ARCHIVES OF ELECTRICAL ENGINEERING VOL. 68(2), pp. 309–324 (2019)

DOI 10.24425/aee.2019.128270

Application of iterative learning control for ripple torque compensation in PMSM drive

ADRIAN WÓJCIK, TOMASZ PAJCHROWSKI

Institute of Control, Robotics and Information Engineering, Poznań University of Technology
Piotrowo 3a, 60-965 Poznań, Poland
e-mails: {Adrian.Wojcik/Tomasz.Pajchrowski}@put.poznan.pl

(Received: 22.09.2018, revised: 03.12.2018)

Abstract: The aim of the study was to find an effective method of ripple torque compensation for a direct drive with a permanent magnet synchronous motor (PMSM) without time-consuming drive identification. The main objective of the research on the development of a methodology for the proper teaching a neural network was achieved by the use of iterative learning control (ILC), correct estimation of torque and spline interpolation. The paper presents the structure of the drive system and the method of its tuning in order to reduce the torque ripple, which has a significant effect on the uneven speed of the servo drive. The proposed structure of the PMSM in the *dq* axis is equipped with a neural compensator. The introduced iterative learning control was based on the estimation of the ripple torque and spline interpolation. The structure was analyzed and verified by simulation and experimental tests. The elaborated structure of the drive system and method of its tuning can be easily used by applying a microprocessor system available now on the market. The proposed control solution can be made without time-consuming drive identification, which can have a great practical advantage. The article presents a new approach to proper neural network training in cooperation with iterative learning for repetitive motion systems without time-consuming identification of the motor.

Key words: ripple torque, iterative learning control, artificial neural network, permanent magnet synchronous motor

1. Introduction

Requirements for positioning quality, dynamics and repeatability of motion trajectories in servo drives have contributed to the development of so-called torque motors. Direct drive – it is a drive in which the working machine is directly connected to a motor without a mechanical transmission. The motor has a special construction that allows the drive system to operate at low angular speeds. The lack of mechanical transmission introduces many benefits, such as: elimina-

tion of backlash introduced by the transmission, which improves static accuracy of operation and dynamic properties of the drive, increases efficiency of the drive system (no mechanical losses in the transmission) and reliability (fewer mechanical elements). In addition, in the case of a direct drive with the PMSM, additional advantages are: no energy loss in the rotor, high overload torque, and therefore they are popular in machine tool technology and robot drives [1, 14].

However, direct drive testing with the PMSM has highlighted some of their specific features, which make them difficult to control precisely. No gearbox means that certain changes of the moment of inertia have a direct impact on the operation of the drive motor. Also, the change of the load torque, friction, directly affects the electric motor. In addition, torque ripple is also a disadvantage. This affects the speed ripple, which ultimately decreases the quality of the electric drive control. The main reasons of harmonic torque are: the lack of sinusoidal flux density distribution around the air gap (electromagnetic torque ripples), cogging torque and current measurement errors [2, 3, 15]. Despite the low amplitude of velocity ripples, these negative phenomena significantly reduce the precision of speed and position control. Therefore, one of the aims of drive systems is to achieve a smooth electromagnetic torque [4, 5].

The achievement of the smooth electromagnetic torque in the PMSM, especially in direct drives, is the subject of research in many scientific centers around the world [3–7]. A review of the literature leads to the conclusion that there are appropriate magnet designs in the rotor, or the special shape of the stator slots [5, 15]. This approach reduces the average torque and, due to special requirements, increases the cost of production. An alternative solution is to compensate for this parasitic phenomenon by means of control algorithms. Today's use of complex algorithms in microprocessor systems can significantly reduce torque ripple.

In many scientific studies, it is important to identify factors of the ripple torque. In many works, a key problem is the correct identification of these parameters. Different methods are used, e.g. the calculation of the Fourier coefficients [8], in [2] a torque estimator is used, which utilizes the measurement of the electromotive force at low speeds. In that paper [8], a torque controller with an additional torque compensation signal is applied. In many other works, methods of estimating torque and flux are used, which significantly complicates implementation. The paper [2] shows that the use of a very fast current controller, operating according to a dead-beat forecasting algorithm, significantly improves the quality of compensation. In the article [2], the values of the current compensating for electromagnetic and ripple torque are placed in the tables as a function of the rotor position. The paper [3] shows that power supply errors of electronic circuits and current path scaling are the reasons of additional torque fluctuation. The studies [9–11] showed that iterative learning control (ILC) and repetitive control (RC) methods can also be used to the minimization of torque ripples. In this work, the structure of PMSM control in *dq* axes was proposed, which was equipped with a neural compensator of the ripple torque. An overarching system of iterative learning has been introduced. The iterative learning is carried out on the basis of iterative torque estimation, performed on the basis of interpolation with spline functions, so that we can quickly estimate it.

2. Structure of the control system

Fig. 1 shows the structure of the control system that has been used in the work. The drive is a cascade speed control system for a permanent magnet synchronous motor and consists of: a three-phase inverter with space vector pulse width modulation (SVPWM); a current loop in

the *dq* axis with current transformer blocks and an optimized 2DFC current [1]; a digital speed measurement and a controller [12].

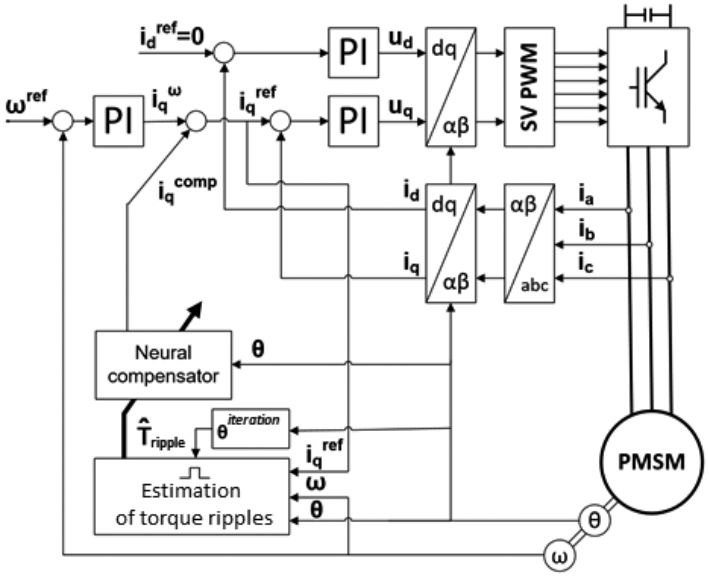


Fig. 1. Block diagram of control system

(a) Mechanical part of the drive system

The mechanical part of the drive is described as transfer function (1) [17]:

