Adaptive Control for Torque Ripple Minimization in PM Synchronous Motors

Mahmoud M. Saafan, Amira Y. Haikal, Sabry F. Saraya, Fayez F.G. Areed

Computer and Systems

Engineering Department, Faculty of Engineering

Mansoura University, Egypt.

ABSTRACT

This paper presents two methods for designing special purpose controllers for permanent magnet synchronous motor. The main target of the designed controllers is to reduce torque ripples of this type of motors. The first proposed adaptive method is based on two loop controllers (current controller and speed controller) in addition to using space vector pulse width modulation to maximize fundamental component of torque. The second proposed method is based on PI current controllers enabling tracking of quadrature current command values. Simulation results of the suggested adaptive controller are compared with that of the PI controller. Comparative analysis proves the effectiveness of the suggested adaptive controller than the classical PI one according to ripple reduction as well as dynamic response. Moreover, the suggested adaptive controller when compared with other controllers shows great success in torque ripples reduction, enabling speed tracking while minimizing the torque ripple.

Keywords

Model reference adaptive system, PM synchronous motor, torque control.

1. INTRODUCTION

We ask that authors follow some simple guidelines. In essence, we ask you to make your paper look exactly like this document. The easiest way to do this is simply to download the template, and replace the content with your own material.

Permanent magnet synchronous motor (PMSM) is wildly used from home appliance to electric vehicles since PMSM has high power density and efficiency than reluctance and induction motor. The magnetic and thermal capabilities of the PM were increased by employing the high-coercive PM material, and thus, the PMSM is often employed as an effective actuator in special purpose applications.

The PMSM is essentially a wound rotor AC synchronous machine with no damper windings. The idea resides in the fact that the rotor excitation is fixed and provided by permanent magnets instead of being supplied by an external circuit through slip rings and brushes. Actually PMSMs are distinguishable due to several advantages compared to their counterparts [1]:

- It operates at a higher power factor compared to induction motor (IM) due to the absence of magnetizing current.
- It doesn't require regular brush maintenance like conventional wound rotor synchronous machines due to absence of external rotor excitation.

- Rotor doesn't require any supply nor does it incur any loss due to absence of rotor windings.
- Low noise and vibration than switched reluctance motors (SRM) and IMs.
- Lower rotor inertia leading to rapid response.
- Larger energy density and compact structure.

However, PMSMs suffer from torque ripples. Since PMSMs aim for high performance applications such as machine tools or direct drive robotics, torque oscillations are not acceptable. Torque ripples lead to speed oscillations, which cause deterioration in system performance. In machine tool applications, these oscillations leave visible patterns in high-precision machined surfaces.

In fact, the electromagnetic torque generated by a PMSM are divided into Fundamental, slot and reluctance torque components, in which each can contribute to higher ripples in total torque. Fundamental torque is the desired torque, which is generated by the stator current and rotor flux linkage. While, Slot torque is due to the slotting in the stator. Reluctance torque is due to the rotor's variable reluctance. Reluctance torque can be negligible if the d-axis current is controlled to be nearly zero ($i_d \approx 0$) or the rotor saliency is not significant.

Researchers presented various techniques for torque ripple minimization [2]. These techniques either depends on changing motor design with the aim of reaching ideal characteristics [1], [3]-[4].or on designing special purpose controller to manage the problem of ripples which is the aim of the proposed work. The approach proposed in this work is based on harmonic cancellation using the pre-programmed current waveforms [5], [6]. This approach relies on knowing torque ripple characteristics of the specific motor, and using the torque production model to get the optimal currents that need to be injected to cancel the undesired value of the torque. This approach is sensitive to parameter variations so, its performance degrades when changing the operating conditions. However, on-line estimation of parameter variations during the motor operation can deal with that [6]–[10].

Common control strategy for PMSM is vector control oriented in rotor dq reference frame. Control approaches have two alternatives, either dealing with speed loop or current loop. When choosing torque controller current loop can be replaced with the torque loop due to mathematical relation joining both then use motor torque observers to get the necessary feedback. Different methods have been investigated for torque observation, such as recursive least squares [8] and model

reference adaptive system [9] to find out the complete torque waveform. The resulted waveform is then used as a feedback signal for the torque controller. The accurate current measurement enables accurate estimation and control in the faster loop which is effective in regulation of torque output to the free reference ripple [10]. The toque ripple minimization requirements need the control system to be excellent in both dynamic and steady state responses.

In this work, two design methods are suggested to minimize torque ripples of PMSM. The first method is based on adaptive control technique which is a modified approach of [11], in addition to a space vector pulse width modulation (SVPWM) technique to obtain variable output having a maximum fundamental component with minimum harmonics. Adaptive controller is based on two loops controller structure (current controller and speed controller) with the modification as converting the reference dq voltages in to αβ voltages (will be discussed later) which serve as inputs to the SVPWM. The second method is using PI controller to produce q reference current (iq*) from speed error signal and set the i_d current to zero (i_d=0).

The remainder of this paper is organized as the following. Section 2 presents the mathematical model of the PMSM involved in the work. Section 3 describes the two proposed algorithms for minimization torque ripples in PMSM. Section 4, presents the simulation results obtained from the Adaptive and PI-controlled PMSM. Then, a comparison analysis is made to previously related work in section 5.

