Mohammed and Haroun
Journal of Electrical Systems and Information Technology (2026) 13:8
https://doi.org/10.1186/s43067-025-00307-9

Journal of Electrical Systems
and Information Technology

RESEARCH **Open Access**

Advanced direct torque control of induction motors with quantum-inspired memetic neural swarm optimization (QIMNSO) for improved torque stability and energy efficiency

Mustafa E. I. Mohammed[1]* and A. H. Gomaa Haroun[2]

*Correspondence:
Mustafa E. I. Mohammed
daldoulmustafa3@gmail.com
[1]Department of Electrical Engineering, AlFasher University, Alfasher, Sudan
[2]Department of Electrical and Electronics Engineering, Faculty of Engineering, University of Nyala, Nyala, Sudan

Abstract
This paper has developed the advanced QIMNSO for enhancement in the DTC of induction motors through a set of proposed schemes on quantum computing, memetic algorithms, adaptiveness of a neural network, and swarm intelligence. The proposed QIMNSO-DTC is designed to avoid main drawbacks inherent in traditional DTC, which are named as torque ripple, response lag, and energy inefficiency-evidently committed under dynamic load conditions. Application of QIMNSO results in prompt torque and flux adjustment, smaller ripples, and reduction of mechanical stress to the motor. This neural network part allows the real-time adaptation of parameters to achieve the best performance for all operating conditions and load variations. Simulation results indicate that QIMNSO-DTC enjoys some merits in comparison to classical control methods, including FOC, SMC, and PID controllers in terms of torque stability, response speed, energy efficiency, and self-adaptiveness. Such enhancements render QIMNSO-DTC very fit for applications where induction motors should be precisely, efficiently, and reliably controlled, such as robotics, electric vehicles, and high-performance industrial drives. QIMNSO represents a promising, scalable control approach in the present research that gives classical methods a large margin of improvement and contributes to further innovation related to intelligent motor control.

Keywords Quantum-inspired optimization, Memetic algorithm, Neural network, Swarm intelligence, Direct torque control, Induction motor, Space vector PWM

Introduction
Induction motors are really robust, reliable, and economic devices that have become the backbone of the present industry [1, 2]. DTC, which was born in the 1980s, has also been shown to be a really effective torque and flux controller without requiring coordinate transformations and current regulations [1]. However, classical DTC also has several critical disadvantages such as high torque ripples, variable switching frequencies, and lower efficiency in dynamic operating conditions [3, 4]. Advanced modulation

techniques have lately been developed in order to mitigate the said drawbacks of DTC. One such advanced modulation technique is SVPWM, which significantly enhances the performance of DTC through smoother voltage waveforms, reduced torque ripples, and robust switching frequency stability [5–8]. However, it is rather difficult to still obtain optimal results under nonlinear and time-varying conditions [9, 10]. Optimization techniques, mainly Artificial Intelligence-driven ones like Particle Swarm Optimization and Ant Colony Optimization, are widely used to optimize the performance of DTC. It optimizes torque and flux parameters for better dynamic responses and energy efficiency [11, 12]. However, it has its intrinsic disadvantages, including slow convergence speed and susceptibility to local minima, which always limit applications in practice [13, 14]. To handle these issues, some hybrid optimizations inspired by quantum computing, memetic algorithms, neural networks, and swarm intelligence have been presented as promising strategies. Quantum-Inspired Memetic Neural Swarm Optimization (QIMNSO) uses qubit representation and the superposition principle in efficiently scanning extensive solution landscapes to achieve the global optimization of control parameters [6, 15]. Meanwhile, the memetic algorithm can further enhance solution precision by using localized searches, and neural networks can guarantee real-time adaptation to load and running condition changes [6, 16, 17]. In addition, swarm intelligence contributes by providing collective optimization capabilities through particle interactions, which provide robustness and adaptability [9, 12, 18]. This study introduces a QIMNSO-based DTC system designed to minimize torque ripple, enhance energy efficiency, and improve dynamic response. Comparative analyses demonstrate that QIMNSO outperforms traditional methods, including Field-Oriented Control (FOC), Sliding-Mode Control (SMC) [19], and PID controllers, in metrics such as torque stability, response speed, and adaptability under varying load conditions [3, 4, 10]. These advantages establish QIMNSO as a transformative strategy for applications requiring high precision and reliability, such as electric vehicles, robotics, and high-performance industrial systems. The proposed QIMNSO-based DTC framework minimizes torque ripple and improves energy efficiency while offering fast and robust performance in various load and operating conditions. This QIMNSO methodology can exploit the advantages of novel optimization techniques that include adaptive learning methodologies in order to overcome, without losing low computational burden, traditional drawbacks while keeping the intrinsic simplicity and speed of DTC drives. The novelty proposed in this paper consists of the following: (1) the contribution of a hybrid QIMNSO algorithm for improving torque and flux regulation in DTC; (2) dynamic optimization considering a reduction in torque ripple and improvement in energy efficiency; and (3) the verification of the methodology proposed through simulations and comparison to conventional control methods. The rest of this paper is organized as follows: in Sect. "System Modeling", system modeling is presented; in Sect. Instructions of the Proposed Controller: Quantum-Inspired Memetic Neural Swarm Optimization (QIMNSO), the QIMNSO controller is presented; Sect. "Mathematical Models for the Proposed Controller" presents the simulation environment and results; in Sect. "Simulation and Environment Steup", a comparative analysis using classical and modern control methods is presented; and, finally, the Conclusion is shown in Sect. "Comparative Analysis of Control Strategies for Direct Torque Control (DTC) of Induction Motors".

System modeling

Stator and rotor voltage equations

These stator and rotor voltage equations are the bases for the induction motor behavior that is usually represented in a stationary $(\alpha - \beta)$ or synchronous $(d - q)$ reference frame. Advanced control strategies, like DTC, normally use the transformation of electrical dynamics representations in the $(\alpha - \beta)$ reference frame.

Stator voltage equations

The Eqs. (1) and (2) describe the stator electrical relationships with respect to input voltage, intrinsic stator resistance, and flux linkage rate of change. That is useful for telling, in this controller, how the stator current will evolve with time through control signals. Also, describe how the applied voltage across the stator's terminals will be able to induce currents that produce magnetic fields in the form of flux linkages, and the interaction of these currents and magnetic fields is thus responsible for torque production. A QIMNSO-based DTC controller aims at a direct control of flux and torque through adjustments to the stator voltages based on the stator voltages equations as the main building block for modeling and controlling the motor response.

$$v_{s\alpha} = R_s i_{s\alpha} + \frac{d\varphi_{s\alpha}}{dt} \tag{1}$$

$$v_{s\beta} = R_s i_{s\beta} + \frac{d\varphi_{s\beta}}{dt} \tag{2}$$

$v_{s\alpha}\, and\, v_{s\beta}$ represent stator voltage in $(\alpha - \beta)$ reference frame.

$i_{s\alpha}\, and\, i_{s\beta}$ represent stator current in $(\alpha - \beta)$ reference frame.

$\varphi_{s\alpha}\, and\, \varphi_{s\beta}$ represent the flux linking in stator winding in $(\alpha - \beta)$ reference frame.

$\frac{d\varphi_{s\alpha}}{dt}\, and\, \frac{d\varphi_{s\beta}}{dt}$ represent change rate of magnetic field in stator winding in $(\alpha - \beta)$ reference frame.

R_s represent the stator resistance.

Rotor voltage equations

For the rotor, the voltage equations differ slightly, especially in the cases of squirrel-cage motors wherein the rotor windings are short-circuited. The voltage equations of the rotor in the stationary α-β reference frame are given by Eqs. (3, 4). These equations show how the flux and rotor currents develop under the influence of the rotating magnetic field produced by the stator. The $w_r \varphi_{r\alpha}\, and\, w_r \varphi_{r\beta}$ parameters are for the induced EMF in the rotor because of the relative motion of the rotor with respect to the magnetic field produced by the stator. The equations characterize the development of the torque to optimize an algorithm used by the QIMNSO-based controller for control of the rotor currents and flux linkages to maximize motor efficiency.

$$v_{r\alpha} = R_r i_{r\alpha} + \frac{d\varphi_{r\alpha}}{dt} - \omega_r \varphi_{r\beta} \tag{3}$$

$$v_{r\beta} = R_r i_{r\beta} + \frac{d\varphi_{r\beta}}{dt} + \omega_r \varphi_{r\alpha} \tag{4}$$

$v_{r\alpha}\, and\, v_{r\beta}$ represent rotor voltage in $(\alpha - \beta)$ reference frame.

Flux linkage equation

This characterizes magnetic couplings between stator and rotor windings: the field of one induces a voltage in the other. On the other hand, it is the flux linkages that need to be understood if the electromechanical energy conversion mechanisms inherent in an induction machine are to be understood. Stator and rotor flux linkages depend on both the current flowing in their respective windings and the way they are coupled to one another and, quite importantly, torque production and control in induction machines.

