A Neural Network-PSO Based Control for Brushless DC Motors for Minimizing Commutation Torque Ripple

M. Aghashabani[1] **J. Milimonfared**[2] **A. Kashefi Kaviani**[3] **M. Ashabani**[4]

1-BSc, Department of Science, Payame-Noor University, Boroojen Branch
maghashabani@yahoo.com

2- Professor, Department of Electrical Engineering, Amirkabir University of Technology, Tehran, Iran
milimonfared@aut.ac.ir

3- MSc, Department of Electrical Engineering, Florida International University, Miami, Florida, USA
alikashefi107@yahoo.com

4-MSc, Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada.
mehdi_ashabani@yahoo.com

Abstract :

This paper presents the method of reducing torque ripple of brushless DC (BLDC) motor. The commutation torque ripple is reduced by control of the DC link voltage during the commutation time. The magnitude of voltage and commutation time is estimated by a neural network and optimized with an optimization method named particle swarm optimization (PSO) algorithm analysis. The goal of optimization is to minimize the error between the command torque and real torque and doesn't need knowledge of the conduction interval of the three phases. It adaptively adjusts the DC link voltage in commutation duration so that commutation torque ripple is effectively reduced. In this paper, the performance of the proposed brushless DC (BLDC) control is compared with that of conventional BLDC drives without input voltage control.

Keywords: BLDC machines, Commutation, Optimized input voltage, Torque ripple

Submission date: Jan. 12, 2008.

Acceptance data: Dec. 15, 2010

Corresponding author: Seyed Mahdi Ashabani

Corresponding author's address: 2nd floor, Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada,T6G 2V4.

1. Introduction

The permanent magnet brushless DC (BLDC) machine is extensively used in a wide range of applications such as electric vehicle, robotics, aerospace, computer and household products, because of its high power to current ratio and high efficiency. In spite of their advantages, they have some drawbacks like stator current commutation torque ripples, which occurs due to the loss of ideal phase current commutation. Fig. 1 shows the ideal current and back-EMF of a 120° elec. conduction mode in a 3-phase brushless DC machine. In the ideal case, the stator currents instantaneously reach to their final values. However, in a practical BLDC drive, since machine has both inductance and resistance, the stator currents are different from the ideal case and the currents reach to their final values with delay. In other words, circuit has a time constant. Therefore, due to machine inductance, the current ripple is generated. With a trapezoidal back-EMF waveform, the current ripple leads to the torque ripple. Several theoretical approaches for analysis of the commutation torque ripple have been reported in literature [1-2].

Also, several methods have been introduced for mitigating the commutation torque ripple of brushless DC machines. A direct torque control (DTC) by employing a hybrid 2-phase and 3-phase switching mode during the commutation periods is presented in [3]. A commutation torque ripple reduction approach is reported in [4], which is based on this fact that current

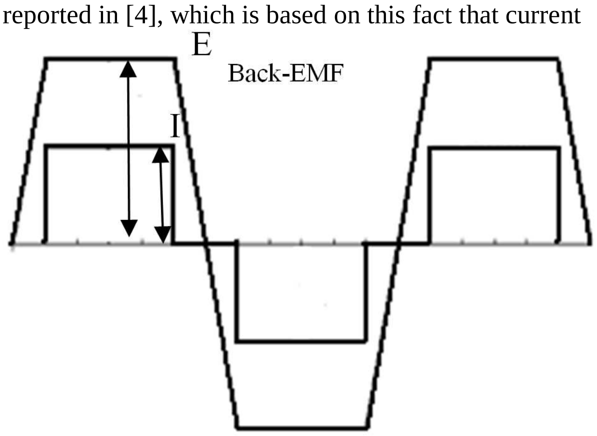


Fig.1. Ideal current and back-EMF waveforms of a single phase in a BLDC motor

slopes of the incoming and outgoing phases during the commutation interval can be equalized by a proper duty-ratio control. An adaptive torque ripple control for current shaping during commutation is proposed in [5]. A current control method based on Fourier series coefficients and space vector PWM is used to reduce the torque ripple in a BLDC motor [6].

