International Journal of Electrical Engineering.
ISSN 0974-2158 Volume 5, Number 7 (2012), pp. 863-880

http://www.irphouse.com

Hybrid Intelligent Controllers Based Speed Control of PMBLDC Motor Using Soft-Switching Inverter

N. Muruganantham[1] and Dr. S.Palani[2]

[1]Asst. Professor of Electrical & Electronics Engineerin
, Periyar Maniammai University, Thanjavur (d.t), Tamil Nadu, India.
E-mail: murugaksr@rediffmail.com
[2]Dean of Electronics Engineering,
Sudharsan Engineering College, Sathiyamangalam,
Pudukkottai (d.t), Tamil Nadu, India.
E-mail: keeranur_palani@yahoo.co.in

Abstract

Permanent Magnet Brushless DC (PMBLDC) motor drives are increasingly popular in industrial applications due to rapid progress of technologies in power electronics and the growing demand for energy saving. The increasing demand of energy saving from society is the external force for the development of PMBLDC motor drives. It is however driven by a hard-switching pulse width modulation (PWM) inverter, which has low switching frequency, high switching loss, high electro-magnetic interference (EMI), high acoustic noise and low efficiency, etc. To solve these problems of the hard-switching inverter, many soft-switching inverters have been designed in the past. Unfortunately, high device voltage stress, large dc link voltage ripples, complex control scheme and so on are noticed in the existing soft-switching inverters. This paper presents the comparative analysis between conventional PI, fuzzy, hybrid fuzzy-PI, GA-PI and adaptive neuro-fuzzy inference system (ANFIS) controller based soft switching inverter using transformer, which can generate dc link voltage notches during chopping which minimize the drawbacks of existing soft-switching. Hence all switches work in zero-voltage switching condition. The performance of the hybrid intelligent controllers is compared with conventional PI controller. The simulation results show that the ANFIS controller renders a better transient response than the one obtained using conventional PI controller with negligible overshoot, smaller settling time and rise time. Further the ANFIS controller provides low torque ripples and high starting torque. The simulation results are presented to show the superiority of the proposed hybrid intelligent controllers based soft switching inverter.

Keywords: Brushless DC Motor, PI Controller, Fuzzy Logic Controller, Hybrid Fuzzy-PI Controller, GA-PI Controller, ANFIS Controller, Zero-Voltage Switching, Zero-Current Switching.

1. Introduction

Brushless DC motor (BLDCM) has been broadly used in drive systems and servo control because of its fast response, high power density, high efficiency, low inertia, high reliability and maintenance free. The operating characteristics of brushless dc (BLDC) motor resemble that of a conventional commutated dc permanent magnet motor but without the mechanical commutators and brushes. Hence many problems associated with brushes such as radio-frequency interference and sparking which is the potential source of ignition inflammable atmosphere are eliminated. It is usually supplied by a hard switching pulse width modulation (PWM) inverter, which normally has relatively low efficiency since the power losses across the switching devices are high. The high dv/dt and di/dt will result in severe electromagnetic interference (EMI) and rigorous problems with the reverse recovery of the freewheeling diodes, especially in high switching frequency. As the switching frequency of the hard switching is not very high when the switching frequency is within audio spectrum, it may produce severe acoustic noise. In order to solve these problems, many soft switching inverters have been designed in the past but they have their own limitations [1].

Proportional plus integral (PI) controller when connected to BLDC motor has superior performance as compared to the fuzzy controller under steady state conditions. However, design of PI controller requires mathematical model of the motor. Further, due to parameter variation of the motor when the motor is disturbed, the PI controller requires fine tuning of proportional gain (K_p) and integral time constant (T_i). Which is difficult to accomplish on-line especially under fast changing operating conditions. These problems are overcome by designing the fuzzy logic controllers [2], which do not require any mathematical model of the motor and are based on linguistic rules obtained from the experience of the system operator. But the performance of the fuzzy controller as compared to the PI controller is superior only under transient conditions. A simple gain scheduled PI speed controller has been proposed in [3], where the controller gains are varied with the input error signal. This controller suffers from the drawback of its proper performance, the limits of the controller gains and the rate at which they would change have to be appropriately chosen.

Fuzzy based gain scheduling of PI controller has been proposed in [4], but the limits of the gains have to be determined manually. The advantages of the fuzzy and PI controllers can be combined with a hybrid fuzzy-PI controller which can be implemented as a speed controller where the PI controller is active near and at steady state conditions and the fuzzy controller is active during transient conditions. Hybrid fuzzy-PI speed controller has been in use for the control of the induction motor, where the fuzzy controller is active during speed overshoot or undershoot only [5]. In a permanent magnet brushless dc (PMBLDC) motor with hybrid fuzzy-PI speed controller, the fuzzy logic controller is only activated under the condition of overshoot and oscillations, else the output of the fuzzy logic controller is null and hence inactive [6]. However, the major drawback of fuzzy control is the lack of design technique [7], [8]. Most of the fuzzy rules are human knowledge oriented and hence rules will deviate from person to person inspite of the same performance of the system. The selection of suitable fuzzy rules, membership functions and their definitions along the universe of discourse always involve trial-and-error process [9].

To assure an independent good performance, the controller must be able to adapt the changes of plant dynamic characteristics. For these reasons, it is highly desirable to increase the capabilities of PI controllers by adding new features. Many random search methods, such as genetic algorithm (GA) have recently received much interest for achieving high efficiency and searching global optimal solution in problem space [10], [11] such as the search of optimal PI controller parameters. But optimal PI controller parameters vary based on population size, generation number, selection method, crossover and mutation probabilities. Also, no solid theoretical basis is available and parameter turning is largely based on trial and error and there is no guarantee for finding optimal solutions within a finite amount of time.

Fuzzy based controllers develop a control signal which yields on the selection of the rule base, which is written on the previous experiences and these rules are selected which is random in nature. As a result, the outcome of the controller is also random and optimal results may not be obtained. Selection of the proper rule base depending upon the situation can be achieved by the use of an ANFIS controller, which becomes an integrated method of approach for the control purposes and yields excellent results [12]. In the designed ANFIS scheme, neural network techniques are used to select a proper rule base, which is achieved using the back propagation algorithm.

