Online neural network application for compensation of the VSI voltage nonlinearities

BUCHTA, L.; KOZOVSKÝ, M.

IECON 2023- 49th Annual Conference of the IEEE Industrial Electronics Society

eISBN: 979-8-3503-3182-0

DOI: https://doi.org/10.1109/IECON51785.2023.10312305

Online neural network application for compensation of the VSI voltage nonlinearities

1st Ludek Buchta
Central European Institute of Technology
Brno University of Technology
Brno, Czech Republic
Ludek.Buchta@ceitec.vutbr.cz

2nd Matus Kozovsky
Central European Institute of Technology
Brno University of Technology
Brno, Czech Republic
Matus.Kozovsky@ceitec.vutbr.cz

***Abstract*—The paper aims to solve the distortion problem of the inverter output voltages that cause harmonic deformation of the phase currents and ripple of dq- currents of the three-phase permanent magnet synchronous motor (PMSM). The inverter non-linearities adversely affect the effectiveness of the PMSM control algorithm. The compensation strategy is based on the neural network and knowledge of the three-phase PMSM model structure and its parameters. The input data for the neural network consist of the normed values and detected polarities of the phase currents and rotor position information. As a result, the proposed artificial neural network (ANN) can extract non-linear functions from the measured data to compensate for the VSI output voltages. The ANN is designed to learn online while the PMSM is running. The back-propagation algorithm is used for neural network learning. The proposed strategy was implemented in an AURIX TC397 microcontroller and validated by experiments on a real PMSM. The presented results demonstrate the effectiveness of the proposed solution.**

***Index Terms*—dead-time compensation, artificial neural network (ANN), voltage source inverter (VSI), permanent magnet synchronous motor (PMSM)**

I. Introduction

Voltage source inverters (VSI) have been standardly used to control AC motors in industrial applications. These applications put ever higher demands on precise control algorithms that requires not only accurate measurement of the currents and position but also accurate application of the command voltages. However, this effort is complicated by the non-linear characteristics of the VSI, such as the dead-time period, voltage drops on the switching elements and anti-parallel diodes or snubber and parasitic capacities, which distort the output voltages. The required voltages generated by the control algorithm will not agree with the inverter output voltages due to these VSI non-linearities. The output voltages deformation will be reflected by a harmonic distortion of the phase currents and subsequently by the ripple of the currents in the dq-coordinate system and torque pulsations [1], [2], which negatively affects the applied control algorithm. Therefore, it is desirable to compensate for VSI non-linearities. The standard compensation strategy that uses a feed-forward approach is the simplest solution to this issue [3]. It compensates for the average voltage losses of the one PWM period based on the polarities of the phase currents. The polarity detection near to zero-crossing region is critical for a standard approach [4].

In article [5], the compensation strategy is based on the injection of the harmonic signals calculated based on a precise mathematical model of the VSI harmonic distortion. These methods are always dependent on the accuracy of the mathematical model. The accurate mathematical models of the VSI that include the snubber and parasitic capacitance are assembled for compensation strategy in [6], [7]. These methods are depending precise current measurement and a high sampling current rate. The paper [8] describes a compensation strategy that uses a second-order adaptive filter to extract the 6^{th} harmonic component of the dq- current. Subsequently, the compensation voltages are calculated based on the value of the 6^{th} harmonics and a mathematical model of the PMSM. This approach does not suppress higher parasitic harmonic components, e.g. 12^{th} harmonics in dq- currents and other multiples of the 6^{th} harmonics, which is its disadvantage.

Authors increasingly use neural networks to suppress voltage distortions of the VSI. In [9], the compensation method is based on the assumption that the vector trajectory of the output $\alpha\beta$- voltage is a circle. Depending on the value of the circular trajectory error, an adaptation rule is constructed to adjust one common weight in the input layer of the simple neural network. Subsequently, the neural network can identify the non-linear functions of compensation voltages. The disadvantage is that the neural network adapts only one weight, and others are set to a constant value. In [10], the authors used one separated neuron to suppress each 6^{th} and 12^{th} harmonic components in dq- currents. The authors in both articles do not use the full potential of the neural networks to identify the non-linear function of compensation voltages.

In order to solve the issue of the VSI output voltage deformation, the compensation strategy is designed based on an artificial neural network (ANN). The ANN approach is appropriate because it is suitable for solving strongly non-

The work has been performed in the project AI4CSM: Automotive Intelligence for/at Connected Shared Mobility No 101007326/8A21013. The work was co-funded by grants of Ministry of Education, Youth and Sports of the Czech Republic and Electronic Component Systems for European Leadership Joint Undertaking (ECSEL JU). The work was supported by the infrastructure of RICAIP that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857306 and from Ministry of Education, Youth and Sports under OP RDE grant agreement No CZ.02.1.01/0.0/0.0/17_043/0010085.

