

Article

Optimization of PI Controller Parameters by GWO Algorithm for Five-Phase Asynchronous Motor

Malika Fodil ¹, Ali Djerioui ^{2,*} , Mohamed Ladjal ¹ , Abdelhakim Saim ^{3,*} , Fouad Berrabah ² , Hemza Mekki ², Samir Zeglache ¹, Azeddine Houari ³  and Mohamed Fouad Benkhoris ³

¹ LASS, Laboratoire d'Analyse des and nd Gnaux et Systemes, University Mohamed Boudiaf of M'Sila, M'Sila 28000, Algeria; malika.fodil@univ-msila.dz (M.F.); mohamed.ladjal@univ-msila.dz (M.L.); samir.zeglache@univ.msila.dz (S.Z.)

² LGE, Laboratoire de Genie Electrique, University Mohamed Boudiaf of M'Sila, M'Sila 28000, Algeria; fouad.berrabah@univ-msila.dz (F.B.); hemza.mekki@univ-msila.dz (H.M.)

³ IREENA Laboratory, University of Nantes, 44600 Saint-Nazaire, France; azeddine.houari@univ-nantes.fr (A.H.); mohamed-fouad.benkhoris@univ-nantes.fr (M.F.B.)

* Correspondence: ali.djerioui@univ-msila.dz (A.D.); abdelhakim.saim@univ-nantes.fr (A.S.)

Abstract: Operation at low speed and high torque can lead to the generation of strong ripples in the speed, which can deteriorate the system. To reduce the speed oscillations when operating a five-phase asynchronous motor at low speed, in this article, we propose a control method based on Gray Wolf optimization (GWO) algorithms to adjust the parameters of proportional–integral (PI) controllers. Proportional–integral controllers are commonly used in control systems to regulate the speed and current of a motor. The controller parameters, such as the integral gain and proportional gain, can be adjusted to improve the control performance. Specifically, reducing the integral gain can help reduce the oscillations at low speeds. The proportional–integral controller is insensitive to parametric variations; however, when we employ a GWO optimization strategy based on PI controller parameters, and when we choose gains wisely, the system becomes more reliable. The obtained results show that the hybrid control of the five-phase induction motor (IM) offers high performance in the permanent and transient states. In addition, with this proposed strategy controller, disturbances do not affect motor performance.

Keywords: PI-controller; five-phase asynchronous motor; Gray Wolf optimization; speed oscillations



Citation: Fodil, M.; Djerioui, A.; Ladjal, M.; Saim, A.; Berrabah, F.; Mekki, H.; Zeglache, S.; Houari, A.; Benkhoris, M.F. Optimization of PI Controller Parameters by GWO Algorithm for Five-Phase Asynchronous Motor. *Energies* **2023**, *16*, 4251. <https://doi.org/10.3390/en16104251>

Academic Editor: Dinko Vukadinović

Received: 16 April 2023

Revised: 12 May 2023

Accepted: 16 May 2023

Published: 22 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The study and control of the asynchronous machine are currently one of the major concerns of researchers in the field compared to the direct current machine; this machine has many advantages in terms of dynamic performance, longevity, and its ability to withstand overloads, which justifies its use in various industrial environments, including electric traction [1]. However, the reduced number of phases can be a handicap in particular areas of application, such as rail traction, naval propulsion, automotive, and aerospace [2]. In contrast, the properties of polyphase machines are very appealing, as a rise in the number of phases results in a decrease in the ripples of the electromagnetic torque [3,4]. The fractionation of the power, which results in a reduction in the sizing constraints on the power electronic components supplying each phase in addition to this increase, makes it possible to operate one or more faulty phases [3]. This issue is fundamental for applications that must guarantee an excellent continuity of service.

Many works have studied the control of polyphase machines, such as the flux orientation control [3], the fuzzy logic control [5], the direct torque control (DTC) [4], the predictive control (PC) [6], and the sliding mode control (MGC) [7]. Research has aimed to develop a control system for the polyphase asynchronous machine that operates at low speed and

high torque to reduce the occurrence of periodic torque ripples. These ripples can cause speed oscillations and potentially harm the machine [8].

The fields of control and regulation theory have been considerably developed over the past decades; however, the importance of the proportional–integral–derivative (PID) controller in the industry remains unaffected. Because of its straightforward design, it is simple to comprehend and operate, as well as resilient and stable across a wide range [5].

Unfortunately, proportional–integral (PI) controllers are widely used in control systems to regulate a process variable, such as the speed or position of a motor. The PI controller has two parameters, the proportional gain (K_p) and integral gain (K_i), that determine the response of the controller. The PI controller parameters can be tuned using several methods, including metaheuristic algorithms [9]. Metaheuristic algorithms are optimization algorithms that can be used to find optimal solutions for complex problems. Here are some examples of metaheuristic algorithms that can be used to tune PI controllers: Inspired by the social behavior of bird flocks, Particle Swarm Optimization (PSO) is an optimization algorithm based on population. In PSO, a population of particles is used to search for the optimal solution by adjusting their positions based on the best solution [10]. Genetic Algorithm (GA) is an optimization algorithm that imitates the process of natural selection. It uses a group of individuals to explore the optimal solution by applying genetic operators, such as crossover and mutation, to produce new solutions [11–14]. The optimization algorithm known as Ant Colony Optimization (ACO) draws inspiration from the foraging behavior of ants. In ACO, a population of ants is used to search for the optimal solution by depositing pheromones on the paths that lead to better solutions [15,16]. Inspired by the foraging behavior of honeybees, Artificial Bee Colony (ABC) is an optimization algorithm. The Artificial Bee Colony (ABC) algorithm imitates the behavior of real bees, where a population of artificial bees is employed to explore the optimal solution [17]. Grey Wolf Optimization (GWO) is an optimization algorithm inspired by the hunting behavior of grey wolves [18]. In Grey Wolf Optimization (GWO), a population of wolves is employed to search for the optimal solution by mimicking the hunting behavior of a wolf pack. The Grey Wolf Optimizer and Particle Swarm Optimization are two popular metaheuristic optimization algorithms that are used to solve complex optimization problems. Both algorithms are based on the principles of swarm intelligence and are inspired by the behavior of animals in nature. However, there are some key differences between GWO and PSO: Optimization approach: GWO is a population-based optimization algorithm, where a group of grey wolves search for the optimal solution to a problem. PSO, on the other hand, is based on the behavior of a swarm of particles that explore the solution space. Solution representation: In GWO, the solution to a problem is represented as a position vector of the grey wolves, while, in PSO, it is represented as a position vector of the particles. Exploration vs. exploitation: GWO emphasizes the exploration of the search space in the early stages of the optimization process and gradually shifts towards exploitation in the later stages. PSO, on the other hand, maintains a balance between exploration and exploitation throughout the optimization process. Convergence rate: GWO has been shown to have a faster convergence rate than PSO for some optimization problems. However, PSO is known to be more effective at handling high-dimensional problems. Initialization: GWO requires an initial population of grey wolves to begin the optimization process, while PSO only requires an initial set of particles. In summary, both GWO and PSO are effective optimization algorithms that can be used to solve a wide range of problems. The choice between them depends on the specific problem being addressed and the performance requirements of the solution [17–20]. To achieve precise monitoring of the rotational speed with minimal ripple in the low-speed and high-torque operations of the five-phase asynchronous machine, the algorithm precisely tunes the parameters of the proportional–integral controller (PI) [19–24]. Although the advantages mentioned earlier are noteworthy, conventional PID controllers need a linear model, which cannot guarantee robustness against uncertainties because of the significant nonlinear dynamics of a five-phase induction motor (IM). In this paper, we propose a recently developed

optimization method based on the Grey Wolf algorithm (GWO). Under low speed and high torque conditions, this algorithm optimizes the gains of the PI controller to determine the optimal path of input commands that achieves the desired speed while minimizing ripple in the five-phase asynchronous machine [18].