2. PMSM MODEL

The most commonly used dq-reference model [2] can be described as:

$$\frac{di_d}{dt} = -\frac{R}{L_d}i_d + \frac{L_q}{L_d}0.5pwi_q + \frac{v_d}{L_d} \quad (1)$$

$$\frac{di_q}{dt} = -\frac{R}{L_q}i_q - \frac{L_d}{L_q}0.5pwi_d - 0.5pwk_e + \frac{v_q}{L_q} \quad (2)$$

$$\frac{dw}{dt} = 1.5 * \frac{k_e}{J}i_q + 1.5 * \frac{(L_d - L_q)}{J}i_di_q\frac{T_l}{J} - \frac{B}{J}w - \frac{F}{J}\frac{w}{|w|} \quad (3)$$

$$\frac{d\theta}{dt} = 0.5pw \quad (4)$$

Where, Table 1 illustrates previous variable's definition.

i_d	direct -axis stator currents
i_q	quadrature -axis stator currents
v_d	direct -axis stator voltages
v_q	quadrature-axis stator voltages
R	stator phase resistance
L_d	direct-axis phase inductances
L_q	quadrature-axis phase inductances
J	Rotor and load inertia
F	Coulomb friction
B	Moment of inertia load
T_l	Load torque
p	Number of phase
w	Rotor speed
θ	Rotor position
k_e	Torque constant

Table 1. Variable's definition of PMSM model.

From the speed state equation (3), it's shown that if the d-axis current is maintained constant, the generated torque is proportional to the q-axis current. So, the q-axis current is the only control variable in the generated torque, which is similar

to the torque control technique used in DC machines. The responsible for higher harmonics in fundamental and reluctance torque is non-ideal distribution of the stator winding or rotor magnet. The fundamental torque is the dominant torque production mechanism in most PMSMs but, the fundamental torque ripples appear at the high frequency range, i.e., multiple of six electrical shaft speed ($6f_e$, where f_e is the electrical shaft speed in Hz). Slot torque is the circumferential component of attractive force that attempts to maintain the alignment between the stator teeth and the permanent magnets. Slot torque harmonics appear at frequencies that are multiple of Nsf_e, where Ns is the number of slots per pole pair [11]-[12].

The analytical modeling of the slot torque is challenging since it involves the complex field distributions around the slots of the stator. Thus, the slot torque is usually reduced in machine design procedure with the aid of numerical model [13]. Moreover, a large number of PMSMs are manufactured with anisotropic rotor. In this case the reluctance torque is negligible, and therefore it does not contribute to torque ripple. Motor with this construction produces only fundamental torque with large DC component and with higher harmonics [3]-[4]. It is assumed in the proposed work the availability of well designed PMSM with negligible reluctance and slot torque, which is often the case in high performance applications [14]. Thus our aim is to model fundamental torque only and concentrates on utilizing control feedback to minimize the ripple components. However, the slot torque may become a dominant ripple component in motors with different construction.

3. SUGGESTED ALGORITHMS FOR MINIMIZING TORQUE RIPPLES IN PMSM

3.1 Adaptive control algorithm.

PMSM dq-reference model can be described also, as:

$$\frac{di_d}{dt} = -\frac{R}{L_d}i_d + \omega\frac{L_q}{L_d}i_q - \omega\frac{1}{L_d}\Phi_d(\theta) + \frac{1}{L_d}v_d \quad (5)$$

$$\frac{di_q}{dt} = -\frac{R}{L_q}i_q - \omega\frac{L_d}{L_q}i_d - \omega\frac{1}{L_q}\Phi_q(\theta) + \frac{1}{L_q}v_q \quad (6)$$

$$T_m = p * (i_d\Phi_d(\theta) + i_q\Phi_q(\theta)) \quad (7)$$

$$\frac{d\omega}{dt} = \frac{T_m}{J} - \frac{B}{J}\omega - \frac{1}{J}T_l \quad (8)$$

Where T_m is torque produced by the motor, J is the moment of inertia, $\Phi_d(\theta)$ is direct flux and, $\Phi_q(\theta)$ is quadrature flux.

A PMSM dq model equation (5)–(8) is similar to the standard model equations (1)–(4). The difference between the two models is that the back emf terms and motor torque expression depend on the position of the rotor, but the standard model does not. The permanent magnet flux linkages can be viewed as the sum of a fundamental component and the series of higher harmonics [11] following two equations (9)-(10) illustrate this:

$$\Phi_d(\theta) = \Phi_{d6}\sin(6\theta) + \Phi_{d12}\sin(12\theta) \quad (9)$$

$$\Phi_q(\theta) = \Phi_{q0}\Phi_{q6}\cos(6\theta) + \Phi_{q12}\cos(12\theta) \quad (10)$$

Where Φ_{d6} is six harmonic direct flux , Φ_{d12} is twelve harmonic direct flux, Φ_{q0} is zero harmonic quadrature flux, Φ_{q6} is six harmonic quadrature flux and, Φ_{q12} is twelve

harmonic quadrature flux. Where the coefficients Φ_{dk} Φ_{qk}, based on back emf measurement and performed on PMSM is used in our simulations.