In this paper, the performance of the hybrid intelligent controllers is compared with conventional PI controller. The simulation results show that the ANFIS controller based soft switching inverter is designed for BLDC motor drive systems which is easy to implement in industries and it has the advantages of low switching power loss, low inductor power loss, low dc link voltage ripple, small device voltage stress, low switching noise and simple control scheme. Moreover the system provides low torque ripples, high starting torque, better transient response with negligible overshoot, smaller settling time and rise time.

2. Soft Switching Inverter Topology

The schematic diagram of the proposed adaptive neuro-fuzzy inference system (ANFIS) controller based soft switching inverter for BLDC motor drive system is shown in Fig. 1. Each pole comprises a resonant inductor and a resonant capacitor at each phase leg. These capacitors are directly connected in parallel to the main inverter switches in order to achieve zero–voltage switching (ZVS) condition. In contrast to the resonant dc link inverter, the dc link voltage remains unaffected during the resonant transitions. The resonant transitions occur separately at each resonant pole when the corresponding main inverter switch needs switching. Therefore the main switches in the inverter phase legs can switch independently from each other and choose the commutation period without restraint.

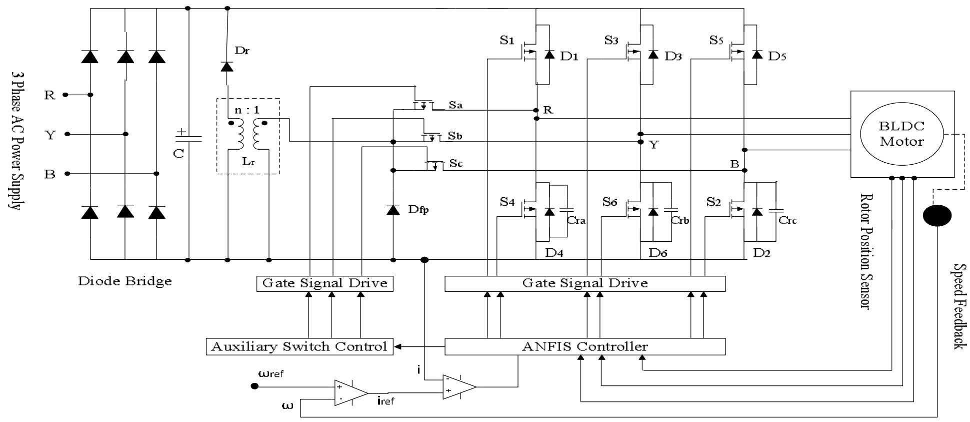


Figure 1: Schematic diagram of the proposed ANFIS controller based soft switching inverter for BLDC motor

Moreover, there is no additional main conduction path switch needed. Thus, the normal operation of the soft switching inverter is entirely the same as that of conventional hard switching inverter. The rotor position is sensed by a Hall effect sensor or a slotted optical disk, which provides three square waves with phase shift of 120º. These signals are decoded by a combinatorial logic [13] to provide the firing signals for 120º conduction on each of the three phases. The three upper switches work under commutation frequency (several hundred Hz) and the three lower switches work under PWM frequency (tens of kHz). So it is not important that the three upper switches work under soft switching condition.

The system contains a diode bridge rectifier, a resonant circuit, a conventional 3Φ inverter and control circuitry. The resonant circuit consists of three auxiliary switches (S_a, S_b, S_c), one transformer with turns ratio 1 : n, and two diodes D_{fp}, D_r. Diode D_{fp} connected in parallel to the primary winding of the transformer and diode D_r which is serially connected with secondary winding across the dc link. There is one snubber capacitor connected in parallel to each switch of phase leg. The snubber capacitor resonates with the primary winding of the transformer. The emitters of the three auxiliary switches are connected together. Thus, the gate drive of these auxiliary switches can use one common output dc power supply. The turns ratio (1 : n) of the transformer, equivalent inductance of the transformer, snubber capacitance and whole switching transition time is determined from [14]. Main switches S_1 to S_6 work under ZVS condition and therefore the voltage stress is equal to the dc link voltage V_s. The device current rate can be load current.

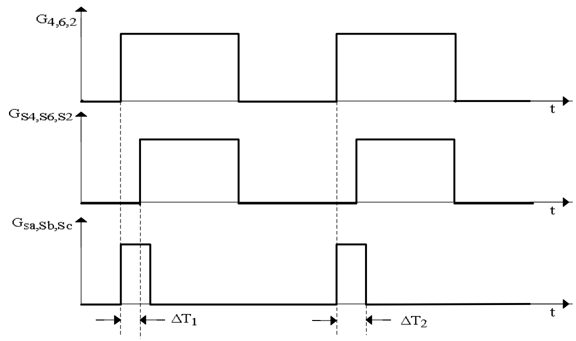


Figure 2: Gate signals $G_{S4,6,2}$ and $G_{Sa,b,c}$ from $G_{4,6,2}$

Auxiliary switches S_a, S_b and S_c work under the ZVS (or) ZCS condition, while the voltage stress is also equal to the dc link voltage V_S. The peak current flowing through them is limited to double the maximum load current. As the auxiliary switches S_a, S_b and S_c carry the peak current only during switching transition, they can be rated with a lower continuous current rating. The additional cost will not be much. The gate signals of three lower main inverter switches and auxiliary switches can be deduced from the output $G_{4,6,2}$ as shown in Fig. 2. The trailing edge of the gate signals for three lower main switches $G_{S4,6,2}$ is same as that of $G_{4,6,2}$, the leading edge of $G_{S4,6,2}$ lags behind $G_{4,6,2}$ for a short time ΔT_1. The gate signals for auxiliary switches $G_{Sa,b,c}$ have a fixed pulse width (ΔT_2) with the leading edge, the same as that of $G_{4,6,2}$.

3. Controller Design

The transient response or dynamic behavior of any system depends on the controller being employed. In general, a conventional PI controller is used for most drive applications. But the conventional PI controller is slow in response and its tuning is a very serious problem if the system is complex to model mathematically. This leads to the search for other suitable hybrid intelligent controllers.

