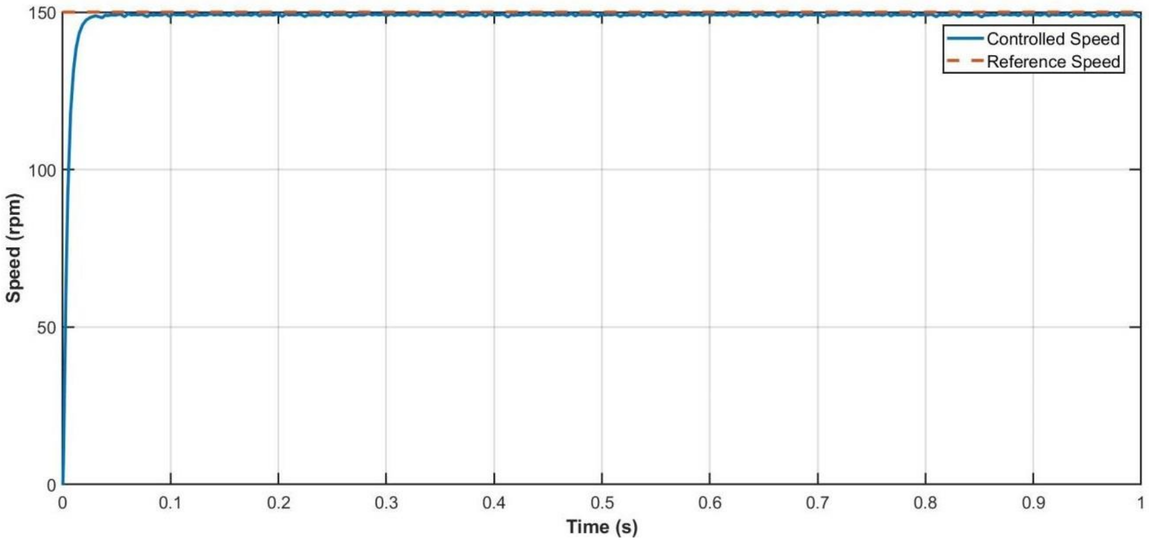

Figure 15: Fuzzy Output Controlled Speed Response at 150rpm.

4.3 Comparative between PID and Singleton Fuzzy Controller

Comparative Speed Tracking Performance of Fuzzy and PI Controllers at 50rpm Reference

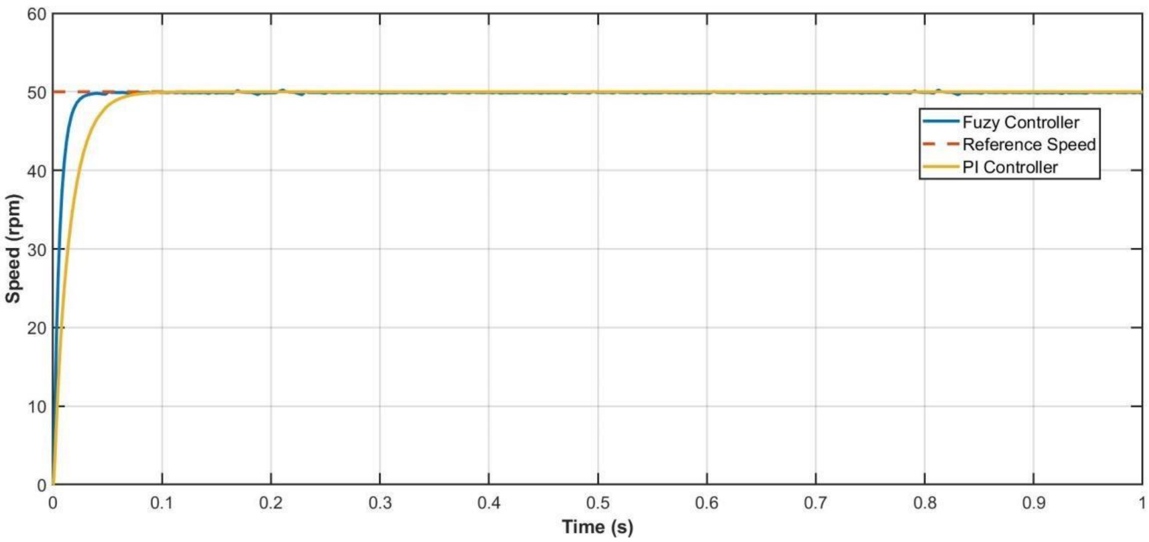

Figure 16: Comparative Output Controller of Speed Response at 50rpm

Comparative Speed Tracking Performance of Fuzzy and PI Controller at 90rpm Reference