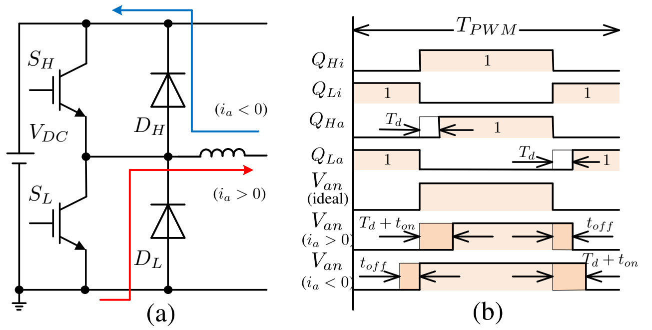

Fig. 1. (a) One leg of the three-phase VSI during the dead-time period. (b) (b) PWM signal patterns and VSI output voltage considering the non-linear effects.

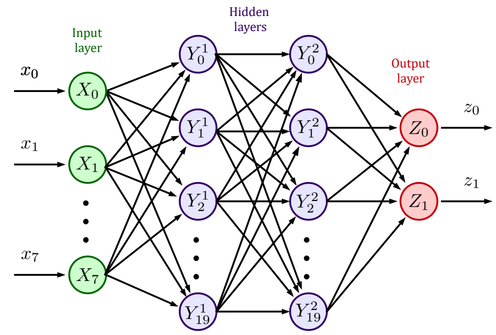

Fig. 2. Neural network topology

linear problems. The proposed ANN allow extracting non-linear functions from quantities commonly available in the vector control algorithm of the PMSM without knowing the inverter parameters. The ANN benefit is that compensation voltages can be identified online while the PMSM is running. Also, the code size of the neural network and its computational complexity are acceptable and can be implemented on a microcontroller AURIX TC397.

II. Analysis of voltage disturbance

The dead-time issue can be explained on the one leg of the typical three-phase VSI shown in Fig.1(a). This dead-time period T_d serves as protection against the simultaneous conductions of the two switching devices on the same leg of the VSI. It is inserted into the gating signals of switching devices. During the dead-time period, switching elements S_H/S_L on the same VSI leg are turned off, and the output voltage is not controlled. As a result, the phase current can only flow through the diodes D_H/D_L and the actual direction of the phase current determines the output voltage, as shown in Fig. (b). Even if the dead-time period T_d is very short, there is still a deformation of the VSI output voltage [11].

The voltage error ΔV_{an} can be defined with respect to the switching period T_{PWM} based on the behavior analysis as

$$\Delta V_{an} = v_{dead} \cdot \mathrm{sign}(i_a) = \frac{T_d + t_{on} - t_{off}}{T_{PWM}} V_{DC} \cdot \mathrm{sign}(i_a) \quad (1)$$

where V_{DC} represents DC bus voltage and the t_{on} and t_{off} represent turn-on/turn-off switching delay time, and the phase current sign function can be represented as

$$\mathrm{sign}(i_a) = \begin{cases} 1, & i_a > 0 \\ -1, & i_a < 0 \end{cases} \quad (2)$$

In paper [3], the equation (3) is used to determine the magnitude of the compensation voltage v_{dead}. It depends on voltage drops of the switching devices (V_{sat}) and diodes (V_d), and its implementation is straightforward.

$$v_{dead} = \frac{T_d + t_{on} - t_{off}}{T_{PWM}} (V_{DC} - V_{sat} + V_d) + \frac{V_{sat} + V_d}{2} \quad (3)$$

III. Compensation strategy design

The abovementioned analysis shows that the VSI output voltage deformations caused by the dead-time effect and other inverter properties can be considered non-linear functions dependent primarily on the inverter parameters. The proposed compensation strategy is based on the artificial neural network (ANN) and knowledge of the three-phase PMSM model structure and its parameter. The neural network is chosen because it is suitable for solving strongly non-linear problems.

The following requirements have been considered in designing a suitable neural network. First, the proposed ANN should extract non-linear functions only from quantities commonly available in the vector control algorithm of the PMSM and without knowing the inverter parameters. Second, the proposed ANN should be able to identify the non-linear compensation voltages online. Third, the computational complexity of the solution should be acceptable for implementation on a micro-controller.

