IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 April 2024, accepted 27 April 2024, date of publication 2 May 2024, date of current version 9 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3396170

RESEARCH ARTICLE

Neural Network Compensator-Based Control for Enhancing IPMSM Dynamics and Copper Loss Efficiency for Air Compressor

JIAWEI GUO, LINFENG SUN, TAKAHIRO KAWAGUCHI, (Member, IEEE), AND SEIJI HASHIMOTO, (Member, IEEE)

Division of Electronics and Informatics, Gunma University, Kiryu, Gunma 376-8515, Japan

Corresponding author: Seiji Hashimoto (hashimotos@gunma-u.ac.jp)

This work was supported in part by JSPS KAKENHI under Grant JP22K04150.

ABSTRACT Although significant efforts have been made to enhance industrial air conditioning systems, there are still efficiency and transient response issues in vehicle air conditioning systems using IPMSM compressors. This paper focuses on the neural network compensator-based control in an interior permanent-magnet synchronous motor (IPMSM) to address the occurrence of reduced power copper loss efficiency and degraded velocity response in the motor system when confronted with periodic dynamic disturbances of step signals. This paper encompasses two main objectives: the first objective is to introduce a neural network (NN) compensator to improve the power copper loss efficiency. The NN compensator is developed using the velocity loop and current loop control model equation of an IPMSM, and trained to implement optimal compensation control based on the back propagation algorithm. The second objective is to optimize the dynamic performance of velocity response compared to the traditional maximum torque per ampere (MTPA) current control method under step disturbance and dynamic control conditions by building an experimental system for validation, incorporating both hardware and simulation. Another significant advantage is the low computational load introduced by the neural network compensator, rendering it well-suited for implementation within low-order DSP systems. The results indicate that the neural network compensator surpasses conventional MTPA control method in both simulation and hardware-based implementations concerning power copper loss and velocity response in an IPMSM control system.

INDEX TERMS Interior permanent-magnet synchronous motor (IPMSM), neural network (NN), back propagation (BP), *dq* axis synthesis current control.

I. INTRODUCTION

In recent years, global warming and the depletion of energy resources have become increasingly serious. Moreover, the protection of the global environment and the stable supply of energy are recognized as global challenges. In the context of improving industrial air conditioning systems, dynamic response is equally important. Especially when dealing with situations that require periodic signal control, fast response and reduced errors are crucial aspects when studying transient characteristics. The interior permanent magnet synchronous motor (IPMSM) has gained popularity due to its advantageous features, including high power factor, high power density, and fast dynamic response. Numerous active research efforts have been dedicated to maximizing both efficiency and velocity performance [1], [2]. Many researchers have conducted simulations and experiments studies focused on PMSMs [3], [4], aiming to design velocity control systems and current control systems.

Enhancing the performance of IPMSMs can be achieved not only through hardware design but also by implementing control methods. The MTPA algorithms have their origins in techniques aimed at minimizing motor power losses, which were developed to enhance the efficiency of the

The associate editor coordinating the review of this manuscript and approving it for publication was Mouloud Denai.

motor or the power converter-motor system. This can help improve the efficiency of the motor system and reduce energy waste. However, in the presence of certain specific dynamic disturbances, the power dissipation in the motor could experience an increase, and the velocity response may not be outstanding.

Some online MTPA improvement algorithms have been subject to certain classifications [5] in the latest publications, such as parameters estimation, signal injection, and perturb and observation, have their own limitations. Parameter estimation methods [6], [7] require knowledge of certain parameter relationships and may not be accurate in highly saturated motors. Signal injection methods [8], [9] are effective only in low-noise environments and may introduce additional small losses to the system. Perturb and observation methods [10] are sensitive to changes in the load and are limited to steady-state conditions. All these online MTPA technique methods also require processors with relatively high performance to handle real-time processing. The other major category of offline MTPA algorithms, includes analytical, parameter calculation, simplified formula, and scalar control methods. Analytical methods [11] are generally applied when motor parameters remain constant. Parameter calculation methods [12], [13] require some previous experiments to identify dependencies of motor parameters and involve significant computational effort. Simplified formula methods [14] also require previous knowledge to create approximate functions. Scalar control methods [15] are only suitable for simple applications and are not taken into account for parameter variations.

The characteristics of the compressor determine that each cycle consists of three stages: aspiration, compression, and exhaustion. During operation, different pressure intensities result in varying load torques in the aspiration and exhaustion stages. When the speed regulator fails to counteract these changes, it causes periodic speed fluctuations [16]. Therefore, artificially introducing periodic signals is used to replace the speed fluctuations. In summary, after applying periodic step disturbances, the existing MTPA algorithms result in increased power copper loss and deteriorated dynamic performance [17], [18], [19]. The neural network algorithm, when confronted with periodic step disturbances, adapts to these disturbances through back propagation, thereby achieving higher efficiency and better velocity response compared to the conventional MTPA algorithm. American physicist Hopfield proposed a fully connected neural network, the Hopfield network model [20], and he introduced the concept of network function for the first time. In 1986, Rumelhart and McClelland proposed the back propagation neural network (BPNN) algorithm for multilayer feedforward networks [21], later referred to as the BPNN algorithm, which solved the problems that perceptual machines could not and instilled a strong catalyst for the widespread use of neural networks, and to date, the BPNN remains the most successful neural network learning algorithm. Neural networks have experienced significant advancements and demonstrated remarkable outcomes from theoretical perspectives, particularly regarding their computational power, capability to approximate arbitrary continuous mappings, learning theory, and stability analysis of dynamic networks [22].

This paper develops a novel control strategy: NN compensator-based current control for a PMSM trained using back propagation [23], [24], [25] method. It is worth emphasizing that the NN compensator is trained entirely offline under various simulated environments, enabling it to adapt and respond in real-time to continuously changing motor parameters [26]. This yields some additional key advantages of the NN compensator control method. The first one is the compensator exhibits significant adaptability, eliminating the need for frequent retuning whenever there are minor changes in the motor parameters. Secondly, computational cost at runtime is not so high, making it easy to implement in low-cost hardware. The final one is because of completed offline training, the weights of the NN compensator remain stable during runtime and do not lead to instability. The stability of the well-trained NN compensator for IPMSM control at runtime is validated through a test set and hardware experiment. This paper demonstrates the NN compensator's improved performance, under both simulation and hardware conditions as compared to the conventional control methods.

The rest of this paper is structured as follows. Section II covers the basic equations of the IPMSM and conventional vector control. Section III elaborates on the NN Compensator control proposed method. Section IV shows the simulation and experiment results of the proposed method. Section V gives the conclusion. Finally, this paper concludes with a summary of the main points.

