A Fractional Delay Variable Frequency Repetitive Control for Torque Ripple Reduction in PMSMs

Mi Tang
Member, IEEE
University of Nottingham
University Park
Nottingham, NG7 2RD, UK
mi.tang2@nottingham.ac.uk

Alberto Gaeta
Member, IEEE
reDrives S.r.l
Via Tommaso Fazello 5
Lentini (SR), Italy
alberto.gaeta@redrives.eu

Andrea Formentini
Member, IEEE
University of Nottingham
University Park
Nottingham, NG7 2RD, UK

Pericle Zanchetta
Senior Member, IEEE
University of Nottingham
University Park
Nottingham, NG7 2RD, UK

Abstract--**Based on the internal model principle, repetitive controller (RC) is capable to reduce periodic torque ripple by generating a compensating action that consequently need to be synchronized with the original ripple. However, the synchronization is difficult to achieve using the conventional RC when the sampling frequency is not integer multiple of the speed (known as fractional delay issue), or when the speed varies widely. To solve this problem, this paper presents a fractional delay variable frequency torque ripple reduction method for PMSM drives using the combination of angle-based RC and deadbeat current control (DBCC). Four aspects of innovations are included in the proposed control to improve the synchronization. The experimental results show that the proposed control can effectively reduce torque ripple even during speed and load transient.**

Index Terms--**Deadbeat control, Repetitive control, Variable speed drives, Permanent magnet machines, Torque ripple reduction**

Introduction

Torque ripple reduction in permanent magnet synchronous machines (PMSMs) has been studied for decades. Using traditional vector control and being the output torque of the machine a mathematic function of current, many of the existing torque ripple suppression methods reduce torque ripple by introducing tailored compensating periodic current ripple [1-6]. To generate the ripple-cancelling current references, torque sensor is used in [1, 2] to acquire a direct measurement of torque ripple, which increases the cost; [3-5] calculate the current references according to pre-known back-EMF function; [6] build a lookup table which is pre-computed from the indirect measurement of cogging torque. It is to be noted that the values of these periodic current references relate not only to the rotor position, but also to the level of saturation and parameter variations of the machine. Also some unexpected disturbances, like the unbalance on the mechanical shaft, cannot be predicted. It is therefore difficult, time and resource consuming to derive the correct back-EMF function or lookup tables for different working conditions. In either case, the bandwidth of the current loop should be high enough to track such current references which may contain high frequency components. The deadbeat current control (DBCC) [7] is therefore preferred for its intrinsic high bandwidth.

Besides, the repetitive controller (RC) can be a solution for generating periodic current references, tracking periodic signals or rejecting periodic disturbances, so it is a perfect candidate for torque ripple reduction since torque ripple is mainly periodic [8]. For example, RC can work in conjunction with normal PI controllers in the current loop to remove errors between periodic current references and current feedback [9]. In such way, a lookup table would be required for generating the periodic current references which cannot be generated from the speed loop due to the limitation of its bandwidth. Alternatively, the RC can be placed in the speed loop where the controller can use the speed error [10, 11] or the torque error [8, 12] as an input producing periodic current references. Among these two methods, the latter one may be preferred to avoid the computation of current lookup tables. Due to its learning capability, RC is effective on reducing periodic torque ripple without necessarily knowing the machine parameters, no matter the ripple is caused by flux harmonics, cogging torque, current offset, current scaling error, mechanical unbalance, or inverter non-linearities [13].

The two paragraphs above imply the possibility of using RC (for generating current reference) combined with DBCC (for tracking the reference). However, the conventional RC is effective only at constant speed. For the torque ripple reduction in variable speed PMSM drives, modifications have to be done in the design of RC. The design of the conventional RC requires pre-knowledge of the fundamental frequency of the periodic torque ripple. RC can therefore memorize the ripple by delaying the ripple signal for the amount of time equals to its period. Through memorizing the ripple for a whole period, the controller can therefore compute the current required for canceling the ripple at any time during one period.

To maximize the ripple suppression, the controller needs to record the ripple waveform for exactly one period, which is difficult in the digital control implementation since the controller can only delay the signal for integer multiple of the sampling period. In order to meet potential fractional delay requirement (i.e. fractional delay issue), a fractional delay RC based on Lagrange interpolations is presented for active filter [14] and inverter [15-18].

Another issue for using RC in variable speed drives is that the fundamental frequency of torque ripple varies with speed and cannot be known in advance. So every time the speed is changed, the periodic compensating current reference stored in the controller memory must be cleared and a new learning process must be initiated or alternatively the memory content must be processed in

order to adapt it to the new operating speed. In order to solve this variable frequency issue, the variable sampling frequency RC is presented in [19, 20] for converters. For PMSM drive, since the torque ripple is in fact periodic with respect to rotor position, [11] presents an angle-based RC which works effectively within a limited variable speed range of 50rpm to 70rpm.

In the angle domain, the frequency of the previously described torque disturbance is fixed and the controller memory does not need to be resized, cleared or processed when the speed is changed; the phase of torque disturbance due to cogging torque and flux harmonics within a full electrical rotation is marginally affected by the speed and by saturation conditions. Therefore, the RC should learn only the amplitude variation of the torque disturbance as a consequence of a speed/load change and parameter variation.

Followed by the previous work in [21], this paper proposes a fractional delay variable frequency RC for torque ripple reduction in PMSM drives using the angle-based RC and DBCC including four innovative aspects: 1) the compensation of DBCC loop delay in the angle-based RC, using position prediction; 2) the compensation of mechanical system delay by torque estimation; 3) the implementation of fractional delay for the angle-based RC; 4) the rejection of oscillations during speed/load transients by using a transient detector. Both simulation and experimental tests show that the proposed strategy significantly improves the effectiveness of RC when the sampling frequency is not integer multiple of the speed; widely expands the frequency range of torque ripple suppression; and can even work during transients without necessarily reset the memory.

This paper is organized as follows. System models equations and diagrams of the proposed control are explained in detail in section II. Simulation results are shown in section III. Experimental results are shown in section IV. The results and advantages of the proposed control are discussed in section V. Finally, conclusions of the improvements introduced by the proposed approach are given in section VI.

I. MODEL AND EQUATIONS

The overall structure of the proposed system is depicted in Fig. 1, where the details of the proposed repetitive control are shown in Fig. 2. For a surface mounted PMSM (Fig. 1(a)), the output torque is considered proportional to only the q-axis current, while the d-axis current is set to zero. The reverse of torque constant $1/K_t$ can be used for converting the torque reference to the current reference. However, it is not compulsory, since this gain can also be included in the gain of PI and RC. This gain block is removed in this paper.