A. Neural network topology design

The neural network topology for identifying compensation voltages in the $\alpha\beta$- coordinate system is defined by an input layer, two hidden layers and the output layer. The topology of ANN is shown in Fig. 2. The input vector to the neural network consists of eight elements. The input vector contains normed values of the phase currents in the range -1 to 1. It also includes the sign functions that represent the polarity of the phase currents. Accurate knowledge about the polarity of the phase currents is essential for the compensation strategy, especially in the region where the phase currents pass through the zero value and where the zero-clamping phenomenon is applied. The last two signals are $\sin(\theta_e)$ and $\cos(\theta_e)$, which transmit information about the current electrical position θ_e of the PMSM to the neural network. The proposed ANN should be able to determine the compensation voltages based on the information contained in the input vector $\boldsymbol{x}_i$.

$$\boldsymbol{x}_i = \left[i_a, i_b, i_c, \mathrm{sign}(i_a), \mathrm{sign}(i_b), \mathrm{sign}(i_c), \sin(\theta_e), \cos(\theta_e)\right]^T \quad (4)$$

The complexity of the problem that can be solved using ANN is determined by the number of hidden layers and the number of neurons contained in them. However, a more complex task can be solved by increasing the number of neurons at the cost of increasing the learning time and the complexity of the ANN calculation. Two hidden layers, each containing 20 neurons, were selected using simulation experiments to obtain an adequate ANN result. It gives the neural network sufficient degrees of freedom to approximate the non-linear compensation voltages.

The fundamental element of the ANN mathematical model is a one neuron, which can be described using the following equations

$$y\_in_j^{(k+1)} = b_j^{(k+1)} + \sum_{i=1}^{n} w_{j,i}^{(k+1)} y_i^{(k)} \tag{5}$$

$$y_j^{(k+1)} = f\left(y\_in_j^{(k+1)}\right) \tag{6}$$

where y_i represents the state of neuron Y_i, for neuron X_i in the input layer is x_i input signal, for neurons Y_j in other layers is the input signal defined $y_j = f(y\_in_j)$. The symbol $y\_in_j^{(k+1)}$ indicates the inner potential of the j^{th} neuron in the $(k+1)^{th}$ layer. The bias of $Y_j^{(k+1)}$ neuron is defined by $b_j^{(k+1)}$ and weight $w_{j,i}^{(k+1)}$ represents connection from i^{th} neuron to j^{th} neuron. The use of the matrices can be described as follows

$$\boldsymbol{y}^{(k+1)} = \boldsymbol{f}\left(\boldsymbol{w}\boldsymbol{y}^{(k)} + \boldsymbol{b}^{(k+1)}\right) \tag{7}$$

$$\begin{bmatrix} y_0^{(k+1)} \\ y_1^{(k+1)} \\ \vdots \\ y_n^{(k+1)} \end{bmatrix} = \boldsymbol{f}\left(\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,m} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n,0} & w_{n,1} & \cdots & w_{n,m} \end{bmatrix} \begin{bmatrix} y_0^{(k)} \\ y_1^{(k)} \\ \vdots \\ y_m^{(k)} \end{bmatrix} + \begin{bmatrix} b_0^{(k+1)} \\ b_1^{(k+1)} \\ \vdots \\ b_n^{(k+1)} \end{bmatrix}\right) \tag{8}$$

where $\boldsymbol{w}$ represents the weight matrix, and $\boldsymbol{b}$ is the vector of the threshold values contributing to each neuron in hidden layers and each neuron in the output layer.

The crucial part is the process of determining an appropriate activation function $f(x)$. The rectified linear unit (ReLU) function is chosen as the activation function $f(x)$ for both hidden layers.

$$f_{hidden}(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad f'_{hidden}(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \tag{9}$$

The benefit of using the RuLU is that it is simple to calculate its derivative, requiring only a small amount of computing resources. Also, the ReLU function and its derivative are monotonic functions. The linear function is selected as the activation function $f(x)$ in the output layer.

$$f_{out}(x) = x \qquad\qquad f'_{out}(x) = 1 \tag{10}$$

B. Neural Network training

The ANN is designed to identify the compensation voltages online when the vector control algorithm of the PMSM is running. A new training pattern is generated in each step of the control algorithm in the learning phase of the ANN. The training pattern is a vector of the errors between persistent deformation voltages and compensation voltages from the previous step. The observer calculates the persistent deformation voltages based on the model of the PMSM in dq-coordinates and measured phase currents and velocity. The persistent deformation voltages Δu_d and Δu_q are given by the following equations.

