Design and Implementation of an Adaptive Controller for Torque Ripple Minimization in PM Synchronous Motors

Vladan Petrović, Romeo Ortega, Aleksandar M. Stanković, and Gilead Tadmor

***Abstract*—This paper addresses the problem of torque ripple minimization in Permanent Magnet Synchronous Motors (PMSMs) and proposes an adaptive feedback structure as a solution. A model of PMSM that includes a torque ripple phenomenon is first developed and tested. While slightly different from the conventional one, our model is still compact and suitable for control. All parameters of the model have physical interpretation, and can either be measured directly or estimated in a numerically reliable procedure. An adaptive control algorithm is then described, enabling speed tracking while minimizing the torque ripple. Finally, the proposed algorithm is verified in simulations and implemented in a hardware setup. Experimental results show significant reduction of torque ripple (by 27 dB). Extensive analysis and simulations of hardware imposed limitations were performed as well, revealing and quantifying the issues that might affect practical ripple minimization performance.**

***Index Terms*—Motion control, motor drives, permanent magnet motors, torque ripple minimization.**

I. Introduction

PERMANENT magnet synchronous motors (PMSMs) are widely used in high performance variable frequency drives. Their popularity is justified by several advantages over commonly used motors in these applications. The absence of the rotor winding eliminates losses on the rotor and makes PMSMs highly efficient. In addition, the absence of the external rotor excitation renders slip rings on the rotor and brushes obsolete, and thus reduces the maintenance costs. New magnetic materials are capable of creating high magnetic fields that yield high power density. This in turn results in a rapid dynamic response due to high torque-to-inertia ratio.

The main disadvantage of PMSMs is nonuniformity in the developed torque, i.e., the torque ripple. Since PMSMs aim for high performance applications such as machine tools or direct drive robotics, torque oscillations are not acceptable. Torque ripple leads to speed oscillations which cause deterioration in system performance. In machine tool applications, these oscillations leave visible patterns in high-precision machined surfaces. As technology of permanent magnets advances, the power range of PMSMs also expands. In large machines (e.g. ones intended for use in elevator drives), inertia becomes large enough to lower mechanical resonant frequencies of the system to the range where torque ripple harmonics exist, risking further increase in speed oscillations.

A large number of techniques for torque ripple minimization has been proposed in literature [1]. Broadly speaking, these techniques fall into two major categories. The first class consists of techniques that concentrate on the motor design so that PMSM more closely approaches its ideal characteristics for achieving smooth torque production [2]–[4]. Although effective in ripple minimization, machine design techniques additionally complicate the production process and increase the final machine cost.

The second class of algorithms, which is in our focus, consists of techniques for minimization of torque ripple using an additional control effort to correct for nonideal characteristics of the machine [5]–[12]. One popular approach is harmonic cancellation using the preprogrammed current waveforms [5], [6]. This method relies on knowledge of torque ripple characteristics of the specific motor, and uses the torque production model to calculate optimal currents that need to be injected to cancel the undesired torque components. Being based on off-line calculations, these techniques are sensitive to parameter variations and thus their performance degrades when the operating conditions change. To account for parameter variations during the motor operation, on-line estimation techniques were proposed and reported in literature [7]–[12]. Estimation and control schemes are employed either in speed or current (torque) loops. Instantaneous torque controllers, for example, replace current loop with torque loop and use motor torque observers to obtain necessary feedback. Different methodologies have been proposed for torque observation (e.g., recursive least squares [9] and model reference adaptive system [10] techniques), and all of them extract information from the electrical subsystem (i.e., current measurements) to estimate the complete torque waveform. This waveform is then used as a feedback signal for the torque controller. Quality of current measurements enables accurate estimation, and control in the faster inner loop is effective in regulation of the output torque to the ripple free reference. On the other hand, this approach can be used only for those ripple components that are observable from currents—cogging torque and load oscillations can not

Manuscript received September 10, 1999; revised April 3, 2000. This work was supported in part by the National Science Foundation under Grant ECS-9502636, the Office of Naval Research under Grant N14-97-1-0704, and ARO under Grant DAAD19-99-1-0145. Recommended by Associate Editor J. Ojo.

V. Petrović, A. M. Stanković, and G. Tadmor are with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115 USA.

R. Ortega is with Laboratoire des Signaux et Systemes Supelec, Gif-sur-Yvette 91192, France.

Publisher Item Identifier S 0885-8993(00)07311-7.

TABLE I
DEPENDENCE OF MUTUAL AND RELUCTANCE TORQUE ON MOTOR CONSTRUCTION

Sinusoidal distribution of		*Cylindrical*	*Mutual torque*	*Reluctance torque*
stator phases	*rotor magnet*	*rotor*		
Yes	Yes	Yes	DC component only	None
Yes	Yes	No	DC component only	DC component only
Yes	No	Yes	DC component only	None
Yes	No	No	DC component only	DC component only
No	Yes	Yes	DC component only	None
No	Yes	No	DC component only	DC & higher harmonics
No	No	Yes	DC & higher harmonics	None
No	No	No	DC & higher harmonics	DC & higher harmonics

be minimized using this approach. The other possibility is to use mechanical variables (speed and position) in observation, and speed controller for ripple minimization ([11], [12]). All possible sources of ripple are observable from mechanical states, hence this method has potential for complete ripple minimization. However, quality of speed feedback and slow dynamics of the outer loop limit the achievable performance of these algorithms.

In this paper we present a novel adaptive control algorithm for minimization of torque ripple in PMSM drives, and discuss controller implementation and possible restrictions imposed by experimental conditions. The controller is based on the same principles (energy shaping and damping injection) as the passivity-based controller in [13]. The information about torque ripple harmonics used in adaptation is extracted from the electrical subsystem, and a current controller is designed to achieve ripple minimization. Compared with the previously reported algorithms, we propose an alternative to the estimation of the complete flux (torque) waveforms by parameterizing the flux expressions with a small number of parameters and adapting on these few coefficients. Such parameterization is possible since accurate modeling of torque ripple reveals information of the phase of harmonics. Thus, only amplitudes remain unknown and need to be estimated. The rate of change of harmonic magnitudes is much slower than the rate of change of the waveform itself, implying that less aggressive adaptation is needed, which generally yields better results as it is less sensitive to the experimental noise.

The remainder of this paper is organized as follows. In Section II we introduce a PMSM model suitable for torque ripple analysis. This model is then used as a basis for controller development presented in Section III. Section IV shows simulation results supporting the theoretical derivations and deals with controller parameter selection. A detailed analysis and simulations of the limitations imposed by hardware implementation is performed, and results are presented in this section as well. In Section V we show experimental results and discuss issues of practical importance for controller implementation. We conclude the paper with some final remarks.

II. A dq MODEL OF PMSM FOR TORQUE RIPPLE ANALYSIS

In this section we briefly discuss the torque production mechanism in PMSMs, and develop a model suitable for torque ripple analysis taking into account special features of PMSM design.

The torque produced by a PMSM can be divided into three components:

1) mutual torque, which is due to the interaction of the rotor field and stator currents;
2) reluctance torque, which is due to rotor saliency;
3) cogging torque, which is due to the existence of stator slots.