It should be noted that the proposed RC can be easily adapted also when an interior permanent (Fig. 1(b)) is used. In this case the RC provides a torque-ripple compensating reference which should be added to load-compensating torque reference generated by the speed-loop's PI regulator.
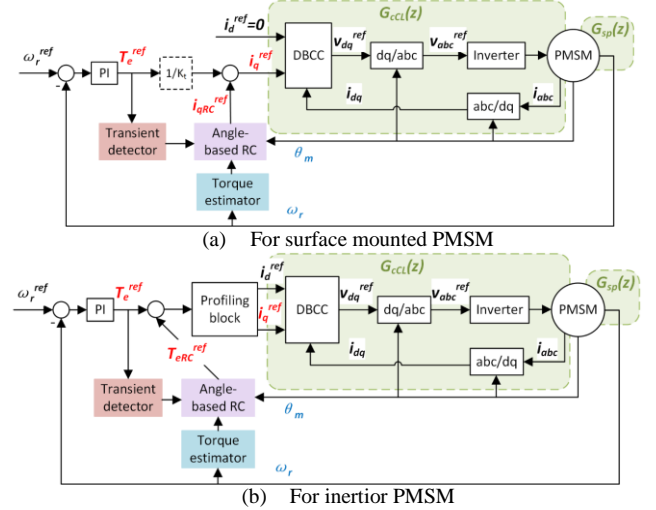

(a)   For surface mounted PMSM

(b)   For inertior PMSM

Fig. 1.   Block diagram of the proposed angle-based repetitive control with deadbeat current control

(a)   Details of the whole angle-based repetitive control (during the kth sampling interval)
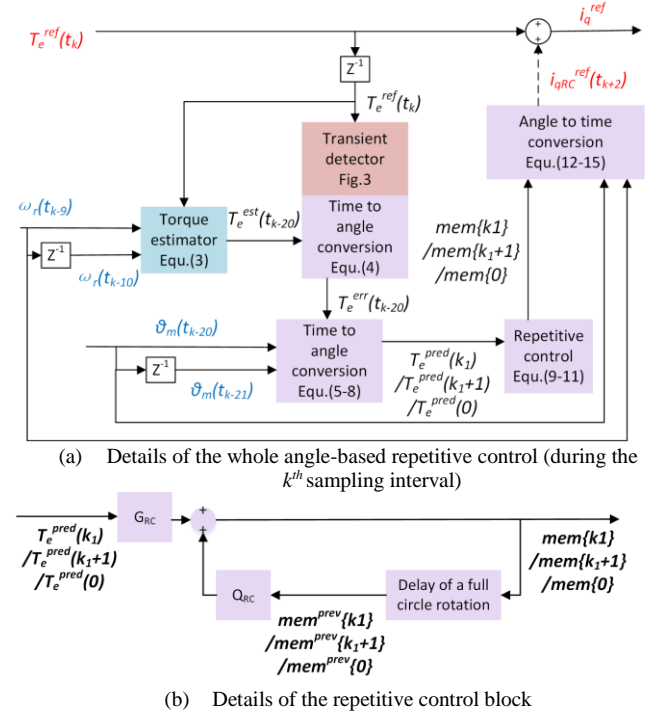

(b)   Details of the repetitive control block

Fig. 2.   Block diagram of the angle-based repetitive control

Obviously, a current profiling block which converts the overall torque reference to a *dq* current reference is required in this case.

A. Deadbeat Current Control Loop

The deadbeat current control loop has been implemented as described in [22]. In closed loop, it can be expressed by a delay of two sampling periods as in (1) when operating within the voltage limitation determined by the inverter DC bus voltage.

$$G_{cCL}(z) = z^{-2} \tag{1}$$

B. Torque Estimator

The discretized mechanical plant of the motor is described as (2):

$$G_{sp}(z) = \frac{T_s}{Jz - (J - BT_s)} = \frac{\omega_r(z)}{T_e(z) - T_l} \quad (2)$$

Where, T_s is the sampling period, J is the motor and load inertia, B is friction factor, and T_l is the load torque.

In general, for control purposes, the mechanical parameters need to be estimated (i.e. inertia J^{est} and friction factor B^{est}). The performance of the proposed control scheme with wrong J^{est} and B^{est} will be investigated in Section II). Moreover, the output torque needs also to be estimated since used as an input for the RC. As may have been noticed from Fig.1, the torque estimation requires the speed feedback. Practically, if the feedback speed used for torque estimation is acquired directly from the measured position, the torque estimation can be very noisy. To overcome the above problem, in this paper, a 9th order finite impulse response (FIR) filter (with cut-off frequency equal to $0.21/T_s$) is used for calculating speed from the measured position; an additional 10th order FIR filter is used for estimating torque from the speed. As shown in the phase diagram of the 9th order FIR filter (Fig. 3), the phase delay increases linearly with frequency in the pass-band. This indicates the nature of a FIR filter that it would delay all the frequencies in pass-band for a fixed amount of time.

The delay time introduced by such filters must be taken into consideration in the implementation of the RC otherwise the proper synchronization of the torque-ripple compensating signal is lost. Due to the $9T_s$ delay caused by the 9th order FIR filter, the speeds used for torque estimation and available during the kth sampling period are $\omega_r(t_{k-9})$ and $\omega_r(t_{k-10})$. Consequently, the output torque is estimated based on the reverse of (2) as follow:

$$
\begin{aligned}
&T_e^{est}(t_{k-10}) \\
&= \frac{J^{est}[\omega_r(t_{k-9}) - \omega_r(t_{k-10})]}{T_s} + B^{est}\omega_r(t_{k-10}) + T_l^{est}(t_{k-10})
\end{aligned}
\quad (3)
$$

$T_e^{est}(t_{k-10})$ is the estimated output torque at t_{k-10}. $\omega_r(t_{k-9})$ and $\omega_r(t_k)$ are the rotor angular speed at t_{k-9} and t_{k-10} respectively. $T_l^{est}(t_{k-10})$ is the estimated load torque at t_{k-10}, which can be approximated using the value of the torque reference (i.e. output of the speed loop PI). This approximation is clearly valid only at steady state and not during speed or load transients; however as will be described in the following, the estimated output torque is not used during such transients since the RC is disabled to prevent the learning of inexistent torque ripple.

Moreover, due to the additional $10T_s$ delay of the aforementioned 10th order FIR filter, the output of torque estimator during the kth period is the torque at t_{k-20} (i.e. $T_e^{est}(t_{k-20})$).

C. Time to Angle Conversion

During the kth sampling interval, after the torque estimation, $T_e^{est}(t_{k-20})$, $T_e^{ref}(t_{k-20})$, and $\theta_m(t_{k-20})$ (the rotor mechanical position at t_{k-20}) are all inputted to the repetitive controller.

The torque error T_e^{err} at t_{k-20} is calculated as in (4):

$$T_e^{err}(t_{k-20}) = T_e^{ref}(t_{k-20}) - T_e^{est}(t_{k-20}) \quad (4)$$
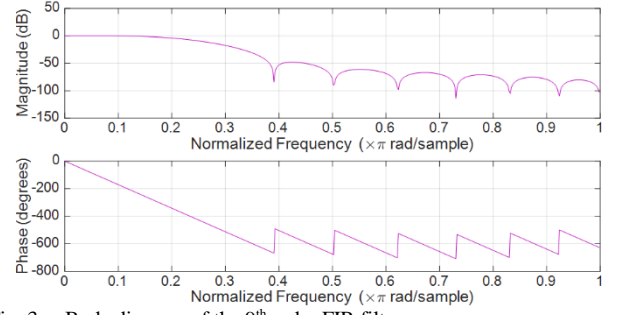

Fig. 3. Bode diagram of the 9th order FIR filter

Again, $T_e^{est}(t_{k-20})$ is the estimated torque at t_{k-20} whereas $T_e^{ref}(t_{k-20})$ is the torque reference generated by the speed loop PI controller at t_{k-20}.

