Abstract: The parametric variation of nonlinear systems remains a significant drawback of automatic system controllers. The Proportional–Integral (PI) and Proportional–Integral–Derivative (PID) are the most commonly used controllers in industrial control systems. However, with the evolution of these systems, such controllers have become insufficient to compete with the complexity of the systems. This problem can be solved with the help of artificial intelligence, and especially with the use of optimization algorithms, which allow for variable gains in PID controllers that adapt to parametric variation. This article presents an analytical and experimental study of the Direct Torque Control (DTC) of a Doubly-Fed Induction Motor. The speed adaptation of the DFIM is achieved using a PID controller, which is characterized by overshoots in the speed and ripples in the electromagnetic torque. The Genetic Algorithm (GA) within the DTC shows very good robustness in speed and torque by reducing torque ripples and suppressing overshoots. The simulation of the GA-DTC hybrid control in MATLAB/Simulink confirms the improvement offered by this strategy. The validation and implementation of this strategy on the dSPACE DS1104 board are in good agreement with the simulation results and theoretical analysis.

Keywords: Genetic Algorithm–Direct Torque Control (GA–DTC); dSPACE DS1104; control desk; Doubly-Fed Induction Motor (DFIM)

1. Introduction

Since the 1960s, with the application of the first power electronic components and the progressive rise of computing, the generation systems mentioned above have been progressively replaced by static converters. The latter, as they become more elaborate and gain more sophisticated control options, ensure progressive control of all electrical machines [1].

For high-power drives, there is a unique solution that employs an alternative machine operating in a rather special mode [2]. This is the DFIM with a wound rotor, offering several power supply configurations [3–5], with the best power supply configuration being the one where the machine is connected by two inverters, which allows the machine to be driven at overspeed, and ensures good performance at very low speeds for operation.
with and without a speed sensor [6]. Indeed, the establishment of any high-performance control for the DFIM must provide concrete solutions to the problems posed. To this effect, there appears to be a problem with the coupling existing between the magnetic behavior (flux) and the mechanical part (speed and torque) [7,8]. Then, there is also the issue of variables that cannot be measured directly, such as the rotor flux, where the use of physical sensors (Hall effect) does not present a perfect solution [9]. There is also the problem of the parametric variation—in particular, the rotor and stator time constants (due to the thermal effect). Indeed, the parameters of the machine can undergo significant variations [10]. On the other hand, due to the diversity of the types of loads used (i.e., different sizes, different natures, and different types), it is hard to obtain precise information on the load torque. In addition, when taking into account the problems raised, for this reason, the control must be simple in order to allow an easy, fast, and less expensive implementation [11,12]. To achieve performance similar to that of the DC motor, it is necessary to separate the flux control and the current control, generating electromagnetic torque [3–6]. In Germany, at the beginning of the 1970s, Blaschke and Hasse introduced a new vector control technique [13,14] named Flux-Oriented Control (FOC). With this strategy, and for the DFIM, the torque current component is fixed to an axis quadrature to the stator flux vector [15]; however, although this strategy is very robust for normal operations without variation of the parameters, this is not the case during operations within a system, because the machine is submitted to physical constraints (e.g., temperature and flux saturation) that disturb these parameters, affecting the robustness of the vector control because of its model, which is rich in machine parameters—especially the time constants. Moreover, the regulation loops are based on six Proportional–Integral (PI) regulators [16–19], and it is well known that Proportional–Integral–Derivative (PID) regulators generally have a wide reputation in the industrial field, as they are very robust for linear systems and known to have poor robustness for nonlinear systems. These facts prompted the authors of [20,21] to discover a new control mechanism called DTC. DTC relies mainly on simple assumptions, making it easy to implement. It features a fast torque response, is less sensitive to stator and rotor parameters—especially time constants—and the speed regulation loop is based on a single PI regulator. In the control of nonlinear systems or systems with variable parameters, the classical control laws such as PID regulators may be insufficient due to their sensitivity to parametric variations. Furthermore, they are not robust—especially in cases where precision and other dynamic performance are required. For this reason, robust control laws are utilized, which are insensitive to variations in the parameters and internal and external disturbances, owing to the use of artificial intelligence—especially optimization algorithms.

Plenty of studies are available in the literature, making it possible to make the PID regulator robust for nonlinear systems to optimize variable gains, which are estimated as optimal for each variation of the system. In [22–24], the authors tuned the PID regulator gains used for a DC motor system using the Grey Wolf Optimization (GWO) and Particle Swarm Optimization (PSO) algorithms. In [25], the authors developed a multi-objective non-dominated sorting GA for the tuning of a PID regulator used for a robotic manipulator. The study in [26] presented a GA technique to tune the PID regulator to overcome the constrained optimization dilemma in a servomotor system. In [27], the results revealed that a GA optimizer produces very interesting results in terms of system response time and overshoot when compared to Evolutionary Programming (EP) and PSO.

In order to advance scientific and technological investigation, this study is focused on improving the DTC by integrating the optimization algorithm. The PID gain regulators are optimized by a Genetic Algorithm; the new GA–PID structure adapts with the machine parameters’ variation, and the GA produces the optimal gains each time, which respond to the parametric changes of the machine as well as to the variation of the reference setpoints of speed and load torque. Figure 1 illustrates the configuration of the proposed GA–DTC; the details of the proposed control system are described in the body of the article.
The organizational structure of this article is presented as follows: Section 2 describes the DFIM model in stationary frame; Sections 3 and 4 are devoted to the description of the DTC and the proposed GA–DTC strategy; Section 5 is focused on the simulation of the GA–DTC using MATLAB/Simulink, and the elaboration of the results; Section 6 is devoted to the discussion and comparison of some studies on DFIMs and describing the real implementation of the adopted GA–DTC strategy; finally, Section 7 presents the conclusions and planned future work.

