Direct torque control of dual star induction motor using a fuzzy-PSO hybrid approach

Ghoulemallah Boukhalfa, Sebti Belkacem, Abdesselem Chikhi and Said Benaggoune
Department of Electrical Engineering, Faculty of Technology, University of Batna2, Fesdis, Algeria

Abstract
This paper presents the particle swarm optimization (PSO) algorithm in conjunction with the fuzzy logic method in order to achieve an optimized tuning of a proportional integral derivative controller (PID) in the DTC control loops of dual star induction motor (DSIM). The fuzzy controller is insensitive to parametric variations, however, with the PSO-based optimization approach we obtain a judicious choice of the gains to make the system more robust. According to Matlab simulation, the results demonstrate that the hybrid DTC of DSIM improves the speed loop response, ensures the system stability, reduces the steady state error and enhances the rising time. Moreover, with this controller, the disturbances do not affect the motor performances.

Keywords Dual stator induction motor (DSIM), Direct torque control (DTC), Speed control, particle swarm optimization (PSO), Fuzzy logic control (FLC)

Paper type Review Article

Nomenclature
\[ P \] Number of pole pairs
\[ J \] The moment of inertia
\[ f_r \] The friction coefficient
\[ T_{em} \] The electromagnetic torque
\[ T_r \] The load torque
\[ \Omega_r \] is the mechanical rotation speed of the rotor
\[ R_{s1}, R_{s2} \] Stators resistances
\[ R_r \] Rotor resistance
\[ L_{s1}, L_{s2} \] Stators inductances
\[ L_r \] Rotor Inductance
\[ L_m \] Mutual inductance
\[ n \] number of particles in the group
\[ d \] dimension index
\[ t \] pointer of iterations (generations)

The publisher wishes to inform readers that the article “Direct torque control of dual star induction motor using a fuzzy-PSO hybrid approach” was originally published by the previous publisher of Applied Computing and Informatics and the pagination of this article has been subsequently changed. There has been no change to the content of the article. This change was necessary for the journal to transition from the previous publisher to the new one. The publisher sincerely apologises for any inconvenience caused. To access and cite this article, please use Boukhalfa, G., Belkacem, S., Chikhi, A., Benaggoune, S. (2020), “Direct torque control of dual star induction motor using a fuzzy-PSO hybrid approach”, New England Journal of Entrepreneurship. Vol. 18 No. 1/2, pp. 76-91. The original publication date for this paper was 06/09/2018.
1. Introduction

In high-power applications, the dual star induction motor has largely replaced the induction machines whose roles were predominant in the industry [1–5]. The dual star induction motor is constituted of two windings with phases shifted from one another by an angle of 30 electrical degrees. These windings are usually powered by a six-phase inverter fed by variable speed drives. The main advantages of the DSIM are: [6] their higher torque density, reduced harmonic content of the DC link current and the reliability of this machine which allows a functioning with one or several phases of defective motor. However the control of DSIM is considered to be complicated because the difficulty of obtaining the decoupling of the torque and the flux. To overcome these difficulties, high-performance algorithms have been developed [7–14].

To satisfy the performance of an electromechanical system drive, the generally used strategy consists in controlling the speed by a PID controller to cancel the static error and reduce the response time. This speed is often characterized by an overshoot at startup and depends on the parameters of the machine. In order to overcome these complications, several methods have been developed to adjust the PID regulator.

Auto-tuning is one of these methods, which is used in PID controllers [15]. The performance of the control loops is improved by automatically adjusting the PID gain parameters of the conventional controllers.

The self-tuning method has been suggested by many researchers [16,17]. A self-adjusting mechanism has been set up to adapt the PID regulator in case of any disturbances.

The use of optimization algorithms as alternative methods for tuning PID controllers has been a recent topic of research in electric machines control. New optimization techniques are proposed, for instance, the Imperialist Competitive Algorithm (ICA) [18], evolutionary algorithm [19], Genetic Algorithm (GA) [20–21], BAT algorithm [22], Particle Swarm Optimization (PSO) [23–26], and Ant Colony Optimization (ACO) algorithm [27], Harmony Search (HS) [28], hybrid GA [29–30], adaptive Cuckoo Search algorithm (CS) [31].