The time to angle conversion is then executed. An array of length N, *mem*, is defined in the memory. The N values of the array *mem* are updated by the torque errors at only N chosen rotor angles out of the whole mechanical resolution of 2π, and the N values indicate the q-axis currents required for torque ripple reduction at the N chosen rotor locations. For example, *mem[i]*, the $(i+1)^{th}$ value in *mem*, (i is within [0,N-1]), is the q-axis current i_{qRC} required for reducing torque error when the rotor mechanical position is exactly $i*2\pi/N$. The rotor location index k_1 is defined as in (5)

$$k_1 = floor\left(\frac{N \cdot \theta_m(t_{k-20})}{2\pi}\right), k_1 \in [0, N-1] \quad (5)$$

Where, $\theta_m(t_{k-20})$ is the rotor mechanical position at t_{k-20} and the value of N is chosen to be about the ratio between the sampling frequency f_s and the rated speed ω_r of the machine in Hz.

Since the rotor position cannot always be measured exactly at points coincident with integers multiple of $2\pi/N$, interpolations are necessarily used to predict the torque error at the closest rotor location (θ_m=0, $2\pi/N$, $4\pi/N$,..., 2π). Linear interpolations are executed as in (6)(7)(8) only if the value of k_1 is different from the one obtained at the previous sampling period.

In the case of positive rotation:

$$
\begin{aligned}
T_e^{pred}(k_1) &= T_e^{err}(t_{k-21}) \\
&+ \frac{T_e^{err}(t_{k-20}) - T_e^{err}(t_{k-21})}{\theta_m(t_{k-20}) - \theta_m(t_{k-21})} \cdot [\theta_m(k_1) - \theta_m(t_{k-21})]
\end{aligned}
\quad (6)
$$

In the case of negative rotation and $k_1 \neq (N-1)$:

$$
\begin{aligned}
T_e^{pred}(k_1+1) &= T_e^{err}(t_{k-21}) \\
&+ \frac{T_e^{err}(t_{k-20}) - T_e^{err}(t_{k-21})}{\theta_m(t_{k-20}) - \theta_m(t_{k-21})} \cdot [\theta_m(k_1+1) - \theta_m(t_{k-21})]
\end{aligned}
\quad (7)
$$

In the case of negative rotation and $k_1 = (N-1)$:

$$
\begin{aligned}
T_e^{pred}(0) &= T_e^{err}(t_{k-21}) \\
&+ \frac{T_e^{err}(t_{k-20}) - T_e^{err}(t_{k-21})}{\theta_m(t_{k-20}) - \theta_m(t_{k-21})} \cdot [0 - \theta_m(t_{k-21})]
\end{aligned}
\quad (8)
$$

Where, $T_e^{pred}(k_1)$ is the torque error if the rotor mechanical angular position equals to $\theta_m(k_1)=k_1(2\pi)/N$, $T_e^{err}(t_{k-20})$ and $T_e^{err}(t_{k-21})$ are the torque errors at t_{k-20} and t_{k-21}, $\theta_m(t_{k-20})$ and $\theta_m(t_{k-21})$ are the rotor mechanical angular position at t_{k-20} and t_{k-21}. When the direction of the rotor movement is opposite to the positive direction of the encoder, thus negative rotation, $T_e^{pred}(k_1+1)$ is calculated instead of $T_e^{pred}(k_1)$. This is because the interpolation is more accurate estimating the torque error at $\theta_m(k_1+1)=2\pi(k_1+1)/N$, which is in between the position $\theta_m(t_{k-20})$ and its previous position $\theta_m(t_{k-21})$, than estimating the torque error at $\theta_m(k_1)=k_1(2\pi)/N$ which has not yet been reached at t_{k-20}.

D. Repetitive Control

After converting the torque error at the time t_{k-20} into the torque error at position $\theta_m(k_1)$ or $\theta_m(k_1+1)$ or $\theta_m(0)$, the memory array can now be updated according to the repetitive control equations (9)(10)(11) (Fig.2(b)). Again, only when the rotor location index k_1 is different from the one at the previous sampling period, the memory array would need to be updated.

For positive rotation:

$$mem\{k_1\} = mem^{prev}\{k_1\}\cdot Q_{RC} + T_e^{pred}(k_1)\cdot G_{RC} \qquad (9)$$

For negative rotation and $k_1\neq(N-1)$:

$$mem\{k_1+1\} = mem^{prev}\{k_1+1\}\cdot Q_{RC} + T_e^{pred}(k_1+1)\cdot G_{RC} \quad (10)$$

For negative rotation and $k_1=(N-1)$:

$$mem\{0\} = mem^{prev}\{0\}\cdot Q_{RC} + T_e^{pred}(0)\cdot G_{RC} \qquad (11)$$

Where, Q_{RC} is the forgetting factor of RC, G_{RC} is the gain of RC, $mem\{k_1\}$ is the $(k_1+1)^{th}$ value in the memory array mem, which indicates q-axis current required for compensating the torque ripple at rotor position $k_1*2\pi/N$, $mem^{prev}\{k_1\}$ is the previous value of $mem\{k_1\}$ obtained at the time of one full circle rotation ago.

E. Angle to Time Conversion

As shown in Fig.2(a), the output of the repetitive controller after the angle to time conversion acts as an additional q-axis current reference, i_{qRC}^{ref}. According to the principle of DBCC [22], the current reference calculated during the kth period, can only be applied at t_{k+1} and achieved at t_{k+2}. The future rotor position at t_{k+2} need to be predicted and converted to be within $[0, 2\pi]$ as in (12), where the speed is assumed to be constant.

$$\theta_m^{fut}(t_{k+2}) = \text{mod}\left[\left\{\theta_m(t_k)+\omega_r(t_k)*2T_s\right\},2\pi\right] \quad (12)$$

A future rotor location index k_2 is defined as in (13):

$$k_2 = floor\left(\frac{N\cdot\theta_m^{fut}(t_{k+2})}{2\pi}\right), k_2\in[0,N-1] \qquad (13)$$

To compensate the current delay of $2T_s$ (between the

start of calculation at t_k and its response at t_{k+2}), the q-axis current reference generated by RC i_{qRC}^{ref}, should be chosen from the memory array according to the future rotor mechanical position at t_{k+2}, $\theta_m(t_{k+2})$. Again, linear interpolations are used for estimating this according to $mem\{k_2\}$ and $mem\{k_2+1\}$, as in (14)(15):

When $k_2\neq(N-1)$:

$$i_{qRC}^{ref}(t_{k+2}) = mem\{k_2\} \qquad (14)$$
$$+\frac{N\cdot(mem\{k_2+1\}-mem\{k_2\})}{2\pi}\cdot\left[\theta_m^{fut}(t_{k+2})-\frac{k_2\cdot2\pi}{N}\right]$$

When $k_2=(N-1)$:

$$i_{qRC}^{ref}(t_{k+2}) = mem\{k_2\}$$
$$+\frac{N\cdot(mem\{0\}-mem\{k_2\})}{2\pi}\cdot\left[\theta_m^{fut}(t_{k+2})-\frac{k_2\cdot2\pi}{N}\right] \quad (15)$$

F. Transient Detector

The angle-based RC allows maintaining torque ripple compensation even after speed and load changes. In fact, the learning process, after a transient, only needs to be updated with the small amplitude changes in the torque ripple due to the changed conditions (saturation of the machine for example) since the phase of the torque ripple is related only to the rotor position. However, the learning process has to be temporarily disabled during each transient, to prevent the RC from outputting an excessive compensating reference current as a consequence of the non-existent large torque ripple estimated during each sudden change in speed.