This document is structured as follows: the first section covers the system description, and the second section addresses the modeling of the five-phase IM. The design methodology of the PI controller based on GWO is presented in the third section. The fourth section illustrates the results of the simulation and discusses the efficacy of the suggested control approach, and the final section provides a summary of the contributions of this work.

2. System Description

Achieving optimal control performance through the PID regulator requires a methodical approach to determine the appropriate values for the controller’s proportional, integral, and derivative gains. We propose a meta-heuristic optimization technique based on the hunting style of grey wolves as illustrated in Figure 1. The five-phase IM control structure employs three control loops: the motor angular speed control loop Ω that uses a PI controller, and two PI controllers for the stator flux ψ and the electromagnetic torque C_{em} . The primary electromagnetic torque is obtained directly from the external speed loop. The two regulators output the reference stator voltage (v_{qsref}, v_{dsref}) along the q and d axes. After the inverse Park transformation, we obtain the real stator voltage, which produces the real speed of the motor. The inputs of the estimator are the real currents along the (d,q) axis obtained after the Park transformation of the real currents. The estimation block determines the instantaneous amplitude of the stator flux ψ_s , the electromagnetic torque C_{em} , and the stator electric angles θ_s .

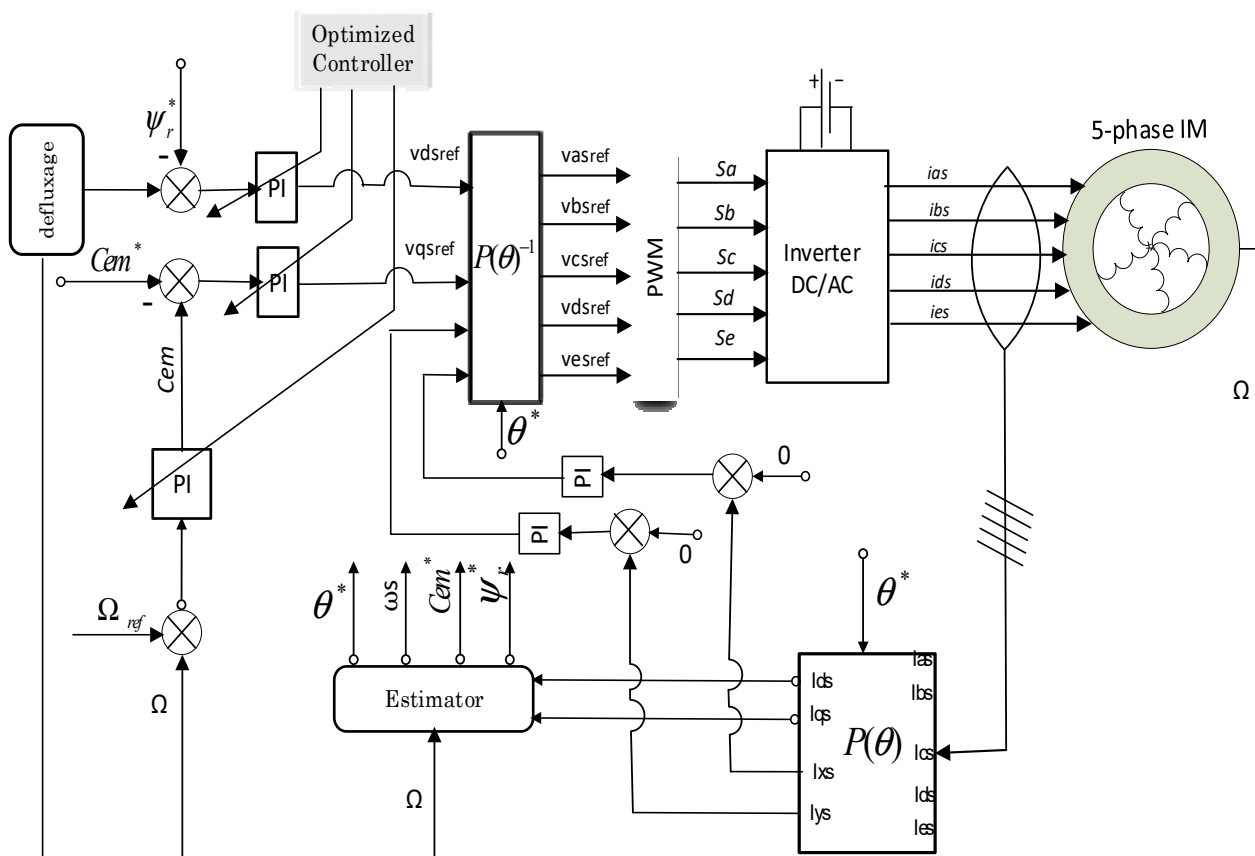


Figure 1. Schematic diagram of Optimized Controller applied on five-phase IM.

3. Modeling of Five Phase Induction Machine

The voltage model of the five phase induction motor is obtained as [5]:

$$\begin{cases} [V_s]_{1 \times 5} = [R_s]_{5 \times 5} [i_s]_{1 \times 5} + \frac{d}{dt} [\psi_s]_{1 \times 5} \\ [V_r]_{1 \times 5} = [R_r]_{5 \times 5} [i_r]_{1 \times 5} + \frac{d}{dt} [\psi_r]_{1 \times 5} \\ [\psi_s]_{1 \times 5} = [L_s]_{5 \times 5} [i_s]_{1 \times 5} + [M_{sr}]_{5 \times 5} [i_r]_{1 \times 5} \\ [\psi_r]_{1 \times 5} = [L_r]_{5 \times 5} [i_r]_{1 \times 5} + [M_{rs}]_{5 \times 5} [i_s]_{1 \times 5} \end{cases} \tag{1}$$

where $[R_s]_{5 \times 5}$ and $[R_r]_{5 \times 5}$ are the resistance matrices, $[L_s]_{5 \times 5}$ and $[L_r]_{5 \times 5}$ are the matrices of inductances, $[M_{sr}]_{5 \times 5}$ and $[M_{rs}]_{5 \times 5}$ are mutual inductance between stator and rotor matrices, $[V_s]_{1 \times 5}$ and $[V_r]_{1 \times 5}$ are the voltage matrices, $[i_s]_{1 \times 5}$ and $[i_r]_{1 \times 5}$ are the current matrices, $[\psi_s]_{1 \times 5}$ and $[\psi_r]_{1 \times 5}$ are flux linkages matrices, and the subscripts s and r refer to the stator and rotor windings, respectively. The mutual inductance matrices are given below.

$$[M_{sr}]_{5 \times 5} = [M_{rs}]_{5 \times 5}^t \tag{2}$$

All resistance matrices are symmetric as shown below, which is a direct result of the perfect symmetry of the machine:

$$\begin{cases} [R_s]_{5 \times 5} = R_s * [I]_{5 \times 5} \\ [R_r]_{5 \times 5} = R_r * [I]_{5 \times 5} \end{cases} \tag{3}$$

where R_s and R_r are stator resistance and stator inductance, respectively, and I is the matrix consisting of ones on the main diagonal and zeros elsewhere; this is referred to as the identity matrix.