The previous studies have relatively complicated modeling and are not very effective. Almost all the previous methods are based on current shaping. These current shaping techniques have complex behavior during commutation interval because inverter must produce difficult current waveforms. On the other hand, because of the machine large electrical time constant fulfillment of these current waveforms during commutation duration is difficult [5]. Another factor that reduces the effectiveness of the previous methods is the method commutation duration method. As mentioned earlier, the current shaping is only applied when phase currents are commutating so it is essential to know commutation duration. In the previous works, this duration was calculated with this assumption that machine works in normal condition, i.e. without applying commutation torque ripple techniques. This may lead to non causal systems since the torque ripple methods themselves affect commutation interval. We present a simple method which uses variable input DC link voltage in order to minimize commutation torque ripple. We use an optimum increased input voltage during commutation interval to keep the stator currents near to ideal case. This method can be implemented in practice by a DC chopper or by control of firing angles of rectifier switches. If a controlled rectifier is used, since rectifier switches must tolerate more voltage stress during commutation to increase input voltage so their voltage rating is higher than the conventional drives. If boost converter is used, an extra switch is required that increases the total system cost with respect to the conventional BLD drives. Online neural network based parameter tuning can be implemented in a DSP or microcontroller so it doesn't increase the system cost. It was first presented in [7], where the authors used an analytical approach to determine the input voltage during commutation period. They implemented the drive practically and they could reduce the torque ripple from 28%, with a fixed input voltage, to 17.9%, by applying variable input voltage during commutation period. As discussed earlier, this method is not causal because commutation interval itself depends on the input voltage and so when input voltage is increased it varies. In our work duration of application of the optimized voltage is adjusted adaptively by a neural network and optimized by PSO. Therefore, the knowledge of circuit behavior and commutation duration is not necessary. We use the evolutionary computation to obtain the optimum values of increased input voltage and the duration that the voltage should be applied at different speeds and load torques. With our proposed method, the commutation torque ripple is reduced to less than 8% which is remarkable reduction. The commutation torque ripple in the previous works [3-6] is about 15-20%.

2. Analysis of Commutation Torque Ripple

In this section we study the current ripple which is a result of commutation events and leads to torque ripple. Fig. 2 shows the current paths during commutation between phases b and c. It is assumed that the back-EMF is constant during phase current. However, it is clear that this assumption is not valid. Following equations describe performance of the circuit during the commutation [8]:

$$\frac{di_a}{dt} = \frac{1}{3L_s}U_{dc} - \frac{4}{3L_s}E \quad (1)$$

$$\frac{di_b}{dt} = \frac{1}{3L_s}U_{dc} + \frac{4}{3L_s}E \quad (2)$$

$$\frac{di_c}{dt} = -\frac{2}{3L_s}U_{dc} + \frac{2}{3L_s}E \quad (3)$$

where U_{dc} is the DC-link voltage.

During the commutation, current of phase b rises and current of phase c falls. Depending on the rate of change of currents in phases b and c, three cases may occur

(a) If $U_{dc} = 4E$, then:

$$\frac{1}{3L_s}U_{dc} + \frac{2}{3L_s}E = \frac{2}{3L_s}U_{dc} - \frac{2}{3L_s}E \quad (4)$$

Therefore, from (1-4), we have

$\frac{di_b}{dt} = -\frac{di_c}{dt}$ and $\frac{di_a}{dt} = 0$, i.e. i_a is constant.

(b) If $U_{dc} < 4E$, then:

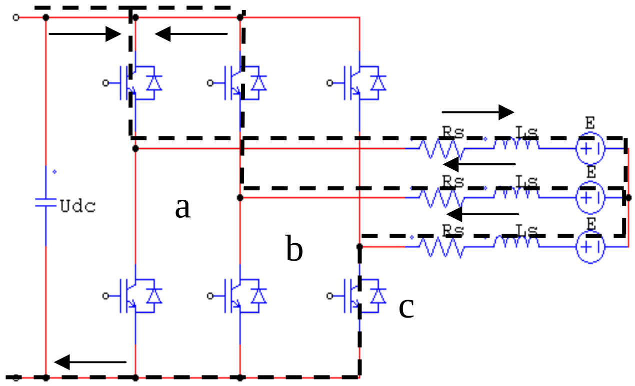


Fig. 2. Current flow during commutation

$$\frac{1}{3L_s}U_{dc} + \frac{2}{3L_s}E > \frac{2}{3L_s}U_{dc} - \frac{2}{3L_s}E \quad (5)$$

Therefore, $\frac{di_b}{dt} > -\frac{di_c}{dt}$ and $\frac{di_a}{dt} < 0$, i.e. the current of phase ‘a’ decreases.