II. CONVENTIONAL VECTOR CONTROL

A. PMSM MODEL

In general, the mathematical model of a PMSM is based on the Park transformation and can be expressed in the dq synchronous rotating coordinate as follows

$$\begin{pmatrix} v_d \\ v_q \end{pmatrix} = \begin{pmatrix} R_s + L_d \cdot \frac{d}{dt} & -\omega_e L_q \\ \omega_e L_d & R_s + L_q \cdot \frac{d}{dt} \end{pmatrix} \begin{pmatrix} i_d \\ i_q \end{pmatrix} + \begin{pmatrix} 0 \\ \omega_e \psi_f \end{pmatrix} \tag{1}$$

where R_s is the resistance of the stator winding; ω_e is the motor electrical rotational velocity; v_d, v_q, i_d, and i_q, are the d and q components of instant stator voltage and current; L_d and L_q are the stator and rotor d and q axis inductances; and ψ_f is the flux linkage produced by the permanent magnet.

The torque balance equation of a PMSM is

$$T_e = J\frac{d\omega_m}{dt} + T_L + B_a\omega_m \tag{2}$$

where J is the inertia of the motor; ω_e is the motor rotational velocity; B_a is the friction coefficient; T_L is the load torque; and T_e is the electromagnetic drive torque. Depending on the type of the PMSM, an interior PMSM, T_e can be expressed

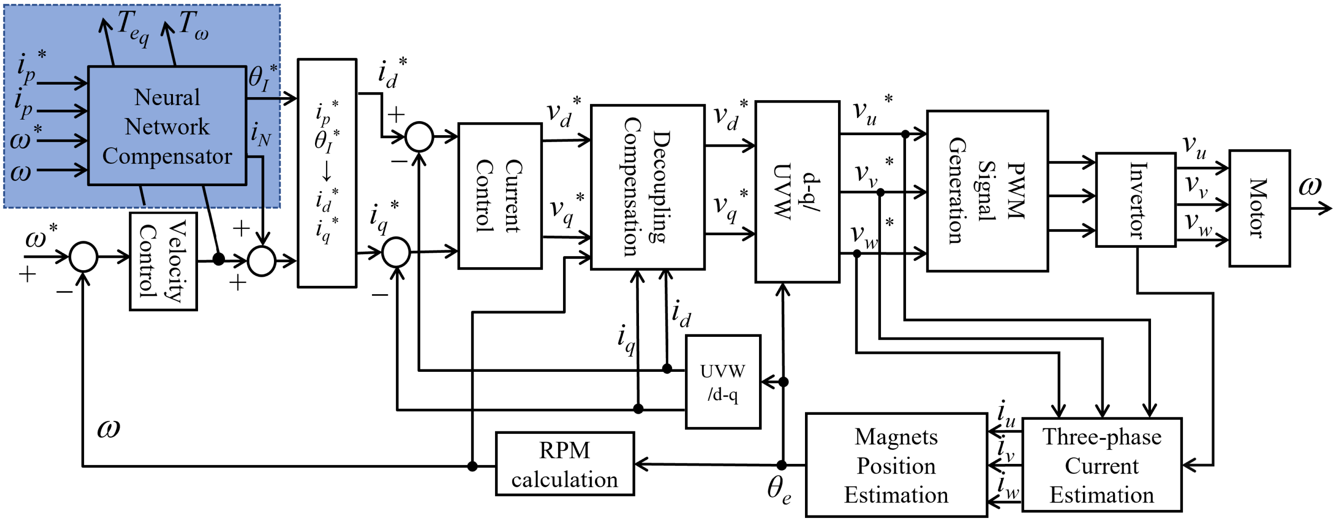


FIGURE 1. System configuration of NN Compensator-based vector control of PMSM.

as follows:

$$T_e = P_n\left[\psi_f i_q + (L_d - L_q) i_d i_q\right] \tag{3}$$

in which P_n represents the number of motor pole pairs. Lastly, the relation between ω_m and ω_e is given by

$$\omega_e = \omega_m \cdot P_n \tag{4}$$

B. CONVENTIONAL MTPA CONTROL

From (3), it can be observed that when $i_d = 0$, there is a partial waste of torque, leading to inefficient current utilization and a decrease in system efficiency. Therefore, a reevaluation of the control strategy is necessary for IPMSM. In order to find the optimal match between current and torque, minimizing current while maximizing torque is essential.

In this situation, i_d can be deduced from the stator current i_s. i_s can be represented as follows

$$i_s = \sqrt{i_d^2 + i_q^2} \tag{5}$$

After that, i_d should be derived

$$i_d = \frac{-\psi_f + \sqrt{\psi_f^2 + 4\left(L_d - L_q\right)^2 i_q^2}}{2\left(L_d - L_q\right)} \tag{6}$$

From the above equations one can notice that the MTPA is a fixed locus as long as the motor parameters remain constant. However, in practical applications, motor parameters often experience uncertain variations due to increases in system operating fluctuations in rotational velocities. This will subsequently lead to a deterioration in the performance of the conventional MTPA algorithm, resulting in lower system efficiency and degraded transient velocity characteristics.

III. NN COMPENSATOR-BASED CONTROL

To address the issues of motor power copper loss and deteriorated dynamic performance in conventional MTPA control under periodic step disturbances, a novel neural network compensator, utilized prior to the generation of d axis and q axis current command values, has been proposed. The reason why neural networks have attracted great interest as a new technology and are increasingly used in the field of control [27], [28], [29], [30], [31], [32], [33] is that, compared to conventional MTPA control techniques, they have the following advantages

(1) It has a nonlinear mapping capability. Neural networks are capable of adequately approximating arbitrarily complex nonlinear relationships, which is of great interest to control researchers.

(2) It possesses adaptive capabilities. Neural networks exhibit strong adaptive capabilities, allowing them to continuously and adaptively adjust network weights to learn and adapt to the dynamic properties of highly uncertain systems.

(3) It possesses a generalization function. Neural networks can process untrained data and derive appropriate solutions corresponding to them. Moreover, they can handle noisy or incomplete data, demonstrating excellent fault tolerance. The generalization capability is highly valuable for many real-world problems, as data collected in real-world scenarios is often tainted by noise or incompleteness.

(4) It is suitable for multivariate systems. The inherent structure of neural networks, which supports multiple inputs and multiple outputs, makes them well-suited for addressing multivariate problems.

A. NN COMPENSATOR WORKFLOW

Neural networks can serve effectively as compensators to optimize the motor control system based on the aforementioned characteristics. The NN compensator is implemented as shown in Fig. 1. The outer velocity loop and the inner current loop remain unchanged. The process of designing a neural network compensator consists of two steps. First, the