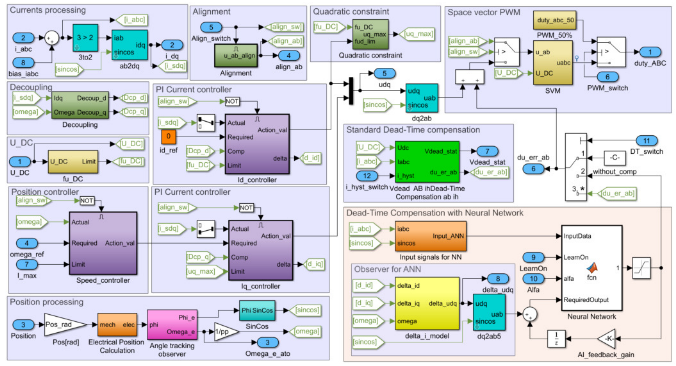

Fig. 3. Control algorithm integration

$$\Delta u_d = R_s \Delta i_d + L_d \frac{d}{dt}\Delta i_d - \omega_e L_q \Delta i_q \tag{11}$$

$$\Delta u_q = R_s \Delta i_q + L_q \frac{d}{dt}\Delta i_q + \omega_e L_d \Delta i_d \tag{12}$$

The observer model (11)-(12) does not consider the expression containing back electromotive force ($\omega_e \lambda_m$) because it does not directly contribute to the i_q current ripple. The direct and quadrature inductance and stator resistance are represented by symbols L_d, L_q and R_s, respectively. The symbol ω_e denotes the electrical angular velocity. The current errors Δi_d and Δi_q are defined as

$$\Delta i_d = i_{d_{ref}} - i_d \qquad\qquad \Delta i_q = i_{q_{ref}} - i_q \tag{13}$$

where i_d and i_q are measured currents. $i_{d_{ref}}$ is required current in d- axis and $i_{q_{ref}}$ is required current in q- axis generated by speed controller.

Assuming a steady state, when the motor operates at a constant speed, and the load torque is also stable, the current errors should be zero. However, in reality, this will not be the case. The ripple caused by VSI nonlinearities will appear in dq- currents. The dominant component of the ripple is the 6^{th} harmonic component and its multiples (the frequency of the harmonics is related to the fundamental frequency of the phase currents). The magnitude of the voltage v_{dead} from the equation (3) determines the amplitude of these harmonic components.

The observer (11)-(12) is converted to discrete form by the Euler approximation with sampling period T_s.

$$\begin{aligned} \Delta u_d(k) = R_s \Delta i_d(k) + L_d \frac{\Delta i_d(k) - \Delta i_d(k-1)}{T_s} \\ -\omega_e(k) L_q \Delta i_q(k) \end{aligned} \tag{14}$$

$$\begin{aligned} \Delta u_q(k) = R_s \Delta i_q(k) + L_q \frac{\Delta i_q(k) - \Delta i_q(k-1)}{T_s} \\ +\omega_e(k) L_d \Delta i_d(k) \end{aligned} \tag{15}$$

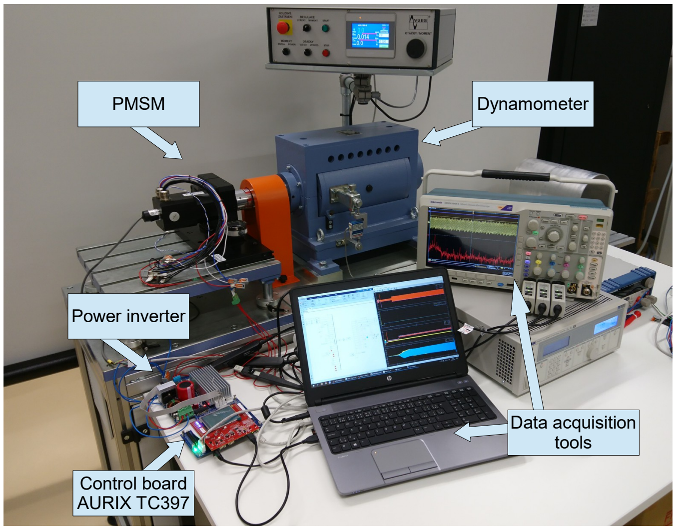

Fig. 4. The configuration of the testing platform

Then, the voltage errors Δu_d and Δu_q from equations (11)-(12) are transformed into $\alpha\beta$- coordinates.

$$\begin{bmatrix}\Delta u_\alpha(k)\\ \Delta u_\beta(k)\end{bmatrix} = \begin{bmatrix}\cos(\theta_e(k)) & -\sin(\theta_e(k))\\ \sin(\theta_e(k)) & \cos(\theta_e(k))\end{bmatrix}\begin{bmatrix}\Delta u_d(k)\\ \Delta u_q(k)\end{bmatrix} \quad (16)$$