(c) If $U_{dc} > 4E$, then:

$$\frac{1}{3L_s}U_{dc} + \frac{2}{3L_s}E < \frac{2}{3L_s}U_{dc} - \frac{2}{3L_s}E \quad (6)$$

Therefore, $\frac{di_b}{dt} < -\frac{di_c}{dt}$ and $\frac{di_a}{dt} > 0$, i.e. the current of phase ‘a’ increases.

From commutation torque ripple perspective, DC link voltage should be equal to four times of the back-EMF amplitude during commutation interval to eliminate the current ripple and, consequently, commutation torque ripple. As mentioned above, assumption of constant back-EMF, during commutation interval, is not valid and this may lead to a large error.

Therefore, this analysis is based on approximation and can not precisely describe the commutation period. In this regard, we propose an evolutionary computing based method to estimate the optimum DC link voltage and its optimum. This method, does not have aforementioned limitations, finds the optimum values with trial and error, and does not require the knowledge of commutation interval. The next section explains the basis of this technique.

3. Proposed Technique

The major goal of this paper is to find the optimum DC link voltage during commutation intervals and the best duration which this voltage should be applied, at different speeds and load torques, to achieve a high-performance drive with minimum torque ripple. As it can be inferred from (4-6), by regulating U_{dc} in commutation period, phase currents can be controlled and by choosing appropriate values for applied voltage and also duration of applying this voltage, phase currents can be appropriately shaped and, consequently, commutation torque ripple can be effectively reduced. If the current variations of two phases, that commutation occurs between them, have equal slopes with opposite signs, the current of 3rd phase will not be affected. In other words, by adjusting input voltage during commutation time, phase currents can be regulated in a good manner so that the torque ripple reaches its lowest value. Similar approach with varying input voltage was implemented in [7], but as discussed before, because of weakness and complexity in modeling due to variation of back-EMF in commutation period, analysis model has some limitations.

An effective approach for voltage control during commutation period is to employ an optimization algorithm to obtain the best values of input voltage and its duration in commutation time. In the conventional control, the input dc link voltage is tuned by a PI controller and this controller shows the same behavior in the commutation and non commutation time. In the proposed method, in duration of ΔT^* after starting time of commutation, the output of PI controller is multiplied by a gain of K_1. If the values of ΔT^* and K_1 are optimized, the commutation torque ripple reaches to its weakest value.

The optimal values of K_1 and ΔT^* depend on the speed reference and load torque. Thus, the best values of these parameters should be obtained at different speeds and torques. These values are used to train an artificial neural network offline. This trained neural network is employed in voltage control loop to determine the best values of ΔT^* and K_1 at different speeds.

For optimization, a powerful intelligent algorithm, named particle swarm optimization (PSO) is used. In order to achieve minimum commutation torque ripple in steady state, the fitness function is defined as the difference between maximum and minimum torques in steady state. Following section describes this method.

3.1. Particle swarm optimization method

PSO is a population-based algorithm that exploits a population of individuals to probe promising region of the search space. In this context, the population is called swarm and the individuals are called particles. Each particle moves with an adaptable velocity within the search space and retains in its memory the best position it ever encountered. The global variant of PSO, the best position ever attained by all individuals of the swarm is communicated to all the particles [9 - 10].