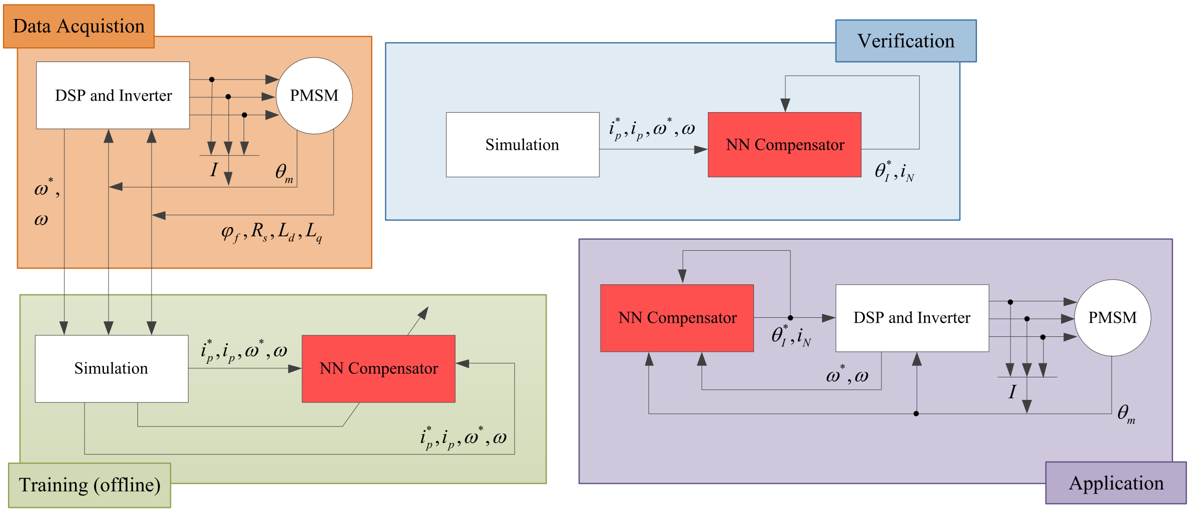


FIGURE 2. Generation steps (data acquisition, training, verification, and application) of the NN compensator.

structure of the neural network should be specified. Second, the NN Compensator needs to be trained, which is similar to the tuning process of a conventional controller.

Fig. 2 shows the overall generation steps of the NN compensator in the motor system, covering the entire process from data acquisition to experimental application. Initially, the motor's parameter data and operational data are obtained from the DSP and PMSM of the experimental platform, constituting the training dataset. Subsequently, various structures of the NN compensator are generated and trained using the training dataset. During the training process, i_p^*, i_p, ω^*, and ω are utilized as inputs and outputs of the compensator. In the verification phase, the structural parameters of the best-performing NN compensator are applied to the experimental setup. Once the training is complete, this NN compensator can be continuously used in experimental operations without the need for additional training, and the computationally intensive backpropagation is only performed in simulation.

B. NN COMPENSATOR STRUCTURE

Fig. 3 shows the whole neural network structure, including the diagrammatic representation of back propagation (BP). The BP structure consists of four different layers, namely an input layer, two hidden layers, and an output layer. The input layer u^T contains four input vectors while y^T contains two output vectors. Two of these inputs comprise the dq current vector synthesis amplitude i_p and its command value $i_p{}^*$, and the other two comprise the motor velocity ω and its command value ω^* in (7). One of the output vectors is the angle of dq axis vector decomposition $\theta_I{}^*$ and another is the compensation quantity of the dq axis synthetic vector current i_N. A two-hidden-layer NN was selected because it generally yields a stronger approximation ability [34] than a one-hidden-layer NN. The number of nodes 10 in each hidden layer v^T and z^T given in (7) was selected via the trial-and-error method. The NNs were also investigated with more hidden layers and more nodes in each hidden layer–but no major improvement was found. The computational burden was considered with fewer layers and fewer nodes in each hidden layer, but compensation performance deteriorated.

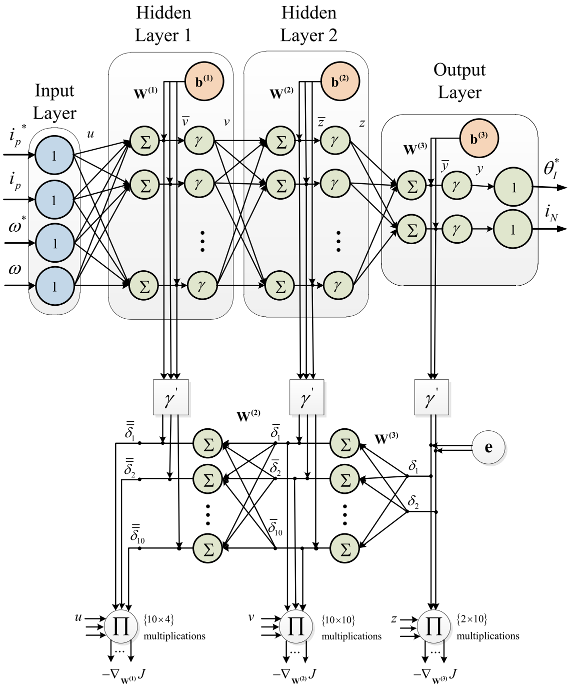


FIGURE 3. Neural networks compensator structure.

$$\begin{cases} u^T \triangleq [i_p^*, i_p, \omega^*, \omega] \\ y^T \triangleq [\theta_I^*, i_N] \\ v^T \triangleq [v_1, v_2, \ldots, v_{10}] \\ z^T \triangleq [z_1, z_2, \ldots, z_{10}] \end{cases} \tag{7}$$

$\mathbf{W^{(k)}}$ and $\mathbf{b^{(k)}}$ represent the weight matric and bias matric for each layer in Fig. 3 respectively. $\bar{v}, v, \bar{y}, y$ are the intermediate vector matrices for the computation of the two middle layers in the neural network. $\bar{z}, z$ are the output vector matrices of the output layer. Their relationship is as shown in (8). γ is the activation function, chosen as the leaky Relu function in (8). For the portion of the Leaky ReLU activation function where the input is less than zero, gradients can still be computed (unlike ReLU, where it is zero). This helps to avoid the gradient direction zig-zagging issue.

$$\begin{cases} \gamma\left(\bar{v}_i\right) = v_i \\ \gamma\left(\bar{y}_k\right) = y_k \\ \gamma\left(\bar{z}_l\right) = z_l \end{cases}$$

$$\gamma\left(x\right) = \begin{cases} x, x > 0 \\ \alpha x, x \leq 0 \end{cases} \tag{8}$$

where $\bar{v}_i, \bar{z}_k, \bar{y}_l$ are the elements of $\bar{v}, \bar{z}, \bar{y}$, respectively. Generally, α is typically around 0.01, here it is set to 0.017.

In Fig. 3, there is also a section that explains the process of back propagation. Two teacher signals have been chosen as references for back propagation, serving as supervision for neural network learning. On the one hand, the error of the *dq* axis vector synthesized current is utilized for compensating the control angle of the *dq* axis current. On the other hand, the output of the velocity loop is selected as the reference for compensating the *dq* axis vector synthesized current. These two terms are defined by (9). The performance criterion J is then defined as (10) where the summation is carried out over all patterns in a given set S. The gradient of the performance function with respect to $\mathbf{W}$ is computed as $\nabla_{\mathbf{W}}$ and $\mathbf{W}$ is adjusted along the negative gradient as (11). The bias gradient updates are also the same as (11). From a strictly theoretical perspective, the adjustment of parameters should involve the computation of the gradient of J in parameter space. However, in common practice, parameter adjustments are made at each time step, considering the error at that specific moment and a small step size η.

$$\begin{cases} e^T \triangleq [T_{e_q}, T_\omega] \\ T_{e_q} = i_p^* - i_p, T_\omega = i_{pf}^* \end{cases} \tag{9}$$

$$J = \sum_{s} \|\mathbf{e}\|^2 \tag{10}$$

$$\begin{aligned} \mathbf{W^{(k)}} &= \mathbf{W^{(k)}} - \eta \nabla_{\mathbf{W^{(k)}}} J \\ \mathbf{b^{(k)}} &= \mathbf{b^{(k)}} - \eta \nabla_{\mathbf{b^{(k)}}} J \end{aligned} \tag{11}$$

where ∇ called nabla, is an operator used in mathematics (particularly in vector calculus) as a vector differential operator.