Each component can contribute to higher harmonics in total torque, i.e., to torque ripple. The phenomenon responsible for higher harmonics in mutual and reluctance torque is nonideal (nonsinusoidal) distribution of the stator winding or rotor magnet. Ideally, mutual flux (the part of the flux through stator windings due to the rotor field) is purely sinusoidal, and only then the motor would produce a constant mutual torque. This requires a sinusoidal spatial distribution of either the stator windings or of the field due to rotor magnets. As a perfect sinusoidal distribution is not achievable in practice, the resulting mutual flux contains higher harmonics, causing ripple in the mutual steady state torque in response to a purely sinusoidal voltage excitation. This ripple appears at multiples of $6f_e$ (f_e being the electrical shaft speed in Hz). Reluctance torque is nonzero only in the salient rotor case. Then, it has only a DC component if the stator windings are sinusoidally distributed. Otherwise, it has higher harmonics as well (also at multiples of $6f_e$). All possible cases for the harmonic contents of the reluctance and mutual torque are summarized in Table I. The cogging torque spectrum depends only on the geometry and number of stator slots. This torque component has no DC value, and thus contributes only to torque ripple. Cogging torque harmonics appear at frequencies that are multiple of $N_{slpp}f_e$, where N_{slpp} is the number of slots per pole pair. Analytical modeling of the cogging torque is challenging since its production mechanism involves complex field distributions around stator slots. Thus, the cogging torque and its minimization is usually addressed in machine design with the aid of various numerical tools [4].

The majority of PMSMs are manufactured with an isotropic (or almost isotropic) rotor, and in these cases reluctance torque (both DC and higher harmonics) is negligible. In addition, in a standard manufacturing procedure, rotor magnets or stator lamination stack can be skewed, thus reducing the alignment preference of the rotor magnets toward stator slots, i.e., minimizing the cogging torque. With the above construction, PMSM develops only the mutual torque with a large DC component but also with higher harmonics. In this paper we assume the availability of

a well-designed PMSM with negligible reluctance and cogging torque, which is often the case in high performance applications [10]. This assumption is also motivated by the design of a particular PMSM used in our experiments. Thus, we model only the mutual torque and concentrate on the ways to utilize control feedback to minimize its ripple components. Note, however, that cogging torque may become a dominant ripple component in motors with different construction. In that case, the adaptation should be employed in the speed loop, as discussed in the previous section.

Based on the integral Ampere's and flux conservation laws, expressions for mutual flux can be derived for various stator winding and rotor magnet configurations. In all cases these expressions are Fourier expansions with rapidly decaying coefficients, allowing good approximations based only on the first few terms. After the standard modeling of the mechanical and the rest of the electrical subsystems (using basic physical laws), the Park transformation can be applied yielding the following model of PMSM in the dq (synchronously rotating) frame

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

$$L_q\frac{di_q}{dt} = -R_s i_q - \omega L_d i_d - \omega\Phi_q(\theta) + v_q \tag{2}$$

$$\tau_m = P\cdot(i_d\Phi_d(\theta) + i_q\Phi_q(\theta)) \tag{3}$$

$$J\frac{d\omega}{dt} = \tau_m - B\omega - \tau_l. \tag{4}$$

Here P is the number of pole pairs, L_d and L_q are stator inductances in the dq frame (which are equal in the case of a cylindrical rotor—the case we concentrate on), R_s is stator winding resistance, τ_m is torque produced by the motor and τ_l is a load torque. J and B are the moment of inertia and the friction constant (both normalized with P). The position θ and angular velocity ω are measured in electrical radians and rad/sec, respectively.

PMSM dq model (1)–(4) is similar to the standard model. The main difference between the two models is that the back emf terms and motor torque expression depend on the rotor position in the above model, but do not in the standard one. This dependence is modeled with the $\Phi_d(\theta)$ and $\Phi_q(\theta)$ functions which are given as

$$\Phi_d(\theta) = \Phi_{d6}\sin(6\theta) + \Phi_{d12}\sin(12\theta)$$
$$\Phi_q(\theta) = \Phi_{q0} + \Phi_{q6}\cos(6\theta) + \Phi_{q12}\cos(12\theta).$$

Off-line identification of the coefficients Φ_{dk}, Φ_{qk} based on back emf measurements was performed on a PMSM used in our experimental setup. These are the parameter settings that we use as "true" values in our simulations. In practice, however, these coefficients may slowly change with shifting operating conditions (e.g., due to magnetic saturation). Thus, we have explored an adaptive scheme that tracks the coefficients of $\Phi(\theta)$ on-line.

For the purpose of control design, the model (1)–(4) is compressed into two equations in vector notation

$$\boldsymbol{L}\frac{di}{dt} = -\boldsymbol{R}i - \omega\boldsymbol{Y}\boldsymbol{L}i - \omega\Phi(\theta) + v \tag{5}$$

$$J\frac{d\omega}{dt} = P\Phi^{\top}(\theta)i - B\omega - \tau_l \tag{6}$$

where

$$\boldsymbol{L} = \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix},\quad i = \begin{bmatrix} i_d \\ i_q \end{bmatrix},\quad \boldsymbol{R} = \begin{bmatrix} R_s & 0 \\ 0 & R_s \end{bmatrix}$$
$$\boldsymbol{Y} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},\quad \Phi(\theta) = \begin{bmatrix} \Phi_d(\theta) \\ \Phi_q(\theta) \end{bmatrix},\quad v = \begin{bmatrix} v_d \\ v_q \end{bmatrix}.$$

III. An Adaptive Control Algorithm

The algorithm presented in this section focuses on adaptation on the "ripple" parameters Φ_{dk} and Φ_{qk}. A two loop controller structure is adopted and the standard time separation principle is used in the controller design. First, an adaptive controller for the inner (current) loop is designed neglecting a much slower mechanical dynamics and then, a speed loop controller is designed assuming that the electrical dynamics is infinitely fast.

A. Current Loop

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

$$\Phi(\theta) = \begin{bmatrix} \Phi_d(\theta,\,\eta_\star) \\ \Phi_q(\theta,\,\eta_\star) \end{bmatrix} = \chi(\theta)\eta_\star$$

where $\chi(\theta)$ is a known 2×5 matrix whose entries are periodic functions of the rotor position, and $\eta_\star$ is the unknown vector of the actual coefficients of $\Phi(\theta)$

$$\eta_\star = [\Phi_{d6},\,\Phi_{d12},\,\Phi_{q0},\,\Phi_{q6},\,\Phi_{q12}]^{\top}$$
$$\chi(\theta) = \begin{bmatrix} \sin(6\theta) & \sin(12\theta) & 0 & 0 & 0 \\ 0 & 0 & 1 & \cos(6\theta) & \cos(12\theta) \end{bmatrix}.$$

The unknown $\eta_\star$ will be dynamically estimated by a new vector $\hat{\eta}$ in the control algorithm.

Since currents are closely related to torque production, next we specify a desired current vector that will produce the desired (ripple free) torque. From the motor torque expression (3), it follows that a nonzero i_d contributes only to torque ripple [since the function $\Phi_d(\theta)$ does not have a dc component]. Thus it is desirable to set the i_d current to zero and to choose the i_q current to produce the desired torque reference $\tau_\star$, using the estimate $\hat{\eta}$

$$i_\star = \begin{bmatrix} 0 \\ \dfrac{\tau_\star}{P\Phi_q(\theta,\hat{\eta})} \end{bmatrix}. \tag{7}$$

The controller voltage output is then selected to satisfy (5) when the desired currents are reached

$$v = \boldsymbol{L}\frac{di_\star}{dt} + \boldsymbol{R}i_\star + \omega\boldsymbol{Y}\boldsymbol{L}i_\star + \omega\chi(\theta)\hat{\eta} + \rho(i_\star - i) \tag{8}$$

where we have injected an additional damping (specified by a design parameter $\rho > 0$) into the electrical subsystem to increase the dissipation of the incremental current errors.