Subsequently, it is possible to determine the ANN training pattern $\boldsymbol{T}(k) = [\Delta u_\alpha^{inp}(k), \Delta u_\beta^{inp}(k)]^T$ for the current step of the algorithm, which is defined by the following equation

$$\begin{bmatrix}\Delta u_\alpha^{inp}(k)\\ \Delta u_\beta^{inp}(k)\end{bmatrix} = \begin{bmatrix}\Delta u_\alpha(k)\\ \Delta u_\beta(k)\end{bmatrix} + \boldsymbol{K}_{gain}\begin{bmatrix}\Delta \hat{u}_\alpha^{comp}(k-1)\\ \Delta \hat{u}_\beta^{comp}(k-1)\end{bmatrix} \quad (17)$$

where $\Delta \hat{u}_\alpha^{comp}(k-1)$ and $\Delta \hat{u}_\beta^{comp}(k-1)$ represents ANN output from the previous step. These voltages are multiplied by the correction gain vector $\boldsymbol{K}_{gain}$. The magnitude of the gains is set to a value less than 1. Then, the training pattern is calculated and brought to the input of ANN. The task of the ANN is to identify non-linear functions and approximate compensation voltages. The back-propagation algorithm is chosen for neural network learning purposes.

The maximum values of the identified compensation voltages are saturated at the neural network's output. The saturation limits are set concerning the parameters of the VSI and assumed operating conditions of the PMSM. The compensating voltages are added to voltages generated by the PI current controller in the $\alpha\beta$-coordinates. These are applied to the inverter.

MATLAB/Simulink simulation verified the proposed compensation strategy in the first validation phase. The scheme of the control algorithm with ANN is shown in Fig. 3. In this phase, the appropriate number of neurons in the hidden layers and the initialization values of the weights were set experimentally. The weights in the hidden layers are set randomly in the interval $\langle 0, 0.1\rangle$, and the weights in the output layer are also set randomly in the interval $\langle 0, 0.01\rangle$. Subsequently, the ANN was prepared for implementation into the Tricore AURIX TC397 microcontroller.

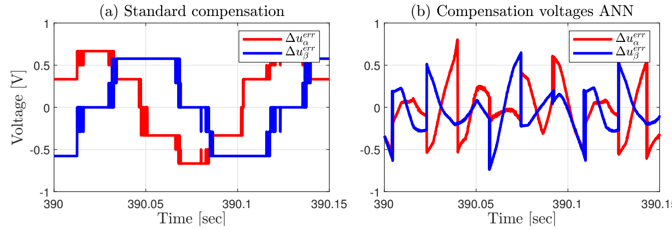

Fig. 5. Compensation voltages from standard compensation method and ANN without load

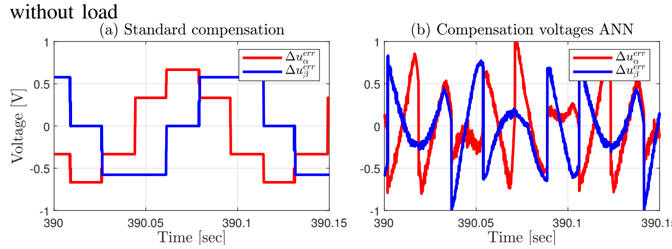

Fig. 6. Compensation voltages from standard compensation method and ANN with load 0.4 Nm

IV. EXPERIMENTAL RESULTS

The compensation strategy was validated at the test bench shown in Fig. 4 in the second phase. The test bench consists of the three-phase PMSM, which is connected to a dynamometer that loads the motor. The microcontroller Tricore AURIX TC397 controls a three-phase power inverter that drives the PMSM. The parameters of the IGBT switching elements and PMSM are listed in Tab. III. The microcontroller has six separate cores. The application uses only two of them. The motor control algorithm and the proposed ANN are implemented on the first core. The second core is dedicated to data acquisition and communication processing. The measured data is obtained from the platform via UDP communication in MATLAB/Simulink. The sampling frequency of 10 kHz is used by the first core responsible for calculating the vector control algorithm and the proposed ANN. The calculation of the control algorithm takes an average of 22 μs; another 5 μs are occupied by measuring currents and generating PWM signals. The computational complexity of the ANN during the learning phase is acceptable, taking an average of 32 μs. However, it is significantly higher than the standard compensation strategy, which takes less than 1 μs.

In this chapter, the results achieved after learning ANN are presented. The online learning of the neural network took place when the engine was operating in a steady state, i.e. the engine speed and load torque were constant. In the first experiment, the rotor speed is set to 20 rad/s, and the motor is unloaded, as shown in Fig. 7. The influence of VSI non-linearities is most noticeable when the PMSM operates in low-speed mode and is not loaded. The second test was performed at the same rotor speed but with a load of 0.4 Nm, corresponding to 50 % of the nominal PMSM load, as shown in Fig. 8. The dead-time period value was set to 1 μs. The learning coefficient α was gradually reduced during ANN training in the range of $0.008 > \alpha > 0$, thereby influencing the speed of convergence of weights during learning.

