Comparative Study of Optimised Artificial Intelligence Based First Order Sliding Mode Controllers for Position Control of a DC Motor Actuator

Submitted: 13th June 2016; accepted 26th August 2016

Bassey Etim Nyong-Bassey, Benjamin Akinloye

DOI: 10.14313/JAMRIS_3-2016/25

Abstract:
This paper aims at critically reviewing various sliding mode control measures applied to Permanent Magnet DC Motor actuator for position control. At first, a hybrid sliding mode controller was examined with its advantages and disadvantages. Then, the smooth sliding mode controller in the same manner. The shortcomings of the two methods were overcome by proper switch design and also using tanh-sinh hyperbolic function. The sliding mode controller switches on when either disturbance or noise is detected. Genetic Algorithm Computational tuning technique is employed to optimize the gains of the controllers for optimal response.The performance of the proposed controller architecture, as well as the reviewed controllers, have been compared for performance evaluation with respect to several operating conditions. This includes load torque disturbance injection, noise injection in a feedback loop, motor nonlinearity exhibited by parameters variation, and a step change in reference input demand.

Keywords: *adaptive fuzzy control, DC motor position control, genetic algorithm, particle swarm optimization, sliding mode control*

1. Introduction

The purpose of the paper is to provide a critical review while presenting new elements with respect to some fundamental issues and challenges related to Adaptive Fuzzy Sliding Mode Controller for position control of a DC motor actuator.

The utilization of the DC motor extends across a wide gamut of industrial applications. This includes; rolling mills, electric cranes, electric locomotives; trains, trams, and robotics [1–8].

For precision, a controller of high performance is preferred and characterized by good load torque disturbance rejection, and exhibits minimal or no overshoot [2, 9].

The sliding mode controller (SMC) is well known for it's robust to disturbance and insensitive to parameter variation. In [10] fully decentralized fuzzy sliding mode controller (FDFSMC) technique was used for the stabilization of a Quadrotor UAV attitude. The choice of the SMC scheme was based on the existing literature on this nonlinear control approach because of its robustness against the incertitude and external perturbation. However, the chattering phenomenon is the major drawback of the proposed techniques of FDFSM.

The proportional integral derivative (PID) Controller is very common for industrial control system use. The gains of the PID controller can be tuned by Ziegler-Nichols (Z-N), root locus pole placement, trial and error method. Also, it may be tuned by some form of optimization technique such as Genetic algorithm or particle swarm optimization.

The optimal response is not guaranteed with the PID controller after tuning the gains as a degree of overshoot, steady state error and long settling time are often a trade-off for a fast rise time [11]. Author [12] worked on optimization of performance specifications of industrial permanent magnet DC motor based on particle swarm optimization (PSO) method. The enhanced performance was achieved by optimally tuning of the Proportional-Integral-Derivative (PID) controller gains. The proposed method was applied to the linear model of PMDC motor. The performance of the suggested objective function of PSO was then compared with the performance of other objective functions of PSO and conventional Ziegler Nichols (ZN) method. The comparison clearly indicated the effectiveness of the optimisation technique.

During motor operation, since the conventional PID controller gains are tuned offline they remain fixed in the prevalence of nonlinearities. Hence, the output response is affected [11]. The fuzzy logic controller is suitable for negating the effect of non-linearity which may result from mechanical wear and tear in the motor drive as well as disturbance. Author [13] applied both the conventional SMC and a decentralized fuzzy architecture (FDFSMC). The decentralized architecture comprised of a fuzzy SMC in conjunction with an adaptive Fuzzy-PID. A fuzzy inference mechanism was used to reduce the chattering phenomenon inherent in the conventional SMC. However, the sign function used for switching the sliding mode controller exhibited chattering.

In [14] the output of a PID controller is used as to establish the sliding surface of a Fuzzy SMC as applied in position control of a DC motor. Thus the controller structure incorporates a parallel control action by PID and Fuzzy logic Sliding mode (PID+FSMC).

From previous study, various methods based on fuzzy sliding mode control have been deduced for position control of a DC motor. However, not one of them can be said to be completely satisfactory. This work, therefore, carries out a comparative study on various sliding mode control methods. The strengths

and shortcomings associated with each fuzzy sliding strategy has been underpinned. Modifications to the fuzzy sliding mode reaching law and switching function has been proposed in this work. The proposed model has been compared with the existing models with respect to robustness to nonlinearities and exogenous disturbance as well as setpoint tracking.

2. Mathematical Modeling of DC Motor

The mathematical equations for modeling the DC motor is established by Kirchhoff's voltage law and Newton's second law of motion for the armature circuit and mechanical section in turn. The DC motor schematic diagram is shown in Figure 1.

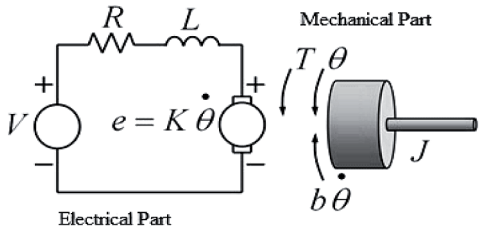

Fig. 1. PMDC motor schematic [15]

By Kirchhoff's voltage law, the armature circuit equation is determined from first principle:

$$L\frac{di}{dt} + Ri = V - E \tag{1}$$

$$E = K_e\,\omega \tag{2}$$

Consequently, by Newton's second law of motion, the mechanical rotor equation is given:

$$J\ddot{\theta} = T - b\dot{\theta} \tag{3}$$

Where,

$$\omega = \frac{d\theta}{dt} = \dot{\theta} \tag{4}$$

$$\mathrm{T} = K_t i. \tag{5}$$

$$K = K_t = K_e \tag{6}$$

By re-arranging equations (1) and (3):

$$L\frac{di}{dt} + Ri = V - K\omega \tag{7}$$

$$J\dot{\omega} + b\omega = Ki \tag{8}$$

By applying Laplace transform to equation (7) and (8) the equations become:

$$LsI(s) + RI(s) = V - KW(s) \tag{9}$$

$$JsW(s) + BW(s) = KI(s) \tag{10}$$

By the elimination of I (s) in equation (8) and (9) the transfer function of the input voltage to output speed of the DC motor is given as:

$$G(s) = \frac{W(s)}{V(s)} = \frac{K}{\{(JS+B)(LS+R)+K^2\}} \tag{11}$$

Similarly, the transfer function encompassing output position and input voltage is derived by substituting equation (2.4) into equation (2.10) as shown:

$$G(s) = \frac{\theta(s).}{V(s).} = \frac{K}{\{(JS+B)(LS+R)+K^2\}S} \tag{12}$$

The DC motor state space representation is shown in canonical form:

$$\begin{bmatrix}\dot{\theta}\\ \dot{\omega}\\ i\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ 0 & -B/J & K/J\\ 0 & -K/L & -R/L\end{bmatrix}\begin{bmatrix}\theta\\ \omega\\ i\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 1/L.\end{bmatrix}V \tag{13}$$

Where,

- I is the current in the armature winding (In ampere),
- E is the armature winding back E.M.F (In volt),
- R is the armature winding resistance (In ohm),
- V is the armature voltage (In volt),
- T is mechanical torque (In Nm),
- K_e is the back E.M.F constant (Nm/A),
- L is the armature winding inductance (In Henry),
- J is moment of inertia of the motor (In Kgm^{-2}),
- B is the motor's coefficient of frictional (In Nm/(rad/sec)),
- ω is angular velocity of the mechanical rotor shaft (rad/sec),
- θ is angular position of the mechanical rotor shaft (rad).

3. PID Controller

The PID controller is popular because it offers a fairly good response, it is modest and simple to build. However, due to non-linearity, model uncertainty, and exogenous disturbance, the linear PID controller falls short in performance chiefly due to static controller gains [3, 16].

The mathematical representation of the PID controller is:

$$U_{PID}(t) = K_P e(t) + K_I \int e(t)dt + Kd * de(t)/dt + \\ + Kd * de(t)/dt \tag{14}$$

Where: e signifies error, K_p denotes the proportional gain constant, K_D represents the derivative gain constant, K_I is the integral gain constant, T_d is the derivative time and T_i is the integral time [3].

4. Fuzzy Logic Controller

In 1973, L.A Zadeh introduced the fuzzy logic theory, thereafter Mamdani in 1974, implemented it for controlling systems structurally complex to model. In a typical control problem the input to the fuzzy controller is the error signal; error and change in error [3, 5]. The Fuzzy controller comprises the fuzzification interface, rule base, the inference mechanism and defuzzification interface. The optimum response depends on upon the tuning of the scaling gains, the rule base, and the membership functions [2, 9, 17, 18].

A. Fuzzification

The input values depicting change-in-error and error are received by this interface. The values are mapped according to the linguistic rules and converted to degree of one or more membership functions [20].