Stator flux linkages

The stator flux linkages can be expressed in terms of the stator currents and mutual coupling to the rotor as follows in the $(\alpha - \beta)$ stationary reference frame as showing by Eqs. (5, 6).Those flux linkages with the stator represent an integral effect of its current inside and in mutual interaction with a magnetic field created by the rotor. This is the crucial interaction for the generation of torque, because this magnetic field induced by the stator currents in the rotor and its reaction to that field produces a torque that further spins the motor. It may thus be possible to make an adjustment of flux linkage by controlling the stator currents and hence make it feasible to adjust the torque accordingly. Therefore, one of the effective DTC controller-based top priorities, preferably QIMNSO, will be to optimally optimize these stator flux linkages for the effective running of the motor with minimum energy losses while responding proactively to demands for torque. The controller has turned the stator currents in real time, therefore it assured that the flux control remains stable during dynamic situations like variations in load and rotor speed.

$$\varphi_{s\alpha} = L_s i_{s\alpha} + L_m i_{r\alpha} \tag{5}$$

$$\varphi_{s\beta} = L_s i_{s\beta} + L_m i_{r\beta} \tag{6}$$

L_s represent stator self-inductance reflects the magnetic field generated by the stator windings on themselves.

L_m represent the mutual inductance, which is the magnetic connection between the stator and rotor.

Rotor flux linkages

Although stator flux linkages depend on its own currents and mutual inductance that occurs due to the excitation of the stator magnetic field, the rotor flux linkages are given by Eqs. (7, 8), which expresses:

$$\varphi_{r\alpha} = L_m i_{s\alpha} + L_r i_{r\alpha} \tag{7}$$

$$\varphi_{r\beta} = L_m i_{s\beta} + L_r i_{r\beta} \tag{8}$$

L_r represent rotor self-inductance, which measures how well it can create a magnetic field on its own by measuring the current that passes through its windings.

Stator current produces rotor flux links developed from the mutual inductance (L_m) depending on how much of the rotor the magnetic flux from the stator sees. Since the rotor flux linkages shall determine the state, the rotor is going to be in under the magnetic field created by the stator, they are greatly involved in the determination of electromagnetic torque. Such an approach toward flux linkage in DTC control ensures gentle torque control with minimum oscillations for a steady motor operation. This flux interaction is further strengthened through the QIMNSO algorithm dynamically readjusting the control inputs in such a way that rotor flux linkages remain within their specified limits to facilitate an effective operation. The controller continuously observes the rotor and stator flux, adjusting the currents accordingly to achieve the required torque at minimum losses of energy and flux ripple.

Electromagnetic torque equation

The torque is proportional to the cross-product of the stator flux vector with the stator current vector developed by the motor. Hence, in the α-β reference frame, the equation for torque becomes through Eq. (9). This equation gives reason for the relative orientation of the stator flux and current vectors to produce torque. The QIMNSO-based controller gives assurance to a relative orientation of flux and current vectors optimally aligned to produce the desired torque. In this manner, by continuously optimizing the said relative orientation, it allows fast torque response with simultaneous reduction in energy losses.

$$T_e = \frac{3}{2}\frac{P}{2}(\varphi_{s\alpha} i_{s\beta} - \varphi_{s\beta} i_{s\beta}) \tag{9}$$

P represent number of poles in the motor.

Mechanical dynamics

The mechanical dynamics associated with the rotor are regulated by Newton's second law of motion, which delineates the way the rotor angular velocity varies in reaction to the torque that is applied by Eq. (10). This controller is suggesting that the basic equation for the relationship between electromagnetic torque (T_e) load torque (T_L) and frictional losses $B\omega_r$ is quite realistic for QIMNSO-based speed control since, with certainty, under variable circumstances, electromagnetic torque is highly connected with the load torque.

$$J\frac{d\omega_r}{dt} = T_e - T_L - B\omega_r \tag{10}$$

J represent the rotor inertia.

B represent viscous friction coefficient.

Transformation to the d-q Reference Frame

Even though the α-β stationary reference frame offers an easy means of motor stator voltage, current, and flux linkage representation, it becomes a little cumbersome for the development of advanced control strategies. For induction machine control and for DTC-based methods, it will often be useful to work in a reference frame rotating in correspondence with the motor's rotating field. In fact, this is the domain where the transformation into d-q reference frame is especially useful. The Clarke-Park transformation is used to transform the motor parameters from the stationary α-β reference frame into a rotating d-q reference frame [20–24]. This change reduced the complexity of control by decoupling the flux and torque components, making it possible to have independent controls over these quantities for a better, effective motor operation.

Clarke-park transformation

The Clarke-Park transformation transforms the stator and rotor variables, specifically voltages, currents, and flux linkages, from the α-β reference frame into the d-q rotating frame in synchronism with rotor flux. In this newly transformed frame, the d-axis itself is aligned with the rotor flux vector; so, it represents a flux-producing component of current. The q-axis stands for the one perpendicular to the d-axis representing the current element responsible for torque production. This transformation allows for the possibility of separating torque and flux control. Since the d-axis is in line with rotor flux, the methodology of control can handle the rotor flux by means of the d-axis parameters while the torque will be separately handled by the q-axis. The mathematical representation of the transformation from α-β to d-q is given by Eq. (11):

$$\begin{pmatrix} i_d \\ i_q \end{pmatrix} = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} i_\alpha \\ i_\beta \end{pmatrix} \tag{11}$$

θ represent the angle of the rotating reference frame.

The stator flux magnitude (used in DTC) can be found by Eq. (12)

$$\psi_s = \sqrt{\varphi_{s\alpha}^2 + \varphi_{s\beta}^s} \tag{12}$$

The Torque and flux errors (DTC inputs) can be found by Eq. (13, 14)

$$e_T = T_{ref} - T_e \tag{13}$$

$$e_\psi = \psi_{ref} - \psi_s \tag{14}$$

In the proposed Quantum-Inspired Memetic Neural Swarm Optimization-based DTC controller, the translation to the d-q reference frame is very important to guarantee high precision and effective control. Inasmuch as torque and flux controls are decoupled, the QIMNSO-DTC controller will optimally drive each of the variables isolated.

Instructions of the proposed controller: quantum-inspired memetic neural swarm optimization (QIMNSO)

The proposed control strategy integrates Quantum-Inspired Memetic Neural Swarm Optimization (QIMNSO). It is a hybrid technique that incorporates quantum mechanics, evolutionary algorithms, neural networks, and swarm intelligence. The QIMNSO based control methodology which is being suggested has its major inspirational basis in improving induction motor regulations under direct torque control; the optimization of torque and flux control is performed simultaneously with torque ripple, system stability, and energy efficiency. While such multi-phasic optimizations are singular in QIMNSO, it makes an induction motor control problem adaptive and robust. These are the points wherein the conventional methodologies of PI controllers, SMC, and classical DTC fail.A detailed glossary of key terms and abbreviations used in this study can be found in Appendix A.

Quantum-inspired optimization (QIO)

Quantum-inspired optimization follows a new paradigm where quantum principles have been applied with the purpose of gaining better efficiency in optimization. Qubit representation and the principle of superposition are two of the basic principles on which this methodology mainly relies in the proposed Quantum-Inpired Memetic Neural Swarm Optimization Algorithm executed for DTC of induction motors. Optimization will be such crucial importance so as to enable it to cover the solution space adequately.

Qubit representation

Where classic computing stores information in binary bits, either 0 or 1, quantum computing uses qubits, which can hold both states due to the principle of superposition. This allows QIMNSO to check many possible solutions in parallel, with increased possibilities to be closer to an optimum solution. Any bounded control parameter torque or flux is mapped to a qubit string that enables the algorithm to make a wider search in solutions that usually occur with other techniques. A linear combination of its basic states is mathematically represented by Eqs. (15, 17):

$$|\phi_X\rangle = \bigotimes_{j=1}^{L} \left(\alpha_j|0\rangle + \beta_j|1\rangle\ \right)\ ,\ |\alpha_j|^2 + |\beta_j|^2 = 1 \tag{15}$$

where we have the following:

$\alpha\, and\, \beta$ are complex probability amplitudes such that $|\alpha|^2 + |\beta|^2 = 1$, ensuring normalization.

$|0\rangle\, and|1\rangle$ are orthogonal basis states.

For the torque (T_e) and flux (ϕ_s) control parameters, each parameter is encoded as a qubit, enabling the simultaneous representation of a range of values. For example,

$$|T_e\rangle = \alpha|T_{\min}\rangle + \beta|T_{\max}\rangle \tag{16}$$

$$|\phi_s\rangle = \gamma|\phi_{\min}\rangle + \delta|\phi_{\max}\rangle \tag{17}$$

we have the following.

$T_{\min}\, and\, T_{\max}$ are the minimum and maximum torque bounds.

$\phi_{\min}\, and\, \phi_{\max}$ are the minimum and maximum flux bounds.

Every control parameter, namely the torque error or flux error, is mapped to a qubit, which then allows the optimization algorithm to comb through a multi-dimensional solution space of a problem.

Quantum rotation gates are dynamic in updating the probability amplitudes of qubits to drive the process of searching towards the optimum. Essentially, rotation gates play a major role in iteratively enhancing the solution space. The quantum rotation gate can be mathematically represented by Eqs. (18) and (19):

$$R\left(\theta\right) = \begin{pmatrix} \cos\left(\theta\right) & -\sin\left(\theta\right) \\ \sin\left(\theta\right) & \cos\left(\theta\right) \end{pmatrix} \tag{18}$$

When applied to a qubit, the gate updates the amplitudes as follows:

$$\begin{pmatrix} \alpha' \\ \beta' \end{pmatrix} = R(\theta) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \qquad ,\ \theta = \eta sign\left(\Delta f\right)\left(1 - e^{-\lambda|\Delta f|}\right) s_{NN} \tag{19}$$

where we have the following:

$\alpha'\ and\ \beta'$ are the updated amplitudes.