$$G_{\text{mech}}(s) = \frac{\Omega(s)}{T_{\Sigma}(s)} = \frac{1}{Js}, \tag{1}$$

where: $G_{\text{mech}}(s)$ is the mechanical part continuous transfer function, $\Omega(s)$ is the angular velocity transformation, $T_{\Sigma}(s) = T_E(s) + T_R(s) + T_L(s)$ is the total torque transformation (T_E) is the electromagnetic motor torque, T_R is the ripple motor torque, T_L is the load torque), J is the motor momentum of inertia.

(b) Current control loop

The current control loop consists of a 2DOFC current controllers in the *dq* axes. The electromagnetic part of the drive is described as a transfer function:

$$G_i(s) = \frac{I_q(s)}{I_q^{\text{ref}}(s)} = \frac{1}{s\tau_T + 1} e^{-s\tau_T^{\text{delay}}}, \tag{2}$$

where: $G_i(s)$ is the current control loop continuous transfer function, $I_q(s)$ is the active current component transformation, $I_q^{\text{ref}}(s)$ is the active current component reference value transformation, τ_T is the current control loop time constant, τ_T^{delay} is the current control loop delay.

(c) Speed controller and measurement

The outer angular velocity control loop is based on a parallel PI digital controller with the use of a trapezoidal integration method (3):

$$R_\omega(s) = \frac{I_q^{\text{ref}}(s)}{E_\omega(s)} = K_p^\omega + K_I^\omega \frac{\tau_s}{2} \frac{z+1}{z-1}, \tag{3}$$

where: $R_\omega(z)$ is the speed controller discrete transfer function, $E_\omega(z)$ is the speed error transformation, K_P^ω is the controller proportional gain, K_I^ω is the controller integration gain, τ_s is the sample time.

Speed measurement is based on a high-resolution incremental encoder. The measurement is modelled as the procedure of quantization and discrete differentiation of angular position and additional digital filtration with a second order low pass filter [1].

(d) Torque ripple

Two main sources of torque ripple are considered in this model [3, 13]:

– **Cogging torque** – the pulsating torque generated by the interaction of the rotor magnetic flux and the change in the angular position in the stator magnetic reluctance. Cogging torque occurs even when the system is disconnected from the power source [3, 13]. For a clearer interpretation this disturbance can be separated and examined with regard to the cause of existence as: *native components* (NC) – consequence of motor design parameters that exists always, even in ideally manufactured motors; *additional components* (AC) – appear only in permanent magnet motors with irregularities, e.g. stator teeth misplacements (ACT) or width and thickness variations and/or misplacements of rotor's permanent magnets (ACR) [18]. In general, cogging torque components can be expressed as (4), (5) [3, 13].

$$T^{\text{cogg}} = T_{\text{NC}}^{\text{cogg}} + T_{\text{ACT}}^{\text{cogg}} + T_{\text{ACR}}^{\text{cogg}}, \tag{4}$$

$$T_X^{\text{cogg}} = \sum_{i=} A_X^i \sin\left(n_X^i \cdot \theta + \phi_X^i\right), \tag{5}$$

where: T_X^{cogg} is the harmonic component of cogging torque, X is the component index: NC, ACR or ACT, A_X^i is the harmonic component amplitude, ϕ_X^i is the harmonic component phase, n_X^i is the harmonic component order: n_{NC}^i is equal to i times least common multiple of q and p, n_{ACR}^i is equal to i times q, and n_{ACT}^i is equal to i times p (p is the rotor poles number and q is the stator slots number),

– **Flux harmonics** – the demagnetization phenomenon of permanent magnets due to temperature rise has a significant impact on the maximum torque capability and the efficiency of permanent magnet motors [3, 6]. Electromagnetic torque resulting from this effect can be expressed as (6) [3, 13].

$$T^{\text{flux}} = K_e i_q \sum_{i=1} \psi^i \sin\left(6i \cdot \theta + \phi^i\right), \tag{6}$$

where: T^{flux} is the motor torque effected by flux harmonic, ψ^i is the amplitude of flux harmonic component, ϕ^i is the phase of flux harmonic component.

In both cases, relation between harmonic disturbance order, angular velocity and harmonic frequency can be described as (7). This means that the frequency of the torque ripples will change depending on the operating point.

$$f_R = \frac{n \cdot \omega}{2\pi}, \tag{7}$$

where: f_R is the torque ripple frequency, n is the torque ripple order in angular position domain, ω is motor angular velocity.

3. Concept of compensation system

The presented compensation concept utilizes the assumption (8), which means that the task of a compensator is to recreate the value of the cogging torque. This assumption is the basis for the initial selection of network weights and the application of the iterative learning method. A conceptual diagram of the compensation system is presented on Fig. 2.

$$i_q^{\text{comp}} = \frac{\hat{T}_{\text{ripple}}}{K_e}, \tag{8}$$

where: i_q^{comp} is the compensation current, K_e is the constant torque of the motor, $\hat{T}_{\text{ripple}}$ is the estimator of the ripple torque.