In fact, as explained in Section I-B, the reference torque generated by the speed loop PI regulator has been assumed equal to the load torque; such assumption is obviously verified only during steady-state conditions. As a result, during transient conditions, the acceleration torque is seen as a disturbance ripple which is erroneously "learned" by the RC so worsening the torque ripple reduction for a limited amount of time after each transient since such false ripple is progressively "forgotten" thanks to self-learning capabilities of RC. Therefore, a transient detection routine is designed to detect speed/load changes, and disable the learning process accordingly. A flow chat of the proposed transient detector is drawn in Fig. 4.

In details, a speed transient will be detected once the change of the torque reference T_e^{ref} generated by the speed loop PI controller is greater than a chosen threshold ΔT_e^{max}. A speed transient signal S_{speed} will be set to be zero once a speed transient is detected. Similarly, a load change will be detected (a load transient signal S_{load} will be set to zero) if the difference between the actual torque reference and its value a certain time ago (chosen to be $30T_s$), is larger than the same threshold ΔT_e^{max}. The steady state is recognized only if both S_{speed} and S_{load} have been kept one for a time longer than t^{sta}. Consequently, an enable signal S^{ena} will be set to one if steady state is recognized. And when $S^{ena}=0$, torque error T_e^{err} as defined in (4) is set to zero to block the

learning procedure.

It should be pointed out that the proposed transient detection strategy has been chosen for its combined simplicity, as obvious from Fig.4, and efficiency as will be shown in the simulation and experimental results. Although different detection strategies based on a more accurate analysis of the torque reference signal could be used (change point analysis for example), the delay introduced as a consequence of the increased complexity and of the increased number of samples required, would vanish the final effect of the transient detector since the RC would be still able to detect and erroneously compensate part of the false ripple.

II. SIMULATION RESULTS

A PMSM model has been built in MATLAB/Simulink for simulating the effectiveness of the proposed method. It is worth noting that, in order to accurately simulate the non-sinusoidal flux distribution and cogging torque of PMSM, two lookup tables, derived from a finite element (FE) model of the same motor, are used for the magnetic flux, and for the cogging torque. The machine parameters and its back-EMF waveform can be seen in Table I and Fig.5(a), respectively. Parameters for RC are as in Table II.

The resulting torque and speed waveforms with/without interpolation in the time to angle conversion are shown in Fig.6, where the speed reference set to be 490 rpm. In which case, the sampling frequency 10kHz is not integer multiple (i.e.10k*60/490=1224.49) of the fundamental frequency of speed.

The torque and speed waveforms with/without the proposed transient detection strategy under speed/load changes are shown in Fig.7, where a speed change from 501 to 999 rpm is applied at 1s and a load change from 5 to 8 Nm is applied at 2s.
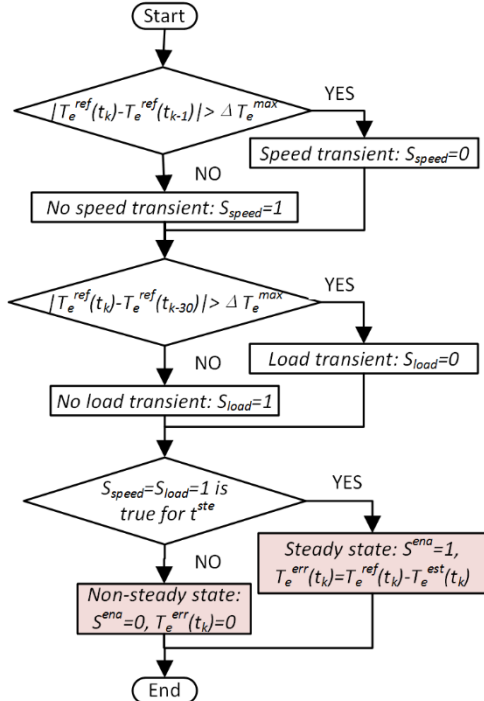

Fig. 4.   Flow chat of the proposed transient detector

Fig.6 and 7 confirm the normal operation of the proposed RC. It is worth to investigate now the performance when the mechanical parameters J^{est} and B^{est} used for the proposed torque estimator are wrong. Fig. 8 shows the torque waveform when J^{est} varies from 10% to 200% of the real J, and B^{est} varies from 10% to 1000% of the real B. As it can be seen, the only condition that degrades the performance of torque ripple reduction is when $J^{est}<J$. Moreover, the best THD seems to be achieved when $J^{est}=2J$ (purple in Fig.8(a) and (b)). However, greater transient spikes are present under such condition.

To better understand this behaviour in Fig.8, the bode diagram of the transfer function $G_{te}(z)G_{sp}(z)$ is plotted in Fig.9. $G_{te}(z)$ is the transfer function of the proposed torque estimator. Neglecting the delays of the FIR filter, it can be represented as in (16). $G_{sp}(z)$ is the mechanical plant as in (2).
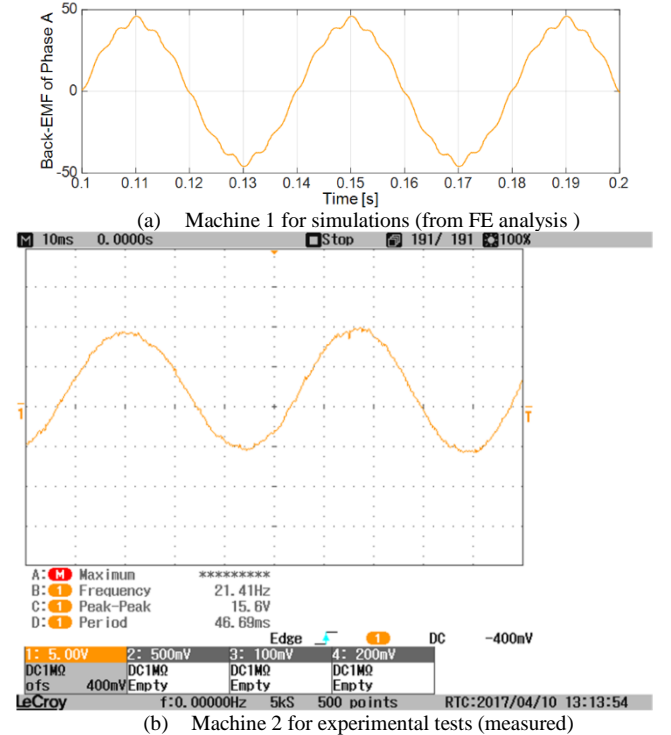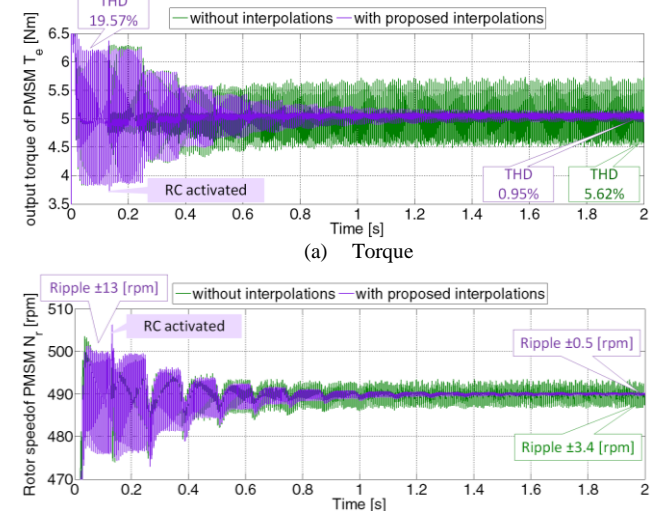