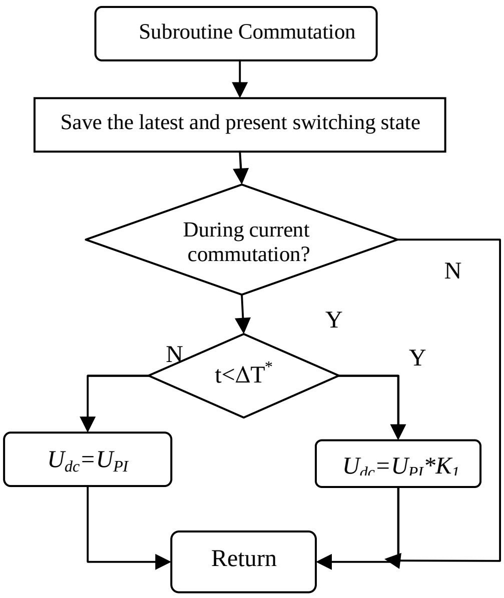


Fig.3. Flowchart of torque ripple control during current commutation

The general principles for the PSO algorithm are stated as follows:

Suppose that the search space is n-dimensional, then the i^{th} particle can be represented by a n-dimensional vector, $X_i=[x_{i1},x_{i2},\ldots,x_{in}]^T$, and velocity $V_i=[v_{i1},v_{i2},\ldots,v_{in}]^T$, where i=1,2,…,$N$ and N is the size of the population.

In PSO, particle i remembers the best position it visited so far, referred to as $P_i=[p_{i1},p_{i2},\ldots,p_{in}]^T$, and the best position of the best particle in the swarm is referred as $G=[g1,g2,\ldots,gn]^T$. Each particle 'i' adjusts its position in next iteration $t+1$ with respect to (7-8) [9]:

$$V_i(t+1)=\omega(t)V_i(t)+c_1r_1\left(P_i(t)-X_i(t)\right)+c_2r_2\left(G(t)-X(t)\right) \tag{7}$$

$$X_i(t+1)=X_i(t)+\chi V_i(t+1) \tag{8}$$

Where $\omega(t)$ is inertia coefficient which is employed to manipulate the impact of the previous history of velocities on the current velocity. χ is constriction factor which is used to limit velocity, here χ=0.7. c_1 and c_2 denote the cognitive and social parameters and r_1 and r_2 are random real numbers drawn from uniformly distributed interval[0, 1]. $\omega(t)$ resolves the trade off between the global and local exploration ability of the swarm. A large inertia coefficient encourages global exploration while small one promotes local exploration. Experimental results suggest that it is preferable to initialize it to a large value (here 1), giving priority to global exploration of search space, and gradually decreasing to a small value about zero (here 0.1) as to obtain refined solution

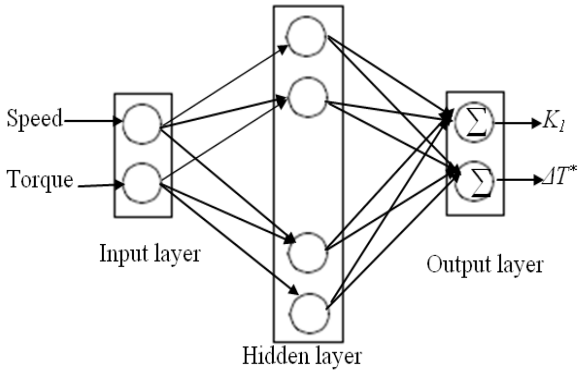


Fig.4. Neural network of RBFN

c_1 and c_2 accelerate the search toward local and global best directions respectively. Experiments indicate to initialize c_1 to 2.5 and decrease it monotonically to 1.5 during optimization procedure. On the other hand, it is better to track c_2 on an inverse trajectory.

In this study, number of population is set to 10 times of number of variables, i.e. 20, and for preventing explosion of swarm, maximum allowable velocity along each dimension is set to 0.5 of its feasible range. Results show that, in this application, algorithm converges within 60 to 70 iterations. Hence, in a conservative manner, the number of iterations is set to 100.

3.2. Neural Network of RBFN

Like most feed forward networks, RBFN has three layers, namely, an input layer, a hidden layer, and an output layer. A schematic view of the specific RBFN with 2 inputs and two outputs is given in Fig. 4.

Using the results of previous section, an RBFN which can map the input variables to the outputs with a nonlinear relationship is trained. The input variables are speed and load torque, and the outputs of RBFN are the voltage gain of K_1 and applying time of ΔT^*. The trained RBFN is used for on-line tuning of optimum K_1 and ΔT^*.