The model equations (5)–(8) are compressed into two equations for simplification, producing the next two equations (11)-(12):

$$\begin{bmatrix} \frac{di_d}{dt} \\ \frac{di_q}{dt} \end{bmatrix} = \\ -\begin{bmatrix} \frac{R}{L_d} & 0 \\ 0 & \frac{R}{L_q} \end{bmatrix}\begin{bmatrix} i_d \\ i_q \end{bmatrix} - \omega \begin{bmatrix} 0 & -\frac{L_q}{L_d} \\ \frac{L_d}{L_q} & 0 \end{bmatrix}\begin{bmatrix} i_d \\ i_q \end{bmatrix} - \omega \begin{bmatrix} \frac{1}{L_d}\Phi_d(\theta) \\ \frac{1}{L_q}\Phi_q(\theta) \end{bmatrix} + \\ \begin{bmatrix} \frac{1}{L_d}v_d \\ \frac{1}{L_q}v_q \end{bmatrix} \tag{11}$$

$$J\frac{d\omega}{dt} = p\begin{bmatrix} \Phi_d(\theta) \\ \Phi_q(\theta) \end{bmatrix}^T\begin{bmatrix} i_d \\ i_q \end{bmatrix} - B\omega - T_l \tag{12}$$

Two loops controller structure is adopted with the standard time separation being used in the design of the controller. An adaptive controller for the current loop design is neglecting a much slower mechanical dynamic. Then a speed loop controller is designed assuming electrical dynamic is infinitely fast.

3.1.1 Current Loop

Usually the current control subsystem consists of PI controllers of d-axis and q-axis currents. It is used to convert the torque reference value to current reference value.

A preparatory design step is to factor out the unknown parameters of the $\Phi(\theta)$ function by the following:

$$\Phi(\theta) = \begin{bmatrix} \Phi_d(\theta,\eta^*) \\ \Phi_q(\theta,\eta^*) \end{bmatrix} = \\ X(\theta).[\Phi_{d6} \quad \Phi_{d12} \quad \Phi_{q0} \quad \Phi_{q6} \quad \Phi_{q12}]^T \tag{13}$$

$$\text{Let} \quad \eta^* = [\Phi_{d6} \quad \Phi_{d12} \quad \Phi_{q0} \quad \Phi_{q6} \quad \Phi_{q12}]^T \tag{14}$$

Where $X(\theta)$ is a known 2*5 matrix whose entries are periodic functions of the rotor position, and η^* is the unknown vector of the actual coefficients of $\Phi(\theta)$.

By using the vector $\hat{\eta}$, to dynamically estimate η^* it is desirable to set the i_d current to zero and to choose the i_q current to produce the desired torque reference T^* [12], since from equation (7), $\Phi_d(\theta)$ doesn't have a DC component and so, the nonzero i_d contributes only to torque ripples. Conversion of torque reference value T^* to current reference value $i_q{}^*$ is realized using the extended current factor equation (15). The torque constant k_e in equations (1)-(4) is replaced by a term which contains harmonic functions Φ_{dk} Φ_{qk} as in equations (5)-(8). The current waveform is then shaped so that the produced fundamental torque is smooth and its ripple is suppressed.

$$\begin{bmatrix} i_d{}^* \\ i_q{}_* \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{T_*}{P\Phi_q(\theta,\hat{\eta})} \end{bmatrix} \tag{15}$$

Since currents are closely related to torque production, next we specify a desired current vector that will produce the desired (ripple free) torque.

Voltage output of the controller is selected to satisfy equation (11) when desired values of currents are reached.

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \\ \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix}\begin{bmatrix} \frac{di_d{}^*}{dt} \\ \frac{di_q{}^*}{dt} \end{bmatrix} + \begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix}\begin{bmatrix} i_{d*} \\ i_{q*} \end{bmatrix} + \omega \begin{bmatrix} 0 & -L_q \\ L_d & 0 \end{bmatrix}\begin{bmatrix} i_d{}^* \\ i_q{}^* \end{bmatrix} + \\ \omega\, X(\theta)\hat{\eta} + \mathcal{P}\left(\begin{bmatrix} i_d{}^* \\ i_q{}^* \end{bmatrix} - \begin{bmatrix} i_d \\ i_q \end{bmatrix}\right) \tag{16}$$

Where design parameter $\mathcal{P} > 0$. One can Let $\tilde{\eta} = \hat{\eta} - \eta^*$ and $\tilde{\imath} = \hat{\imath} - i^*$ represent the deviations from the desired values.

Subtracting equation (16) from (11):

$$\begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix}\begin{bmatrix} \frac{\widetilde{di_d}}{dt} \\ \frac{\widetilde{di_q}}{dt} \end{bmatrix} + \left(\begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix} + p\begin{bmatrix} i_d \\ i_q \end{bmatrix}\right)\begin{bmatrix} \tilde{\imath}_d \\ \tilde{\imath}_q \end{bmatrix} + \\ \omega\begin{bmatrix} 0 & -L_q \\ L_d & 0 \end{bmatrix}\begin{bmatrix} \tilde{\imath}_d \\ \tilde{\imath}_q \end{bmatrix} + \omega\, X(\theta)\tilde{\eta} = 0 \tag{17}$$

An update law for $\tilde{\eta}$ has to guarantee both $\tilde{\eta} \rightarrow 0$ and $\tilde{\imath} \rightarrow 0$.

By changes in the design parameter, $\mathcal{P}$ we can control the parameter convergence rate. Where increasing it, will lead to rapid current convergence and a more aggressive control.

3.1.2 Speed Loop

PI controllers are used widely for motion control systems that produce the torque reference and its time derivative. Speed controller calculates the difference between the reference speed and the actual speed producing an error, which is fed to the PI controller.