B. Membership Function

The shape of the membership function (MF) usually depends on the control variable. The triangular and trapezoidal MFs are easy to implement hence commonly selected. At the initial design, a starting point is to keep the MFs equal also they have to overlap by 50% in order to avoid null firing rule or undefined function. The Fuzzy set typically consists of "NB (Negative Big)", "NM (Negative Medium)", "Z (Zero)", "PM (Positive Medium)" and "PB (Positive Big)" membership functions [19].

C. Rule Base

The fuzzy rules are heuristic logical rules that depend basically on Operator's experience with the system. Deducing the rules may also involve observing the phase plane of the error and derivative error and consequently the step response of the closed loop system.

A typical fuzzy rule is of the form:

If Error is Positive Big (NB) and Change in Error is Zero (Z) then Control output is PB.

This is logical as the rule seeks to reduce the output since a positive error signifies an undershoot situation and has been presented by authors [15, 19, 20].

D. Defuzzification

The output required for controlling the plant is a crisp value and it is determined from the complete fuzzy set by a defuzzification scheme. The commonly used defuzzification methods are: [19]

Center of Gravity (COG):

$$u_{COG} = \frac{\sum_i \mu_c(x_i) x_i}{\sum_i \mu_c(x_i)} \quad (15)$$

Bisector of Area (BOA):

$$\left|\sum_{i=1}^{n} \mu_c(x_i) - \sum_{i=n+1}^{i_{max}} \mu_c(x_i)\right|, i < n < i_{max} \quad (16)$$

Where, x_i depicts the point on the universe of discourse ($i = 1, 2, \ldots k$) and $\mu_c(x_i)$ is the degree of membership for the input set.

4.1. Fuzzy-PID Controller

The PID controller tuned offline has shortcomings in dealing with nonlinearities and parameter variation. Hence this paved way for an adaptive online tuning. During the drive operation, the fixed PID controller gains are constantly adjusted by the Fuzzy logic controller to counteract the nonlinear effect and disturbance [9]. The adaptive Fuzzy-PID structure is presented in paper [1] and schematics shown in Figure 2

The fuzzy logic controller's scaling gains have to be tuned precisely as the influence the optimal response of the system. There is, however, no consensus

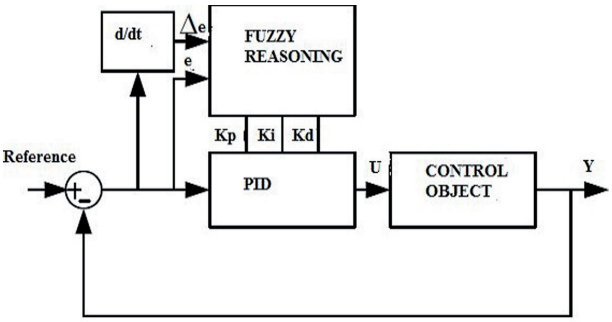

Fig. 2. An Adaptive Fuzzy-PID Controller Structure [1]

for tuning the scaling gains as authors have suggested using trial and error which is tedious [2, 3, 17].

5. Tuning Methods

5.1. Root Locus (Pole Placement)

The altering of the pole and zeros may be achieved graphically or numerically in the MATLAB, Single-input-single-output – SISO – tool. This avails the user opportunity to derive the best response by observing the root locus plot during the adjustment. Design constraints such as the maximum overshoot, settling time can be specified. Though the poles may be placed in the desired location the optimal response is not guaranteed. It is based on mathematics and should be regarded as an intelligent guess. Thus this becomes a tiresome process [21].

5.2. Manual Tuning Method

In paper [22] a summarized rule of manually tuning the controller gains based on experience is proffered as an alternative method. However, the author assets to the fact that it doesn't work all the time and thus takes a long time.

The process involves tuning first making derivative and integral action zero. Afterward, a proportional action is increased gradually and the derivative action is used to dampen overshoot. While the integral term is used to eliminate the steady state error. This process goes on till the optimal response is derived and proportional term large as possible. Table 1 summarizes this method.

Table 1. Manual Tuning Rules [24]

Operation	Rise time	Overshoot	Stability
Kp ↑	Faster	Increases	Decreases
TD ↑	Slower	Decreases	Increases
1/TI ↑	Faster	Increases	Decreases

5.3. Genetic Algorithm

Genetic Algorithm is a stochastic searching optimization method patterned to imitating the theory of natural selection and genetics. In GA individuals constituting a population exist as a potential solution to a given problem. Some advantages of GA are the ease of implementation and that the evaluation (fitness) function knowledge of plant parameters are not required. It also works for non-linear systems as well as avoids local minima convergence. GA comprises three key phases; selection, crossover and mutation [9, 23, 24].

A. Selection Phase

In [24] the selection process is done using the roulette wheel while in [2] stochastic uniform is used to constitute the individuals in the first generation. An iterative process occurs with each iteration representing a generation. This is such that at the end of each generation, individuals are compared to an evaluation function and selected for mating to give rise to new offspring. The off-spring usually would possess the good traits of their progenitor haven evolved with some degree of mutation and crossover genetic process as specified in the algorithm [25]. The population size for consequent generations is governed by the mathematical representation:

$$Number\ of\ offspring = N * relative\ fitness \quad (17)$$

where, N is Population size.

B. Crossover Phase

The mating pairs after the selection phase undergo crossover genetic procedure which is performed by crossover probability in order to give rise to new offspring with enhanced genetic traits. Mathematically, crossover operation is given as: [2]

$$\tilde{P}_{offspring} = \alpha\tilde{P}_{parent1} + (1-\alpha)\tilde{P}_{parent2}$$

$$\tilde{P}_{parent1} \neq \tilde{P}_{parent2} \quad (18)$$

Where, $\tilde{P}_{parent1}$ and $\tilde{P}_{parent2}$ are dissimilar parent chromosomes, α is a stochastically acquired natural number, α∈[0, 1].

A small crossover value is usually selected for a large population size in other to enhance and the individuals while retaining the best characteristics of both parents [23, 24].

C. Mutation Phase

The mutation phase occurs after the crossover process mimicking the real life process of mutation which involves the modification of the genome on rare occasions. Hence the occurrence of mutation is usually low for the preservation of good chromosomes. The objective of introducing mutation is to avoid convergence of local minima and increase the fitness of the individual as a possible solution by chromosome mutation. Author [24] suggests using the following formula to determine mutation rate:

$$P_m = \frac{1}{P_S\sqrt{L}} \quad (19)$$

Where, P_m is the finest mutation rate, P_S is the population size and L is the length of individual random string.

One of the benefits of the GA algorithm is that it can be used to tune and optimize the PID and fuzzy controller gains. The GA despite its benefit it does not guaranty optimal solution for all time. The GA flow chart is shown in Figure 3.

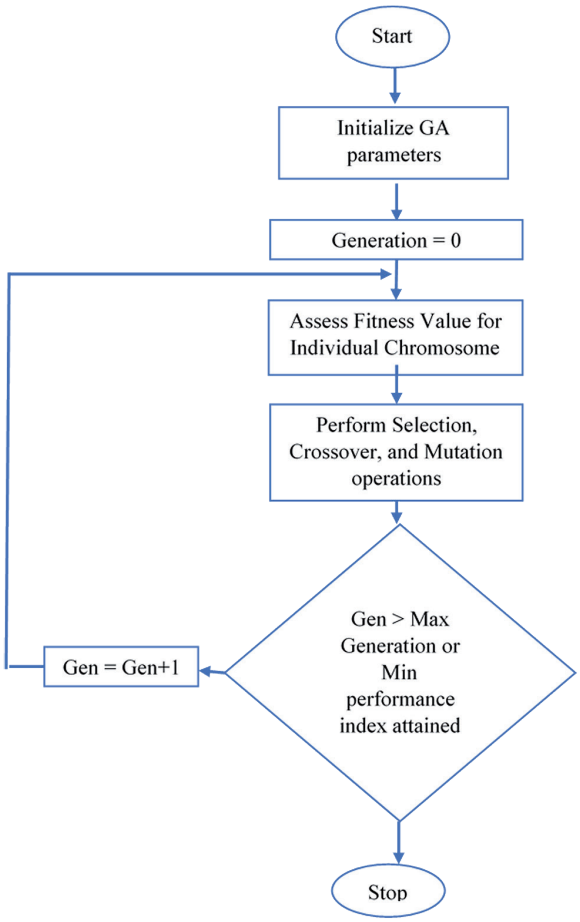

Fig. 3. Flowchart for genetic algorithm

5.4. Particle Swarm Optimization

The particle swarm optimization (PSO) technique is a population based search algorithm where individuals are termed as particles and denote a candidate solution [26].