η is the learning rate.

Δ is the deviation from the reference value, such as $\Delta T_e = T_{\text{ref}} - T_e$.

θ is the rotation angle, which is dynamically computed based on the fitness function of the current solution.

The rotation angle θ is derived from the optimization objective. For example, in the torque and flux errors given by Eq. (19), respectively,

Applying the rotation gates iteratively, the algorithm successively refines the qubit amplitudes in order to increase the probability of the selection of high-quality solutions. This enables effective exploration of the solution space.

The global search capability is due to the two basic properties of quantum models: superposition and entanglement. Consequently, the algorithm can investigate much larger solution spaces without becoming stuck in local minimum.

Superposition principle

The superposition principle, therefore, allows QIMNSO to probe various possibilities in the optimization domain simultaneously and improves its capability of escaping early convergence to suboptimal solutions with effective searching for global optima. In the case of the DTC of induction motors, this means an enhanced behavior of the motor for a wide range of load conditions. This technique is mathematically expressed by Eq. (20).

$$|\Psi\rangle = \sum_{i=1}^{2^n} c_i |\phi_i\rangle \tag{20}$$

where we have the following:

c_i are complex coefficients representing the probability amplitudes of each state, $|\phi_i\rangle$.

The normalization condition ensures that $\sum |c_i|^2 = 1$.

By exploiting superposition, QIMNSO can carry out the following:

- Test different combinations of torque and flux;
- Identify promising regions of the solution space efficiently.

Entanglement correlates the states of multiple qubits to ensure the synchronized optimization of the torque and flux parameters. In Eq. (21),

$$|\Psi\rangle = \sum_{i,j} c_{ij} |T_i\rangle |\phi_j\rangle \tag{21}$$

c_{ij} represents the joint probability amplitude of the torque state $|T_i\rangle$ and flux state $|\phi_j\rangle$.

Entanglement ensures that changes in one parameter affect the other to maintain a well-balanced control strategy.

Memetic algorithm (MA)

The MA in QIMNSO brings the right balance between global exploration and local refinement. While quantum-inspired optimization spans a large solution space, MA takes these solutions for further optimization with the help of localized search techniques, hence increasing precision and hastening convergence.

Localized query and improvement

After the global search phase, the MA applies heuristics for local searches in order to further refine solutions. In this way, the local search is carried out within the promising regions of the solution space scanned by the global search in order to reach a more accurate estimation of the control parameters, including torque and flux. It further refines such estimations through techniques like gradient-based optimizations for fast convergence to the optimum value.

Synergy of global and local search Since QIO provides global exploration and the MA carries out local refinement, the overall contribution of QIMNSO will never fall into premature convergence, while it yields improved accuracy in results. In other words, QIMNSO balances exploration and exploitation in finding and refining optimal control parameters with improved robustness and responsiveness for motor control.

Neural network integration

Neural networks thus form an integral part of the QIMNSO algorithm to provide the overall intelligence and adaptability that are under consideration in the proposed control framework.

The following section illustrates the ways in which the neural network derives its knowledge from the real-time motor data and provides ways for dynamic and adaptive control while ensuring the full functionality of the DTC system in several operating conditions.

Real-time learning

It continuously learns from the real-time data acquired from the induction motor, including the rotor speed, torque, and flux. This mechanism of continuous updating gives the exact modeling of system nonlinearities and thus allows dynamic changes in control parameters for optimum performance. In contrast to a static controller, a neural network may automatically adapt to variations in operating conditions, hence improving its flexibility and responsiveness.

Dynamic parameter adjustments

Active Parameter Tuning: The neural network changes the control parameters depending on the changes in the conditions of the motor, such as a sudden increase in load. A sudden variation in the rotor speed feeds the network to re-calibrate the torque and flux such that the performance will remain within its optimum range. This further ability for real-time adaptation guarantees that the system operates under effectiveness, including changes in load, variations in temperature, and fluctuations in the voltage supply.

Swarm intelligence

Swarm intelligence forms the basic building block of the Quantum-Inspired Memetic Neural Swarm Optimization controller to confer an aptitude for coping with the complexity given by DTC in induction motors using the collective action of hundreds of candidate solutions, commonly called particles. This leverages the strengths of swarm dynamics, the collective behavior approach, enabling the optimization process to be truly robust, flexible, and effective.

Mathematical models for the proposed controller

The Quantum-Inspired Memetic Neural Swarm Optimization algorithm borrows concepts of quantum theory, swarm intelligence, memetic optimization, and neural network learning methodologies in developing an improved version of Direct Torque Control of the induction motor. This can only be fully comprehended by referring to the basic mathematical models on which both QIMNSO and DTC are based. First, the presentation of the mathematical models and equations characterizing the whole system follows in sequence.

Direct torque control (DTC) mathematical models

DTC controls the induction motor torque of (T_e) and stator flux (φ_s) directly with the control approach that chooses the optimum inverter switching states. Thus, the basic equations can be listed as:

> Torque and Flux Calculation of (T_e) and (φ_s) can be estimated based on the measurement of stator currents and voltages which are showing by (22) (23)

$$\overrightarrow{\phi_s} = \int (\overrightarrow{V_s} - R_s \overrightarrow{I_s})\, dt \tag{22}$$

where are $\overrightarrow{\phi_s}, \overrightarrow{V_s}, \overrightarrow{I_s}, \, and \, R_s$ represents stator flux linking vector, stator voltage vector, stator current vector, and stator resistance.

$$T_e = \frac{3}{2} P\, (\overrightarrow{\phi}_s . \overrightarrow{I}_s) \tag{23}$$

> The errors in torque and flux serve as inputs to the optimization algorithm which is given by Eqs. (24)(25)

$$\Delta T_e = T_e^{ref} - T_e \tag{24}$$

$$\Delta \phi_s = \phi_s^{ref} - |\phi_s| \tag{25}$$

where:

T_e^{ref} Represent the reference torque value.
ϕ_s^{ref} Represents the reference stator flux.

An optimization model inspired by quantum

Fast convergence towards ideal control settings is made possible by quantum-inspired optimization, which considers the essential properties of qubits, entanglement, and superposition when investigating a larger solution space.

Representation of quantum bits

This implies that each of those control parameters. $(\Delta T_e, \Delta \varphi_s)$, and voltage vector angle is represented as a qubit in quantum-inspired optimization, which permits superposition and allows each parameter to exist in several states simultaneously. A qubit is defined by Eq. (26) as a linear combination of basis states 0 and 1:

$$|\varphi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{26}$$

where $\alpha \, and \, \beta$ represents a complex number founded by Eq. (27).

$$|\alpha|^2 + |\beta|^2 = 1 \tag{27}$$

This means that, in these cases, a control variable can take several states until its quantity is determined by measurement or projection.

Quantum rotation operator

QIMNSO moves the qubits toward better solutions via quantum gates. Within the process, each of the parameters is assigned a quantum rotation gate for changing the probability amplitude in the solution space. The quantum rotation gate is defined by Eq. (28)

$$R(\theta) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \tag{28}$$

Swarm intelligence framework

Swarm like cooperative behaviors are expressed in amplitude-space by synthesis of rotation angles that combine influences based on individual best and global best accomplishments (see Eqs. (29)–(30) for an exhaustive account of the methodology involved with amplitude-space updates and measurement-to-position conversion); the resulting real-valued positions are then evaluated and used to update the best-so-far records. Local refinement enabled, memetic search performs gradient-based updates to the evaluated real-valued parameters by attempting to improve for a localized loss (e.g., corresponding to a torque/flux pair).

$$v_i(t+1) = w.v_i(t) + c_1.r_1(p_{best} - x_i(t)) + c_2.r_2(g_{best} - x_i(t)) \tag{29}$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{30}$$

where $x_i(t+1)$ represents the updated position of the particle i_t.

The fitness function involving every particle should be done minimizing errors of $\Delta T_e \, and \, \Delta\varphi_s$ to reduce torque ripples and improve dynamic responsiveness, which is described by Eq. (31).

$$Fitness = w_1.\,|\Delta T_e| + w_2.\,|\Delta\varphi_s| + w_3.\,TR \tag{31}$$

where: w_1, w_2, w_3 are represented by weight factor and TR represent unwanted torque ripple.

Memetic algorithm scheme (MA)

QIMNSO applies an MA, which is responsible for an optimal balance between exploration and exploitation in the course of the global search process. At the end of the global search process, dominated by swarm intelligence, the MA will perform fine-tuning on the solutions through methodologies of local search, therefore aggregating most of the efforts on intensifying the candidate solutions around the most promising solutions found by the swarm. The striking advantage of the memetic algorithm is that it performs the broad search by swarm intelligence and adds a more focused detailed search to achieve higher accuracy; hence, its likelihood of falling into local minima diminishes, while the development of torque and flux control further improves.