Let $\tilde{\eta} = \hat{\eta} - \eta_\star$ and $\tilde{i} = i - i_\star$ represent deviations from the desired values. Subtracting (8) from (5), we have the following equation governing incremental dynamics:

$$\boldsymbol{L}\frac{d\tilde{i}}{dt} + (\boldsymbol{R} + \rho\boldsymbol{I})\tilde{i} + \omega\boldsymbol{Y}\boldsymbol{L}\tilde{i} - \omega\chi(\theta)\tilde{\eta} = 0. \tag{9}$$

An update law for $\tilde{\eta}$ has to guarantee both $\tilde{\eta} \to 0$ (so that the desired current vector (7) really produces the desired torque) and $\tilde{i} \to 0$ (so that actual currents converge to the desired ones). Such an update law can be obtained using the following Lyapunov analysis. Let us choose Lyapunov function candidate as

$$H = \tfrac{1}{2}\alpha \tilde{i}^\top \boldsymbol{L}^2 \tilde{i} + \tfrac{1}{2}\tilde{\eta}^\top \tilde{\eta}$$

where α is a design parameter. The time derivative of this function along the trajectories determined by (9) is

$$\begin{aligned}\dot{H} = &-\alpha \tilde{i}^\top \boldsymbol{L}(\boldsymbol{R} + \rho \boldsymbol{I})\tilde{i} - \alpha\omega \tilde{i}^\top \boldsymbol{L}\boldsymbol{Y}\boldsymbol{L}\tilde{i} \\ &+ \alpha\omega \tilde{i}^\top \boldsymbol{L}\chi(\theta)\tilde{\eta} + \dot{\tilde{\eta}}^\top \tilde{\eta}. \end{aligned} \tag{10}$$

The second right-hand side term equals zero since it is a quadratic form involving a skew symmetric matrix. In an attempt to make $\dot{H}$ negative definite, we choose the adaptation law to cancel the third term on the right, i.e.

$$\dot{\hat{\eta}} \approx \dot{\tilde{\eta}} = -\alpha\omega \chi^\top(\theta)\boldsymbol{L}\tilde{i} \tag{11}$$

where $\dot{\hat{\eta}} \approx \dot{\tilde{\eta}}$ since $\eta_\star$ is constant or slowly varying. Substitution of (11) in (10) finally yields

$$\dot{H} = -\alpha \tilde{i}^\top \boldsymbol{L}(\boldsymbol{R} + \rho \boldsymbol{I})\tilde{i}.$$

Notice that $\dot{H}$ is only negative semidefinite, which assures that $\tilde{i} \to 0$, but not necessarily that $\tilde{\eta} \to 0$. Further analysis using LaSalle's invariance principle [14] is needed to prove that $\tilde{\eta} \to 0$ as well.[1] One of the important consequences of the invariance principle is that the origin (i.e. $\tilde{i} = 0, \tilde{\eta} = 0$) is asymptotically stable if $\dot{H} = 0, \ \forall t > t_0$ is achieved only with the trivial solution (i.e. $\tilde{i}(t) = 0, \tilde{\eta}(t) = 0, \ \forall t > t_0$). In our case, $\dot{H} \equiv 0 \Leftrightarrow \tilde{i} \equiv 0$, i.e. all solutions that render $\dot{H} = 0, \ \forall t > t_0$ have to satisfy $\tilde{i}(t) = 0, \ \forall t > t_0$. Now, going back to the system dynamics we have

$$\begin{aligned}&\tilde{i}(t) \equiv 0 \overset{(9)}{\Rightarrow} \omega\chi(\theta)\tilde{\eta}(t) \equiv 0 \\ &\tilde{i}(t) \equiv 0 \overset{(11)}{\Rightarrow} \tilde{\eta}(t) \equiv const. \end{aligned} \tag{12}$$

For $\omega \neq 0$, (12) is satisfied only for $\tilde{\eta}(t) \equiv 0$, which completes our proof. Therefore both $\hat{\eta} \to \eta_\star$ and $i \to i_\star$ after an initial perturbation. For $\omega \equiv 0$, no information about ripple parameters is available (since the back emf term is zero), and parameter estimation is not possible. The system is still stable with $\hat{\eta}(t) = \hat{\eta}(0) = const$ and $\tilde{i} \to 0$.

By changes in the design parameters α and ρ we can control the parameter convergence rate. An increase in ρ will result in a more aggressive control and a faster current convergence, and an increase in α will result in a more aggressive adaptation.

B. Speed Loop

The proposed control law, consisting of (7) and (8) with (11), enables tracking of a torque reference and adaptation to parameters that possibly shift with operating conditions. A common way to achieve speed servo operation is to design a controller that produces the torque reference (and its time derivative) from the difference between the desired and actual speed. This is achieved with the following second order linear controller

$$G_c(s) = K_c \cdot \frac{s + z_c}{s(s + p_c)}.$$

Notice that the control law (8) requires the derivative of the desired current vector $i_\star$, which in turn requires the derivative of the torque reference $\tau_\star$. The proposed form of the speed loop controller is then suitable since $\dot{\tau}_\star$ can be generated from the speed error ($\tilde{\omega}$) using the same controller without integral action (i.e., without the pole at the origin). This method is equivalent to generating $\tau_\star$ using a PI controller and a filtered $\tilde{\omega}$ signal, and $\dot{\tau}_\star$ using the same PI controller and an approximate derivative of the speed error [16], [17]. The parameter settings of the speed loop controller are chosen to achieve the desired performance in speed tracking.

TABLE II
R43H MOTOR PARAMETERS

Parameter	*Value (in $V \cdot s$)*
Φ_{d6}	0.0018
Φ_{d12}	0.0011
Φ_{q0}	0.1994
Φ_{q6}	0.0091
Φ_{q12}	0.0012

Parameter	*Value [Unit]*
P	2
L_d	9.1 $[mH]$
L_q	9.1 $[mH]$
R_s	1.45 $[\Omega]$
J	0.0011 $[kg \cdot m^2]$
B	0.0009 $[\frac{kg \cdot m^2}{s}]$

IV. SIMULATIONS AND IMPLEMENTATION ANALYSIS

After the theoretical derivation, the proposed controller was implemented and tested in simulations. An ideal scenario is simulated with results confirming the correctness of the design. In addition, some limitations imposed by the hardware implementation were discussed and their effects were simulated as well. All simulations were performed using the Matlab Simulink toolbox, with equations (1)–(4) used to model the motor behavior. Parameter values that were used are given in Table II. All of those values were either measured or taken from the nameplate of the motor used in the experimental setup.

A. Design Support

As a first step, the controller was simulated without taking into account any limiting effects that emerge from hardware implementation. All inputs and outputs of the controller are assumed continuous and feedback signals are assumed measurable without any errors. The results of these simulations are intended to support the theoretical derivations and to provide insight into the controller parameter selection.