The parameter of the speed loop controller is chosen to achieve the desired performance in speed tracking. The speed controller is based on a PI regulator and the output of this regulator is a reference torque and derivative of this torque T^*, can be calculated from the speed error (ω) using the same controller without integral action. The method is equivalent to generating T^*, and $\dot{T}^*$ using the same PI controller and an approximate derivative of the speed error [15]-[18]. Using the two controllers (speed and current) to produce the reference dq voltages. Then conversion to $\alpha\beta$ voltages equations (18)-(19) as well as using SVPWM represents the novelty in our proposed approach.

$$v_\alpha = \cos(\theta) * v_d - \sin(\theta) * v_q \tag{18}$$

$$v_\beta = \sin(\theta) * v_d + \cos(\theta) * v_q \tag{19}$$

The $\alpha\beta$ voltages then serve as inputs to (SVPWM) to produce the control signal to the inverter. Space vector modulation is based on the representation of the three phase quantities as vectors in a two-dimensional ($\alpha\beta$) plane. SVPWM can be implemented by the following steps: 1) Determining Voltage Space Vector 2) Determine time duration 3) Determining the switching time of each transistor (S1 to S6).

SVPWM technique has become a popular pulse width modulation technique for three phase voltage-source inverter in the control of AC motors. It is a more sophisticated technique for generating sine wave that provides a higher voltage to the motor with lower total harmonic distortion. The main aim this technique is to obtain variable output having a maximum fundamental component with minimum harmonics. It is used for the creation of alternating current (AC)

waveforms; most commonly to drive three phase AC powered motors at varying speeds from DC using multiple class-D amplifiers. There are various variations of SVM that result in different quality and computational requirements. One active area of development is in the reduction of total harmonic distortion (THD) created by the rapid switching inherent to these algorithms. Space vector modulation is based on the representation of the three phase quantities as vectors in a two-dimensional (αβ) plane. The main target of introducing SVPWM is to minimize harmonic distortion in the current by selecting the appropriate switching vectors and determining their corresponding dwelling widths [19, 20]. The αβ voltages serve as inputs to SVPWM to produce the control signals to inverter.

The flow chart of the first suggested control algorithm describing its steps is shown in Fig. 1. The Simulink block diagram of proposed torque ripple minimization using adaptive controller is show in Fig.2

The suggested adaptive control algorithm is as follows:

- Define the mathematical model of the system.
- Compare reference and actual speed.
- If difference between the speed and reference speed is less than or equal ε, stop and print torque and speed curves.
- Else, generate the torque reference T^* and $\dot{T^*}$ from the speed error using PI controller.
- Estimate flux (η̂) then, Calculate the reference dq current i^* and the derivative of this currents using equation (15).
- Calculate the reference dq voltages (v_q , v_d) using vdq* generator block as shown in fig.2.
- Convert the reference dq voltages to αβ voltages by using equations (18) and (19).
- Generate the control signal from αβ voltages using SVPWM which introduces its output signal to the inverter.

3.2 PI control algorithm.

In order to validate the robustness of the proposed adaptive method over the classical PI, the feasibility of reducing torque ripples of PMSM using PI controller, is also tested. Illustrating the methodology of designing this controller explained in the following section.

The output control signal from the PI controller U(k) at time step k is given by [21].

$$U(k) = K_p \left(e(k) + \frac{1}{T_i} \sum_{j=1}^{k} e(j) \, T \right) \qquad (20)$$

Where, K_p is the proportional gain, e(k) is the speed error at time step k (rad/sec), T_i is the integral time constant (sec), and T is the sampling period (sec).
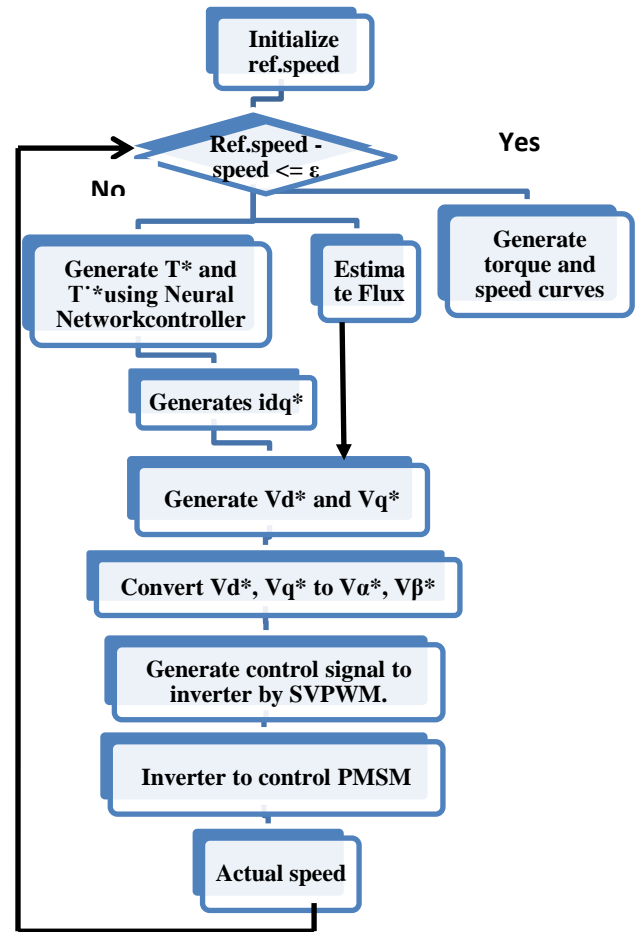

Fig.1. the flow chart of the adaptive suggested control algorithm.

The proposed PI controller can be applied to this class of motor with current controller enables tracking of quadrature current command values. The Simulink block diagram of proposed using PI controller for torque ripple minimization is show in Fig.3.