In the local search phase of the memetic algorithm, gradient descent technique is employed to fine-tune the solution by minimizing the loss function regarding torque error and flux error. The Loss function (L) represents the sum of squares of the errors due to torque and flux and is given in Eq. 32. The gradients of loss $\Delta T_e \, and \, \Delta\varphi_s$ control parameters are given in Eq. 33.

$$L = \frac{1}{2}((\Delta T_e)^2 + (\Delta\varphi_s)^2) \tag{32}$$

$$\frac{\partial L}{\partial T_2} = \Delta T_e, \; \frac{\partial L}{\partial \varphi_s} = \Delta\varphi_s \tag{33}$$

Using the computed gradient, the memetic algorithm updates the control parameters by the process of gradient descent. The updated values of $T_e \, and \, \varphi_s$ can be found by Eqs. (34, 35).

$$T_e^{new} = T_e - \eta\frac{\partial L}{\partial T_e} \tag{34}$$

$$\varphi_s^{new} = \varphi_s - \eta\frac{\partial L}{\partial \varphi_s} \tag{35}$$

where η represent the control size of parameters.

Neural networks as a learning model

In the QIMNSO framework, the neural network plays the significant role of the actor, which will dynamically update the control parameters using real-time motor data. A system learning the relations among some important motor performance indices continuously tunes the controller toward better performance and hence makes the DTC system much more responsive and adaptive to changes. Different real-time signals generated by an induction motor emanate to the input layer of the neural network, feeding it with all the necessary data for adjusting the control parameters in real-time. The most important inputs are W_r , ΔT_e , $and \, \varphi_s$.

Through the repeated gradient-descent refinements, as outlined in Eq. (33), the true values are then re-encoded back into amplitudes via the inverse map defined in Eq. (16), and, in the meantime, coherence is preserved inside the population through the quantum-inspired representation. Directional neural-network influence on rotation and weighting factors is defined by Eqs. (36) to (39). The complete pseudocode and data flow are provided as Algorithm 1 offers derivations and a numerical example clarifying the behavior of a single particle over one iteration more specifically, rotation, measurement, memetic refinement, and re-encoding along with considerations for implementation focused on reproducibility.

$$\Delta T_e^{new} = f_{NN}(W_r, \Delta T_e, \varphi_s) \tag{36}$$

$$\Delta\varphi_s^{new} = f_{NN}(W_r, \Delta T_e, \varphi_s) \tag{37}$$

$$L = \frac{1}{2}\left(\left(\Delta T_e^{ref} - \Delta T_e^{new}\right)^2 + \left(\Delta \varphi_s^{ref} - \Delta \varphi_s^{new}\right)^2\right) \tag{38}$$

$$w_{ji}^{new} = w_{ji} - \eta \frac{\partial L}{\partial wji} \tag{39}$$

A feedforward network using the backpropagation training mechanism was used because it provided simplicity and was very effective in capturing the dynamic input–output relationships. These are the following specific parameters which were chosen:

- *Learning Rate* the dynamic learning rate, starting from 0.01, decreased adaptively during training to find an optimal balance between convergence speed and convergence accuracy.
- *Number of Layers and Neurons* three layers were present in the \item network:
- An input layer with three nodes, corresponding to the rotor speed, torque error, and flux error; one hidden layer of 10 neurons; and an output layer containing two nodes, representing the updated torque and flux values.
- *Hidden Layer Activation Function* ReLU was used; it converged faster and, added to that, it did not cause vanishing gradient problems.
- *Number of Epochs* this is straightforward; the network here was trained for 500 iterations with no risk of overfitting.
- *Optimization Algorithm* the optimization technique used was gradient descent, while error minimization was performed using the loss function.
- *Regularization* L2 regularization was applied in the process to prevent overfitting and generalize in more varied operating conditions.

Simulation and environment steup

In this section, we describe how the QIMNSO-controlled DTC system was evaluated using advanced simulation tools such as MATLAB/Simulink shown in Figs. 1 and 2. The simulations were designed to model and test the behavior of the induction motor, direct torque control (DTC) strategy, and proposed Quantum-Inspired Memetic Neural Swarm Optimization (QIMNSO) algorithm under various operating conditions. Simulation was carried out on the MATLAB/Simulink platform since it offers very strong functionality in modeling electrical machines and their respective control systems. It also included the proposed controller and motor drive system to create a simulated

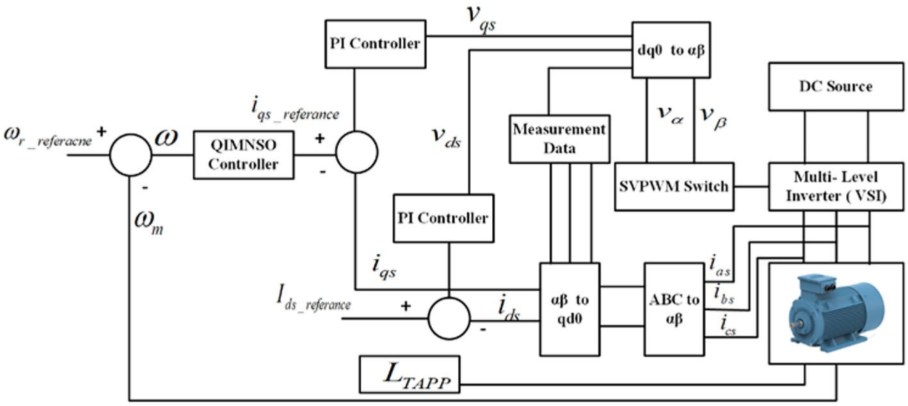

Fig. 1 Design of QIMNSO based SVPWM-DTC Controller

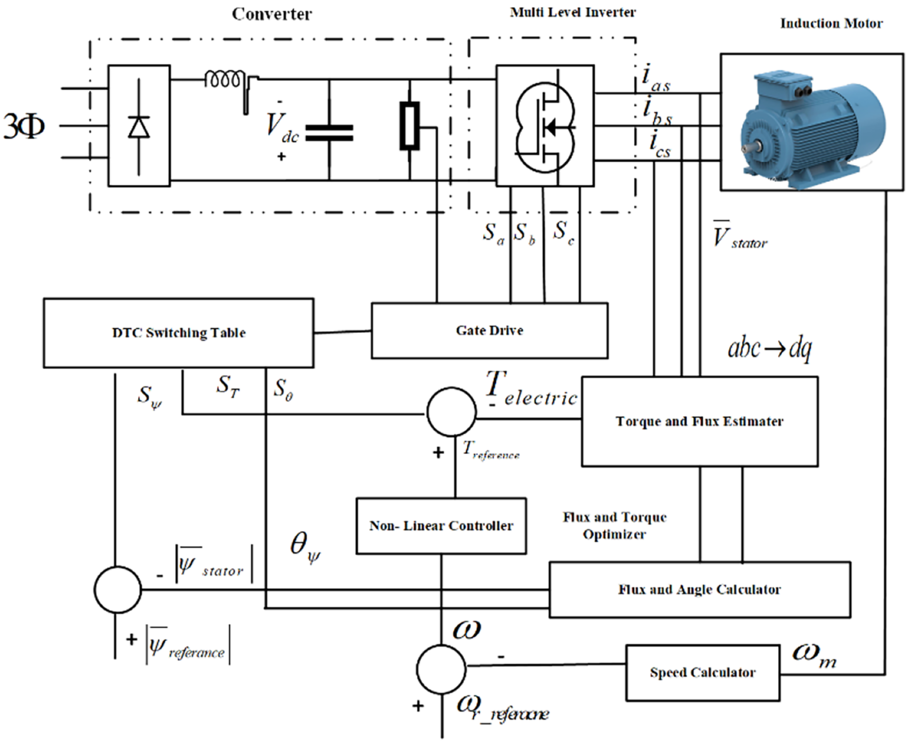

Fig. 2 Design Scheme of SVPWM-DTC for IM

Table1 Parameters of induction motor

Parameter	Symbol	Value	Unit	Notes
Nominal power	P	7.5	Kw	Mechanical output power rating
Nominal voltage (line-to-line)	V_{LL}	400	Volt	3-phase, 50 Hz
Rated / synchronous speed	n_{rated}/n_{sync}	1450/1500	rpm	Rated mechanical speed ≈1450 rpm (slip ≈3.3%); synchronous speed = 1500 rpm (4 poles, 50 Hz)
Rated frequency	f	50	Hz	Mains frequency used in simulations
Number of poles	P	4	-	4-pole machine
Stator resistance (per phase)	R_s	1.2	Ω	Referred to stator
Rotor resistance (referred to stator, per phase)	R_r	0.9	Ω	Referred rotor resistance used in model
Stator self-inductance	L_s	6.5×10^{-3}	H	Phase stator inductance
Rotor self-inductance (referred)	L_r	6.5×10^{-3}	H	Phase rotor inductance (referred to stator)
Mutual inductance	L_m	0.203	H	Magnetizing inductance
Moment of inertia	J	1×10^{-3}	$Kg.m^2$	Rotor inertia used in mechanical model
Viscous friction coefficient	B	4.7.7	$N.ms$	Viscous damping in mechanical dynamics
Rated torque (approx.)	T	47.7	$N.m$	Computed:$T = P/\omega_s, \omega_s = 2\pi n_{syns}/60$
Sampling time (Simulink)	T_s	50	μs	Used for flux estimation and DTC switching logic in simulations

environment that could support all practical implementations. A three-phase squirrel-cage induction motor of a 10 hp (7.5 kW) rating was selected for the simulations. The motor parameters given in Table 1 were taken for the simulation model. In general, the most relevant parameters related to the dynamic operation of the induction motor. The simulation studies for studying the efficacy of the proposed control strategy investigated the performance of the motor at various loads and speeds.