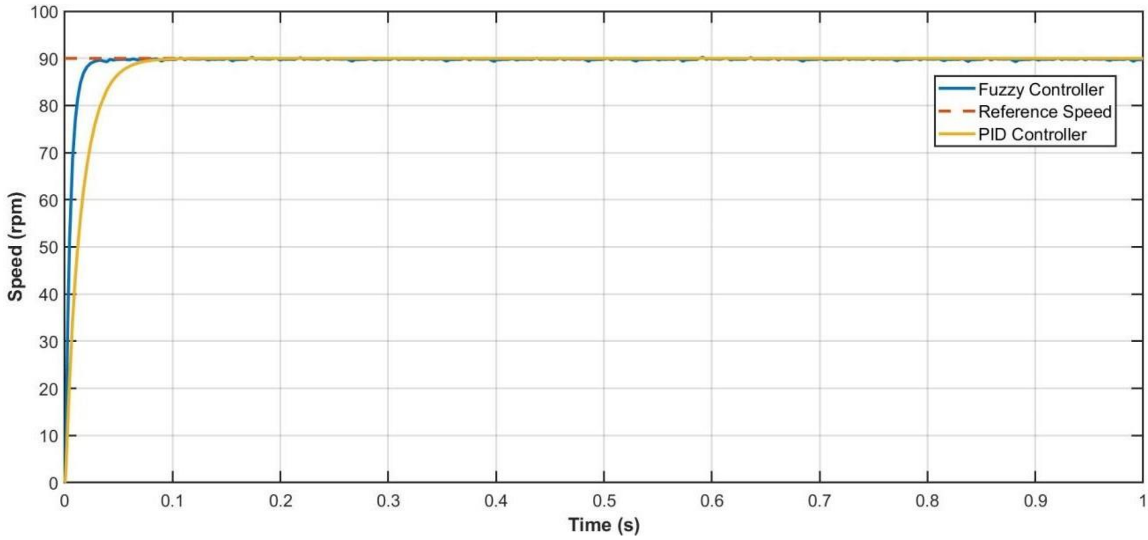

Figure 17: Comparative Output Controller of Speed Response at 90rpm

Comparative Speed Tracking Performance of Fuzzy and PI Controllers at 150rpm Reference

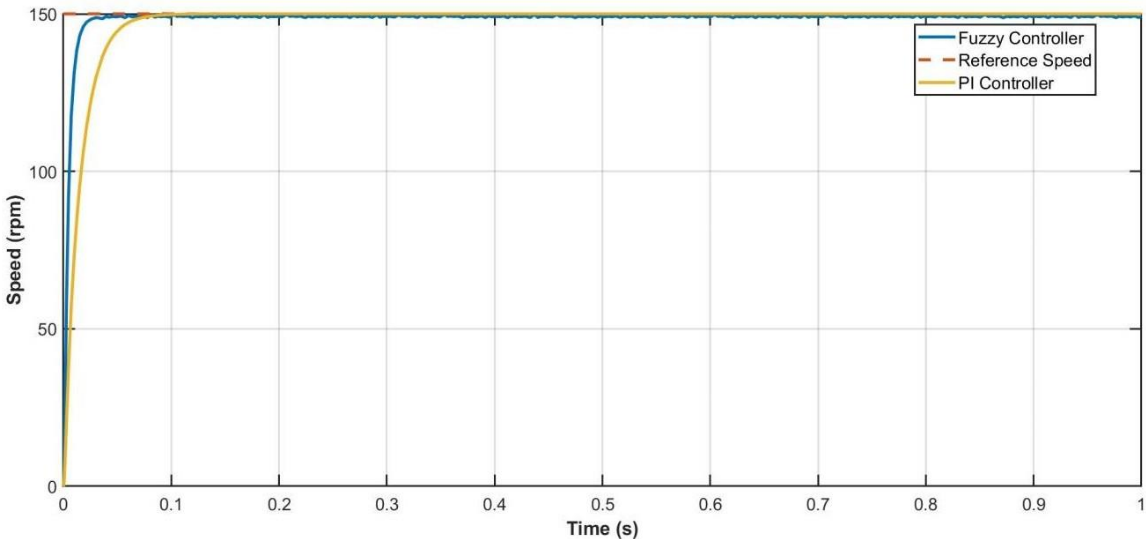

Figure 18: Comparative Output Controller of Speed Response at 150rpm

Comparative Speed Tracking Performance of Fuzzy and PI Controllers at 180rpm Reference Similarly, table 3 highlight the summary of the controllers responses in numerical form.