(a)   Machine 1 for simulations (from FE analysis)

(b)   Machine 2 for experimental tests (measured)

Fig. 5.   Back-EMF waveform of the machine used in simulation and experimental tests

(a)   Torque

(a) Speed during a speed transient at 1s

(b) Speed during a load transient at 2s

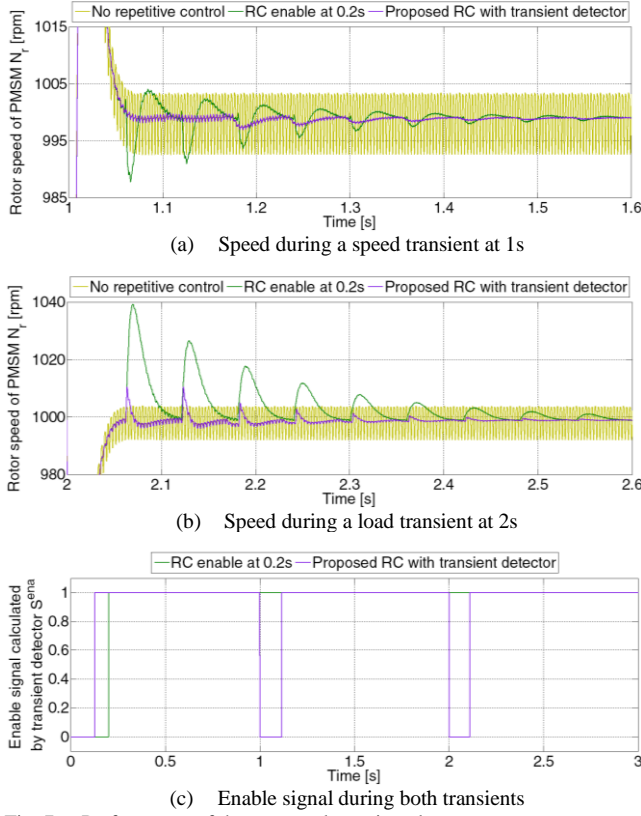(c) Enable signal during both transients

Fig. 7. Performance of the proposed transient detector

$$G_{te}(z) = \frac{J^{est}z - \left(J - B^{est}T_s\right)}{T_s z} \quad (16)$$

Ideally, when $J^{est}=J$ and $B^{est}=B$, $G_{te}(z)G_{sp}(z)=1/z$. As shown in Fig.9, for low frequencies (<100Hz), both magnitude and phase would be affected by the detuned J^{est} and B^{est}. However, such low frequency torque ripple is likely to be within the speed loop bandwidth. Hence, the speed loop can work to reduce such low frequency torque/speed ripple. For frequencies higher than 100Hz, when $J^{est}<J$, the amplitude is attenuated. This indicates that under such condition, the proposed torque estimator will "see" the torque ripple much smaller than the actual value. Consequently, torque ripple cannot be reduced effectively as shown in Fig.8(a).

III. EXPERIMENTAL RESULTS

An experimental rig has been set up to verify performance of the proposed control method in case of fractional delay and during speed transient. As shown in Fig.10, the rig comprises PMSM with a 17 bits encoder, a two level converter, sensors for measuring three-phase currents and DC bus voltage, FPGA /DSP board for reading the measurements and generating gate signals for the converter, a load motor and a load drive. The machine parameters and control parameters can be seen in Table I and II respectively. The back-emf waveform of the machine can be seen from Fig.5(b).

The three-phase currents are converted into dq-axis currents in the DSP. Through properly tuning the deadbeat current controller, Fig.11(c) shows a successfully implementation of the DBCC in experiments, where, the q-
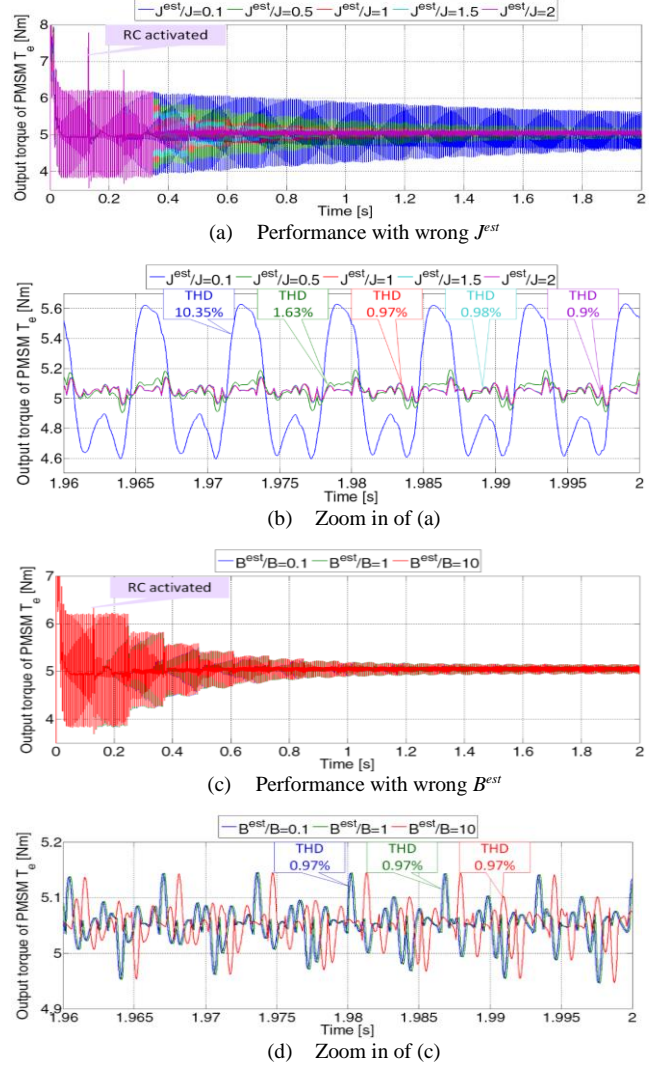

(a) Performance with wrong J^{est}

(b) Zoom in of (a)

(c) Performance with wrong B^{est}

(d) Zoom in of (c)

Fig. 8. Performance of the proposed RC with wrong mechanical parameters used in the proposed torque estimator ($f_s=10kHz$, $\omega_r=500rpm$)

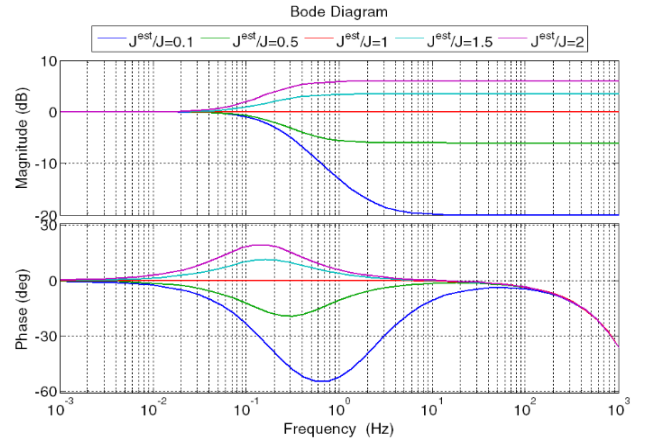Fig. 9. Bode diagram of $G_{te}(z)G_{sp}(z)$

axis current is able to reach the reference after exactly $2T_s$ (=200μs). The test condition in Fig.11 will be introduced later.

