We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Design of a Real Coded GA Based Fuzzy Controller for Speed Control of a Brushless DC Motor

Omer Aydogdu and Ramazan Akkaya

Additional information is available at the end of the chapter

1. Introduction

Fuzzy controllers are nonlinear elements used in the control of linguistically defined systems, which can not be modeled accurately. Besides, they are an effective approach for various complex and ill-defined systems [1,2]. In design of fuzzy controllers, there is not well defined approach. The sophisticated and tedious design process is usually implemented by an expert. The success of the controller depends on the knowledge and skill of the expert. In some cases, even a very experienced and skillful expert's extensive efforts may not yield optimal solution for fuzzy controller design. The optimal design of fuzzy controller has a critical role for their more widespread and effective use. The design inherently requires the determination of a great deal of features and parameters. For this reason, fuzzy controller design problem has a number of local values in a large solution space in the direction of a number of objectives. The conventional trial-and-error based method makes the solution of the problem very difficult [3].

Genetic algorithms (GAs) are optimization strategies performing a stochastic search by iteratively processing 'populations' of solutions according to their fitness, i.e. a predefined scalar index of satisfaction of the design objectives. In control applications, the fitness is usually related to performance measures such as integral error, settling time, and so on. Fitness function may contain more than one objective like the minimization of settling time, steady state error and maximum overshoot. Thus, fitness function can be addressed as a multi objective function. GAs are effective in solving multi-objective optimization problems [4,5].

Most of the GA approaches in use represent the constraint variable using binary form of coding. One of the major disadvantages of using binary coding is the slow convergence

speed of the fitness function. Binary coding is also not at all efficient to be used in computer memory. Therefore, the use of real coded genetic algorithm can overcome the inefficient use of computer memory and can contribute the performance. This contribution becomes clear when a lot of parameters are needed to be adjusted in the same problem and higher precision is required for the final result. In literature, for real valued numerical optimization problem, floating point representations were proven to outperform binary representations because they are more consistent, more precise and they lead to faster execution. Also, real coded GA is inherently faster than the binary code GA, because the chromosomes do not have to be decoded prior to the evaluation of the objective function [6].

Brushless DC (BLDC) motors are one of the motor types rapidly gaining popularity. Due to their favorable electrical and mechanical properties like high starting torque, high efficiency and noiseless operation, the BLDC motors are widely used in various consumer and industrial systems such as actuation, robotics, machine tools, servo motor drives, home appliances, computer peripherals, and automotive applications [5,7]. Operation of the BLDC motors requires non-linear control due to their non-linear characteristics and presence of sensors to estimate rotor position. Moreover, owing to some drawbacks of position sensors such as cost, space requirement and instability, sensorless speed control has recently gained importance [8]. It is possible to determine when to commutate the motor drive voltages by sensing the back emf voltage on an undriven motor terminal during one of the drive phases. Use of fuzzy controllers that have non-linear processes such as fuzzification, defuzzification and fuzzy inference is suitable for BLDC motor control.

In this study, sensorless speed control of the BLDC motor with real coded GA based fuzzy controller has been designed, simulated and practically implemented. ADSP-21992 digital signal processor (DSP) is used to realize the conventional and optimal fuzzy controller algorithms and sensorless speed control of the BLDC motor has been experimentally implemented successfully.

2. Real coded GA based fuzzy controller

The realized control system general diagram is shown in Figure 1. As shown in the figure, the control system includes two closed loops. The inner loop is the fuzzy controller loop that accomplishes speed control of BLDC motor. The outer loop, which processes the fuzzy controller and system operations in background, is the genetic algorithm loop that tunes the controller parameters in regard to performance index of the control system.

2.1. Fuzzy controller

In this study, Mamdani type fuzzy controller, which has five blocks namely normalization, fuzzifier, inference mechanism, defuzzifier and denormalization, has been used [8]. Block diagram of the real coded GA based fuzzy controller consisting two inputs (e_1, e_2) and one output (u) is shown in Figure 2.
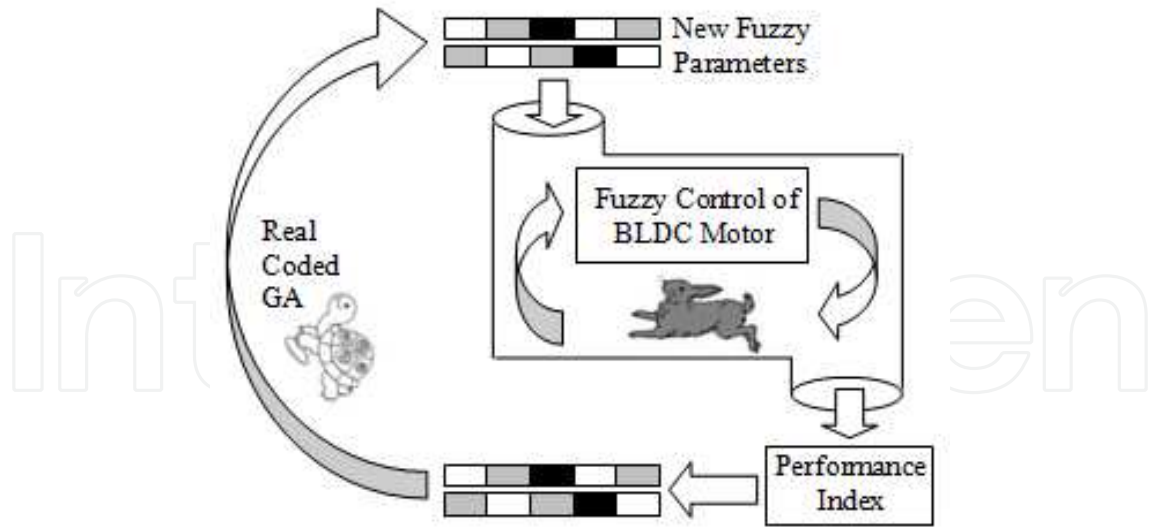
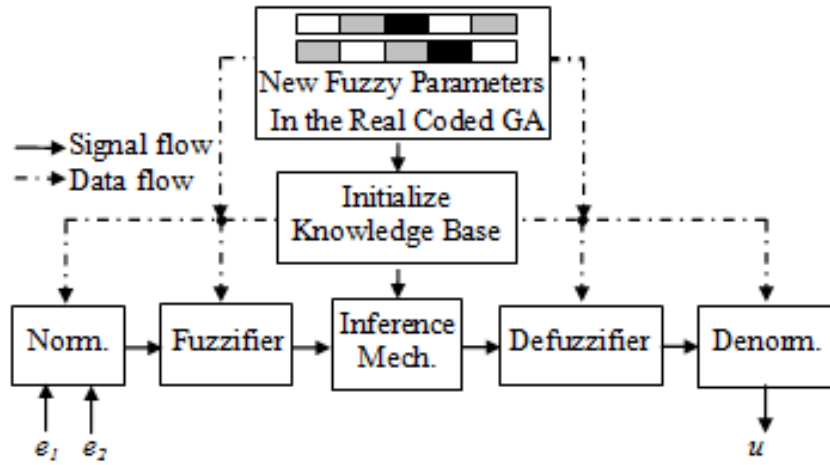**Figure 1.** Real Coded GA based fuzzy control of the BLDC motor

Figure 2. Block diagram of the Real Coded GA based fuzzy controller

In closed-loop control systems, the use of error (e_1) and the change in error (e_2) as controller input is a universal approach. In the implemented fuzzy controller, error and change in error have been used as inputs. As given in Eq. (1), the error is the difference between the reference speed and the actual rotor speed.

$$e_1(t) = \omega^*(t) - \omega(t) \tag{1}$$

Here, $e_1(t)$ is the speed error, $\omega^*(t)$ is the reference speed and $\omega(t)$ is the actual motor speed. The change in error $e_2(t)$ is determined by Eq. (2) [9,10].

$$e_2(t) = \frac{d}{dt} e_1(t) \tag{2}$$

In a fuzzy control system as shown in Figure 2, two normalization parameters (n_1, n_2) for inputs (e_1, e_2) and one denormalization parameter (n_3) for output (u) is defined. In

normalization process, the input values are scaled in the range [-1,+1] and in denormalization process the output values of fuzzy controller are converted to a value depending on the terminal control element. Obtaining the normalization and denormalization parameters of fuzzy controller is important for system stability.

