





ANN-Based Improved Direct Torque Control of Open-End Winding Induction Motor

Kaif Ahmed Lodi , Member, IEEE, Abdul R. Beig , Senior Member, IEEE, Khaled Ali Al Jaafari , Senior Member, IEEE, and Zeyar Aung , Senior Member, IEEE

Abstract—This article presents an improved artificial neural network (ANN)-based direct torque control (DTC) algorithm for open-end winding induction motor (OEWM) sensorless drives. The ANN replaces the traditional look-up table and serves as an efficient data storage system, allowing for optimal voltage vector selection based on small values of torque error and flux error for every angular position of the stator flux vector. By replacing the hysteresis controllers, improved torque and flux tracking are achieved with improved transient response. The proposed ANN-based DTC is verified through simulation as well as experimental tests. The experimental results and comparative results are presented.

Index Terms—Artificial neural network (ANN), direct torque control (DTC), open-end winding induction motor (OEWM), torque ripple.

NOMENCLATURE

r_s, r_r	Stator, rotor resistance (Ω).
l_s, l_r	Stator, rotor inductance (H).
i_s, i_r	Stator, rotor current space vector (A).
ψ_s, ψ_r	Stator, rotor flux space vector (Wb).
v_s	Stator voltage space vector (V).
ω_r	Speed of the motor (electrical) (rad/s).
ω	Speed of the motor (mechanical) (rad/s).
ω_s	Synchronous speed (electrical) (rad/s).
l_m	Mutual inductance (H).
P	Number of poles.
J	Moment of inertia ($kg \cdot m^2$).
δ	Angle between ψ_s and ψ_r (rad).
θ_s	Angle of ψ_s (rad).
T_e	Torque developed by motor ($N \cdot m$).
T_L	Load torque ($N \cdot m$).

Manuscript received 29 June 2023; revised 14 October 2023 and 22 December 2023; accepted 13 January 2024. Date of publication 8 February 2024; date of current version 19 June 2024. This work was supported by the Advanced Power and Energy Center (APEC), Khalifa University, Abu Dhabi, United Arab Emirates, under Grant RC2-2018-06. (Corresponding author: Kaif Ahmed Lodi.)

Kaif Ahmed Lodi, Abdul R. Beig, and Khaled Ali Al Jaafari are with the Department of Electrical Engineering Advanced Power and Energy Center, Khalifa University, Abu Dhabi 127788, UAE (e-mail: kaif.lodi@ku.ac.ae; balanthi.beig@ku.ac.ae; khaled.aljaafari@ku.ac.ae).

Zeyar Aung is with the Department of Computer Science, Khalifa University, Abu Dhabi 127788, UAE (e-mail: zeyar.aung@ku.ac.ae).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TIE.2024.3357865>.

Digital Object Identifier 10.1109/TIE.2024.3357865

k_e	Torque coefficient of dynamic torque error.
k_ω	Speed coefficient of dynamic torque error.
T_s	Sampling interval (s).
s, r	Subscript for stator, rotor quantities.
*	Superscript for reference signals.
α, β	Subscript for stationary reference axes.

I. INTRODUCTION

DIRECT torque control (DTC) is the preferred algorithm for drives when fast transient response is required [1]. However, the conventional DTC (CDTC) has few limitations, such as variable switching frequency, poor torque ripple, poor flux ripple, and flux instability at low speeds [2], [3], [4], [5]. DTC makes use of hysteresis controllers with finite error bands for torque and flux control. The applied voltage vectors are primarily used to keep the errors within the error band rather than tracking them accurately to zero [3], [6]. The fixed voltage vectors within the sector do not guarantee optimal voltage vector selection under all operating conditions.

Researchers have attempted to overcome these issues associated with CDTC through various techniques. One such approach is to use space vector pulsewidth modulation with DTC (SVMDTC), but this requires additional controllers, such as PI controllers for flux and torque tracking [7], [8], dead-beat control [9], and sliding-mode control [10]. Although a constant switching frequency is achieved with SVMDTC, it comes at the cost of increased computational complexity, greater dependency on motor parameters, and it exhibits a poor transient response. A variant of SVMDTC with only one PI controller has been proposed in [11], which reduces tuning effort, but it still exhibits poor transient response. Another approach is to use duty ratio modulation-based DTC (Duty-DTC) [12], [13], [14], but with increased motor parameter dependency and computational burden. In [15], a simplified duty calculation method with improved steady-state performance is introduced but with deteriorated transient performance during load disturbances. There is also an increase in the average switching frequency of the inverter and the gains of the PI controller are evaluated using machine parameters. Another Duty-DTC approach is presented in [12], where the instantaneous slopes of the d and q -axis currents are used to determine the duty cycle, but it is computationally expensive. Yet another approach is to increase the number of levels in torque hysteresis controllers [16]. This approach is effective in reducing torque ripple when multilevel voltage source inverters

(MLVSI) are utilized. However, this increases the size of look-up table.

The open-end winding induction motor (OEWIM) powered from two separate two-level voltage source inverters (VSI) has voltage vectors similar to that of three-level VSI powered single-ended induction motor drive. The OEWIM drive can be powered using either single dc link or isolated dc links [17]. In the isolated configuration of the OEWIM system, the zero-sequence current is blocked [18]. The dual inverters with isolated dc links enhance system reliability [19]. In the presented OEWIM drive, two isolated dc sources with equal voltage levels are used. The DTC for OEWIM drives are given in [16], [20], [21], and [22]. In [16], a multilevel torque bands and dual table approach for OEWIM systems, is applied. The results show improvement in the drive performance. However, it has flux instability at low speed. In [22], a solution to address flux instability for OEWIM is proposed with the help of two switching tables. But this approach results in increased torque ripples at low speeds and two independent look-up tables these algorithms computationally expensive.

Ideally, selecting voltage vectors based on instantaneous torque and flux errors, rather than relying on hysteresis error bands, can reduce torque and flux ripples. Implementing this concept especially in MLVSI drives will significantly increase the complexity of the look-up table, making it impractical. To overcome this challenge, this article proposes an artificial neural network (ANN) to select the optimal voltage vector based on the instantaneous values of flux error and torque error, while retaining the core concept and simplicity of the DTC algorithm. This concept is applied to the OEWIM drive. Few attempts have been made to improve the performance of DTC with the help of ANN and fuzzy logic controllers [23], [24], [25], [26], [27], [28], [29], [30], [31], [32]. But in these methods, the hysteresis bands of DTC are not eliminated completely. In [23] and [26], the ANN-based DTC is presented for single-ended induction motor powered by conventional two level inverter. Here, the ANN is trained for a given flux error band and torque error band. As a result, it selects the available vectors within a sector without optimization. Thus, it fails to remove the effect of hysteresis bands. The proposed method is different from the fuzzy-based DTC algorithms [24], [25], [27], [28], [29], [30], [31], [32], as these use hysteresis controllers externally and use the output of the hysteresis controllers for voltage vector selection. Fuzzy-based DTC approaches often require extensive parameter tuning and have complex implementations due to their extensive rule bases [30]. The motivation behind using ANN is to capture and map the nonlinear relationship between torque error and flux error in order to determine the optimal output voltage vector, which can be challenging for traditional control methods to capture. In the case of OEWIM, their symmetric distribution of voltage vectors makes them suitable for ANN to learn and determine the optimal vectors required to drive the motor. The main contributions of this article are as follows.

- 1) The ANN serves as an efficient large data storage system, allowing the selection of optimal voltage vectors that minimize both flux and torque ripple for instantaneous flux position based on flux error and torque error. This

eliminates the variable flux and torque ripple problem associated with the table-based DTC algorithm.

- 2) The proposed method replaces the hysteresis controllers and look-up table by an ANN network with low memory requirement. A general framework for selecting optimum voltage vector based on its influence on torque and flux is presented. This results in significant improvement in the torque ripple, flux ripple, and steady-state performance without compromising the transient response.
- 3) The training data is generated offline and ANN is trained offline. The control system becomes more deterministic and easier to implement.
- 4) Stable flux operation is achieved even at low speeds, ensuring smooth and reliable drive operation.
- 5) Correction to torque reference to compensate the effect of speed and electromagnetic torque is implemented resulting in improved torque response.

The rest of this article is organized as follows. The brief review of CDTC is given in Section II. In Section III, the ANN-based DTC is developed. Simulation results, experimental validation, and performance analysis are given in Section IV. Finally, Section V concludes this article.