The stator inductance matrix is obtained using:

$$[L_s]_{5 \times 5} = \begin{bmatrix} L_m + L_{fs} & L_m \cos\left(\frac{2\pi}{5}\right) & L_m \cos\left(\frac{4\pi}{5}\right) & L_m \cos\left(\frac{6\pi}{5}\right) & L_m \cos\left(\frac{8\pi}{5}\right) \\ L_m \cos\left(\frac{8\pi}{5}\right) & L_m + L_{fs} & L_m \cos\left(\frac{2\pi}{5}\right) & L_m \cos\left(\frac{4\pi}{5}\right) & L_m \cos\left(\frac{6\pi}{5}\right) \\ L_m \cos\left(\frac{6\pi}{5}\right) & L_m \cos\left(\frac{8\pi}{5}\right) & L_m + L_{fs} & L_m \cos\left(\frac{2\pi}{5}\right) & L_m \cos\left(\frac{4\pi}{5}\right) \\ L_m \cos\left(\frac{4\pi}{5}\right) & L_m \cos\left(\frac{6\pi}{5}\right) & L_m \cos\left(\frac{8\pi}{5}\right) & L_m + L_{fs} & L_m \cos\left(\frac{2\pi}{5}\right) \\ L_m \cos\left(\frac{2\pi}{5}\right) & L_m \cos\left(\frac{4\pi}{5}\right) & L_m \cos\left(\frac{6\pi}{5}\right) & L_m \cos\left(\frac{8\pi}{5}\right) & L_m + L_{fs} \end{bmatrix} \tag{4}$$

It is possible to express the mutual matrix inductance in the following manner:

$$[M_{sr}]_{5 \times 5} = L_m \begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{5}\right) & \cos\left(\theta - \frac{4\pi}{5}\right) & \cos\left(\theta - \frac{6\pi}{5}\right) & \cos\left(\theta - \frac{8\pi}{5}\right) \\ \cos\left(\theta - \frac{8\pi}{5}\right) & \cos(\theta) & \cos\left(\theta - \frac{2\pi}{5}\right) & \cos\left(\theta - \frac{4\pi}{5}\right) & \cos\left(\theta - \frac{6\pi}{5}\right) \\ \cos\left(\theta - \frac{6\pi}{5}\right) & \cos\left(\theta - \frac{8\pi}{5}\right) & \cos(\theta) & \cos\left(\theta - \frac{2\pi}{5}\right) & \cos\left(\theta - \frac{4\pi}{5}\right) \\ \cos\left(\theta - \frac{4\pi}{5}\right) & \cos\left(\theta - \frac{6\pi}{5}\right) & \cos\left(\theta - \frac{8\pi}{5}\right) & \cos(\theta) & \cos\left(\theta - \frac{2\pi}{5}\right) \\ \cos\left(\theta - \frac{2\pi}{5}\right) & \cos\left(\theta - \frac{4\pi}{5}\right) & \cos\left(\theta - \frac{6\pi}{5}\right) & \cos\left(\theta - \frac{8\pi}{5}\right) & \cos(\theta) \end{bmatrix} \tag{5}$$

where L_s , L_{fs} , and L_m are stator, stator leakage, and magnetizing inductances, respectively.

The transformation matrix $[P_n]$ for the five-phase induction motor is defined as follows [7]:

$$[P_n] = L_m \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 & 0 & 0 \\ -\sin(\theta) & \cos(\theta) & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

After applying Park's transformation matrix to the stator and rotor equations, the five-phase system can be decomposed into the dq coordinate system as well as into an additional xy coordinate system. The stator voltages of the system are written in the $d-q-x-y$ axis as follows [5–7]

$$\begin{cases} V_{ds} = r_s i_{ds} - \omega_s \psi_{qs} + \frac{d}{dt} \psi_{ds} \\ V_{qs} = r_s i_{qs} + \omega_s \psi_{ds} + \frac{d}{dt} \psi_{qs} \\ V_{xs} = r_s i_{xs} + \frac{d}{dt} \psi_{xs} \\ V_{ys} = r_s i_{ys} + \frac{d}{dt} \psi_{ys} \end{cases} \quad (7)$$

The flux equations, which become:

$$\begin{cases} \psi_{ds} = (L_m + L_{fs}) i_{ds} + L_m i_{dr} \\ \psi_{qs} = (L_m + L_{fs}) i_{qs} + L_m i_{qr} \\ \psi_{xs} = L_{fs} i_{xs} \\ \psi_{ys} = L_{fs} i_{ys} \end{cases} \quad (8)$$

The expression for the electromagnetic torque C_{em} has been utilized to finalize the model of the five-phase asynchronous machine.

$$C_{em} = \frac{P}{2} ([i_s]^t \frac{\partial [M_{sr}]}{\partial \theta} [i_r] + [i_r]^t \frac{\partial [M_{sr}]}{\partial \theta} [i_s]) \quad (9)$$

where P is the number of pole pairs.

The mechanical model is obtained using:

$$J \frac{d}{dt} \Omega_r = C_{em} - C_r - f \Omega_r \quad (10)$$

where C_{em} is the electromagnetics torques, C_r is the resistance torque, and J is the inertia.

4. Grey Wolf Optimizer

Grey Wolf Optimization (GWO) is a population-based metaheuristic optimization algorithm inspired by the hunting behavior of grey wolves in nature [12]. The algorithm was proposed by Seyedali Mirjalili, Shima Saremi, and Seyed Mohammad Mirjalili in 2014. This algorithm is applied to the machine as stated by Ali Djerioui in [18]. GWO mimics the social hierarchy and hunting behavior of grey wolves. A group of wolves, comprising of alpha, beta, delta, and omega wolves, operates in this algorithm. The alpha wolf is the pack leader, and the remaining wolves follow its lead. The beta, delta, and omega wolves have a lower pack rank and serve as subordinates to the alpha wolf. The GWO algorithm starts by randomly initializing the positions of the alpha, beta, delta, and omega wolves in the search space. Then, the fitness function is evaluated for each wolf, and the position of the alpha wolf is updated based on the best fitness value obtained so far. The positions of the other wolves are updated using a formula that depends on the positions of the alpha (α), beta (β), delta (δ), and omega (ω) wolves. The GWO algorithm has been applied successfully to a variety of optimization problems, including function optimization, feature selection, and classification. One of the advantages of GWO is its simplicity and ease of implementation. However, as with other metaheuristic algorithms, GWO cannot be guaranteed to find the global optimum for all optimization problems [12].

4.1. Objective Function

A function that is considered as a potential objective function is referred to as a candidate objective function. The objective function is a mathematical function that takes one or more input variables and produces a single output value. The optimization process involves searching for the input values that result in the highest or lowest possible output value, depending on the specific problem being addressed. It is a fundamental tool for

decision making and problem solving, and it enables researchers and practitioners to find the best solutions to complex problems.

To minimize the disparity between the estimated torque and flux and their references, traditional PI regulators are utilized; alternatively, the GWO algorithm is employed to obtain optimal gains. The GWO algorithm boasts several benefits, including fast convergence, robustness, and minimal adjustment of gains. The following candidate objective functions can be optimized using GWO:

$$H[n] = \begin{bmatrix} K_1 & K_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_3 & K_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & K_5 & K_6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & K_7 & K_8 \end{bmatrix} T[n] \quad (11)$$

where n is the current iteration, $K_1 \dots 8$ are the constant gains, and $H[n]$ is the objective function vector with two objective functions along the axis (d, q) .

The first objective function optimization can be applied to optimize the flux of a machine. The objective function in this case would be a function that quantifies the performance or efficiency of the machine in terms of the flux ψ_d with only one control input u_d .