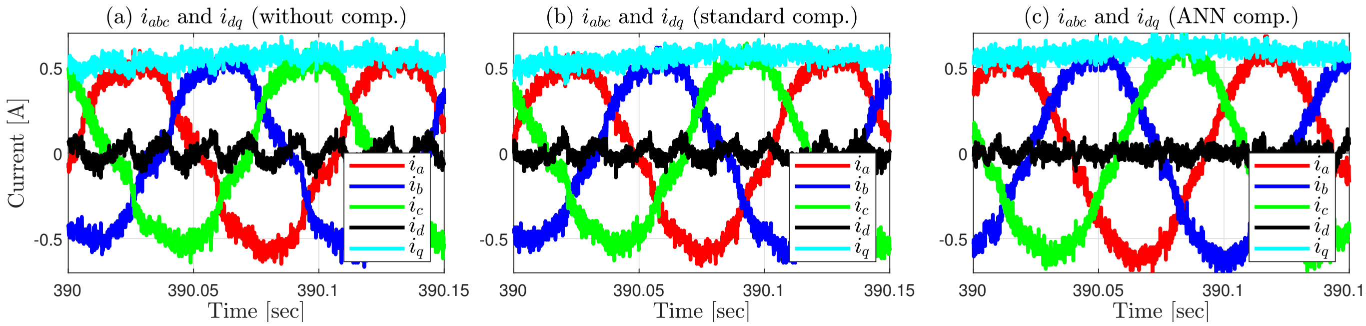

Fig. 7. Phase currents and dq-axes currents without/with standard compensation and with proposed Neural Network compensation without load.

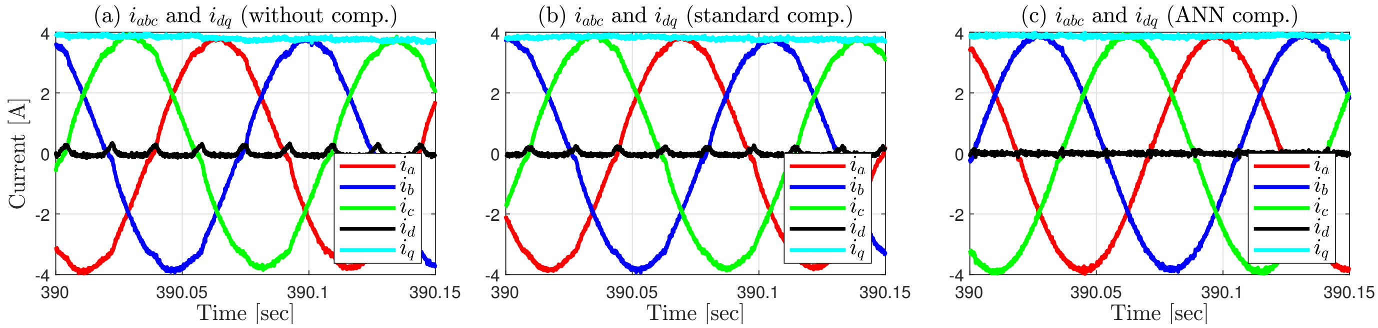

Fig. 8. Phase currents and dq-axes currents without/with standard compensation and with proposed Neural Network compensation with load 0.4 Nm

TABLE I
HARMONIC DISTORTION OF PHASE CURRENTS

Method	**5. harm.** HRI_5 [%] 0 / 0.4 Nm	**7. harm.** HRI_7 [%] 0 / 0.4 Nm	**11. harm.** HRI_{11} [%] 0 / 0.4 Nm	**13. harm.** HRI_{13} [%] 0 / 0.4 Nm
Without	7.21 / 1.84	3.63 / 1.30	1.45 / 1.06	1.06 / 0.81
Stand. comp.	4.69 / 1.30	2.46 / 0.95	0.84 / 0.75	0.49 / 0.60
ANN comp.	1.28 / 0.08	1.01 / 0.10	0.54 / 0.12	0.83 / 0.10

TABLE II
HD INDEX - EXPERIMENTAL RESULTS

Method	**HD index [%]** without	**HD index [%]** with load
Without compensation	8.2679	2.6213
Standard compensation	5.3839	1.8744
Compensation with ANN	1.9041	0.2043

The compensation results achieved at the end of the ANN learning phase are compared with the case without applied compensation voltages and with the results obtained using the standard compensation strategy according to equation (1). Fig. 5 and Fig. 6 show the compensation voltages of the standard compensation strategy and proposed approach with ANN. Without any compensation of the output voltage distortion of the VSI, the phase current waveforms do not have a purely sinusoidal course, as shown in Fig. 7(a) and Fig. 8(a). A zero-current clamping phenomenon also affects the phase currents, as is evident, especially in the case without loading PMSM. This phenomenon complicates current polarity detection in the zero-crossing region, which has a negative impact on the compensation results of the standard method.