The principle of PSO is centered on swarms searching for productive feed haven. The PSO concept is such that ab initio a random population of particles is constituted. Each particle is bestowed with a random velocity and flown through the problem space. Based on an individual particles flight experience and that of other particles, it modifies its flight pattern in the problem space. Each particle keeps track of flight trajectory related with the optimal solution (fitness) attained during the flight through the problem space. This value is referred to as the Also the optimal solution of any particle in the search space is called $g_{best.}$ The key idea is such that the particle exhibit change in velocity (acceleration) with time advancing toward the g_{best} and p_{best} position. Each particle tends to alter its present position and velocity with respect to the distance between itself and g_{best} and itself and p_{best} [26, 27]. Figure 4 shows the flowchart of the PSO

The particles position and velocity modification are thus achieved as thus:

$$v_{i,g}^{(k+1)} = K * [v_{i,g}^{(k)} + c_1 r_1 * (P_{best\ i} - p_{i,g}^{(k)}) + c_2 r_2 * (g_{best\ i} - p_{i,g}^{(k)})] \quad (20)$$

$$p_{i,g}^{(k+1)} = p_{i,g}^{(k)} + v_{i,g}^{(k+1)} \quad (21)$$

Where:
- $v_{i,g}^{(k+1)}$ denotes the velocity of particle *i* at dimension *g* at the iteration *k*+1,
- $p_{i,g}^{(k+1)}$ denotes the position of particle *i* at dimension g at the iteration *k*+1,
- r_1 and r_2 randomly generated numbers between 0 and 1,
- c_1 and c_2 are cognitive and social acceleration constants, respectively,
- *K* constriction factor denoted by

$$K = \frac{2}{\left|2-\varphi\sqrt{\varphi^2-4\varphi}\right|} \text{ where } \varphi = c_1+c_2,\ \varphi>4\,. \quad (22)$$

The velocity range is [-V_{max}, V_{max}] I = 1, 2,...,n; g = 1, 2,..., d, where n is the number of particle swarm and the dimension of the optimization problem is "d"

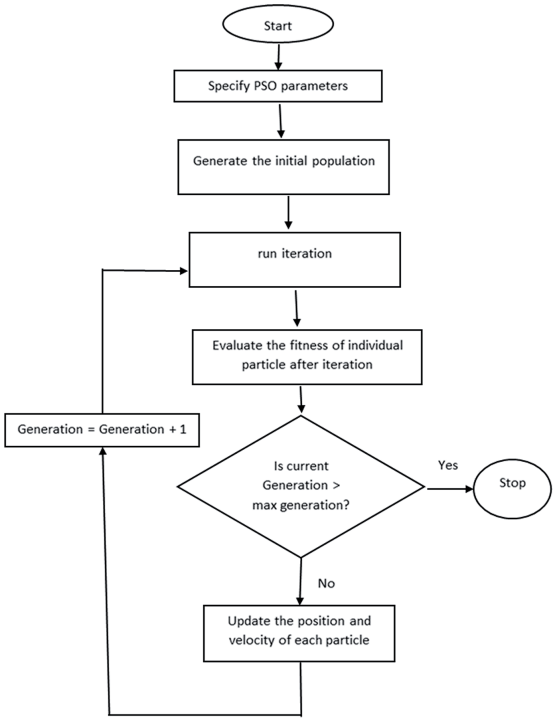

Fig. 4. PSO algorithm flowchart

[12, 27].

6. Fuzzy-PID Sliding Mode Controller (FPIDSMC)

6.1 Sliding Mode

The proposed controller design comprises of a complementary controller structure. This comprises of an online tuned Fuzzy-PID and sliding mode controller for an improved transient and steady state performance.

The influence of an integral term in the PID controller deployed in position control causes offset in the output. This is due to an incrementing action while the absence of it causes poor rejection of load disturbance.

The sliding mode output is switched ON in the event disturbance or noise is detected. This succinctly provides for the integral action required to counteract the abnormality and preserve the response. The injected integral term is summed up with the adaptive fuzzy controller integral term. This approach, therefore, harnesses the robustness of the sliding mode and the advantage of the fuzzy controller with respect to model insensitivity.

The high frequency chattering of the SMC may be solved by a trade-off for robustness. In [28, 29] the problem is resolved by selecting a suitable smoothening signum function with a negligible boundary layer width ∅ in the order of ±10^{-4}. As the output of the switched integral term is smoothened before it is introduced to the system input. For estimating the presence of disturbance and noise in the controlled variable output $\theta(t)$ a first order filter with a time constant τ of 8 mS is used. An advanced signal $\theta(t+1)$ is generated from the controlled output variable $\theta(t)$ which is the present signal. The first order filter is thus expressed as a differential equation [30]:

$$\tau * \frac{d\theta(t)}{dt} + \theta(t) = \theta(t) \quad (23)$$

And in the S domain as a transfer function:

$$\mathrm{E(s)} = 1 / (\tau S + 1) \quad (24)$$

The sliding surface S_s is presented for tracking $\theta(t)$ in the presence of torque load disturbance and noise as:

$$S_s = Ce_o(t) + de_o(t)/dt = 0 \quad (25)$$

where, C is a positive constant for scaling

$$e_o(\mathrm{t}) = \theta\,(t) - \theta\,(t-1)$$
and
$$e_o(t) \neq \theta_{ref}(t) - \theta(t)$$

The system is confined to the sliding surface as the first derivative of the sliding surface converges to zero. Hence, reaching surface is defined as:

$$S_s(dS_s/dt) = S_s\,(C\frac{de_o(t)}{dt} + \frac{d^2e_o(t)}{dt^2}) \quad (26)$$

If

$$S_s\left(\frac{dS_s}{dt}\right) \leq -\eta|S_s/\phi| \quad (27)$$

Where η is a positive real constant.
Then $S_s(dS_s/dt)<0$ will ensure sliding occurs at $S_s = 0$. The DC motor control law for $U_{FPIDSM}(t)$ is expressed mathematically, as the summation of the continuous $U_{cont}(t)$ and discontinuous $U_{disc}(t)$ functions:

$$U_{FPIDSMC}(t) = U_{cont}(t) + U_{disc}(t) \quad (28)$$

$$U_{cont}(t) = -K1 * \theta(t) - K2 * d\theta(t)/dt + K_P e(t) + [\Delta Kp_{min} - \Delta Kp_{max}] + K_I \int_0^T e(t)dt + [\Delta Ki_{min} - \Delta Ki_{max}] + KD * de(t)/dt + [\Delta Kd_{min} - \Delta Kd_{max}] \quad (29)$$

Where,
- K1 and K2 are scaling gains for position and velocity feedback attenuators,

- ΔKp_{min} and ΔKp_{max} defines the finite range of the Fuzzy gain values for incrementing proportional control action.
- ΔKi_{min} and ΔKi_{max} are the Fuzzy output finite range for incrementing integral control action.
- ΔKd_{min} and ΔKd_{max} are defines the finite range of the fuzzy gains for derivative control action.
- K_{P}, K_{I} and K_{D} are the PID controllers proportional, integral and derivative gains.

Furthermore, for an improved damping during the transient state, the DC motor output, and derivative of the DC motor output are scaled and utilized as a damping feedback loop to motor input.

The discontinuous control law $U_{disc}(t)$ functions concurrently with $U_{cont}(t)$. The sliding mode controller is switched ON/OFF with respect to the switching function defined in equation (33).

As with the sliding mode controller the high frequency ON/OFF switching results in the chattering phenomenon. Authors have consistently tried to minimize the chattering exhibited. In [13, 14] a simple reaching law was presented

$$U_{disc} = -Ksgn(S_s) \tag{30}$$

Where K is a real constant.

The output was smoothened by fuzzy logic mimicking a Sat function in [14]. The sliding mode switching sign function is mimicked by a fuzzy logic controller in other to eliminate the high frequency switching. The PID controller is combined in parallel with the fuzzy (PID+FSMC) shown in Figure 5. The output of the PID is used to generate the sliding surface. The 3D