Table 3: Controller Response.

Reference Speed RPM	Rise Time(s) PI	Fuzzy	Settling PI	Time(s) Fuzzy	Overshoot (%) PI	Fuzzy	Steady PI	State (%) Fuzzy
50	0.035	0.014	0.099	0.040	0	0	0.2	0.4
90	0.035	0.012	0.150	0.049	0	0	0	0.3
150	0.035	0.011	0.099	0.017	0	0	0.66	0.66

5. DISCUSS

5.1 PI Controller Performances for DC Motor Speed Control

Figure 10 to 12 evaluates the dynamic response of a DC motor operating under PI speed control for four different reference speed commands i.e 50 rpm, 90 rpm, and 150 rpm. Each response contains two signals the reference speed (dashed blue line) and the controlled speed (orange line) in the horizontal time span of 1s each response. The PI controller demonstrates fast transient performance, zero stable steady-state error, and tracking reference with a minimal oscillation.

5.1.1 Transient Performance

The transient response of the DC motor under PI speed control provides the details how effectively the controller handles rapid changes in the reference input speed set points with the corresponding speed responses under the behaviour of rise time, settling time, and overshoot.

i. Rise Time

In all four response reference speed point, the DC motor responds rapidly to step changes in the reference speed. The controlled speed rises from 0 rpm to approximately 90% of the reference value within a range of about 0.035 seconds. These short rise times indicates that the PI controller provides sufficiently corrective action in the presence of large initial errors after been fine-tune. The fast rise time also expresses how rapid the model accelerate both appropriate controller tuning and the ability of the motor to develop sufficient torque for threshing efficiency. Moreover, the consistency of rise time across different reference values from the low speeds to higher speeds i.e 50 rpm to 150 rpm indicates that the controller retains its effectiveness over a wide operating range, without challenges at higher set points.

ii. Settling Time

From the initial rise, model was able to reduce the error and eliminate any oscillations along the responses. The settling time was able to settle the speed response within the small tolerance also the motor speed settles the reference within approximately 0.099 to 0.150 seconds. The PI model controller demonstrates that is not only fast but also well damped. The controller was able to drives the error down by gradually eliminating any steady state these processes resulted into a controlled trajectory that quickly converges to the desired speed without prolonged oscillations. However, there is little ripple disorder in the responses which is as a result of noise or minor electrical disturbances within the system.

iii. Overshoot

This is also another important transient performance of the controller, in which the speed attends without exceeding the reference value before settling across all four set responses results that the overshoot is absent (zero) or extremely small beyond measurable this indicates that the PI controller achieves a near critically damped response, that is a careful in balancing between speed response and its stability. The fact that overshoots remains negligible across all the different reference speeds this implies that the controller gains are not overly aggressive, that is to say the proportional and integral gains are high enough to provide fast correction but not so high as to drive the system into oscillatory behaviour. This performance strongly advice that the tuning strategy prioritized a robust and smooth response over extremely fast transient's performance, that ensure a stable, well-damped response that is suitable for real hardware implementation of intelligent groundnut threshing machine.

5.1.2 Steady State Performance

The steady-state performance of a PI model controller indicates its ability to maintain the desired output accurately over time of different responses. For DC motor system controller performance was evaluated in terms of steady-state error, noise sensitivity, and oscillatory behaviour for accurate, stable, and smooth threshing operation.

i. Steady-State Error

For all reference values examined in this experiment, ranging from 50 rpm to 150 rpm, the results clearly show that the controlled motor speed aligns almost perfectly with the reference command. The steady-state error in all cases is 0 to 0.66**%** this indicated that the motor output precisely matches the desired speed at the time when transient response has completely settled. This outcome confirms that the PI controller is highly effective in eliminating the long period of errors across the entire operation range. However the integral component ensures that the system maintains zero error control signal at the output.

ii. Noise and Oscillation

The performance of steady state accuracy is also base on the quality of the steady-state response where the noise sensitivity is established. The speed trajectories have shown a very small ripple around the reference set point after the output has settled. This minor fluctuation is as a result effects from the system components. The absence of sustained oscillations further confirms that the closed-loop system is stable and well damped that the integral was precise tuned that ensure a robust noise suppression for better threshing operation.