II. REVIEW OF DTC

The dynamics of the induction motor are given by [22]

$$v_s = r_s i_s + \frac{d\psi_s}{dt} \quad (1)$$

$$0 = r_r i_r + \frac{d\psi_r}{dt} - j\omega_r \psi_r \quad (2)$$

$$T_e - T_L = J \frac{d\omega}{dt} \quad (3)$$

The stator flux ψ_s and rotor flux ψ_r are given by

$$\psi_s = \int (v_s - r_s i_s) dt; \quad \psi_r = \frac{l_r}{l_m} (\psi_s - \sigma l_s i_s) \quad (4)$$

where v_s is the stator voltage and $r_s i_s$ is the voltage drop in stator. The electromagnetic torque (T_e) is given by

$$T_e = \frac{3P}{2} (\psi_{s\alpha} i_{\beta} - \psi_{s\beta} i_{\alpha}) = \frac{3P}{2} \frac{l_m}{\sigma l_s l_r} |\psi_s| |\psi_r| \sin \delta \quad (5)$$

The $\sigma = 1 - l_m^2 / l_s l_r$ is the leakage coefficient. Equation (5) indicates that the electromagnetic torque can be rapidly changed by changing the angle between ψ_s and ψ_r . In (1), the change in ψ_s in $(n+1)$ th sample can be expressed as follows:

$$\psi_s(n+1) = \psi_s(n) + (v_s(n) - r_s i_s(n))(T_s) \quad (6)$$

For small values of sampling interval (T_s), the effect of stator winding resistance drop can be neglected [16] and the ψ_s moves in the direction of the applied voltage vector. This is the basic concept of the DTC.

The dual inverter OEWIM configuration has 64 switching states generating 18 active voltage vectors and one zero voltage vector. Their representation in $\alpha - \beta$ plane is shown in Fig. 1. The $\alpha - \beta$ plane is divided into six sectors with each sector further divided into two subsectors a and b of 30° each. These

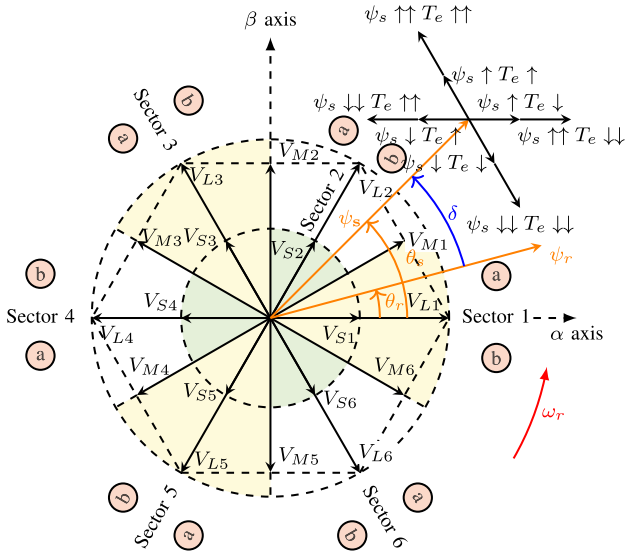


Fig. 1. Dual inverter voltage vectors and their effect on torque and flux.

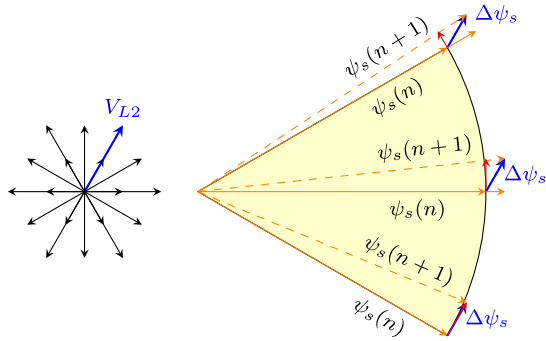


Fig. 2. Effect of voltage vectors in sector 1.

voltage vectors are composed of six small vectors (V_{Si} , where $i = 1$ to 6), six medium vectors (V_{Mi}), six large vectors (V_{Li}) vectors, and a zero vector (V_0). The effect of voltage vectors on torque and stator flux deviation (increase or decrease) depends on the position of the stator flux vector in the $\alpha - \beta$ plane. This is illustrated for flux position in sector 2b in Fig. 1. In CDTC, the large variations in flux ripple, torque ripple, and switching frequency are due to the hysteresis controller bands and sector by sector selection of voltage vectors [5], [33]. In CDTC, the same active voltage vector is selected based on flux hysteresis controller and torque hysteresis controller outputs for all the flux angular positions within a sector. But the active voltage vectors have different influence on torque and flux within the sector, which is not accounted in CDTC [5], [33]. This is illustrated for sector 1 in Fig. 2. The active voltage vector that produces large tangential component (tangential component has larger influence on torque) and small radial component (radial component has large influence on flux) for flux positions near the start of the sector. In CDTC, the same active voltage vector is selected throughout the sector but it has reduced tangential component and increased radial component as the flux position moves away. If proper voltage vectors based on its influence on torque and flux along with the torque hysteresis controller output and flux

hysteresis controller output for different flux positions within the sector are selected, the flux ripple and torque ripple can be reduced. Ideally, such selection for every instantaneous angular position of flux is desirable. In order to achieve this in CDTC, a large look-up table is required. But the same can be achieved in ANN with small memory requirement, where ANN memorizes the proper voltage vectors based on its tangential component and radial component for every flux position during training phase.

III. PROPOSED ANN-BASED CONTROLLER

The block diagram of the proposed ANN-based DTC is given in Fig. 3. ANN networks are well-known for their pattern recognition capabilities. In the case of DTC, the deterministic data comprising OEWM voltage vectors follows a specific pattern of response to different errors. The ANN-based approach leverages this pattern recognition capability to effectively respond to a wide range of torque error and flux error with fine resolution. The ANN network is made up of basic artificial neuron (nodes), which consists of dendrites that receive inputs and axons that generate outputs after processing these inputs [34]. The input nodes collectively form the input layer and represent the features for the ANN. The weight vectors (w) specify the contribution of input features for computing the next node. The existence of a bias term (b) in each node of an ANN enables the adjustment of output values, even in the absence of input information. The w and b are the primary design variables learned by the ANN. A three-node input layer ($x \in [\epsilon_T, \epsilon_\psi, \theta_s]$) is used to capture the instantaneous torque error (ϵ_T), instantaneous flux error (ϵ_ψ), and instantaneous angle of flux (θ_s). The output layer ($z_i \in (1$ to 6)) consists of six nodes, with the first three providing switching pulses for the first inverter (VSI-1), and the remaining three providing switching pulses for the second inverter (VSI-2). The complementary pulses for the bottom switches in each inverter is generated in the hardware from these switching pulses. After several iterative studies of training, a multilayer perceptron consisting of two-hidden layers (H1 and H2) with 50 nodes each as shown in Fig. 3 is finalized. This structure has a good balance between the training and the computational burden with satisfactory accuracy of mean square error (MSE) less than 10^{-3} . The output layer nodes (z) receive weighted inputs from the second-hidden layer and computes the switching pulses. Given that these switching pulses can only assume the binary values of logic “0” or logic “1”, the outputs are rounded off to generate the appropriate switching pulses.

A. Offline Data Creation

$|\psi_s|$ is set at rated value. To provide sufficient resolution for the ANN controller, flux vectors considered at every degree, resulting in 360 flux vector samples. The dual inverter OEWM configuration has 18 active and one zero voltage vectors, which are defined from the switching states of the VSI-1 ($S_{i(a_1, b_1, c_1)}$) and VSI-2 ($S_{i(a_2, b_2, c_2)}$) as

$$v_{i(a,b,c)} = (V_{dc1} S_{i(a_1, b_1, c_1)} - V_{dc2} S_{i(a_2, b_2, c_2)}) \quad (7)$$

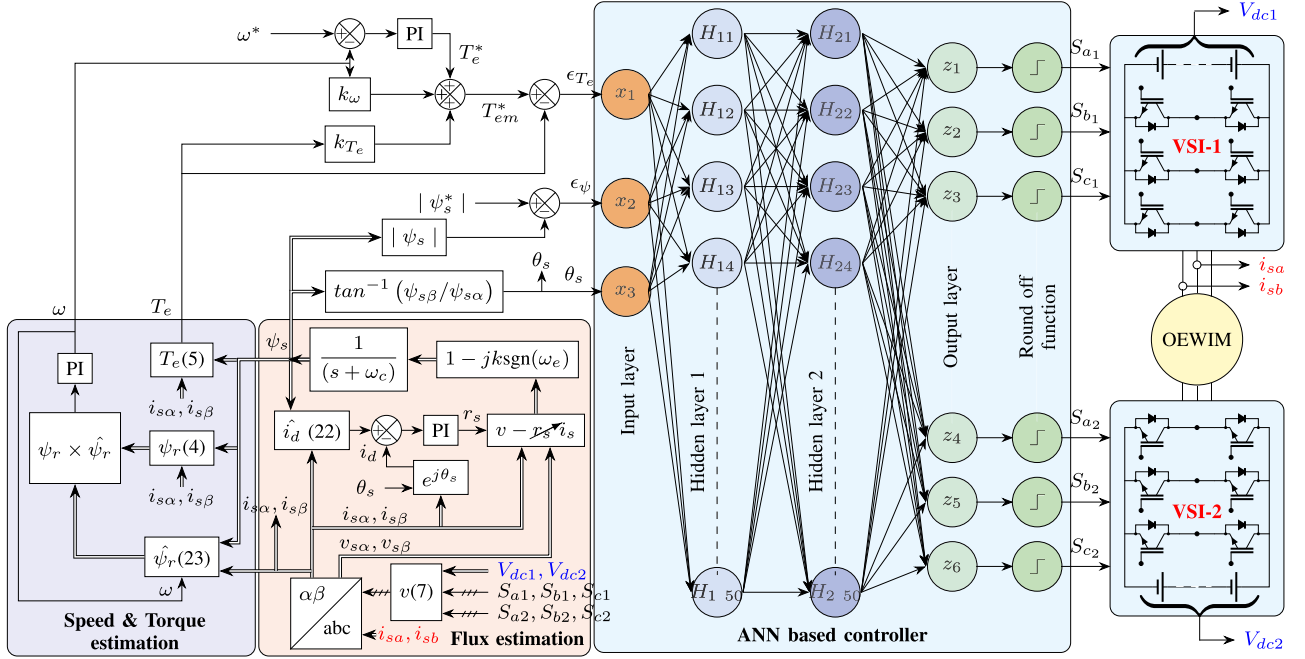


Fig. 3. Control structure of proposed ANN-based DTC.