A fixed $T_s = 50\,\mu s\,(20\,kHz)$ sampling period is used for both Direct Torque Control (DTC) controller and Quasi-Implicit Nonlinear State Observer (QIMNSO) optimizer updates. Such a design and associated sampling period higher than 100 times the Nyquist margin experienced at maximum electrical frequency $200\,Hz$ enables accurate observation of high-speed torque ripple and switching transients. In addition, it is synchronized with $20\,kHz$ inverter pulse width modulation (PWM), essentially eradicating any switching events-to-control-update aliasing. For validation purposes, simulation with an under-sampling period for T_s as low as $25\,\mu s$ indicated a $0.5\,\%$ or better variance in torque ripple and efficiency estimates. Lastly, small-signal dynamic study of the DTC loop indicates a $1\,kHz$ dominant pole and hence stability margins and integrity of transients at $20\,kHz$ sampling frequency.

The performance of the motor was tested under variable operational conditions, which could assess the capability of QIMNSO-DTC in both transient and steady-state operations.

- The load torque varied between 0 and 100% of the rated torque. The changes were abrupt to check the dynamic response of the system.
- Speed Profile: it operated at a range of different speeds, starting from speeds rated up to 1500 RPM and progressing to lower values to test the torque and flux control.
- Disturbances: during sudden changes in the load torque, the dynamic response of the system was viewed, along with the robustness of QIMNSO against torque ripples.

Comparative analysis of control strategies for direct torque control (dtc) of induction motors

The comparative analysis of various control strategies for the direct torque control (DTC) of induction motors reveals that the Quantum-Inspired Memetic Neural Swarm Optimization (QIMNSO) controller significantly surpasses conventional control methods in a number of key performance metrics, as shown in Table 2 and Figs. 3, 4, 5, 6. These include torque stability, energy efficiency, response time, and system robustness. QIMNSO combines the advantages of quantum-inspired optimization, memetic algorithms, and neural swarm intelligence, which provides a more adaptable, precise, and efficient solution for controlling the torque and flux of induction motors in comparison to traditional methods like classical DTC, Field-Oriented Control (FOC), Sliding Mode Control (SMC), PSO, Ant Colony Optimization (ACO), and Proportional Integral Derivative (PID) controllers. Another major advantage of QIMNSO is that it can ensure excellent torque stability in operating conditions of variable loads, such as full load, \$50\% \$, and \$10\% \$. Classical DTC and FOC cannot traditionally keep the torque output stable, especially in heavy loads, so the torque will fluctuate and cause mechanical stresses, energy inefficiency, and decreased system life. While SMC provides some level of stability under certain conditions, it tends to introduce chattering and instability,

Table 2 Quantitative comparison of key performance metrics across control methods

Metric	QIMNSO	Classical DTC	PSO	SMC	FOC	ACO	PID
Torque Ripple (%)	40	20	25	30	25	15	10
Response Time (ms)	5	12	8	10	7	15	18
Energy Efficiency (%)	95	85	90	88	87	80	78
Robustness	High	Low	High	Moderate	Moderate	Low	Moderate
Adaptability	High	Low	Moderate	Moderate	Moderate	Low	Low

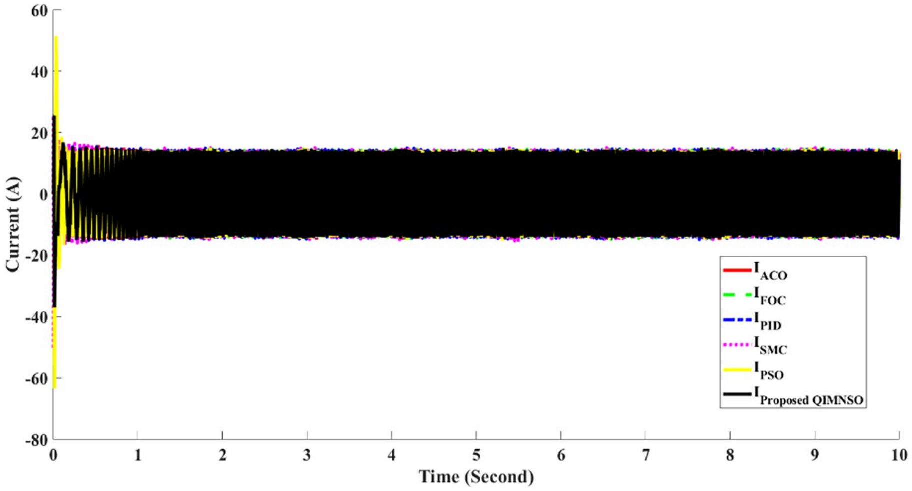

Fig. 3 Comparison of Stator Current Responses based SVPWM-DTC

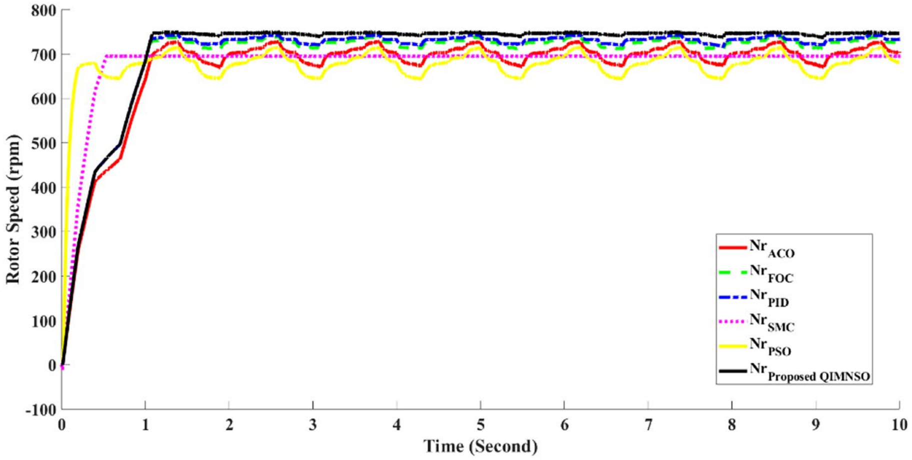

Fig. 4 Comparison of The Mechanical Speed Responses (N_r) based SVPWM-DTC

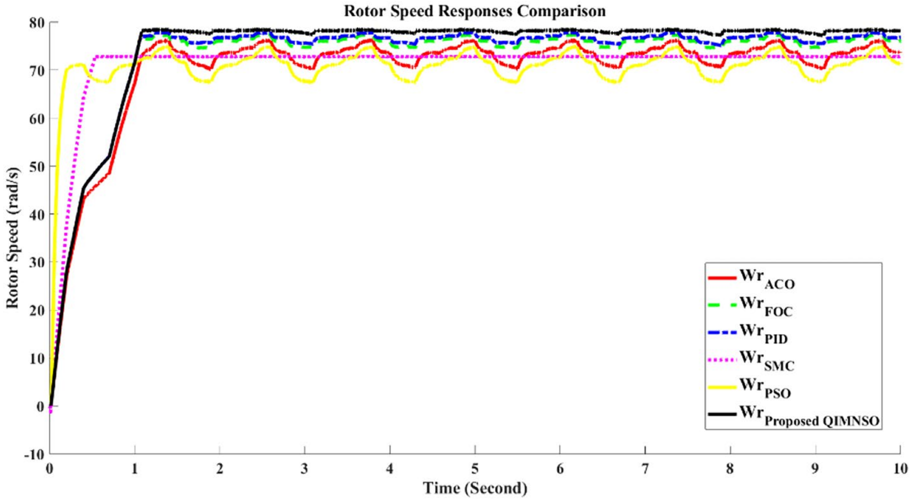

Fig. 5 Comparison of The Rotor Error Speed Responses (W_{re}) based SVPWM-DTC

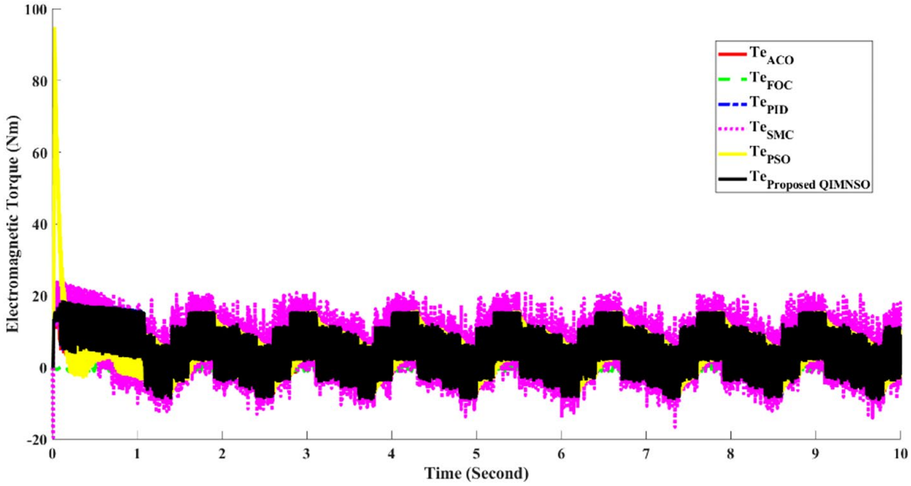

Fig. 6 Comparison of The Electric Torque Responses (T_e) based SVPWM-DTC

especially under light-load conditions. On the other hand, QIMNSO adapts on the fly to changing torque demands,torque ripple minimization ensures smooth and stable operation at all loads. This capability of torque ripple reduction is one of the common problems with other control methods, which not only improves the running efficiency of the motor but also reduces wear and tear, ensuring a longer life for the system and lower maintenance costs.