5.1.3 Reference Tracking

One of the important observation drawn from the stated figure 10 to 12 that the tracking reference of the PI controlled DC motor are consistent in their behaviour across all reference speeds. Whether operating at a low reference speed of 50 rpm to 90 rpm, moderate levels of 100 to 150 rpm, the controller exhibits nearly identical dynamic characteristics. The rise time, settling time, overshoot behaviour, and steady-state accuracy remain almost unchanged across this wide operating range. This uniformity in response is a strong indication the fundamental control objective where a system's output is continuously adjusted to match and follow a desired input signal.

5.2.1 Fuzzy Controller Performance for DC Motor Speed Control

The performance of the Fuzzy Controller was evaluated under four reference speed commands 50 rpm, 90 rpm, 150 rpm as shown in figure 13 to 15. The controlled speed closely tracks the reference trajectory, demonstrating the effectiveness of the fuzzy inference mechanism in handling nonlinearities and uncertainties in the DC motor system. Model demonstrates fast rise time, short settling time, and negligible overshoot that reflect the accuracy of transient and steady-state performance of the desired output speed.

5.2.2 Transient Response Characteristics of the Fuzzy Controller

The transient response behaviour of the Fuzzy Controller proves beyond doubt which accepts changes in reference speed while maintaining stability and smoothness of operation. In this study, the transient characteristics are evaluated using three fundamental performance indices rise time, settling time, and overshoot.

i. **Rise Time**

The transient performance across all reference values points, by the Fuzzy Controller achieves a remarkably rapid initial rise in motor speed. The controlled output reaches approximately 90% of the target speed within 0.011 to 0.014 seconds. This fast rise time indicates that the fuzzy controller is effective in control action during the transient performance and is capable of fixing error immediately after a step change in reference, thereby minimizing delay in achieving the desired operating output speed. This enhanced rise performance can be directly attributed to the nonlinear, rule-based structure of the fuzzy controller. Also the DC motor is quick to attain the desired operating speed that is desirable in the application of this intelligent threshing machine. The fuzzy controller is uniform in rise time across low, medium, and high reference speeds in term of fast rising.

ii. **Settling Time**

The Fuzzy Controller guides the motor speed smoothly into the steady state region that made the responses to have a short settling time. The motor output consistently settles within approximately 0.040 to 0.017 seconds for all tested reference commands. The reduced settling time reflects the ability of the fuzzy controller to transition seamlessly from aggressive error correction to fine steady-state regulation. As the error decreases, the fuzzy inference mechanism gradually shifts the control output toward rules associated with moderate or small errors. Another important factor contributing to the short settling time is the inherent damping effect provided by the fuzzy controller's rule base. Properly designed membership functions ensure that rapid changes in control output are naturally moderated as the system approaches the set point the result is a speed response that converges rapidly to the target value without overshoot. Moreover, the consistency of settling time across all reference speeds indicates that the fuzzy controller maintains robust dynamic performance regardless of operating point. The DC motor applications where rapid speed variations occur frequently with that ability to settle quickly after each command change ensures high responsiveness, improved productivity, and greater operational precision for this intelligent threshing machine.

iii. **Overshoot**

The overshoot characteristics of the Fuzzy Controller are particularly impressive. The response shows no significant overshoot is observed for any of the reference commands. The controlled speed approaches the reference smoothly and does not significantly exceed the set point i.e reference point. Minor steady-state ripples are visible observed however, these fluctuations are very small and do not constitute meaningful overshoot. The motor speed approaches the desired value asymptotically and smoothly without exceeding the target speed and because of this that is why is consider being favourable to the intelligent threshing machine for effective operation and other automation device.