The PMSM model employed in our simulation has two inputs: the load torque and three phase input source and six outputs: motor measurement vector speed, torque, abc currents, dq currents, theta, and dq voltages. The abc currents to dq currents converter (abc 2 dq block) uses the following equations (21)-(22).

$$i_q = \frac{2}{3}[\cos(\theta) * i_a + \cos(\theta - 120) * i_b + \cos(\theta + 120) * i_c] \quad (21)$$

$$i_d = \frac{2}{3}[\sin(\theta) * i_a + \sin(\theta - 120) * i_b + \sin(\theta + 120) * ic] \qquad (22)$$

It is done by converting the three phase currents to dq variables using Parks transformation [8]. Park's transformation is:

$$P = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - 120) & \cos(\theta + 120) \\ \sin(\theta) & \sin(\theta - 120) & \sin(\theta + 120) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \qquad (23)$$
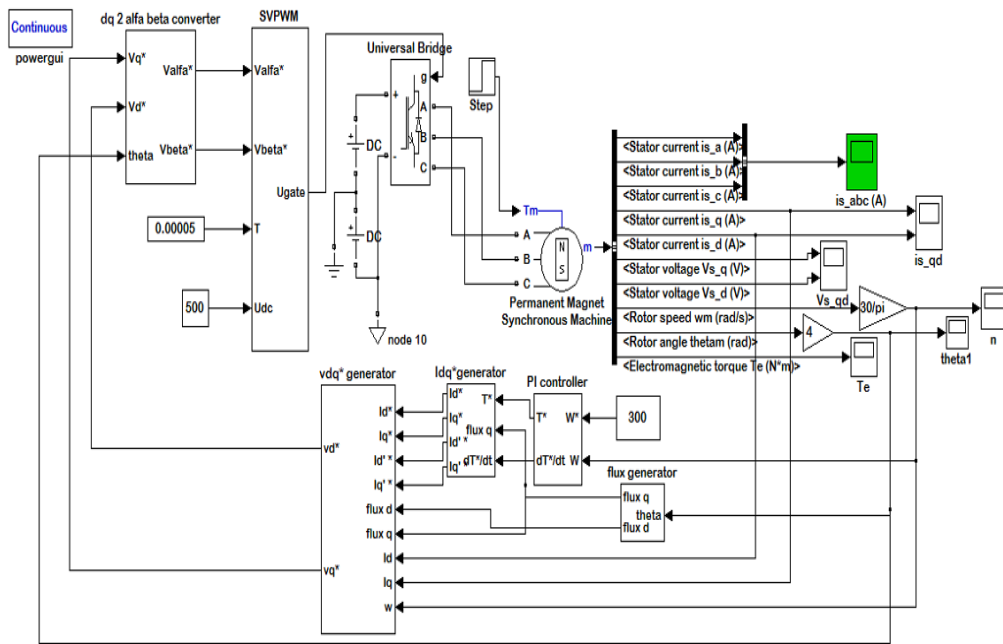
Fig. 2 Simulink Block diagram of the torque ripple minimization using the suggested adaptive controller.
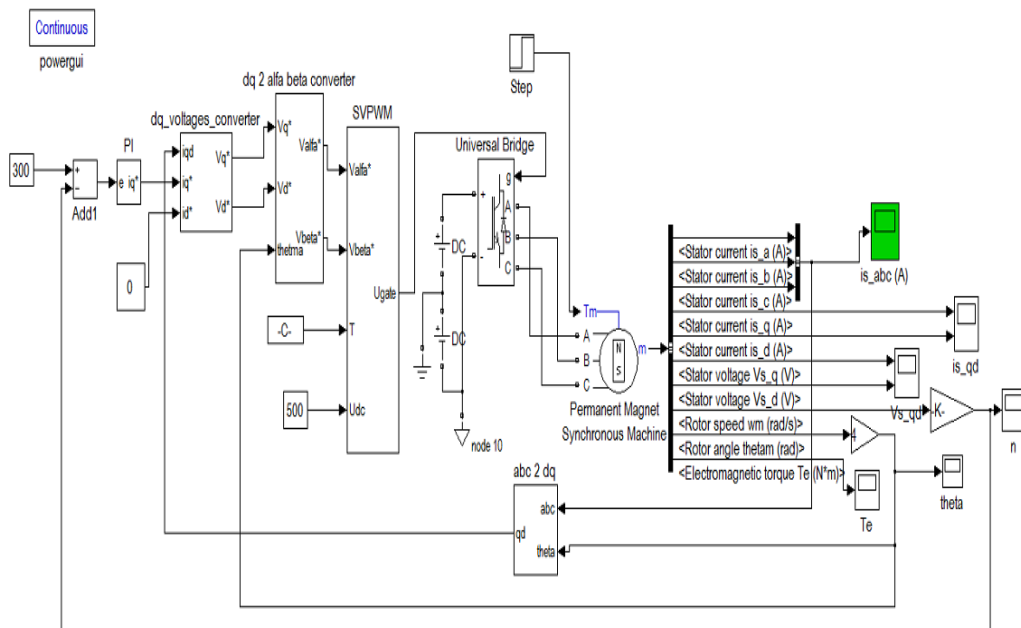

Fig. 3 Simulink Block diagram of the torque ripple minimization using the suggested PI controller.

PI controller produces q reference current (iq*) after introducing speed error signal as an i/p. Using this generated iq* current, the dq actual current, and the d reference current (which equal zero) dq reference voltages are produced by dq_voltages_converter block, which is then converted to αβ voltages using equations (18) and (19). The αβ voltages serve as inputs to SVPWM to produce the control signals to inverter.