In the fuzzifier process, the crisp input values (e_1, e_2) are converted into fuzzy values. Also, the fuzzy values obtained in fuzzy inference mechanism must be converted to crisp output value (u) by defuzzifier process. For this purpose, triangular fuzzy membership function is defined for each input and output values by seven clusters. Figure 3, illustrates the membership function used to fuzzify two input values (e_1, e_2) and defuziffy output (u) of the realized fuzzy controller. For seven clusters in the membership functions as shown in Figure 3, seven linguistic variables are defined as; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM) and Positive Big (PB). Initially, the overlap rates of membership functions are 50%. As illustrated in Figure 3, peak or bottom points of membership functions to be tuned are defined as a_1 and a_2 for e_1, b_1 and b_2 for e_2, c_1 and c_2 for u. Therefore, the design of optimal fuzzy controller requires optimization of at least six parameters (a_1, a_2, b_1, b_2, c_1, c_2) by using real coded GA for fuzzification and defuzzification processes that are both nonlinear.
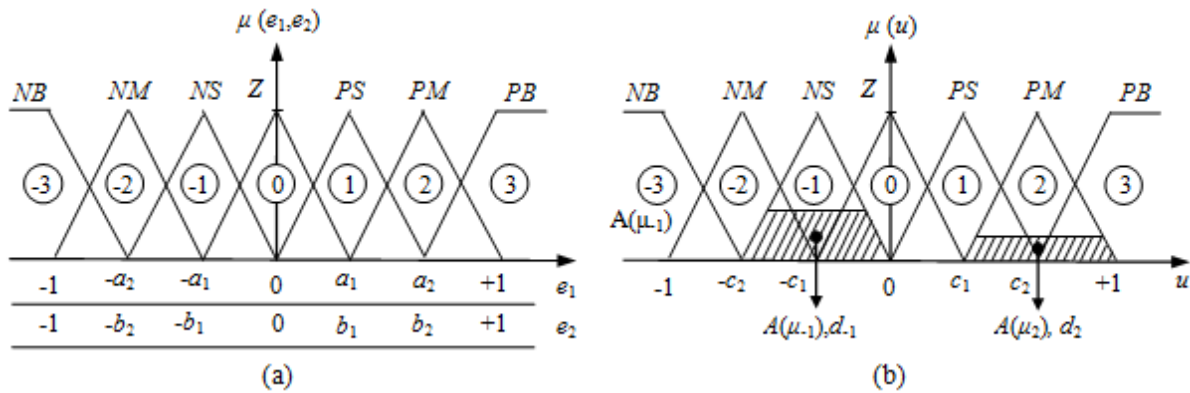

Figure 3. Membership functions of (a) input (e_1,e_2), (b) output (u)

In this study, the center-of-gravity method given by Eq. (3) has been used in defuzzifier process for simulation and real time DSP application.

$$u = \frac{\sum\limits_{i=1}^{m} d_i A(\mu_i)}{\sum\limits_{i=1}^{m} A(\mu_i)} \tag{3}$$

Here, u is the fuzzy controller output, d_i is the distance between i^{th} fuzzy set and the center, $A(\mu_i)$ is the area value of ith fuzzy set.

The rule definition is subjective and based on the expert's knowledge and experience. For the system with two inputs and seven membership functions in each range, it leads to a 7x7 decision table and 49 fuzzy rules. For example, two fuzzy rules are decribed as;

if e_1 is *NB* **and** e_2 is *NB* **then** *u* is R_1,
if e_1 is *NB* **and** e_2 is *NM* **then** *u* is R_2,

Sliding mode rule base table used by fuzzy controller is given in Table 1. In the sliding mode rule base, when an assumption is made such that $R_1 = -R_{13}$, $R_2 = -R_{12}$, $R_3 = -R_{11}$, $R_4 = -R_{10}$, $R_5 = -R_9$, $R_6 = -R_8$, it is required to determine a minimum of seven parameters (R_1-R_7) by using real coded GA.

Input-e_1	Input-e_2						
	NB	*NM*	*NS*	*Z*	*PS*	*PM*	*PB*
NB	R_1	R_2	R_3	R_4	R_5	R_6	R_7
NM	R_2	R_3	R_4	R_5	R_6	R_7	R_8
NS	R_3	R_4	R_5	R_6	R_7	R_8	R_9
Z	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}
PS	R_5	R_6	R_7	R_8	R_9	R_{10}	R_{11}
PM	R_6	R_7	R_8	R_9	R_{10}	R_{11}	R_{12}
PB	R_7	R_8	R_9	R_{10}	R_{11}	R_{12}	R_{13}

Table 1. Sliding mode rule base

The developed fuzzy logic uses the min-max compositional rule of inference. The inference mechanism of fuzzy controller is implemented in regard to the rule base given by Eq. (4).

$$\mu_i(u) = min\left(\mu_i(e_1), \mu_i(e_2)\right)$$
(4)

Fuzzy controllers are nonlinear tools because of the nonlinearity of logical inference, fuzzifier and defuzzifier processes. As mentioned earlier, design of fuzzy controller requires determination of a minimum of 16 parameters comprised of three normalization parameters, six membership function parameters and seven sliding mode rule base parameters. The parameters are decided in regard to the symetrical properties of the controller and that imposes limitations for the controller. The reduction of the number of parameters gives acceptable results and it can be preferred in design process to simplify the sophisticated optimization process.

2.2. Optimization of fuzzy controller by means of real coded GA

In problem solving, the precision of the numbers is crucial. In binary coded GA, the accuracy is limited by the size of the chromosomes. However, the use of real coded GA, which can be coded by real numbers, is advantageous. Real coded GA is more accurate and occupies less space in memory. In literature, it has been reported that, real coded GAs operate faster than binary coded GAs and they can converge to global optimum faster [11]. Also, in the optimization of systems consisting a great deal of parameters to be optimized as in fuzzy controllers, the chromosomes in the binary coded GA becomes too long and the parameter accuracy can not be handled. However, in the developed algorithm, the parameters are coded by integer number set and the parameter accuracy can be determined arbitrarily. Also, in the

developed algorithm, binary coding can be accomplished in the limited valued parameters for the optimization of sliding mode rule base table (R_1, R_2, R_3, R_4, R_5, R_6, R_7).

In Figure 4, the flowchart of real coded GA used in the study is shown. Some of the GA parameters such as the possible lower and upper limit values of parameters, the precision of parameters, the termination criterion or loop number, the mutation probability, the crossover probability, the population number and elitism property is required to be initialized.
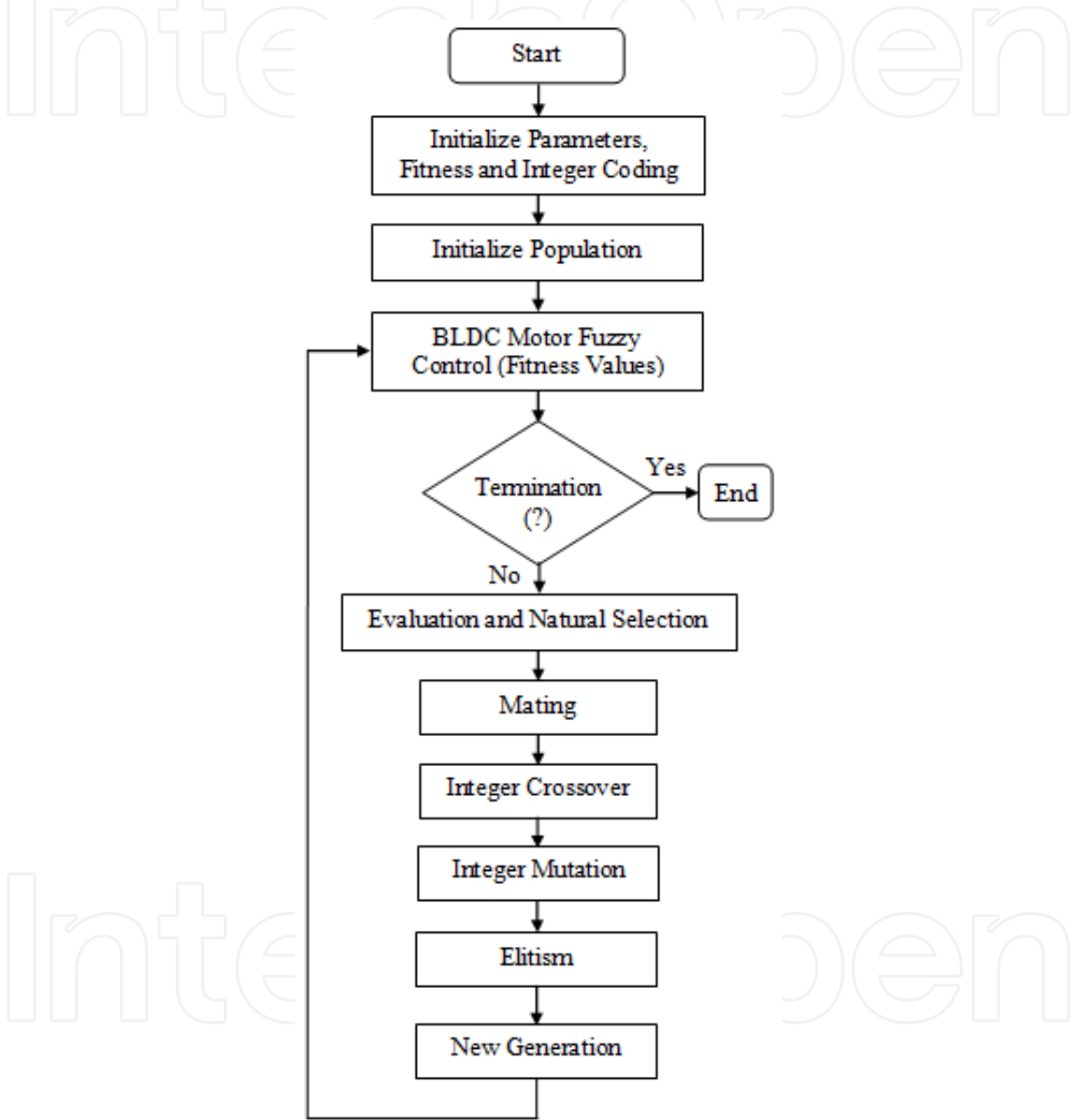