Next, it is evident that $u, v, z, \gamma'(\bar{v}), \gamma'(\bar{z})$ and $\gamma'(\bar{y})$, as well as the error vector matric $\mathbf{e}$, are used in the computation of the back propagation gradient, where $\gamma'(x)$ is the derivatives of the $\gamma\left(x\right)$ with respect of x. $\delta, \bar{\delta}$ and $\bar{\bar{\delta}}$ are the intermediate variables at each layer in back propagation calculations respectively. $\{2 \times 10\}$, $\{10 \times 10\}$ and $\{10 \times 4\}$ multiplications are used to compute the partial derivatives with respect to the elements of each layer in $\mathbf{W}$. The weight matrices in the derivative computation network exhibit an identical structure to those in the original network, although the signal flow is reversed. This rationale supports the adoption of the term "back propagation."

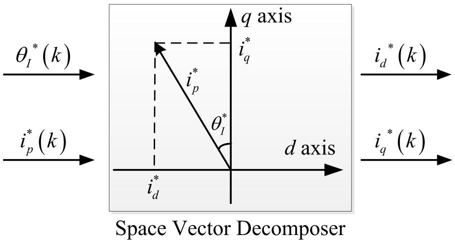


FIGURE 4. Space vector decomposer.

TABLE 1. PMSM data used in experiment.

Parameters	Value	Unit
Nominal Speed	1000	$\min^{-1}$
Permanent magnet flux ψ_f	0.052	Wb
Inductance in q axis L_q	0.002	mH
Inductance in d axis L_d	0.0012	mH
Stator copper resistance R_s	0.343	Ω
Pole pairs	4	—

Finally, the space vector decomposer was shown in Fig. 4, they are used for re-computing the *dq* axis current by (12).

$$\begin{aligned} i_d^* &= -i_p^* \sin\theta_I^* \\ i_q^* &= i_p^* \cos\theta_I^* \end{aligned} \tag{12}$$

C. NN OFFLINE TRAINING MECHANISM

The NN Compensator is trained to approximate optimal control by using gradient descent to adjust the weights of the NN until the most suitable output is obtained. As shown in Fig. 1, the NN compensator receives the *dq* synthesized current and angular velocity feedback signal from the IPMSM. Thus, the output control action of the NN at time step k changes the output current and angular velocity of the IPMSM at time step $k+1$, the output motor current and angular velocity then change NN inputs at time step $k+1$ and then, the NN output control action at time step $k+1$ is modified. This recursive process continues, incorporating the IPMSM using the compensation values and NN similar to a recurrent NN. This neural network compensator is shown in Fig. 5, unrolled in time, illustrating the neural network corrects and compensates for the state of the motor and how the IPMSM and the NN compensator interact with each other.

IV. SIMULATION AND EXPERIMENTAL RESULTS ANALYSIS

To demonstrate the effectiveness of the proposed IPMSM system control compensator, the NN Compensator of the IPMSM control system has been considered in two IPMSM cases: one for simulation and one for hardware experiment.

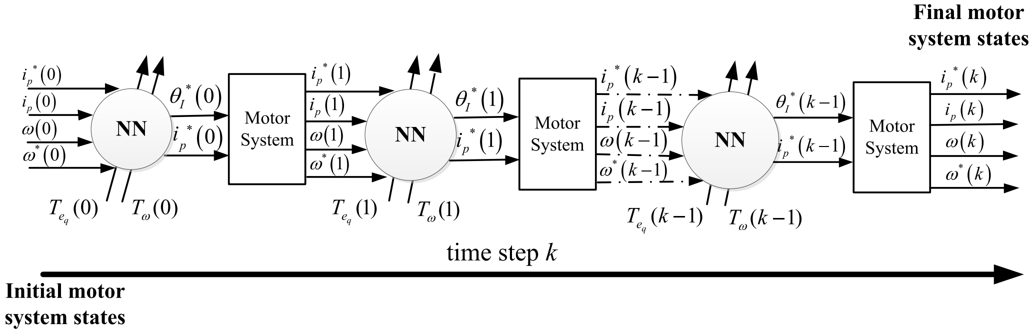


FIGURE 5. Neural networks compensator feedback loop.

TABLE 2. PMSM partial parameters changed data used in simulation.

Parameters	Value	Unit
Nominal Speed	1000	min^{-1}
Permanent magnet flux ψ_f	0.0509	Wb
Inductance in q axis L_q	0.0019	mH
Inductance in d axis L_d	0.0011	mH
Stator copper resistance R_s	0.343	Ω
Pole pairs	4	/

The simulation case uses the parameters of an IPMSM that are typically abstracted from the laboratory IPMSM. The hardware experiment is based on a laboratory IPMSM, which is used for an air compressor and also mainly for the purpose of experimental validation.

A. SIMULATION ENVIRONMENT

To align more closely with the experimental conditions, the simulation emulated variations in motor parameters that were expected to arise during the experiments. Table 1 provides all the nominal motor parameters from experiments, while Table 2 presents partial data after changes to motor parameters. Fig. 6 shows the step disturbance introduced in the simulation. Fig. 7 demonstrates the variations of certain parameters of the neural network compensator during the learning process. For instance, Fig. 7(a) illustrates the teacher signal, while Fig. 7(b) presents a randomly selected layer weight. After applying periodic step disturbance signals, both the teacher signal and layer weight gradually adapt to the disturbances and eventually stabilize. These findings indicate that the neural network compensator is capable of learning and adapting to disturbances and motor parameters changes, resulting in performance optimization in the IPMSM control system.

B. SIMULATION ANALYSIS

Initially, by introducing a single step disturbance to stabilize the NN compensator's training, testing was carried out. The outcomes for the d axis current i_d and the dq vector-synthesized results i_p as depicted in Fig. 8, were acquired. In comparison to the traditional MTPA method, there is a certain level of optimization.

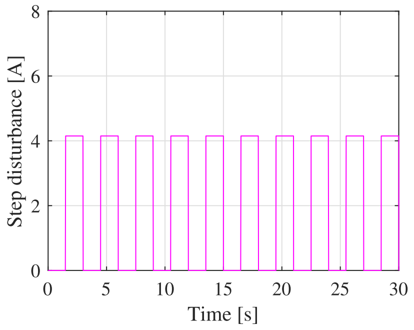


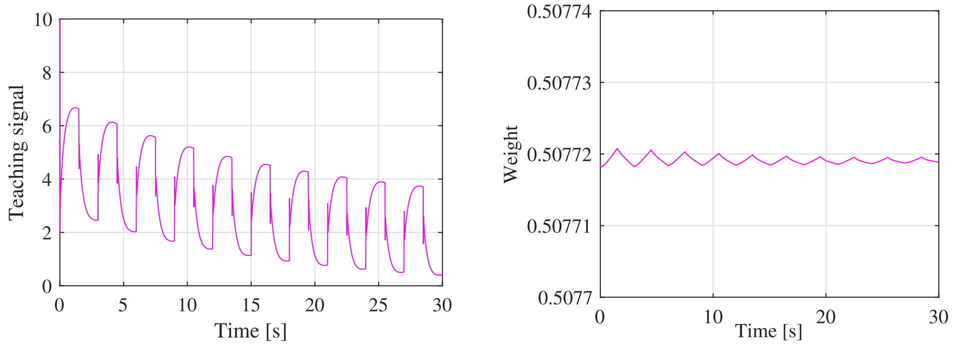
FIGURE 6. Step disturbance introduced in simulation.