After tuning the current loop, the proposed RC has been implemented. Comparing Fig.11(a) and (b), it can be seen that more ripple is present in the current reference with the proposed RC. This confirms the fact that the proposed RC works to reduce torque ripple by inducing current ripple.
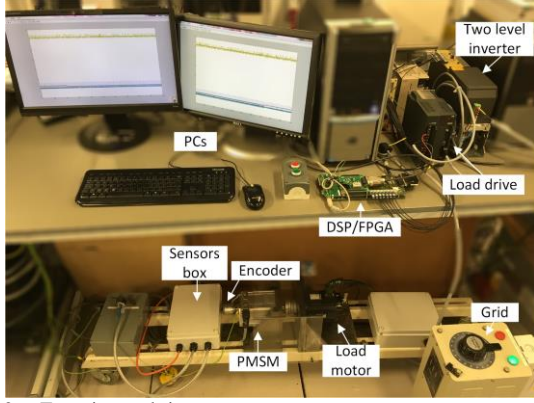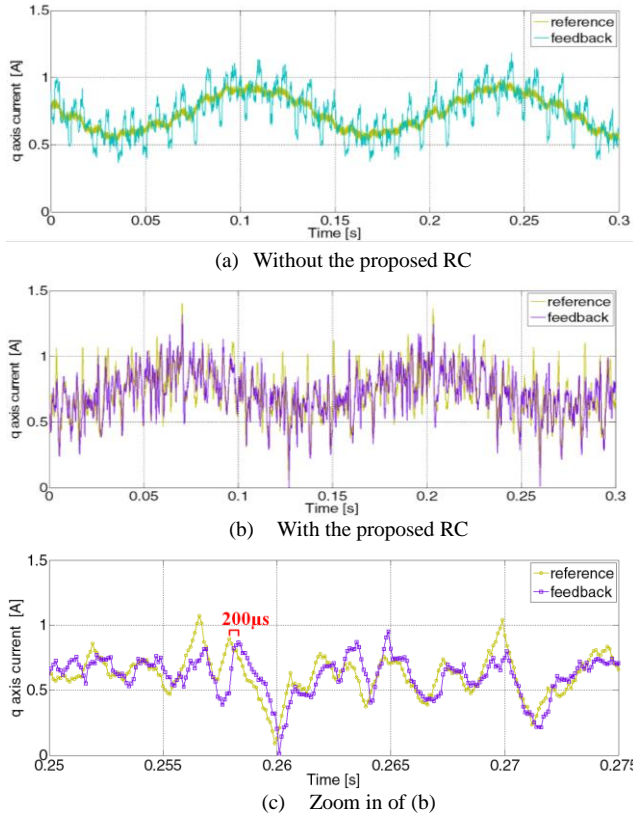
Fig. 10. Experimental rig

(a) Without the proposed RC

(b) With the proposed RC

(c) Zoom in of (b)

Fig. 11. Performance of the DBCC at 451 rpm

(a) Without RC

(b) With RC

Fig. 12. FFT of speed waveforms with/without the proposed RC in experiment (integer delay condition)

(a) Without RC

(b) With RC

Fig. 13. FFT of speed waveforms with/without the proposed RC in experiment (fractional delay condition)

The performance of the proposed RC is verified from the three tests as described below:

1) *Variable Frequency Performance Test:*
   The aim of this test is to proof the effectiveness of the proposed RC under a widely range of frequencies or speeds. The Fast Fourier Transformation (FFT) of speed waveforms with/without RC at a range of speed from 80rpm to 500rpm are compared as shown in Fig.12. It is to be noted that the x-axis in Fig.12 is the harmonic order with respect to the fundamental harmonic to show the fact that the frequency of torque ripple is proportional to speed. During the test, the PMSM is controlled under speed control mode.

2) *Fractional Delay Performance Test:*
   The same tests as in 1) are repeated for a range of speed from 123 to 451 rpm (fractional delay conditions). The FFT of the speed waveforms are shown in Fig.13.
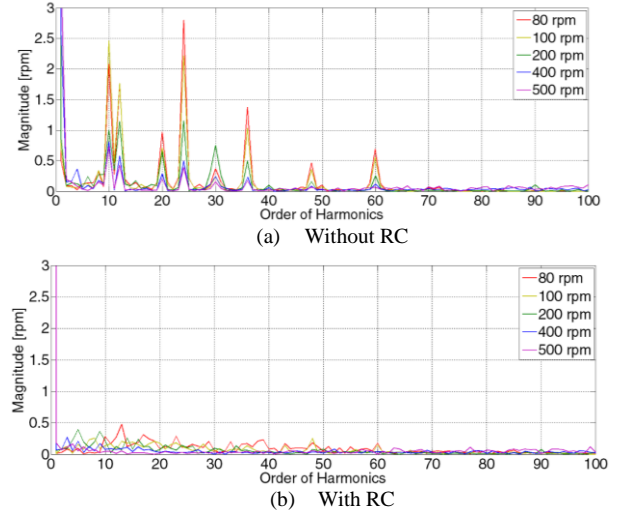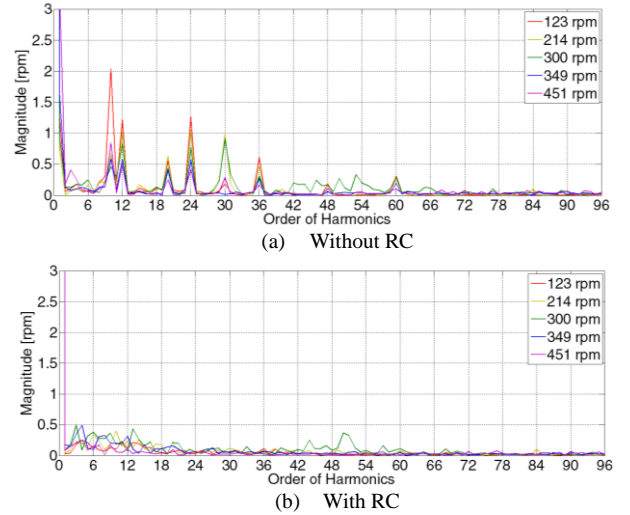
Specially, the current response at 451 rpm with the proposed RC is shown in Fig.11.