The flow chart of the second suggested control algorithm is shown in Fig. 4.

The suggested control algorithm is as follows:

- Define the mathematical model of the system.
- If the difference between reference and actual speed smaller or equal constant ε, print speed and torque curves.
- Else, using the PI controller to minimize the torque ripple as the following:-
- Generate the quadrature current (i_q*) from the speed error using PI controller.
- Calculate the reference dq voltages (v_q *, v_d *) by using dq_voltages_converter block as shown in fig.3.
- Convert the reference dq voltages to αβ voltages by using equations (18) and (19).
- Generate the control signal from αβ voltages using SVPWM which generate signal to inverter.
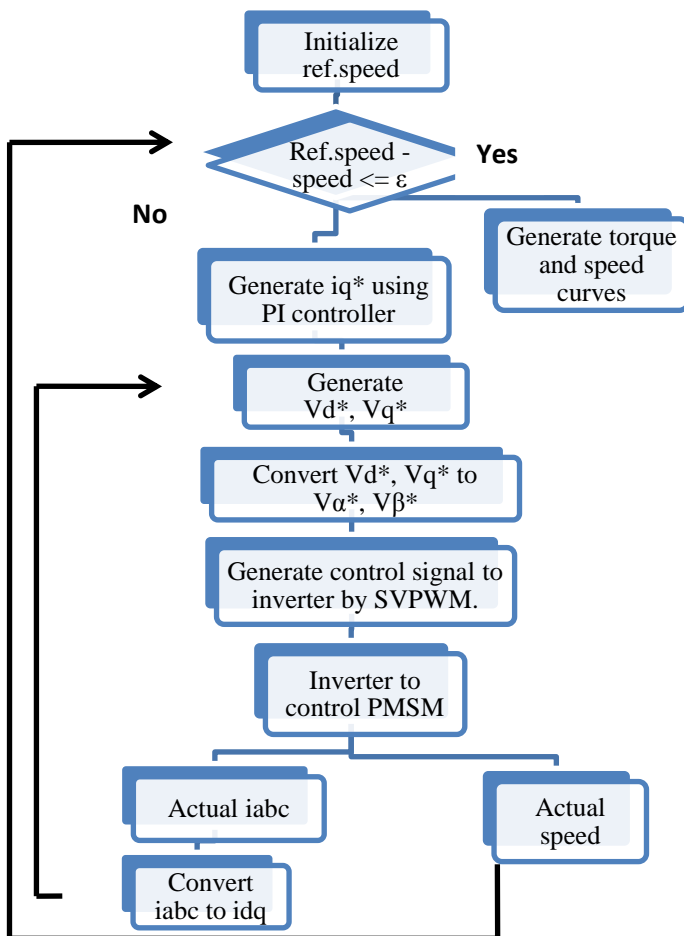

Fig.4. the flow chart of the PI suggested control algorithm.

The outputs of the algorithm are the torque and speed when the difference between the speed and reference speed less than or equal ε which is constant.

4. SIMULATION RESULTS

After the theoretical derivation, the proposed controllers were suggested and tested in simulations. An ideal scenario is simulated with results confirming the correctness of the design. All simulations were performed using the Matlab 2009 Simulink toolbox, with equations (9)–(12) used to model the motor behavior. Table 2 illustrate the parameter values for PMSM that is used in the proposed work. Dynamic characteristic values. For the proposed adaptive controller, torque curve is shown in fig.5. It has overshoot percentage as 427%, the delay time as 3.57e-4s, the Peak time as 1.73e-3s, the rise time as 2.225e-3s, and the torque ripple percentage as 1.096%. For its speed curve shown in fig.6, it has overshoot percentage as 26.667 %, the delay time as 0.03126s, the Peak time as 0.0347s, and the rise time as 0.0029s. For the PI controller: the torque curve is shown in fig.7 with the overshoot percentage as 880%, the delay time as 0.0108s, the Peak time as 0.01348s, the rise time as 0.0002s, and the torque ripple percentage as 26.7%. The speed curve is also shown in fig.8 for the PI method, with an undershoot percentage as 154%, the delay time as 0.055s, and the rise time as 0.0145s. Table 3 shows all previous results.
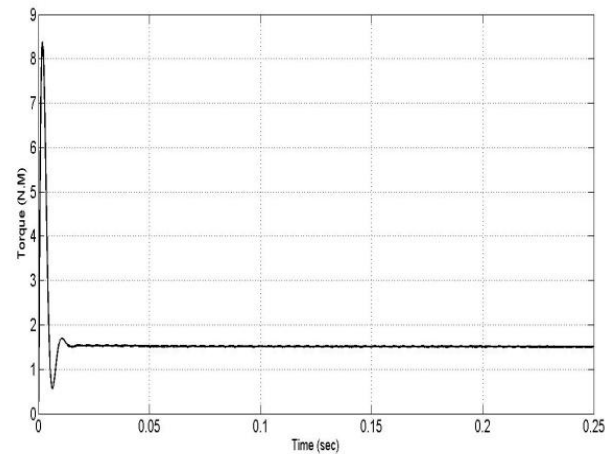