2. $\alpha, \beta$ Model of a DFIM

It is possible to derive the mathematical model from the three-phase model by going through the Concordia hypotheses while neglecting the slot effect of the DFIM, the core loss, the saturation, etc., in a stationary reference frame. As seen in Equations (1)–(5), the DFIM model may be stated as follows:

- Stator voltage components:
  \[
  \begin{align*}
  v_{sa} &= R_s i_{sa} + \frac{d\psi_{sa}}{dt} \\
  v_{sb} &= R_s i_{sb} + \frac{d\psi_{sb}}{dt}
  \end{align*}
  \]
  (1)

- Rotor voltage components:
  \[
  \begin{align*}
  v_{ra} &= R_r i_{ra} + \frac{d\psi_{ra}}{dt} + \omega_m \psi_{rb} \\
  v_{rb} &= R_r i_{rb} + \frac{d\psi_{rb}}{dt} - \omega_m \psi_{ra}
  \end{align*}
  \]
  (2)

- Stator flux components:
  \[
  \begin{align*}
  \psi_{sa} &= L_s i_{sa} + L_m i_{ra} \\
  \psi_{sb} &= L_s i_{sb} + L_m i_{rb}
  \end{align*}
  \]
  (3)

- Rotor flux components:
  \[
  \begin{align*}
  \psi_{ra} &= L_r i_{ra} + L_m i_{sa} \\
  \psi_{rb} &= L_r i_{rb} + L_m i_{sb}
  \end{align*}
  \]
  (4)

- Mechanical subsystem:
\[
\begin{aligned}
T_{em} &= p \left( \psi_{s \alpha} i_{s \beta} - \psi_{s \beta} i_{s \alpha} \right) \\
j \frac{d\Omega}{dt} + f \Omega &= T_{em} - T_r 
\end{aligned}
\]

(5)

3. Direct Torque Control

DTC entails directly manipulating the motor torque and fluxes via the control sequences applied to the voltage source inverter switches [20,21]. This choice is based on the utilization of hysteresis controllers whose function is to control the state of the system—specifically, the amplitude of the fluxes (stator and rotor) and the electromagnetic torque. It is a matter of keeping these quantities within specified error limits; the outputs of the hysteresis regulators, when combined with information about the flux position, determine the optimal voltage vectors. These vectors enable the inverter VSI to reach seven distinct phase-plane positions, which correspond to the inverter output voltage vector’s eight sequences [1,8].

3.1. Control Scheme

The fundamental technique of DTC of a DFIM has two parallel control channels: fluxes and torque. These are managed through three hysteresis regulators: two for stator and rotor fluxes, and the third for torque. In the first mode, the reference fluxes are direct inputs, while in the second, the reference torque (i.e., speed controller output) is the input. Estimation and regulation blocks calculate signals such as the developed torque, the flux magnitudes, and the vector sector of the fluxes. Hysteresis controllers provide digital outputs from estimated and reference values. The outputs of these controllers, combined with the main position, are used as switching table inputs to regulate the VSIs. The DFIM receives the output voltages from the inverters [1,5,8].

3.2. Block of Signal Estimation

The stator and rotor (\(\alpha, \beta\)) fluxes’ magnitudes can be evaluated as follows [5,8]:

\[
\begin{aligned}
\psi_{s \alpha\beta} &= \int \left( V_{s \alpha\beta} - R_s i_{s \alpha\beta} \right) dt \\
\psi_{r \alpha\beta} &= \int \left( V_{r \alpha\beta} - R_r i_{r \alpha\beta} \right) dt \\
|\psi_s| &= \sqrt{\psi_{s \alpha}^2 + \psi_{s \beta}^2} \\
|\psi_r| &= \sqrt{\psi_{r \alpha}^2 + \psi_{r \beta}^2}
\end{aligned}
\]

(6)\(\rightarrow\)(10)

The developed torque can be evaluated as follows [5]:

\[
T_{em} = p \cdot \left( \psi_{s \alpha} i_{s \beta} - \psi_{s \beta} i_{s \alpha} \right)
\]

(10)

The sectors of the fluxes are divided into six sectors, each covering an angle \(\pi/3\) and represented in a complex two-dimensional plane. The location of the space vector of fluxes in the complex plane can be represented as follows [5,8]:

\[
\begin{aligned}
\theta_s &= \tan^{-1} \frac{\psi_{s \beta}}{\psi_{s \alpha}} \\
\theta_r &= \tan^{-1} \frac{\psi_{r \beta}}{\psi_{r \alpha}}
\end{aligned}
\]

(11)\(\rightarrow\)(12)

Each sector angle is represented by a number, which is utilized to identify the voltage vector.
3.3. Hysteresis Regulators

The calculated torque and flux values are compared to their respective references. To maintain the torque and flux values within their permissible limits, appropriate commands are generated: \( \Delta \psi_s = \Delta \psi_r = 1 \) for an increase in flux, and \( \Delta \psi_s = \Delta \psi_r = -1 \) for a decrease \([5,6]\).

\[
\begin{align*}
\Delta \psi_s &= 1 \text{ if } |\psi_s| \leq |\psi^*_s| - |\text{hysteresis limit}| \\
\Delta \psi_b &= -1 \text{ if } |\psi_s| \geq |\psi^*_s| + |\text{hysteresis limit}| \\
\Delta \psi_r &= 1 \text{ if } |\psi_r| \leq |\psi^*_r| - |\text{hysteresis limit}| \\
\Delta \psi_r &= -1 \text{ if } |\psi_r| \geq |\psi^*_r| + |\text{hysteresis limit}| 
\end{align*}
\]

To enlarge the torque, \( \Delta T_e = 1 \); to reduce the torque, \( \Delta T_e = -1 \); to maintain it constant, \( \Delta T_e = 0 \). A three-level regulator is utilized for managing the torque; meanwhile, a two-level regulator is used for the flux \([6]\):

\[
\begin{align*}
\Delta T_e &= 1 \text{ if } T_e \leq T^*_e - |\text{hysteresis limit}| \\
\Delta T_e &= 1 \text{ if } T_e = T^*_e \\
\Delta T_e &= -1 \text{ if } T_e \leq T^*_e + |\text{hysteresis limit}| 
\end{align*}
\]

3.4. Inverters Switching Table

Table 1 illustrates the hysteresis comparator outputs with sector numbers. The hysteresis output voltage vectors and sector numbers are shown. The table shows the switch positions required to produce the necessary vector.

<table>
<thead>
<tr>
<th>( H_{\psi} ) or ( H_{\psi_r} )</th>
<th>( H_{\text{tem}} )</th>
<th>Sector ( S_i )</th>
<th>( S_1(S_a S_b S_c) )</th>
<th>( S_2(S_a S_b S_c) )</th>
<th>( S_3(S_a S_b S_c) )</th>
<th>( S_4(S_a S_b S_c) )</th>
<th>( S_5(S_a S_b S_c) )</th>
<th>( S_6(S_a S_b S_c) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>( v_2(110) )</td>
<td>( v_3(010) )</td>
<td>( v_4(011) )</td>
<td>( v_5(001) )</td>
<td>( v_6(101) )</td>
<td>( v_7(100) )</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>( v_7(111) )</td>
<td>( v_5(000) )</td>
<td>( v_6(010) )</td>
<td>( v_4(101) )</td>
<td>( v_3(100) )</td>
<td>( v_2(011) )</td>
<td>( v_1(001) )</td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td>( v_6(101) )</td>
<td>( v_7(110) )</td>
<td>( v_5(010) )</td>
<td>( v_4(011) )</td>
<td>( v_3(101) )</td>
<td>( v_2(010) )</td>
<td>( v_1(001) )</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>( v_5(010) )</td>
<td>( v_4(011) )</td>
<td>( v_6(001) )</td>
<td>( v_6(101) )</td>
<td>( v_7(110) )</td>
<td>( v_2(010) )</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>( v_5(000) )</td>
<td>( v_6(101) )</td>
<td>( v_7(111) )</td>
<td>( v_6(010) )</td>
<td>( v_7(010) )</td>
<td>( v_2(011) )</td>
<td>( v_1(001) )</td>
<td></td>
</tr>
</tbody>
</table>

A VSI drives the DFIM model by DTC. The inverter’s main function is to convert control signals into voltage signals for the motor. \( S_a, S_b, \) and \( S_c \) control the inverter states. Six switches are used for the three phases. Each phase contains two switches, which are in opposite positions \([5,6,8]\).