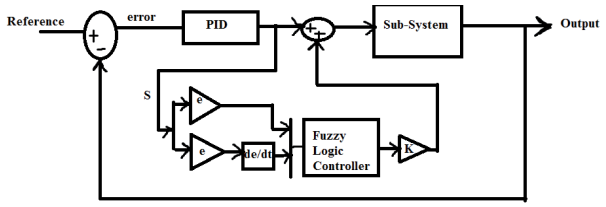

Fig. 5. PID+FZSMC structure

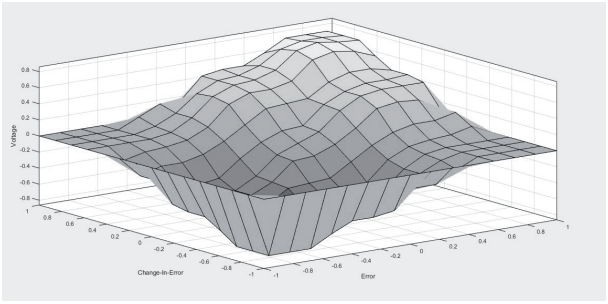

Fig. 6. PID+FSMC 3D surface plot

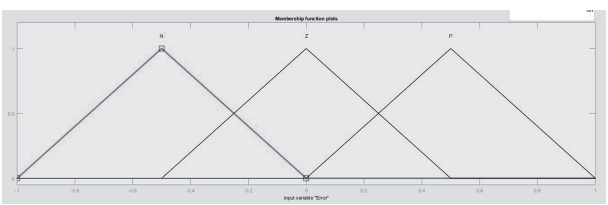

Fig. 7. Input MF for error and change-in-error

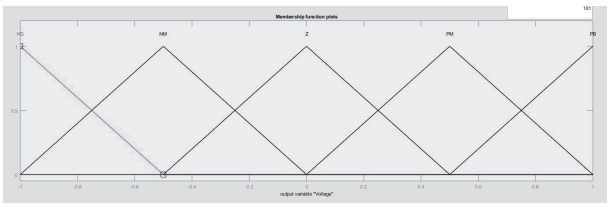

Fig. 8. Output MF for PID+FSMC

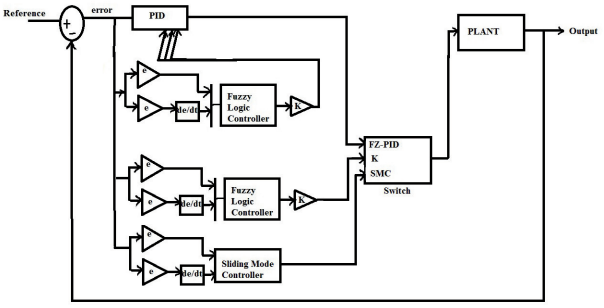

Fig. 9. FDFSMC architecture

surface plot of rule base used is shown in Figure 6 and the MFs are shown in Figures 7 and 8. The decentralized fuzzy architecture is shown in Figure 9.

Also in [31] the following equation was proposed in other to eliminate chattering:

$$U_{disc} = -Ksinh^{-1}(S_s) \tag{31}$$

6.2. Proposed Smoothening for Sliding Mode Controller Output

In this work, equation (32) a hyperbolic function with suitable is proposed for smoothening the signal, thereby eliminating chattering. The effectiveness of the suggested variant functions for smoothening the sliding mode output have been compared in the result section.

The mathematical representation of the reaching law of the sliding mode discontinuous control is represented:

$$U_{disc}(t) = -K\tanh\left(\sinh\left(\frac{S_s}{\emptyset}\right)\right) \tag{32}$$

$$\sinh\left(\frac{S_s}{\emptyset}\right) = \begin{Bmatrix} 1; & \theta(t)-\theta(t-1) < 0 \\ 0; & \theta(t)-\theta(t-1) = 0 \\ -1; & \theta(t-1)-\theta(t) > 0 \end{Bmatrix}$$

$$t = 1, 2, 3 \ldots n \tag{33}$$

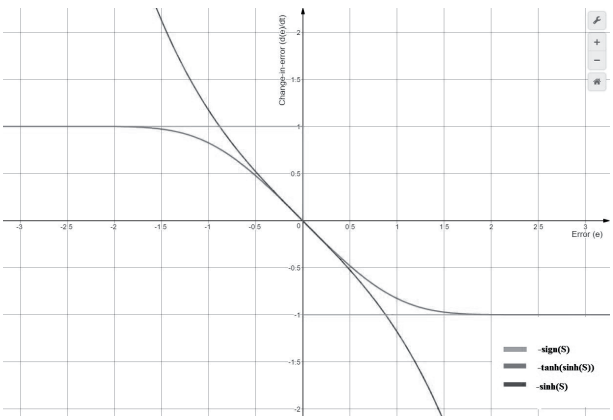

Fig. 10. Sliding mode surface

Where,

- $sinh(S_s/\emptyset)$ is the discontinuous sliding mode switching function with hysteresis boundary for robust control.
- $\theta(t-1)$ and $\theta(t)$, respectively, depict the past and present angular position responses of the DC motor.

The sliding mode control surface for equations (30–32) is shown in Figure 10.

6.3. Adaptive Fuzzy Logic Controller Design

In this work the fuzzy-PID controller's control actions rule base for K_p, K_i and K_D are engineered to negate model uncertainty and parameter variations and GA is used to optimize the controller. The MF comprises of seven Fuzzy sets, 'NB', 'NM', 'NS', 'Z' ,'PS', 'PM', 'PB'. The 'NM', 'NS', 'Z' ,'PS', 'PM' Fuzzy sets are triangular while the 'NB' and 'PB' are both trapezoidal. The MFs are adjusted heuristically to ensure optimal performance. By narrowing the base of the error and change in error, triangular MF to be narrow, a constricted control action is established which improves the steady state performance. The universe of discourse ranges from –3 to 3 for all MFs. Figures 11, 12 and 13 depict the MF set for error and change in error, K_p, K_i and K_D rule base as well as the 3D surface plot respectively. Table 2, shows the 49 heuristically determined fuzzy rules for adjusting K_P, K_I and K_D.

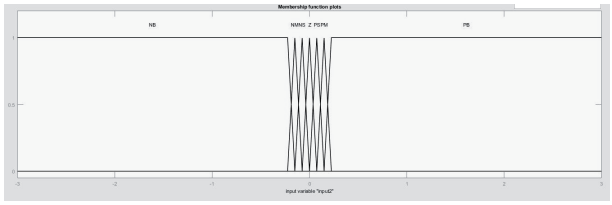

Fig. 11. MF for error and change in error input variables

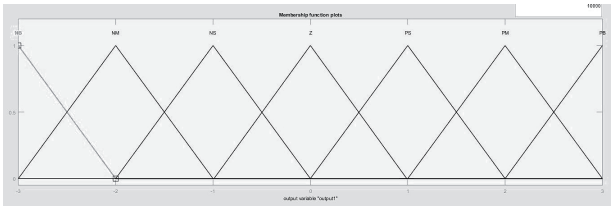

Fig. 12. MF for K_P, K_I and K_D output variables

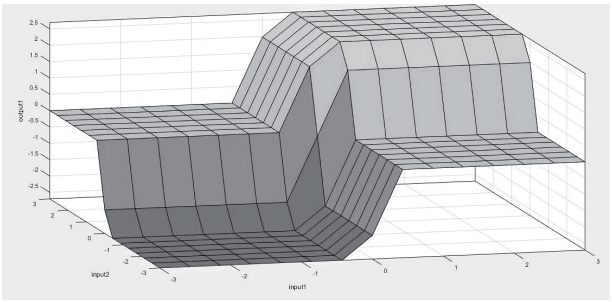

Fig 13. 3D surf mapping of the adaptive rules

The MFs overlap by 50% in order to avoid any misfiring. The defuzzification scheme used is the Bisector of Area (BOA).

The FPIDSM controller schematic diagram is shown in Figure 14.

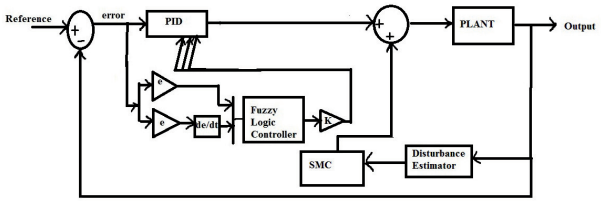

Fig. 14. FPIDSMC

6.4. Proposed Fitness Function for Genetic Algorithm

The fitness function is used to evaluate the distinctive offspring for evolution in GA algorithm. Basically, the fitness function used here originates from one of the following integrals of error function: the integral square error (ISE), integral absolute error (IAE), integral time absolute error (ITAE) [30].

$$ISE = \int_0^T e^2(t)\,dt \tag{34}$$

$$\text{IAE} = \int_0^T |e(t)|\,dt \tag{35}$$

$$ITAE = \int_0^T t|e(t)|\,dt \tag{36}$$

The ISE integrates the square of error over time, penalizing large errors more severely than small errors hence reducing overshoot and settling time [30].

Table 2. Fuzzy Control Rules for ΔK_p, ΔK_i and ΔK_d

e/ce	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NM	NS	Z
NM	NB	NB	NM	NM	NS	Z	PS
NS	NM	NM	NS	NS	Z	PS	PS
ZE	NM	NS	NS	Z	PS	PS	PM
PS	NS	NS	Z	PS	PS	PM	PM
PM	NS	Z	PS	PM	PM	PB	PB
PB	Z	PS	PS	PM	PB	PB	PB

IAE integrates the absolute error over time, long and short errors are treated the alike. The oscillations are reduced with IAE [30].

The ITAE penalizes longer error more severely than an error occurring in a short duration, thus resulting in a reduction of settling time than the ISE and IAE. It is employed mostly for tuning PID gains for robust response [30]. ITAE is used as in paper [2].