5.2.3 Steady State Performance of the Fuzzy Controller

The steady-state performance of the Fuzzy Controller has maintain an accurate, stable, and reliable speed regulation, this steady state behaviour is evaluated in terms of steady-state error, noise sensitivity and ripple and strong robustness.

i. **Steady-State Error**

The response results stated in figure 16 to 18 that the controlled motor speed aligns almost perfectly with the reference trajectory, with the steady-state error effectively reduced between 0.3 to 1.1% across all reference commands speed point, this includes both from the low speed to high speed condition i.e 50 rpm to 150 rpm. The motor consistently achieves and maintains the desired speed without any persistent offset once the transient response has settled and this confirms the high precision of the fuzzy control system on speed regulated output. Unlike conventional linear controllers, the Fuzzy controller does not rely on an explicit integral term to eliminate steady-state error. Instead, zero steady-state error is achieved through the implicit integral like behaviour embedded within the fuzzy inference mechanism. The mapping of error and change of error through the membership functions and rule base allows the controller to apply corrective action continuously until the output fully matches the reference.

ii. **Noise Sensitivity and Ripple**

The steady state response is strongly influenced by the controller's sensitivity to noise and small disturbances. The speed trajectories response has a small oscillatory ripple. The amplitude of this ripple is extremely small relative to the full operating speed range and therefore does not affect the functional performance of the system. This small ripple is most likely attributable to measurement noise, sensor resolution limits, and minor electrical disturbances, Importantly, the fuzzy controller does not amplify this noise into large oscillations, which demonstrates that the rule base and membership functions provide a natural filtering effect on high-frequency disturbances. Most of the challenge been often is associated with nonlinear and rule-based controllers is chattering, which manifests as rapid switching or oscillations in the control signal around the set point.

iii. **Robustness and Set point Consistency of the Fuzzy Singleton Controller**

A notable outcome of the fuzzy control responses has a high level of consistency that is observed in a dynamic and steady state behaviour across all reference speed commands, at when the motor operates at low speed 50 to 90 rpm, moderate speed 100 to150 rpm, the controller exhibits nearly identical rise time, settling time, and steady-state characteristics. This uniformity in the performance of the model that indicates that the fuzzy controller is successfully accommodates the nonlinearities of the DC motor and maintains robust regulation over a wide operating range.

5.3 Comparative Performance Analysis of PI and Fuzzy Controller

The comparative evaluation of the PI controller and the Fuzzy controller was conducted across multiple reference speeds 50 rpm, 90 rpm, 150 rpm, in order to comprehensively assess their transient and steady state performance in DC motor speed regulation as show in figure 16 to 18 and table 3. These reference values were selected to represent low, medium, and high speed respectively as operating conditions for DC motor model control applications. The reason to this comparison is to determine not only the accuracy of each controller but also their dynamic responsiveness, robustness to operating variations. The comparison plots reveal clear distinctions in how each controller reacts to fast changes in reference command and how effectively each maintains stability thereafter.

5.3.1 Comparison of Transient Response

The performance of transient response in speed control systems that defines how quickly and effectively the system reacts to sudden changes in reference commands across different set point where Fuzzy controller consistently demonstrates faster transient behaviour when compared with the PI controller. The fuzzy controller reaches approximately 90% of the desired speed more rapidly than the PI controller.

i. Rise Time

The performance of the controllers in term rise time the Fuzzy Logic Controller (FLC) demonstrates a significantly faster response than the PI controller. From the obtained results stated in figure 16 to 18 it was observed that, the rise time of the PI controller lies within the range of approximately 0.035, reflecting the limitation imposed by its fixed linear structure. While, the Fuzzy Logic Controller achieves a much shorter rise time in the range of about 0.011 to 0.014 second, attributed to its nonlinear, rule-based control mechanism that applies stronger corrective action for large speed errors. This reduction in rise time confirms the superior transient tracking capability of the Fuzzy Controller compared to the conventional PI controller.

ii. Settling Time

The settling time further highlights the dynamic advantage of the fuzzy controller. The Fuzzy controller consistently stabilizes the motor speed and demonstrates superior speed and precision, achieving a settling time within a range of 0.040 to 0.017 seconds, whereas the PI controller typically requires 0.099 to 0.150 seconds often suffers from prolonged oscillations before reaching stability before fully settle. The differences may appear modest in absolute numerical terms, but Fuzzy controller provides a more robust and responsiveness than the PI Controller.