The second objective function optimization can be used to optimize the speed $\Omega[k]$ and torque C_{em} of a machine by determining the input variables that result in the highest possible speed with only one control input u_q . The objective function in this case would be a function that quantifies the performance or efficiency of the machine in terms of its speed. The objective function could be the maximum torque (or speed) that the machine can achieve while maintaining a certain level of speed (or torque). The extent to which the function should be minimized is closely associated with the amplitude of the speed ripple:

$$T[k] = [T_d[k], T_q[k]]^T \quad (12)$$

with

$$T_d[k] = [\psi^{ref}[k] - \psi^*[k], \psi^{ref}[k-1] - \psi^*[k-1]] \quad (13)$$

$$T_q[k] = \begin{bmatrix} \Omega^{ref}[k] - \Omega[k], \Omega^{ref}[k-1] - \Omega[k-1], \\ C_{em}^{ref}[k] - C_{em}^*[k], C_{em}^{ref}[k-1] - C_{em}^*[k-1] \end{bmatrix} \quad (14)$$

4.2. Store the Best Particle

The GWO algorithm is based on a set of mathematical equations that are used to simulate the hunting behavior of grey wolves. Here is the general equation used in GWO [12]:

$$\begin{cases} H[k+1] = H_p[k] - P * D \\ D = |C * H_p[k] - H[k]| \end{cases} \quad (15)$$

here, k denotes the current iteration, H_p represents the position vector of the optimal solution, and H represents an objective function and also the position vector of a search agent. D indicates the distance between the i th wolf and the alpha wolf. P is a random number ranging from 0 to 2, which reflects the encircling behavior of the wolves. The values of P and C are calculated as follows:

$$\begin{cases} P = 2p * r_1 - p \\ C = 2p * r_2 \end{cases} \quad (16)$$

where r_1 and r_2 are random numbers between $[0, 1]$. The GWO algorithm also includes equations for updating the positions of the beta and delta wolves, as well as equations for adjusting the search space and controlling the exploration–exploitation trade-off. However, the general equation shown above is the key equation used in the encircling behavior of

the wolves in GWO. The hunt is guided by the leader (alpha), which is the best solution candidate; the beta and delta have better knowledge about the potential location of the prey.

$$\begin{cases} D_{x\alpha} = C_{x1}v_{x\alpha}(k) - H_x(k) \\ D_{x\beta} = C_{x2}v_{x\beta}(k) - H_x(k) \\ D_{x\delta} = C_{x3}v_{x\delta}(k) - H_x(k) \end{cases} \quad (17)$$

We use the optimal distance vector $[D_{x\alpha}, D_{x\beta}, D_{x\delta}]$ to formulate an intermediate input control variable $(v_{x\alpha}, v_{x\beta}, v_{x\delta})$ defined by the following equation:

$$\begin{cases} v_{x1}(k+1) = v_{x\alpha}(k) - P_{x1} * D_{x\alpha}(k) \\ v_{x2}(k+1) = v_{x\beta}(k) - P_{x2} * D_{x\beta}(k) \\ v_{x3}(k+1) = v_{x\delta}(k) - P_{x3} * D_{x\delta}(k) \end{cases} \quad (18)$$

where $C_{xi} = 2p_{xi}r_{xi}$, $P_{xi} = 2p_{xi}r_{xi} - p_{xi}$, $i \in (1, 2, 3)$, and $x = (d, q)$.

The following equation is used to compute the candidate dq -input control, which corresponds to the inverter voltages:

$$v_{xref}(k+1) = \sum_{i=1}^{i=n} \frac{v_{xi}(k+1)}{3} \quad (19)$$

Following the previous step, the search for the specific position for an ideal hunt is based on the control of the particular distance of each wolf. For this, the voltage reference is chosen as follows:

$$\begin{cases} u_{xref} = v_{xref}(k+1) \text{ if } (|P_{xi}|) < 1 \\ u_{xref} = v_{xref}(k) \text{ if } (|P_{xi}|) > 1 \end{cases} \quad (20)$$

where v_{xref} is the actual input control, $v_{xref}(k+1)$ is the future input control, and u_{xref} is the input control.

4.3. The Flowchart of the Algorithm

The Grey Wolf Optimization (GWO) algorithm is a metaheuristic optimization algorithm inspired by the social hunting behavior of grey wolves. The flowchart of the GWO algorithm is as follows:

- Initialize the grey wolves' population, which is a set of candidate solutions.
- Calculate the objective function of each grey wolf in the population.
- Assign the best, second best, and third best grey wolves as alpha, beta, and delta, respectively.
- Update the position of each grey wolf using Equation (14).
- Update the objective function of each grey wolf in the new population.
- Check if any of the updated grey wolves have a better fitness than alpha, beta, or delta. If so, update the position of alpha, beta, or delta accordingly.
- Repeat steps 4 to 6 until the stopping criteria are met, such as a maximum number of iterations or a desired level of convergence.

In order to compare the performance of the Grey Wolf Optimizer (GWO) with another optimization algorithm, we need to select a suitable algorithm for comparison. We propose Particle Swarm Optimization (PSO) as a potential candidate for this purpose. The flowchart for the basic PSO algorithm and GWO algorithm is shown in Figure 2.

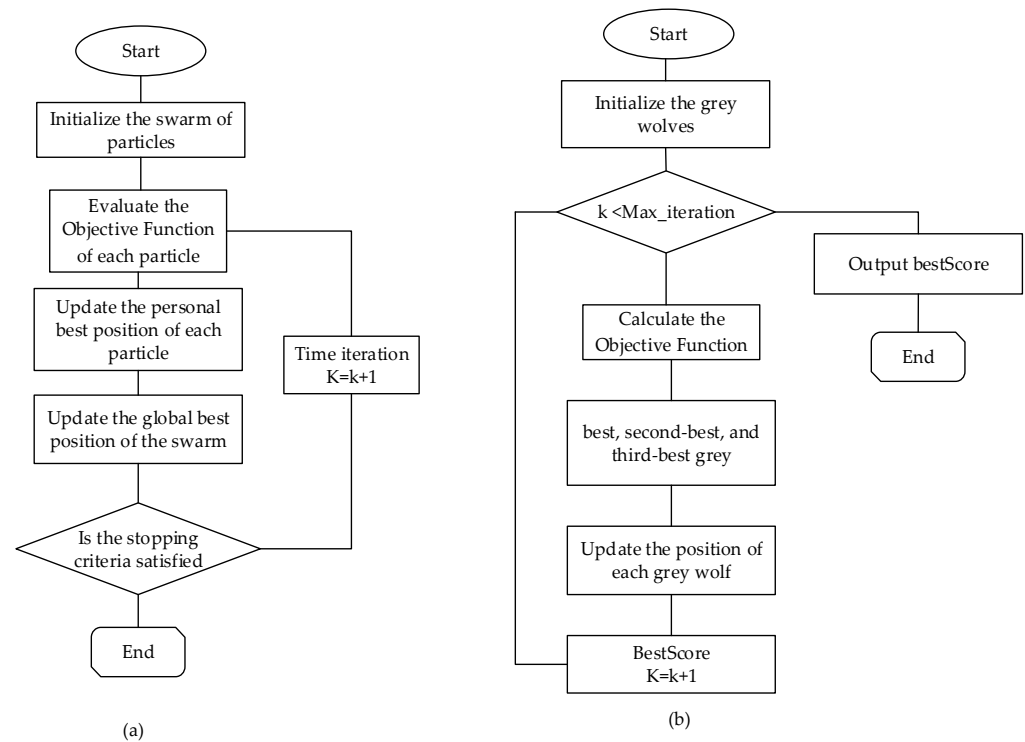


Figure 2. Flowchart for (a) basic PSO algorithm and (b) GWO algorithm.