Fig. 9 shows the velocity response and power copper loss efficiency of the IPMSM control system after incorporating the neural network compensator. The d axis current is an important variable for current control, as shown in Fig. 9(a). The dq axis vector-synthesized current significantly affects the IPMSM control system's power copper loss efficiency, as demonstrated in Fig. 9(b) and Fig. 9(c). After adapting for the initial few periods, Fig. 9(b) illustrates the NN compensator has a clear learning ability in response to disturbances, with the power copper loss efficiency gradually increasing as the dq axis vector-synthesized current i_p decreases. In contrast, the traditional MTPA algorithm lacks any ability to adapt to error signals, fluctuating with the disturbances in the error signal. Additionally, in order to enhance the dynamic performance of the system, the neural network compensator adapts to the disturbances, resulting in remarkable improvements in the velocity response ω^*, as shown in Fig. 9(d). Both the overshoot and settling time experience substantial enhancements.

After undergoing disturbances many times and motor parameters changing, these findings indicate that the neural network compensator effectively mitigates disturbances and optimizes the IPMSM control system's performance.

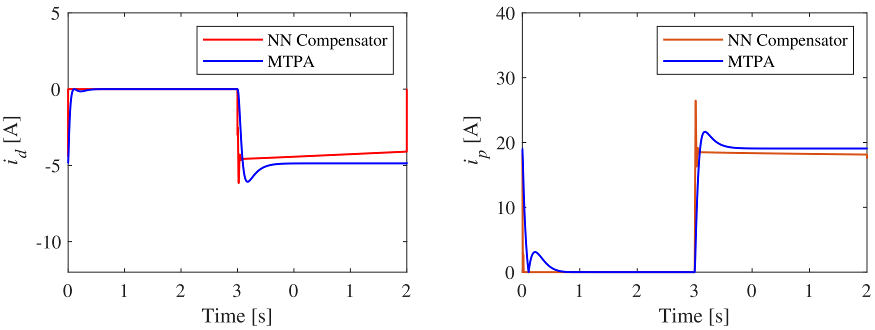
C. EXPERIMENTAL ENVIRONMENT

After evaluating the system performance by simulation, the motor control algorithm was compiled and deployed to the experimental IPMSM drive control system, as presented in Fig. 10. Moreover, to analyze the system, the system data



(a) (b)

FIGURE 7. The learning parameters of the neural network compensator. (a) teaching signal. (b) a sample of weight.



(a) (b)

FIGURE 8. Simualtion results for comparison of currents after applying step disturbance signal one-time. (a) *d* axis current. (b) *dq* axis synthesized current.

were transmitted to MATLAB Simulink by the SCI function, integrated into the DSP with a sampling frequency of 2 kHz. The experimental system involves an IPMSM coupled to a generator, an inverter, a control circuit, and a DSP RH850/C1M-A2. The DSP RH850/C1M-A2 is equipped with peripheral functions ideal for motor control of traction inverter applications for HEV/EV, such as an R/D converter (RDC3A) that converts the output signal of the position sensor into digital angle data, and a motor control unit (EMU3) that can operate in parallel. Moreover, it has a 320 MHz CPU, 2 MB Flash, 16-bit/12-bit ADCs, 12-bit DACs, SubCPU, etc. The parameters of PI controller are set as $K_p = 0.07, K_i = 0.0001$. The membership function's values are set the same as the simulation configuration except for the velocity. Furthermore, due to certain voltage limitations imposed by the experimental platform, the amplitude of the step disturbance signal introduced in the experiment has been reduced by a factor of 10, while other characteristics remain unchanged.

D. EXPERIMENTAL ANALYSIS

After the motor goes through the startup mode, a step disturbance is introduced during steady-state operation using the NN compensator. MTPA control is an optimization based on the $i_d = 0$ control method. However it lacks good dynamic performance in motor parameters change and is challenging to perform precise complex calculations in engineering, thus unable to fully exploit the motor's optimal performance. Firstly, Fig. 11 represents the motor's current performance with a single-step disturbance applied. It is clearly shown that the neural network compensator compensates for the d axis current based on the step disturbance in Fig. 11(a), and also optimizes the recovery of the dq axis synthesized current to a steady-state trend in Fig. 11(b).

TABLE 3. Power copper loss calculations from the experiment platform.

Control Method type	Copper loss (W)	Copper loss (%)
MTPA	10.1	100
NN Compensator	9.37	93

Second, in the experiment, periodic step disturbances were applied similarly to the simulation, but with a longer disturbance period of 10 seconds. However, consistent results were obtained, similar to those observed in the simulation. The NN compensator underwent learning during each application of the disturbance. The trends in the changes of the d axis current and the synthesized dq axis current are essentially the same. Fig. 12(a) illustrates the specific changes in the learning adaptation step response error of the d axis current. The d axis current gradually stabilizes from the initial maximum value of -0.549 A to -0.231 A.

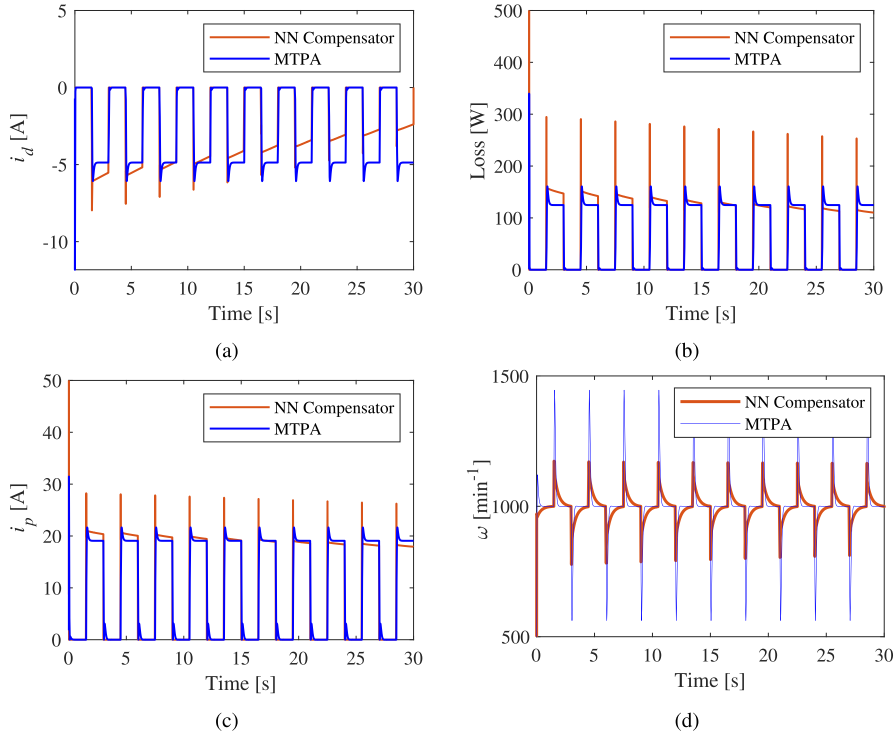


FIGURE 9. Simulation results for comparison of efficiency and dynamics after applying step disturbance signal. (a) *d* axis current. (b) copper loss. (c) *dq* axis synthesized current. (d) velocity response.