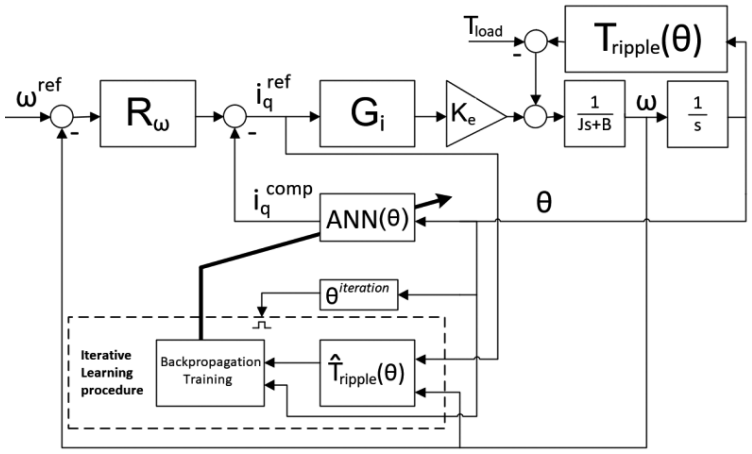


Fig. 2. Diagram of the compensation system ($R\omega$ is the PI speed controller, G_i is the transfer function of a current control loop)

The compensation system contains three basic elements: an artificial neural network, a torque estimator and an iterative network training procedure. The ripple torque compensation procedure takes place in the domain of angular position of the motor shaft, so depending on the rotational speed, the compensation uses a different number of measurement samples for the estimation task. The algorithm was designed for slow revolutions, which translates into a larger number of measuring points. Starting from the initial position, the compensator performs acquisition

of active current and rotational speed samples. After turning some selected angle $\theta^{\text{iteration}}$, the collected data is used in the estimation procedure of local torque disturbances (i.e. from $\theta^{\text{iteration}}$ – wide section) using spline interpolation and a simple dynamic model of the mechanical part of the motor. Next, the neuron compensator undergoes supervised training, based on the obtained estimation. This procedure is repeated each time the multiplicity of the $\theta^{\text{iteration}}$ angle is exceeded.

The iterative re-training of the neural network, using the generalizing properties of neural networks, allows one to obtain a neural model of torque disturbances, adequate for the entire range of angular positions of the motor shaft, i.e. a global model of disturbances.

A diagram of the neural ripple torque compensator is shown in Fig. 3. The compensator is a one-way, sigmoid, three-layer neural network. Compensation is based on the current angular position sample of the motor shaft. The initial processing of network input data consists in creating a vector composed of the remainder of dividing the current angular position by successive multiples of the full revolution related to the native harmonic components of the ripple torque. The initial learning of the network has been done as follows:

– For each of the native harmonic components a two-layer sigmoid network with four neurons were trained in the hidden layer and one in the output layer. 100 samples of the corresponding harmonic signal were used as teaching data,
– The obtained nets were connected in parallel into one neural structure. A third layer was added, the output of which is a simple sum of the outputs of the second hidden layer.

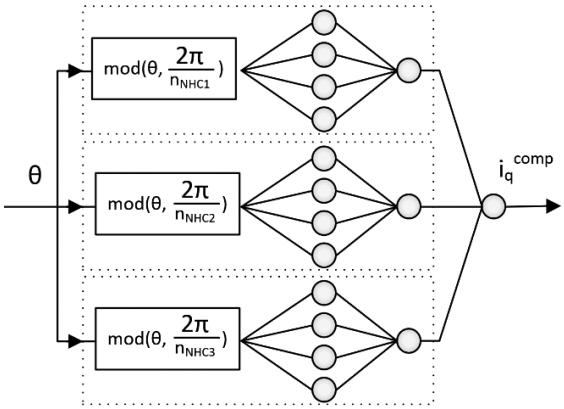


Fig. 3. Initial neural compensator structure

The neural network thus obtained has two main advantages: it provides a simple and effective way of obtaining initial weights that provide good compensation for the native components, and it contains a large number of unused (zero) weights, the modification of which during iterative learning should improve the quality of the final compensation. The proposed iterative learning is done on the basis of the collected measurement data after the motor shaft has made a certain angle $\theta^{\text{iteration}}$. The algorithm used to teach neural networks is the Levenberg-Marquardt one. This method is one of the groups of supervised learning methods [16], therefore it requires the knowledge of the selected network output. For this purpose, the course of the ripple moment within each iteration is estimated. Fig. 4 shows a diagram of the estimation procedure.

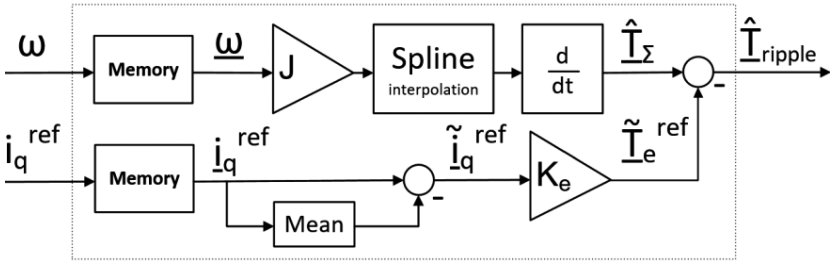


Fig. 4. Diagram of ripple torque estimation procedure (ω, $\underline{\omega}$ are the angular velocities: measured value and vector of samples from previous iteration, i_q^{ref}, $\underline{i}_q^{\text{ref}}$, $\underline{I}_q^{\text{ref}}$ are the active currents: measured, vector of samples and vector of harmonic components from previous iteration, $\underline{\widehat{T}}_\Sigma$, $\underline{\widetilde{T}}_e^{\text{ref}}$, $\underline{\widehat{T}}_{\text{ripple}}$ are vectors of estimation of total torque, vector of estimation of harmonic components of motor electromagnetic torque and vector of estimation of torque ripples)

The compensation system consists of a one-way, sigmoidal neural network, which as an input argument takes the current angular position of the motor shaft. The iteration learning system performs the compensator training on the basis of vectors: angular position speed, reference electromagnetic torque. The network training procedure is carried out each time after the shaft rotates by a certain angle. Cogging torque can be divided into two basic groups of harmonic components (4): NHC – *Native Harmonic Components*, related to the construction of the motor and additional – AHC – *Additional Harmonic Copponent* related to asymmetry resulting from production errors [4]. The estimation is made on the basis of Eq. (9). The procedure requires the speed vector to be differentiated. For this purpose, it is interpolated with spline functions and then symbolically differentiated. One of the basic advantages of using interpolation by spline functions is the stability of calculations even for a large number of points. Symbolic differentiation guarantees better noise immunity than finite difference methods. After obtaining the coupling torque vector for a given iteration, the network weights for each of the teaching signal samples are updated.

$$\widehat{T}_{\text{ripple}} = J\frac{\mathrm{d}}{\mathrm{d}t}\omega - K_e i_q^{\text{ref}}. \tag{9}$$

The main advantage of using artificial neural networks as an intermediary component in the compensation of torque ripples is their generalizing properties. Disturbance estimation takes place locally (for subsequent $\theta^{\text{iteration}}$ segments) and iterative neural network re-learning allows obtaining a compensating signal, ensuring a high level of speed unevenness reduction for the whole range of angular position of the motor shaft. The spline estimation process returns *local analytical models*, the neural network generalizes these results to the *global neural model*.