5. PI Controller Parameters Using the GWO Algorithm

The following steps can be followed to optimize the PI controller parameters for a five-phase asynchronous motor using the GWO algorithm:

1. Define the problem: First, you need to clearly define the problem and identify the variables that need to be controlled. In this case, the objective is to optimize the PI controller parameters for controlling the speed of a five-phase asynchronous motor.
2. Choose the PI controller: A proportional–integral (PI) controller is commonly used in control systems to regulate a process variable. It is a type of feedback controller that uses the error between the desired setpoint and the measured process variable to adjust the control signal.
3. Determine the controller parameters: The PI controller has two parameters that need to be tuned: the proportional gain (K_p) and the integral gain (K_i). These parameters determine the response of the controller and can be adjusted to optimize the performance of the five-phase asynchronous motor.
4. Use the GWO algorithm for optimization: Grey Wolf Optimization (GWO) is a population-based optimization algorithm that is inspired by the hunting behavior of grey wolves. It is a metaheuristic optimization algorithm that can be used to find optimal solutions for complex problems. You can use GWO to optimize the PI controller parameters for the five-phase asynchronous motor.
5. Define the objective function: The fitness function is used to evaluate the performance of the PI controller. In this case, the fitness function should measure how well the controller is able to regulate the speed of the five-phase asynchronous motor.
6. Set up the GWO algorithm: The GWO algorithm requires several parameters to be defined, including the population size, maximum number of iterations, and search range for each parameter. The GWO algorithm can be set up to optimize the PI controller parameters for the five-phase asynchronous motor.
7. Run the optimization: Once the GWO algorithm is set up, it can be used to find the optimal values of K_p and K_i for the PI controller. The GWO algorithm will iterate through the population and adjust the values of K_p and K_i until the fitness function is optimized.

8. Evaluate the results.

Finally, the same can be performed to optimize the PI controller parameters for a five-phase asynchronous motor using the PSO algorithm.

6. Simulation

To validate the performance of the proposed scheme depicted in Figure 1, we present the simulations for a five-phase IM. The PI controller parameters are optimized using the GWO and PSO algorithm, whose rated values and nominal electrical and mechanical parameters are shown in Table 1. We obtained the values of PI controller gains using:

Table 1. Parameters of five-phase IM.

Parameters	Values	Parameters	Values
R_s	4.85 Ω	L_s	0.024 H
R_r	3.805 Ω	L_r	0.024 H
p	2	L_m	0.258 H
k	0.0005	j	0.085

Basic PSO algorithm: The controllers' gains are: $K_{pi} = 21.7200$, $K_{ii} = 47,700$, $K_{p\omega} = 0.025375$ and $K_{i\omega} = 3.0290$.

GWO algorithm: The controllers' gains are: $K_{pi} = 22.1200$, $K_{ii} = 43,400$, $K_{p\omega} = 0.032365$ and $K_{i\omega} = 2.0230$.

The results shown in Figure 3a–d demonstrate the favorable dynamic behavior of speed and torque achieved through the optimization of the PI controller parameters using the GWO algorithm. When the rotor speed and torque are in steady state, the stator resistance is changed from R_s to $R_s/4$ (stator faults). The PI control is a commonly used control strategy for induction motors, as it provides a simple and effective way to regulate the motor speed and torque. In this control strategy, the motor speed and torque are controlled by adjusting the magnitude and phase of the stator current. The speed reaches 314 V, given as Ω_{ref} at 0.05 s.

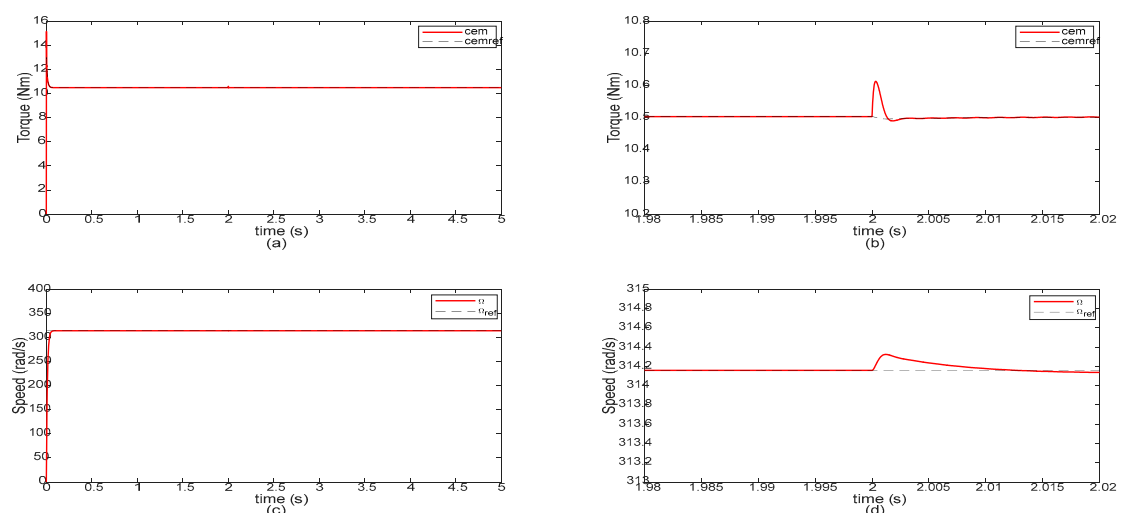


Figure 3. Torque (a,b) and speed (c,d) responses in the presence of stator faults with optimization of PI controller parameters using GWO algorithm.

In Figure 4a–d shows the current (i_{qs}) and relative tracking errors in the presence of stator faults in the optimization of the PI controller parameters using the GWO algorithm. The stator currents (i_{qs}) are shown in Figure 4a with the starting transients of the stator fault as it develops the relative tracking error, as displayed in Figure 4d. When the relative

tracking error is in steady state, the stator resistance is changed from R_s to $R_s/4$ to show the current harmonics.