When \( S_a = 1 \), \( \overline{S_a} \) is off. When \( S_a = 0 \), \( \overline{S_a} \) is on. The other switches can be controlled in the same way. The VSI’s output voltage vector is as follows \([5]\):

\[
V_s = \frac{2V_{DC}}{3} \left( 2S_a + e^{i2\pi/3}S_b + e^{i4\pi/3}S_c \right)
\]

where \( V_{DC} \) is the DC voltage.

4. PID Optimization by GA

Genetic Algorithms (GAs) are used to solve commercial and research challenges by computationally simulating natural selection. Gas were created by J. Holland in the 1960s and 1970s \([28]\) to examine the impacts of mate selection, reproduction, mutation, and genetic information transfer. Constrained by their environment, various species (and individuals within species) strive to generate the fittest progeny. In the domain of genetic algorithms, the fittest potential solutions develop to produce even more optimal ones \([29,30]\).
PID controllers are known for their robustness in linear systems, but this robustness is limited in the nonlinear case due to parametric variation. The GA optimizes the KP, KI, and KD parameters to generate optimal values for the PID regulator. Figure 2 shows the simplified structure of the GA optimization method.

![Figure 2. Optimization of the PID controller parameters by the GA.](image-url)

GAs are a subclass of evolutionary algorithms that use evolutionary-biology-inspired techniques such as selection, crossover, and mutation [29,31]. Section 4.1 presents descriptions of the GA sequences of operations.

4.1. GA Parameters and Operators

GAs are based on three essential steps, where each step is characterized by its own configuration parameters; the steps and parameters are described by the following axes:

4.1.1. Chromosome Coding

A GA’s design begins with the binary coding of solutions in the form of chromosomes, which are a collection of genes or bits [32]. The main difference between GAs and other search optimization techniques is that GAs use coding methods. A GA generally uses binary coding [33,34]. It is impossible to say which coding method is optimal, as it depends on the situation. Real numbers are easier to work with, but only for specific situations. Before determining the coefficients of the PID controller, the performance limits of the PID controller must be specified. The approach created defines the lower limit at zero PID coefficients. A GA does not work alone on an encoding problem. Therefore, the incorrect encoding format affects the GA’s results. In this study, the PID parameters are encoded as a single chromosome, where each parameter is seen as a gene [32].

4.1.2. Assignment of the First Population

The GA performs a search from many places to find the optimal resolution. To ensure that these initial values are established correctly, it is crucial to assign the first values for the points. Population one is usually produced randomly to represent the search space as a whole. Some research implies that the first population is produced using heuristics [33].

4.1.3. Learning of the Regulator Gains by the GA

The GA parameters greatly influence the results of the GA. There have been numerous experiments to establish the optimal control settings [31]. The Pittsburgh learning approach was employed in this project [32]. This strategy is successful in genetic fuzzy systems. Using this technology, the PID controller coefficients are stored on a single chromosome. In Figure 3, the resulting chromosomal structure is shown.
limit at zero PID coefficients. A GA does not work alone on an encoding problem. Therefore, the incorrect encoding format affects the GA’s results. In this study, the PID parameters are encoded as a single chromosome, where each parameter is seen as a gene [32].

4.1.2. Assignment of the First Population

The GA performs a search from many places to find the optimal resolution. To ensure that these initial values are established correctly, it is crucial to assign the first values for the points. Population one is usually produced randomly to represent the search space as a whole. Some research implies that the first population is produced using heuristics [33].

4.1.3. Learning of the Regulator Gains by the GA

The GA parameters greatly influence the results of the GA. There have been numerous experiments to establish the optimal control settings [31]. The Pittsburgh learning approach was employed in this project [32]. This strategy is successful in genetic fuzzy systems. Using this technology, the PID controller coefficients are stored on a single chromosome. In Figure 3, the resulting chromosomal structure is shown.

**Figure 3.** Chromosome structure created for a PID controller.

4.1.4. Fitness

The choice of cost functions used to assess each gene’s eligibility is essential in the GA process. In some publications [35,36], performance indices are used as cost functions. In [35], the authors compare three cost functions separately and in a weighted combination: Integral Time Absolute Error (ITAE), Integral Absolute Error (IAE), and Integral Square Error (ISE). Because the latter is more effective than the other cost functions in terms of performance, the ISE is used in this work to increase the performance of the GA [37].

\[
ISE = \int_{0}^{t} e(t)^2 dt \tag{21}
\]

The operating principle of the GA seeks to maximize or minimize a function; in our case study, the fitness function was maximized, i.e., the minimization of the ISE that presents the error corrected by the PID corrector; these parameters were generated by the GA to make it converge to zero as much as possible [35,37].

\[
Fitness Value = \frac{1}{ISE} \tag{22}
\]

4.1.5. Initialization of Populations

In a population, individuals each solve an issue. Other studies have tried to develop population size parameters. Sometimes, small solution sizes result in inferior solutions. However, obtaining a large size takes far too long. Robertson used up to 8000 people to classify difficulties [33,38]. Grefenstette said between 10 and 160 people is optimal, while Odeyato said between 100 and 400 [39]. The GA is unable to identify the intended result for small values of the population, yet the population as a whole takes a long time to compute. As a consequence, the best experiment was run and \( n = 20 \) was chosen as the population size.

4.1.6. Selection Operator

In general, the selection operator refers to how the chromosomes are selected. It is possible to consider the fitness function to be a means of measuring how fit a particular chromosome is, and the fitter the chromosome, the greater the likelihood that it will be selected to spread its genes.
Three critical selection techniques exist: Stochastic Sampling with Replacement Selection (SSRS) (or Roulette Wheel), Universal Stochastic Sampling (USS), and Tournament Selection (TS) [40]. The selection procedure utilized in this method was TS, as it provided the desired results after performing several experimental tests on the global system.