iii. Overshoot

The performance of the two controllers on overshoot of the responses model was compared that the PI controller comparative findings with Fuzzy controller clearly show that the Fuzzy has better overshoot and stability characteristics. The fuzzy controlled response maintains a smooth, steady convergence towards the intended reference speed in all evaluated speed response, exhibiting little to no overshoot. The motor speed slowly approaches but stays below the set point this behaviour shows how well the membership functions and rule base of the fuzzy controller modulate control effort as the system gets closer to the set point. The PI controller, on the other hand, occasionally exhibits slight overshoot, especially at higher reference speeds in these situations, the motor speed briefly surpasses the target value before progressively returning to the steady-state level. Even yet, the overshoot magnitude is still low and falls has within reasonable negligible boundary of zero overshoot.

iii. Steady State Accuracy and Robustness

Despite the differences observed in transient behaviour, both controllers demonstrate excellent steady-state accuracy. The responses were measured for the motor speed closely follows the reference signal, with virtually zero steady-state error for both control approaches. This confirms that both the PI and Fuzzy controllers are capable of eliminating long-term bias and maintaining precise regulation under constant operating conditions. For the PI controller, this steady-state error is primarily achieved through the action of the integral component, which accumulates the error over time and forces the steady-state output which is a good haven the highest value of 0.66% steady-state error among its responses which is better than Singleton, while For the fuzzy controller, steady-state error is achieved through its implicit integral behaviour, embedded within the structure of the fuzzy rule base and inference mechanism but have the highest 1.1 % among the various responses. However, despite this similarity in steady-state accuracy, the robustness of the PI Controller is clearly superior performance across all reference speeds, ranging from 50 rpm to 150 rpm. Thus, fuzzy controller maintain a low steady-state error of 1.1% this indicates strong tolerance to nonlinearities, parameter uncertainties, and operating-point variations inherent in DC motor systems.

6. CONCLUSION

This study successfully developed and evaluated PI and Singleton Fuzzy Logic Controllers for adaptive DC motor speed regulation in an intelligent groundnut threshing machine using a MATLAB/Simulink model of a 220V series DC motor with an armature inductance of 0.0025H, armature resistance of 0.5 Ω, rotor inertia of 0.0013 kg·m², and load torque of 21 N·m. The simulation results demonstrated that the Singleton Fuzzy Logic Controller achieved superior dynamic performance over the PI controller by reducing the rise time from 0.035 s to 0.011 to 0.014 s and the settling time from 0.099 to 0.150 s to 0.017–0.049 s, while maintaining 0% overshoot and a low steady-state error of 0.3 to 1.1% across reference speeds of 50 to 150 rpm. Although both controllers provided accurate speed tracking, the fuzzy controller exhibited greater robustness and faster adaptation to speed variations, making it a more suitable control strategy for intelligent groundnut threshing machines, where rapid response, stable operation, and improved threshing performance are essential.

REFERENCE

1. Bello, B., Tokan, A., Jiya, D., & Musa, A. (2021). *Design Model of Automated GroundnutThreshing Machine*. Majlesi Journal of Mechatronic Systems, 8(1), 19-24.ms.majlesi.info
2. Bello,B., A. Tokan, J.D. Jiya . (2019) Design and Fabrication of an Automated Groundnut Threshing Machine International Journal of Mechatronics, Electrical and Computer Technology (IJMEC) PISSN:2411-6173 – EISSN:2305-0543 Vol.9(34), PP 4383 4400
3. Chiu, C.-H., and Peng, Y.F. (2019). Design of Takagi-Sugeno fuzzy control scheme forrealworld system control. *Sustainability, 11*(14), 3855. https://doi.org/10.3390/su11143855
4. Ibrahim, I. A., and Baballe, M. A. (2024). *The DC motor advantages: Key benefits and improvements over AC motors. Global Journal of Research in Engineering andComputer Sciences, 4(5). https://gjrpublication.com/gjrec* ISSN: 2583-2727 (Online)
5. Jain, A., and Sharma, A. (2020). Membership function formulation methods for fuzzy logic systems: A comprehensive review. *Journal of Critical Reviews, 7*(19), 2394–5125.
6. Jia, Y., and Wang, Z. (2024). Application of artificial intelligence based on the fuzzy control algorithm in enterprise innovation.
7. https://doi.org/10.1016/j.heliyon.2024.e28116