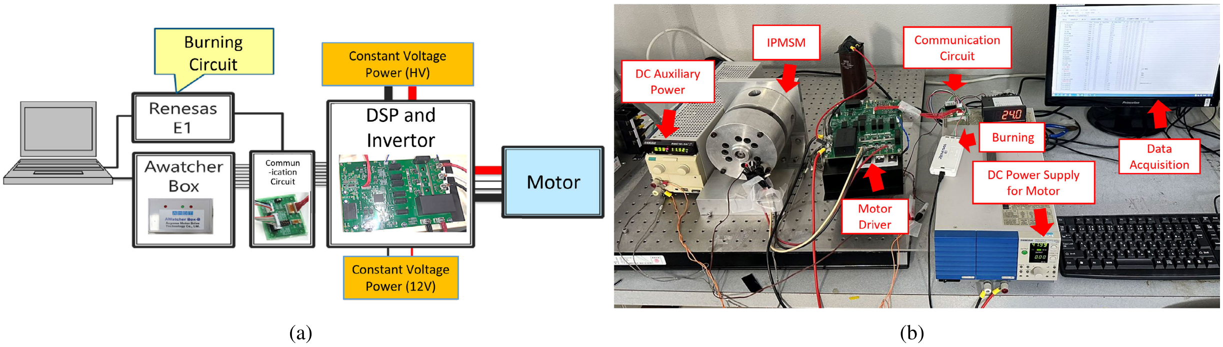


FIGURE 10. Experiment test bench. (a) block diagram. (b) experiment hardware platform.

Additionally, within each cycle, the d-axis current also adapts to changes in the step disturbance applied during one cycle, transitioning from -0.216 A to -0.143 A. From Fig. 12(c), the first kinds of learning occurred during the entire process from the 0 s to the 20 s, resulting in a significant decreasing trend in the overall dq axis synthesized current. The maximum value of the current gradually decreases from 7.69 A to around 5.67 A over each period and stabilizes. One of the second kinds of learning occurred around the 30 s, within one period of applied disturbance, exhibiting a detailed gradual recovery trend from 5.58 A to 4.69 A of the dq axis synthesized current to steady-state. Similarly, the power copper loss and dq synthesized currents exhibit the same changing trend in Fig. 12(b).

In Fig. 12(d), after going through the startup state, the overall overshoot at each period in the velocity response is smaller compared to the conventional MTPA control, and the velocity response fluctuations are reduced, resulting in a more stable overall performance. It can be observed that the NN compensator optimizes the current performance of the motor and dynamically improves the velocity response, enabling the motor control system to better cope with step disturbances. Although not as pronounced as in the simulation, it still exhibits significant optimization.

The detailed data presented in Table 3 reveals an average improvement of approximately 7% in the motor's power copper loss efficiency after adapting to disturbances and motor parameters changed. This indicates that the neural network compensator effectively compensates for disturbances

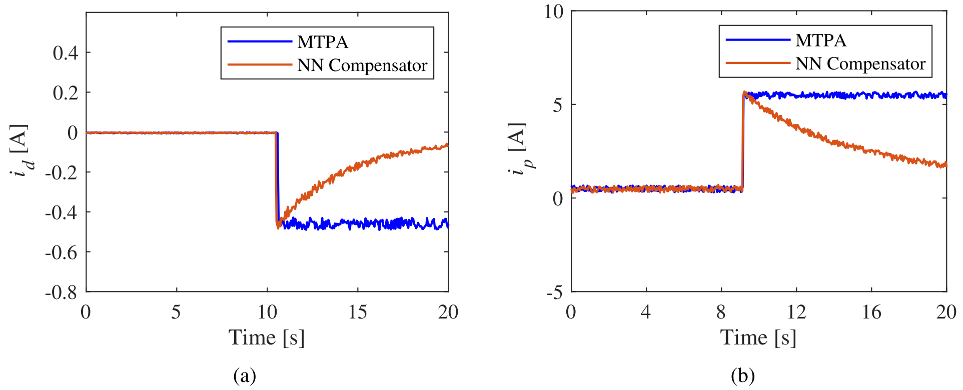


FIGURE 11. Experiment results for comparison of currents after applying step disturbance signal one-time. (a) *d* axis current. (b) *dq* axis synthesized current.

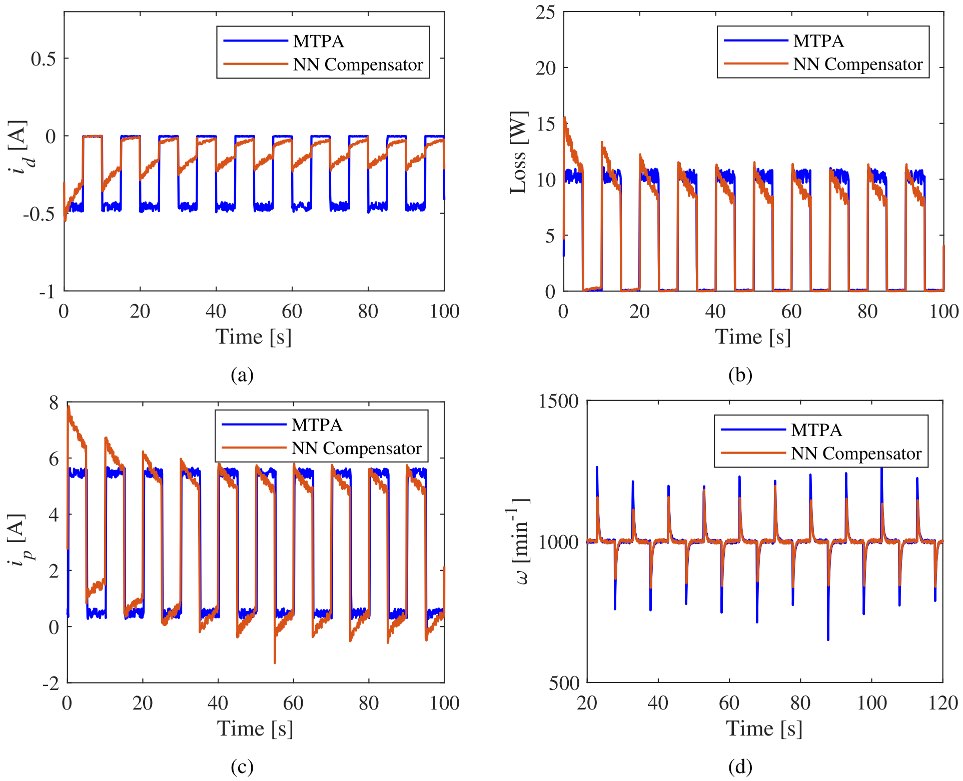


FIGURE 12. Experiment results for comparison of efficiency and dynamics after applying step disturbance signal. (a) *d* axis current. (b) copper loss (c) *dq* axis synthesized current. (d) velocity response.

in the *dq* axis current of the motor under step disturbance conditions, resulting in enhancing the power copper loss efficiency.