3.1. Proportional Integral Controller

The model of PI speed controller is given by,

$$G(s) = K_p + \frac{K_i}{s} \tag{1}$$

where G(s) is the controller transfer function which is torque to error ratio in s-domin, K_p is the proportional gain and K_i is the integral gain. The tuning of these parameters is done using Ziegler Nichols method [15]. The specifications of the drive application are usually available in terms of percentage overshoot and settling time. The PI parameters are chosen so as to place the poles at appropriate locations to get the desired response. These parameters are obtained using Ziegler Nichols method which ensures stability. From the dynamic response obtained by simulation, the percentage overshoot (M_p), settling time (t_s) and rise time (t_r) which are the measures of transient behaviour are obtained. The closed loop transfer function with PI controller is given by

$$T(s) = \frac{(K_p s + K_i)/J}{s^2 + \left(\frac{B + K_p}{J}\right)s + \frac{K_i}{J}} \tag{2}$$

where T(s) is the closed loop transfer function and K_p, K_i are the PI controller parameters, J is the moment of inertia and B is the coefficient of friction of the BLDC motor. Comparing the characteristic equation of (2) with that of a standard 2nd order system characteristic equation we get

$$K_p = 2\xi\omega_n J - B \tag{3}$$

$$K_i = J\omega_n^2 \tag{4}$$

Fuzzy Logic Controller

Fuzzy logic controller is a rule-based controller. Fuzzy logic enables the designer to describe the general behavior of the system in a linguistic manner by forming IF-THEN rules which are in the form of statements. The general fuzzy logic controller [16]-[18] consists of four parts as shown in Fig. 3. They are fuzzification, fuzzy rule-base, fuzzy inference engine and defuzzification. The design steps are as follows, i) Define inputs, output and universe of discourse ii) Define fuzzy membership functions and rules.

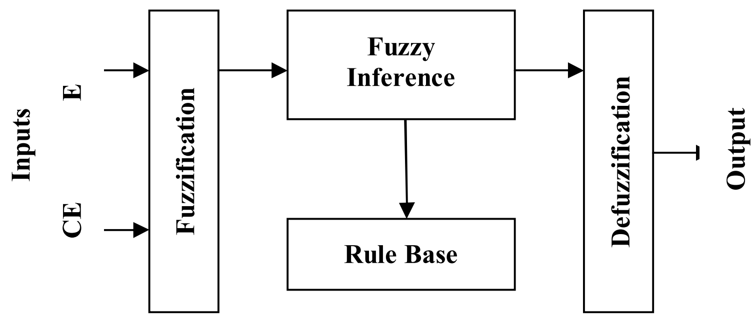


Figure 3: Structure of fuzzy logic controller

In order to define fuzzy membership function, the designer can choose any different shapes based on their preference and experience. The popular shapes are triangular and trapezoidal because these shapes are easy to represent designer's ideas and requires less computation time. In order to fine tune the controller for improving the performance, the adjacent fuzzy subsets are overlapped by about 50%. The performance of the controller can be improved by adjusting the membership function and rules. Fuzzy associative memories (FAM) are transformations which map fuzzy sets to fuzzy sets. A FAM matrix maps antecedents to consequents and it is a collection of IF-THEN rules. Each composition involves seven fuzzy variables and each fuzzy variable is further quantized into seven. This has resulted in fourty nine possible two input and single output FAM rules as illustrated in Table 1.

Finally the fuzzy output is converted into real value output i.e. crisp output by the process called defuzzification. Even though many defuzzification methods are available, the most preferred one is centroid method because it can be easily implemented in digital control systems using microcontrollers and requires less computation time. The formula for this method is given by,

$$Z = \frac{\sum_{x=1}^{n} \mu(x)x}{\sum_{x=1}^{n} \mu(x)} \tag{5}$$

where z is the defuzzified value, and μ(x) is the membership value of member x. This crisp value which is either positive or negative is added to the previous output to control the duty cycle of the switching devices in the power inverter so as to control the applied voltage across the armature winding and hence the speed of the motor [1].

Table 1: 7x7 FAM Matrix

CE / E	**NB**	**NM**	**NS**	**Z**	**PS**	**PM**	**PB**
NB	NB	NB	NB	NM	NS	NS	Z
NM	NB	NM	NM	NM	NS	Z	PS
NS	NB	NM	NS	NS	Z	PS	PM
Z	NB	NM	NS	Z	PS	PM	PB
PS	NM	NS	Z	PS	PS	PM	PB
PM	NS	Z	PS	PM	PM	PM	PB
PB	Z	PS	PS	PM	PM	PB	PB

Hybrid Fuzzy-PI Controller

The tuning of hybrid fuzzy-PI controller parameters are done by internal model control (IMC) [19] and its configuration is shown in Fig. 4.

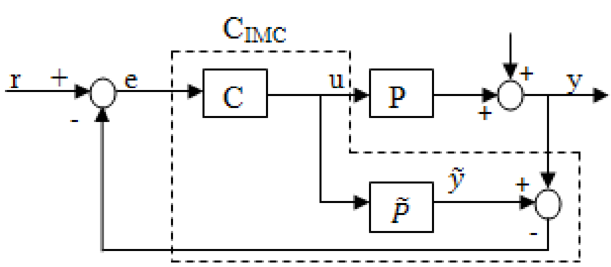


Figure 4: Internal model control configuration of hybrid fuzzy-PI controller

where P is the plant, $\tilde{P}$ is a nominal model of the plant, C_{IMC} is an arbitrary controller, r is the set-point, d is the disturbance, y and $\tilde{y}$ are the outputs of the plant and its nominal model respectively. The IMC structure is equivalent to the classical feedback controller,

$$C_{IMC}(s) = \frac{C(s)}{1 - C(s)\tilde{p}(s)} \tag{6}$$

$$C(s) = \frac{1}{\tilde{P}_s(s)} f(s) \tag{7}$$

where $\tilde{P}_-(s)$ is the minimum phase part of the plant model $\tilde{P}(s) = \tilde{P}_-(s)\tilde{P}_+(s)$, and f(s) is a low-pass filter with a steady-state gain of one, which typically has the form

$$f(s) = \frac{1}{(1 + t_c s)^n} \tag{8}$$

In analogy with the direct synthesis (DS) method, t_c is the desired closed-loop time constant. Parameter n is a positive integer. Set $u^{PI}(s) = C_{IMC}(s)$ because the nonlinear compensation is treated as the disturbance. The IMC-based self-tuning for fuzzy PI controller can be simplified as follows,