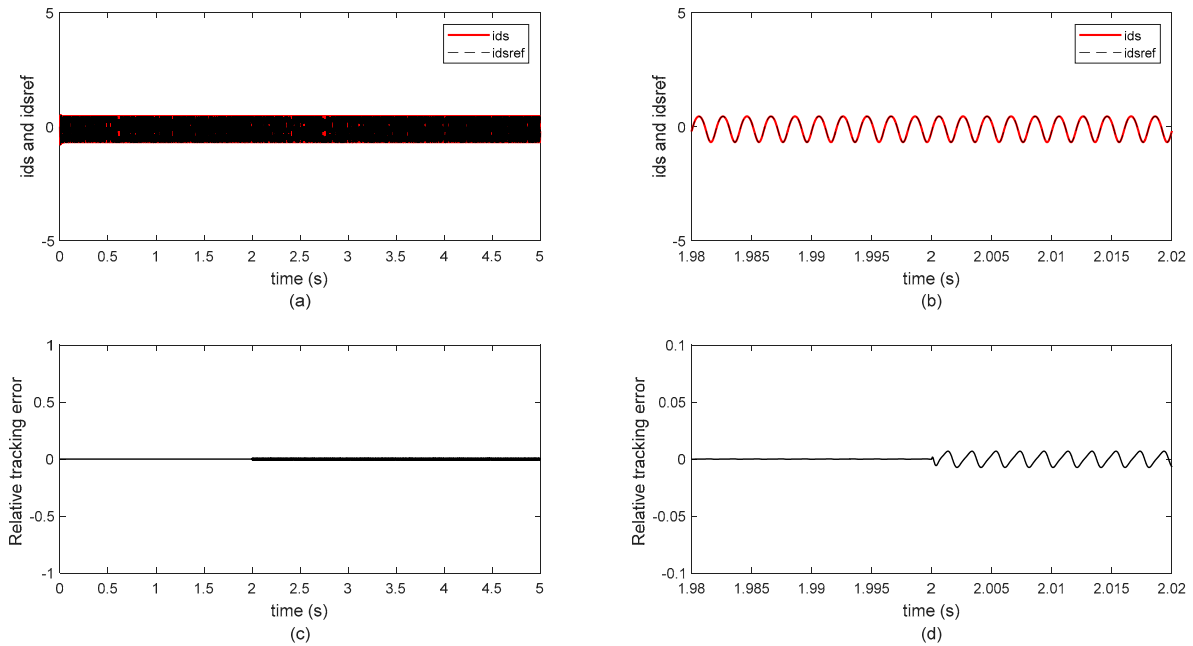


Figure 4. Current (i_{ds}) (a,b) and relative tracking error (c,d) in the presence of stator faults with the optimization of PI controller parameters using GWO algorithm.

Figures 5 and 6 show the current i_{qs} component and current i_{qp} component in the presence of stator faults with the optimization of PI controller parameters using the GWO algorithm, respectively. The currents (i_{qs}) are shown in Figure 4a with the starting transients of the stator fault as it develops the relative tracking error, as displayed in Figure 5d. When the relative tracking error is in steady state, the stator resistance is changed to show the current harmonics. This is also true for i_{qp} .

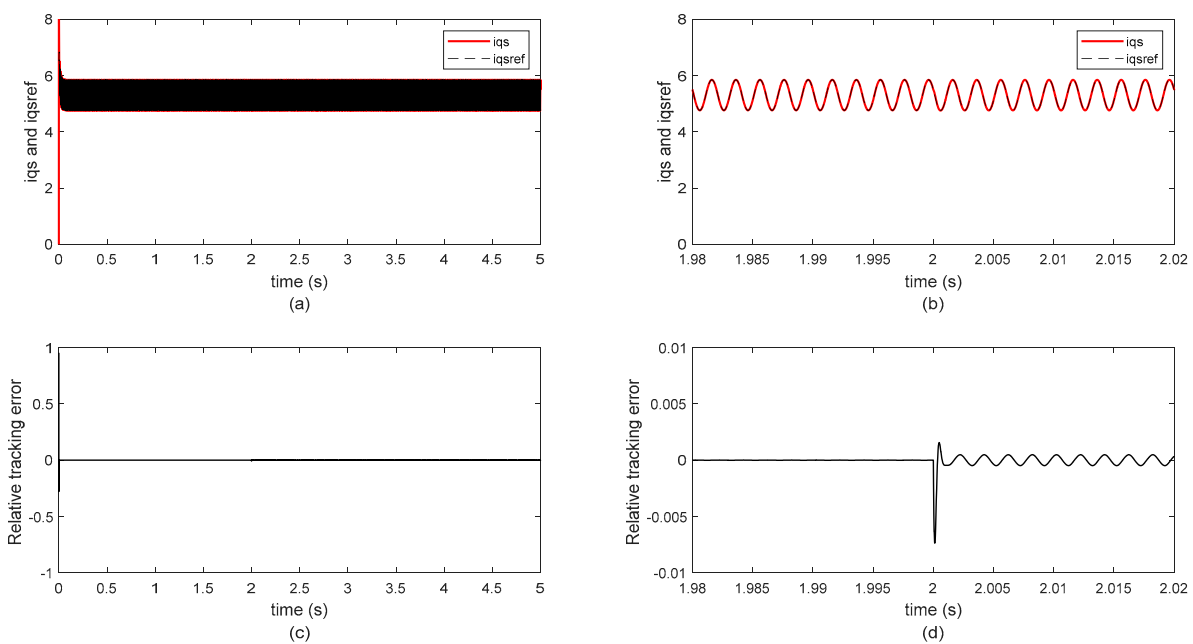


Figure 5. Current (i_{qs}) (a,b) and relative tracking error (c,d) in the presence of stator faults with the optimization of PI controller parameters using GWO algorithm.

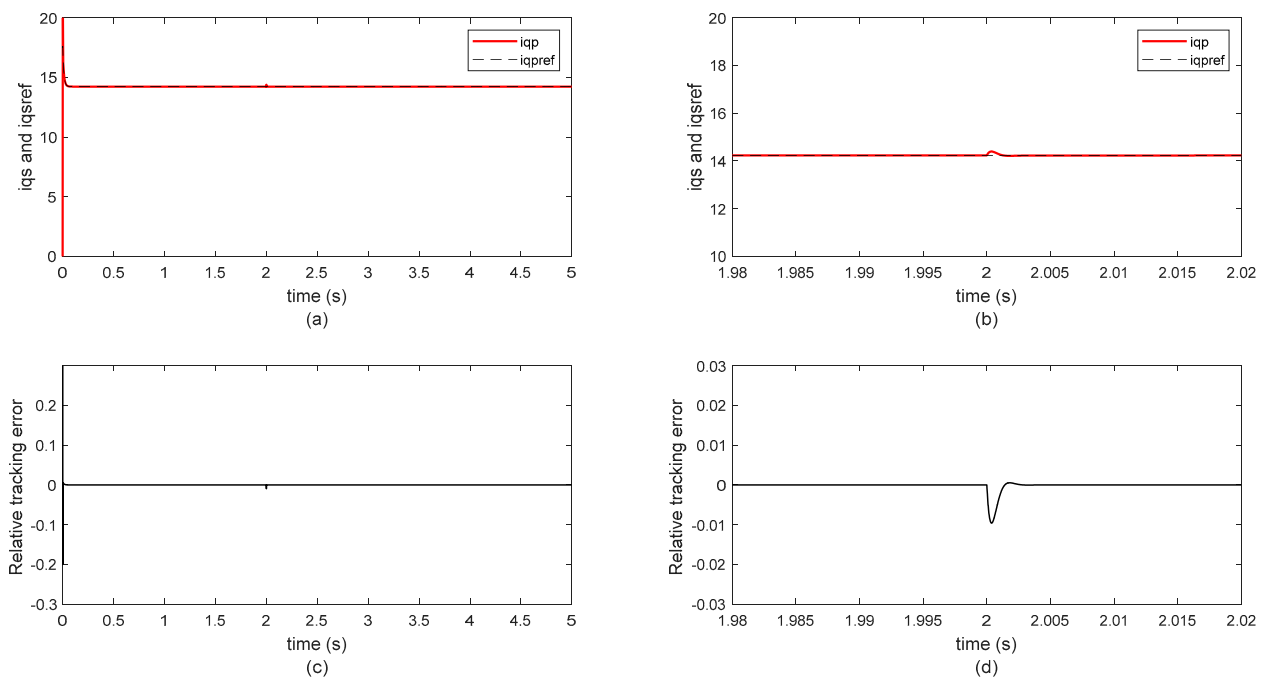


Figure 6. Current (i_{qp}) (a,b) and relative tracking error (c,d) in the presence of stator faults with the optimization of PI controller parameters using GWO algorithm.