In this work, a modified ISE fitness function is coupled with first order lag filter observers used for evaluating the systems error. The modification employs a penalty function which is effective in taking off the overshoot and improving the transient response of the system as well as for minimizing controller energy.

The mathematical representation of the ISE and penalty fitness function used for position control optimization is given as:

$$J = ISE + penalty\ function\ for\ angular\ position\ and\ its\ derivatives \quad (37)$$

The position control fitness function is aimed at attaining steady state response in the shortest period of time without overshoot. This is modeled and shown in Figure 15:

$$\int_0^T (e(t)^2 + w1\left|S_p(t)\right| + w2\left|P_p(t)\right| + w3 * U(t)^2 dt) \quad (38)$$

$$Error\ 'e(t)' = reference\ angular\ position\ 'r' - actual\ angular\ position\ output\ 'y' \quad (39)$$

$$P(t) = \theta(t+1) - \theta(t) \quad (40)$$

$$S(t) = W(t+1) - W(t) \quad (41)$$

Where $W = d\theta/dt$.

Where, *w1, w2, w3* are real positive constant weights and $S_p(t)$ and $P_p(t)$ are overshoot penalty terms. W is angular speed, $\theta(t)$ is the angular position of the motor, and $e(t)$ is the error. $U(t)$ is the motor's input voltage included in the objective function for minimising controller energy. The designer determines the weighting coefficients [9, 17–18].

A first order transfer function with time constant is arbitrarily chosen to be smaller than the plants dominant time constant with suitable value usually in the order of 1 mS is used to obtain $S_p(t)$ and 9 mS for $P_p(t)$. From trial and error suitable weighting factor *w* was found by making the value of the penalty term equal to 1.5*Reference input value.

$$w = 1.5*\ (Reference\ input\ value)/(penalty\ term\ value) \quad (42)$$

If overshoot violation does not occur, the fitness function equation is strictly ISE alone.

The fitness function *J* mechanism of operation for position control is such that when the absolute value of position and the derivative of position is greater than zero, this value is multiplied by a weighting constant to induce the penalty operation. Ideally, at steady state the derivative of position and speed should be zero for position control. Therefore, if the derivative of angular position is driven to a zero value early enough the overshoot does not occur. The weights ensure that the value of the penalty term is significant when an overshoot occurs. However, a very large weighting factor might results in convergence to local minima. A challenge in designing the penalty function is that it might require in-depth analysis of the plant model dynamics and operation and the weighting might be difficult to determine heuristically.

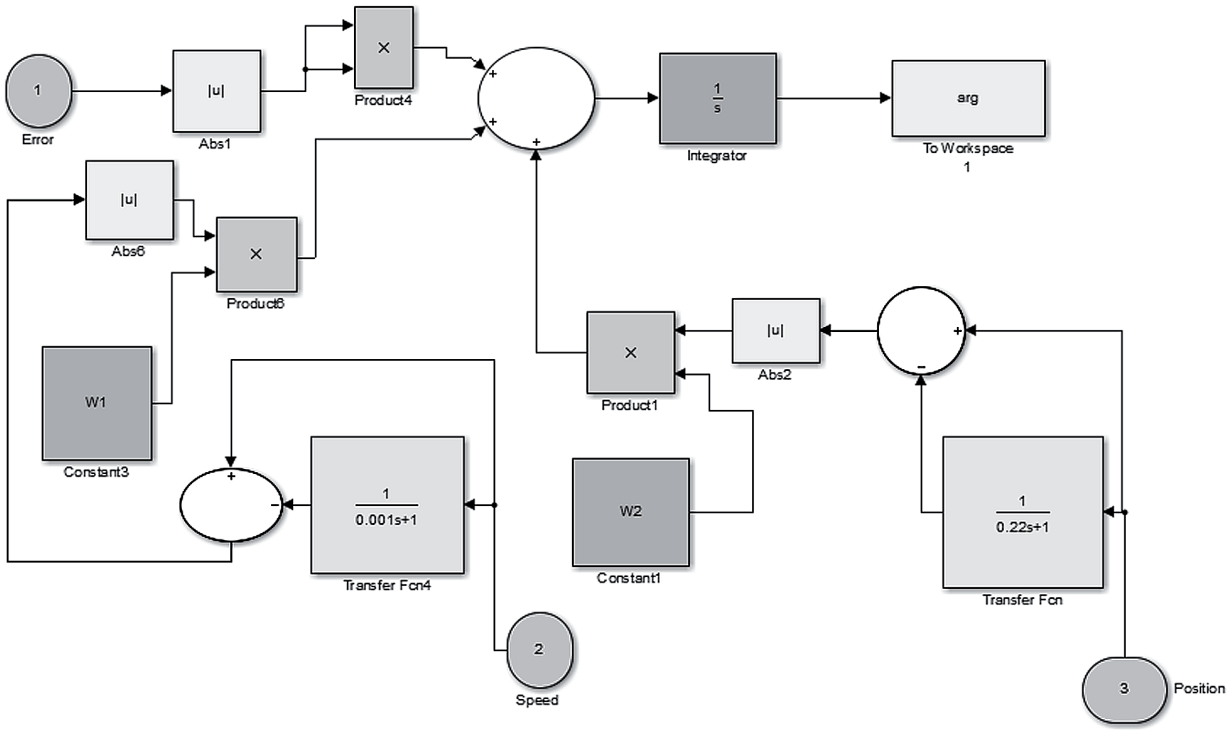

Fig 15. Proposed fitness function for position control incorporating ISE and overshoot penalty function

The practical operational constraints such as the range of set point and torque load disturbance (as well as feedback noise) have been incorporated during the optimization searching process in other to obtain the optimal controller gains, robust for such operations [35].

7. Simulation Results and Discussion

7.1 Simulation

The performance evaluation for the controllers was performed in MATLAB/SIMULINK environment.

Table 3. DC motor parameters

Motor Parameters	Value
Back E.M.F constant 'K'	1.2 *Nm/A*
Moment of inertia for motor rotor 'J'	0.022 $Kg.m^2$
Mechanical damping (friction) factor 'B'	0.0005*Nms*
Resistance of the armature 'R'	2.45 Ω
Inductance of the armature 'L'	0.035 *H*

Table 4. Genetic algorithm parameter selection

Number of generations	Convergence
Population size	20
Selection method	Stochastic uniform
Crossover Probability	0.8
Mutation	0.05

The DC motor parameters are shown in Table 2. The DC motor open loop transfer function for position control is given as:

$$G(s) = \frac{\theta(s)}{V(s)} = \frac{1558}{(S^3+70.02S^2+1872S)} \quad (43)$$

The gains of the controllers are optimised using GA with the exception of PID+FSMC which is optimised using PSO. The optimisation is performed until convergence of the modified fitness function value occurs. The suitable GA parameter constants used are shown in Table 4

The performance analysis is aimed at comparing the optimized controllers. The comparison is with respect to normal operation, motor parameter variations (increased Resistance and Motor Inertia) disturbance rejection, noise injection.

A second order low pass filter with a bandwidth of 77 Hz is used to block off the high frequency chattering introduced during feedback loop noise injection. The transfer function is given as:

$$T(s) = \frac{K_{DC}\, {w_f}^2}{S^2+\frac{w_f}{Q_f}+{w_f}^2} \quad (44)$$

Where, K_{DC} is the DC gain, w_f is frequency of roll-off and Q_f is pole quality factor.

$$T(s) = \frac{10^4}{S^2+70.71s+10^4} \quad (45)$$

Position control processes require a great deal of precision. Hence a decent transient response; no overshoot and fast rise time, zero steady state error and good disturbance rejection are essential.

The performance is evaluated using the integral error criterion; *IAE* and *ITAE*. Also the classical performance evaluation; rise time *(tr),* settling time *(ts),* maximum percentage overshoot *(Mp)* is explored.

Table 5. DC motor transient and steady state performance for normal operation

Controller Structure	Controller tuning method	Percentage Overshoot (%)	Rise time (sec)	Reaction to Torque and 1nd steady state error	2nd Steady state error	Settling time (sec)
FPIDSMC	GA	0	0.9	0.988 rads and 0 after 10s	-0.000081	1.3
FDFSMC	GA	0.44	1.4	0.980 rads and -0.0001 after 10s	-0.000359	1.5
PID+FSMC	PSO	0.06	0.3	0.999 rads and 0 after 10s	0.013111	0.4
PID	Root-Locus and GA	3	0.8	0.986 rads and 0.008 after 10s	0.001891	1.2
FZ-PID	GA	0	1	0.985 rads and 0 after 10s	0.007160	1.3

During the simulation, the controllers are made to track a step input reference command of 1 rad and 2 rads. During the 1 rad reference command tracking at 15 s into steady state at a torque load disturbance of 0.3 Nm is applied. At 20 s the load torque is reduced to 0.27 Nm. At 30 s the reference command is changed to 2 rads with total simulation runtime of 50 s.