where $i(\in 1$ to 19). The effect of the voltage vectors on ψ at n th sample is given by

$$\Delta\psi_n = v_i T_s \quad \forall i \in [1, 19]. \quad (8)$$

The effect of all the voltage vectors on the stator flux is analyzed as explained in Section II (see Fig. 1). The torque error (ϵ_T) and flux error (ϵ_ψ) at the n th sample due to all the 19 voltage vectors are obtained as

$$\begin{aligned} \epsilon_{T_{v_{i=1,2,\dots,19}}} &= K_\delta (\angle\psi_{n+1} - \angle\psi_n) \text{ and} \\ \epsilon_{\psi_{v_{i=1,2,\dots,19}}} &= |\psi_{n+1}| - |\psi_n| \end{aligned} \quad (9)$$

where $K_\delta = (3/2)(P/2)(l_m/\sigma l_s l_r)|\psi_s|\psi_r|$ [11]. The optimal voltage vector (v_n) for n th sample of the flux vector is selected as per the objective function (10), which minimizes the torque and flux feedback errors

$$\begin{aligned} v_n = \min((k) | \epsilon_T^* - \epsilon_{T_{v_{i=1,2,\dots,19}}} | \\ + (1-k) | \epsilon_\psi^* - \epsilon_{\psi_{v_{i=1,2,\dots,19}}} |) \end{aligned} \quad (10)$$

where k is the torque weighting factor. This factor is typically chosen to be greater than 0.5, to assign higher priority to torque control [35]. The switching pulses corresponding to the v_n are stored as optimal switching vector ($S(n)^*$). The process is repeated for the ψ_s vector for every degree. In the training data, there will be $n = 360$ flux samples. For every sample the range of ϵ_T is taken as $\pm 10\%$ with a step size of 0.5%. And the range of ϵ_ψ is taken as $\pm 5\%$ of rated flux with a step size of 0.5%. So the optimal switching states obtained from the objective function (10) consists of $360 \times 40 \times 20 = 288\,000$ optimal switching states. These will be stored in a matrix as ($S^* = (S_{a1}^*, S_{b1}^*, S_{c1}^*, S_{a2}^*, S_{b2}^*, S_{c2}^*)$) and used as training data

to train the ANN network. The flowchart for offline training data generation is shown in Fig. 4.

B. Offline Training of Neuron

Training the ANN involves adjusting the w and b of all the nodes of all the layers of the ANN to achieve a close match between the estimated output pulses and actual switching pulses in training data (S^*). The process is performed offline, where the ANN utilizes the Levenberg–Marquardt algorithm to fine-tune the weights and biases [36]. The abovementioned dataset is randomly divided into three groups, with 90% of the data used for training, 5% for validation and the remaining 5% for testing. Large training data is used as the system is deterministic, so overfit network would perform good enough [34]. The training algorithm is given as follows.

- 1) *Step-1*: The w and b are initialized with random initial values. The MSE is set to zero.
- 2) *Step-2*: The training dataset is shuffled to ensure random distribution. Initialize the sample number $n' = 1$. The input values ($\epsilon_T(n')$, $\epsilon_\psi(n')$ and $\theta_s(n')$) from shuffled dataset are selected.
- 3) *Step-3*: The outputs of the hidden layer 1 are computed using logistic regression function

$$H_{1m} = \frac{1}{1 + e^{-(w_{m1}\epsilon_\psi + w_{m2}\epsilon_T + w_{m3}\theta_s + b_m)}} \quad \forall m \in [1, 50]. \quad (11)$$

Using these, the outputs of hidden layer-2 are computed using logistic regression function

$$H_{2l} = \frac{1}{1 + e^{-(\sum_{m=1}^{50} w_{lm}H_{1m} + b_l)}} \quad \forall l \in [1, 50]. \quad (12)$$

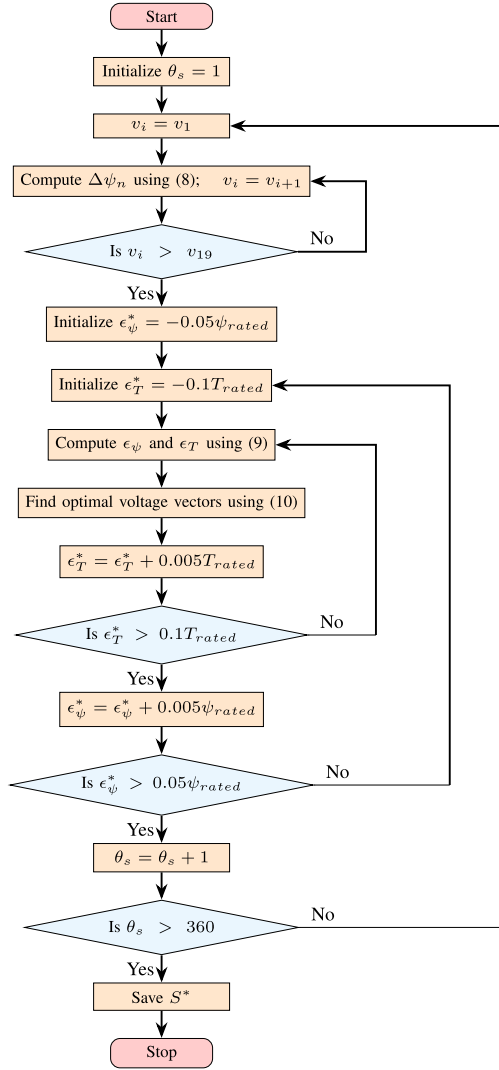


Fig. 4. Flowchart of generating training data.

Using these outputs, the outputs of output layer are computed using logistic regression function

$$z_i = \frac{1}{1 + e^{-(\sum_{l=1}^{50} w_{il} H_{2l} + b_i)}} \quad \forall i \in [1, 6]. \quad (13)$$

- 4) *Step-4*: The outputs are passed through the round off function to determine the switching states $S(n') = (S_{a1}, S_{b1}, S_{c1}, S_{a2}, S_{b2}, S_{c2})$.
- 5) *Step-5*: Sample number $n' = n' + 1$; The steps 2 to 5 are repeated till the first 90% of the data samples in the shuffled dataset (i.e., 259 200 samples) are completed.
- 6) *Step-6*: The MSE for every output nodes $z_i \forall i \in [1 - 6]$ is computed using (14)

$$\text{MSE}(z_i) = \frac{1}{2 \times 259200} \sum_{n=1}^{259200} (S_i(n) - S_i^*(n))^2 \quad \forall i \in [1-6] \quad (14)$$

where $S_1 = S_{a1}, S_2 = S_{b1}, S_3 = S_{c1}, S_4 = S_{a2}, S_5 = S_{b2}$ and $S_6 = S_{c2}$.

- 7) *Step-7*: The average value of this MSE for each of the output nodes is the loss function. This loss function indicates the amount of tuning required for the w and b of the neural network. This loss function is used in back propagation technique with steepest descent algorithm as shown in (15) to tune the w and b

$$\begin{aligned} w_i^h(K+1) &= w_i^h(K) - l_A B_p^{-1} \frac{\partial F}{\partial w_i^h} \\ b_i^h(K+1) &= b_i^h(K) - l_A B_p^{-1} \frac{\partial F}{\partial b_i^h} \end{aligned} \quad (15)$$

where $H = 1$ for hidden layer 1, $H = 2$ for hidden layer 2 and $H = 3$ for output layer, K is the iteration number, F is loss function gradient (local*upstream gradient) for each neuron, l_A is the learning rate, learning rate is introduced for avoiding oscillations in gradient descent method around the output. Learning rate is a hyperparameter, in this work, l_A is chosen to be 0.1. B_p^{-1} is a back propagation parameter. The MSE for the validation dataset (next 5% from the shuffled dataset) is computed similar to the abovementioned procedure. The value of MSE from validation data is used to fine tune the B_p^{-1} in (16) and μ in (19)

$$B_p^{-1} = \Delta^2 \text{MSE}(w, b)|_{(w, b = w(K), b(K))}. \quad (16)$$

The recurrence gradient of nodes of the whole network is represented in Jacobian matrix as

$$\Delta \text{MSE}(w, b) = J(w, b) = \begin{bmatrix} \frac{\partial E_1}{\partial w_{m1}} & \frac{\partial E_1}{\partial w_{m2}} & \cdots & \frac{\partial E_1}{\partial b_{z6}} \\ \frac{\partial E_2}{\partial w_{m1}} & \frac{\partial E_2}{\partial w_{m2}} & \cdots & \frac{\partial E_2}{\partial b_{z6}} \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \frac{\partial E_6}{\partial w_{m1}} & \frac{\partial E_6}{\partial w_{m2}} & \cdots & \frac{\partial E_6}{\partial b_{z6}} \end{bmatrix} \quad (17)$$

where $E_i = \frac{1}{2 \times 259200} \sum_{n=1}^{259200} (S_i(n) - S_i^*(n))^2 \forall i \in [1 - 6]$. The hessian matrix ($\Delta^2 \text{MSE}(w, b)$) is defined as

$$[\Delta^2 \text{MSE}(w, b)](h) = \frac{\partial \text{MSE}(w, b)}{\partial w \partial b}. \quad (18)$$