Figure 7a–d show the electromagnetics torques’ components (c_{emp} and c_{ems}) with the optimization of PI controller parameters using the GWO algorithm while subjecting the stator’s resistance to a step changed from its nominal value to one-quarter of the nominal value (the speed reference is maintained at a constant level). The electromagnetic torque components for a five-phase machine can be expressed in the P–S reference frame. The total electromagnetic torque is the sum of the two torque components.

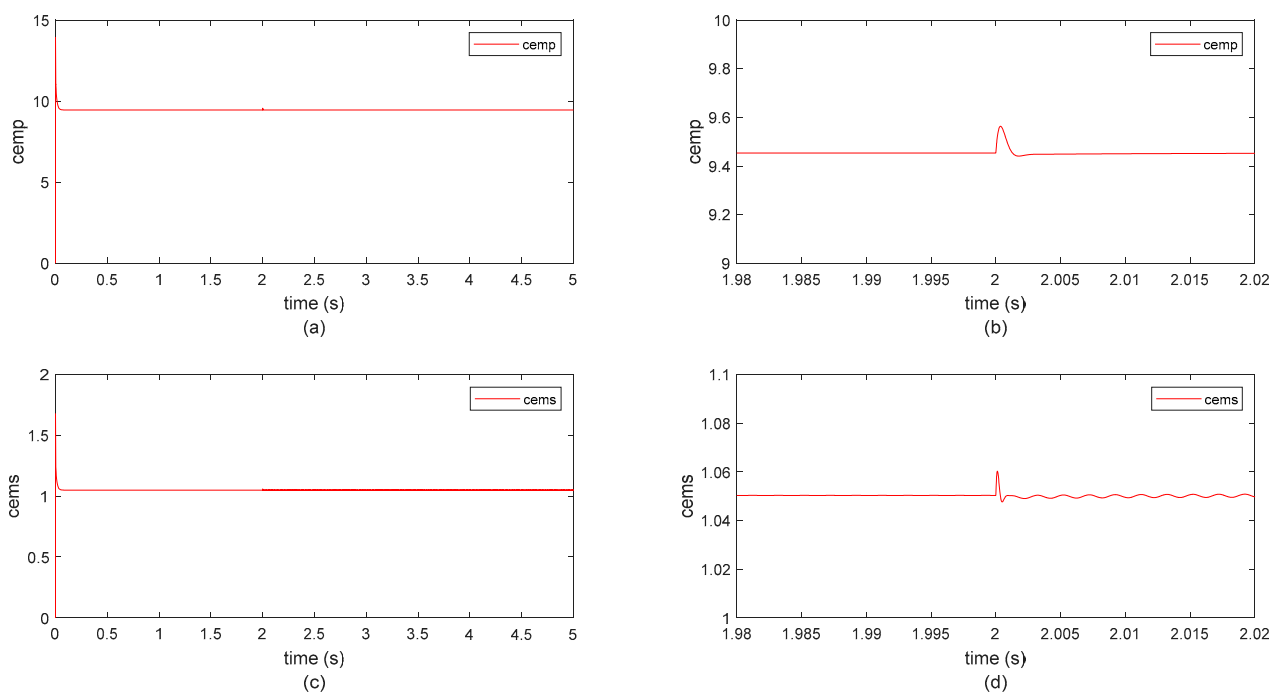


Figure 7. The electromagnetic torques’ components (c_{emp} (a,b) and c_{ems} (c,d)) in the presence of stator faults with the optimization of PI controller parameters using the GWO algorithm.

The proposed control using GWO has a turnaround time of 35.3 μs in Matlab, with 21.5 μs used for implementing the control. In comparison, using PSO, the turnaround time is 56.3 μs , with 39.5 μs used for implementing the control. Additionally, the proposed method has minimal impact on the treatment time and central processing unit (CPU) utilization.

Figure 8 shows that the rotor speed remains in a steady state. However, at 2 s, the stator resistance changes from R_s to $R_s/2$ due to stator faults. The implementation of a PI controller with GWO Control is more robust and results in a higher performance than a PI controller utilizing PSO.

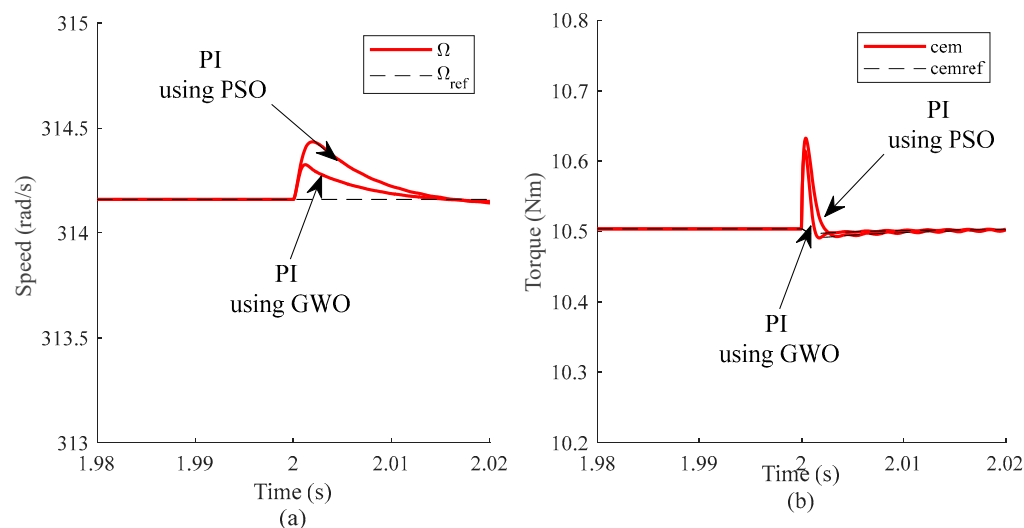


Figure 8. (a,b) Comparative study with a PI using GWO and PI using PSO.

From Table 2, the rotor speed is in a steady state, and the stator resistance is changed from R_s to $R_s/2$, $R_s/4$, and $R_s/10$ (stator faults) at 2 s. It can be asserted that utilizing the PI with GWO Control yields a high-performance control system that is more resilient than a conventional PI controller and PI using PSO (Setting time (T_{Se}) and Maximum Peak Overshoot (MPO)). To summarize the improvement in low-speed conditions, the rotor speed and electromagnetic torque of a five-phase induction motor with PI control depends on various factors such as the operating conditions, motor parameters, and control strategy used. When combined, the five-phase induction motor with PI control and GWO optimization can provide precise speed control and efficient operation of the motor, making it suitable for a wide range of applications.

Table 2. Comparative study with a classical PI controller, PI using GWO, and PI using PSO.