4. Simulation results

The illustrations (line diagrams and photographs) should be suitable for direct reproduction. The simulation was conducted to verify the proposed compensation algorithm for motor torque ripple compensation. The simulation model used is based on the assumed mathematical models of

drive system components (1)–(3), connected according to the diagram shown in Fig. 2. Adopted parameters of the simulation model are as following:

- moment of inertia: $J = 0.753$ kg m^2,
- torque constant: $K_e = 17.5$ Nm/A,
- rated load torque: $T_N = 50$ Nm,
- sample time: $\tau_s = 100$ µs,
- maximum reference current: $i_q^{\max} = 6$ A,
- current control loop delay: $\tau_T^{\text{delay}} = 200$ µs,
- speed controller proportional gain: $K_P^{\omega} = 0.5$,
- speed controller integral gain: $K_I^{\omega} = 0.04$,
- number of motor poles: 24,
- number of motor slots: 216 (9 per pole),
- iteration angle: $\theta^{\text{iteration}} = 1.047$ rad (60 deg).

Disturbance in a form of torque ripples was modeled based on (4), (5) and (6), therefore fundamental orders of, respectively, the cogging torque and torque generated by flux harmonics are 216 and 6 in angular position domain. This corresponds to an angular period of $1\frac{2}{3}$ and 60 degrees. The iteration angle $\theta^{\text{iteration}}$ was selected as the least common multiple of these values.

Fig. 5 shows an example of a rotational speed error as time series, which visualizes operations of the compensation algorithm. The first iteration (from system start to turning by angle $\theta^{\text{iteration}}$) is to collect training data for a neural compensator: angular velocity, angular position and an active current component (e.g. current of q axis in PMSM drive). From the second iteration, iterative network training takes place, each time based on data from the previous iteration. The estimation of the torque ripples is used as the training target (Fig. 3). During the third iteration, the active

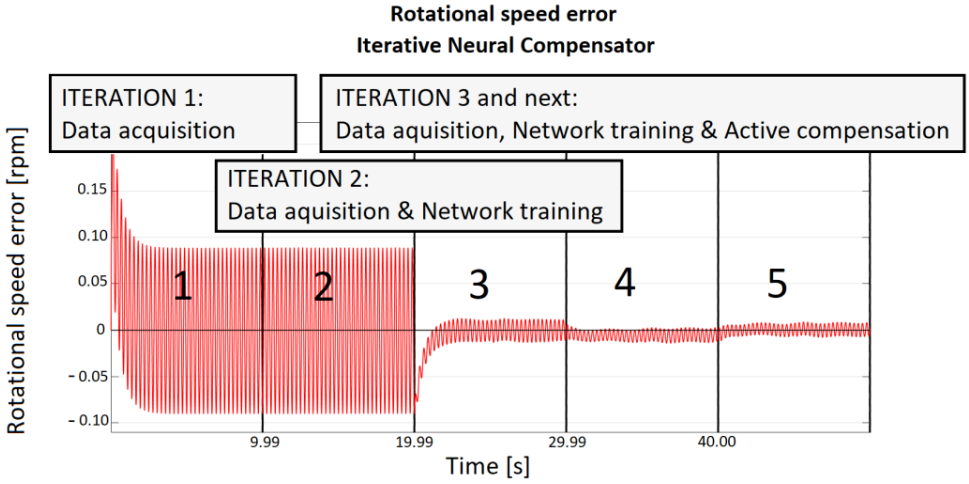
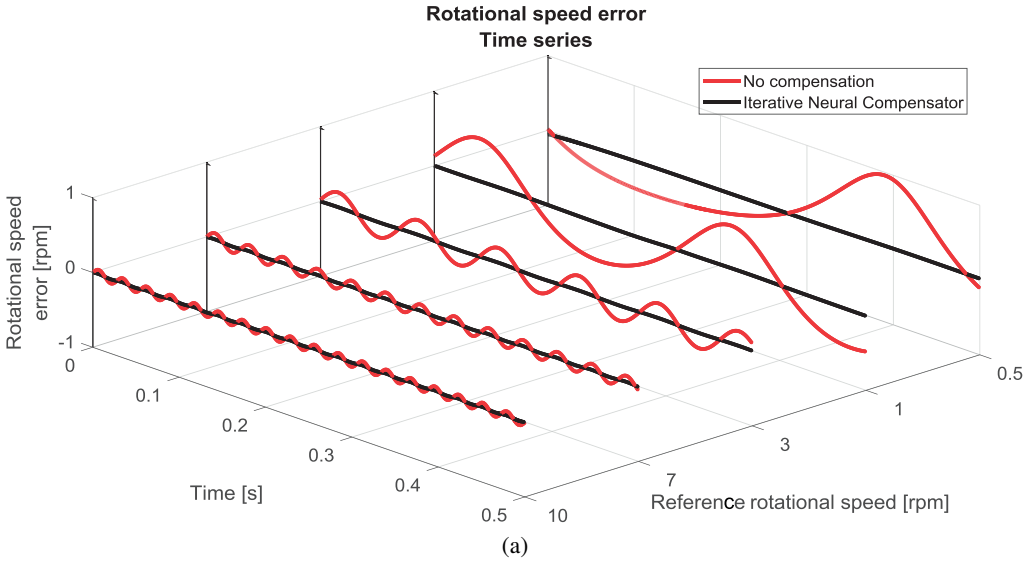