PSO was first used by Eberhart and Kennedy in 1995 [32]. This approach is inspired by the social behavior shown by the natural species. In recent years, particle swarm optimization has appeared as a new and popular optimization algorithm due to its simplicity and efficiency. The role of the PSO in this study is to suggest an adequate adjustment of the parameters (kp, Ki, Kd) to satisfy some drive system requirements.
In last years, the FLC has improved results of nonlinear and complex processes [33]. The main idea of this approach is that it does not need a precise mathematical model of the electric machine, FLCs are robust and their performance is insensitive to parameter variations. With the increasing evolution of approximation theory, the adaptive control methods have been presented to cope with the nonlinear systems with parametric uncertainty based on fuzzy logic system (FLS) [34], neural networks (NNs) [35], adaptive fuzzy and NN control approaches via backstepping methods [36,37].

There are two disadvantages in the conception of a FLC. The first one is the obtaining of a suitable rule-base for the application, while the second is the selection of scaling factors prior to fuzzification and after defuzzification, in order to overwhelm these drawbacks and expedite the determination of the design parameters and to reduce the time consumption. Several solutions are adapted to remedy these problems. In [38,39] the authors presents an online method for adapting the scaling factors of the FLC, the authors suggest a solution to design an adaptive fuzzy controller. The objective of the proposed form is to adapt online scaling factors according to a performance measure in order to refine the controller and increase the performance of the drive system.

In this paper, we investigate the performance of PSO for optimizing the gains of the fuzzy-PID speed controller of the DSIM.

2. Modeling of the dual star induction motor
The DSIM dynamic equations in the reference d-q can be reported as follow [4]:

\[
\begin{align*}
V_{ds1} &= R_{s1}i_{ds1} + \frac{d}{dt}\psi_{ds1} - \omega_s\psi_{qs1} \\
V_{qs1} &= R_{s1}i_{qs1} + \frac{d}{dt}\psi_{qs1} + \omega_s\psi_{ds1} \\
V_{ds2} &= R_{s2}i_{ds2} + \frac{d}{dt}\psi_{ds2} - \omega_s\psi_{qs2} \\
V_{qs2} &= R_{s2}i_{qs2} + \frac{d}{dt}\psi_{qs2} + \omega_s\psi_{ds2} \\
V_{dr} &= 0 = R_r i_{dr} + \frac{d}{dt}\varphi_{dr} - (\omega_s - \omega_r)\psi_{qr} \\
V_{qr} &= 0 = R_r i_{qr} + \frac{d}{dt}\varphi_{qr} + (\omega_s - \omega_r)\psi_{dr}
\end{align*}
\]

where the fluxes equations are:

\[
\begin{align*}
\psi_{ds1} &= L_{s1}i_{ds1} + L_m(i_{ds1} + i_{ds2} + i_{dr}) \\
\psi_{qs1} &= L_{s1}i_{qs1} + L_m(i_{qs1} + i_{qs2} + i_{qr}) \\
\psi_{ds2} &= L_{s2}i_{ds2} + L_m(i_{ds1} + i_{ds2} + i_{dr}) \\
\psi_{qs2} &= L_{s2}i_{qs2} + L_m(i_{qs1} + i_{qs2} + i_{qr}) \\
\psi_{dr} &= L_r i_{dr} + L_m(i_{ds1} + i_{ds2} + i_{dr}) \\
\psi_{qr} &= L_r i_{qr} + L_m(i_{qs1} + i_{qs2} + i_{qr})
\end{align*}
\]

For studying the dynamic behavior, the following equation of motion was added:

\[
J \frac{d\Omega_r}{dt} = T_{em} - T_r - f_r\Omega_r
\]
The model of the DSIM has been completed by the expression of the electromagnetic torque $T_{em}$ given below:

$$T_{em} = p \frac{L_m}{L_m + L_r} (\psi_{dr} (i_{qs1} + i_{qs2}) - \psi_{qr} (i_{ds1} + i_{ds2}))$$

(4)

A schematic representation of the stator and rotor windings of dual star induction motor is given in Figure 1.