Convergence of the parameter estimates for various values of the adaptation constant (α) is shown in Fig. 1.[2] Note that our

[1]Note that system (9), (11) is nonautonomous (time varying), and LaSalle's invariance theorem is not applicable to general nonautonomous systems. However, under the time separation assumption (i.e., the assumption $\omega = const$ during electrical transients) made in the beginning, (9), (11) represent a periodic system, which thus belongs to a special class of nonautonomous systems for which the invariance principle still holds (see [15, Subsection VIII.5.9]).

[2]Only the two largest parameters are shown to simplify the figure.

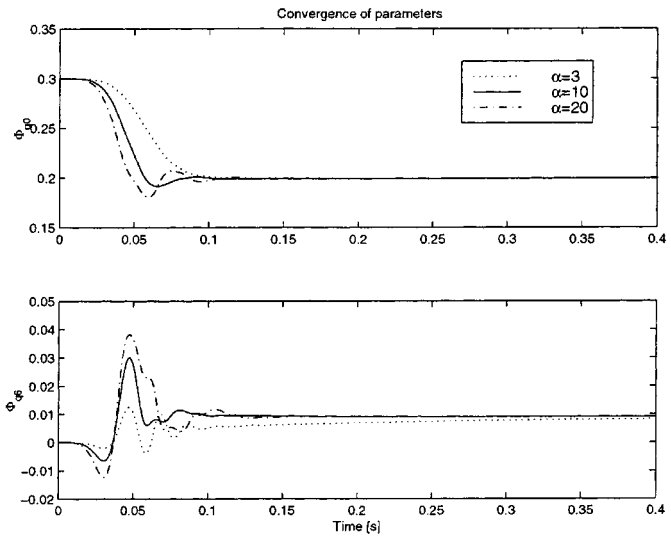


Fig. 1. Convergence of Φ_{q0} and Φ_{q6} parameter estimates for various choices of adaptation constant α.

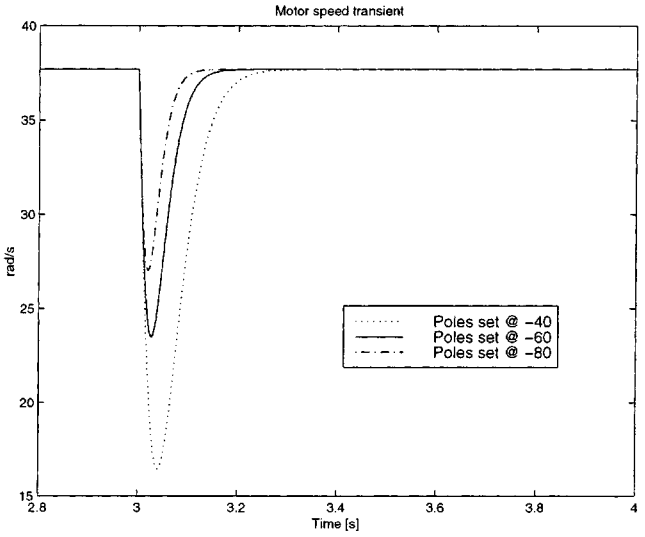


Fig. 2. Speed response to a step change in load torque at $t = 3s$. Performance is determined by the speed loop controller design.

adaptive controller starts with initial estimates that are 50% or more away from the "true" values. Nevertheless, we can see that parameters converge quickly to their true values. Control design parameters (α and ρ) are chosen based on a trade-off between speed and oscillations in parameter convergence. The larger the α is, the faster the convergence is, but excessive increase in this parameter leads to oscillations in the parameter response. The increase in ρ dampens the oscillations but requires higher controller gain and slows down the parameter convergence if α is kept small. With this trade-off, a good set of parameter values was found to be $\alpha = 10$ and $\rho = 0.1$.

Initial estimates can be taken to be zero for all parameters, except for Φ_{q0} because of the division in (7). Φ_{q0} is closely related to the back emf constant, which is usually given in motor data sheets with some tolerance. Thus, an initial estimate for this parameter close to its true value is often available. The initial estimates in our simulations are chosen as $\hat{\eta} = [0,\ 0,\ 0.3,\ 0,\ 0]^{\top}$.

A family of speed responses to a step change in the load torque is given in Fig. 2. The speed reference is taken to be $\omega_{\star} = 12\pi$ and the torque changes from 0 to 30% of its nominal value (1.1 Nm) at $t = 3$ s. These simulation results show that the speed tracking performance is completely determined by the choice of the speed loop controller parameters. The parameter choice is based on the closed loop system poles assignment.[3]

Finally, the simulated steady state motor torque spectrum with and without adaptation is shown in Fig. 3. As we can see, torque ripple in simulations is completely eliminated with our adaptive controller.[4]

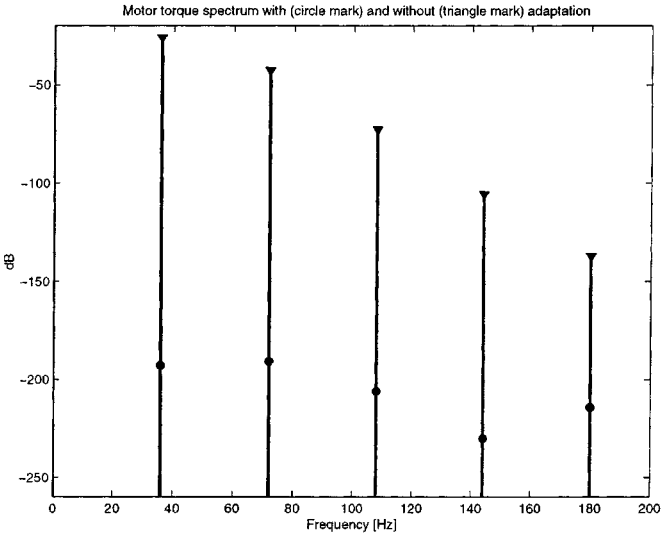


Fig. 3. Simulated steady state motor torque spectrum ($f_{rot} = 3$ Hz and $\tau_l = 1.1$ Nm) without adaptation (triangular mark), and with adaptation (circular mark).

B. Hardware Implementation Limitations

The simulations in the previous subsection give insight in the controller performance and confirm the correctness of the design, but they neglect several implementation issues. There are certain details in hardware implementation that limit the torque ripple minimization level achievable by controller action. At high motor speeds the main problems become controller discretization, sampling of the inputs, and holding of the outputs. On the other hand, at very low motor speeds the resolution of the position sensor as well as the speed feedback calculation become primary factors that restrict controller performance. The effects of these issues will be discussed in this section.

Let f_s be the controller sampling frequency (i.e. the rate of input signals sampling and output signals updating). With this rate of input sampling, the largest recoverable harmonic in the input signal is at $f_s/2$ (Nyquist frequency). This imposes an upper limit on the mechanical frequency of the motor rotation (f_{rot}) because the harmonics of all the signals in the motor reside at multiples of that frequency. Since the largest harmonic we need to recover is at $12Pf_{rot}$, the theoretical maximum for f_{rot} at which we can still expect the ripple to be minimized is

[3]For this purpose current loop controller is assumed infinitely fast. All poles are (arbitrarily) chosen to be equal.

[4]The residual ripple below -190 dB is due to numerical errors in simulation.