Substituting (16) to (18) in (15), (15) will be

$$\begin{aligned} w_i^h(K+1) &= w_i^h(K) - [J(w, b)(K)^T J(w, b)(K) + \mu I]^{-1} \\ &\quad J(w, b)(K)^T E_i \\ b_i^h(K+1) &= b_i^h(K) - [J(w, b)(K)^T J(w, b)(K) + \mu I]^{-1} \\ &\quad J(w, b)(K)^T E_i \end{aligned} \quad (19)$$

where I is identity matrix and the variable $\mu(K+1) = \mu(K) \cdot \kappa$ if $\text{MSE}(w, b)_{K+1} \leq \text{MSE}(w, b)_{K+1}$, else $\mu(K+1) = \mu(K)/\kappa$.

- 8) *Step-8*: Testing of the network: Last 5% of the shuffled dataset is used to test the network. The average value of MSE is computed for this dataset as explained above. In this case, the updated value of w and b obtained in step-7

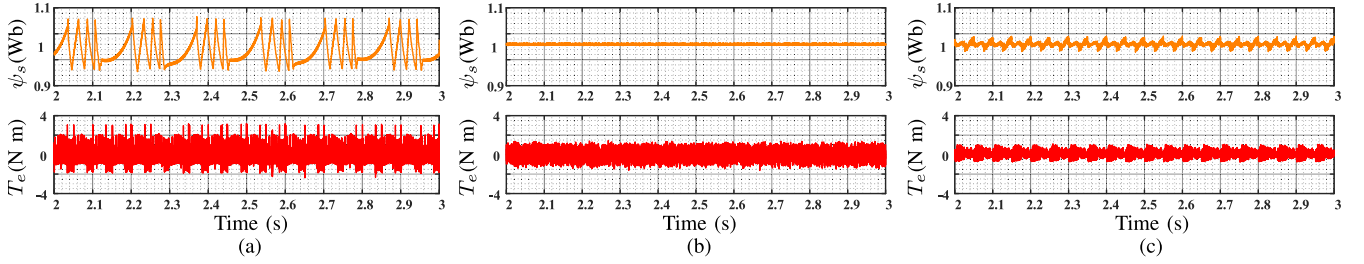


Fig. 5. Comparative results: Steady-state flux and torque at low speed (30 RPM) and no load condition for (a) CDTC [16], (b) SVM DTC [8], (c) proposed ANN-based DTC.

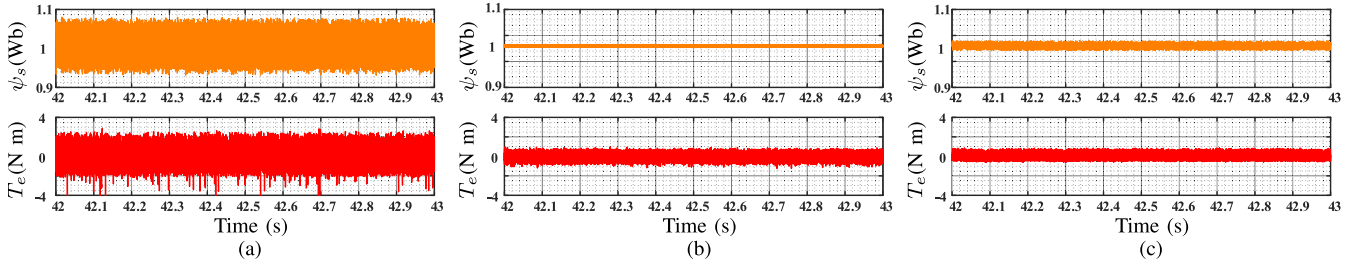


Fig. 6. Comparative results: Steady-state flux and torque at rated speed (1440 RPM) and no load condition for (a) CDTC [16], (b) SVM DTC [8], (c) Proposed ANN-based DTC.

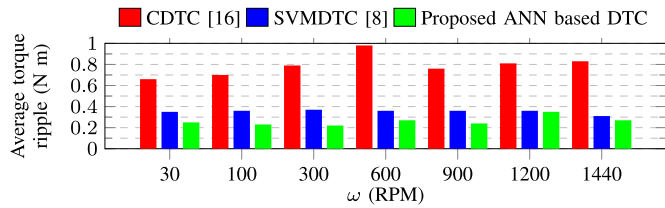


Fig. 7. Comparative results of average torque ripples at different speeds under no load for CDTC [16], SVM DTC [8], and Proposed ANN-based DTC.

above are used. If this $MSE \geq 1e^{-3}$, the above step-2 to step-6 are repeated till $MSE < 1e^{-3}$. If this $MSE < 1e^{-3}$, the training, validation, and testing are stopped and the ANN is ready for use.

C. Switching Pulses Generation Using ANN

At the end of offline training, validation, and testing, the w and b values that guarantee optimal switching pulses for any instantaneous value of ϵ_{ψ} , ϵ_T , and θ_s are stored in the memory. So only $((3 \times 50) + (50 \times 50) + (50 \times 6))$ values of w and $(50 + 50 + 6)$ values of b need to be stored in memory. Using this, for instantaneous value of ϵ_{ψ} , ϵ_T , and θ_s , the ANN computes output of first layer using (11), then output of second hidden layer using (12), and outputs of output layer using (13). The switching pulses are generated by rounding off the outputs of the nodes of the output layer. In OEWIM configuration, ten zero switching states combinations are available. To reduce the switching frequency, if n th sample uses the active switching state ($S(n)$) as per the procedure given above and in the next

sample ($(n + 1)$ th), the controller selects zero vector (S_0) that has minimum Euclidean distance from the previously selected active voltage vector as

$$S(n + 1) = \min(|S(n) - S_0|). \quad (20)$$

D. Stator Flux Estimation

In the given equation, ϵ_{ψ} is computed using $\epsilon_{\psi} = \psi_s^* - \psi_s$ and the $\theta_s = \tan^{-1} \frac{\psi_{s\beta}}{\psi_{s\alpha}}$. The simple model given in (4) for the estimation of ψ_s is sensitive for dc bias of the input signal and r_s [37]. The dc-drift problem of integrator is mitigated using the modified integrator shown in Fig. 3 [35], [37], [38]. In this case, the digital integrator is replaced with a low-pass filter characterized by a variable cutoff frequency, ω_c . This frequency is determined as a function of the synchronous frequency (ω_s), enhancing the filtering processes effectiveness across a wide range of stator frequencies, expressed as $\omega_c = k \cdot \omega_s$. The value of k is typically chosen in the range of (0.1–0.5) [37], [38]. The stator frequency can be computed from the stator flux vector as [39], [40]

$$\omega_s = \frac{1}{\psi_s^2} \left[\left(\psi_{s\alpha} \frac{d\psi_{s\beta}}{dt} - \psi_{s\beta} \frac{d\psi_{s\alpha}}{dt} \right) \right]. \quad (21)$$

To eliminate the errors introduced by the low-pass filter, the input signal is augmented with a compensating term, $(1 - jk\text{sgn}(\omega_e))$ [37]. To compensate the effect due to variations in r_s , a model reference adaptive system (MRAS)-based resistance estimator is used [41]. The functional block diagram of resistor estimator is given in Fig. 3. The d -axis component of the motor current is

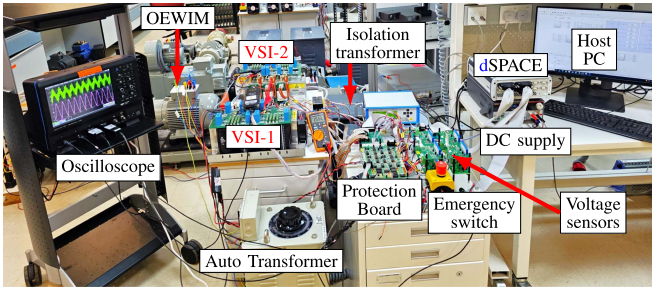


Fig. 8. Experimental prototype of OEIWM drive.