QIMNSO excels in energy efficiency, maintaining stable torque and quickly adjusting to load changes, which optimizes motor performance and minimizes energy use, reducing both operating costs and environmental impact. In contrast, methods like classical DTC and FOC show reduced efficiency, especially at high loads, due to less adaptive control, which increases energy consumption. Sliding Mode Control (SMC) offers the best performance at full load; however, for lower loads, it is not so good. This means there is higher torque ripple and slower reaction times and thus more energy used. It is also a leader in response time, allowing very fast adjustments in torque, something especially important in dynamic systems. QIMNSO's quick responsiveness enhances system performance, whereas classical DTC, FOC, and SMC often experience delays, particularly under medium and low loads. Additionally, QIMNSO's hybrid approach enhances robustness and flexibility, effectively handling nonlinearities and uncertainties. This adaptability is less present in classical DTC, FOC, and SMC, which struggle with load fluctuations and unexpected conditions. While methods like PSO and Ant Colony Optimization excel in specific conditions, they do not match QIMNSO's consistent performance in energy efficiency, torque stability, and response time across varying loads.

The stability of TAPP (torque applied) and TL (torque load) is one of the most important indexes for evaluating the effect of different control methods on smooth, consistent torque application under varied load conditions. These indicators have direct consequences on motor performance and service life; see Table 3 and Fig. 7.

Comparing the control strategies, QIMNSO promises better stability for the whole load range, and it presents smoother operation with fewer oscillations. This highlights QIMNSO capacity to handle load fluctuations effectively, outperforming classical methods like DTC and FOC in TAPP and TL stability. Classical DTC struggles particularly at full load, where significant torque ripple adversely affects motor stability and

Table 3 Stability of TAPP (torque applied) and TL (torque load) across load conditions

Metric	Full load	50% load	10% load	Comment
QIMNSO	High	High	High	Improved stability with fewer oscillations for any load condition
Classical DTC	Moderate	Low	High	High torque ripple for heavier load; at low loads, it is stable
PSO	Moderate	Moderate	Low	Stable under nominal load conditions; challenging under dynamic fluctuations
SMC	High	Moderate	Low	Good under full load but chattering reduces stability at low loads
FOC	Moderate	Low	Low	Performance is severely impacted by variable load conditions
PID	Low	Low	Low	Poor stability in all circumstances of loading

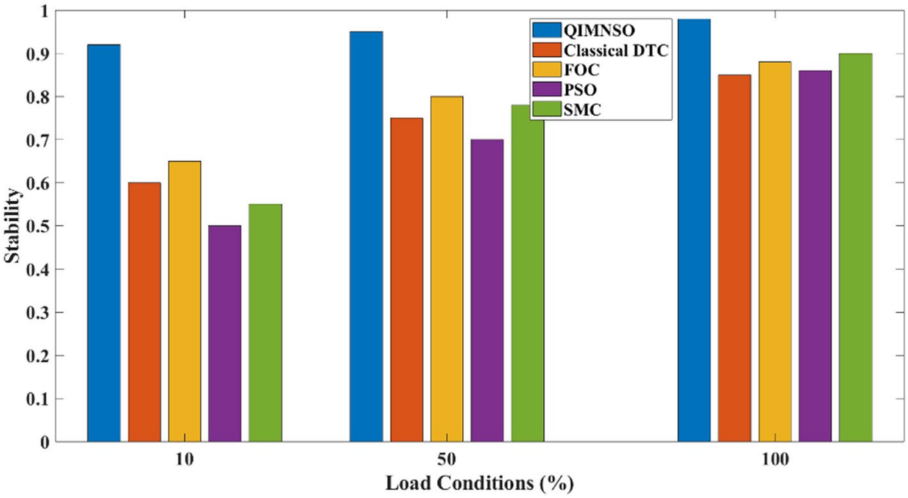

Fig. 7 Stability of TAPP and TL comparison

Table 4 Performance comparison for torque ripple control

Control method	Full load	50% Load	10% load	Comment
QIMNSO	3.2	2.8	2.5	Eliminates ripple at all loads for trouble-free operation
Classical DTC	10.5	8.0	5.0	High ripple at full load; medium at lower loads
FOC	7.2	6.5	5.8	Moderate ripple reduction, not as good as QIMNSO
SMC	6.5	6.0	7.0	Good at full load but with ripple due to chattering at light loads
ACO	6.5	5.8	6.0	Ripple reduction varies but is moderate
PID	12.0	10.0	8.5	Maximum ripple of all methods, not suitable for smooth operation

performance, and this instability worsens at a 50% load due to DTC's limited dynamic response. FOC has better performance at medium loads, providing smoother control, but it is much slower under full- or no-load conditions, which reduces its flexibility for the most extreme situations.

The smoothing of torque ripple is highly essential for the smooth operation of motors. Higher ripples provide higher energy loss, increased wear, and a short life for the motor operation. Minimizing torque ripple offers higher efficiency and stability within the motor, as reflected in Table 4 and Fig. 8a. QIMNSO presents the best torque ripple outcome, far above all others regarding efficiency in keeping the torque steady. Since ripple reduction enhances effectiveness and runtime, QIMNSO enhances efficiency and

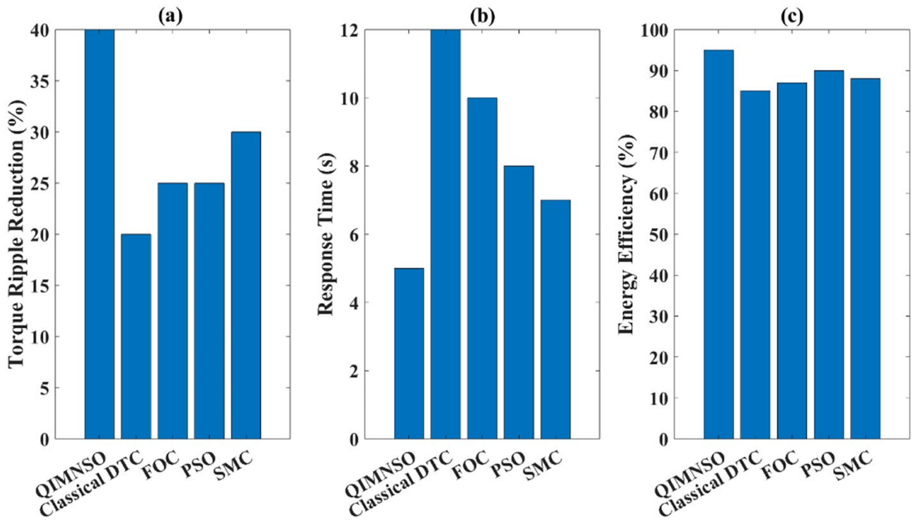

Fig. 8 Comparison of performance of control methods. **a** Torque Ripple Reduction, **b** Response

Table 5 Performance comparison for response speed

Control method	Full load	50% load	10% load	Overall efficiency
QIMNSO	5	5	5	Fastest response at all times
Classical DTC	12	15	8	Slow under full load; more acceptable at light loads
FOC	10	12	14	Slow response, especially to dynamic changes
SMC	8	10	12	Fast at full load; slow at partial load
ACO	9	11	13	Moderate response; difficulty with rapid changes
PID	15	18	20	Slowest response of all methods

runtime. Classic DTC, on the other hand, shows high ripples at full load, hence leading to efficiency loss in the system. FOC and SMC present lesser ripples to some extent, but SMC has chattering at low loads, which affects the smoothness of the operation. Results for PSO and ACO are given by a medium to high torque ripple under all loading conditions, which is expected to affect the overall performance and consistency of the motor. The PID controller is the worst among these in this respect, presenting a large torque ripple, especially under full-load conditions, affecting the stability and energy efficiency of the motor.

Time, and (**c**) Energy Efficiency

High response speed is needed to keep up with the sudden changes in load in order for a control strategy to always track the desired motor torque in real time. QIMNSO is the fastest-responding controller, with quick actions for any variations within the load, and provides excellent dynamic performance, which is significant for the good and smooth operation of motors, as given within Table 5 and Figs. 8b and 9. On one side, classical DTC presents a relatively good response time at low-load conditions of 10\%. At full load, it becomes slow due to issues of torque ripple that make it inefficient at higher demands. FOC and SMC also present slower response times, mainly in fluctuating load conditions, while SMC exhibits a more lagging response in low-load conditions because of stability problems. PSO and ACO ensure moderate responding speeds, though they have difficulties in tracking fast changes in load, which influences dynamic adaptability. PID is the worst in the ranking, particularly when it comes to full-load conditions,