Figure 4. Flowchart of the Real Coded GA based fuzzy controller

In the proposed algorithm, the normalization parameters (n_1, n_2) and de-normalization parameter (n_3), the membership function parameters (a_1, a_2, b_1, b_2, c_1, c_2) and rule base (R_1, R_2, R_3, R_4, R_5, R_6, R_7) of the fuzzy controller have been optimized. Two approaches have been used in determination of parameters in optimal design of fuzzy controller by using real coded GA. In the first approach, the parameters for optimal fuzzy controller have been determined

sequentially. This approach is similar to the self-tuning adaptive system employed in adaptive control systems. The second approach is a method based on the simultaneous optimization of normalization factors, membership functions and rules. Fuzzy controller parameters are not fully independent of each other [12]. For this reason, simultaneously tuning of all the parameters is the optimum solution. However, the large number of fuzzy control parameters makes the second approach more difficult to implement.

The performance index defined for the fuzzy controller is given in Eq.(5) [3]. This equation has also been used as multiple objective function employed in optimization process for real coded GA.

$$J_{in} = \int_0^{t_1} |e_1| \, dt + \int_{t_1}^{t} |e_1| \, t \, dt + 6 \int_0^{t} \delta \left(\frac{d\omega}{dt} \right) |e_1| \, dt \tag{5}$$

In Eq.(5), J_{in} is the performance index used as fitness value in real coded GA, e_1 is the error value and t_1 is the settling time of the reference speed in BLDC motor fuzzy control system.

In Figure 5 (a), the coding of three normalization parameters in six digit number string in real coded GA is depicted. Here, in six digit number string, two of it is assigned for integer part and four of it is for the fractional part. As shown in Figure 5 (b), a total of six coding has been implemented for three membership functions. Here, a_1 and a_2 are the membership function parameters defined for input (e_1); b_1 and b_2 are the membership function parameters defined for input (e_2); and c_1 and c_2 are the membership function parameters defined for output (u). All obtained membership function parameters are less than one. In Figure 5 (c), the sliding mode rule base parameters presented in Table 1 are shown. These parameters take integer values in the range [1,2,3]. Therefore, each parameter is denoted by one digit. As shown in Figure 5, a total of 63 digits have been used for all of the codes. Additionally, the precision of parameters can be increased by increasing the number of digits. In coding structure shown in Figure 5, the precision of membership functions is 0.0001.
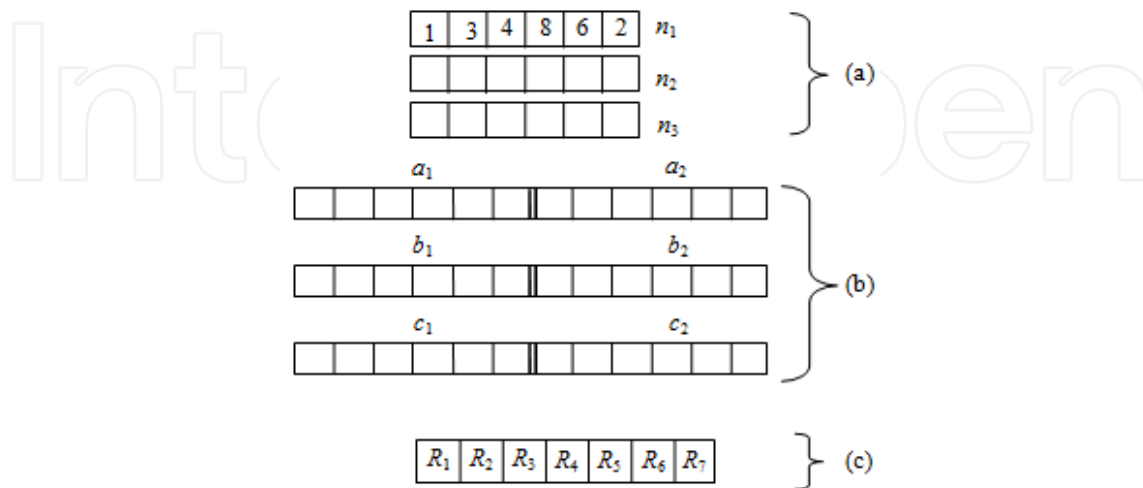

Figure 5. Coding of the fuzzy parameters in real coded GA, (a) Normalization, (b) Membership functions, and (c) Sliding mode rule table

In the implemented algorithm, the initial population can be preferably set by the user or can be randomly appointed a value in regard to Eq.(6). An important feature of random generator is that in each execution of the algorithm different values are initialized.

$$N_{ipop} = (P_h - P_\ell) \ Random \ (P_{pop}) + P_\ell \tag{6}$$

Here, N_{ipop} is the initial population, P_h and P_ℓ are the minimum and maximum values the parameter can have and P_{pop} is the random number generated between zero and one. Fitness value of each chromosome in initial population is determined by using performance index which is given by Eq.(5). As shown in the Figure 4, if the termination criterion is met, real coded GA loop ends but optimal fuzzy control system keeps operating. Else, next step in real coded GA is executed. In this study, Roulette wheel method is employed in natural selection process. In binary coded GA, the crossover operation can be implemented by using methods like single-point or two-point crossover. However, in real coded GA, the mixing methods give better results in crossover operation. In mixing method, the values of two parameter are compared and new generations are produced using Eq. (7).

$$P_{new} = \beta P_{an} + (1 - \beta)P_{bn} \tag{7}$$

where β is a number randomly generated between 0 and 1, P_{an} is the nth parameter of mother chromosome, P_{bn} is the nth parameter of father chromosome [13,14].

Genetic algorithms sometimes converge prematurely. In order to prevent convergence to a local point, new solutions are obtained by means of mutation. In this study, the number of parameters to be applied to the mutation operation has been determined by multiplying the total number of parameters with mutation rate. The elements to apply mutation operation are determined and they are replaced by randomly generated numbers.

In the developed algorithm, the elitism, a mechanism in which the individuals with the best fitness values in previous population are guaranteed their place in the next population, is implemented by selecting. Later, new fitness value is determined again for new population. Optimization process goes on until the termination criterion is met.

3. Sensorless control of BLDC motor

The BLDC control drive system is based on the feedback of rotor position, which is obtained at fixed points typically every 60 electrical degrees for six-step commutations of the phase currents. So, the operation of BLDC motor requires a control system and position sensors to estimate rotor position. Rotor position is sensed using Hall effect sensors embedded into the stator frequently. However, sensorless speed control has recently gained importance owing to the elimination of some drawbacks of sensors such as cost, space requirement and instability. Figure 6 shows basic block diagram of sensorless control of BLDC motor with real coded GA based fuzzy controller.