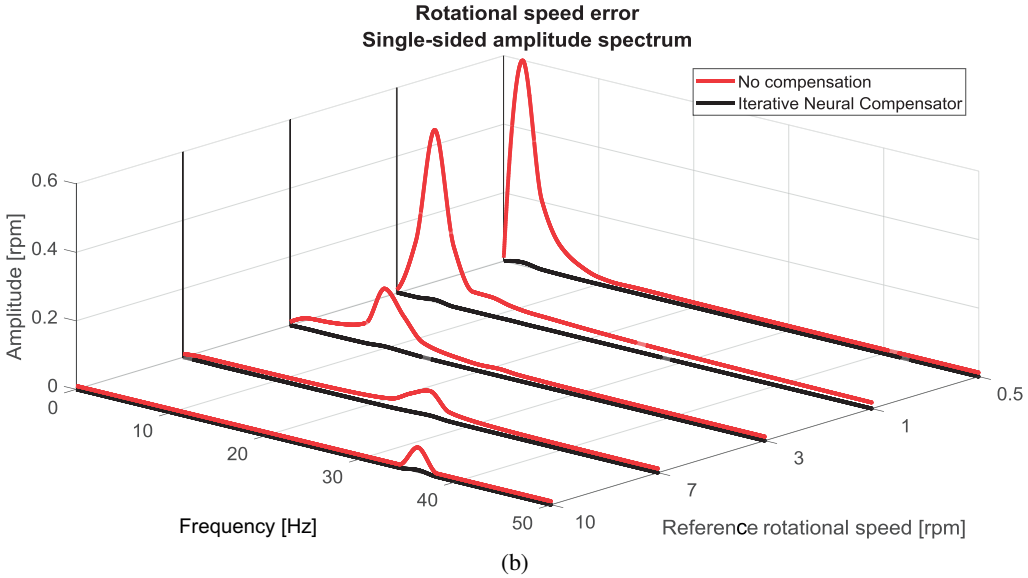
Fig. 5. Rotational speed error – example simulation results for iterative neural compensator of torque ripples ($n^{\text{ref}} = 1$ rpm)

neural compensation of the torque ripples takes place in every step of a rotational speed control loop. In addition, the compensator can be pre-learned to recreate selected harmonic components and reduce rotational speed unevenness at start-up.

Fig. 6 shows a rotational speed error as time series and the single-sided amplitude spectrum for various values of reference speed. In the range shown, with the decrease of reference speed



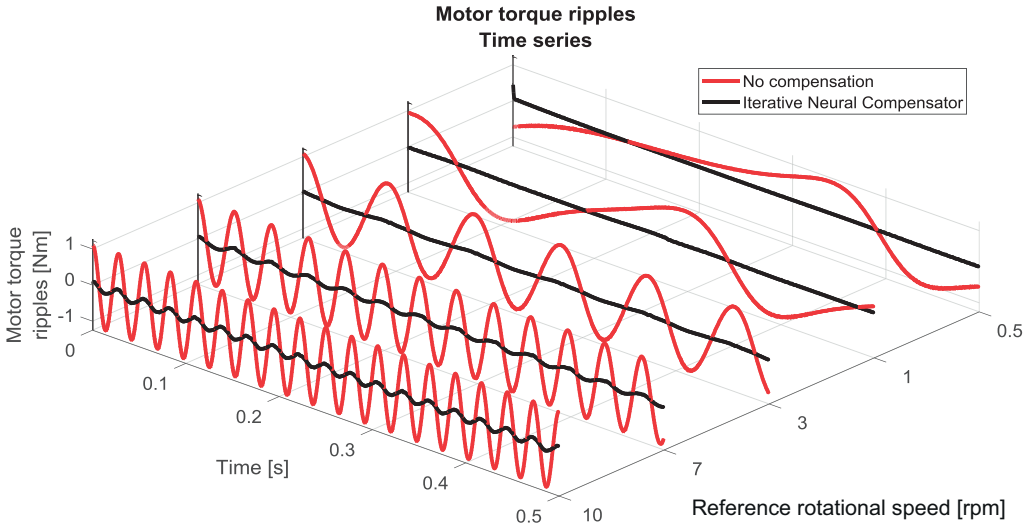
(a)



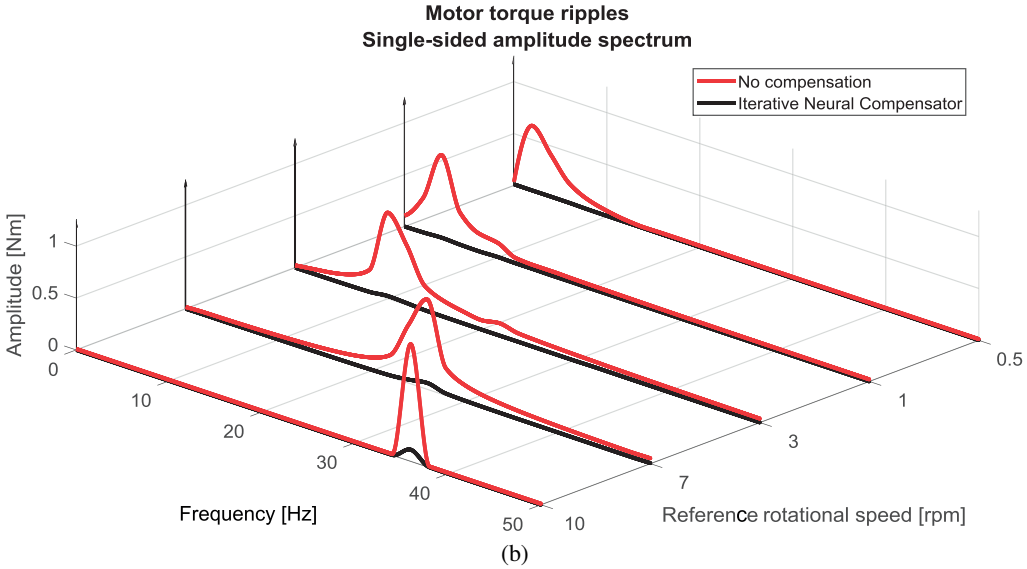
(b)

Fig. 6. Rotational speed error: time series (a) and amplitude spectrum (b) for various reference speed

value, the amplitude of the rotational speed error increases. This trend is consistent with the frequency response of the assumed model. However, compensation quality, expressed as (10), is approximately constant for various reference speed and takes value of 90% to 95%. Fig. 7 shows motor torque as time series and a single-sided amplitude spectrum for various values of reference speed, in a steady-state without load – the amplitude of the torque ripples does not change, only



(a)



(b)

Fig. 7. Motor torque: time series (a) and amplitude spectrum (b) for various reference speed

its frequency component.

$$Q_\omega = \left(1 - \frac{RMSE\left(e_\omega^{\text{comp}}\right)}{RMSE\left(e_\omega\right)}\right) \times 100\%, \tag{10}$$

where *RMSE* is the root of the mean square error.