Stator Resistance	$R_s/2$			$R_s/4$			$R_s/10$		
	PI	PI using GWO	PI using PSO	PI	PI using GWO	PI using PSO	PI	PI using GWO	PI using PSO
Method	PI	PI using GWO	PI using PSO	PI	PI using GWO	PI using PSO	PI	PI using GWO	PI using PSO
Speed	PI	PI using GWO	PI using PSO	PI	PI using GWO	PI using PSO	PI	PI using GWO	PI using PSO
Setting time	0.2	0.04	0.09	0.24	0.07	0.101	0.32	0.078	0.103
Maximum Peak Overshoot	1%	0.02%	0.056%	2.5%	0.05%	0.076%	3.5%	0.066%	0.098%
Relatives' tracking error	6×10^{-3}	1×10^{-5}	4×10^{-4}	13×10^{-2}	2×10^{-5}	6.7×10^{-4}	5	2.5×10^{-5}	6.7×10^{-4}

7. Conclusions

In this paper, we evaluated the performance of the optimized PI controller by analyzing the output data and comparing it to the desired setpoint. If the speed of the five-phase asynchronous motor is regulated within an acceptable range and the ripples are minimized, then the PI controller parameters can be considered optimized for this application. If not,

the objective function may need to be adjusted. The proposed approach exploits the rapid optimization process of the GW technique under real-time working conditions to determine the optimal voltage vector that ensures the desired tracking speed at high torque and with less ripple. The mathematical principle of the proposed GWO controller is detailed, and the simulation results show that the GWO algorithm offers the best convergence towards optimal solutions with the best quality. Moreover, GWO shows a good balance between exploration and exploitation, which makes it a better algorithm to solve complex engineering optimization problems. To gain further insights into the effectiveness of the PI controller using GWO under real-time working conditions, it would be beneficial to conduct additional comparisons with advanced control methods such as intelligence-based algorithms. Additionally, the GWO algorithm could be extended to solve speed or torque-tracking tasks in other five-phase IM drives where low-order harmonics are undesired and traditional control algorithms are ineffective.

Author Contributions: Conceptualization, A.S.; Methodology, M.F., A.D. and M.L. Software, M.F.B.; Validation, H.M.; Formal analysis, A.S.; Investigation, H.M.; Resources, A.D., M.L. and A.H.; Data curation, F.B. and A.H.; Writing—original draft, S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Barrero, F.; Duran, M.J. Recent advances in the design modeling and control of multiphase machines. Part I. *IEEE Trans. Ind. Electron.* **2015**, *63*, 449–458. [[CrossRef](#)]
2. Duran, M.J.; Barrero, F. Recent advances in the design modeling and control of multiphase machines. Part II. *IEEE Trans. Ind. Electron.* **2016**, *63*, 459–468. [[CrossRef](#)]
3. Xu, H.; Toliyat, H.A.; Petersen, L.J. Resilient current control of five-phase induction motor under asymmetrical fault conditions. In Proceedings of the 17th Annual IEEE Applied Power Electronics Conference and Exposition (Cat. No. 02CH37335), Dallas, TX, USA, 10–14 March 2002; pp. 64–71.
4. Bermudez, M.; Gonzalez-Prieto, I.; Barrero, F.; Guzman, H.; Duran, M.J.; Kestelyn, X. Open-Phase Fault-Tolerant Direct Torque Control Technique for Five-Phase Induction Motor Drives. *IEEE Trans. Ind. Electron.* **2017**, *64*, 902–911. [[CrossRef](#)]
5. Iffouzar, K.; Taraft, S.; Aouzellag, H.; Ghedamsi, K.; Aouzellag, D. DRFOC of Polyphase Induction Motor based on fuzzy logic controller speed. In Proceedings of the 2015 4th International Conference on Electrical Engineering (ICEE), Boumerdes, Algeria, 13–15 December 2015; Volume 978, pp. 4673–6673.
6. Wu, X.; Song, W.; Xue, C. Low-Complexity Model Predictive Torque Control Method Without Weighting Factor for Five-Phase PMSM Based on Hysteresis Comparators. *IEEE J. Emerg. Sel. Top. Power Electron.* **2018**, *6*, 1650–1661. [[CrossRef](#)]
7. Bogado, B.; Prieto, I.G.; Rahal, M.R. Comparative study of predictive and resonant controllers in fault tolerant five-phase induction motor drives. *IEEE Trans. Ind. Electron.* **2016**, *63*, 606–617.
8. Ali, E. Speed Control of Induction Motor Supplied by Wind Turbine via Imperialist Competitive Algorithm. *Energy* **2015**, *89*, 593–600. [[CrossRef](#)]
9. Djerioui, A.; Houari, A.; Ait-Ahmed, M.; Benkhoris, M.F.; Chouder, A.; Machmoum, M. Grey Wolf based control for speed ripple reduction at low speed operation of PMSM drives. *ISA Trans.* **2018**, *74*, 111–119. [[CrossRef](#)]
10. Dogruer, T. Grey Wolf Optimizer-Based Optimal Controller Tuning Method for Unstable Cascade Processes with Time Delay. *Symmetry* **2023**, *15*, 54. [[CrossRef](#)]
11. Gong, R.; Li, X. A Short-Term Load Forecasting Model Based on Crisscross Grey Wolf Optimizer and Dual-Stage Attention Mechanism. *Energies* **2023**, *16*, 2878. [[CrossRef](#)]
12. Mirjalili, S.; Mohammad, S.; Lewis, A. Advances in Engineering Software Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [[CrossRef](#)]
13. Garg, H. A hybrid PSO-GA algorithm for constrained optimization problems. *Appl. Math. Comput.* **2016**, *274*, 1292–1305. [[CrossRef](#)]
14. Gacem, A.; Benattous, D. Hybrid GA-PSO for optimal placement of static VAR compensators in power system. *Int. J. Syst. Assur. Eng. Manag.* **2017**, *8*, 247–254. [[CrossRef](#)]
15. Pramanik, P.K.D.; Pal, S.; Mukhopadhyay, M.; Singh, S.P. Big Data classification: Techniques and tools. In *Applications of Big Data in Healthcare Theory and Practice*; Academic Press: Cambridge, MA, USA, 2021; pp. 1–43.
16. Oshaba, A.S.; Ali, E.S.; Abd Elazim, S.M. ACO Based Speed Control of SRM Fed by Photovoltaic System. *Int. J. Electr. Power Energy Syst.* **2015**, *67*, 529–536. [[CrossRef](#)]

17. Ullah, K.; Jiang, Q.; Geng, G.; Rahim, S.; Khan, R.A. Optimal Power Sharing in Microgrids Using the Artificial Bee Colony Algorithm. *Energies* **2022**, *15*, 1067. [[CrossRef](#)]
18. Djerioui, A.; Houari, A.; Machmoum, M.; Ghanes, M. Grey Wolf Optimizer-Based Predictive Torque Control for Electric Buses Applications. *Energies* **2020**, *13*, 5013. [[CrossRef](#)]
19. Deng, W.; Zhao, H.; Zou, L.; Li, G.; Yang, X.; Wu, D. A novel collaborative optimization algorithm in solving complex optimization problems. *Soft Comput.* **2017**, *21*, 4387–4398. [[CrossRef](#)]
20. Qais, M.; Hasanien, H.M.; Turkey, R.A.; Alghuwainem, S.; Tostado-Véliz, M.; Jurado, F. Circle Search Algorithm: A Geometry-Based Metaheuristic Optimization Algorithm. *Mathematics* **2022**, *10*, 1626. [[CrossRef](#)]
21. Sivamani, D.; Harikrishnan, R.; Essakiraj, R. Genetic algorithm based PI controller for DC-DC converter applied to Renewable energy applications. *Int. J. Pure Appl. Math.* **2018**, *118*, 1053–1071.
22. Ali, E.S. Optimization of power system stabilizers using BAT search algorithm. *Int. J. Electr. Power Energy Syst.* **2014**, *61*, 683–690. [[CrossRef](#)]
23. Bekakra, Y.; Ben Attous, D. Optimizing of IP speed controller using particle swarm optimization for FOC of an induction motor. *Int. J. Syst. Assur. Eng. Manag.* **2015**, *8*, 361–659. [[CrossRef](#)]
24. Bharti, O.P.; Saket, R.K.; Nagar, S.K. Controller design for doubly fed induction generator using particle swarm optimization technique. *Renew. Energy* **2017**, *114*, 1394–1406. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.