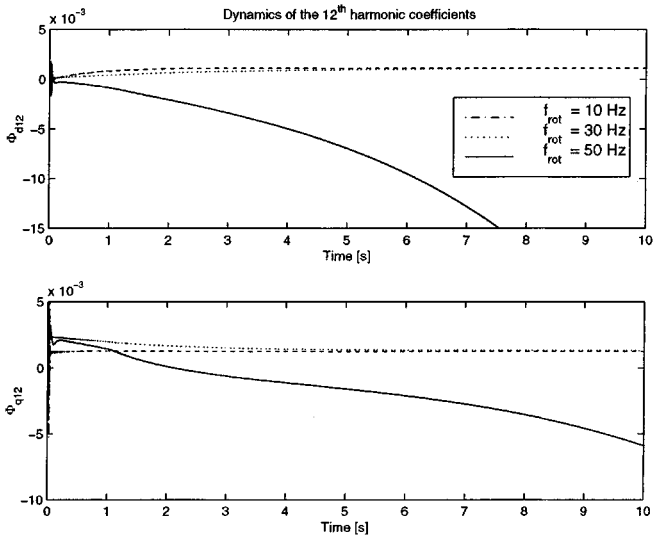


Fig. 4. Convergence of the 12th harmonic coefficients (Φ_{d12} and Φ_{q12}) for different motor speeds ($f_s = 2$ kHz). The parameter divergence occurs when frequency of the 12th harmonic exceeds the Nyquist frequency.

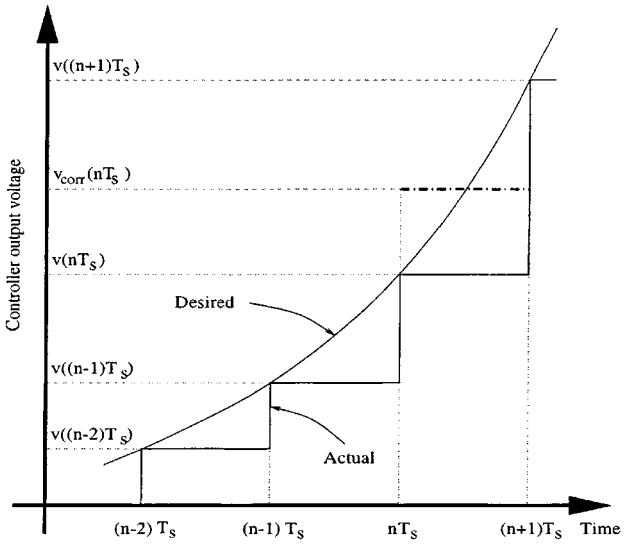


Fig. 5. An illustration of the desired (continuous) and actual (stepwise) voltage output.

$f_{rot_{\max}} = f_s/(2 \cdot 12P)$.[5] Simulations of this effect of input sampling, with results in Fig. 4, indeed show that when the maximum motor speed is exceeded the corresponding parameters (Φ_{d12} and Φ_{q12}) start to diverge.

In addition to the input sampling, the output signals are calculated at the same rate (f_s) and held constant over one sampling period. This results in a stepwise output voltage instead of a continuous one. This problem is depicted in the Fig. 5. In every sampling interval the actual voltage supplied to the motor differs from the desired one (the actual is smaller for positive voltage slope and vice versa). The error in voltage increases when the voltage slope increases (i.e. when the motor speed increases), and at some point the difference in voltage becomes large enough to obstruct the parameter adaptation. Therefore,

[5]As an example, in our hardware setup $f_s = 2$ kHz $\Rightarrow$ $f_{rot_{\max}} = 41.67$ Hz.

TABLE III
EFFECTS OF SAMPLING FREQUENCY ON TORQUE RIPPLE PERFORMANCE ($f_{rot} = 2$ Hz)

Sampling frequency [kHz]	*Torque harmonic value [dB]*	
	6^{th}	12^{th}
1	-48.28	-53.72
2	-54.41	-60.35
10	-68.54	-74.96
20	-74.57	-81.06

this phenomenon puts another upper limit on the motor speed that is even more restrictive than the previous one. This limit, as the one due to the input sampling, depends on the sampling frequency. Clearly, the torque ripple minimization performance can be improved by increase in f_s, and this improvement is quantified in Table III. However, by making certain corrections to the controller output voltages, we can improve performance even without changing the sampling frequency. Based on the previous values of the output voltage, we can predict the voltage trajectory in the next sampling interval and try to approximate it. The prediction is easily made by fitting a polynomial (we have chosen a second order one) through the known points. Since the voltage output can be only constant in between two sampling instances, we can select that constant value such that the integrals (voltage seconds) of the actual and desired voltages over the next sampling interval are equal. After some straightforward calculations the following corrected voltage output signal can be derived:

$$v_{corr}(nT_s) = \tfrac{5}{12}v((n-2)T_s) - \tfrac{4}{3}v((n-1)T_s) + \tfrac{23}{12}v(nT_s).$$

After the application of this correction, the upper speed limit becomes approximately twice the pre-corrected value.[6] In addition, the torque ripple minimization performance at speeds below the limit is improved, as can be seen from the torque spectra in Fig. 6.

We should also mention that controller discretization alone (i.e., discrete implementation of the continuous controller equations with inputs and outputs assumed continuous) has negligible effects on torque ripple minimization performance. Simulations show that even the simple first-order Euler integration method yields excellent results (minimization down to residual -190 dB as in the continuous case).

To summarize, adaptation of ripple parameters at high motor speeds, requires a high sampling frequency. In practice, the sampling frequency certainly has an upper limit (due to A/D conversion of the current signals, speed feedback calculation, and controller equations calculation), which in turn imposes an upper limit on the motor speed at which the adaptation is reliable.

Different problems become dominant at low speeds. The position measurement with incremental encoder has finite resolution determined by the number of encoder pulses per revolution. Since the calculation of the controller voltages includes information about the rotor position, stepwise position information

[6]For the sampling frequency of $f_s = 2$ kHz divergence of parameters occurs at about $f_{rot} = 5$ Hz without the correction and at about $f_{rot} = 10$ Hz with the correction.

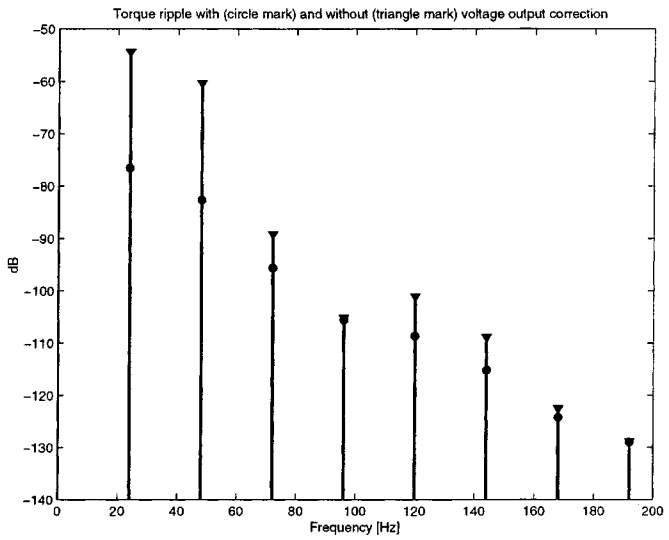


Fig. 6. Comparison between torque spectra with (circular mark) and without (triangular mark) voltage output correction $f_s = 2kHz, f_{rot} = 2Hz, and \tau_l = 1.1Nm)$.