In Figure 6, ω^* is the reference speed (rad/sec), ω is the actual rotor speed (rad/sec), θ is the rotor position (degree), u is the control signal used to reference moment (N-m), i_a, i_b, i_c are

the actual phase currents (Amper), i_a^*, i_b^*, i_c^* are the reference phase currents (Amper), S_1-S_6 are switches of the inverter and V_{dc} is the supply voltage of the inverter (Volt).
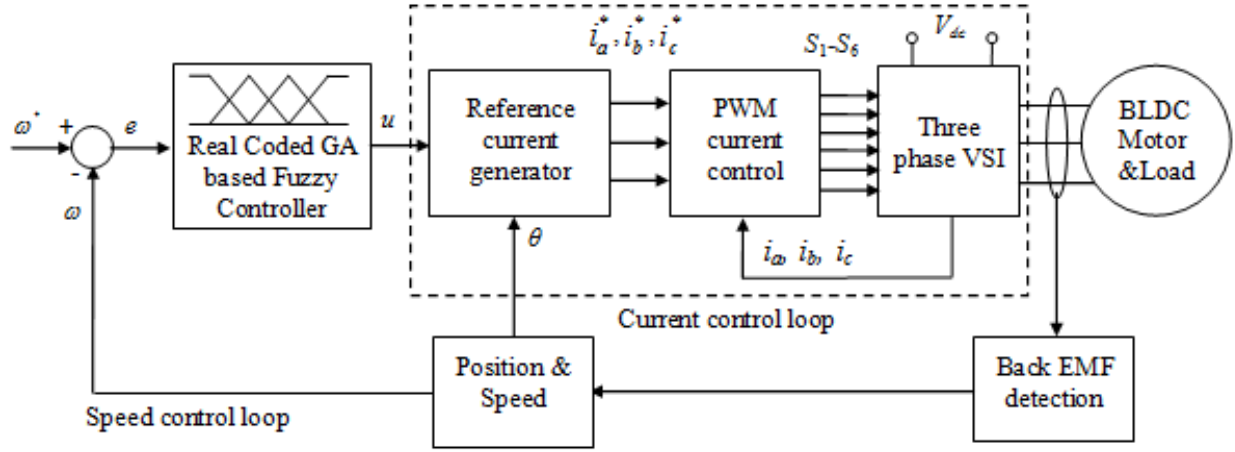

Figure 6. Block diagram of sensorless control of the BLDC motor drive system

In speed control loop as shown in the block digram, the reference speed and the actual motor speed is compared and the error signal is obtained. These signals (e_1, e_2) are employed in fuzzy controller and reference current (I^*) is produced for control system. The current control loop regulates the BLDC motor current to the reference current value generated by the speed controller. The current control loop consists of reference current generator, PWM current control unit and a three phase voltage source inverter (VSI). Position of the BLDC motor is obtained by employing zero crossing back emf detection method eliminating position sensor requirement.

3.1. Modeling of the BLDC motor

Figure 7 describes the basic building blocks of the BLDC motor and inverter that results in a system producing a linear speed-torque characteristic similar to the conventional DC motor. BLDC motor has three phase windings on the stator similar to three phase squirrel cage induction motor and magnets are placed on the rotor to provide air gap flux resulting in brushless rotor construction. When the motor is operated at a certain speed, trapezoidal emfs are induced in stator phase windings. The quasi-square wave AC current is fed to stator phase windings through electronic commutator using current controlled voltage source inverter and rotor position sensor resulting in constant torque development by the motor.

At any instant, two out of three phase stator windings of the motor carry currents synchronized with developed electromagnetic torque as shown in Figure 8. Active switching states for three phase inverter operation, three phase back-emf waveforms and torques of all phases are illustrated in Figure 8. Here, three phase PWM inverter operation can be divided into six modes according to the current conduction states.
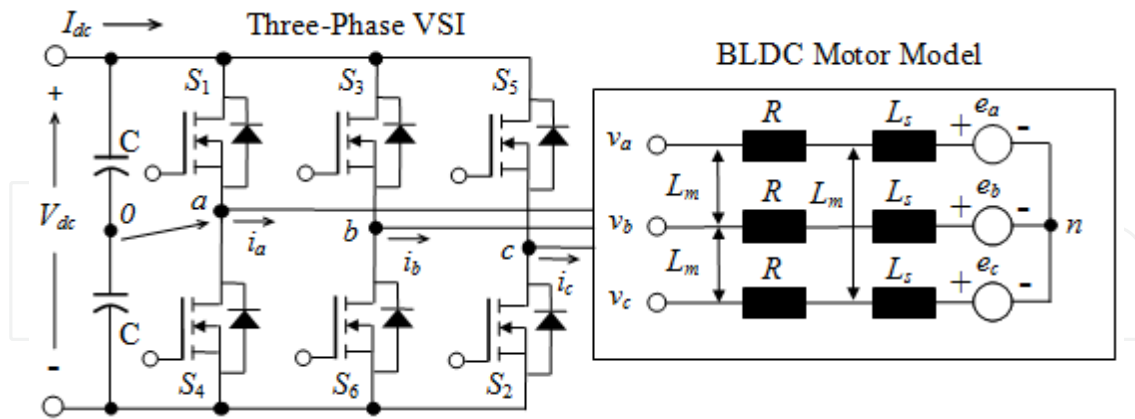
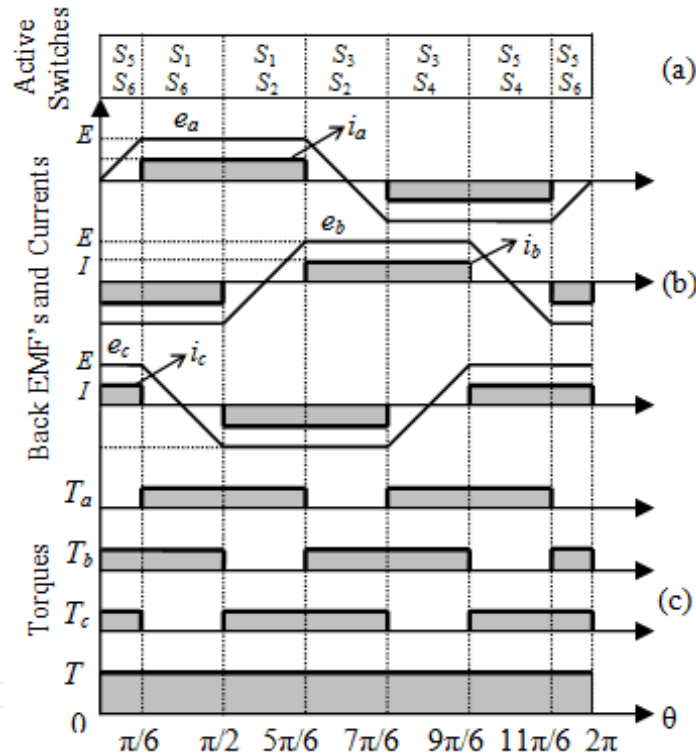**Figure 7.** Configuration of the BLDC motor and inverter system

Figure 8. (a) Active switches, (b) Back emf and phase current waveforms, and (c) Three phase torques of the BLDC motor drive system

Analysis of the BLDC motor is based on the following assumptions for simplicity and accuracy [3,10,15,16].

1. The BLDC motor is not saturated.
2. Stator resistances of all the windings are equal, self and mutual inductances are constant.
3. Semiconductor devices in the inverter are ideal.
4. Iron losses are negligible.

Back-emf waveforms of all phases are equal under above assumptions. A BLDC motor model can be represented as;

$$\frac{d}{dt}\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} = \begin{bmatrix} 1/L & 0 & 0 \\ 0 & 1/L & 0 \\ 0 & 0 & 1/L \end{bmatrix}\begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} - \begin{bmatrix} R/L & 0 & 0 \\ 0 & R/L & 0 \\ 0 & 0 & R/L \end{bmatrix}\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} - \begin{bmatrix} 1/L & 0 & 0 \\ 0 & 1/L & 0 \\ 0 & 0 & 1/L \end{bmatrix}\begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix}, \quad (L = L_s - L_m)$$

(8)

where v_a, v_b, and v_c are the phase voltages, i_a, i_b, and i_c are the phase currents, R and L_s are the stator resistance and inductance, L_m is the mutual inductance, e_a, e_b, and e_c are the trapezoidal back-emfs.