4.1.7. Crossover Operator

To accomplish this, the crossover operator selects a locus at random and exchanges subsequences to the left and right of that locus across two chromosomes (i.e., parent genes) that were selected during the selection process, to provide the children who inherit combined features with the good genes of one parent. Using binary representation, for example, two strings 11111111 and 00000000 might be crossed over at the sixth locus in one another, resulting in the production of two new offspring—11111000 and 00000111, respectively—using binary representation [41]. To avoid crossbreeding, the chance of crossbreeding must be chosen between 0.6 and 0.99 [40]. The probability value of the crossover operator was chosen to be 0.8 in this case.

4.1.8. Mutation Operator

The mutation operator modifies the bits or digits at a specific locus on a chromosome in a random manner; however, this occurs with a very small probability in most cases with a probability $P_m$. For example, upon crossover, the 11111000 child string could be changed at locus two to become the 10111000 child string. Mutation adds new knowledge to the genetic pool, and prevents the population from converging too quickly to a local optimum in a given environment; [0.001, 0.01] [42] is the range in which this number must fall. The probability $P_m$ value is set to 0.001, since this is the most likely.

Considering the evolutionary rules of the GA method, a flow chart summarizing the sequence of implementation of the algorithm is shown in Figure 4. The execution steps are also addressed as follows (Algorithm 1):

**Algorithm 1. Genetic Algorithm**

**Begin**

Step 1. Initialize the algorithm parameters (It, Pop, Pc, gamma, mu, sigma, nVar, VarMax, VarMin).

Step 2. Randomly generate the regulator gains.

Step 3. Apply the DTC control.

Step 4. Evaluate the fitness function instantaneously.

Step 5. Apply binary coding.

Step 6. Move to the selection operation.

Step 7. Move to the crossover operation.

Step 8. Move to the mutation operation.

Step 9. Apply binary decoding.

Step 10. Update the optimal individual and repeat Step 3 until the maximum number of iterations is fulfilled.

Step 11. Save the optimal solutions.

**End**
Figure 4. GA flowchart.

5. Simulation Procedure and Interpretation
5.1. Simulation Procedure

Figure 5 depicts the suggested technique for DTC utilizing a PID controller based on GA employing ISE applied to a DFIM.
Step 5. Apply binary coding.

Step 6. Move to the selection operation.

Step 7. Move to the crossover operation.

Step 8. Move to the mutation operation.

Step 9. Apply binary decoding.

Step 10. Update the optimal individual and repeat Step 3 until the maximum number of iterations is fulfilled.

Step 11. Save the optimal solutions.

End

5. Simulation Procedure and Interpretation

5.1. Simulation Procedure

Figure 5 depicts the suggested technique for DTC utilizing a PID controller based on GA employing ISE applied to a DFIM.

Figure 5. GA–DTC schematic applied to a DFIM.

A simulation of the theoretically given approach was performed using MATLAB/Simulink. The PID controller parameter values optimized by the GA fell within certain variation ranges, as shown in Table 2. Tables A1 and A2 in the Appendix A show the overall system configuration. The system is also subject to speed and torque reference conditions. The machine used for the two strategies is 1.5 kW, and they are subjected to the conditions of simulation to arrive at a valid comparison. The procedures for implementing the reference instructions are organized as follows:

Table 2. The PID parameter bands.

<table>
<thead>
<tr>
<th>PID Parameters</th>
<th>K_P</th>
<th>K_I</th>
<th>K_D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum Value</td>
<td>100</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Minimum Value</td>
<td>0</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

The overall system consisting of the controls—whether DTC or GA–DTC coupled with a DFIM—is subject to speed and torque references:
1. A speed step of 157 rad/s is applied from 0.7 s to 2.35 s, and then the setpoint begins to decrease in the form of an affine function of negative slope down to $-157$ rad/s, which presents the opposite direction of rotation. This speed remains constant up to 4.65 s, and then the motor is stopped.

2. During the speed variation, a load torque is applied to the system, which starts with a nominal load torque step of 10 Nm at 1.1–2.1 s, and then the motor remains idle until the instant 3.05–4.25 s, when a negative load torque is applied, which follows the direction of machine rotation.

At the beginning of the system’s operation, the following parameters are set:

- Sampling frequency: $f_s = 10$ kHz.
- Hysteresis bands widths: $\Delta T_{em} = \pm 0.01$ Nm, $\Delta \Psi_s = \Delta \Psi_r = \pm 0.001$ Nm.

After several attempts to generate the gains from the PID controllers, the algorithm ultimately converged to its optimum, giving optimal gains that gave the best results for machine performance under DTC. Controller gains from the DTC were generated by the standard synthesis method using the Bode and Nyquist diagrams. Finally, Table 3 illustrates the gains synthesized by the conventional method for the DTC, as well as the optimal optimizer gains by the GA algorithm.

<table>
<thead>
<tr>
<th>Controller Parameters</th>
<th>Classic DTC</th>
<th>GA–DTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_P$</td>
<td>0.776</td>
<td>72.8895</td>
</tr>
<tr>
<td>$K_I$</td>
<td>28.74</td>
<td>0.0729</td>
</tr>
<tr>
<td>$K_D$</td>
<td>0</td>
<td>0.5262</td>
</tr>
</tbody>
</table>

Figures 6–18 illustrate the motor performance of both approaches.

![Figure 6](image-url)

**Figure 6.** Speed responses (a–d) of both strategies.
Figure 6. Speed responses (a–d) of both strategies. The speed of the DTC control is represented in red, showing reference tracking throughout the reference setpoint, but representing a speed overshoot value of 60.14 rad/s, as shown in Figure 6a, and the response rejoins its reference, marking a response time of 61.1 ms and an undershoot of 9.826 ms due to the application of the nominal load at the instant of 1.25 s. The response speed is directly linked to its reference, marking a rejection time of 84 ms, as shown in Figure 6b. This response represents an almost-complete reference tracking during the decrease in speed and reversal of the speed rotation, as shown in Figure 6c, but one can notice overshoot and undershoot peaks due to instantaneous speed variation and the sudden application of load torque. According to Figure 6, the response speed of the GA–DTC (in blue) shows absolute tracking of the reference without overshoot and without static error, regardless of the variation in the reference and the load torque, marking several improvements in the DTC speed performance, such as response time, overshoot, reject time, and undershoot, with remarkable rates of 18.66%, 100%, 81.07%, and 51.86%, respectively.