4. Simulation Results

The parameters of 500 W BLDC machine are given in Table I. The 500 W BLDC machine is simulated in the MATLAB SIMULINK environment. The input AC voltage is 220 V and rectified DC link voltage is 298V. Several tests are carried out to evaluate the proposed BLDC drive system in simulation.

The stator current and torque response are obtained at different speeds and load torques. Fig. 5 shows the variations of the optimum values of K_1 for different values of speed and load torque, obtained from optimization process and prediction of the neural network, trained with optimum values obtained from optimization process. The optimum values are obtained when speed varies from

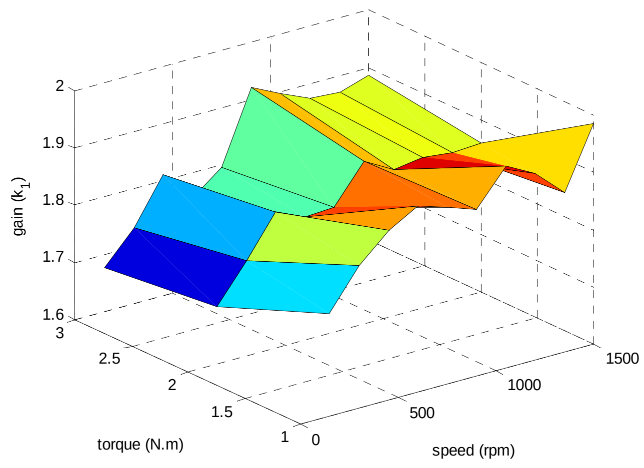
Table.1. Parameters of BLDC motor

Number of poles	8
Rated speed (rpm)	1500
Rated torque (N.m)	3
PM excitation (Wb)	0.085
Phase resistance (Ω)	1.875
Self inductance (mH)	8.5
Ac input voltage (v)	220

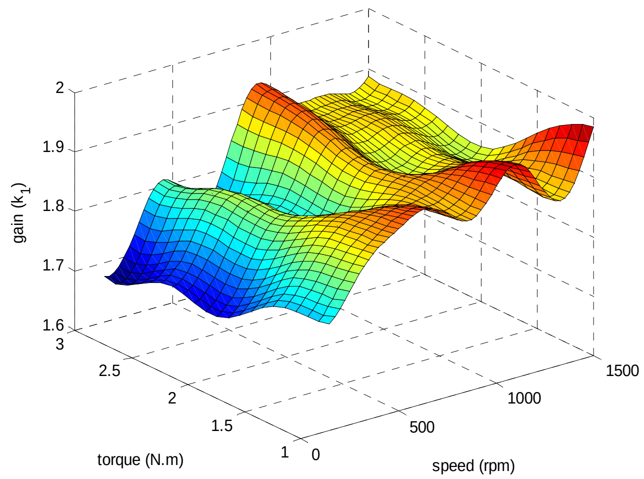
150 to 1500 rpm with steps of 150 rpm, and in each speed, torque varies from 1 to 3 N.m with step of 1 N.m. It can be inferred from this figure that ΔT^* has an incremental characteristic with respect to speed and torque. Fig. 6 illustrates the optimum values of ΔT^* and compares it with neural network prediction. This neural network interpolates other optimum values of K_1 and ΔT^* which are not obtained from optimization.

In this case, at first an external circuit detects the beginning of commutation which occur every 60º elec. After detection, the output of PI controller is multiplied by a gain of K_1 in duration of ΔT^*. The parameters K_1 and ΔT^* is on-line provided by an artificial neural network, depending on the reference speed and load torque. As it will be seen, since the input voltage is changed to optimum value during commutation, the resultant torque ripple is considerably reduced. Phase current, and electromagnetic torque waveforms of the simulated machine with and without proposed technique, when speed is 1500 rpm and load torque is equal to 3 N.m (Rated speed and torque), are depicted in Fig. 7. Fig. 8 shows similar parameters when command speed and load torque are 300 rpm and 1 N.m, respectively.

It is seen that for speed of 1500 rpm torque ripple is reduced from 1.2 N. to 0.1 N. which shows a significant reduction of about 85%. Also, at low speed (300 rpm) and low torque (1 N.m), torque ripple is reduced from 0.32 N.m to 0.04 N.m.