Fig. 8 shows the characteristic of iterative neural compensation quality from the value of the moment of inertia assumed in an interpolation procedure (in respect to real value of the moment of inertia). Fig. 9 shows analogous characteristics for quality and motor torque gain. In

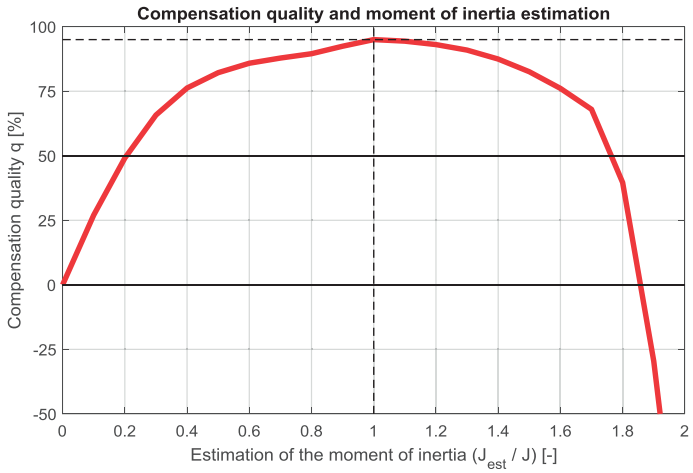


Fig. 8. Compensation quality in function of moment of inertia estimation accuracy

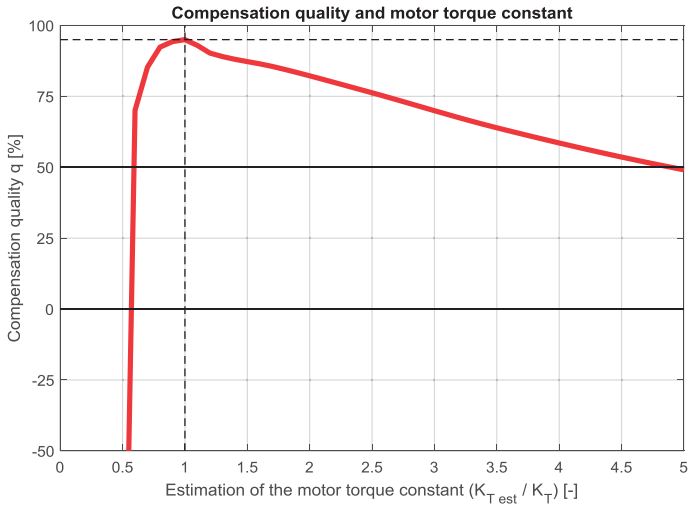


Fig. 9. Compensation quality in function of motor torque gain estimation accuracy

both cases, a wide range of parameter values is available, for which the performance algorithm fulfills its task, with maximum compensation quality for perfectly identified parameters. In the case of delay in a current control loop – increasing the delay value monotonically worsens the quality of compensation (Fig. 10). The simulation research was originally conducted in the MATLAB/Simulink environment, allowing one to evaluate the accuracy of the interpolation and training procedures. In addition, the unity tests were repeated in the VisualDSP++ environment, using the previously obtained data, in order to verify the correctness of method implementation.

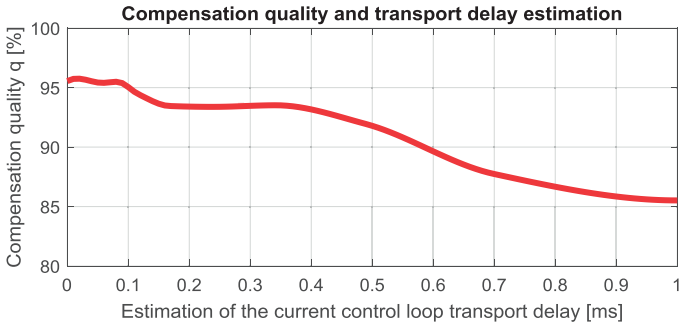


Fig. 10. Compensation quality in function of current control loop delay

5. Experimental results

The laboratory set-up, presented in Fig. 11, consists of a three-phase PMSM supplied from a PWM inverter. The control system is based on 32-bit floating-point processors ADSP-21061, the performing task of the current controller and rotational speed controller with iterative neural compensation of torque ripples. The speed and position measurement are carried out by an incremental encoder. The mechanical part of the drive is directly connected to the motor shaft and consists of the set of wheels with a well-calibrated moment of inertia.

The drive system was tested in a steady-state in various operating points. Fig. 12 shows the time series and single-sided amplitude spectrums of a rotational speed error in various reference speed. Naturally, in the real drive system harmonic composition of speed unevenness is wider and results from remaining unmolded disturbances (e.g. frictions) and measurement noises. Quality of the torque ripple compensation was on the level of 75%.

When comparing the rotational speed error time series with and without compensation, it can be noted that the speed unevenness increases with the decreasing value of the reference speed. In the case of $n^{\mathrm{ref}} = 0.5$ rpm, the peak-to-peak amplitude of the speed error without compensation $\Delta e_\omega = 1$ rpm $= 2n^{\mathrm{ref}}$, which means that the drive operates in cycles of braking and stating up, not in even rotation. The iterative training of a neural compensator, based on the process of torque ripples estimation using data acquisition and interpolation to analytical form, provides an effective way of identification of fundamental harmonic components of disturbance. The frequency spectrum of the rotational speed error shows considerable reduction of amplitude of those fundamental harmonics. With the main source of speed unevenness being cogging torque