3) *Transient Performance Test:*
   The aim of this test is to investigate the performance of the proposed RC during speed/load transients. A speed step is applied from 123 to 415 rpm. A load step is applied from 5 to 8 Nm using the load drive. The speed waveforms with/without RC during such step changes are compared as shown in Fig.14 and Fig.15.

IV. DISCUSSION

In general, the effectiveness of RC depends on whether the repetitive controller can learn or predict the torque ripple at different rotor locations precisely, and whether the compensation torque can be synthesized at exactly the desired rotor positions.

The results in Fig.12(a) and Fig.13(a) confirm the feasibility of learning and compensating the torque ripple according to the rotor position. Therefore, the proposed RC can work for a widely range of speed/frequencies from 80 to 500 rpm without necessarily change any settings of the repetitive controller.

Summarizing, besides of using the angle-based RC instead of the traditional time-based RC, additional four innovation aspects of the proposed control method have been proposed in this paper and are further discussed in the following based on the results in part III and IV.

A. Implementation of a fractional phase lead compensator in angle-domain

The proposed angle to time conversion have two steps: 1) use (12-13) to predict the future position considering $2T_s$ delay of the DBCC loop in normal operations [22] (no physical limits are hit, as shown in Fig.11); 2) use (14-15) to interpolate the compensation current corresponding to the future position. In fact, such implementation is equivalent to a phase lead compensator Z^M, which is a commonly used type of filter in a time-domain RC. Traditionally, M has to be an integer, however, thanks to the interpolation, the proposed angle to time conversion can be seen as a phase lead compensator with M variable and possibly a fractional number.

B. Compensation of the Mechanical System Delays by Torque Estimation

The low-pass filter effect of the mechanical system causes a phase shift between the rotor speed and the output torque of the machine. For a low-pass filter, high frequency signals are delayed more than the low frequency signals, so synchronization of the output of RC to target ripple reduction would be more challenging at high speed if the feedback speed were used as the input for the RC. Since the torque is made up of components at different frequencies, the delay would be very difficult to be compensated in RC. Alternatively, in the proposed control topology, torque is used in place of speed and the delay introduced by the mechanical system is compensated by the torque estimator as in(3).

C. Improvement of RC Learning Accuracy for Fractional Delay

RC compensates the torque ripple based on the torque error it learns during previous operations. Therefore, the more accurate the torque errors are learned or recorded, the better the performance. Due to the floor function (as in (5)), without the proposed interpolations, the compensation current saved in each memory location k_1 (or the torque error T_e^{err} used for calculating it) corresponds to a rotor position in the range $[2\pi k_1/N, 2\pi(k_1+1)/N]$. With the worst case happening, when the sampling frequency and the rotor speed are not synchronized and the measured rotor position slowly shifts with time, the RC will generate a wrong compensation action. As shown in Fig.6, when the sampling frequency is not integer multiple of the motor speed, the performance of RC is improved simply with interpolations. There is no need to enlarge the memory or increase the sampling frequency.

The experimental results in Fig.12 and Fig.13 also support the effectiveness of the proposed RC for fractional delay, since comparable performance is achieved for the integer delay cases (80, 100, 200, 400, 500 rpm) and the fractional delay cases (123, 214, 300, 349, 451 rpm).
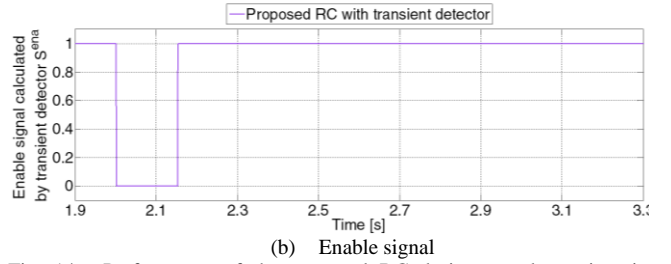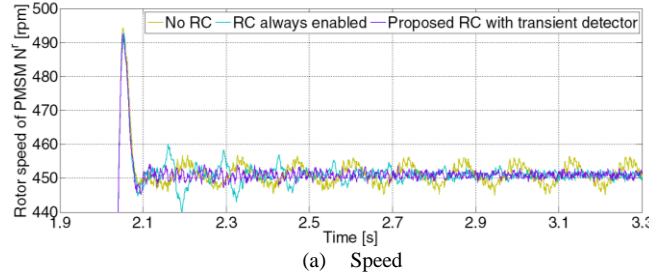

(a)   Speed

(b)   Enable signal

Fig. 14.   Performance of the proposed RC during speed transient in experiment (speed steps from 123 to 451rpm at 2s)

(a)   Speed
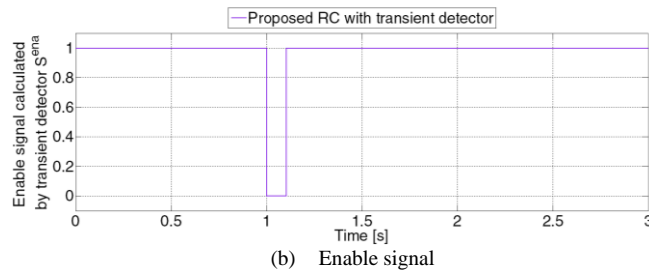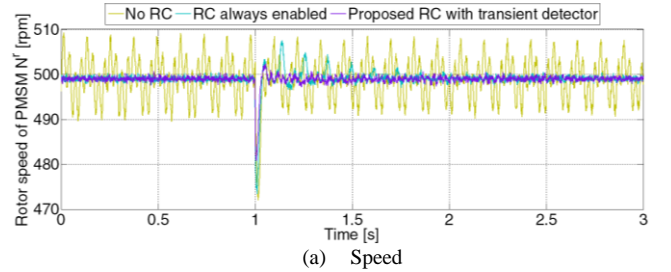

(b)   Enable signal

Fig. 15.   Performance of the proposed RC during speed transient in experiment (Load steps from 5 to 8 Nm at 1s)

D. Rejection of Oscillations during Speed Transient

In order to minimize interferences between the speed PI controller and the RC and minimize convergence time, the latter is connected in a plug-in configuration in the proposed control topology instead of a parallel implementation as it was done in [8, 10, 12].

However, still, the interaction between speed PI and RC may cause oscillations during speed/load transients as shown in Fig.7 (green), where RC is kept on after being activated at 0.2s. Meanwhile, by properly disable only the learning process, the purple curve proves that the proposed transient detection strategy can reduce the oscillations while maintaining the reduction of torque ripple during the speed transients.

Similarly, the experimental results in Fig.14 and Fig.15 show smoother transient behaviour and faster convergence is achieved with the proposed transient detector.