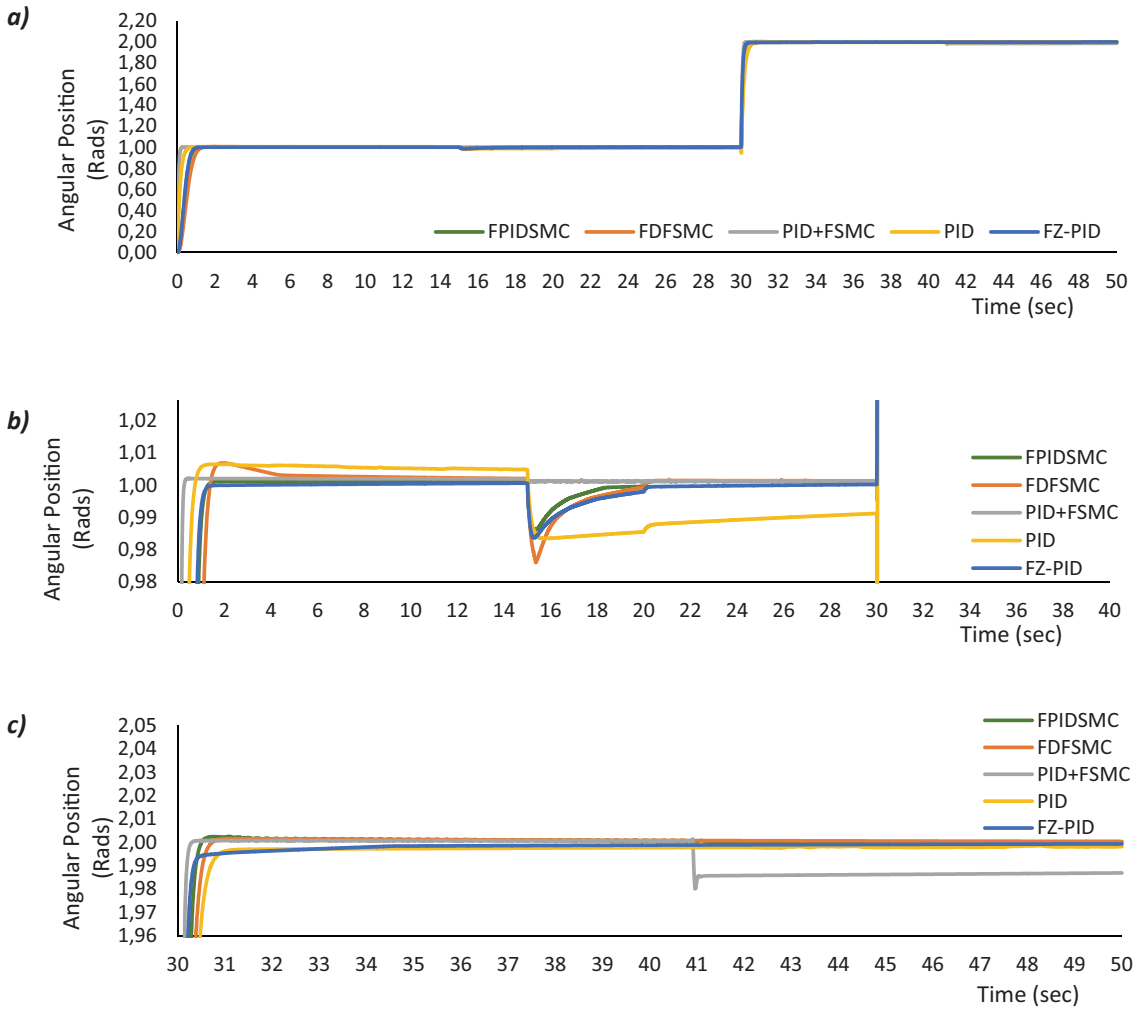

Fig. 16 a, b, c. Motor response for normal operation

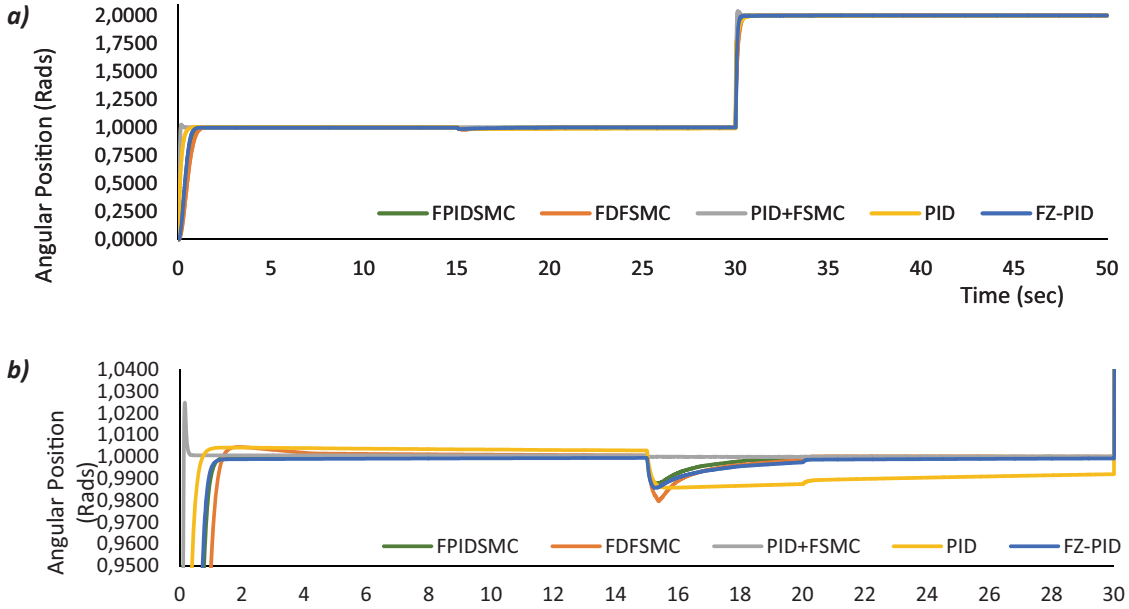

Fig. 17 a, b. Motor response for 50% increase in inertia

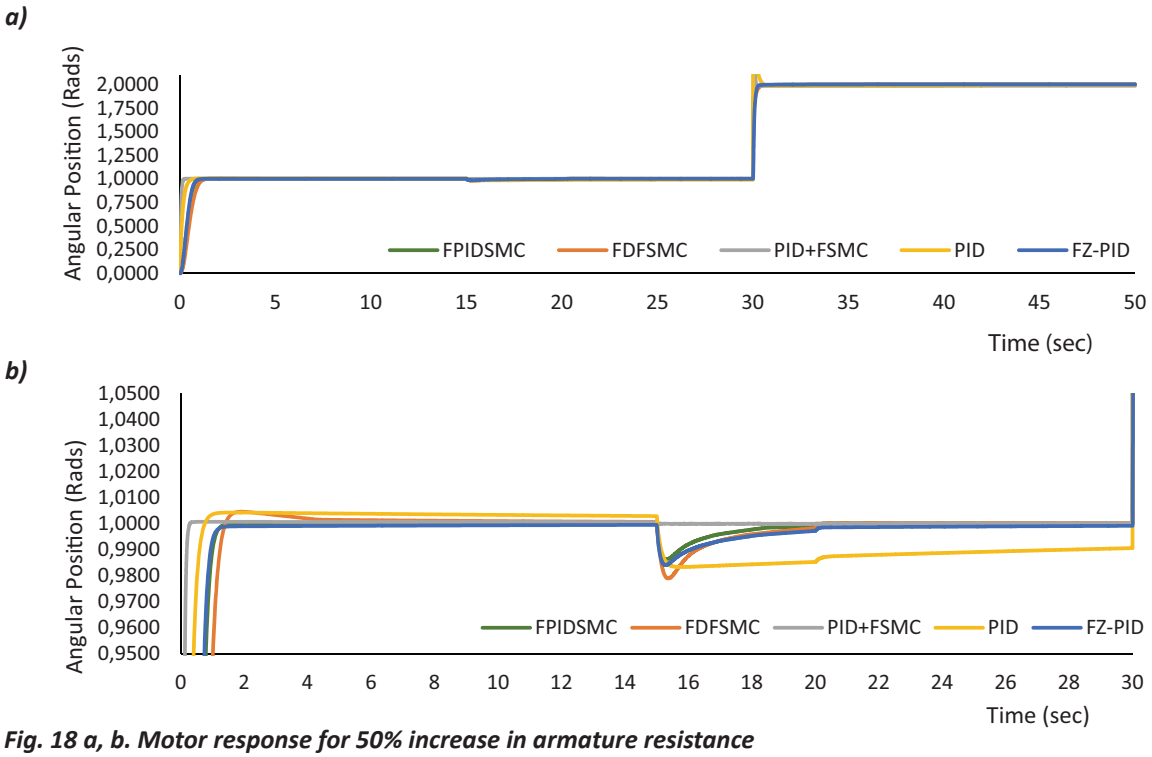

Fig. 18 a, b. Motor response for 50% increase in armature resistance

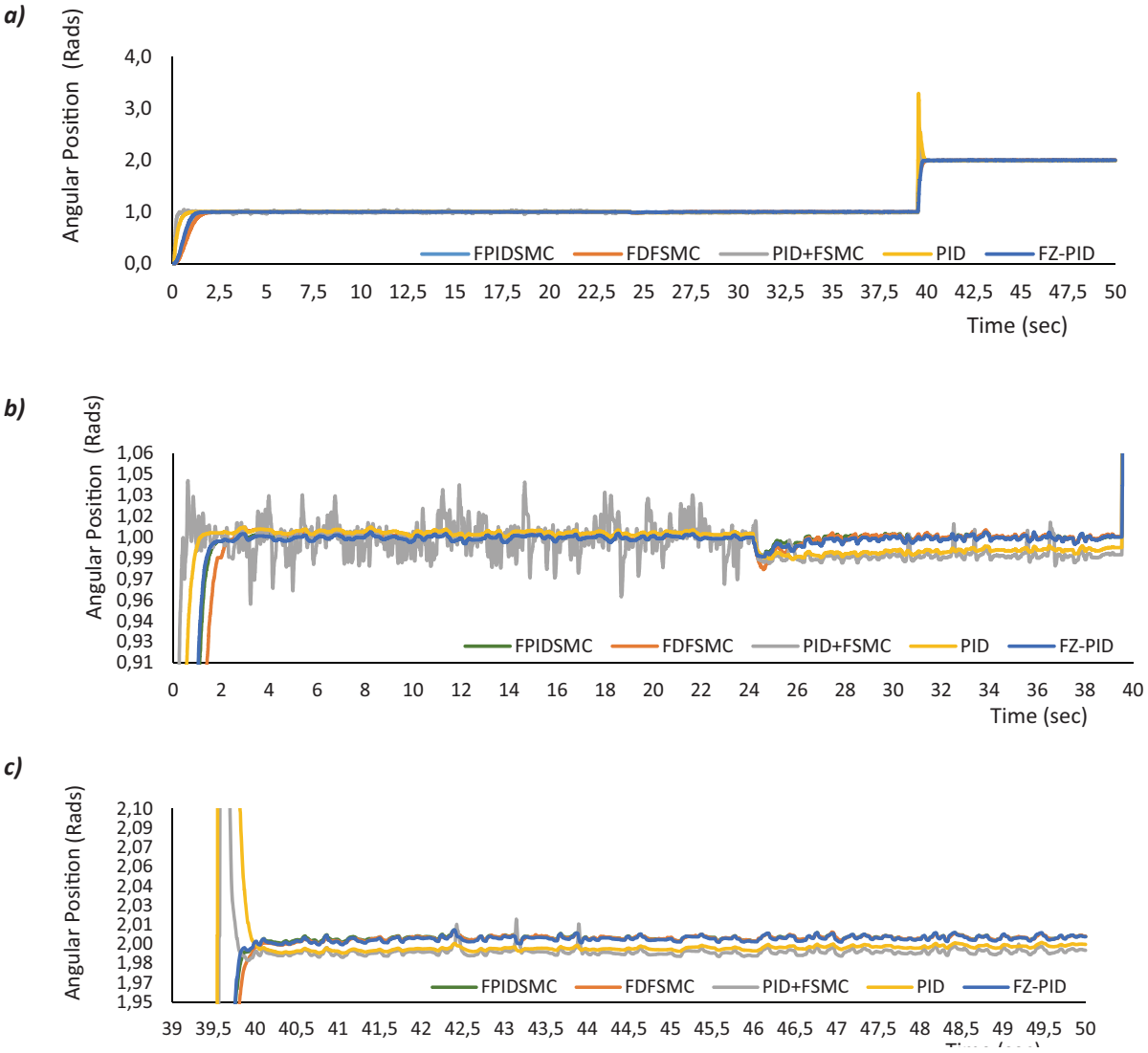

Fig. 19 a, b, c. Motor response during noise injection

Table 6. ITAE ad ISE DC motor controllers performance indices

Operational Condition	Controller Type	ISE	ITAE
Normal operation	FPIDSMC	0.3504	3.959
	FDFSMC	0.4464	5.484
	PID+FSMC	0.0945	8.077
	PID	0.1734	10.59
	FZ-PID	0.3207	4.431
50% increase in Inertia	FPIDSMC	0.3365	2.954
	FDFSMC	0.4266	4.434
	PID+FSMC	0.0782	1.707
	PID	0.1057	9.397
	FZ-PID	0.3269	4.474
50% increase in Resistance	FPIDSMC	0.3850	2.491
	FDFSMC	0.3900	3.551
	PID+FSMC	0.0749	14.03
	PID	0.1406	15.12
	FZ-PID	0.3209	4.447
Noise Injection	FPIDSMC	0.3072	3.481
	FDFSMC	0.3897	4.441
	PID+FSMC	0.07996	16.45
	PID	0.1539	14.22
	FZ-PID	0.3179	4.613

7.2. Discussion

The performance evaluation for all five controllers; PID, FZ-PI, FPIDSMC, FZ-PID and FDFSMC during normal and abnormal operation are presented in Table 5 and 6, respectively. As shown in Figure 16 a,b,c during normal motor operation, the PID+FSMC exhibits the fastest transient response with 0.3 s rise time and 0.4 s settling time while having 0.06% overshoot. This controller has most robust torque rejection, however, the output exhibits chattering despite the deployment of Fuzzy logic to mimic the sliding mode Sat function. Despite having the least ISE value of 0.0945 as ISE only penalizes sluggish transient response. It, therefore, has the poorest steady state error response depicted by a value of 10.59 over the 50 s simulation run time. The PID controller follows closely with an overshoot of 3% and a rise time of 0.8 s but settles only after 0.4 s. Upon torque load injection and decrease, it dips to 0.986 and recovers leaving with the largest state error. However, this reduces to 0.0018 after applying a command reference of 2 rads. The FPIDSMC has a fairly decent rise time of 0.8 s which is closely