$$u^{PI}(s) = \frac{C(s)}{1 - C(s)\tilde{P}(s)} \tag{9}$$

Genetic-Based PI Controller

The genetic algorithm (GA) was inspired by the mechanism of natural selection, a biological process in which stronger individual is likely to be the winners in a challenging environment. GA uses a direct analogy of such natural evolution to do global optimization in order to solve highly complex problems [20]. In the beginning, an initial chromosome population is randomly generated. The chromosomes are candidate solutions to the problem. Then, the fitness values of all chromosomes are evaluated by calculating the objective function in a decoded form. So, based on the fitness of each individual, a group of the best chromosomes is selected through the selection process. In each generation, the genetic operators are applied to selected individuals from the current population in order to create a new population. Generally, the three main genetic operators namely reproduction, crossover and mutation are employed. By using different probabilities for applying these operators, the speed of convergence can be controlled. Crossover and mutation operators must be carefully designed, since their choice highly contributes to the performance of the whole genetic algorithm.

Reproduction: A part of the new population can be created by simply copying without any change in the selected individuals from the present population. Also new population has the possibility of selection by already developed solutions. There are a number of other selection methods available and it is up to the user to select the appropriate one for each process. All selection methods are based on the same principle i.e. giving fitter chromosomes a larger probability of selection. Here Roulette Wheel selection method is used.

Crossover: New individuals are generally created as offspring of two parents (i.e., crossover being a binary operator). One or more so called crossover points are selected (usually at random) within the chromosome of each parent, at the same place in each. The parts enclosed by the crossover points are then interchanged between the parents. The individuals resulting in this way are the offspring.

Mutation: A new individual is created by making modifications to one selected individual. The modifications can consist of changing one or more values in the representation or adding/deleting parts of the representation.

Objective Function: The most essential step in applying GA is to choose the objective functions that are used to evaluate fitness of each chromosome. The objective function is Integral of the Squared Error (ISE) [21],[22].

$$ISE = \int_{0}^{\tau} e(t)dt \tag{10}$$

Fitness Values: The PI controller is used to minimize the error signal, or define more rigorously, in term of error criteria: to minimize the value of performance indices mentioned below. Since smaller the value of performance indices of the corresponding chromosomes the fitter the chromosomes will be, and vice versa. The fitness value of the chromosomes is expressed as [22],[23].

$$\text{Fitness value} = \frac{1}{\text{Performance Index}} \tag{11}$$

From above introduction, it is evident that GA is a search algorithm that continuously repeats these steps: Reproduction, Crossover, and mutation, then make the new generation fitter than the old generation, until the requirements are completed. So in this paper GA_S are used to optimize PI parameters K_p and K_i. First, K_p and K_i are encoded to 16 bits string [24] as

K_p: 1010110011101011
K_i : 1011101011001000

The length of total chromosome is 32 bits. It is supposed that K_p and K_i are bounded in the closed intervals [0 K_{pm}] and [0 K_{im}] respectively. The decimal values of their corresponding binary strings are linearly related to their range boundaries K_{pm} and K_{im}. Secondly, according to GA_S operation: evaluation, crossover, mutation, K_p and K_i are optimized. After a prescribed number of generations, K_p and K_i are suitable enough to make system have good steady-state and dynamic performance.

3.5 ANFIS Controller

ANFIS is a fuzzy inference system based on Takagi-Sugeno model and this system uses given input and output data set to build fuzzy inference system. To start the

ANFIS learning; first, a training data set that contains the desired input/output data pairs of target systems to be modeled is required. The design parameters required for any ANFIS controller are number of data pairs, training data sets, checking data sets, fuzzy inference systems for training, number of epochs to be chosen to start the training, learning results to be verified after mentioning the step size [12].

Figure 5: Logic of ANFIS controller

Logic of ANFIS controller is shown in Fig. 5. The designed ANFIS has two inputs namely, the actual motor speed and reference speed while the output is the torque, which is used to generate current. Here bell shaped membership function is used.

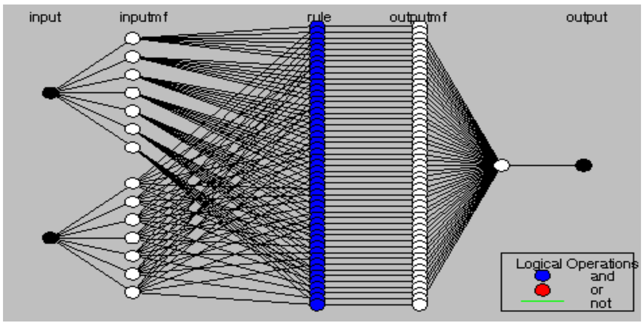


Figure 6: Structure of ANFIS speed controller

Structure of ANFIS speed controller is shown in Fig. 6. It is a five-layer feedforward fuzzy neural networt. Every layer has its definite meaning.

Layer 1 : (Input Layer) Input layer represents input variables of controller, they are speed error and its variance ratio referred as x_1, x_2 respectively. This layer just supplies the input values x_i to the next layer, where i= 1 to n.

Layer 2 : (Fuzzification Layer) This layer (membership layer) checks for the weights of each membership functions (MFs). It receives the input values from the 1st layer and act as MFs to represent the fuzzy sets of the respective input variables. Further, it computes the membership values which specify the degree to which the input value x_i belongs to the fuzzy set, which acts as the inputs to the next layer.

Layer 3 : (Rule layer) Each node (each neuron) in this layer performs the pre-condition matching of the fuzzy rules, i.e., they compute the activation level of each rule, the number of layers being equal to the number of fuzzy rules. Each node of these layers calculates the weights which are normalized.

Layer 4 : (Defuzzification Layer) It provides the output values "y" resulting from the inference of rules. Connections between the layers l_3 & l_4 are weighted by the fuzzy singletons that represent another set of parameters for the neuro fuzzy network.

Layer 5 : (Output Layer) It sums up all the inputs coming from the layer 4 and transforms the fuzzy classification results into a crisp values.

The ANFIS structure is tuned automatically by least-square-estimation and back propagation algorithm. The above mentioned optimization procedures are repeated by using sample data until proper error index or the maximum number of training is achieved. After learning and training, the test data can be used to check the controller to ensure effectiveness of the controller.