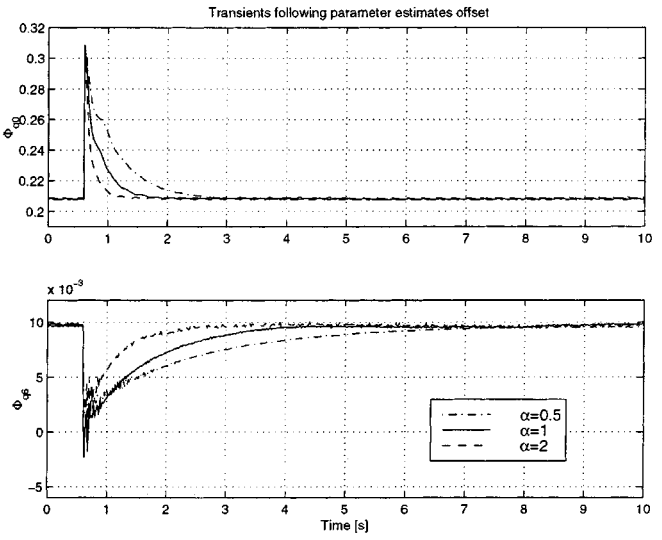


Fig. 8. Experimental transients of Φ_{q0} and Φ_{q6} parameter estimates following their offset to the initial values.

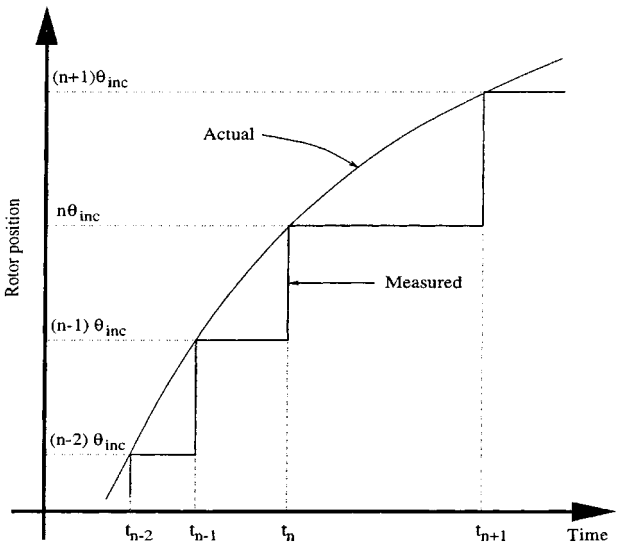


Fig. 7. Illustration of the actual (continuous) and measured (stepwise) position. Encoder pulses occur at times t_i and each pulse changes the position measurement for a fixed increment θ_{inc}.

leads to a problem that is a counterpart to the holding of the output voltages discussed above. The difference is that now the deviation of the actual voltage seconds from the desired ones is larger for lower speeds. Indeed, we can see from Fig. 7 that the height of the position measurement (and thus voltage) step is always constant, but the width is the time between two encoder pulses which is inversely proportional to the position slope (i.e. to the motor speed). However, the position quantization is not a major problem and this phenomenon might deteriorate the performance only at extremely low speeds.[7]

The other issue—speed measurement—is much more restrictive at low motor speeds than the position quantization. At low speeds, the motor speed is calculated as the scaled frequency of incoming encoder pulses (rather than as the number of encoder pulses in a fixed time period). The time between two previous encoder pulses is measured and inverted, and that information is used until the next encoder pulse arrives. Thus the estimated motor speed is an *average* of the actual speed over the *previous* interval between two encoder pulses. As the motor speed decreases, the introduced delay in the speed measurement increases, causing deterioration in torque ripple minimization performance. To compensate for this variable delay, a prediction of the future values of the speed measurement based on several past values is employed. This extrapolated value is then used in feedback, thus reducing the delay and improving the performance. When a second order polynomial is used, the following corrected speed feedback signal can be derived[8]

$$\omega_{corr}(n\theta_{inc}) = \omega_{ms}((n-2)\theta_{inc}) - 3\omega_{ms}((n-1)\theta_{inc}) + 3\omega_{ms}(n\theta_{inc})$$

where ω_{ms} is the speed measurement derived from the frequency of incoming encoder pulses and θ_{inc} is the increment in position corresponding to one encoder pulse. Simulations show that the above correction of the speed measurement greatly improves the performance at low speeds. For example, with inclusion of the practical speed measurement in simulations (in addition to position quantization) and without speed feedback correction, at $f_{rot} = 0.1$ Hz torque ripple harmonics increase up to -55 dB, but fall back below -100 dB after the correction for the delay in the speed measurement.

In summary, practical limitations impose both upper and lower speed limits outside of which parameter estimates start drifting from their actual values. Limits can be extended with an additional effort in improving measurements and outputs, but can not be eliminated completely.

[7]With inclusion of position quantization in simulations, at $f_{rot} = 0.1$ Hz torque ripple harmonics are still below -100 dB.

[8]Notice that the speed measurement updates on every encoder pulse.

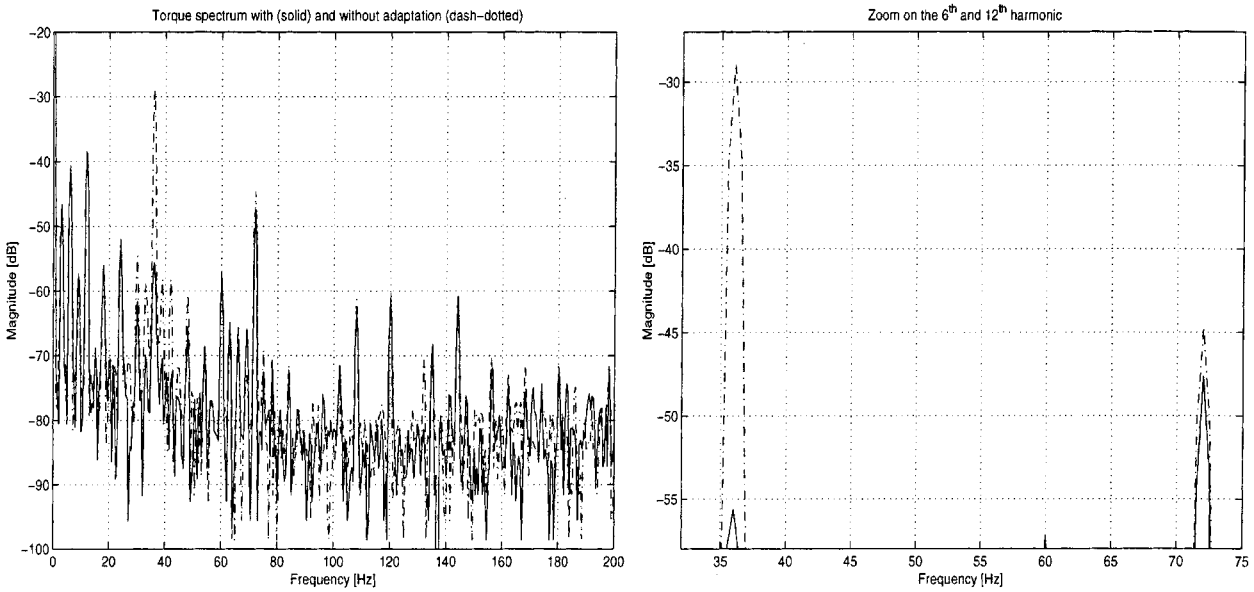


Fig. 9. Experimental steady state torque spectrum ($f_{rot} = 3$ Hz and $\tau_l = 1.1$ Nm) with adaptation (solid line), and without adaptation (dashed line). In the right panel expanded region around the 6th and 12th harmonic.