Figure 7. Torque dynamics of the DTC and GA–DTC. The torque dynamics of the two controls are illustrated in Figure 7. The two responses show a follow-up of the reference setpoint of the fully complete load torque throughout the reference. The responses represent a starting torque of up to 15 Nm, which is normal because of the current demands at the start of rotation and at the sudden instants of increasing speed, whether loaded or unloaded. One can see that the undershoot in the torque presented by the DTC is corrected by the proposed GA–DTC, and is eliminated during the instants 0.7 s and 4.65 s, so one can observe that the DTC presents torque ripples of 2.445 Nm and 2.05 Nm for the GA–DTC, marking an improvement of 16.16%, knowing that the torque ripples are the essential factor determining the life of the DFIM. For example, if in our case the motor is predicted to stop working after 10 years with the use of DTC, the lifetime will expand at a rate of 16.16%, i.e., the machine will stop working after 11.616 years, with an addition of 1.616 years as a result of using the GA–DTC.

Figure 8. Stator currents (a–c) of the DTC.

Figure 9. Rotor currents (a–c) of the DTC.
Figure 10. Stator (a) and rotor (b) current THD of the DTC.

Figure 11. Stator Vsa (a) and rotor Vra (b) voltages of the DTC.

Figure 12. Stator currents (a–c) of the GA–DTC.
Energies 2022, 15, x FOR PEER REVIEW 15 of 27

Figure 13. Rotor currents (a–c) of the GA–DTC.

Figure 14. Stator (a) and rotor (b) current THD of the GA–DTC.

Figure 15. Stator Vsa (a) and rotor Vra (b) voltages of the GA–DTC.
The RTI allows the adjustment of inputs and outputs through the use of a graphical interface. In addition to the RTI library, which is included in the software section, the Toolbox RTI library includes blocks specific to the “Real-Time Interface” (RTI) Toolbox and its components. These blocks enable the design of real-time applications using software that has been expressly placed in the Toolbox.

The computer, a part of which contains emulating software (MATLAB), enables the designers to operate the inverter IGBTs in conjunction with the TMS320F240 slave DSP. On the dSPACE board, which transfers the signals from simulation environment to the physical system of the board, including the RTI interface, takes care of the application and administration, and generates 0–5V TTL (PWM) logic signals. Suitable for research and development purposes, serves as the basis of the control system. To link the “R&D DS1104” board is equipped with the Real-Time Workbench (RTW) tool, implemented on the computer, and it is housed inside a physical system as an electrical device.

The DTC enhanced by using GA allows the DFIM to be controlled more reliably and to validate the necessary improvements. The GA–DTC method adjusts to the machine’s parametric fluctuation, it is supposed to be compared to their respective values in Table 4.

The “R&D DS1104” board is equipped with the Real-Time Workbench (RTW) tool, which is used to compile and transform the source code into an .sdf file. Using a compiler, a program file is converted to an .sdf file is loaded and executed. This is also used to develop designer.

Practical Validation

With respect to the previously stated assumptions, as well as the various performance measures, the proposed DTC based on GA provides a satisfactory response in terms of speed, attaining the target speed with minimal response time and without overshoot. Furthermore, in the case of a rapid change in load, the speed can be managed slowly.

Finally, based on the results of the entire study, the suggested method outperforms the approach suggested in this paper.

Implementing a realistic solution allows us to evaluate the performance of the different strategies applied to the DFIM.

**Figure 16.** Stator (a) and rotor (b) fluxes of both strategies.

**Figure 17.** Histogram of various responses time of different strategies applied to the DFIM.

**Figure 18.** Histogram of various torque ripples of different strategies applied to the DFIM.
Figure 6a–c depict the motor’s speed performances. The speed of the DTC control is represented in red, showing reference tracking throughout the reference setpoint, but representing a speed overshoot value of 60.14 rad/s, as shown in Figure 6a, and the response rejoins its reference, marking a response time of 61.1 ms and an undershoot of 9.826 ms due to the application of the nominal load at the instant of 1.25 s. The response speed is directly linked to its reference, marking a rejection time of 84 ms, as shown in Figure 6b. This response represents an almost-complete reference tracking during the decrease in speed and reversal of the speed rotation, as shown in Figure 6c, but one can notice overshoot and undershoot peaks due to instantaneous speed variation and the sudden application of load torque. According to Figure 6, the response speed of the GA–DTC (in blue) shows absolute tracking of the reference without overshoot and without static error, regardless of the variation in the reference and the load torque, marking several improvements in the DTC speed performance, such as response time, overshoot, reject time, and undershoot, with remarkable rates of 18.66%, 100%, 81.07%, and 51.86%, respectively.

The torque dynamics of the two controls are illustrated in Figure 7. The two responses show a follow-up of the reference setpoint of the fully complete load torque throughout the reference. The responses represent a starting torque of up to 15 Nm, which is normal because of the current demands at the start of rotation and at the sudden instants of increasing speed, whether loaded or unloaded. One can see that the undershoot in the torque presented by the DTC is corrected by the proposed GA–DTC, and is eliminated during the instants 0.7 s and 4.65 s, so one can observe that the DTC presents torque ripples of 2.445 Nm and 2.05 Nm for the GA–DTC, marking an improvement of 16.16%, knowing that the torque ripples are the essential factor determining the life of the DFIM. For example if in our case the motor is predicted to stop working after 10 years with the use of DTC, the lifetime will expand at a rate of 16.16%, i.e., the machine will stop working after 11.616 years, with an addition of 1.616 years as a result of using the GA–DTC.

Figures 8, 9 and 10a,b illustrate the stator and rotor currents and their THDs, which determine the harmonic rate in the case of the use of the DTC. It should be noted that the currents are of sinusoidal form, which are harmonics of the THD values of 10.47% and 7.57% for the stator and rotor currents, respectively, generating ripples in the torque of 2.445 Nm. It is known that the harmonics have an influence on the torque ripples. If the THD rate is high, we will see more ripples in the torque, decreasing the motor’s lifetime.

Figures 12, 13 and 14a,b illustrate the stator and rotor currents and their THDs, which determine the harmonic rate in the case of the proposed GA–DTC. It should be noted that the currents are of a sinusoidal form, which is less harmonious compared to the DTC, presenting with THD values of 4.17% and 3.95% for the stator and rotor currents, respectively, generating ripples in the torque of 2.05 Nm. This marks improvements in the THDs, reaching 60.17% and 47.82%, respectively, increasing the motor’s lifetime.

Figures 11a,b and 15a,b show the stator and rotor voltages delivered by the voltage inverters; from the first view it can be noted that the voltages are of the two-level type. Figures 11 and 12 of the DTC are rich in pulses that present undesired harmonics, which occur as torque ripples, increasing the mechanical vibrations of the machine. From Figure 15a,b of the GA–DTC control, it can be seen that the voltages have fewer pulses, allowing voltages close to the ideal voltages, and reducing the torque ripples.