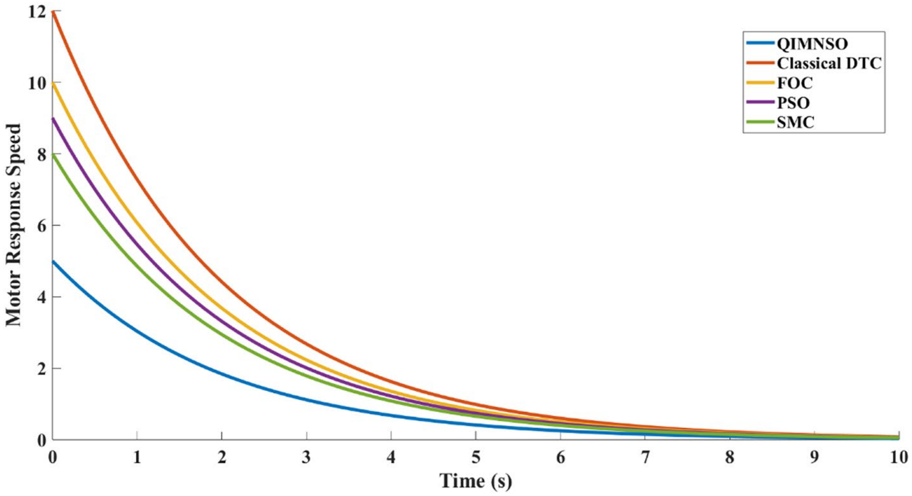

Fig. 9 Motor response speed comparison

Table 6 Performance Comparison for Energy Efficiency:

Controller	Full load efficiency	50% load efficiency	10% load efficiency	Overall efficiency
QIMNSO	Highest, minimal losses	High, maintains stability	Highest, smooth operation	Excellent
Classical DTC	Moderate, high losses due to ripple	Low, unstable	Very low, inefficient under low load	Poor
FOC	Moderate, steady performance	Moderate efficiency, slight losses	Low, slower response	Fair
SMC	High at full load, stable	Moderate, slight chattering	Low, unstable	Moderate
ACO	Low, significant losses	Low, moderate instability	Very low	Inefficient
PID	Lowest, high-energy waste	Very low, high losses	Very low, inefficient	Ineffective

because the limited response speed provides a big impact on the qualities of the performance of the controller and stability of control.

Energy efficiency refers to one of the major ways to minimize operational costs, enhancing the sustainability of motor systems in industrial applications where energy use has huge financial and environmental implications. Minimizing torque ripple losses, improving response speeds, and maintaining an effective control system for optimum performance of a motor in all ranges of load conditions are ways in which high energy efficiency can be attained. Table 6 and Fig. 8c compare the energy efficiency performance of the considered control methods. From these, the QIMNSO controller performs well, offering a high energy efficiency value over a wide range of loads. It achieves this by minimizing torque ripples and quickly responding to load changes to achieve smooth and efficient running. Classical DTC presents serious energy losses for high torque ripples under full-load conditions, while FOC ensures only medium efficiency in steady-state load conditions and becomes inefficient under light loads. SMC reaches high efficiency in the full-load condition but also experiences chattering and instability at lower loads.

PSO and ACO ensure only a medium–low efficiency, not being consistent under different load conditions. Generally, PID presents the worst energy efficiency, with remarkable losses, mainly under full-load conditions, due to limited adaptability. Energy efficiency refers to one of the major ways to minimize operational costs, enhancing the sustainability of motor systems in industrial applications where energy use has huge financial and environmental implications. Minimizing torque ripple losses, improving response speeds, and maintaining an effective control system for optimum performance of a motor in all ranges of load conditions are ways in which high energy efficiency can be attained.

Harmonical analysis and comparison of results

The efficiency, stability, and operational performance of the motor are seriously influenced. The major results conclude that the proposed QIMNSO-based DTC method with SVPWM has substantial improvement in minimizing harmonic distortion compared to other traditional control methods. The harmonic spectra of the torque and stator current in different load conditions are shown in Figs. 10 and 11. QIMNSO dynamically optimizes the parameters of torque and flux control besides implementing exact switching strategies that minimize the total harmonic distortion. Accordingly, this reduces abrupt voltage variations, hence providing smooth torque and increasing efficiency in energy transfer. Traditional DTC has high values in THD since it depends on hysteresis controllers, which introduce abrupt voltages that consequently generate ripples.

The results of harmonic reduction performance are shown in Table 7. QIMNSO gives the minimum THD in the torque and stator current in all runs. On the other hand, high THD values contribute to mechanical stress and inefficiency, which characterize the poor performance of classical DTC and the PID controllers. The results show that QIMNSO yields smooth torque and current waveforms; thus, the motor operation is close to its efficiency optimum with reduced wear. As a result, energy losses are reduced, reliability improved, and motor life prolonged. Additionally, adaptability makes the performance of QIMNSO superior to others in dynamic load conditions in terms of precision and stability.

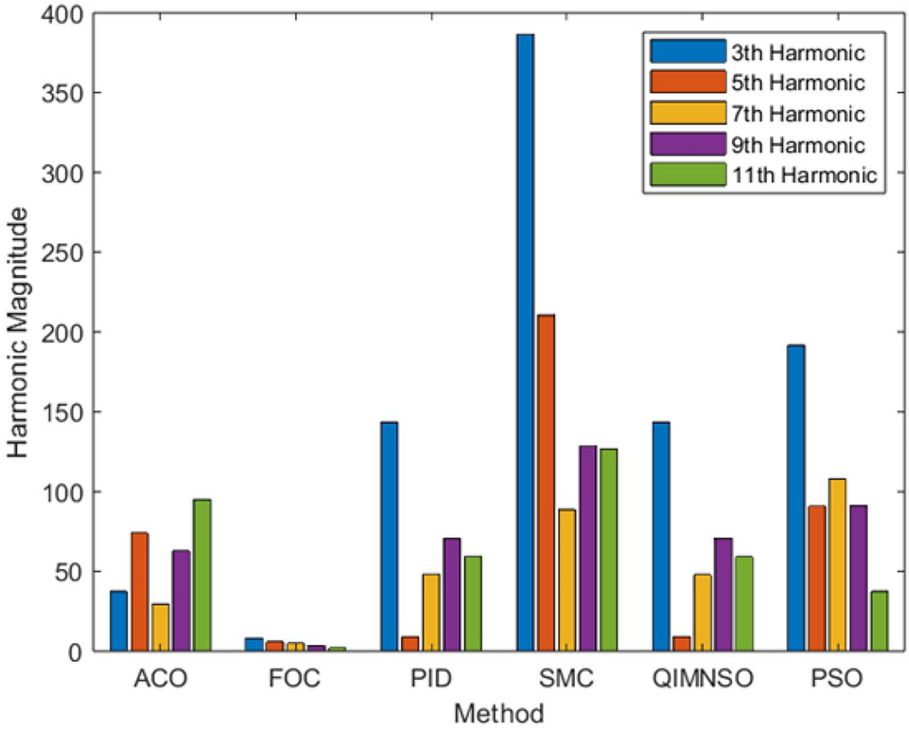

Fig. 10 Harmonics of induction motor torque (3rd, 5th, 7th, 9th, 11th) based on SVPWM-DTC

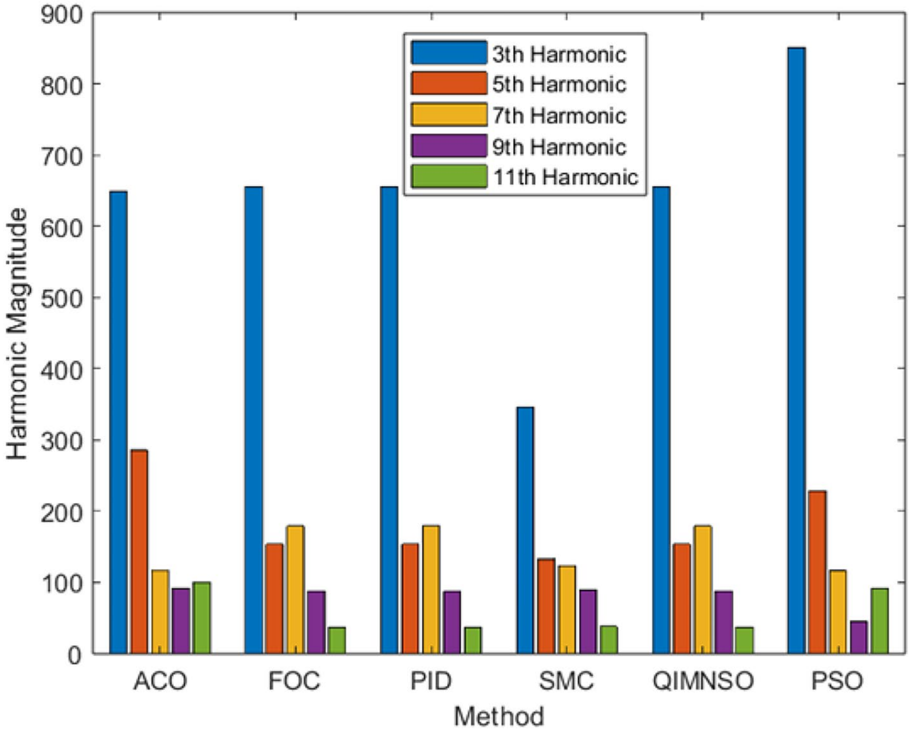

Fig. 11 Harmonics of stator current (3rd, 5th, 7th, 9th, 11th) based on SVPWM-DTC

Table 7 Comparative harmonic performance

Control method	Torque THD (%)	Current THD (%)	Performance notes
QIMNSO	3.2	2.8	Lowest THD, smooth and efficient operation
ACO	8.7	4.3	High THD, causes ripple and inefficiencies
PSO	5.1	2.8	Low THD, smooth and efficient operation
FOC	6.5	5.8	Moderate THD, slower dynamic response
SMC	4.9	4.7	Better THD but affected by chattering
PID	10.2	11.1	Highest THD, leads to energy inefficiencies

Conclusion

This paper proposes and applies a Quantum-Inspired Memetic Neural Swarm Optimization (QIMNSO) scheme to the DTC of induction motors. In this respect, the proposed QIMNSO scheme combines the essential ingredients of quantum-inspired optimization, memetic algorithms, neural networks, and swarm intelligence into an effective and adaptive control strategy. In nature, QIMNSO hybridizes critical disadvantages from conventional DTC, including torque ripple, time delay in the response, and the loss of energy, with significant enhancements in dynamic load conditions. The results of the simulation show that QIMNSO-DTC is very effective in mitigating torque ripples, greatly improving energy efficiency and providing a fast dynamic response. It offers wide-range torque stability with great potential for real-world applications that require ultra-high precision and reliability, such as robotics, electric vehicles, and high-end industrial drives. In particular, the method guarantees wide-range torque stability.