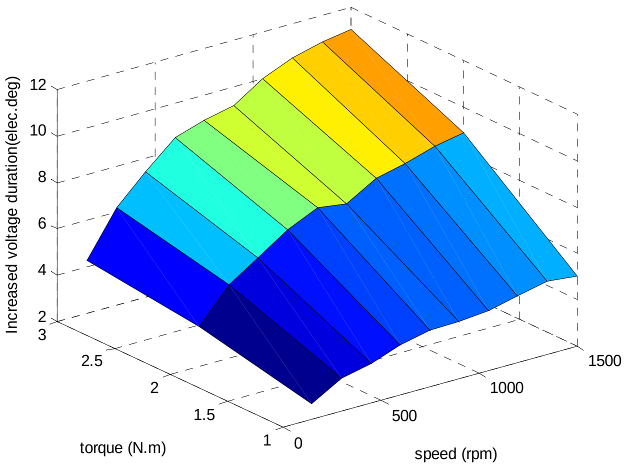


(a)

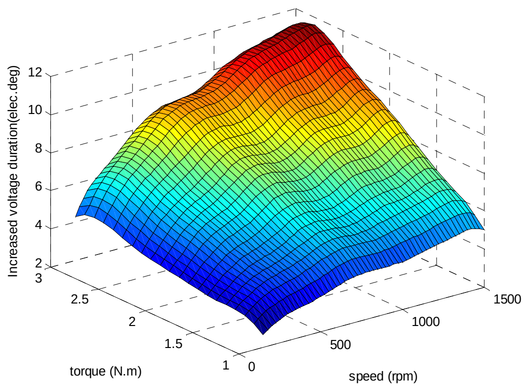


(b)

Fig. 5. Variation of K_1 (a) results of optimization (b) neural network prediction

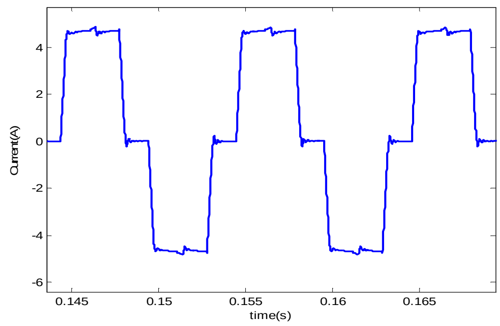


(a)

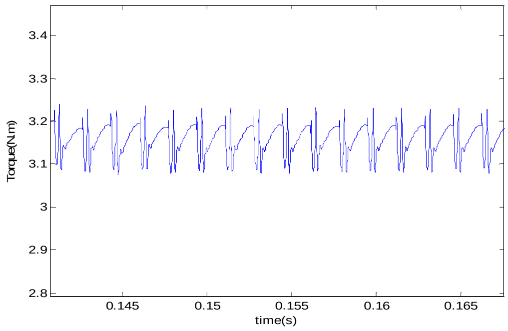


(b)

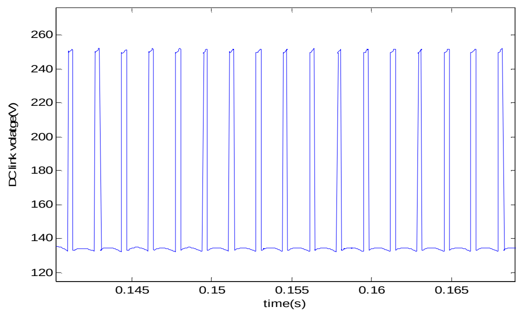
Fig. 6. Variation of ΔT^* (elec.deg) (a) results of optimization (b) neural network prediction



(a)

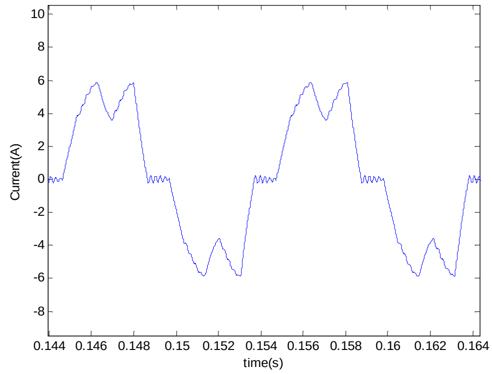


(b)