### 3. Direct torque control (DTC) of the DSIM

The classical DTC, proposed by [40], is based on the following algorithm:

- Divide the time domain into periods of reduced duration $T_s$;
- For each clock struck, measure the line currents and phase voltages of the DSIM;
- Reconstitute the components of the stator flux vector and estimate the electromagnetic torque, through the estimation of the stator flux vector and the measurement of the line current;
- The error between the estimated torque and the reference one is the input of a three level hysteresis comparator when this latter generates at its output the value of $+1$ to increase the flux and $0$ to reduce it and thus increasing the torque $-1$ it reduce this flux and $0$ to keep it constant in a band;
- The error between the estimated stator flux magnitudes is the input of a two levels of the hysteresis comparator, which generates at its output the value $+1$ to increase the flux and $0$ to reduce it;
- Select the state of the switches to determine the operating sequences of the inverter using the switching table. The input quantities are the stator flux sector and the outputs of the two hysteresis comparators.

The block diagram of the DTC of DSIM is shown in Figure 2.
Moreover, Table 1 presents the sequences corresponding to the position of the stator flux vector to the different sectors. The flux and the torque are controlled by two hysteresis comparators at 2 and 3 levels, respectively, in the case of a two-level voltage inverter Table 5.

The expression of the stator flux is described by:

\[
\psi_{s_{a1,2}} = \int_0^t (V_{sa1,2} - R_{s}i_{sa1,2}) dt \\
\psi_{s_{b1,2}} = \int_0^t (V_{sb1,2} - R_{s}i_{sb1,2}) dt
\]

where \( V_{sa1,2} \) and \( V_{sb1,2} \) are the estimated components of the stator vector voltage. They are expressed from the model of the inverter.

4. Particle swarm optimization algorithm

PSO uses a population of individuals to discover the high solution in a search area among the neighboring solutions. The individual is defined by a particle, which displaces stochastically in the guidance of its preceding finest position and the best past location of the swarm.

Presume that the size of the swarm is \( n \) and the search area is \( m \), next the position of the ith particle is given as \( xi = (x_{i1, x_{i2, \ldots x_{id}}}) \). The finest previous positions of the ith particle are considering by [32]:

<table>
<thead>
<tr>
<th>cflx</th>
<th>ccpl</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Corrector</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>( V_2 )</td>
<td>( V_3 )</td>
<td>( V_4 )</td>
<td>( V_5 )</td>
<td>( V_6 )</td>
<td>( V_1 )</td>
<td>2 levels</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>( V_7 )</td>
<td>( V_6 )</td>
<td>( V_7 )</td>
<td>( V_6 )</td>
<td>( V_7 )</td>
<td>( V_0 )</td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>( V_6 )</td>
<td>( V_1 )</td>
<td>( V_2 )</td>
<td>( V_3 )</td>
<td>( V_4 )</td>
<td>( V_5 )</td>
<td>3 levels</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>( V_5 )</td>
<td>( V_4 )</td>
<td>( V_5 )</td>
<td>( V_6 )</td>
<td>( V_1 )</td>
<td>( V_2 )</td>
<td>2 levels</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>( V_5 )</td>
<td>( V_6 )</td>
<td>( V_1 )</td>
<td>( V_2 )</td>
<td>( V_3 )</td>
<td>( V_4 )</td>
<td>3 levels</td>
</tr>
</tbody>
</table>

Table 1. Control strategy with hysteresis comparator.
The index of the best particle amongst the group is $g_{best_d}$. The velocity of particle $i$th is represented as:

$$v_i = (v_{i1}, v_{i2}, \ldots, v_{id})$$

(7)