Figure 16 shows the stator and rotor flux characteristics of conventional DTC and GA–DTC, which enhance the flux amplitudes by 29.7% and 24.32%, respectively, when compared to their respective values in Table 4.

When opposed to a classic DTC, the DFIM with GA–DTC reaches its reference torque value in a far shorter amount of time. Furthermore, the torque is carefully regulated in cases where the load fluctuates significantly. In terms of response time, as seen in Figure 7, the proposed DTC based on GA provides a satisfactory response in terms of speed, attaining the target speed with minimal response time and without overshoot. Furthermore, in the case of a rapid change in load, the speed can be managed slowly.
Finally, based on the results of the entire study, the suggested method outperforms the classic DTC. Motor control with low torque ripples may be achieved by using the approach suggested in this paper.

With respect to the previously stated assumptions, as well as the various performances presented in Table 4, the GA–DTC proposed in this article achieves its stated goal by showing its resilience and robustness to speed, fluxes, and torque ripples, as well as currents and THDs.

Table 4. Performance measures of classic DTC and GA–DTC.

<table>
<thead>
<tr>
<th>Performances</th>
<th>Characteristics</th>
<th>DTC</th>
<th>GA–DTC</th>
<th>Improvements (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Omega$</td>
<td>Response Time (ms)</td>
<td>61.1</td>
<td>49.7</td>
<td>18.66</td>
</tr>
<tr>
<td></td>
<td>Overshoot (rad/s)</td>
<td>60.14</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>Rejection Time (ms)</td>
<td>84</td>
<td>15.9</td>
<td>81.07</td>
</tr>
<tr>
<td>$T$</td>
<td>Undershoot (rad/s)</td>
<td>9.826</td>
<td>4.73</td>
<td>51.86</td>
</tr>
<tr>
<td>$\psi_s$</td>
<td>Ripples (Nm)</td>
<td>2.445</td>
<td>2.05</td>
<td>16.16</td>
</tr>
<tr>
<td>$\psi_r$</td>
<td>Ripples (wb)</td>
<td>0.06123</td>
<td>0.04304</td>
<td>29.71</td>
</tr>
<tr>
<td>$i_{sa}$</td>
<td>THD (%)</td>
<td>10.47</td>
<td>4.17</td>
<td>60.17</td>
</tr>
<tr>
<td>$i_{ra}$</td>
<td>THD (%)</td>
<td>7.57</td>
<td>3.95</td>
<td>47.82</td>
</tr>
</tbody>
</table>

The DTC enhanced by using GA allows the DFIM to be controlled more reliably under a wider range of operating conditions than previously possible. Due to the fact that the GA–DTC method adjusts to the machine’s parametric fluctuation, it is supposed to be an optimal solution for variable-speed drive applications, as shown above.

5.2. Discussion and Comparison

A number of techniques for controlling the motor at different speeds and torques have been described in the technical literature for the DFIM. As an example, consider the FOC control developed by [6], which is very sensitive to variations in the parameters of the motor. The torque ripples of this strategy are 2.5 Nm, and the response time is 0.56 s. In [43,44], the authors employed the SMC—also known as the chattering phenomena—which limits the robustness of this technique in terms of performance. This method has a torque ripple of 2.5 Nm and a response time of 0.19 s. Because of the drawbacks of the controls outlined above, DTC emerges as the most suited option, and artificial intelligence approaches are among the strategies used to strengthen its resilience. As shown in Table 5, the suggested GA–DTC method enables us to achieve the highest feasible level of performance. The performance of several techniques used in the DFIM is shown in the following table:

Table 5. Comparison between our proposed approach and some control strategies published recently.

<table>
<thead>
<tr>
<th>Publication</th>
<th>Approaches</th>
<th>Response Time (s)</th>
<th>Torque Ripples (Nm)</th>
<th>Robustness</th>
</tr>
</thead>
<tbody>
<tr>
<td>[6]</td>
<td>Field-Oriented</td>
<td>0.56</td>
<td>2.5</td>
<td>Not Robust</td>
</tr>
<tr>
<td></td>
<td>Sliding Mode</td>
<td>0.19</td>
<td>2.4</td>
<td>Not Robust</td>
</tr>
<tr>
<td>[40]</td>
<td>Classical DTC</td>
<td>0.0507</td>
<td>2.445</td>
<td>Robust</td>
</tr>
<tr>
<td>[41]</td>
<td>FL–DTC</td>
<td>0.28</td>
<td>1.14</td>
<td>Robust</td>
</tr>
<tr>
<td>Studied in This Work</td>
<td>GA–DTC</td>
<td>0.0497</td>
<td>2.05</td>
<td>Robust</td>
</tr>
</tbody>
</table>

The histograms shown in Figures 17 and 18 illustrate the response time of speed and the amplitudes of the torque ripples produced by each approach. On the basis of Figure 17, one can conclude that the fastest controller is the proposed GA–DTC, while on the basis of Figure 18, one can conclude that the proposed strategy is the one with the least amount of ripples.
6. Practical Validation and Interpretation

6.1. Practical Validation

Implementing a realistic solution allows us to evaluate the performance of the GA–DTC nonlinear robust control, and to validate the necessary improvements. The DS1104 “R&D” control board, manufactured by dSPACE and specifically designed for research and development purposes, serves as the basis of the control system. To link between the simulation and the DS1104 board, the Real-Time Interface (RTI) library is installed on the computer, and it is housed inside a physical system as an electrical device on the dSPACE board, which transfers the signals from simulation environment to the real system. The physical system of the board, including the RTI interface, takes care of the application and administration, and generates 0–5V TTL (PWM) logic signals. Suitable isolation of the TTL-CMOS control boards with a supply voltage of 0/15 V is required to operate the inverter IGBTs in conjunction with the TMS320F240 slave DSP. On the computer, a part contains emulating software (MATLAB) that enables the designers to design real-time applications using blocks that have been expressly placed in the Toolbox RTI library, which is included in the software section. The Toolbox RTI library includes blocks specific to the “Real-Time Interface” (RTI) Toolbox and its components. In addition, the RTI allows the adjustment of inputs and outputs through the use of a graphical user interface.

The “R&D DS1104” board is equipped with the Real-Time Workbench (RTW) tool, which is used to compile and transform the source code into an .sdf file. Using a computer loaded with the “ControlDesk” software, the source code that was created and converted to an .sdf file is loaded and executed. This is also used to develop designer interfaces and perform real-time controls, thus archiving programs in fully compatible “Matlab-Simulink” files and allowing observation and real-time monitoring of the system and the evolution of calculated and measured data, among other applications. When creating graphical designer interface prototypes, the following are the main procedures to follow:

1. First of all, the control system is designed using the Simulink modeling program.
2. It is necessary to simulate the system in order to generate several control results, in order to know the control validity.
3. Generation of the .sdf file using the RTI interface.
4. When the global model is run in real time, the DS1104 R&D board is used through the ControlDesk environment with a processor (MPC8240) operating at a clock frequency equal to 250 MHz. An image of the experimental setup and a diagram showing the link between the DS1104 R&D Board and the DFIM are shown in Figures 19 and 20, respectively. In order to perform the experimental test, the control board and the RTW tool are used together.