Fig. 5 The torque curve of the PMSM driven by an adaptive controller.
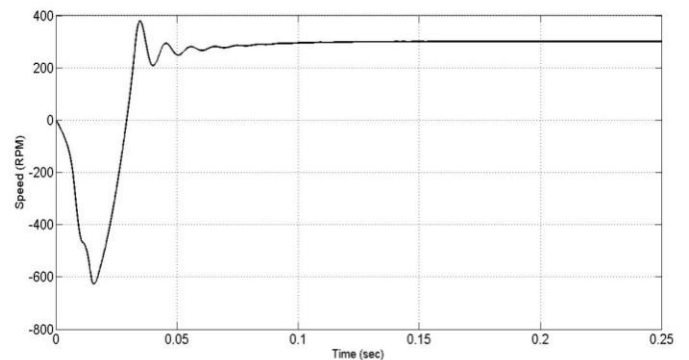

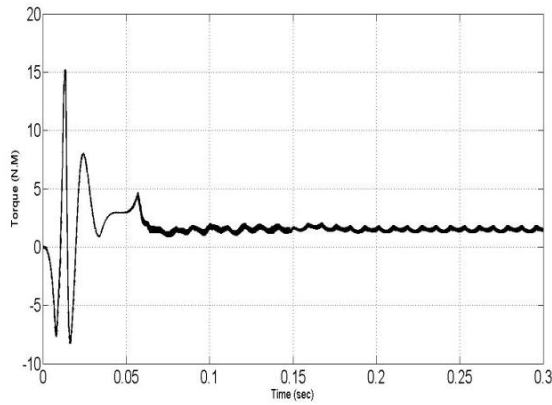Fig. 6 The speed curve of the PMSM driven by an adaptive controller.

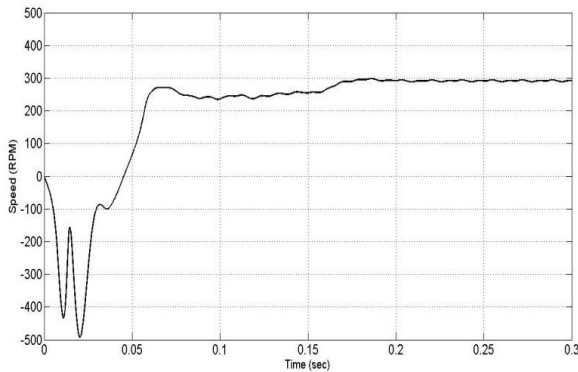**Fig. 7 The torque curve of the PMSM driven by a PI controller.**

Fig. 8 The speed curve of the PMSM driven by a PI controller.

Table 2 illustration of the parameter values for PMSM that is used in the present work.

Parameter	Value
P	2
L_q	0.0091 H
L_d	0.0091 H
R_s	1.45 Ω
J	0.0011 [kg. m^2]
B	0.0009 $\left[\frac{\text{kg.m}^2}{\text{s}}\right]$
Φ_{q0}	0.1994(v.s)
Φ_{q6}	0.0091(v.s)
Φ_{q12}	0.0012(v.s)
Φ_{d6}	0.0018(v.s)
Φ_{d12}	0.0011(v.s)

Table 3 Comparative Analysis between dynamic characteristics of a two suggested methods.

		PI	Adaptive
Torque	Maximum overshoot (Mp)	15.3 n.m	8.225 N.m
	delay time (td)	0.0108 s	3.57e -4 s
	peak time (tp)	0.01348 s	1.73e-3 s
	rise time (tr)	0.0002 s	2.225 e -3 s
	steady state	1.56 n.m	1.56 n.m
	Ripple (max-min)	0.414	0.017

speed	Maximum overshoot (Mp)	no overshoot	80 rpm
	delay time (td)	0.055 s	0.03126 s
	peak time (tp)	no	0.0347 s
	rise time (tr)	0.0145 s	0.0029 s
	steady state	300 rpm at 0.14 s	300 rpm at 0.128 s

It's observed that the proposed adaptive controller succeeded to enhance the ripples as well as the dynamic characteristics in comparison with the PI proposed one. This reduction was around 1.096% with the value of ripples (max-min) = 0.017 N.M for the adaptive controller and 0.414 N.M for the PI one. Moreover, this result is compared to other publication for more validation. W. Qian and Panda, presented an iterative learning control (ILC) scheme implemented in time domain to reduce periodic torque pulsations. A forgetting factor was introduced in this scheme to increase the robustness of the algorithm against disturbances. However, this limited the extent to which torque pulsations can be suppressed. In order to eliminate this limitation, a modified ILC scheme was implemented by [8] in frequency domain by means of Fourier series expansion. Torque ripple percentage was 3.9% which is still higher than a proposed adaptive method but of course better than the PI classical proposed method. While H.M. Hasanien et al., presented a digital observer controller that was applied to the surface-mounted PMSMs for the purpose of minimizing its torque ripple [18]. Their result was 12%. Torque ripples in compression to our adaptive method are actually higher, however still better than the PI proposed one. P. Mattavelli et al., proposed a repetitive-based control for torque-ripple reduction that was committed to the reduction of the high-frequency periodic current error, which cannot be compensated by the PI due to its limited bandwidth. The modification, which contained an alternating component superposed to the conventional dc component, was tracked by a PI parallel of repetitive controller [22]. Torque ripple percentage was ≈ 3.8% which is still higher than the proposed adaptive method but still better than the PI classical proposed method. M.Tárník et al., proposed an additional adaptive current controller. It is assumed that the basis of the control system was a conventional current control loop commonly used in vector control [23]. Torque ripple percentage was ≈ 4% which is also higher than the proposed adaptive method but also still better than the PI classical proposed method.

Table 4, presents comparison analysis between different related work to torque ripples in percentage.

Table 4: The percentage torque error obtained in various control strategies in different works.