Simulation Results

The BLDC motor parameters are shown in Table 2 and its performance is obtained by simulation using MATLAB/ SIMULINK 7.5. The simulation was run for 1 seconds (simulation time). The dc link voltage V_S is chosen as 30 V and the maximum load current as 10 A. The transformer turns ratio is chosen as 1 : 4 and the leakage inductance of the primary and secondary windings are 0.7 µH and 2.8 µH respectively. With the equivalent transformer inductance L_r = 1.5 µH, the resonant capacitance C_r is 0.1 µF. $\Delta t_1 + \Delta t_2 + \Delta t_3$ is determined for various load current I_o. Considering the turn off time of switch lagging ΔT_1 and pulse width ΔT_2 they are set to 2.1µs and 5µs respectively. The frequency of the PWM is 20 kHz.

Table 2: The Parameters of BLDC Motor

Parameters		Value
Rated Input Voltage	V_{in}	24 V
Rated Armature Current	I_a	10.4 A
Rated Speed	N	1500 rpm
Armature Resistance	R_a	0.3 Ω
Armature Inductance	L_a	1.15 mH
Magnetic Flux Linkage	Φ	wb
No. of Poles	P	4
Moment of Inertia	J	0.002 kg.m^2
Friction Factor	F	0.0001 Nm.s

Substituting the values of the motor parameters and using Ziegler Nichols method, the tuning parameters of PI controller are determined as K_p=3.3 and K_i=300. The electromagnetic torque and speed response curves obtained for PI controller is shown in Fig. 7. Fig. 7(a) shows electromagnetic torque for the reference speed of 1500 rpm and load torque is applied at 0.4 seconds. From Fig. 7(a), it is observed that the starting torque is 1.7 Nm and torque ripple has the amplitude variation of ± 0.2 Nm.

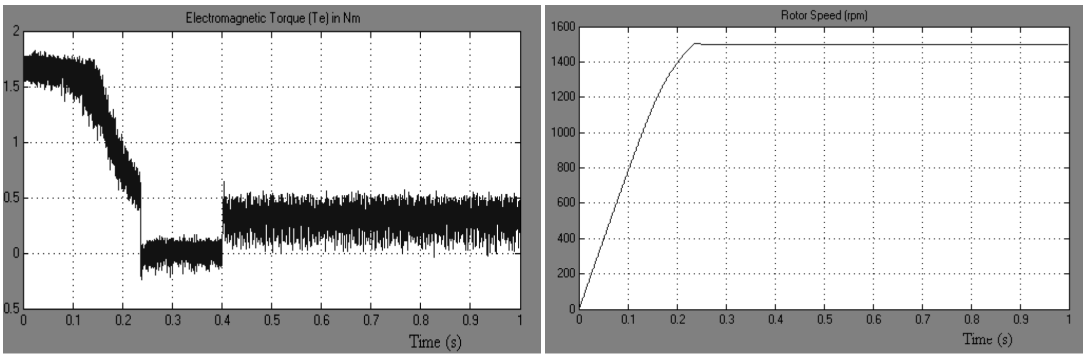
From speed response curve of Fig. 7(b), it is observed that the rise time is 0.19 seconds, overshoot is 0.266% and settling time is 0.27 seconds with PI controller.

From Fig. 8(a), it is observed that the starting torque is 6 Nm and torque ripple has the amplitude variation of ± 0.1 Nm. From the speed response curve of Fig. 8(b) it is observed that the rise time of the motor with fuzzy controller is about 0.114 seconds, overshoot is almost eliminated and settling time is 0.18 seconds.

From Fig. 9(a), it is observed that the starting torque is 6.8 Nm and torque ripple has the amplitude variation of ± 0.15 Nm. From speed response curve of Fig. 9(b), it is observed that the rise time is 0.112 seconds, overshoot is almost eliminated and settling time is 0.16 seconds with hybrid fuzzy-PI controller [25].

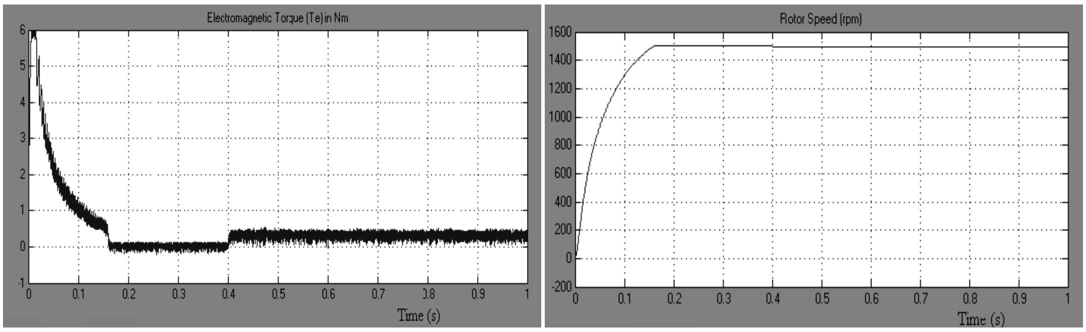
Similarly by substituting the values of the motor parameters and using genetic algorithms, the tuning parameters of genetic based PI controller are determined as K_p= 4.0639 and K_i= 0.7411. The Parameters of genetic algorithm are shown in Table 3. The electromagnetic torque response and speed response curves obtained with genetic based PI controller is shown in Fig. 10. Fig. 10(a) shows electromagnetic torque for the reference speed of 1500 rpm and load torque is applied at 0.4 seconds. From Fig. 10(a), it is observed that the starting torque is 6.9 Nm and torque ripple has the amplitude variation of ± 0.1 Nm. From speed response curve of Fig. 10(b), it is observed that the rise time is 0.091 seconds, overshoot is almost eliminated and settling time is 0.12 seconds with genetic based PI controller.