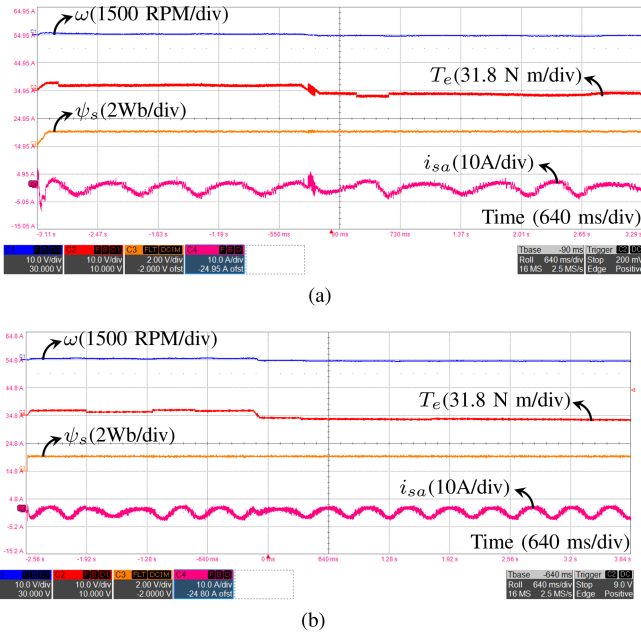


Fig. 9. Experimental results: Dynamic response with 30 RPM and 10% of rated load (a) CDTC [16]. (b) Proposed ANN-based DTC.

represented in the following equation [40], [41]:

$$i_d^* = \frac{\psi_s}{l_s + \sigma l_s} + \frac{l_s \sigma l_s}{(l_s + \sigma l_s) \psi_s} i_s^2. \quad (22)$$

In (22), i_d^* is independent of r_s . Equation (22) is used as an adaptive model within the MRAS framework. The d -axis component of the measured stator current vector is then assessed and compared with i_d^* (22). Deviations between the two current components are indicative of discrepancies in stator resistance (r_s). Notably, the dependency of (22) on the parameter ψ_s enables the tuning of r_s [41]. An adaptation algorithm with PI control, is used to adjust the r_s until the $i_d^* = i_d$ [42].

E. Speed Controller

For sensorless speed control, the motor speed (ω) is estimated using the MRAS technique in which the flux equations of the current model, defined as an adaptive model in the following equation:

$$\hat{\psi}_r = \frac{l_m}{\tau_r} i_s \left(1 - \frac{1}{\tau_r} - j\omega \frac{P}{2} \right) \psi_s \quad (23)$$

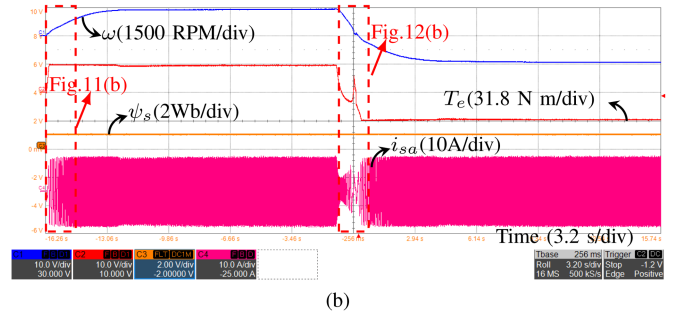
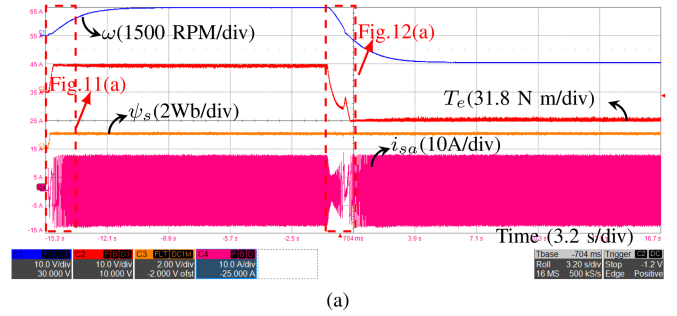


Fig. 10. Experimental results: Dynamic response with rated speed and rated load. (a) CDTC [16]. (b) Proposed ANN-based DTC.

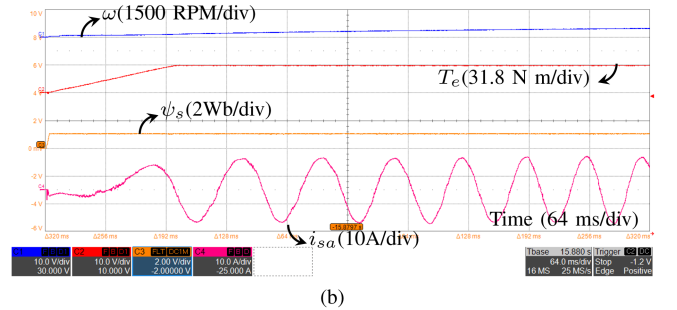
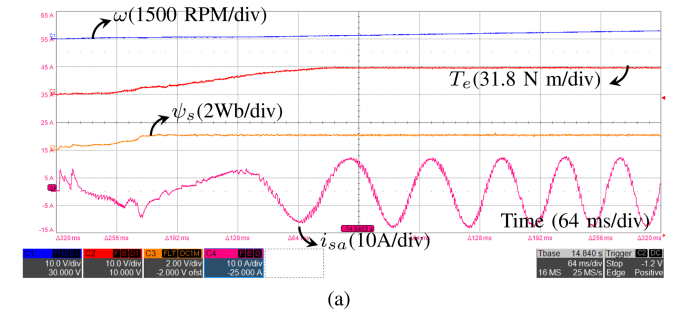
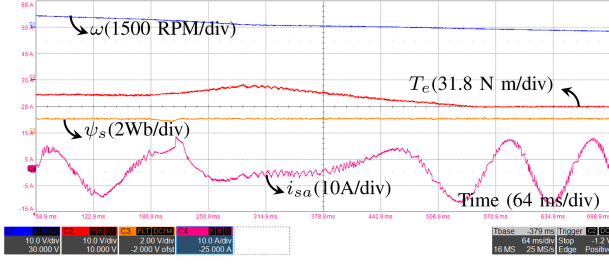
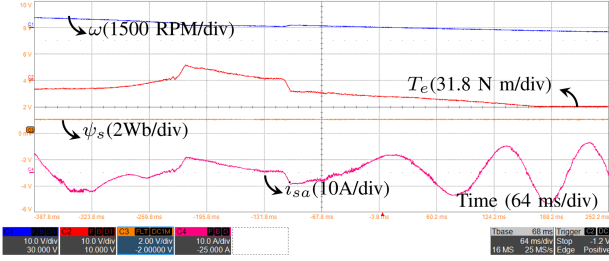


Fig. 11. Experimental results: Start up response with rated speed and rated load. (a) CDTC [16]. (b) Proposed ANN-based DTC.

where $\tau_r = l_r/r_r$ and flux estimated from voltage model is used as a reference system (4). Ideally, with the correct speed signal, the fluxes calculated from (4) and those calculated from the adaptive model should match. An adaptation algorithm with PI control, is used to adjust the speed ω until the error $\xi = \hat{\psi}_r \times \psi_r = 0$ [43]. The estimated speed is used as feedback signal to the speed PI controller. The speed controller is shown in Fig. 3. The output of the speed controller is the reference torque (T_e^*). In order to take into account the negative effects of speed (T_e)

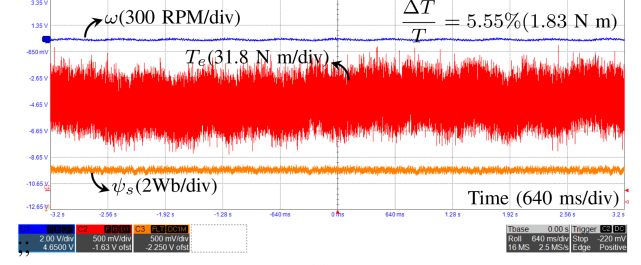


(a)

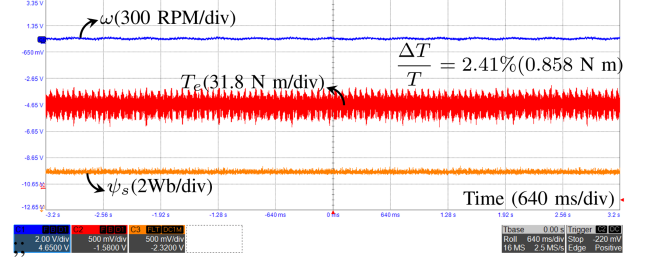


(b)

Fig. 12. Experimental results: Speed reversal from 1440 RPM to -1440 RPM with rated load. (a) CDTC [16]. (b) Proposed ANN-based DTC.



(a)



(b)

Fig. 14. Experimental results: Steady-state ripple at 30 RPM and no load. (a) CDTC [16]. (b) Proposed ANN-based DTC.