Figure 19. Image of the test bench.
6.2. Results and Interpretation

It is possible to validate the objectives previously set by the experiment on dSPACE DS1104 by testing the system’s behavior under various working conditions after applying a variable-speed signal in both directions to the proposed GA–DTC and DFIM, either at zero charge or at full charge, using 10 Nm and −10 Nm as the parameters.

Figure 21a,b, as well as Figures 22–24, show that the results of the simulations and experiments are identical. When the speed rapidly changes in both directions, the transition from 0 to 157 rad/s and from −157 rad/s to 0 occurs quickly and without exceeding the limit (Figure 21a). When applying a variable torque, temporary changes in the reference values cause the appearance of peaks in the speed profiles, and the stator and rotor currents have a sinusoidal form that varies with the charge (Figures 22 and 23). As shown in the figures, there is excellent agreement between the results of the simulation and the results obtained on the ground during these applications.

Figure 21. Speed (a) and torque (b) of the GA–DTC.

Figure 20. Global structure of the test bench.
6.2. Results and Interpretation

It is possible to validate the objectives previously set by the experiment on dSPACE DS1104 by testing the system’s behavior under various working conditions after applying a variable-speed signal in both directions to the proposed GA–DTC and DFIM, either at zero charge or at full charge, using 10Nm and −10Nm as the parameters.

Figure 21a,b, as well as Figures 22–24, show that the results of the simulations and experiments are identical. When the speed rapidly changes in both directions, the transition from 0 to 157 rad/s and from −157 rad/s to 0 occurs quickly and without exceeding the limit (Figure 21a). When applying a variable torque, temporary changes in the reference values cause the appearance of peaks in the speed profiles, and the stator and rotor currents have a sinusoidal form that varies with the charge (Figures 22 and 23). As shown in the figures, there is excellent agreement between the results of the simulation and the results obtained on the ground during these applications.

**Figure 21.** Speed (a) and torque (b) of the GA–DTC.

**Figure 22.** (a) Stator currents of the GA–DTC; (b,c) zooms of currents from 0.9 s to 1.2 s, and from 1.9 s to 2.2 s, respectively.

**Figure 23.** Stator currents of the GA–DTC; (a–c) zooms of currents from 0.9 s to 1.2 s and from 1.9 s to 2.2 s, respectively.
Figure 23. Stator currents of the GA–DTC; (a) zooms of currents from 0.9 s to 1.2 s and from 1.9 s to 2.2 s, respectively.

Figure 24. Isofluxes under GA–DTC: (a) stator and (b) rotor.

Figure 21a,b show the speed response at variable setpoints (from 0 rad/s to 157 rad/s, and from −157 rad/s to 0 rad/s), as shown in the previous figure. Generally speaking, the response times are reasonable (in simulation and in practice). Furthermore, it is important to point out that the speed is set to the specified values, with good dynamic behavior and no detectable exceedances. In addition, there are no static errors, despite the fact that there is a slight increase in torque ripples in practice as a result of the measurement noise and the machine’s parameters, as illustrated in Figure 24.

Figure 25a,b show the control pulses delivered by the switching tables of the GA–DTC, which are delivered to control the inverters of the DFIM at variable frequencies. It can be seen that the pulses are in the form of a variable-frequency pulse train, to control the inverters and the DFIM according to the variation of the speed and load.

The circular flux curves of the stator and rotor are shown in Figure 26, which demonstrates that the fluxes are maintained within a specified range determined by the hysteresis comparator’s range. The results obtained are consistent with those obtained as a result of simulating the situation.

Figure 26a,b show the stator and rotor voltages delivered by the voltage inverters, respectively; at first glance, the voltages appear to be similar to the voltages found by simulation, indicating that they are harmonized with fewer harmonics than the classical DTC.
fact that there is a slight increase in torque ripples in practice as a result of the measurement noise and the machine's parameters, as illustrated in Figure 24.

Figure 25a,b show the control pulses delivered by the switching tables of the GA–DTC, which are delivered to control the inverters of the DFIM at variable frequencies. It can be seen that the pulses are in the form of a variable-frequency pulse train, to control the inverters and the DFIM according to the variation of the speed and load.

The circular flux curves of the stator and rotor are shown in Figure 26, which demonstrates that the fluxes are maintained within a specified range determined by the hysteresis comparator's range. The results obtained are consistent with those obtained as a result of simulating the situation.

Figure 25. Inverter control pulses of the stator (a) and rotor (b) of the GA–DTC.

Figure 26. Stator Vsa (a) and rotor Vra (b) voltages of the GA–DTC.

Finally, we can summarize the advantages and disadvantages of using optimization algorithms such as the GA used in this study, as follows:

**Advantages:**
- Allows an optimal reference monitoring profile.
- Adaptation with a sudden change in front of a system's internal and external disturbances.
- Rapid optimization of the gains of a regulator such as the PID
- Can be used as an estimator of parameters sensitive to physical variations.

**Disadvantages:**
- Convergence of solutions towards local solutions.
- Estimates of the parameters of the genetic algorithm, such as the population size and the number of iterations; requires more than two weeks to have a reduced execution time with optimal gains.

7. Conclusions

A Genetic-Algorithm-based PID optimization algorithm for DTC applied to a DFIM was developed. This configuration was simulated in MATLAB/Simulink and practically validated using a dSPACE DS1104testboard. This study shows an enhancement of the dynamics of classic DTC via optimizing and updating the PID coefficients KP, KI, and KD in each interval to adapt to the system nonlinearity. The proposed GA–DTC control demonstrated significant improvements in speed overshoot and rejection time, fluxes and torque ripples, and current THD. The following items highlight the improvements made to the DFIM’s performance:

- Reducing the speed overshoot, with and without load torque.
- Reducing the disturbance rejection time by 81.07%.
- Minimizing the ripples of the stator and rotor fluxes, as well as torque ripples, with improvements of 29.71%, 24.32%, and 16.16%, respectively.
- Acceptable enhancements in the current THDs, by 60.17% and 47.82%, respectively.
- The practical validation findings obtained using ControlDesk confirmed the simulated results obtained via MATLAB/Simulink.

A significant improvement was shown in the proposed technique in the form of faster response, fewer ripples, and higher precision and robustness, confirming the achievement of the targets outlined earlier in this paper.