The modified speed and position of each particle can be calculated using the current and the distance from $p_{best_i,d}$ to $g_{best_d}$ as expressed in the following equations:

$$v_{i,m}^{t+1} = w \cdot v_{i,m}^{t} + C_1 \cdot \text{rand()} \cdot (p_{best_{i,m}} - x_{i,m}^t) + C_2 \cdot \text{rand()} \cdot (g_{best_{i,m}} - x_{i,m}^t)$$

(8)

$$x_{i,m}^{t+1} = x_{i,m}^t + v_{i,m}^{t+1} \quad i = 1, 2, \ldots, n; \quad m = 1, 2, \ldots, d$$

(9)

where: $n$ is the number of particles in the group, $d$ is the dimension index, $t$ is the pointer of iterations (generations), $v_{i,m}^t$ is the velocity of particle at iteration $t$, $w$ is the inertia weight factor, $\text{rand()}$ is the random number between 0 and 1, $p_{best_i}$ is the best previous position of the $i$th particle, $g_{best}$ is the best particle among all the particles in the population, $C_1, C_2$ are acceleration constant. Velocity vector for position update is depicted in Figure 3.

4.1 Algorithm steps for PSO

The working of PSO algorithm is interpreted in the consecutive steps.

Step 1: We establish the values of PSO algorithm constants as an inertia weight factor $W = 0.8$, with acceleration constants $C_1 = C_2 = 2$. The PSO main program has to optimize in this case three parameters, $K_e, K_d$ and $\beta$ to the fuzzy controller, and search optimal value of the three-dimensional search space.

Step 2: So we arbitrarily configured a swarm of “50” population in three-dimensional search space using $(x_{i1}, x_{i2}, x_{i3})$ and $(v_{i1}, v_{i2}, v_{i3})$ as preliminary situation along with velocity. Considered the primary fitness function of any also to the position with minimum fitness function is displayed as best, so the optimal fitness function as $p_{best1}$.

Figure 3. Description of a searching point by PSO.
Step 3: Run the program by means of PSO algorithm through n numbers of iterations, as well, the program finds the final optimal value of the fitness function as “best fitness” with the last overall optimal point as “gbest”. The PSO parameters are described in Table 2 in the Appendix. The flowchart for fuzzy PSO-DTC-DSIM is shown in Figure 4.

4.2 Fitness function

4.2.1 The conception of fitness function. To evaluate the static and dynamic conduct of the control system, it is found that IAE offers good system stability with reduced oscillations the IAE criterion is widely adopted [41]:

![Flowchart for fuzzy PSO-DTC-DSIM.](image-url)
\[ IAE = \int_0^\infty |e(t)| dt \] (10)

5. Design of PID-PSO controller type FLC for the DSIM

The optimization of the FLC gains using PSO can be given by the input variable \( \{e\} \), and the error change \( \{e_c\} \) as follows:

\[ e(t) = \Omega_{ref} - \Omega_r(t) \] (11)

\[ e_c(t) = \frac{de(t)}{dt} \] (12)

Table 3 illustrated the performance of PID controller in the Appendix.

The fuzzy PI controller is the commonly used because the PD one encounters difficulties in deleting the steady state error. However, the fuzzy-PI gives a poor performance in the transient response in higher order systems because of its inherent internal integration operation. It is therefore more convenient to combine PI and PD actions to design a fuzzy PID controller (FLC-PID) to achieve proportional, integral, and derivative control action. It is imperative to obtain an FLC-PID controlled by adding the fuzzy-PD controller output and its embedded part. The fuzzy-PID controller is depicted in Figure 5. Table 4 represented the performance of fuzzy controller is in the Appendix.

<table>
<thead>
<tr>
<th>Descriptions</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of particles in the swarm</td>
<td>50</td>
</tr>
<tr>
<td>Number of Iterations</td>
<td>10</td>
</tr>
<tr>
<td>Number of components or dimension</td>
<td>3</td>
</tr>
<tr>
<td>Inertia weight factor ( w )</td>
<td>0.8</td>
</tr>
<tr>
<td>( C_1=C_2 )</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 2. Parameters of PSO algorithms.