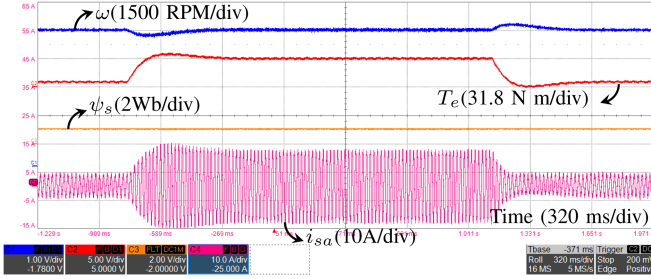


Fig. 13. Experimental results: Response to external load disturbance—Step change in load from 10% to 100% and 100% to 10% at rated speed.

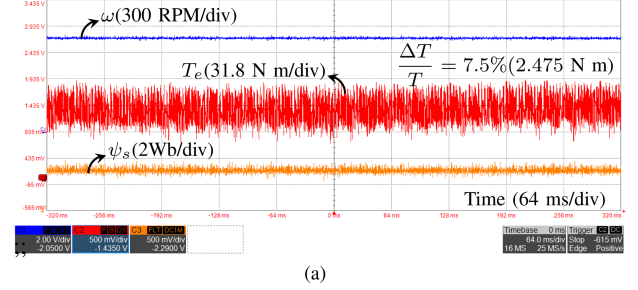
and developed torque (T_e), the corrected torque reference is T_{em}^* is given by [4]

$$T_{em}^* = T_e + \left[\frac{3P}{2} \left(\frac{l_m}{\sigma l_s l_r} j\omega_r (\psi_s \cdot \bar{\psi}_r) \right) + T_e \left(\frac{r_s}{\sigma l_s} + \frac{r_r}{\sigma l_r} \right) \right] T_s. \quad (24)$$

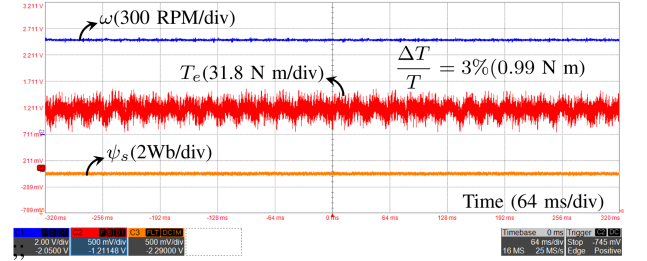
Taking, $|\psi_s| \approx |\psi_r|$. Equation (24) is simplified as

$$T_{em}^* = T_e + K_\omega \omega_r (|\psi_s|^2) + K_{T_e} T_e \quad (25)$$

where $K_\omega = \left(\frac{3P}{2} \frac{l_m}{\sigma l_s l_r} \right) T_s$ and $K_{T_e} = \left(\frac{r_s}{\sigma l_s} + \frac{r_r}{\sigma l_r} \right) T_s$. The constants K_ω and K_{T_e} are computed through simulation as follows. The motor is run for typical value of ω at no load. At steady state due to no load, the $T_e = 0$, so the dynamic torque error is primarily affected by the speed of the motor and K_ω is calculated from the motor torque. Next, the motor is run at rated speed and rated load. At steady state, the value of K_{T_e}



(a)



(b)

Fig. 15. Experimental results: Steady-state ripple at 1440 RPM and 10% of rated load. (a) CDTC [16]. (b) Proposed ANN-based DTC.

is computed from the motor torque using above K_ω . The ϵ_T is computed as $\epsilon_T = T_{em}^* - T_e$, where the T_e is estimated using (5).

IV. RESULTS AND DISCUSSIONS

The proposed ANN-based OEWM drive is verified through simulation in MATLAB/Simulink environment as well as through experiment on a laboratory prototype. The parameters used in simulation and experiment are listed in Table I.

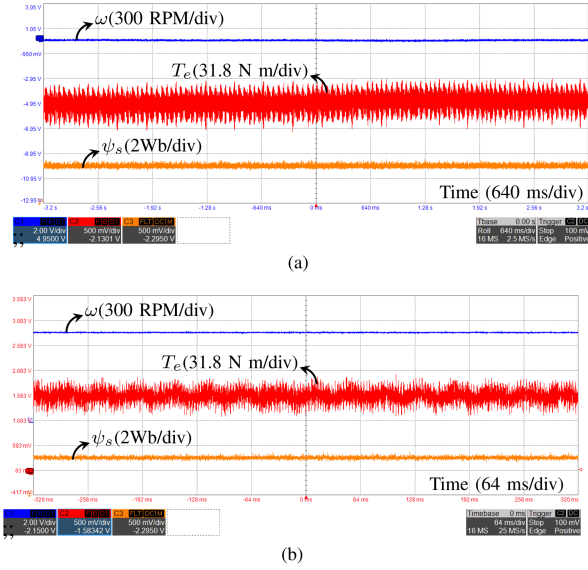


Fig. 16. Experimental results: Steady-state ripple when motor is hot at 10% of rated load for Proposed ANN-based DTC. (a) 30 RPM. (b) 1440 RPM.

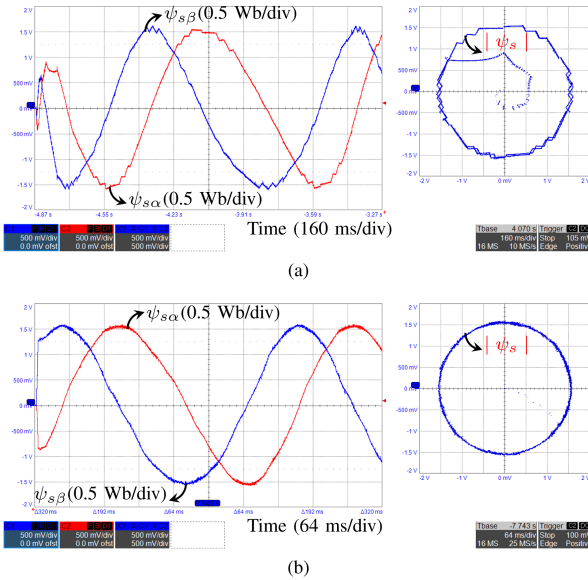


Fig. 17. Experimental results: Start up ψ_s trajectory at 30 RPM and 10% of rated load. (a) CDTC [16]. (b) Proposed ANN-based DTC.

TABLE I
MOTOR RATING AND PARAMETERS

5kW, 400V, 1440RPM, 50Hz, 11.2A induction motor coupled to 5kW, 1500RPM, 300V separately excited dc Generator.		
$r_s = 1.12 \Omega$	$r_r = 1.033 \Omega$	$l_s = 0.177 \text{ H}$
$l_r = 0.177 \text{ H}$	$l_m = 0.1702 \text{ H}$	$J = 0.38 \text{ kg/m}^2$
$P = 4$	$\psi_{\text{rated}} = 1.04 \text{ Wb}$	$T_{\text{rated}} = 31.8 \text{ N-m}$

A performance comparison is made among three DTC schemes namely, the following:

- 1) CDTC with a seven-level torque hysteresis band [16];
- 2) SVM DTC [8];
- 3) proposed ANN-based DTC through simulation studies.

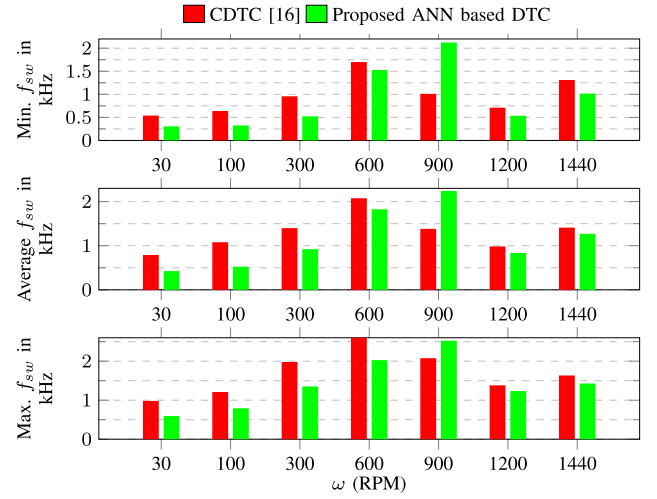


Fig. 18. Minimum, average and maximum value of f_{sw} , for CDTC [16] and proposed ANN-based DTC.

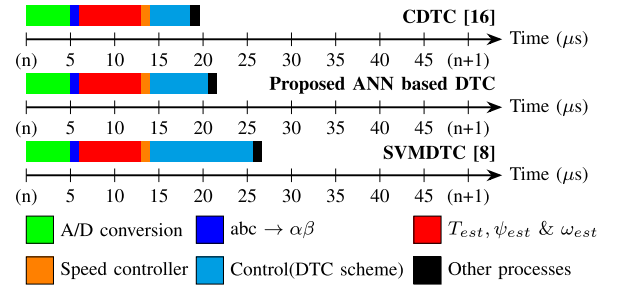


Fig. 19. Computational timing diagram of real-time implementation for CDTC [16], SVM DTC [8], and proposed ANN-based DTC.