The phase current distortion is reflected by the occurrence of odd harmonic components. The 5^{th}, 7^{th}, 11^{th}, and 13^{th} are dominant. These harmonic components are converted into 6^{th} and 12^{th} harmonics in the dq- currents using the Clarke and Park transforms and cause a ripple in the dq- currents. The spectra of the a-phase currents and dq-currents before and after applying the compensation methods are compared in Fig. 9 and Fig. 10; the results were obtained by the fast Fourier transform (FFT).

The results show that applying the compensation voltages in both cases suppresses the 5^{th}, 7^{th}, 11^{th}, and 13^{th} harmonics of the phase currents and 6^{th}, 12^{th} of currents in dq- axis. However, the proposed approach with ANN achieves significantly better results, as is demonstrated by the decrease of the nth harmonic component defined by an index of harmonic ratio of the current (HRI_n) and decrease of the harmonic distortion index (HD). The indexes are defined as follows

$$HRI_n = \frac{I_n}{I_1} \qquad HD = \frac{\sqrt{I_5^2 + I_7^2 + I_{11}^2 + I_{13}^2}}{I_1} \cdot 100 \quad (18)$$

where I_1 and I_n are the magnitude of the fundamental and n^{th} harmonics of the a-phase current. Tab. I and Tab. II present an overview of the achieved results. The proposed ANN reduced the distortion index from 8.27 % to 1.9 % in the case of the unloaded PMSM and from 2.62 % to 0.2 % in the case of the loaded PMSM due to the accurate identification of non-linear compensation voltages.

V. CONCLUSION

The paper focuses on solving the distortion problem of the inverter output voltages. For this purpose, the compensation strategy based on the neural network and knowledge of the three-phase PMSM model structure and its parameters is proposed. The topology of ANN is designed as a compromise that gives the neural network sufficient degrees of freedom

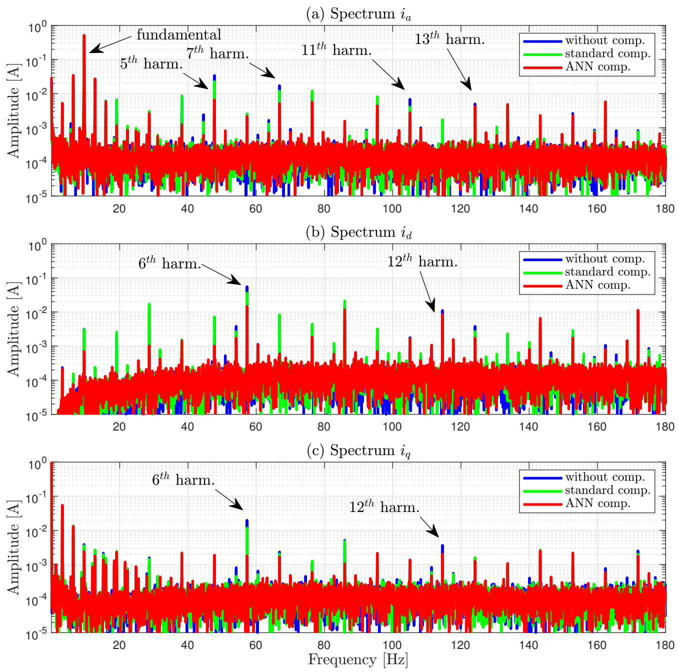

Fig. 9. Spectra of the a-phase currents and dq-axes currents without/with standard compensation and with proposed ANN (without load).

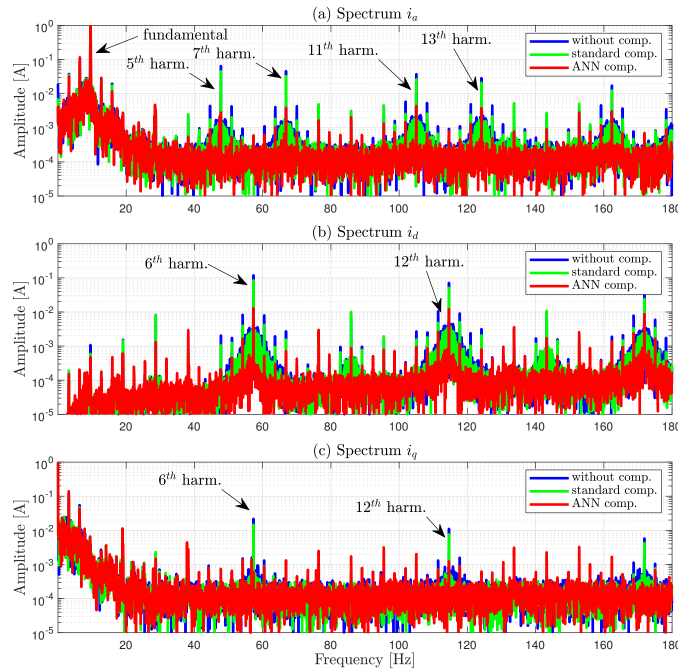

Fig. 10. Spectra of the a-phase currents and dq-axes currents without/with standard compensation and with proposed ANN (load 0.4 Nm).