The time response curves of the BLDC motor with ANFIS based controller is shown in Fig. 11. From Fig. 11(a), it is observed that the starting torque is 7 Nm and torque ripple is with the amplitude variation of ± 0.1 Nm. From speed response curve of Fig. 11(b), it is observed that the rise time is 0.072 seconds, overshoot is almost eliminated and settling time is 0.1 seconds. Waveforms of PWM, main switch S_6,auxiliary switch S_b gate signal, switch S_6 voltage drop $U_{s6,}$ and transformer primary winding current i_{Lr} under low load current (I_o = 3A) and high load current (I_o = 10A) are shown in Fig. 12.



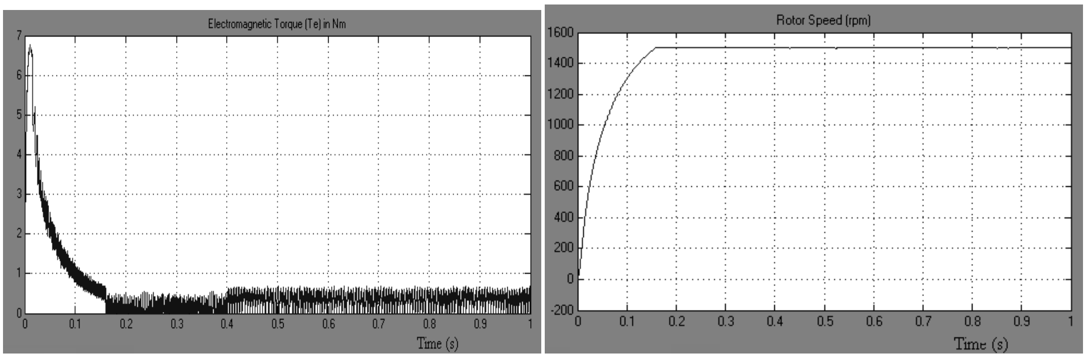
(a) (b)

Figure 7: (a) Electromagnetic torque and (b) speed response at 1500 rpm with PI controller



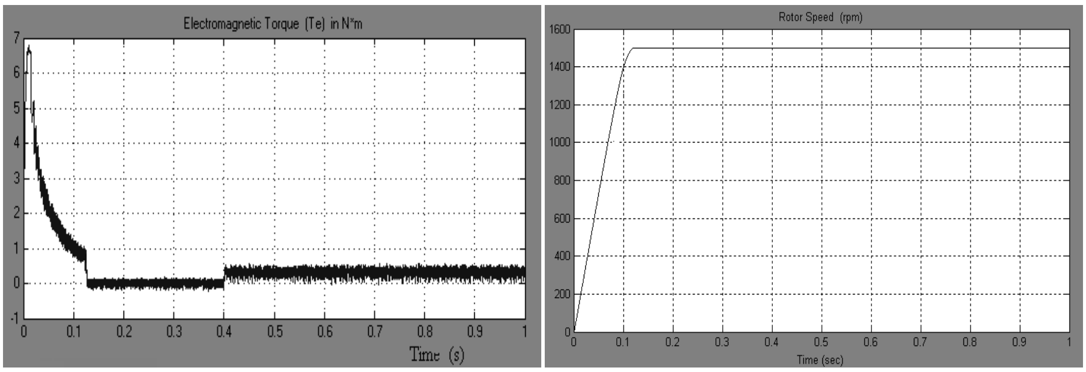
(a) (b)

Figure 8: (a) Electromagnetic torque and (b) speed response at 1500 rpm with fuzzy controller



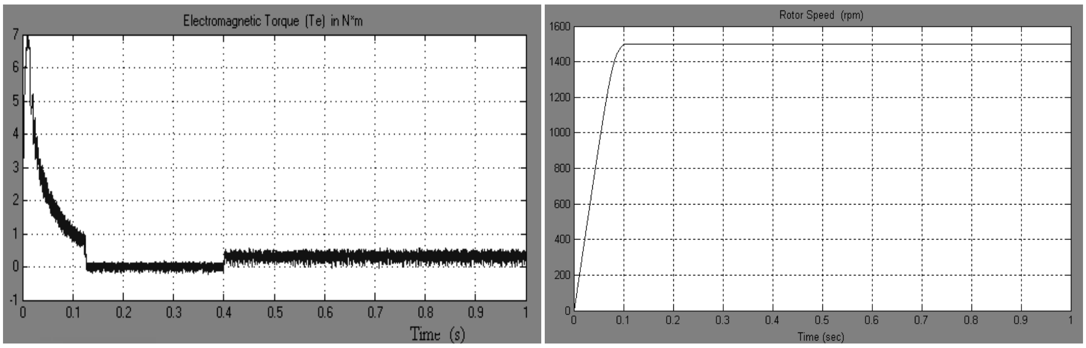
(a) (b)

Figure 9: (a) Electromagnetic torque and (b) speed response at 1500 rpm with hybrid fuzzy-PI controller



(a) (b)

Figure 10: (a) Electromagnetic torque and (b) speed response at 1500 rpm with GA-based PI controller