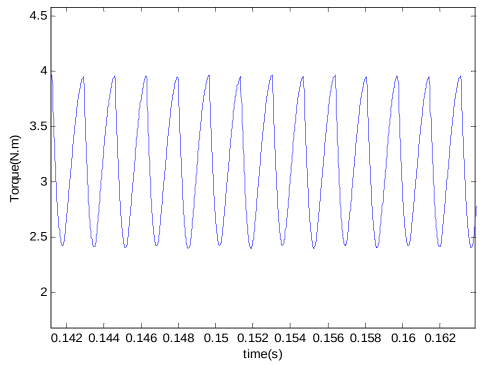


(c)

Fig .7. Simulation results of the proposed method at high speed (1500 rpm) and high torque (3 N.m) (a) phase current (b) electromagnetic torque (c) command voltage

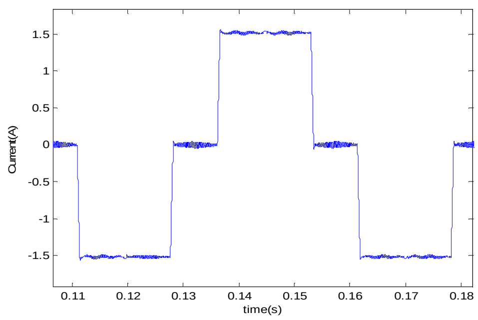


(a)

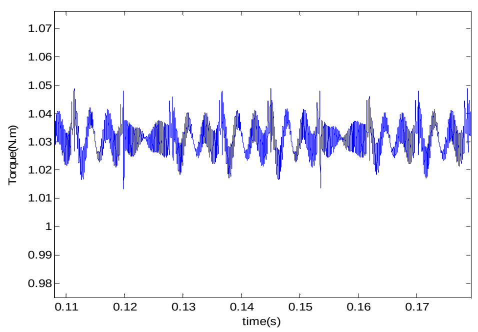


(b)

Fig. 8. Simulated performance of the conventional control at high speed (1500rpm) and high torque (3N.m)

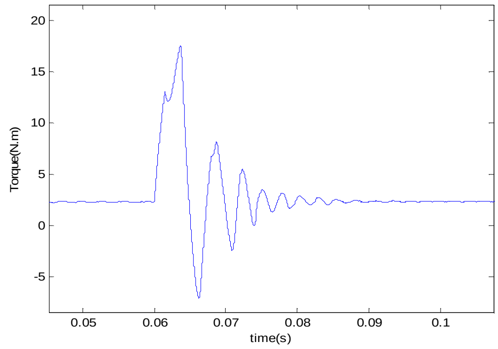


(a)

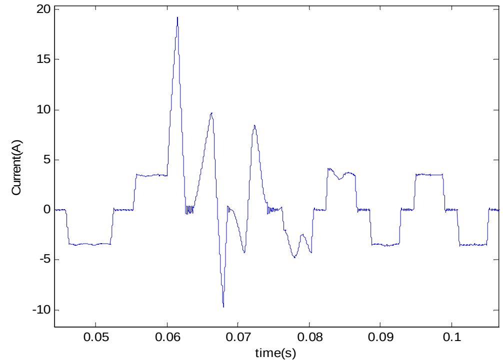


(b)

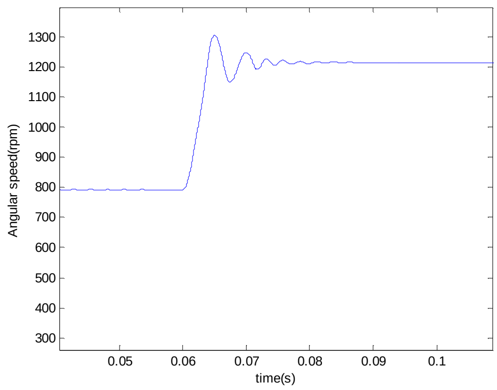
Fig. 9. Simulated performance with proposed control at low speed and low torque (300 rpm and 1 N.m)