to approximate the non-linear compensation voltages and can still be implemented in the microcontroller. The proposed strategy is verified on the test bench with vector-controlled PMSM. The presented results show that the proposed approach can effectively suppress adverse effects caused by the dead-time and other VSI non-linearities. The achieved results are significantly better than the suppression of current harmonic distortion by standard compensation strategy. The advantage of this solution is that the neural network can learn to identify the compensation voltage online while the control algorithm is running. Future work will focus on the implementation of the neural network on the separated core of the microcontroller AURIX TC397. It will allow multiple hidden layers and more neurons to be applied in the ANN design. The more effective design of the neural networks for compensation of the VSI non-linearities will be studied.

TABLE III
PARAMETERS OF THE PMSM AND VSI

Name	Symbol	Value	Unit
DC voltage	U_{DC}	50	V
Maximum motor current	i_{max}	8.55	A
Back-EMF constant	λ_m	0.0299	Vs/rad
d-axis inductance	L_d	430	μH
q-axis inductance	L_q	450	μH
Nominal speed	ω_N	1500	rpm
Maximum speed	ω_{max}	12000	rpm
Nominal power	P_N	180	W
Number of pole pairs	p_p	3	-
Switching device voltage drop	V_{sat}	1.5	V
Diode voltage drop	V_d	1.7	V

REFERENCES

[1] Z. Tang and B. Akin, "A New LMS Algorithm Based Deadtime Compensation Method for PMSM FOC Drives," *IEEE Transactions on Industry Applications*, vol. 54, no. 6, pp. 6472–6484, 2018.

[2] W. Dafang, Y. Bowen, Z. Cheng, Z. Chuanwei, and Q. Ji, "A Feedback-Type Phase Voltage Compensation Strategy Based on Phase Current Reconstruction for ACIM Drives," *IEEE Transactions on Power Electronics*, pp. 5031–5043, 2014.

[3] Sam-Young Kim, Wootaik Lee, Min-Sik Rho, and Seung-Yub Park, "Effective Dead-Time Compensation Using a Simple Vectorial Disturbance Estimator in PMSM Drives," *IEEE Transactions on Industrial Electronics*, pp. 1609–1614, 2010.

[4] Y. Zhao, W. Qiao, and L. Wu, "Dead-Time Effect Analysis and Compensation for a Sliding-Mode Position Observer-Based Sensorless IPMSM Control System," *IEEE Transactions on Industry Applications*, pp. 2528–2535, 2015.

[5] J. Ye, Y. Huang, S. Huang, H. Wang, B. Li, J. Xu, and A. Shen, "An Accurate Dead Time Compensation Method for SPWM Voltage Source Inverters," *IEEE Transactions on Power Electronics*, vol. 38, no. 4, pp. 4894–4908, apr 2023.

[6] N. Bedetti, S. Calligaro, and R. Petrella, "Accurate modeling, compensation and self-commissioning of inverter voltage distortion for high-performance motor drives," in *2014 IEEE Applied Power Electronics Conference and Exposition - APEC 2014*. IEEE, 2014, pp. 1550–1557.

[7] Z. Zhang and L. Xu, "Dead-Time Compensation of Inverters Considering Snubber and Parasitic Capacitance," *IEEE Transactions on Power Electronics*, pp. 3179–3187, 2014.

[8] Z. Wu, H. Zhang, K. Liu, W. Hua, G. Zhang, B. Wang, S. Ding, and J. Hang, "Dead-Time Compensation Based on a Modified Multiple Complex Coefficient Filter for Permanent Magnet Synchronous Machine Drives," *IEEE Transactions on Power Electronics*, vol. 36, no. 11, pp. 12 979–12 989, nov 2021.

[9] T. Liu, Q. Li, Q. Tong, Q. Zhang, and K. Liu, "An Adaptive Strategy to Compensate Nonlinear Effects of Voltage Source Inverters Based on Artificial Neural Networks," *IEEE Access*, vol. 8, pp. 129 992–130 002, 2020.

[10] X. Wang and H. Zhu, "Vibration Compensation Control of BPMSM with Dead-Time Effect Based on Adaptive Neural Network Band-Pass Filter," *IEEE Transactions on Power Electronics*, vol. 37, no. 6, pp. 7145–7155, jun 2022.

[11] L. Buchta and O. Bartik, "Dead-time compensation strategies based on kalman filter algorithm for pmsm drives," in *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, 2019, pp. 986–991.