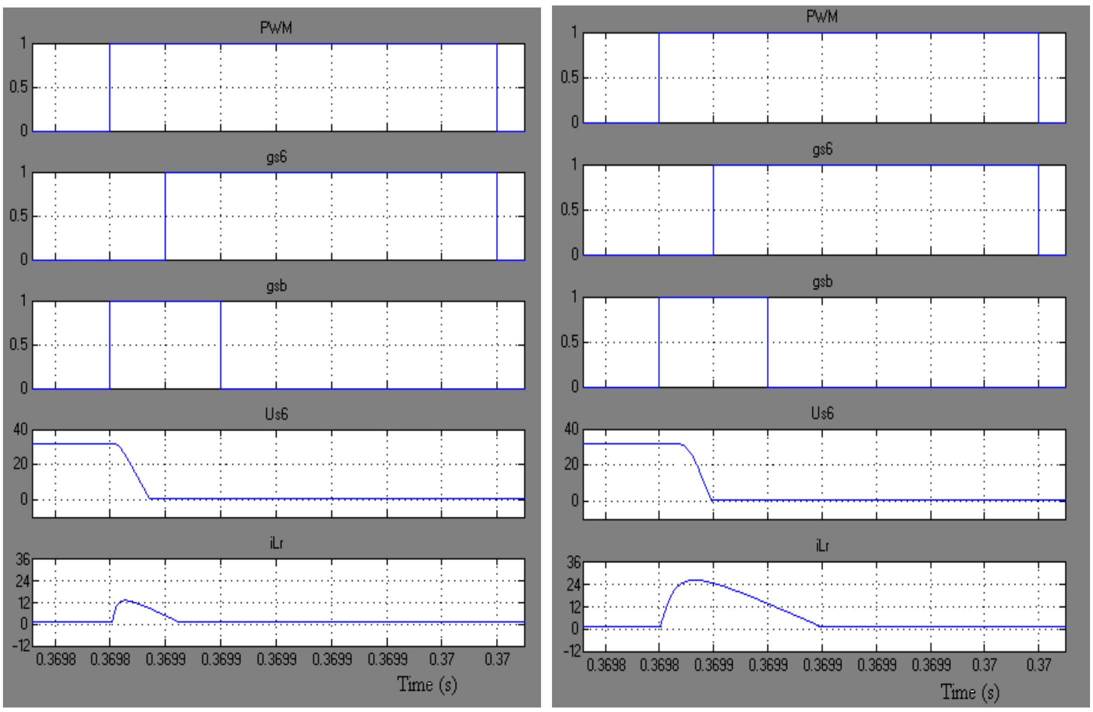


(a) (b)

Figure 11: (a) Electromagnetic torque and (b) speed response at 1500 rpm with ANFIS controller

Table 3: The Parameters of Genetic Algorithm

Parameters	Value
Population Size	30
Generation Number	250
Selection Method	Roulette Wheel
Crossover probability	85 %
Mutation Probability	0.2 %



(a) (b)

Figure 12: Simulation waveforms of PWM, S_6,S_b gate signal, U_{s6}, and i_{Lr} under various load current (a) low load current (I_o = 3A) and (b) high load current (I_o = 10A)

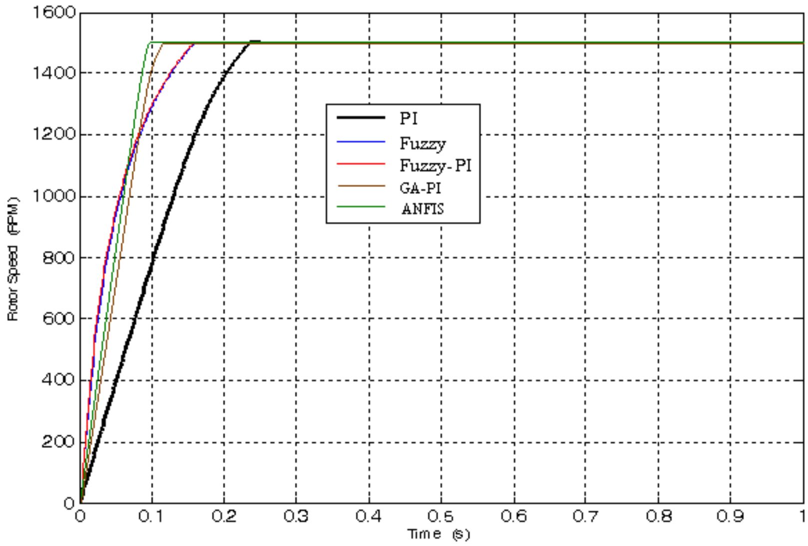


Figure 13: Speed response curves for PMBLDC motor obtained using PI, fuzzy, hybrid fuzzy-PI, GA-PI and ANFIS controllers

The Speed response curves of PI, fuzzy, hybrid fuzzy-PI, GA-PI and ANFIS controller are redrawn for comparison as shown in Fig. 13. The rise time, settling time, percentage overshoot, starting torque and torque ripples are considered for performance evaluation of these controllers and are shown in Table 4. From Table 4 it is evident that, the performance specifications obtained using ANFIS controller is better than those obtained using other controllers. From the results obtained, it can be seen that the soft switching inverter performs well under various load currents. Due to soft switching condition, the switching power losses are low. The efficiency Vs torque curves of hard switching and soft switching under rated speed are shown in Fig. 14, and it is observed that efficiency is improved with the soft switching inverter. This validates the soft switching inverter topology used in this paper.

Table 4: Performance Analysis of PI, Fuzzy, Hybrid Fuzzy-PI, GA-PI and ANFIS Controller

Controller	Delay Time (Sec)	Rise Time (Sec)	Peak Time (Sec)	Percentage Overshoot (%)	Settling Time (Sec)	Starting Torque (Nm)	Torque Ripples (Nm)
PI	0.096	0.19	0.24	0.266	0.27	1.7	± 0.2
Fuzzy	0.036	0.114	0.18	0	0.18	6	± 0.1
Hybrid Fuzzy-PI	0.034	0.112	0.16	0	0.16	6.8	± 0.15
GA-Based PI	0.062	0.091	0.12	0	0.12	6.9	± 0.1
ANFIS	0.041	0.072	0.1	0	0.1	7	± 0.1

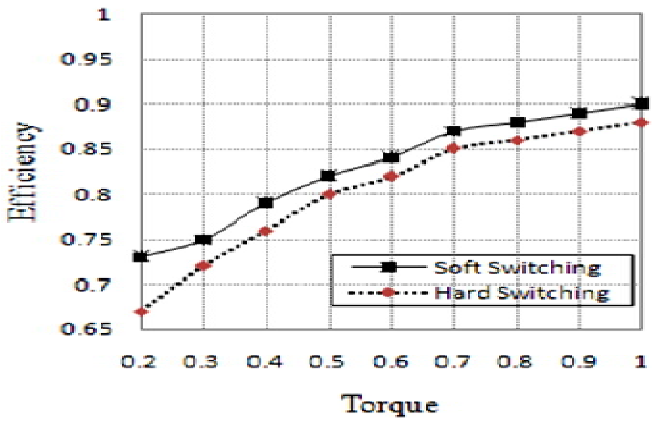


Figure 14: Efficiency Vs torque curves of hard switching and soft switching under rated speed

5. Conclusion

The dynamic behavior of the BLDC drive system with conventional PI, fuzzy, hybrid fuzzy-PI, GA-PI and ANFIS controllers are presented and compared for torque and speed operation. It is observed that the ANFIS controller gives much better dynamic response for the system. From the results of proposed inverter topology, it is observed that all the switches work under soft switching condition and freewheeling diodes are turned off under zero current condition which greatly reduces the reverse recovery problem of the diodes. Further, voltage stress on all the switches is very low and it is not greater than the dc supply voltage. The switching acoustic noise is very much reduced as the switching frequency is as high as 20 kHz and moreover dv/dt and di/dt are reduced significantly and as a result EMI is reduced. Furthermore, in the proposed method very simple auxiliary switches control scheme is needed and the normal operation of the inverter is essentially the same as that of the hard switching inverter. The performance characteristics of conventional PI and hybrid intelligent controllers are compared interms of delay time, rise time, peak time, percentage overshoot, settling time, starting torque and torque ripples. It is validated by simulation results that ANFIS controller performs better than other controllers proposed in this paper. Therefore, ANFIS controller based soft switching inverter can be implemented as future work.