Control Strategies	Torque Ripple Percentage
Proposed Adaptive controller first method.	1.096%
Proposed PI controller second method.	26.7%
(Qian and Panda, 2004) [8].	3.9%
(H. Hasanien, 2010) [18].	12%
(P. Mattavelli, 2005) [22].	≈ 3.8%
(M.Tarnik and J.Murgas, 2011) [23].	≈ 4%

5. CONCLUSION

A versatile PMSM model for torque ripple minimization has been developed and tested in simulations. A modified approach for an adaptive controller is suggested to minimize torque pulsations based on flux estimation and SVPWM. Also, a PI controller was tested to compare with the adaptive one. A simulation of the continuous controller model shows a great reduction of torque ripples using the adaptive approach indicating a proper controller design.

6. ACKNOWLEDGMENTS

7. REFERENCES

[1] T. Li and G. Slemon, "Reduction of slot torque in permanent magnet motors," IEEE Trans. Magn., vol. 24, no. 6, 1988.

[2] T.M. Jahns and W. L. Soong, "Pulsating torque minimization techniques for permanent magnet AC motor drives—A review," IEEE Trans. Ind. Electron., vol. 43, Apr. 1996.

[3] A. Kaddouri and H. Le-Huy, "Analysis and design of a slotlessNdFeB permanent magnet synchronous motors for direct drive," in Proc. Rec. IEEE IAS Annu. Meeting, 1992.

[4] C. Studer,A.Keyhani, T. Sebastian, and S. K.Murthy, "Study of slot torque in permanent magnet machines," in Proc. Rec. IEEE IAS Annu. Meeting, New Orleans, LA, October 1997.

[5] J. Y. Hung and Z. Ding, "Design of currents to reduce torque ripple in brushless permanent magnet motors," Proc. Inst. Elect. Eng. B, vol. 140, no. 4, 1993.

[6] J. Holz and L. Springob, "Identification and compensation of torque ripple in high-precision permanent magnet motor drives," IEEE Trans. Ind. Electron., vol. 43, Apr. 1996.

[7] K.Gulez, A. A.Adam, Pastacı H., "Passive Filter Topology to Minimize Torque Ripples and Harmonic Noises in IPMSM Derived with HDTC", IJE-International Journal of Electronics, Vol. 94, No:1, Jan. (2007).

[8] W. Qian and S. K. Panda,"Torque Ripple Minimization in PM Synchronous Motors Using Iterative Learning Control" IEEE Trans, VOL. 19, NO. 2, MARCH 2004

[9] A. A. Adam And K.Gulez, "A New Sensorless Hysteresis Direct Torque Control Algorithm for PMSM with Minimum Torque Ripples", COMPEL, Vol.28, No.2, April 2009.

[10] S. O. Bogosyan and M. Gokasan, "Adaptive torque ripple minimization of permanent magnet synchronous motors for direct drive applications," in Proc. Rec. IEEE IAS Annu. Meeting, vol. 1, Orlando, FL, Oct. 1995.

[11] V.Petrovic and R.Ortega,"Design and Implementation of an Adaptive Controller for Torque Ripple Minimization in PM Synchronous Motors" "IEEE Transactions on Power Electronics, VOL. 15, NO. 5, 2000"

[12] H. Khalil, Nonlinear Systems, 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 1996.

[13] Zhang Jianzhong, Cheng Ming, Chen Zhe. "Optimal design of stator interior permanent magnet machine with minimized slot torque for wind power application", Energy Convers Manage 2008.

[14] D. C. Hanselman, "Minimumtorque ripple, maximum efficiency excitation of brushless permanent magnet motors," IEEE Trans. Ind. Electron., vol. 41, June 1994.

[15] C. Elmas and O. Ustun "A hybrid controller for the speed control of a permanent magnet synchronous motor drive"Control Engineering Practice 16 (2008).

[16] R. Ortega, A. Loria, P. J. Nicklasson, and H. Sira-Ramirez, Passivity Based Control of Euler–Lagrange Systems. London, U.K.: Springer Verlag, 1998.

[17] Fernández P, Güemes JA and Iraolagoitia AM. "Speed control of permanent magnet synchronous motors by current vector control". In: International conference on electrical machines (ICEM); September 2006.

[18] H. Hasanien, "Torque ripple minimization of permanent magnet synchronous motor using digital observer controller" Elsevier, 2010.

[19] K. V. kumar, P. A.l Michael, J. P. John and S. Suresh Kumar, "simulation and comparison of spwm and svpwm control for three phase inverter", arpn journal of engineering and applied sciences, vol. 5, no. 7, july 2010.

[20] A. Iqbal and S. M. Ahmed, M.A.Khan, H.Abu-Rub "Generalised simulation and experimental implementation of space vector PWM technique of a three-phase voltage source inverter", ijest Vol. 2, No. 1, pp. 1-12, 2010.

[21] P. Stewart and V. Kadirkamanathan "Dynamic model reference PI control of permanent magnet AC motor drives ", Elsevier, Control Engineering (2001).

[22] P. Mattavelli, L.Tubiana, and M. Zigliotto, "Torque-Ripple Reduction in PM Synchronous Motor Drives Using Repetitive Current Control", IEEE Transactions On Power Electronics, VOL. 20, NO. 6, November 2005.

[23] M.Tarnik and J.Murgas, "Additional Adaptive Controller for Mutual Torque Ripple Minimization in PMSM Drive Systems", Slovakia, 18th IFAC World Congress, Milano (Italy) August 28 - September 2, 2011.