The motion equation is expressed as;

$$\frac{d}{dt}\omega = \frac{1}{J}(T_e - T_\ell - B\omega)$$

(9)

$$\frac{d}{dt}\theta = \omega$$

(10)

where T_ℓ is the load torque in Nm, J is the moment of inertia in kgm², B is the frictional coefficient in Nms/rad, and ω is the rotor speed in electrical rad/sec. The output torque is redefined by Eq.(11) using back-emfs.

$$T_e = (e_a i_a + e_b i_b + e_c i_c) / \omega$$

(11)

This torque expression causes a computational difficulty at zero speed as the induced emf is zero. In this study, the trapezoidal back-emf waveforms are modeled as a function of rotor position to be able to estimate position actively according to the operation speed.

3.2. Modeling of trapezoidal back-emf

The back-emfs can be expressed as a function of rotor position (θ) [17];

$$\begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} = E \begin{bmatrix} f_a(\theta) \\ f_b(\theta) \\ f_c(\theta) \end{bmatrix}, \quad (E = k_e\omega)$$

(12)

where k_e is back-emf constant, $f_a(\theta)$, $f_b(\theta)$, and $f_c(\theta)$ are the functions of rotor position as shown in Figure 8. In this study, $f_a(\theta)$ trapezoidal function with limit values between +1 and -1 expressed by Eq.(13) has been employed.

$$f_a(\theta) = \begin{bmatrix} (6/\pi)\theta & (0 < \theta \leq \pi/6) \\ 1 & (\pi/6 < \theta \leq 5\pi/6) \\ -(6/\pi)\theta + 6 & (5\pi/6 < \theta \leq 7\pi/6) \\ -1 & (7\pi/6 < \theta \leq 11\pi/6) \\ (6/\pi)\theta - 12 & (11\pi/6 < \theta \leq 2\pi) \end{bmatrix} \quad (13)$$

$f_b(\theta)$ and $f_c(\theta)$ can be determined in a similar way considering Figure 8. Substituting equations (12)-(13) into equation (11), the output torque expression becomes,

$$T_e = k_e \left(f_a(\theta) i_a + f_b(\theta) i_b + f_c(\theta) i_c \right) \quad (14)$$

3.3. Advanced simulation model of BLDC motor by using Runge-Kutta numerical integration method

Runge-Kutta method is a frequently used method for solving differential equations numerically. In engineering solutions, fourth order Runge-Kutta method is the most widely used one [18]. For BLDC motor simulations, i_a, i_b, i_c, ω and θ parameters, which have been given by Eqs. (8), (9) and (10) are calculated by using fourth order Runge-Kutta method. For example, the current associated with the a-phase (i_a) is calculated using Eq. (15). Other parameters are calculated in the similar way.

$$k_1 = \frac{1}{L}(v_a - R\,i_a(kT) - e_a)$$
$$k_2 = \frac{1}{L}(v_a - R\,(i_a(kT) + \frac{k_1}{2}) - e_a)$$
$$k_3 = \frac{1}{L}(v_a - R\,(i_a(kT) + \frac{k_2}{2}) - e_a) \quad (15)$$
$$k_4 = \frac{1}{L}(v_a - R\,(i_a(kT) + k_3) - e_a)$$
$$i_a((k+1)T) = i_a(kT) + \frac{T}{6}(k_1 + 2\,k_2 + 2\,k_3 + k_4)$$

where k is the sample and T is the sampling period.

3.4. Reference current generator

Reference current generator determines reference phase currents (i_a^*, i_b^*, i_c^*) of the motor in regard to reference current amplitude (I^*), which is calculated using rotor position (θ). Reference current amplitude (I^*) can be obtained from Eq.(16).

$$I^* = u / k_t \quad (16)$$

where u is the control signal and k_t is the torque constant. Phase currents given in Table 2 can be attained from Figure 8. These currents are input to PWM current control block [19].

Rotor position (θ-Degree)	Reference currents (A)		
	i_a^*	i_b^*	i_c^*
0-30	0	$-I^*$	I^*
30-90	I^*	$-I^*$	0
90-150	I^*	0	$-I^*$
150-210	0	I^*	$-I^*$
210-270	$-I^*$	I^*	0
270-330	$-I^*$	0	I^*
330-360	0	$-I^*$	I^*

Table 2. Reference currents of the BLDC motor

3.5. Current control block

In PWM current control block, reference phase currents (i_a^*, i_b^*, i_c^*) acquired from reference current generator is compared with actual phase currents of the motor (i_a, i_b, i_c). These current error values (e_{ia}, e_{ib}, e_{ic}) obtained using Eq. (17) are applied to inverter hysteresis band ($\pm h_b$) and in regard to the switching states shown in Figure 8 (a), switching signals of three-phase PWM inverter are generated [15,16].

$$
\begin{aligned}
e_{ia} &= i_a^* - i_a \\
e_{ib} &= i_b^* - i_b \\
e_{ic} &= i_c^* - i_c
\end{aligned}
\tag{17}
$$

Inverter phase voltages (v_{ao}, v_{bo}, v_{co}) in reference to midpoint of DC supply voltage (V_{dc}) are obtained using Eq. (18) by using current error values.

$$
\begin{aligned}
v_{ao} &= \begin{bmatrix} V_{dc}/2 & e_{ia} \geq h_b \\ -V_{dc}/2 & e_{ia} \leq h_b \end{bmatrix} \\
v_{bo} &= \begin{bmatrix} V_{dc}/2 & e_{ib} \geq h_b \\ -V_{dc}/2 & e_{ib} \leq h_b \end{bmatrix} \\
v_{co} &= \begin{bmatrix} V_{dc}/2 & e_{ic} \geq h_b \\ -V_{dc}/2 & e_{ic} \leq h_b \end{bmatrix}
\end{aligned}
\tag{18}
$$

BLDC motor phase voltages (v_a, v_b, v_c) are given in Eq. (19) related to inverter phase voltages determined from Eq. (18).

$$
\begin{aligned}
v_a &= \frac{1}{3}(2v_{ao} - v_{bo} - v_{co}) \\
v_b &= \frac{1}{3}(2v_{bo} - v_{ao} - v_{co}) \\
v_c &= \frac{1}{3}(2v_{co} - v_{ao} - v_{bo})
\end{aligned}
\tag{19}
$$

4. Simulation results

An algorithm has been developed to simulate the proposed real coded GA based fuzzy controller in BLDC motor drive. In all simulations and practical applications, the BLDC motor and inverter having the parameters listed in Table 3 have been used.

BLDC motor type	Ametek 119003-01
Rating (P)	106 watt
Number of Phase (Connection)	3 (Star)
Rated speed	4228 rpm.
Rated current	6.8 A
Stator equivalent resistance (R)	0.348 Ω
Stator equivalent inductance (L)	0.314 mH
Moment of inertia (J)	0.0019 Ncm-s^2
Number of Pole (p)	8
Voltage constant (k_e)	0.0419 V/rad/s
Torque constant (k_t)	4.19 Ncm/A
PWM frequency (f_{PWM})	20 kHz
Inverter DC supply (V_{dc})	24 volt
Inverter hysteresis limits (h_b)	± 0.5
Inverter current limiter (I_{base})	20 A

Table 3. Parameters of the BLDC motor and Inverter

First of all, conventional fuzzy controller is designed based on trial-error method and simulation. The parameters of conventional fuzzy controller are obtained as; n_1=10, n_2=1, n_3=2.2, a_1=0.33, a_2=0.66, b_1=0.33, b_2=0.66, c_1=0.33, c_2=0.66, R_1=NB, R_2=NB, R_3=NM, R_4=NM, R_5=NS, R_6=NS, R_7=Z, R_8=PS, R_9=PS, R_{10}=PM, R_{11}=PM, R_{12}=PB, R_{13}=PB.