V. Conclusion

A fractional delay variable frequency repetitive control is presented in this paper, using the combination of the angle-based RC and deadbeat current control. The DBCC, angle-based RC, torque estimator and the topology are chosen considering the compensation of current control delays and the low-pass filter effect of mechanical system. Interpolations algorithms and a transient detection strategy

TABLE I
MACHINE PARAMETERS FOR TESTS

Quantity	Machine 1 for simulations	Machine 2 for experiments
Number of pole pairs P_p	3	5
Stator resistance R_s	1.4 Ω	0.5 Ω
d axis inductance L_d	0.0048 H	0.0009 H
q axis inductance L_q	0.0071 H	0.0012 H
Magnetic flux F_m	0.2691~0.2732 Wb (vary with position)	0.0566 Wb
Inertia J	0.00078 kg/m^2	0.0009 kg/m^2
Friction factor B	0.001 Nms/rad	0.004 Nms/rad

TABLE II
REPETITIVE CONTROLLER PARAMETERS FOR TESTS

Symbol	Quantity	Value
ω_r	Rated speed of motor	3000 rpm (=50 Hz)
f_s	Sampling frequency	10 kHz
N	Length of memory array	200 (=10k/50)
T_s	Sampling period	100μs
G_{RC}	Gain of RC	0.3
Q_{RC}	Forgetting factor of RC	0.999
ΔT_e^{max}	Threshold for transient detector	0.4 Nm
t^{sta}	Minimum stable time for transient detector	0.1 s (=1000T_s)

are proposed to improve the accuracy of RC. The effectiveness of the proposed control topology is proven by both simulation and experimental tests. The results show that the proposed control can reduced torque ripple not only for a widely range of frequency, but also when sampling frequency is not integral multiple of the rotor speed and even during speed/load transients.

REFERENCES

[1] M. Carpita, D. Colombo, and A. Monti, "A Generalised Approach to Torque Ripple Compensation In Permanent Magnet Machine Control," in *Industrial Electronics, 2000. ISIE 2000. Proceedings of the 2000 IEEE International Symposium on*, Cholula, Puebla 2000 pp. 396 - 400 vol.2

[2] B. H. Lam, S. K. Panda, and J. X. Xu, "Torque Ripple Minimization in PM Synchronous Motors - An Iterative Learning Control Approach," in *Power Electronics and Drive Systems, 1999. PEDS '99. Proceedings of the IEEE 1999 International Conference on*, 1999, pp. 144 - 149.

[3] B.-J. Brunsbach, G. Henneberger, and T. Klepsch, "Compensation of torque ripple," in *Electrical Machines and Drives, 1993. Sixth International Conference on (Conf. Publ. No. 376)* Oxford 8-10 Sep 1993 pp. 588 - 593

[4] E. Favre, L. Cardoletti, and M. Jufer, "Permanent-magnet synchronous motors: a comprehensive approach to cogging torque suppression," *Industry Applications, IEEE Transactions on,* vol. 29, pp. 1141 - 1149 Nov/Dec 1993

[5] L. Guo and L. Parsa, "Torque ripple reduction of the modular Interior Permanent Magnet machines using optimum current

profiling technique," in *Electric Machines and Drives Conference, 2009. IEMDC '09. IEEE International* Miami, FL 3-6 May 2009 pp. 1094 - 1099

[6] J. Holtz and L. Springob, "Identification and compensation of torque ripple in high-precision permanent magnet motor drives," *Industrial Electronics, IEEE Transactions oon,* vol. 43, pp. 309 - 320, Apr 1996.

[7] L. Springob and J. Holtz, "High-Bandwidth Current Control for Torque-Ripple Compensation in PM Synchronous Machines," *Industrial Electronics, IEEE Transactions on,* vol. 45, pp. 713 - 721, Oct 1998.

[8] W. Qian, S. K. Panda, and J.-X. Xu. (2004, March) Torque Ripple Minimization in PM Synchronous Motors Using Iterative Learning Control. *Power Electronics, IEEE Transactions on*. 272 - 279.

[9] P. Mattavelli, L. Tubiana, and M. Zigliotto, "Torque-ripple reduction in PM synchronous motor drives using repetitive current control," *Power Electronics, IEEE Transactions on* vol. 20, pp. 1423- 1431 2005.

[10] L. Wenshan, Z. Jian, and L. Yong, "A simpler and more efficient iterative learning controller for PMSM torque ripple reduction," in *Electrical Machines and Systems (ICEMS), 2013 International Conference on*, 2013, pp. 1231 - 1235.

[11] Y. Yuan, F. Auger, L. Loron, S. Moisy, and M. Hubert, "Torque Ripple Reduction in Permanent Magnet Synchronous Machines Using Angle-Based Iterative Learning Control," in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, 25-28 Oct. 2012, pp. 2518 - 2523.

[12] Z. Enrang and Q. Wei, "A ILC scheme and passive filter for PMSM speed control to minimize pulating torque," in *Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on*, Chengdu, 2010, pp. V5-492 - V5-496.

[13] B. Lazhar, "On the compensation of dead time and zero-current crossing for a PWM-inverter-controlled AC servo drive," *IEEE Trans. Ind. Electron.,* vol. 51, pp. 1113-1117, 2004.

[14] Z. X. Zou, K. Zhou, Z. Wang, and M. Cheng, "Frequency-Adaptive Fractional-Order Repetitive Control of Shunt Active Power Filters," *IEEE Transactions on Industrial Electronics,* vol. 62, pp. 1659-1668, 2015.

[15] T. Liu and D. Wang, "Parallel Structure Fractional Repetitive Control for PWM Inverters," *IEEE Transactions on Industrial Electronics,* vol. 62, pp. 5045-5054, 2015.

[16] Y. Yang, K. Zhou, H. Wang, F. Blaabjerg, D. Wang, and B. Zhang, "Frequency Adaptive Selective Harmonic Control for Grid-Connected Inverters," *IEEE Transactions on Power Electronics,* vol. 30, pp. 3912-3924, 2015.

[17] G. Escobar, G. A. Catzin-Contreras, and M. J. Lopez-Sanchez, "Compensation of Variable Fractional Delays in the 6k±1 Repetitive Controller," *IEEE Transactions on Industrial Electronics,* vol. 62, pp. 6448-6456, 2015.

[18] Y. Yang, K. Zhou, and F. Blaabjerg, "Enhancing the Frequency Adaptability of Periodic Current Controllers With a Fixed Sampling Rate for Grid-Connected Power Converters," *IEEE Transactions on Power Electronics,* vol. 31, pp. 7273-7285, 2016.

[19] M. A. Herr, x00E, J. R. Fischer, S. A. Gonz, x00E, lez, *et al.*, "Repetitive Control With Adaptive Sampling Frequency for Wind Power Generation Systems," *IEEE Journal of Emerging and Selected Topics in Power Electronics,* vol. 2, pp. 58-69, 2014.

[20] P. Zanchetta, M. Degano, J. Liu, and P. Mattavelli, "Iterative Learning Control With Variable Sampling Frequency for Current Control of Grid-Connected Converters in Aircraft Power Systems," *IEEE Transactions on Industry Applications,* vol. 49, pp. 1548-1555, 2013.

[21] M. Tang, A. Gaeta, A. Formentini and P. Zanchetta, "A variable frequency angle-based repetitive control for torque ripple reduction in PMSMs," *8th IET International Conference on Power Electronics, Machines and Drives (PEMD 2016)*, Glasgow, 2016, pp. 1-6.
doi: 10.1049/cp.2016.0325

[22] M. Tang, A. Gaeta, A. Formentini, K. Ohyama, P. Zanchetta, and G. Asher, "Enhanced DBCC for high-speed permanent magnet synchronous motor drives," *IET Power Electronics,* 2016.