Among the table-based CDTC schemes, the CDTC with a seven-level torque hysteresis band is chosen as it has the best performance in terms of torque and flux ripple mitigation in OEWM drives [16]. To ensure a fair comparison, all the above-mentioned three DTC algorithms were executed with a sampling time (T_s) of $50 \mu\text{s}$. For CDTC, the hysteresis bands for torque and flux were set to 10% of the rated torque and 5% of the rated flux, respectively. In the SVM DTC scheme, switching frequency of 1.5 kHz corresponding to the mean switching frequency observed in the proposed ANN-based DTC is used. The steady-state performance of these schemes is evaluated by analyzing torque ripples at low speed (30 RPM) as well as rated speed (1440 RPM) under no load conditions. These results are given in Figs. 5 and 6. The proposed ANN-based controller achieves a torque ripple reduction of 66.6% at low speed (30 RPM) and 67.5% at rated speed (1440 RPM) compared to CDTC, and 28.5% at low speed (30 RPM) and 12.9% at rated speed (1440 RPM) compared to SVM DTC. The comparison of average torque ripple for different speeds at no load is summarized in Fig. 7. These results demonstrate that the proposed ANN-based DTC scheme has low torque ripple compared to that of CDTC and SVM DTC for all operating conditions.

The proposed ANN-based DTC is experimentally verified. The laboratory prototype setup is shown in Fig. 8. The voltage source inverters (VSI-1 and VSI-2) receive dc inputs from

two separate isolated three-phase uncontrolled bridge rectifiers. The control algorithms are implemented in dSPACE RTI 1204 digital controller. Two phase currents (i_{sa} and i_{sb}) of induction motor are sensed using hall sensors (TELCON HTP50). DC-bus voltages (V_{dc1} and V_{dc2}) are sensed using isolation amplifiers (AD215BY). The performance of the proposed DTC is compared experimentally with that of the CDTC with seven levels in torque hysteresis band [16].

The plots of ω , T_e , ψ_s and phase current (i_{sa}) at low speed (30 RPM) and rated speed (1440 RPM) with full load are given in Figs. 9 and 10, respectively. The motor is started at $t = 0$, and speed is reversed when motor is in steady state. The steady-state torque is reached in 180 ms for proposed ANN-based DTC, it takes 390 ms in CDTC. This improvement can be attributed to the utilization of a corrected torque reference (25). The start up transient responses under full load and transient response during speed reversal at rated speed and full load are given in Figs. 11 and 12, respectively.

Fig. 13 shows the response for the step change in load. In Fig. 13, in steady state when the motor is running a rated speed, a step change of load from 10% to 100% of the rated load and a step change of load from 100% to 10% of rated load is applied at steady state. A $\pm 1.7\%$ change in speed is observed but the speed controller is able to correct this in 0.5 s.

The plot of torque ripples for speed of 30 RPM and 1440 RPM is shown in Figs. 14 and 15, respectively. From these results, it can be observed that the proposed ANN-based DTC has reduced torque ripple compared to CDTC. A 45% reduction in torque ripple is achieved. The drive is run for six hours at rated load and the case temperature of motor increased from 22 °C to 39.2 °C. The stator resistance at this temperature is found to be 1.2 Ω that is an increase by 10%. The plot of torque ripples for low speed (30 RPM) and rated speed (1440 RPM) at high temperature is shown in Fig. 16. There is no change in ripples and response between high temperature and normal temperature conditions. From these results, it can be observed that the flux estimator and speed estimator work satisfactorily under motor parameter variation.

The maximum, minimum, and average value of switching frequency (f_{sw}) at different speeds at light load conditions are given in Fig. 18. The proposed ANN-based DTC has lower average f_{sw} and narrowed the variation f_{sw} . The plot of $\alpha - \beta$ component of experimentally observed stator flux are given in Fig. 17 for CDTC and proposed ANN-based DTC. The result shows that the proposed DTC has reduced flux ripple. The computational burden of algorithm was measured by running it on dSPACE MicroLabBox DS1204. A sampling frequency of 20 kHz is used. Fig. 19 shows the computational time required for real-time implementation for the proposed ANN-based DTC, CDTC [16], and SVM DTC [8]. Even though proposed ANN-based controller has much higher resolution compared to that of CDTC, the computational time is comparable with that of CDTC.

V. CONCLUSION

ANN-based DTC for OEWM drives is presented. The look-up table and hysteresis controllers of CDTC are replaced by an

efficient ANN. The optimal voltage vectors are selected directly by tracking flux and torque errors, resulting in reduced torque and flux ripples and good transient response. The proposed ANN-based DTC is verified through simulation and laboratory experimental tests. The performance is compared with that of CDTC with seven-level torque hysteresis bands and SVM DTC. The comparative results demonstrate the improved performance of the proposed ANN-based DTC in terms of torque ripple reduction, flux ripple reduction, flux stability, reduced switching frequency variation, and transient response. The proposed ANN-based DTC is trained using large dataset, thus, has higher accuracy with low-memory requirement and comparable computation time. The improved performance of the drive will help several applications especially the electric vehicles. Developing an online ANN, and developing a proper deterministic method to determine the optimal size and hyperparameters for an ANN architecture will be challenging works for the future.

REFERENCES

- [1] G. Buja and M. Kazmierkowski, "Direct torque control of PWM inverter-fed AC motors—A survey," *IEEE Trans. Ind. Electron.*, vol. 51, no. 4, pp. 744–757, Aug. 2004.
- [2] S. G. Petkar and V. K. Thippiripati, "A novel duty-controlled DTC of a surface PMSM drive with reduced torque and flux ripples," *IEEE Trans. Ind. Electron.*, vol. 70, no. 4, pp. 3373–3383, Apr. 2023.
- [3] P. Naganathan and S. Srinivas, "Direct torque control techniques of three-level h-bridge inverter fed induction motor for torque ripple reduction at low speed operations," *IEEE Trans. Ind. Electron.*, vol. 67, no. 10, pp. 8262–8270, Oct. 2020.
- [4] J.-K. Kang and S.-K. Sul, "New direct torque control of induction motor for minimum torque ripple and constant switching frequency," *IEEE Trans. Ind. Appl.*, vol. 35, no. 5, pp. 1076–1082, Sep./Oct. 1999.
- [5] A. Nasr, C. Gu, X. Wang, G. Buticchi, S. Bozhko, and C. Gerada, "Torque-performance improvement for direct torque-controlled PMSM drives based on duty-ratio regulation," *IEEE Trans. Power Electron.*, vol. 37, no. 1, pp. 749–760, Jan. 2022.
- [6] I. M. Alsofyani, K. Y. Kim, S. S. Lee, and K.-B. Lee, "A modified flux regulation method to minimize switching frequency and improve DTC-hysteresis-based induction machines in low-speed regions," *IEEE Trans. Emerg. Sel. Topics Power Electron.*, vol. 7, no. 4, pp. 2346–2355, Dec. 2019.
- [7] Y.-S. Lai and J.-H. Chen, "A new approach to direct torque control of induction motor drives for constant inverter switching frequency and torque ripple reduction," *IEEE Trans. Energy Convers.*, vol. 16, no. 3, pp. 220–227, Sep. 2001.
- [8] B. R. Vinod and M. R. Baiju, "Direct torque control implemented on a three-level open-end winding induction motor drive," in *Proc. IEEE Int. Conf. Power Electron., Drives Energy Syst.*, 2016, pp. 1–6.
- [9] W. Wang, C. Liu, H. Zhao, and Z. Song, "Improved deadbeat-direct torque and flux control for PMSM with less computation and enhanced robustness," *IEEE Trans. Ind. Electron.*, vol. 70, no. 3, pp. 2254–2263, Mar. 2023.
- [10] C. Lascu, A. Argeșeanu, and F. Blaabjerg, "Supertwisting sliding-mode direct torque and flux control of induction machine drives," *IEEE Trans. Power Electron.*, vol. 35, no. 5, pp. 5057–5065, May 2020.
- [11] A. K. Peter, J. Mathew, and K. Gopakumar, "A simplified DTC-SVPWM scheme for induction motor drives using a single PI controller," *IEEE Trans. Power Electron.*, vol. 38, no. 1, pp. 750–761, Jan. 2023.
- [12] Z. Zhang and X. Liu, "A duty ratio control strategy to reduce both torque and flux ripples of DTC for permanent magnet synchronous machines," *IEEE Access*, vol. 7, pp. 11820–11828, 2019.
- [13] X. Lin, W. Huang, W. Jiang, Y. Zhao, D. Dong, and X. Wu, "Direct torque control for three-phase open-end winding PMSM with common DC bus based on duty ratio modulation," *IEEE Trans. Power Electron.*, vol. 35, no. 4, pp. 4216–4232, Apr. 2020.
- [14] X. Wu, W. Huang, X. Lin, W. Jiang, Y. Zhao, and S. Zhu, "Direct torque control for induction motors based on minimum voltage vector error," *IEEE Trans. Ind. Electron.*, vol. 68, no. 5, pp. 3794–3804, May 2021.