V. CONCLUSION

IPMSMs find wide application in the field of electric drives, particularly in electric vehicles and air compressors. This paper proposes a neural network compensator control method to overcome the limitations of MTPA control approaches. It describes how to utilize the neural network compensator to achieve dynamic improvements and enhance power efficiency under step disturbance, with the neural network trained offline for experimental use. The proposed control algorithm was designed in MATLAB Simulink and deployed to the real-time platform, based on a RH850/C1M-A2.

Compared to MTPA control, the neural network compensator exhibits the fastest response speed, lowest overshoot, and overall superior performance. Furthermore, since the neural network is trained under step disturbance, it demonstrates remarkable performance when sampling time changes and system parameters become difficult to identify, especially in hardware experiment conditions.

Under hardware experimental conditions, the conventional MTPA control method usually requires readjustment when disturbance signals are applied and motor parameters real-time change. In contrast, the neural network compensator after adapting to disturbances retains great dynamics performance and higher copper loss efficiency under step disturbance conditions, making it feasible to implement the neural network compensator in practical IPMSM environments.

REFERENCES

[1] C. Lai, G. Feng, J. Tian, Z. Li, Y. Zuo, A. Balamurali, and N. C. Kar, ''PMSM drive system efficiency optimization using a modified gradient descent algorithm with discretized search space,'' *IEEE Trans. Transport. Electrific.*, vol. 6, no. 3, pp. 1104–1114, Sep. 2020, doi: 10.1109/TTE.2020.3004463.

[2] L. Yuan, Y. Jiang, L. Xiong, and P. Wang, ''Sliding mode control approach with integrated disturbance observer for PMSM speed system,'' *CES Trans. Electr. Mach. Syst.*, vol. 7, no. 1, pp. 118–127, Mar. 2023, doi: 10.30941/CESTEMS.2023.00009.

[3] C. Jo, J.-Y. Seol, and I.-J. Ha, ''Flux-weakening control of IPM motors with significant effect of magnetic saturation and stator resistance,'' *IEEE Trans. Ind. Electron.*, vol. 55, no. 3, pp. 1330–1340, Mar. 2008, doi: 10.1109/TIE.2007.910524.

[4] A. Kamalaselvan and S. L. Prakash, ''Modeling simulation and analysis of closed loop speed control of PMSM drive system,'' in *Proc. Int. Conf. Circuits, Power Comput. Technol. (ICCPCT)*, Nagercoil, India, Mar. 2014, pp. 692–697, doi: 10.1109/ICCPCT.2014.7055050.

[5] F. Tinazzi, S. Bolognani, S. Calligaro, P. Kumar, R. Petrella, and M. Zigliotto, ''Classification and review of MTPA algorithms for synchronous reluctance and interior permanent magnet motor drives,'' in *Proc. 21st Eur. Conf. Power Electron. Appl. (EPE ECCE Europe)*, Sep. 2019, pp. P.1–P.10.

[6] S. J. Underwood and I. Husain, ''Online parameter estimation and adaptive control of permanent-magnet synchronous machines,'' *IEEE Trans. Ind. Electron.*, vol. 57, no. 7, pp. 2435–2443, Jul. 2010, doi: 10.1109/TIE.2009.2036029.

[7] H.-W. Sim, J.-S. Lee, and K.-B. Lee, ''On-line parameter estimation of interior permanent magnet synchronous motor using an extended Kalman filter,'' *J. Electr. Eng. Technol.*, vol. 9, no. 2, pp. 600–608, Mar. 2014.

[8] S. Kim, Y.-D. Yoon, S.-K. Sul, and K. Ide, ''Maximum torque per ampere (MTPA) control of an IPM machine based on signal injection considering inductance saturation,'' *IEEE Trans. Power Electron.*, vol. 28, no. 1, pp. 488–497, Jan. 2013, doi: 10.1109/TPEL.2012.2195203.

[9] H. Kim, J. Hartwig, and R. D. Lorenz, ''Using on-line parameter estimation to improve efficiency of IPM machine drives,'' in *Proc. IEEE 33rd Annu. IEEE Power Electron. Spec. Conf.*, vol. 2, 2002, pp. 815–820.

[10] E. Daryabeigi, H. Abootorabi Zarchi, G. R. Arab Markadeh, J. Soltani, and F. Blaabjerg, ''Online MTPA control approach for synchronous reluctance motor drives based on emotional controller,'' *IEEE Trans. Power Electron.*, vol. 30, no. 4, pp. 2157–2166, Apr. 2015.

[11] C.-T. Pan and S.-M. Sue, ''A linear maximum torque per ampere control for IPMSM drives over full-speed range,'' *IEEE Trans. Energy Convers.*, vol. 20, no. 2, pp. 359–366, Jun. 2005.

[12] I. Jeong, B.-G. Gu, J. Kim, K. Nam, and Y. Kim, ''Inductance estimation of electrically excited synchronous motor via polynomial approximations by least square method,'' *IEEE Trans. Ind. Appl.*, vol. 51, no. 2, pp. 1526–1537, Mar. 2015.

[13] T. Inoue, Y. Inoue, S. Morimoto, and M. Sanada, ''Mathematical model for MTPA control of permanent-magnet synchronous motor in stator flux linkage synchronous frame,'' *IEEE Trans. Ind. Appl.*, vol. 51, no. 5, pp. 3620–3628, Sep. 2015.

[14] M.-S. Huang, K.-C. Chen, and C.-H. Chen, ''Modeling and analysis of IPM synchronous motor under six step voltage control by Fourier series,'' in *Proc. IECON 41st Annu. Conf. IEEE Ind. Electron. Soc.*, Nov. 2015, pp. 002451–002455.

[15] Z. Tang, X. Li, S. Dusmez, and B. Akin, ''A new V/f-based sensorless MTPA control for IPMSM drives,'' *IEEE Trans. Power Electron.*, vol. 31, no. 6, pp. 4400–4415, Jun. 2016.

[16] W. Zhaodong, W. Xiaoqin, Y. Linru, and M. Fankun, ''An effective optimized strategy on speed fluctuation suppression for air conditioner compressor,'' in *Proc. 36th Chin. Control Conf. (CCC)*, Dalian, China, Jul. 2017, pp. 4935–4939, doi: 10.23919/ChiCC.2017.8028134.