8. Judy., (2025). *Explore key types of DC motors & their uses*. GR Winding Blog.https://grwinding.com
9. Kabir, M., Ahmad, H., Jaiaid, M., & Rahma, M. (2024). *Image contrast enhancement using fuzzy logic with metaheuristics*. In *Proceedings of the SysS '24* (Khulna,Bangladesh).
10. Kuczmann, M. (2024). *Review of DC motor modeling and linear control: Theory with laboratory tests*. Electronics, 13(11), 2225. https://doi.org/10.3390/electronics13112225
11. Lashin, M. M. A., Khan, M. I., Ben Khedher, N., and Eldin, S. M. (2022). Optimization of display window design for females' clothes for fashion stores through artificialintelligence and fuzzy system. *Applied Sciences, 12*(22), 11594.https://doi.org/10.3390/app122211594
12. Lin, J., Zhou, J., Lu, M., Wang, H., & Yi, A. (2020). Design of robust adaptive fuzzycontroller for a class of single-input single-output (SISO) uncertain nonlinearsystems. *Mathematical Problems in Engineering, 2020,* Article 6178678.https://doi.org/10.1155/2020/6178678
13. Lingxiao, L., and Pang, S. (2020). An implementation of the adaptive neuro-fuzzyinference system (ANFIS) for odor source localization. In *Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*
14. Lokossou, J. C., Affognon, H. D., Singbo, A., Vabi, M. B., Ogunbayo, A., Tanzubil, P.,
15. Segnon, A. C., Muricho, G., Desmae, H., & Ajeigbe, H. (Year). Welfare impacts of improved groundnut varieties adoption and food security implications in the semiarid reas of West Africa.
16. Moghaddas, M., Dastranj, M. R., Changizi, N., and Rouhani, M. (2011). Robust control of DC motor using fuzzy sliding mode control and genetic algorithm. *The Journal ofMathematics and Computer Science, 2*(4), 572–579.
17. Molina-Santana, E., Iturralde Carrera, L. A., Álvarez-Alvarado, J. M., Aviles, M., & Rodríguez Resendiz, J. (2025). *Modeling and control of a permanent magnet DC motor: A casestudy for a bidirectional conveyor belt's application***.** Eng, 6(3),
18. 42.https://doi.org/10.3390/eng6030042
19. Onuoha, S. N., Enaboifo, M. A., & Ibrahim, R. (2022). Development of Cowpea Thresher. *Nigerian Journal of Technology, 41*(2), 385-395. nijotech.com
20. Pancardo, P., Hernández-Nolasco, J. A., Wister, M. A., and Garcia-Constantino, M.

(2021). Dynamic membership functions for context-based fuzzy systems. *IEEE Access, 9,* 29665–29676. https://doi.org/10.1109/ACCESS.2021.3058892

21. Saatchi, R. (2024). Fuzzy logic concepts, developments and implementation.
22. ,https://www.mdpi.com/journal/information*Information,15*(10), 656.https://doi.org/10.3390/info15100656
23. Saleh, A., Fashina, A. B., & Akande, F. B. (2022). Design, Construction and Performance Evaluation of Groundnut Decorticator. *FUDMA Journal of Sciences*.
24. https://doi.org/10.33003/fjs-2022-0601-847 fjs.fudutsinma.edu.ng
25. Setiawan, A., Arumi, E. R., and Sukmasetya, P. (2020). Fuzzy membership functions analysis for usability evaluation of online credit hour form. *Journal of*
26. *Engineering Science and Technology, 15*(5), 3189–3203.
27. Tomasiello, S., Alijani, Z. (2021). Fuzzy-based approaches for agri-food supply chains: a mini review. Soft Computing, 25(11), 7479–7492
28. Wang, Q., Wang, X., Fang, C., Jiao, J. Fuzzy image clustering incorporating local and region level information with median memberships, Applied Soft Computing 105 (2021) 107245
29. Widayat, I. W., Arsyad, A. A., Mantau, A. J., Adhitya, Y., and Köppen, M. (2025). Fuzzymethods in smart farming: A systematic review. *Informatica, 36*(2), 453–489.https://doi.org/10.15388/24-infor579