V. Experiments and Practical Issues

To support simulations, laboratory tests for the same motor were performed. A universal test bed built around a DS1102 controller board (made by dSPACE GmbH) is used for this purpose. The experimental setup consists of the following components: a three-phase, Y-connected PMSM (Pacific Scientific model R43H), DS1102 controller DSP board, interface electronics (including circuitry for inverter dead-time generation, optical insulation of low and high voltage modules, etc.), three-phase voltage source inverter (consisting of six IGBTs, integrated in Motorola MHPM7A15A60A hybrid power module, driven by PWM signals from DS1102 through the IRF2110 gate drivers) supplied from the 60 V programmable power supply, torquemeter and hysteretic load cell mechanically coupled with PMSM, Hall effect current sensors, and an incremental encoder.

Rotor position is measured with two-channel quadrature-output incremental encoder with index pulse and line drivers. Each channel produces 1024 pulses per revolution, and the use of the quadrature decoder/counter (contained in DS1102) results in position resolution of $2\pi/(4 \cdot 1024) \approx 1.5 \cdot 10^{-3}$ rad. The controller board main processor (TMS320C31, floating-point) is used for calculation of the controller equations. A simple first-order Euler approximation with sampling time of $T_s = 500$ μs was used for the controller discretization. The on-board slave processor (TMS320P14, fixed-point) performs lower level functions such as PWM generation and speed calculation from position feedback. Both tasks utilize slave processor's event manager that consists of compare and capture subsystems. The compare system automatically compares the timer and appropriately scaled signal values, thus giving PWM output. For our experiments a 20 kHz PWM was used. Capture system is used for frequency measurement of incremental encoder pulses, thus enabling very accurate measurement of motor rotational speed (especially at low speeds when pulses are scarce). An upper limit on measurable speed exists due to a finite DSP clock frequency, but that limit is well above the operating speeds for the used motor and setup.[9] DS1102 also contains an analog to digital conversion subsystem with two 16-bit and two 12-bit A/D converters. The 16-bit autocalibrating A/D converters are used in discretization of two phase current feedback signals from the Hall effect sensors (the third phase current does not need to be measured since the motor is connected in Y).

To evaluate transient behavior of the adaptation, parameter estimates were first offset on purpose from correct values they converged to, and then convergence back to correct values is observed and recorded. The experiment was performed for various values of adaptation constant α, and typical transients of parameter estimates are shown in Fig. 8 (the error in estimates is artificially introduced at $t = 0.6$ s). Adaptation constants used in hardware implementation are slightly smaller compared to the ones used in simulations in order to limit the influence of the experimental noise on the adaptation.

Fig. 9 shows comparison between torque spectra before and after adaptation. As in simulations, applied load torque is $\tau_l = 1.1$ Nm and motor rotates with constant mechanical frequency of $f_{rot} = 3$ Hz. The results show considerable reduction of 6th (by 27 dB) and 12th (by 4 dB) torque harmonics. It is worth mentioning at this point that the motor under test, although with skewed magnets, develops residual cogging torque at multiples of $12f_e$ ($N_{slpp} = 12$ since stator has 24 slots) whose magnitude is small and can be seen in Fig. 9 after the minimization of the mutual torque ripple. The motor torque waveform given in Fig. 10 also shows improvement in torque ripple (before the adaptation $\delta\tau_{rms} = 0.035$ Nm and after the parameters converge $\delta\tau_{rms} = 0.011$ Nm, where $\delta\tau_{rms}$ is the RMS value of the

[9]In the used configuration maximum measurable frequency of incoming signal is 60 kHz. Given the number of encoder pulses per revolution (1024), the resulting electrical speed range is $\omega \in (0,\ 736.31)$ rad/s.

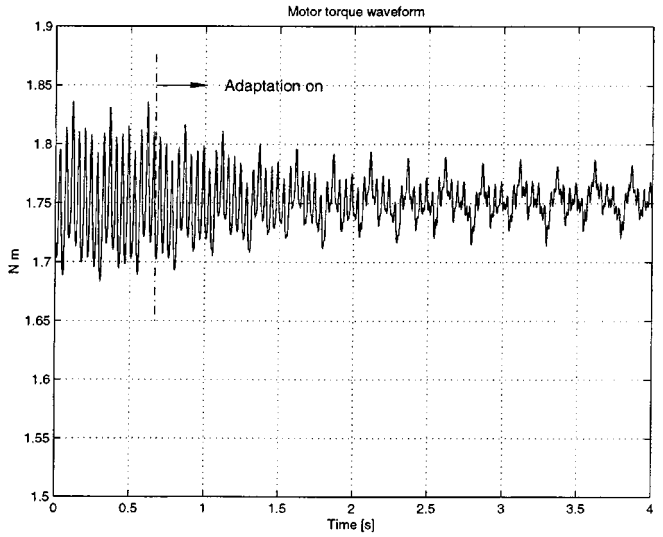


Fig. 10. Motor torque waveform ($f_{rot} = 2$ Hz, $\tau_l = 1.75$ Nm, and $\alpha = 1$) showing ripple minimization after the beginning of adaptation at $t = 0.7$ s.

torque ripple). In the hardware implementation, however, more harmonics appear due to mismatches with modeling assumptions. For example, the inequality in strength of motor poles results in additional harmonic at mechanical frequency of rotation (which is a sub-harmonic of electrical frequency of rotation f_e). Residual DC offsets in phase current measurements result in the first harmonic of f_e. An unbalance of motor phases and phase voltages imposed by hardware, as well as unsymmetry of motor poles, results in the second harmonic of f_e. The most significant source of residual ripple is mechanical in nature. Laboratory test stand requires mechanical coupling of the motor, torquemeter, and load cell, and bearings, joints, and eccentricities in the system introduce mechanical oscillations on every revolution.[10] After the minimization of torque ripple components at multiples of $6f_e$, these additional harmonics actually become dominant and require further attention, such as more emphasis on test stand design.

There are several other issues in hardware implementation that contribute to the dq current (and thus torque) harmonic components at multiples of $6f_e$. For example, switch voltage drop and inverter dead-time impose additional voltage and inject dc and higher harmonics into the system. Those additional voltages can be significant in the case of low inverter supply voltage, which is the case in our experimental setup. A proper compensation (e.g., [18]) of mentioned phenomena should be incorporated into the system to ensure successful implementation of the controller.

VI. Conclusions

A versatile PMSM model for torque ripple analysis has been developed and tested in simulations and experiments. An adaptive controller was then designed to minimize torque pulsations. Simulations of the continuous controller model show a complete elimination of torque ripple indicating a proper controller design. Further simulations including phenomena introduced by hardware implementation show that in practice ripple minimization performs well only in a certain range of motor speeds. The main issues that introduce an upper speed limit were found to be the sampling of the controller inputs and holding of the outputs. The lower speed limit mainly occurs due to increased delay in speed measurement. Several approaches for extending those limits are also described in the paper. Other hardware issues, like dead-time and switch voltage drop compensation, have to be solved for successful controller implementation. Nevertheless, the final results in torque ripple reduction achieved in experiments are promising.