During simulation of sequential approach, normalization parameters, then rule base and the membership functions have been optimized sequentially. In this approach, the chosen values for mutation probability is 0.05, crossover probability is 0.8, population number is 20, the reference speed for the BLDC motor is set at 2000 rpm, and the motor operates in full load. In the second approach which uses simultaneous optimization, the process is implemented by coding all fuzzy controller parameters in GA simultaneously. In this approach, the chosen values for mutation probability is 0.1, crossover probability is 1, population number is 20, the reference speed for the BLDC motor is set at 2000 rpm and the motor operates in full load. The obtained design parameters of optimal fuzzy controllers using sequential and simultaneous real coded GA are given in Table 4.

For a reference speed of 2000 rpm and operating the BLDC motor in full load, the speed responses and error variations of conventional fuzzy control, sequential GA based fuzzy control and simultaneous GA based fuzzy control are shown in Figure 9 (a) and 9 (b), respectively.

Fuzzy Controller	Sequential Real Coded GA	Simultaneous Real Coded GA
Norm. Parameters	n_1=642.348, n_2=29.895, n_3=3.144	n_1=369.102, n_2=8.448, n_3=9.849
Parameters of Membership Func.	a_1=0.8488, b_1=1.1002, c_1=1.1309 a_2=1.6005, b_2=2.1, c_2=1.5	a_1=1.3035, b_1=1.0762, c_1=0.2051 a_2=1.5191, b_2=2.9, c_2=1.62
Rule Base	R_1=NM, R_2=PM, R_3=NB, R_4=NM, R_5=Z, R_6=NM, R_7=Z, R_8=PM, R_9=Z, R_{10}=PM, R_{11}=PB, R_{12}=NM, R_{13}=PM	R_1=NB, R_2=PM, R_3=NS, R_4=NB, R_5=NB, R_6=NM, R_7=NS, R_8=PM, R_9=PB, R_{10}=PB, R_{11}=PS, R_{12}=NM, R_{13}=PB
Performance Index	J_{in}= 3.2349 (For 150 generations)	J_{in}= 3.2327 (For 850 generations)

Table 4. Optimal design parameters of Real Coded GA based fuzzy controllers
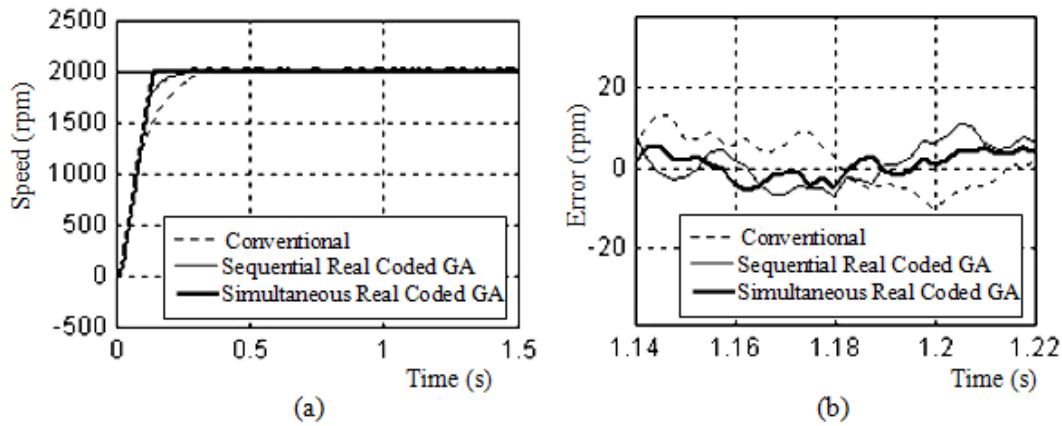
Figure 9. (a) Speed responses and, (b) Errors of conventional and GA based fuzzy controllers

The convergence of performance index (J_{in}) for sequential and simultaneous real coded GA based fuzzy control system are shown in Figure 10 (a) and 10 (b), respectively.
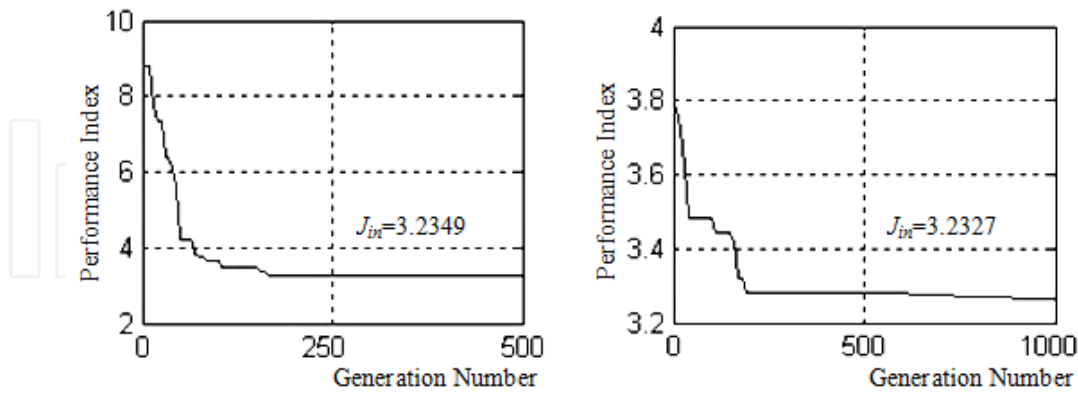

Figure 10. Performance index of the (a) sequential, (b) simultaneous real coded GA based fuzzy control system

In the sequential optimization method, the performance index (J_{in}) of the control system reduces to 3.2349 and this process completes in 150 generations using real coded GA. In the simultaneous optimization method, the performance index of the system (J_{in}) has been reduced to 3.2327. This is the best optimization value obtained in this study. The optimization process

with simultaneous GA completes in 850 generations. Also, the performance index value indicates that simultaneous optimization process gives better results than the sequential optimization process. However, while the overall controller is optimized in 150 generations in sequential optimization method, the optimization is completed in 850 generations in simultaneous approach, in which all parameters are optimized simultaneously.

5. Experimental results

Block diagram of the configuration of DSP based experimental system is shown in Figure 11. The experimental system consist of a brushless DC motor, a voltage source inverter, a current detector for hysteresis current control loop, back-emf detector for sensorless speed control loop, ADSP-21992 Ez-Kit Lite evaluation board, interface devices between ADSP and driver board, a PC and VisualDSP++ software for emulating and programming the DSP.
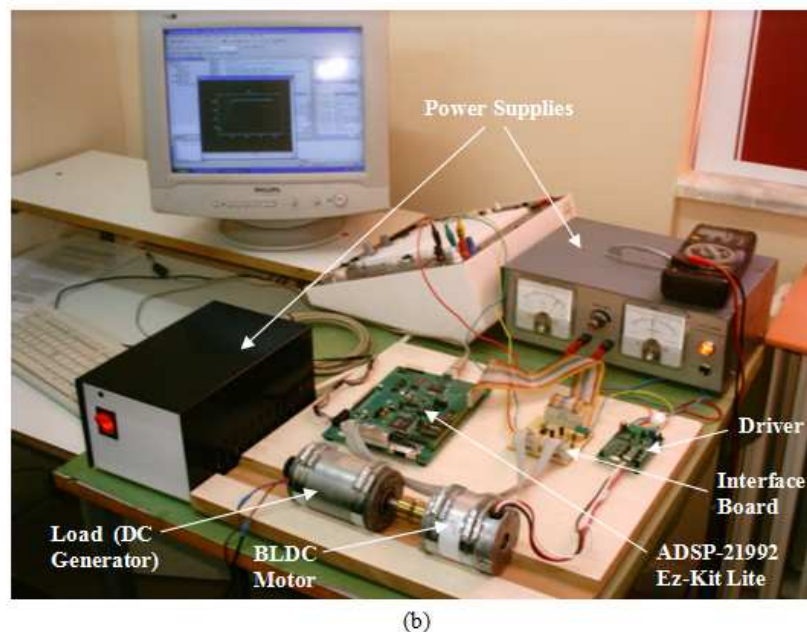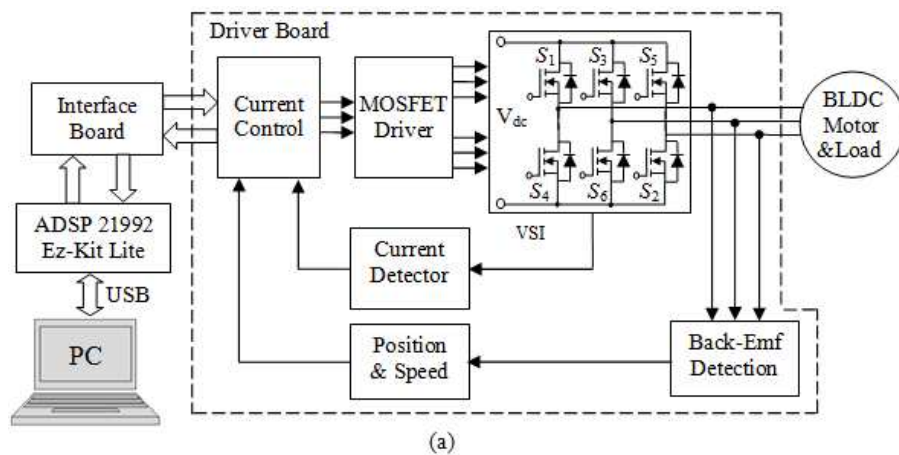