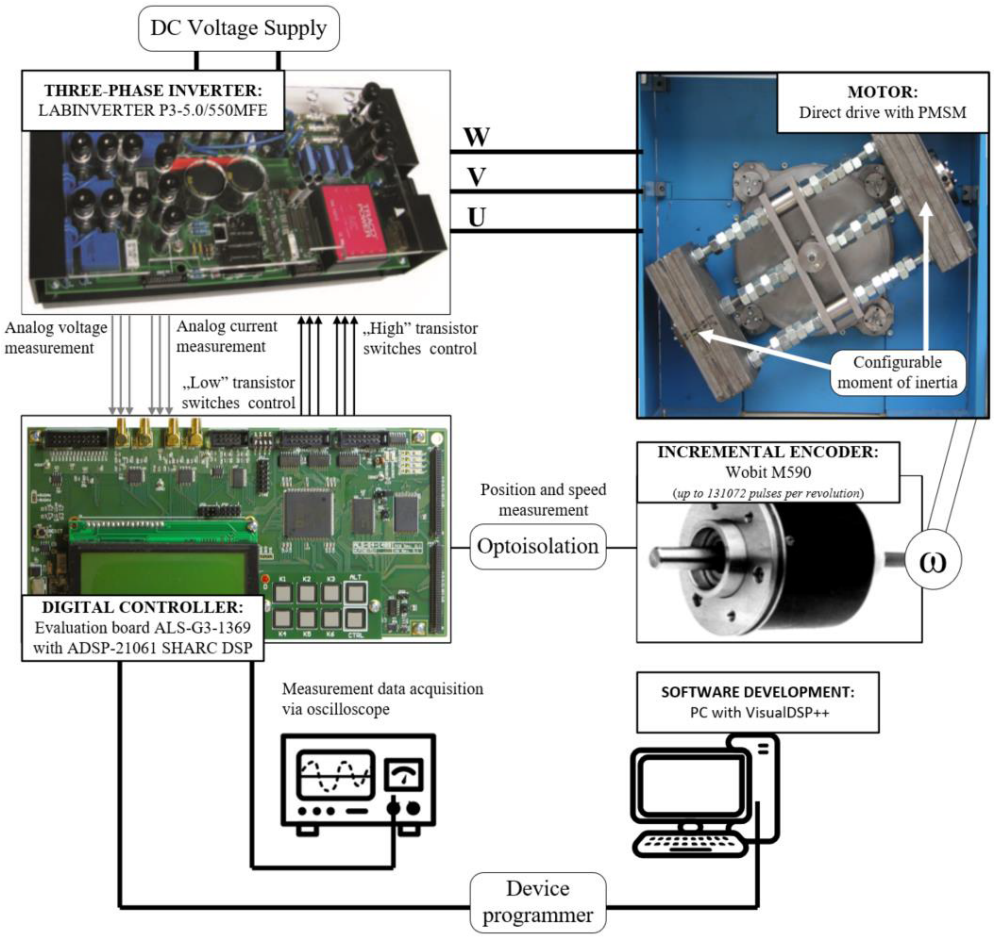
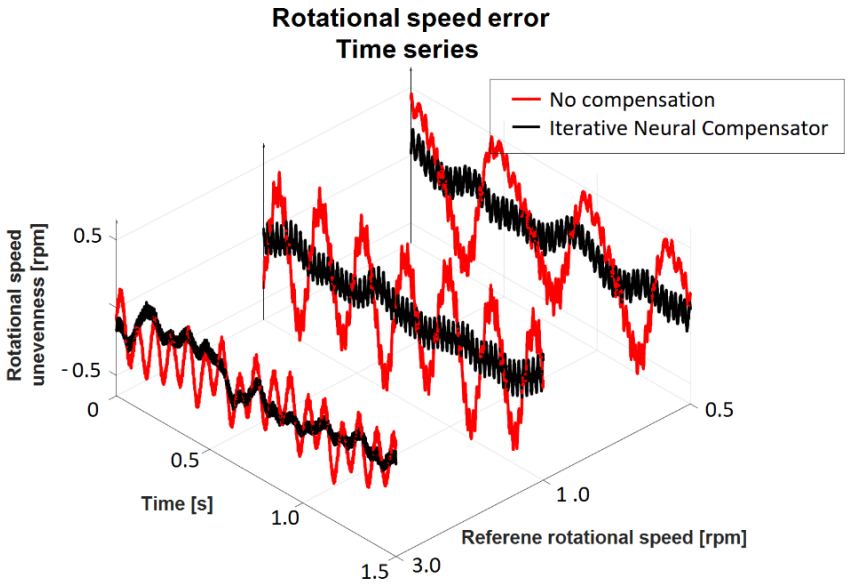


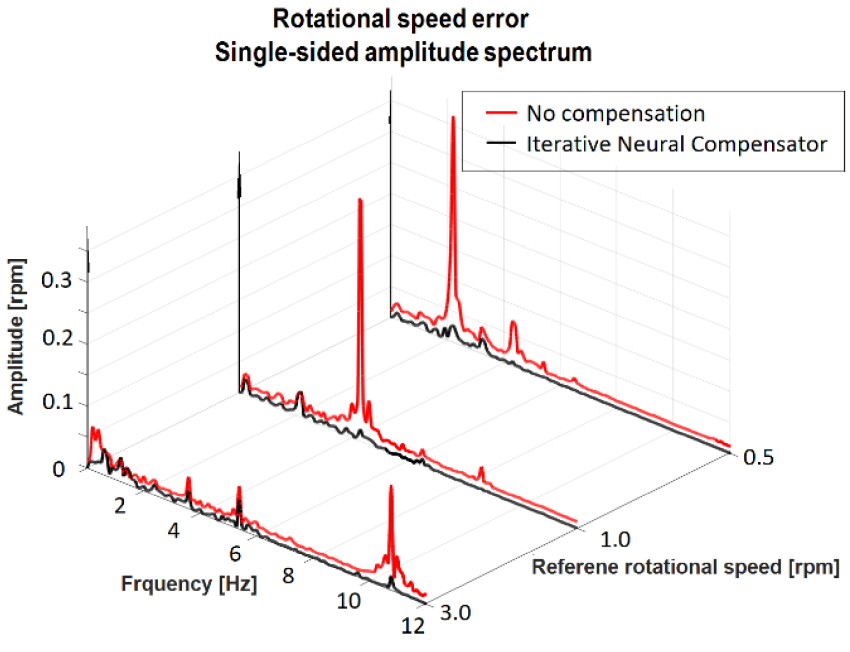
Fig. 11. Laboratory set-up schematic diagram

with an angular order of 216, it corresponds to a frequency of 2, 4 and 11 Hz for the reference speed of a respective value of 0.5, 1 and 3 rpm. As shown, after applying the compensation method, these components have been almost completely removed from the speed error spectrum.