- [15] F. Niu et al., "A simple and practical duty cycle modulated direct torque control for permanent magnet synchronous motors," *IEEE Trans. Power Electron.*, vol. 34, no. 2, pp. 1572–1579, Feb. 2019.
- [16] S. Lakhimsetty, V. S. P. Satelli, R. S. Rathore, and V. T. Somasekhar, "Multilevel torque hysteresis-band based direct-torque control strategy for a three-level open-end winding induction motor drive for electric vehicle applications," *IEEE Trans. Emerg. Sel. Topics Power Electron.*, vol. 7, no. 3, pp. 1969–1981, Sep. 2019.
- [17] S. Chowdhury, P. W. Wheeler, C. Patel, and C. Gerada, "A multilevel converter with a floating bridge for open-end winding motor drive applications," *IEEE Trans. Ind. Electron.*, vol. 63, no. 9, pp. 5366–5375, Sep. 2016.
- [18] H. Stemmeler and P. Guggenbach, "Configurations of high-power voltage source inverter drives," in *Proc. 5th Eur. Conf. Power Electron. Appl.*, 1993, pp. 7–14.
- [19] U. R. Muduli, A. R. Beig, K. A. Jaafari, J. Y. Alsawalhi, and R. K. Behera, "Interrupt-free operation of dual-motor four-wheel drive electric vehicle under inverter failure," *IEEE Trans. Transport. Electrification*, vol. 7, no. 1, pp. 329–338, Mar. 2021.
- [20] S. Lakhimsetty and V. T. Somasekhar, "An efficient predictive current control strategy for a four-level open-end winding induction motor drive," *IEEE Trans. Power Electron.*, vol. 35, no. 6, pp. 6198–6207, Jun. 2020.
- [21] K. V. P. Kumar and T. V. Kumar, "Predictive torque control strategy of an open-end winding induction motor drive with less common-mode voltage," in *Proc. IEEE Int. Conf. Ind. Technol.*, 2018, pp. 498–503.
- [22] R. E. Kodumur Meesala and V. K. Thippiripati, "An improved direct torque control of three-level dual inverter fed open-ended winding induction motor drive based on modified look-up table," *IEEE Trans. Power Electron.*, vol. 35, no. 4, pp. 3906–3917, Apr. 2020.
- [23] "Artificial neural network-based DTC of an induction machine with experimental implementation on FPGA," *Eng. Appl. Artif. Intell.*, vol. 121, 2023, Art. no. 105972.
- [24] Y.-S. Lai and J.-C. Lin, "New hybrid fuzzy controller for direct torque control induction motor drives," *IEEE Trans. Power Electron.*, vol. 18, no. 5, pp. 1211–1219, Sep. 2003.
- [25] S. Gdaim, A. Mtibaa, and M. F. Mimouni, "Design and experimental implementation of DTC of an induction machine based on fuzzy logic control on FPGA," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 3, pp. 644–655, Jun. 2015.
- [26] P. Grabowski, M. Kazmierkowski, B. Bose, and F. Blaabjerg, "A simple direct-torque neuro-fuzzy control of PWM-inverter-fed induction motor drive," *IEEE Trans. Ind. Electron.*, vol. 47, no. 4, pp. 863–870, Aug. 2000.
- [27] V. N. N., A. Panda, and S. P. Singh, "A three-level fuzzy-2 DTC of induction motor drive using SVPWM," *IEEE Trans. Ind. Electron.*, vol. 63, no. 3, pp. 1467–1479, Mar. 2016.
- [28] V. Naik N., S. P. Singh, and A. K. Panda, "An interval type-2 fuzzy-based DTC of IMD using hybrid duty ratio control," *IEEE Trans. Power Electron.*, vol. 35, no. 8, pp. 8443–8451, Aug. 2020.
- [29] L. Romeral, A. Arias, E. Aldabas, and M. G. Jayne, "Novel direct torque control (DTC) scheme with fuzzy adaptive torque-ripple reduction," *IEEE Trans. Ind. Electron.*, vol. 50, no. 3, pp. 487–492, Jun. 2003.
- [30] A. Berzoy, J. Rengifo, and O. Mohammed, "Fuzzy predictive DTC of induction machines with reduced torque ripple and high-performance operation," *IEEE Trans. Power Electron.*, vol. 33, no. 3, pp. 2580–2587, Mar. 2018.
- [31] M. Uddin and M. Hafeez, "FLC-based DTC scheme to improve the dynamic performance of an IM drive," *IEEE Trans. Ind. Appl.*, vol. 48, no. 2, pp. 823–831, Mar./Apr. 2012.
- [32] M. Hafeez, M. N. Uddin, N. A. Rahim, and H. W. Ping, "Self-tuned NFC and adaptive torque hysteresis-based DTC scheme for IM drive," *IEEE Trans. Ind. Appl.*, vol. 50, no. 2, pp. 1410–1420, Mar./Apr. 2014.
- [33] I. M. Alsofyani and K.-B. Lee, "Improved transient-based overmodulation method for increased torque capability of direct torque control with constant torque-switching regulator of induction machines," *IEEE Trans. Power Electron.*, vol. 35, no. 4, pp. 3928–3938, Apr. 2020.
- [34] D. Bashir, G. D. Montañez, S. Sehra, P. S. Segura, and J. Lauw, "An information-theoretic perspective on overfitting and underfitting," in *Proc. Adv. Artif. Intelligence: 33rd Australas. Joint Conf.*, Canberra, ACT, Australia, 2020, pp. 347–358.
- [35] Y. Zhang and J. Zhu, "Direct torque control of permanent magnet synchronous motor with reduced torque ripple and commutation frequency," *IEEE Trans. Power Electron.*, vol. 26, no. 1, pp. 235–248, Jan. 2011.
- [36] J. J. Moré, "The Levenberg-Marquardt algorithm: Implementation and theory," in *Proc. Numer. Anal.: Proc. Biennial Conf. Held Dundee*, 2006, pp. 105–116.
- [37] Y. Zhang, J. Zhu, W. Xu, and Y. Guo, "A simple method to reduce torque ripple in direct torque-controlled permanent-magnet synchronous motor by using vectors with variable amplitude and angle," *IEEE Trans. Ind. Electron.*, vol. 58, no. 7, pp. 2848–2859, Jul. 2010.
- [38] M. Hinkkanen and J. Luomi, "Modified integrator for voltage model flux estimation of induction motors," in *Proc. IEEE 27th Annu. Conf. Ind. Electron. Soc.*, 2001, pp. 1339–1343.
- [39] J. Holtz, "Sensorless control of induction machines with or without signal injection?," *IEEE Trans. Ind. Electron.*, vol. 53, no. 1, pp. 7–30, Feb. 2006.
- [40] H. Joachim, "Sensorless control of induction motor drives," *Proc. IEEE*, vol. 90, no. 8, pp. 1359–1394, Aug. 2002.
- [41] J. Holtz and J. Quan, "Drift-and parameter-compensated flux estimator for persistent zero-stator-frequency operation of sensorless-controlled induction motors," *IEEE Trans. Ind. Appl.*, vol. 39, no. 4, pp. 1052–1060, Jul./Aug. 2003.
- [42] M. Cirrincione, M. Pucci, G. Cirrincione, and G.-A. Capolino, "A new TLS-based MRAS speed estimation with adaptive integration for high-performance induction machine drives," *IEEE Trans. Ind. Appl.*, vol. 40, no. 4, pp. 1116–1137, Jul./Aug. 2004.
- [43] B. Bose, *Modern Power Electronics and AC Drives*, vol. 2. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002, pp. 567–577.



Kaif Ahmed Lodi (Member, IEEE) received the B.Tech. and M.Tech. degrees in electrical engineering from Aligarh Muslim University, Aligarh, India, in 2016 and 2019, respectively. He is currently working toward the Ph.D. degree in electrical engineering with Khalifa University, Abu Dhabi, UAE.

His research interests lie in the areas of electrical drives, soft computing, power converters, and grid interface of converters.



Abdul R. Beig (Senior Member, IEEE) received the B.E. degree in electrical and electronics engineering from the National Institute of Technology Karnataka (NITK) Surathkal, Mangaluru, India, in 1989 and the M.Tech and Ph.D. degrees in electrical engineering from the Indian Institute of Science Bengaluru, Bengaluru, India, in 1998 and 2004, respectively.

He is currently an Associate Professor with the Department of Electrical Engineering, Khalifa University, Abu Dhabi, UAE. His current research interests are power and drives, electric vehicles, and renewable energy.

Dr. Beig is currently an Associate Editor for IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS and IEEE TRANSACTIONS ON TRANSPORTATION ELECTRIFICATION.



Khaled Ali Al Jaafari (Senior Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from Petroleum Institute, Abu Dhabi, UAE, in 2006 and 2011, respectively, and the Ph.D. degree in electrical engineering from Texas A & M University, College Station, TX, USA, in 2016.

He is currently an Associate Professor with the Department of Electrical Engineering, Khalifa University, Abu Dhabi, UAE. His research interests are condition monitoring, power system

protection, and power quality.



Zeyar Aung (Senior Member, IEEE) received the Ph.D. degree in computer science from the National University of Singapore, Singapore, in 2006.

He is currently an Associate Professor with the Department of Computer Science, Khalifa University, Abu Dhabi, UAE. His current research interests are in data analytic, machine learning, and their applications in cyber security, social media, renewable energy, and environmental studies.