[10]Note that such complex mechanical system is not present in direct drive applications, resulting in much less pronounced mechanical oscillations.

References

[1] 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, pp. 321–330, Apr. 1996.

[2] T. Li and G. Slemon, "Reduction of cogging torque in permanent magnet motors," *IEEE Trans. Magn.*, vol. 24, no. 6, pp. 2901–2903, 1988.

[3] A. Kaddouri and H. Le-Huy, "Analysis and design of a slotless NdFeB permanent magnet synchronous motors for direct drive," in *Proc. Rec. IEEE IAS Annu. Meeting*, 1992, pp. 271–278.

[4] C. Studer, A. Keyhani, T. Sebastian, and S. K. Murthy, "Study of cogging torque in permanent magnet machines," in *Proc. Rec. IEEE IAS Annu. Meeting*, New Orleans, LA, October 1997, pp. 42–49.

[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, pp. 260–266, 1993.

[6] D. C. Hanselman, "Minimum torque ripple, maximum efficiency excitation of brushless permanent magnet motors," *IEEE Trans. Ind. Electron.*, vol. 41, pp. 292–300, June 1994.

[7] J. Holz and L. Springob, "Identification and compensation of torque ripple in high-precision permanent magnet motor drives," *IEEE Trans. Ind. Electron.*, vol. 43, pp. 309–320, Apr. 1996.

[8] F. Colamartino, C. Marchand, and A. Razek, "Considerations of nonsinusoidal field distribution in a permanent magnet synchronous motor control," in *Proc. 5th Int. Conf. Power Electron. Variable Speed Drives*, 1994, pp. 508–513.

[9] T. S. Low, T. H. Lee, K. J. Tseng, and K. S. Lock, "Servo performance of a BLDC drive with instantaneous torque control," *IEEE Trans. Ind. Applicat.*, vol. 28, pp. 455–462, Apr. 1992.

[10] S. K. Chung, H. S. Kim, C. G. Kim, and M. J. Youn, "A new instantaneous torque control of PM synchronous motor for high-performance direct-drive applications," *IEEE Trans. Power Electron.*, vol. 13, pp. 388–400, May 1998.

[11] 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, pp. 231–237.

[12] N. Matsui, T. Makino, and H. Satoh, "Autocompensation of torque ripple of direct drive motor by torque Observer," *IEEE Trans. Ind. Applicat.*, vol. 29, pp. 187–194, Feb. 1993.

[13] P. J. Nicklasson, R. Ortega, and G. Espinosa-Pérez, "Passivity-based control of a class of Blondel–Park transformable electric machines," *IEEE Trans. Automat. Contr.*, vol. 42, no. 5, pp. 629–647, 1997.

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

[15] N. Rouche, P. Habets, and M. Laloy, *Stability Theory by Lyapunov's Direct Method*. New York: Springer-Verlag, 1977.

[16] R. Ortega, P. J. Nicklasson, and G. Espinosa-Perez, "On speed control of induction motors," *Automatica*, vol. 32, pp. 455–460, Mar. 1996.

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

[18] D. Leggate and R. J. Kerkman, "Pulse-based dead-time compensator for PWM voltage inverters," *IEEE Trans. Ind. Electron.*, vol. 44, pp. 191–197, Apr. 1997.

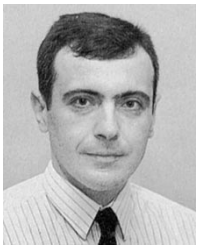
Vladan Petrović was born in Novi Sad, Yugoslavia in 1972. He received the Dipl.Ing. degree in electrical engineering from the University of Novi Sad, Yugoslavia in 1996, and the M.S. degree in electrical engineering from Northeastern University, Boston, MA, in 1998, where he is currently pursuing the Ph.D. degree.

He is a Research Assistant with the Department of Electrical and Computer Engineering, Northeastern University. His research interests include motion control, power electronics, and nonlinear control.

Romeo Ortega was born in Mexico. He received the B.Sc. degree in electrical and mechanical engineering from the National University of Mexico, the M.S. degree in engineering from Polytechnic Institute of Leningrad, Leningrad, Russia, and the Docteur D'etat degree from the Polytechnic Institute of Grenoble, Grenoble, France, in 1974, 1978, and 1984, respectively.

He then joined the National University of Mexico, where he worked until 1989. He was a Visiting Professor at the University of Illinois, Urbana, from 1987 to 1988, and at tMcGill University, Montreal, P.Q., Canada, from 1991 to 1992, and a Fellow of the Japan Society for Promotion of Science (1990–1991). From June 1992 to December 1996, he worked in the Université de Technologie de Compiégne. Currently, he is in the Laboratorie de Signaux, Systémes, Paris, France, where he holds a CNRS position. His research interests are in the fields of nonlinear and adaptive control. He is an Associate Editor of *Systems and Control Letters*, *International Journal of Adaptive Control and Signal Processing*, and the *European Journal of Control*. He is currently the Coordinator of the Nonlinear and Adaptive Control European Network and an INTAS network with countries from the former Soviet Union.

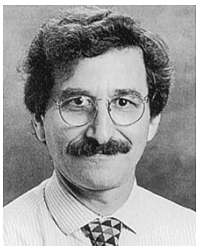
Dr. Ortega was the Chairman of the IEEE Working Group on Adaptive Control and Systems Identification and is currently Chairman of the IFAC Technical Committee on Adaptive Control and Tuning, and Chairman of the *Automatica* Paper Prize Award Committee. He is Associate Editor of the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY.



Aleksandar M. Stanković (M'86) received the Dipl.Ing. and M.S. degrees from the University of Belgrade, Yugoslavia, in 1982 and 1986, respectively, and the Ph.D. degree from the Massachusetts Institute of Technology, Cambridge, in 1993, all in electrical engineering.

He has been with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, since 1993, presently as an Associate Professor. His research interests are in modeling, analysis, estimation and control of energy processing systems.

Dr. Stankoviæ is a member of the IEEE Power Engineering, Power Electronics, Control Systems, Circuits and Systems, Industry Applications, and Industrial Electronics Societies. He serves as an Associate Editor for the IEEE TRANSACTIONS FOR CONTROL SYSTEM TECHNOLOGY.



Gilead Tadmor received the B.Sc. degree from Tel Aviv University, Tel Aviv, Israel, in 1977, and the M.Sc. and Ph.D. degrees (with honors) from the Weizmann Institute of Science, Israel, in 1979 and 1984, respectively, all in mathematics.

He joined the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, in 1989, where he is currently an Associate Professor. Previously he held research and faculty positions also at Tel Aviv University, Brown University, Providence, RI, the University of Texas, Dallas, and the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge. From 1998 to 1999, he visited SatCon Technology Co., Cambridge, MA, and is currently a Consultant at United Technologies Research Center, East Hartford, CT. His background is in the areas of robust and optimal control, distributed parameter systems, and systems theory. His recent active interests include nonlinear control, with applications in power electronics, power systems, electric motor drives, jet engines, and mechanical systems. His research has been supported by NSF, ARO, SatCon Technology Co., and the Charles Stark Draper Laboratory.