The experimental research confirms the practical applicability of the proposed method. The main advantage of using the proposed method is the built-in disturbance identification process. However, current implementation assumes a constant value of active load torque. For low rotational speed, algorithm's operation is stable and gives reproducible results. Faster rotation of the drive shortens the time of a single iteration and decreases the number of samples, which in turn deteriorates identification of ripples and compensator training. For higher speeds, however, compensation is not critical. Algorithm computation time and memory consumption require use of high-performance hardware platforms, such as digital signal processors with hardware floating-point matrix operation support.



(a)



(b)

Fig. 12. Rotational speed error: time series (a) and amplitude spectrum (b) for various reference speed of real drive system

6. Conclusion

The paper presents the structure of PMSM control of repetitive motion in *dq* axes which was equipped with the compensation system that contains three basic elements: an artificial neural network, a torque estimator and an iterative network training procedure. The aim of the study was to reduce ripple torque, which has a significant impact on speed and position fluctuations. The obtained simulation results allow one to state that the proposed method of compensating the cogging torque enables a significant (up to 95%) reduction of the unevenness of the angular velocity of the PMSM. The presented structure of estimation and the learning iterative compensator ensures fast convergence. The use of an iterative, supervised (by the estimator of torque ripples) re-training neural network, implemented in the angular position domain, is the original technique of "intelligent averaging" of local disturbance estimation. The proposed compensation method was verified experimentally, confirming the practical application of the algorithm. The quality of compensation is noticeably lower than in the simulation results (up to 75%), but it is still an effective solution, especially in the low speed range. Further research should include the impact of disruption identification methods, training methods and the structure of the neural network on the quality of compensation.

Acknowledgements

The presented results of research were carried out under the theme No. 04/45/DSPB/0184, which was funded by the Polish Ministry of Science and Higher Education.

References

[1] Brock S., Luczak D., Nowopolski K., Pajchrowski T., Zawirski K., *Two Approaches to Speed Control for Multi-Mass System with Variable Mechanical Parameters*, IEEE Transactions Industrial Electronics, vol. 64, no. 4, pp. 3338–3347 (2017).

[2] Holtz J., Springop L., *Identification and Compensation of Torque Ripple in High-Precision Permanent Magnet Motor Drives*, IEEE Transaction on Industrial Electronics, vol. 43, no. 2, pp. 309–320 (1996).

[3] Yang J., Chen W., Li S., Guo L., Yan Y., *Disturbance/Uncertainty Estimation and Attenuation Techniques in PMSM Drives – A Survey*, IEEE Transactions on Industrial Electronics, vol. 64, no. 4, pp. 3273–3285 (2017).

[4] Sikorski A., Grodzki R., *A new DTC control for PMSM with torque ripple minimization and constant switching frequency*, COMPEL – The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, vol. 30, no. 3, pp. 1069–1081 (2011).

[5] Martinez J., Krischan K., Muetze A., *Minimization of a SynRel's oscillating torque by calculation of the appropriate skew angle*, COMPEL – The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, vol. 36, no. 3, pp. 824–835 (2017).

[6] Sergeant P., Crevecoeur G., Dupré L., Van den Bossche A., *Characterization and optimization of a permanent magnet synchronous machine*, COMPEL – The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, vol. 28, no. 2, pp. 272–285 (2009).

[7] Ferreti G., Magnani G., Rocco P., *Modeling, Identification, and Compensation of Pulsating Torque for PMAC Machines*, IEEE Transaction on Industrial Electronics, vol. 45, no. 6, pp. 912–920 (1998).

[8] Grcar B., Cafuta P., Stumberger G., Stanković A.M., *Control-Based Reduction of Pulsating Torque for PMAC Machines*, IEEE Transaction on Energy Conversion, vol. 17, no. 2, pp. 169–175 (2002).

[9] Pipeleers G., Moore K.L., *Unified Analysis of Iterative Learning and Repetitive Controllers in Trial Domain*, IEEE Transaction on Automatic Control, vol. 59, no. 4, pp. 953–965 (2014).

[10] Mattavelli P., Tubiana L., Zigliotto M., *Torque-Ripple Reduction in PM Synchronous Motor Drives Using Repetitive Current Control*, IEEE Transaction on Power Electronics, vol. 20, no. 6, pp. 1423–1431 (2005).

[11] Xia Ch., Deng W., Shi T., Yan Y., *Torque Ripple Minimization of PMSM Using Parameter Optimization Based Iterative Learning Control*, Journal of Electrical Engineering and Technology, vol. 11, no. 2, pp. 425–436 (2016).

[12] Pajchrowski T., Zawirski K., Brock S., *Application of neuro-fuzzy techniques to robust speed control of PMSM*, COMPEL – The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, vol. 26, no. 4, pp. 1188–1203 (2007).

[13] Černigoj A., Gašpari L., Fišer R., *Native and Additional Cogging Torque Components of PM Synchronous Motors – Evaluation and Reduction*, Automatika, 2017, vol. 51, no. 2, pp. 157–165 (2010).

[14] Bose B.K., Chio K.M., Kim H.J., *Self Tunning Neural Network Controller for Induction Motor Drives*, IEEE Annual Conference of the IEEE Industrial Electronics Society, Sevilla, Spain, pp. 152–156 (2002).

[15] Stamenković I., Jovanović D., Vukosavić S., *Torque ripple Verification in PM Machines*, EUROCON 2005 – The International Conference on "Computer as a Tool", Serbia & Montenegro, Belgrade, pp. 1497–1500 (2005).

[16] Reynaldi A., Lukas S., Margaretha H., *Backpropagation and Levenberg-Marquardt Algorithm for Training Finite Element Neural Network*, 2012 Sixth UKSim/AMSS European Symposium on Computer Modeling and Simulation, Malta, Valetta, pp. 89–94 (2012).

[17] Vas P., *Sensorless Vector and Direct Torque Control*, Oxford University Press (1998).