Figure 11. (a) Block diagram, (b) an overview picture of experimental setup of BLDC motor control systems

In the experimental study, sensorless control of the BLDC motor was implemented successfully using conventional fuzzy controller, sequential and simultaneous real coded GA based fuzzy controller, which have parameters obtained by simulation and given in Table 4. Optimal fuzzy controller was implemented in high level C programming language; the program was compiled by VisualDSP++ C compiler and downloaded to the ADSP-21992 DSP controller board. The block diagram of the configuration of closed loop speed control of BLDC motor executed without position and speed sensors is illustrated in Fig 11(a). The experimental setup implemented in the laboratory is shown in Fig 11(b). The reference speed for BLDC motor is set at 2000 rpm, and the motor operates in full load similar with the simulations.

In Figure 12, phase to phase voltage and phase current waveforms of BLDC motor are shown. The responses of both conventional and sequential real coded GA based fuzzy controller are shown in Figure 13, conventional fuzzy and simultaneous real coded GA based fuzzy controller are shown in Figure 14, sequential real coded GA based fuzzy and simultaneous real coded GA based fuzzy controller are shown in Figure 15, respectively.

The experimental results illustrated in these figures prove that the BLDC motor control system was implemented successfully and it operates stably. Also, it is indicated that simulation results show agreement with experimental results.
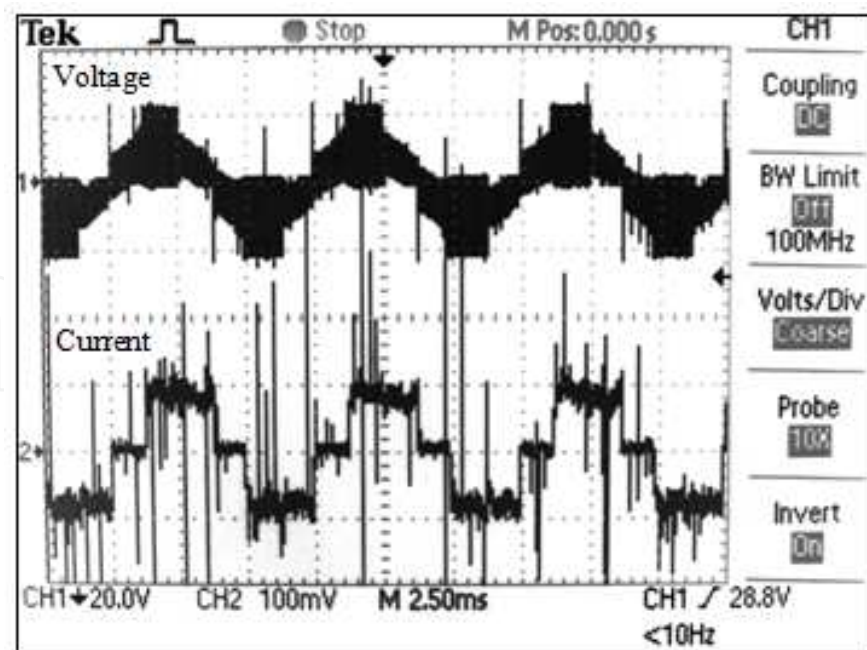

Figure 12. Phase to phase voltage and phase current of the BLDC motor. Time/Div:2.50msec, Volt/Div:20.0V(CH1), Amper/Div:2.00A(CH2)
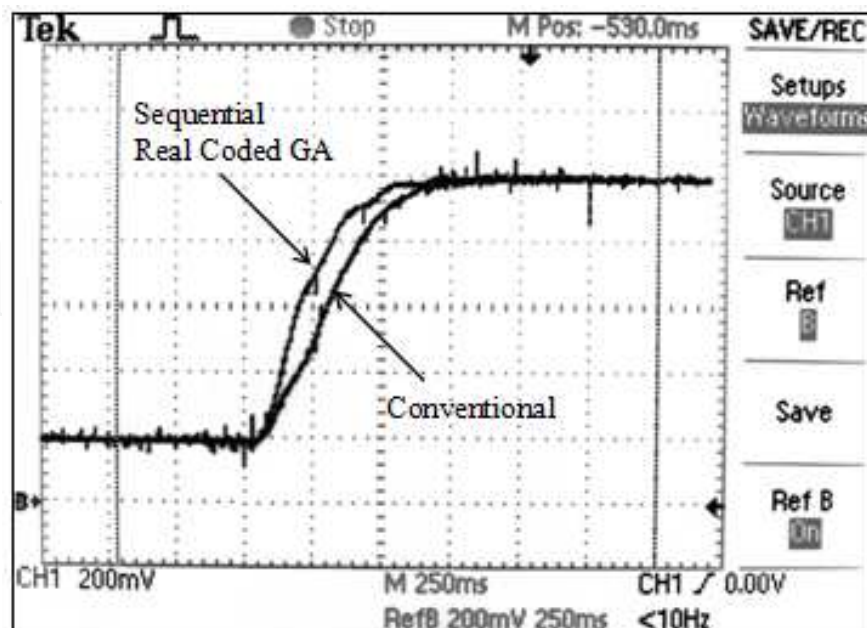
Figure 13. Speed responses of sequential and conventional fuzzy control of the BLDC motor.
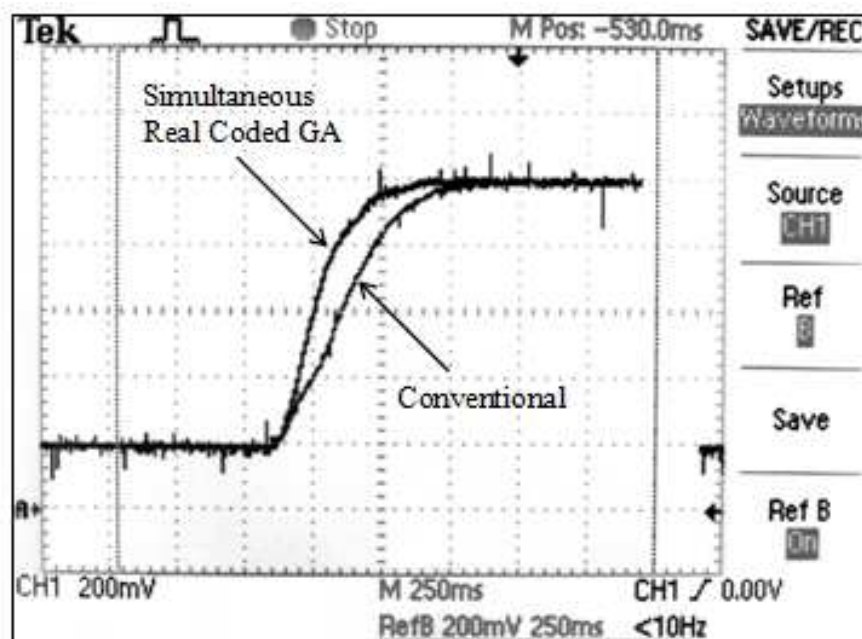Time/Div:250 msec, Volt/Div:400 rpm.