References

[1] Muruganantham, N., and Palani, S., 2011, "Novel Soft Switching Inverter for Brushless DC Motor using Fuzzy Logic", International Journal of Electrical Engineering, 4(5), pp. 601-616.

[2] Uddin M. N., and Rahman M. A., 1999, "Fuzzy Logic Based Speed Control of an IPM Synchronous Motor Drive", in Proc. 1999 IEEE Canadian Conf. Electr. Comput. Eng., May 9–12, pp. 1259–1264.

[3] Panda S. R., Lim J. M. S., Bash P. K., and Lock K. S., 1997, "Gain Scheduled PI Speed Controller for PMSM Drive," in Proc. 23rd Int. Conf. Ind. Electron., Contr. Instrum., (IECON 97), pp. 925–931.

[4] Panda S., Zhu X., and Dash P., 1997, "Fuzzy Gain Scheduled PI Speed Controller for Switched Reluctance Motor Drive," in Proc. 23rd Int. Conf. Industrial Electronics, Control and Instrumentation, IECON 97, pp. 989–994.

[5] Zerikat M., and Chekroun S., 2007, "Design and Implementation of a Hybrid Fuzzy Controller for a High Performance Induction Motor", in Proc. World Academy of Science, Engineering and Technology, 20, pp. 263–269.

[6] McMurray, W., 1989, "Resonant Snubbers with Auxiliary Switches," Proc. IEEE Industry Applications Society Annual Meeting, pp. 289–834.

[7] Ng, K. C., Li, Y., Murray-Smith, D. J., and Sharman, K. C., 1995, "Genetic Algorithms Applied to Fuzzy Sliding Mode Controller Design", Proc. 1st IEE/IEEE Int. Conf. on GA in Eng. Syst., Innovation Appl. Scheffield.

[8] Ng, K. C., and Li, Y., 1994, "Design of Sophisticated Fuzzy Logic Controllers using Genetic Algorithms", in Proc. 3rd IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE'94), 3, Orlando, FL, USA.

[9] Viljamaa, P., 2000, "Fuzzy Gain Scheduling and Tuning of Multivariable Fuzzy Control- Methods of Computing in Control Systems", Technologic Doctoral Thesis, Temper University of technology, Finland.

[10] Mahony, T.O., Downing, C.J., and Fatla, K., 2000, "Genetic Algorithm for PID Parameter Optimization: Minimizing Error Criteria, Process Control and Instrumentation", University of Stracthclyde, pp. 148- 153.

[11] Varsek, A., Urbacic, T., and Filipic, B., 1993, "Genetic Algorithms in Controller Design and Tuning", IEEE Trans. Sys.Man and Cyber, 23(5), pp.1330-1339.

[12] Aware M. V., Kothari A. G., Choube S. O., 2000, "Application of Adaptive Neuro-Fuzzy Controller (ANFIS) for Voltage Source Inverter Fed Induction Motor Drive", The Third International Electronics and Motion Control Conference - IPEMC 2000, 2, pp. 935-939.

[13] Miller, T. J. E., 1989, Brushless Permanent-Magnet and Reluctance Motor Drives, Clarendon, Oxford, U.K.

[14] Zhi Yang Pan, and Fang Lin Luo, 2005, "Novel Resonant Pole Inverter for Brushless DC Motor Drive System", IEEE Transactions on Power Electronics, 20(1), pp. 173 – 181.

[15] Ziegler, J. G., and Nichols, N. B., 1942, "Optimum Setting for Automatic Controllers", ASME Transactions, pp.759-768.

[16] Guillemin, P.,1996, " Fuzzy Logic Applied to Motor Control ", IEEE Transaction on Industrial Application, 32(1), pp.51–56.

[17] Govind, N., 1994, "A Fuzzy Rule Based 5 HP DC Motor Speed Controller", *Fuzzy logic handbook*, Intel Corporation, no. 272589-001, pp. 95–112.

[18] Bay O.F. Bal, and Demirbas, S., 1996, "Fuzzy Logic Based Control of Brushless Servo Motor Drive", Proc. 7th International Power Electronics and Motion Control Conference, pp. 448–452.
[19] Xiao-Gang Duan, and Han-Xiong Li and Hua Deng., 2008, "A Simple Tuning Methods for Fuzzy PID Control", in proc. of Int. Conf. on Fuzzy Systems, (FUZZ 2008), pp. 271-275.
[20] Goldberg, D. E., 1989, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Co., Inc.
[21] Zwe-Lee Gaing, 2004, "A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System", IEEE Transactions on Energy Conversion, 19(2), pp.384-391.
[22] Mahony, T. O., Downing, C. J., and Fatla, K., 2000, "Genetic Algorithm for PID Parameter Optimization: Minimizing Error Criteria, Process Control and Instrumentation", University of Stracthclyde, pp. 148- 153.
[23] Teng, T. K., Shieh. J. S., and Chen, C. S., 2003, "Genetic Algorithms Applied in Online Auto-Tuning PID Parameters of a Liquid-Level Control System", Transaction of the Institute of Measurement and control, 25(5), pp.433-450.
[24] Michel Fliess and Cedric Join, 2008, "Intelligent PID Controllers", 16th Mediterranean Conference on Control and Automation June 25-27, pp.326-331.
[25] Muruganantham, N., and Palani, S., 2012, "Hybrid Fuzzy-PI Controller Based Speed Control of PMBLDC Motor using Soft-Switching Inverter", European Journal of Scientific Research, 78(2), pp. 259-272.