[17] W. Lv, K. Huang, H. Wu, X. Mo, and M. Shen, ''A dynamic compensation method for time delay effects of high-speed PMSM sensorless digital drive system,'' in *Proc. 22nd Int. Conf. Electr. Mach. Syst. (ICEMS)*, Harbin, China, Aug. 2019, pp. 1–5, doi: 10.1109/ICEMS.2019.8921970.

[18] A. Anand, R. Gandhi, D. Bhattacharya, and R. Roy, ''Dynamic analysis of vector controlled PMSM with constant torque angle control strategy using artificial neural network,'' in *Proc. 5th Int. Conf. Energy, Power Environment: Towards Flexible Green Energy Technol. (ICEPE)*, Shillong, India, Jun. 2023, pp. 1–5, doi: 10.1109/icepe57949.2023.10201640.

[19] J. Yu, P. Shi, W. Dong, B. Chen, and C. Lin, ''Neural network-based adaptive dynamic surface control for permanent magnet synchronous motors,'' *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 640–645, Mar. 2015, doi: 10.1109/TNNLS.2014.2316289.

[20] J. J. Hopfield, ''Neural networks and physical systems with emergent collective computational abilities,'' *Proc. Nat. Acad. Sci. USA*, vol. 79, no. 8, pp. 2554–2558, Apr. 1982, doi: 10.1073/pnas.79.8.2554.

[21] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*. Cambridge, MA, USA: MIT Press, 1986, pp. 2554–2558.

[22] G. Ding, ''Efficient-optimization strategy based on back propagation for vector controlled claw pole motor drives,'' in *Proc. Asia–Pacific Power Energy Eng. Conf.*, Wuhan, China, Mar. 2011, pp. 1–4.

[23] F. J. Pineda, ''Generalization of back-propagation to recurrent neural networks,'' *Phys. Rev. Lett.*, vol. 59, no. 19, pp. 2229–2232, Nov. 1987, doi: 10.1103/physrevlett.59.2229.

[24] D. Sun and J. Zhu, ''Efficient-optimization strategy based on back propagation for vector controlled PMSM drives,'' in *Proc. Int. Conf. Intell. Comput. Technol. Automat.*, Changsha, China, 2010, pp. 196–199.

[25] H. Lin, A. Marquez, F. Wu, J. Liu, H. Luo, L. G. Franquelo, and L. Wu, ''MRAS-based sensorless control of PMSM with BPN in prediction mode,'' in *Proc. IEEE 28th Int. Symp. Ind. Electron. (ISIE)*, Vancouver, BC, Canada, Jun. 2019, pp. 1755–1760.

[26] D. V. Prokhorov, L. A. Feldkarnp, and I. Y. Tyukin, ''Adaptive behavior with fixed weights in RNN: An overview,'' in *Proc. Int. Joint Conf. Neural Netw.*, 2002, pp. 2018–2023.

[27] M. J. Willis, G. A. Montague, C. Di Massimo, M. T. Tham, and A. J. Morris, ''Artificial neural networks in process estimation and control,'' *Automatica*, vol. 28, no. 6, pp. 1181–1187, Nov. 1992, doi: 10.1016/0005-1098(92)90059-o.

[28] D. Sbarbaro-Hofer, D. Neumerkel, and K. Hunt, ''Neural control of a steel rolling mill,'' *IEEE Control Syst. Mag.*, vol. 13, no. 3, pp. 69–75, Jun. 1993, doi: 10.1109/37.214948.

[29] K. S. Narendra and K. Parthasarathy, ''Identification and control of dynamical systems using neural networks,'' *IEEE Trans. Neural Netw.*, vol. 1, no. 1, pp. 4–27, Mar. 1990, doi: 10.1109/72.80202.

[30] J. R. Noriega and H. Wang, ''A direct adaptive neural-network control for unknown nonlinear systems and its application,'' *IEEE Trans. Neural Netw.*, vol. 9, no. 1, pp. 27–34, Jan. 1998, doi: 10.1109/72.655026.

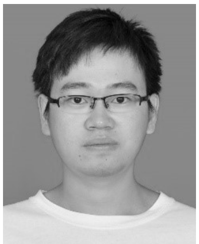
[31] P. S. Sastry, G. Santharam, and K. P. Unnikrishnan, ''Memory neuron networks for identification and control of dynamical systems,'' *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 306–319, Mar. 1994, doi: 10.1109/72.279193.

[32] M. J. Willis, G. A. Montague, C. Di Massimo, A. J. Morris, and M. T. Tham, ''Artificial neural networks and their application in process engineering,'' in *IEE Colloquium on Neural Networks for Systems: Principles and Applications*, 1991, pp. 1–7.

[33] X. Cui and K. G. Shin, ''Direct control and coordination using neural networks,'' *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 3, pp. 686–697, 1993, doi: 10.1109/21.256542.

[34] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural Network Design*. Boston, MA, USA: PWS, 2002.

JIAWEI GUO received the B.S. degree in electrical engineering from Changchun University of Science and Technology, in 2020, and the M.S. degree in electrical engineering from Gunma University, Kiryu, Japan, in 2023, where he is currently pursuing the Ph.D. degree. His current research interests include PMSM motor drives, compensation, and AI-based control.



LINFENG SUN received the B.Sc. and M.Sc. degrees in electrical engineering from Yangzhou University, Yangzhou, China, in 2018 and 2021, respectively. He is currently pursuing the Ph.D. degree with Gunma University, Kiryu, Japan. His current research interests include digital control of power electronics, microgrids, PMSM motor drives, and AI-based control. He is a Student Member of IEEJ. He was honored with the Best Paper Award from the International Conference I-SEEC&ICTSS, in 2022, and the Outstanding Academic Master's Thesis Award from Yangzhou University, in 2022.

TAKAHIRO KAWAGUCHI (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in engineering from Keio University, Tokyo, Japan, in 2011, 2013, and 2017, respectively. From 2013 to 2015, he was with the Toshiba Research and Development Center. From 2017 to 2019, he was a Researcher with the Department of Systems and Control Engineering, School of Engineering, Tokyo Institute of Technology, Tokyo. From 2019 to 2020, he was a specially appointed an Assistant Professor with the Department of Systems and Control Engineering, Tokyo Institute of Technology. He is currently an Assistant Professor with the Division of Electronics and Informatics, School of Science and Technology, Gunma University, Gunma, Japan. His research interests include system identification theory and the application of machine learning techniques. He is a member of the Society of Instrument and Control Engineers and the Institute of System, Control, and Information Engineers.

SEIJI HASHIMOTO (Member, IEEE) received the M.E. and Ph.D. degrees in electrical and electronic engineering from Utsunomiya University, Japan, in 1996 and 1999, respectively. He joined with the Department of Mechanical Engineering, Oyama National College of Technology. Since 2002, he has been a Research Associate with the Department of Electronic Engineering, Gunma University, where he is currently a Professor with the Program of Intelligence and Control. He has consulted for companies in control and energy applications and has been a Visitor Professor in China. He has more than 100 refereed articles and more than ten granted and pending patents. His research interests include system identification, motion control, AI-based control and diagnosis, energy regeneration, and its application to industrial fields. He is a member of IEEJ and SICE.