followed by the FZ-PID and FDFSMC with 1 s and 1.4 s, respectively. The FPIDSMC has a better torque rejection capability that the PID, FZ-PID and FDFSMC. It also has the best steady state response as its ITAE value of 3.959 is the least with zero steady state error during the 1 rad tracking. However, a negligible steady offset of -0.000081 is depicted in the response. The FZ-PID has the next largest steady state error after the PID+FSMC. The FDFSMC exhibits the next best steady state error elimination after the FPIDSMC despite it has the poorest torque load rejection capability.

Figure 17a,b shows the controllers response during 50% parameter variation increase in motor inertia. The PID+FSMC and PID controllers exhibit increased overshoot while the response of the FPIDSMC, FDFSMC, and FZ-PID controllers remain consistent. The FPIDSMC controller maintains the least steady state error capability. From Figure 18 a, b, the response during the effect of 50% increase in armature winding resistance the controllers remain adaptive. This is with the exception of the PID controller which exhibits overshoot upon the second transient.

During the noise injection, the second order filter is applied to block off the high frequency noise signal. As shown in Figure 19 a,b,c the PID+FSMC exhibits an increased high frequency chattering in the output. This is characterized by overshoot during the second transient also. The FPIDSMC, FDFSMC, FZ-PID and PID all exhibit a reduced high frequency chattering compared to the PID+FSMC. The high frequency chattering is the resultant of the derivative block responding to the high frequency rate change in the noisy signal. The PID controller is characterized however by a large overshoot of 40% during the second transient.

Haven compared all five controllers the FPIDSMC has the least ITAE value with respect to the operating conditions considered. Hence it is the most precise for position control. However, with respect to load torque rejection and fast transient the PID+FSMC has

the least ISE value. Hence it is the fastest amongst the controllers despite its poor steady state error elimination.