Figure 14. Speed responses of simultaneous and conventional fuzzy control of the BLDC motor.
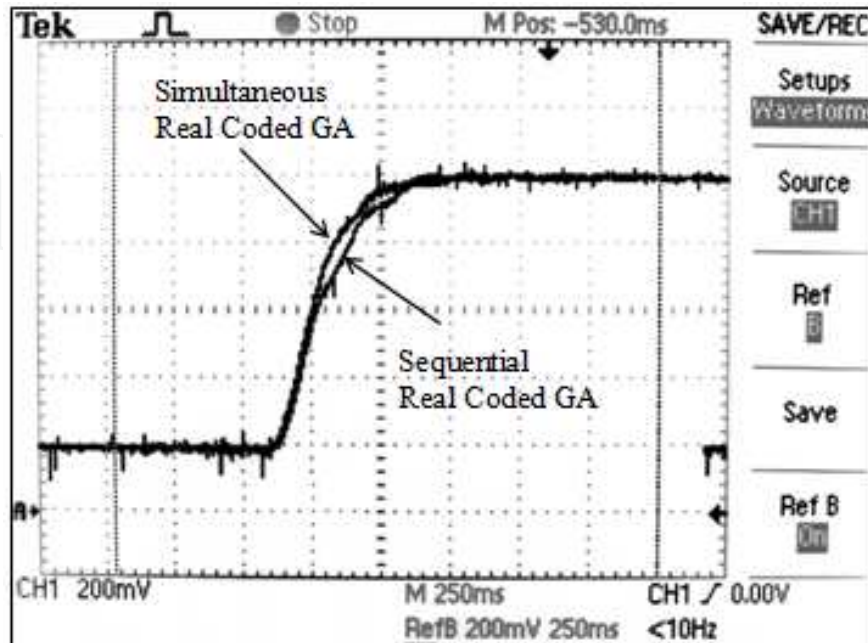Time/Div:2.50 msec, Volt/Div:400 rpm.

Figure 15. Speed responses of sequential and simultaneous Real Coded GA based fuzzy controller. Time/Div:250 msec, Volt/Div:400 rpm.

6. Limitations of the proposed methods

Although real coded genetic algorithm based fuzzy controller has shown a good behaviour for sensorless speed control of a Brushless DC Motor, we think that there are some limitations and disadvantages as follows in proposed method.

- Like other artificial intelligence techniques, the genetic algorithm cannot assure constant optimisation response times. Even more, the difference between the shortest and the longest optimisation response time is much larger than with conventional gradient methods. This unfortunate genetic algorithm property limits the genetic algorithms' use in real time applications.
- Genetic algorithm applications in controls which are performed in real time are limited because of random solutions and convergence, in other words this means that the entire population is improving, but this could not be said for an individual within this population. Therefore, it is unreasonable to use genetic algorithms for on-line controls in real time systems without testing them first on a simulation model.
- For the design of an effective controller, simulation model of the system is necessary to obtain the most accurately. However, even if the best controller have been designed in

simulation environment, fine tuning may be require in order to use the same controller in on-line system.

- Micro genetic algorithm which can produce near optimal solution may be used in on-line system for fine tuning.
- It is not possible to show the stability of the fuzzy controlled system, since the model is not known. If the system model is obtained exactly, the use of tradional gradient descent methods may give better result for optimal control in on-line systems.
- In fuzzy control systems, computing time could be long, because of the complex operations such as fuzzifications and particularly defuzzification. So, the fuzzy controller hardware requires high-speed processors, such as DSP, FPGA.
- The advantage of sensorless BLDC motor control is that the sensing part can be omitted, and thus overall costs can be considerably reduced. In addition, the disadvantages of sensorless control are higher requirements for control algorithms and more complicated electronics.

7. Conclusions

As a result of the study, optimal fuzzy controller has been designed off-line by using real coded GA and the obtained fuzzy controller has been used on-line for DSP-based BLDC motor control system. Modeling of the BLDC motor was performed more accurately to take account of trapezoidal back-emf waveforms and furthermore, fourth order Runge-Kutta numerical integration method was used to decrease the truncation error and numerical instabilities in simulation. Also, the results of our research indicate that an improvement in the transient state and steady state responses of the system has been obtained by means of optimization process using real coded GA. It is clear that, sequential optimization takes less time. Also, observing the system speed response and error curves, it can be deduced that the sequential method gives satisfactory results and that it can be preferred in applications. It was observed that the use of real coded GA makes it possible to adjust system parameters more precisely. Also, the size of chromosomes, in which a great deal of parameters is coded, is reduced. Besides, the proposed method limits the process time to minutes.

Author details

Omer Aydogdu and Ramazan Akkaya
Department of Electrical and Electronics Engineering,
Faculty of Engineering and Architecture, Selcuk University, Konya, Turkey

Acknowledgement

8. References

[1]   F. Cupetino, V. Giordano, D. Naso, B. Turchiano, L. Salvatore (2003) On-Line Genetic Design of Fuzzy Controllers for DC Drives with Variable Load. IEE Electronics Letters, Vol. 39 (5), pp. 479-480.

[2]   G. Acosta, E. Todorovich (2003) Genetic Algorithm and Fuzzy Control: A Practical Synergism for Industrial Applications. Elsevier Science Direct, Computer in Industry, Vol. 52, pp. 183-195.

[3]   O. Aydogdu (2006) Sensorless Control of Brushless DC Machines by means of Genetic Based Fuzzy Controller. PhD. Thesis, Selcuk University, Turkey.

[4]   F. Ashrafzadeh, E.P. Nowicki, M. Mohammadian, J.S. Salmon (1996) An Effective Approach for Optimal Design of Fuzzy Controllers. IEEE Canadian Conference on Electrical and Computer Engineering, pp. 542-545, Alberta, Canada.

[5]   C. Xia, P. Guo, T. Shi, M. Wang (2004) Speed control of brushless dc motor using genetic algorithm based fuzzy controller. Proc. of Int. Conf. on Intelligent Mechatronics and Automation, pp. 460-464, Chengdu, Chine.

[6]   H.P. Stpathy (2003) Real Coded for Parameters Optimization in Short-Term Load Forecasting. Springer-Verlag Berlin Heidelberg IWANN, pp. 417-124.

[7]   N. Hemati, M.C. Leu (1992) A complete model characterization of brushless dc motors. IEEE Transactions on Industry Applications, Vol. 28 (1), pp. 172-180.

[8]   O. Aydogdu, R. Akkaya (2005) DSP Based Fuzzy Control of a Brushless DC Motor Without Position and Speed Sensors. Proceedings of 4th International Advanced Technologies Symposium, pp. 182-187, Konya, Turkey.

[9]   C.K Lee, W.H. Pang (1994) A Brushless DC Motor Speed Control System Using Fuzzy Rules. IEE Power Electronics and Variable Speed Drives, pp. 101-106.

[10] V. Donescu, D.O. Neacsu, G. Griva, F. Profumo (1996) A Systematic Design Method for Fuzzy Controller for Brushless DC Motor Drives. Proc. of the 27th. IEEE Annual Power Electronics Specialists Conference, pp. 689-694, Baveno, İtaly.

[11] M. Çunkaş, R. Akkaya (2002) Compare with Binary and Real Coded Genetic Algorithms. Selcuk University The Journal of Engineering, Vol. 7 (2), pp. 11-17.

[12] C.J. Wu, G.Y. Liu (200) A Genetic Approach for Simultaneous Design of Membership Functions and Fuzzy Control Rules. Kluwer Academic Pub. Journal of Intelligent and Robotic Systems, Vol. 28, pp. 195-211.

[13] N.J. Radcliff (1991) Formal Analysis and Random Respectful Recombination. In Proc. of Fourth International Conference on Genetic Algorithms, San Diego, CA, USA.

[14] R.L. Haupt, S. Haupt (1998) Practical Genetic Algorithms, A Willey-Interscience Publication, USA.

[15] B. Lee, M. Ehsani (2003) Advanced Simulation Model for Brushless DC Motor Drives. Taylor & Francis Inc. Electric Power Component and Systems, Vol. 31, pp. 841-868.

[16] H.A. Toliyat, T. Gopalarathnam (2002) AC Machines Controlled as DC Machines (Brushless DC Machines/Electronics). In: L.S. Timothyl, (eds) The Power Electronic Handbook. CRC Press LLC, New York, pp. 78-100.

[17] B. Sing, K. Jain (2003) Implementation of DSP based digital speed controller for permanent magnet brushless dc motor. IE(I) Journal-EL., Vol. 84, pp. 16-21.

[18] J.R. Rice (1983) Numerical Methods, Software, and Analysis. McGraw-Hill, New York.

[19] T. Kim (2003) Sensorless Control of the BLDC Motors from Near-Zero to Full Speed. PhD. Thesis, Texas A&M University, Texas, USA.