(a)



(b)



(c)

Fig. 10. Simulated performance when speed changes from 780 rpm to 1220 rpm. (a) Electromagnetic torque (b) current and (c) angular speed (rpm)

Fig.10 shows the results of simulation when a disturbance is applied to the system. In this case, command speed varies from 780 rpm to 1220 rpm at t=0.06 (s) and load torque is equal to 1.75 N.m. When command speed changes, neural network senses this deviation and produces the new optimum values of K1 and ΔT^{*}. It is seen that before the disturbance, the electromagnetic torque and current waveforms are near to the ideal case. When speed change occurs, after some cycles machine reaches to its steady state and current and torque again have negligible ripple. This diagram verifies effectiveness of the neural network performance.

5. Conclusion

In this paper a new method for reducing commutation torque ripple with varying input voltage a period of ΔT^* during commutation time is presented and optimized offline by PSO algorithm Results of the optimization are used to train a neural network and this neural network is used for online tuning of K_1 and ΔT^* at different speeds. Results of the simulations validate effectiveness of this method.

Acknowledgment

This paper first was presented in ICEE 2008 conference held in Tarbiat Modarres University and was selected as a technical paper. Also, it has been recognized by the Technical Program Committee as having the potential to be developed and submitted to the journal Iranian association of Electrical and Electronics Engineers.

The authors will to thank Tarbiat Modarres University for its excellent service during holding the conference.

References

[1] R Carlson, M. Lajoie-Mazenc, and J. C. S. Fagundes, “Analysis oftorque ripple due to phase commutation in brushless DC machines,”IEEE Trans. Ind. Applicat., vol. 28, pp. 632–638, May/June 1992.

[2] T. M. Jahns andW. L. Soong, “Pulsating torque minimization techniques for permanent magnet AC motor drives—a review,” IEEE Trans. Ind. Electron., vol. 43, pp. 321–330, Apr. 1996.

[3] Y. Liu, Z. Q. Zhu,and D. Howe, “Commutation-Torque-Ripple Minimization in Direct-Torque-Control PM Brushless DC Drives” IEEE Trans. Ind. Applicat., vol. 43, pp. 1012-1017, July/August 2007.

[4] J. H. Soong, and I. Choy, “Commutation Torque Ripple Minimization in Brushless DC motor using a single DC current sensor” IEEE Trans.Power.Electron, Vol. 19, no.2, pp. 312-319, March 2004.

[5] Y. Sozer and D. A. Torrey, "Adaptive torque ripple control of permanent magnet brushless DC motors," Proc. 13th Annual Applied Power Electronics Conference and Exposition, Vol.1, 15-19 Feb. 1998, pp.86-92.

[6] T. S. Kim, S. C Ahn and D. S. Hyun, "A new current control algorithm for torque ripple reduction of BLDC motors," Proc. of 27th Annual Conference of the IEEE Industrial Electronics Society, Vol. 2, 29th Nov.-2nd Dec. 2001 pp. 1521-1526.

[7] K. Y. Nam, W. T. Lee, C. Lee, J-Pyo Hong, “Reducing torque of DC motor by varying input voltage” IEEE Trans. Magn. Vol 42, pp. 1307-1311, no. 4, Appril 2006.

[8] Y. Liu, Z. Q. Zhu, and D. Howe, “Commutation torque ripple minimization in direct controlled PM brushless DC motors”, IAS 2006 conference, pp. 1642-1648.

[9] E. Konstantinos and N. Vrahatis, “On the Computation on All Global Minimizers Through Particle Swarm optimization”, IEEE Trans. on Evolutionary Computation, vol. 8, No3. 2004.

[۱۰] جعفر سلطانی، حمیدرضا امینی،حسین ابوترابی زارچی، "کنترل مستقیم شتاور و شار موتور سنکرون رلوکتانسی سه فاز تغذیه شده با اینورتر SVM در سطحی سه فاز چهار کلیدی با بکارگیری کنترل کننده لغزشی"مجله انجمن مهندسین برق و الکترونیک ایران، سال سوم، شماره اول، بهار و تابستان ۸۵