8. Conclusion

The work reviewed current trends in fuzzy sliding mode applied in position control of a DC motor. The performance of the controllers reviewed was evaluated during normal and abnormal circumstances. The sliding mode controller is known to be robust and highly effective in negating disturbance. However, the output exhibits chattering. The PID+FSMC despite mimicking the sat function using fuzzy logic didn't eliminate the chattering. However, it had the best transient response. A smoothening sliding mode law is proposed in the work. The capability of the fuzzy logic artificial intelligence controller is used enhance the poor capability of the PID by adapting controller's gains in the event of model imprecision. The adaptive Fuzzy-PID controller is merged with a sliding mode controller to form a congruent robust structure. The FPIDSMC, therefore, harnesses the capabilities of both controllers. Despite not being the fastest controller it is the most precise controller amongst those compared as it has the least ITAE values for all time. The decentralized FDFSMC comprising a Fuzzy sliding mode and FZ-PID, as well as SMC derived from existing work was also implemented. This controller, however, had a good steady state performance only second to the FPIDFSMC. Though a very poor rejection capability. The conventional PID gains were initially tuned by root locus method afterward GA was used to optimize the gains. Also, the FPIDSMC, FDFSMC, FZ-PID scaling gains were tuned by GA for optimal response. Since the PID+FSMC exhibits steady state error PSO was used to optimize the response as it converges faster than GA. Overall a compromise in robustness, however, guarantees pristine steady state response.

AUTHORS

Bassey Etim Nyong-Bassey * - Department of Electrical/Electronics Engineering, Federal University of Petroleum Resources Effurun, P.M.B 1221 Effurun, Delta State, Nigeria.
E-mail: Nyongbassey.bassey@fupre.edu.ng

Benjamin Akinloye – Department of Electrical/Electronics Engineering, Federal University of Petroleum Resources Effurun, P.M.B 1221 Effurun, Delta State, Nigeria.
E-mail: Akinloye.benjmin@fupre.edu.ng

*Corresponding author

REFERENCES

[1] Yunfei L., Hui L., Yong C., "Research on tuning method for fuzzy PID". In: *4th International Workshop on Advanced Computational Intelligence* (IWACI), 2011, 334–337.

[2] Gadoue S. M., Giaouris D., Finch J. W., "Artificial intelligence-based speed control of DTC induction motor drives – A comparative study", *Electric Power Systems Research*, vol. 79, Jan. 2009, 210–219. DOI : 10.1016/j.epsr.2008.05.024.

[3] Fereidouni A., Masoum M. A. S., Moghbel M., "A new adaptive configuration of PID type fuzzy logic controller", *ISA transactions*, vol. 56, May 2014, 222–240. DOI: 10.1016/j.isatra.2014.11.010.

[4] Meshram P. M., Kanojiya R.G., "Tuning of PID Controller using Ziegler-Nichols Method for Speed Control of DC Motor". In: *IEEE International Conference On Advances In Engineering, Science And Managem*ent (ICAESM), 2012.

[5] Mohiuddin M.S., "Comparative study of PID and Fuzzy tuned PID control for speed control of DC Motor", *International Journal of innovations in Engineering and Technology*, vol. 2, August 2013, 291–301.

[6] Pandey S., Pandey B., "DC Motor Torque Control using Fuzzy Proportional-Derivative Controllers", *International Journal of Engineering and Advanced Technology* (IJEAT), vol. 3, no. 6, August 2014, 202–207.

[7] Ingole C. R., Kasat K. N., "A Review on Different Approaches for Speed Control of DC Motor". Available online: www.ijetemr.in/website/CMS/admin/upload/5-6-1-SM.doc.

[8] Kumar P., Prabhat P. K., Kumar M., Choudhary S. D., "Speed Control of DC Motor Using PID & Smart Controller", *International Journal of Scientific & Engineering Research*, vol. 5, no. 11, November 2014, 1044–53.

[9] Hamed B. M., Al-Mobaied M. N., "Fuzzy Logic Speed Controllers Using FPGA Technique for Three-Phase Induction Motor Drives", *Dirasat, Engineering Sciences*, vol. 37, no. 2, 2010, 194–205.

[10] Deia Y., Kidouche M., Ahriche A., "Fully decentralized fuzzy sliding mode control with chattering elimination for a Quadrotor attitude". In: *2015 4th International Conference on Electrical Engineering (ICEE)* 2015, December 2015, 1–6. DOI: 10.1109/INTEE.2015.7416742.

[11] Shallal H., "Comparison Performance of Different PID Controllers for Dc Motor", vol. 5, June 2012, 121–230.

[12] Medewar P.G., Munje R.K., "PSO with modified objective function for performance enhancement of PMDC motor". In: *2015 International Conference on Energy Systems and Applications*, October 2015, 658–662. DOI: 0.1109/ICESA.2015.7503432

[13] Ahmed H., Rajoriya A., "A Hybrid of Sliding Mode Control and Fuzzy Gain Scheduling PID Control using Fuzzy Supervisory Switched System for DC Motor Speed Control System", *International Journal of Grid and Distributed Computing*, vol. 9, no. 5, 2016, 41–54. DOI: 10.14257/ijgdc.2016.9.5.05.

[14] El-Bakly A., Fouda A., Sabry W., "A proposed DC motor sliding mode position controller design

using fuzzy logic and PID techniques". In: *Proceedings of the 13th International Conference on Aerospace Sciences and Aviation Technology (ASAT'09)*. May 2009.

[15] *http://i.stack.imgur.com/ei9cI.gif*. Access: 9th of May, 2016.

[16] Sandhya D., Amarendra Reddy B., Gopala Rao K.A., "Fuzzy I-PD and Fuzzy PID Control of Nonlinear Systems". In: *International Conference on Control, Automation, Communication and Energy Conservation*, INCACEC 2009, 4th–6th June 2009.

[17] Yusof A.M., *A Comparative study of Conventional PI and Fuzzy-PID for motor speed,* M.Sc. Thesis, Faculty of Electrical and Electronic Engineering, University Tun Hussein Onn Malaysia, June 2013.

[18] Li H.-X., Gatland H. B., "Conventional fuzzy control and its enhancement", *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, 1996, 791–797.

[19] Jantzen J., *Tutorial on fuzzy logic*, Technical Report, Technical University of Denmark, Dept. of Automation, 1998.

[20] Zhao J., Bose B. K., "Evaluation of membership functions for fuzzy logic controlled induction motor drive". In: *IECON 2002. 28th Annual Conference of the Industrial Electronics Society*, 2002, 229–234.

[21] Tan C. P., Teoh K. S., Jones L. J. N., "A review of Matlab's SISOTOOL; features and contributions to Control education". In: *International Federation of Automatic Control World Congress*, Hyung Suck Cho, 6th–11th July 2008, 2008, 8473–8474

[22] Namazov M., Basturk O., "DC motor position control using fuzzy proportional-derivative controllers with different defuzzification methods", *Turkish Journal of Fuzzy Systems*, vol. 1, 2010, 36–54.

[23] Wu C.T., Tien J.-P., Li T. S., "Integration of DNA and real coded GA for the design of PID-like fuzzy controllers". In: *2012 IEEE International Conference on Systems, Man, and Cybernetics*, 2012, 2809–2814. DOI: 10.1109/ICSMC.2012.6378174.

[24] Fan L., Joo E. M., "Design for auto-tuning PID controller based on genetic algorithms". In: *4th IEEE Conference on Industrial Electronics and Applications*. ICIEA 2009, 1924–1928.

[25] Yeniay O., "Penalty function methods for constrained optimization with genetic algorithms", *Mathematical and Computational Applications*, vol. 10, 2005, 45–56.

[26] Mohanty B., Hota P.K., "Particle swarm optimization based interconnected Hydro-Thermal AGC system considering GRC and TCPS", *International journal of Electrical, computer, Electronics and Communication Engineering*, vol. 8, 2014, 1175–1181.

[27] Omar A., Khedr T. Y., Zalam A., "Particle swarm optimization of fuzzy supervisory controller for nonlinear position control system". In: 2013 8th International Conference on Computer Engineering & Systems (ICCES), 2013, 138–145.

[28] Martí P., Velasco M., Camacho A., Martín E. X., Fuertes J. M., "Networked sliding mode control of the double integrator system using the event-driven self-triggered approach". In: *2011 IEEE International Symposium on Industrial Electronics (ISIE),* June 2011, 2031–2036. DOI: 10.1109/ISIE.2011.5984472.

[29] El-Bakly A., Fouda A., Sabry W., "A proposed DC motor sliding mode position controller design using fuzzy logic and PID techniques". In: *13th International Conference on Aerospace Sciences & Aviation Technology*, May 2009.

[30] Shuaib A. O., Ahmed M. M., "Robust PID Control System Design Using ITAE Performance Index (DC Motor Model)", *International Journal of Innovative Research in Science, Engineering and Technology* (IJIRSET), Vol. 3, August 2014

[31] Asad M., Bhatti A.I., Iqbal S., Asfia Y., "A smooth integral sliding mode controller and disturbance estimator design", *International Journal of Control, Automation and Systems*, vol. 13, no. 6, 2015, 1326–1336. DOI: 10.1007/s12555-013-0544-4.

[32] Shuaib A. O., Ahmed M. M., "Robust PID Control System Design Using ITAE Performance Index (DC Motor Model)", *International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)*, vol. 3, August 2014.