With a view to advancing the technological research conducted in the laboratory, our research team has chosen to undertake the following tasks as part of its future work:

- Validation of ANN–DTC for DFIMs using a dSPACE board.
- Elaboration of the review of the techniques applied to DFIMs.

The practical validation findings obtained using ControlDesk confirmed the simulated results obtained via MATLAB/Simulink.

A significant improvement was shown in the proposed technique in the form of faster response, fewer ripples, and higher precision and robustness, confirming the achievement of the targets outlined earlier in this paper.

With a view to advancing the technological research conducted in the laboratory, our research team has chosen to undertake the following tasks as part of its future work:

- Validation of ANN–DTC for DFIMs using a dSPACE board.
- Elaboration of the review of the techniques applied to DFIMs.

Funding: This study was funded equally by the Ministry of Higher Education in Egypt, Hassan First University in Morocco, and Lac Hong University in Vietnam.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in the paper are available upon request from the corresponding author (M.A.M.).

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

Appendix A

Table A1. Specifications of the DFIM.

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Values (Unit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_n$</td>
<td>1.5 Kw</td>
</tr>
<tr>
<td>$V_s$</td>
<td>400 v</td>
</tr>
<tr>
<td>$V_r$</td>
<td>130 v</td>
</tr>
<tr>
<td>$P$</td>
<td>2</td>
</tr>
<tr>
<td>$f$</td>
<td>50 Hz</td>
</tr>
<tr>
<td>$R_s$</td>
<td>1.75 Ω</td>
</tr>
<tr>
<td>$R_r$</td>
<td>1.68 Ω</td>
</tr>
<tr>
<td>$L_s$</td>
<td>0.295 H</td>
</tr>
<tr>
<td>$L_r$</td>
<td>0.104 H</td>
</tr>
<tr>
<td>$M$</td>
<td>0.165 H</td>
</tr>
<tr>
<td>$f$</td>
<td>0.0027 kg·m²/s</td>
</tr>
<tr>
<td>$I$</td>
<td>0.01 kg·m²</td>
</tr>
</tbody>
</table>

Table A2. GA parameters.

<table>
<thead>
<tr>
<th>Descriptions</th>
<th>Types/Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Population Size</td>
<td>20</td>
</tr>
<tr>
<td>Maximum Iterations</td>
<td>50</td>
</tr>
<tr>
<td>Crossover Probability</td>
<td>0.9</td>
</tr>
<tr>
<td>Mutation Probability</td>
<td>0.001</td>
</tr>
<tr>
<td>Beta</td>
<td>1</td>
</tr>
<tr>
<td>Sigma</td>
<td>0.1</td>
</tr>
<tr>
<td>Gamma</td>
<td>0.1</td>
</tr>
<tr>
<td>Coding</td>
<td>Binary</td>
</tr>
<tr>
<td>Selection</td>
<td>Uniform</td>
</tr>
<tr>
<td>Crossover</td>
<td>Roulette Wheel Selection</td>
</tr>
<tr>
<td>Mutation</td>
<td>Uniform</td>
</tr>
</tbody>
</table>
Table A3. Nomenclature.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{a\alpha}, V_{a\beta}, V_{r\alpha},$ and $V_{r\beta}$</td>
<td>(α, β) Components of Stator and Rotor Voltages</td>
</tr>
<tr>
<td>$U_{d\alpha}$ and $U_{d\beta}$</td>
<td>Stator and Rotor Direct Voltages</td>
</tr>
<tr>
<td>$I_{a\alpha}, I_{a\beta}, I_{r\alpha},$ and $I_{r\beta}$</td>
<td>(α, β) Components of Stator and Rotor Currents</td>
</tr>
<tr>
<td>$\Psi_{a\alpha}, \Psi_{a\beta}, \Psi_{r\alpha},$ and $\Psi_{r\beta}$</td>
<td>(α, β) Components of Stator and Rotor Fluxes</td>
</tr>
<tr>
<td>$R_s, R_r$</td>
<td>Resistances of Stator and Rotor Windings</td>
</tr>
<tr>
<td>$L_s, L_r$</td>
<td>Inductances of Stator and Rotor Windings</td>
</tr>
<tr>
<td>$L_m$</td>
<td>Magnetizing Inductance</td>
</tr>
<tr>
<td>$p$</td>
<td>Pole Pairs</td>
</tr>
<tr>
<td>$\omega_r$</td>
<td>Angular Rotor Speed</td>
</tr>
<tr>
<td>$\omega_s$</td>
<td>Angular Stator Frequency</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>Rotational Speed</td>
</tr>
<tr>
<td>$T_{em}$</td>
<td>Developed Torque</td>
</tr>
<tr>
<td>$T_r$</td>
<td>Encountered Torque</td>
</tr>
<tr>
<td>$f$</td>
<td>Friction</td>
</tr>
<tr>
<td>$J$</td>
<td>Motor Inertia</td>
</tr>
</tbody>
</table>

Table A4. Abbreviations table.

<table>
<thead>
<tr>
<th>Abbreviations</th>
<th>Wording</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFIM</td>
<td>Doubly-Fed Induction Motor</td>
</tr>
<tr>
<td>IM</td>
<td>Induction Machine</td>
</tr>
<tr>
<td>DC</td>
<td>Direct Current</td>
</tr>
<tr>
<td>THD</td>
<td>Total Harmonic Distortion</td>
</tr>
<tr>
<td>DTC</td>
<td>Direct Torque Control</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>GA-DTC</td>
<td>Genetic Algorithm–Direct Torque Control</td>
</tr>
<tr>
<td>CLFT</td>
<td>Closed-Loop Function Transfer</td>
</tr>
<tr>
<td>PID</td>
<td>Proportional–Integral–Derivative</td>
</tr>
<tr>
<td>DTFC</td>
<td>Direct Torque Fuzzy Control</td>
</tr>
<tr>
<td>DTNC</td>
<td>Direct Torque Neural Control</td>
</tr>
<tr>
<td>DTNFC</td>
<td>Direct Neural Fuzzy Torque Control</td>
</tr>
<tr>
<td>ANFIS</td>
<td>Adaptive Neuro-Fuzzy Inference System</td>
</tr>
<tr>
<td>RTI</td>
<td>Real-Time Interface</td>
</tr>
<tr>
<td>R&amp;D</td>
<td>Research and Development</td>
</tr>
<tr>
<td>RTW</td>
<td>Real-Time Workbench</td>
</tr>
<tr>
<td>CP</td>
<td>Control Panel</td>
</tr>
<tr>
<td>PWM</td>
<td>Pulse-Width Modulation</td>
</tr>
</tbody>
</table>

References