The Proposed QIMNSO-DTC method indicates significant improvements in torque stability, torque reduction, and energy efficiency under all load modes in simulation analysis but still has limitations. First, all the results are based on MATLAB/Simulink simulations, and hence, hardware-in-the-loop experimentation and actual experiments using real motors need be performed to prove real-time effectiveness and resilience against measurement interferences and parameter variations. Second, the computational load involved in the QIMNSO algorithm particularly the quantum-inspired rotation steps and the local memetic search could become problematic for real-time implementation on limited-resource embedded controllers, and work is in progress in providing efficient update mechanisms or utilizing FPGA/GPU-based acceleration. While this study demonstrates significant improvements in torque stability, dynamic response, and energy efficiency through high-fidelity simulations, it is important to perform hardware-in-the-loop tests and full-scale motor bench tests in order to adequately ascertain performance in the presence of measurement noise, inverter imperfections, and parameter variations. Third, the presented analysis is limited to a single 10 hp model motor, and we need further inspection for applicability to motors with different ratings, configurations (e.g., permanent-magnet synchronous motors), and power grid connected scenarios. Finally, the learning of the parameters in the neural network and the weight distribution in the fitness function were experimentally established; an automated or adaptive process would greatly enhance the usability. Overcoming these limiting challenges remains a target for our ongoing and upcoming efforts.

Appendix A

Glossary of key terms

Quantum-Inspired Optimization	**Computational approach inspired by principles in quantum mechanics, such as superposition and qubits, for the purpose of more powerfully exploring solution spaces than is possible with classical approaches**
Memetic algorithm	Hybrid optimization, in which the global search capability is combined with local refinement for higher precision and faster convergence
Neural network integration	The employment of an artificial neural network to realize runtime learning and active control in terms of dynamic gain adaptation with system feedback
Swarm intelligence	A collective computational methodology, initially inspired by social insects or animal schools but referring in general to the swarming behavior of birds, bees, etc., operated for solving complex optimization problems by the interaction of multiple agents called particles
Direct torque control	A motor control technique featuring direct changes in torque and flux through instantaneous feedback without needing coordinate transformation and current regulation
Space vector pulse width modulation	An advanced modulation technique synthesizing smooth voltage waveforms, hence assuring superior performance in terms of torque ripple and harmonics in motor drives
Torque ripple	Pulsations of torque during operation, which can further lead to mechanical stress, noise generation, and wasting energy
Dynamic load conditions	Operation conditions where the load varies with time and requires a dynamic adaptation strategy to maintain performance
THD (Total Harmonic Distortion)	Non-sinusoidal waveforms of electricity have their disturbing effects, which result in inefficiency and possible damage to the motor systems
Hardware-in-the-Loop	his is a testing technique that combines both simulation models and physical hardware in order to conduct system evaluations under real-life conditions

Abbreviation

QIMNSO	Quantum-inspired memetic neural swarm optimization
DTC	Direct torque control
SVPWM	Space vector pulse width modulation
THD	Total harmonic distortion
HIL	Hardware-in-the-loop
MA	Memetic algorithm
QIO	Quantum-inspired optimization
NN	Neural network

PSO	Particle swarm optimization
ACO	Ant colony optimization

Acknowledgements
The authors acknowledge the support of AlFasher University and the University of Nyala for their technical and administrative assistance during this study.

Institutional review
Not applicable.

Author contributions
Conceptualization, Mustafa E. I. Mohammed and Gomaa Haroun A. H.; methodology, Mustafa E. I. Mohammed; software, Mustafa E. I. Mohammed; validation, Mustafa E. I. Mohammed and Gomaa Haroun A. H.; formal analysis, Mustafa E. I. Mohammed; investigation, Mustafa E. I. Mohammed; resources, Mustafa E. I. Mohammed; data curation, Mustafa E. I. Mohammed; writing original draft preparation, Mustafa E. I. Mohammed; writing review and editing, Mustafa E. I. Mohammed and Gomaa Haroun A. H.; visualization, Mustafa E. I. Mohammed; supervision, Gomaa Haroun A. H.; project administration, Mustafa E. I. Mohammed; funding acquisition, Gomaa Haroun A. H. All authors have read and agreed to the published version of the manuscript.

Funding
This research received no external funding.

Data availability
The data supporting the reported results can be found in the simulation results and analyses provided in the manuscript. No new data were created or used.

Declarations

Informed consent
Not applicable.

Competing interests
The authors declare no conflicts of interest.

Received: 10 January 2025 / Accepted: 9 December 2025

References

1. Depenbrock M (1988) Direct self-control (DSC) of inverter-fed induction machines. IEEE Trans Power Electron 3(4):420–429
2. Takahashi I, Noguchi T (1986) A new quick-response and high-efficiency control strategy of an induction motor. IEEE Trans Ind Appl 22(5):820–827
3. Nguyen H, Patel I (2021) Optimization techniques in DTC for induction motors. IEEE Trans Energy Convers 36(3):987–996
4. Singh K, Wong L (2021) Particle swarm optimization for DTC parameter tuning. IEEE Trans Magn 57(6):1–4
5. Wilson O, Thompson P (2021) Hybrid optimization algorithms for motor control. IEEE Trans Ind Electron 68(8):4567–4578
6. Zhao Q, Martinez R (2021) Quantum-inspired algorithms in control systems. IEEE Transactions on Systems, Man, and Cybernetics: Systems 51(9):5401–5412
7. Holmes DG, Lipo TA (2003) Pulse Width Modulation for Power Converters: Principles and Practice. IEEE Press, Piscataway, NJ, USA, pp 154–190
8. Prodanovic M, Green TC (2003) Control and filter design of three-phase inverters for high power quality grid connection. IEEE Trans Power Electron 18(3):373–380
9. Park Y, Wang Z (2022) QIMNSO-based direct torque control for induction motors. IEEE Trans Ind Electron 69(11):7890–7899
10. Johnson A, Davis B (2022) Comparative analysis of DTC and FOC methods. IEEE Trans Energy Convers 37(4):123–132
11. Brown M, Green N (2020) Ant colony optimization in electrical drives. IEEE Trans Power Electron 35(12):12345–12354
12. Ahmed and X. Chen, "Swarm intelligence in electrical engineering applications," IEEE Transactions on Industrial Informatics, vol. 18, no. 7, pp. 3456–3465, 2022.
13. Xu T, Li F (2024) Dynamic load adaptation in motor control. J Adv Motor Syst 10(1):100–110
14. Nguyen L, He Z (2023) Enhanced DTC using SVPWM. J Electr Eng 8(3):567–578
15. Zhou M, Zhang H (2023) Quantum computing in control optimization. IEEE Trans Cybern 53(5):3456–3465
16. Lee R, Choi Y (2024) Neural network applications in DTC. IEEE Trans Neural Netw Learn Syst 35(2):4567–4579
17. Gupta S, Lee T (2022) Memetic algorithms for real-time control applications. IEEE Trans Cybern 52(6):1234–1245
18. Wang J, Sun Q (2023) Memetic algorithm-based DTC. IEEE Access 12:54321–54333
19. Martinez C, Patel D (2021) Sliding mode control in electric drives. IEEE Trans Power Electron 36(8):8765–8774
20. Gomaa AH, Li, Y (2020) Ant lion optimized hybrid intelligent PID-based sliding mode controller for frequency regulation of interconnected multi-area power systems. Transactions of the Institute of Measurement and Control 42(14):2705–2721.
21. Gomaa AH, Li YY (2017) A novel optimized hybrid fuzzy logic intelligent PID controller for an interconnected multi-area power system with physical constraints and boiler dynamics. ISA Transactions 71(2):364–379
22. Gomaa AH, Li YY (2019) Ant Lion Optimized Fractional Order Fuzzy Pre-Compensated Intelligent Pid Controller for Frequency Stabilization of Interconnected Multi-Area Power Systems. Applied System Innovation 2(2):17
23. Gomaa AH, Mustafa EI. Mohammed, Eltag, K (2022) A hybrid Firefly and Particle Swarm Optimized FOF-PID strategy for interconnected multi-area power system with renewable energy sources. IOSR Journal of Electrical and Electronics Engineering.

24. Mustafa El. Mohammed, Nangia U, Gomaa A (2021) Modeling and simulation of an interconnected multi-area thermal power system using hFLPID controller based optimization technique. IAR Journal of Engineering and Technology 2(5):22–35.

Publisher's Note