<table>
<thead>
<tr>
<th>Controllers</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K_p )</td>
<td>37.5</td>
</tr>
<tr>
<td>( K_i )</td>
<td>0.35</td>
</tr>
<tr>
<td>( K_d )</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3. Performance of PID controller.

![Diagram](image)

Figure 5. The proposed control structure for PID-PSO type FLC.
The output $u$ of the fuzzy PID is presented by:

$$u = \alpha U + \int \beta U \, dt$$  \hfill (13)$$

where: $U$ is the output of the FLC.

The relationship between the input and output variables is given by [42]:

$$U = A + PE + D \frac{dE}{dt}$$  \hfill (14)$$

where: $E = K_e e$ and $\frac{dE}{dt} = K_d \frac{de}{dt}$ according to Figure 5.

Therefore, from Eqs. (13) and (14) the controller output is expressed by the following equation:

$$u = \alpha A + \beta A t + \alpha K_e Pe + \beta K_d D e + \beta K_e P \int edt + \alpha K_d D \frac{de}{dt}$$  \hfill (15)$$

Finally, the components of PID-FLC can be deducted as follows:

The proportional gain: $\alpha K_e P + \beta K_d D$;

The integral gain: $\beta K_e P$;

The derivative gain: $\alpha K_d D$.

### 5.1 Fuzzification

The inputs to the Fuzzy-PSO have to be fuzzified before being fed into the control rule and gain rule determinations. The triangular membership functions (MFs) used for the input ($e, e_c$ and, $\Delta T_{em}$) are shown in Figures 6 and 7. Linguistic variables are (NB, NM, NS, EZ, PS, PM, PB).

Where: NB is Negative Big, NM is Negative Medium, NS is Negative Small, EZ is Equal Zero, PS is Positive Small, PM is Positive Medium, PB is Positive Big.

### 5.2 Inference and defuzzification

The present paper uses MIN operation for the calculation of the degree $\mu(\Delta T_{em})$ associated with every rule, for example, $\mu(\Delta T_{em}) = \text{Min}[\mu(e), \mu(e_c)]$.

In the defuzzification stage, a crisp value of the electromagnetic torque is obtained by the normalized output function as [33]:

$$du = \frac{\sum_{j=1}^{m} \mu(\Delta T_{em}) \Delta T_{emj}}{\sum_{j=1}^{m} \mu(\Delta T_{emj})}$$  \hfill (16)$$

where: $m$ is the total number of rules ($7^*7$), $\mu(\Delta T_{em})$ is the membership grade for the $n$ rule, $\Delta T_{em}$ is the position of membership functions in rule $n$ in $U (-15, -10, 5, 0, 10, 15)$.

### Table 4.

<table>
<thead>
<tr>
<th>Controller</th>
<th>Fuzzy-PSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input scaling factor $k_e$ optimized</td>
<td>3.1604</td>
</tr>
<tr>
<td>Input scaling factor $k_d$ optimized</td>
<td>3.6741</td>
</tr>
<tr>
<td>$\beta$ is the gain of the integral component</td>
<td>0.8081</td>
</tr>
<tr>
<td>$\alpha$ scaling factor for the output $u$</td>
<td>1</td>
</tr>
</tbody>
</table>

Controller Fuzzy-PSO Input scaling factor $k_e$ optimized 3.1604 Input scaling factor $k_d$ optimized 3.6741 $\beta$ is the gain of the integral component 0.8081 $\alpha$ scaling factor for the output $u$ 1
5.3 Control rule determination

The logic of determining this rule matrix is based on a global knowledge of the system operation. As an example, we consider the following two rules:

- if $e$ is PB and $ec$ is PB then $\Delta Tem$ is PB
- if $e$ is ZE and $ec$ is ZE then $\Delta Tem$ is ZE

They indicate that if the speed is too small compared to its reference ($e$ is PB), so a big gain ($\Delta Tem$ is PB) is required to bring the speed to its reference and if the speed reaches its reference and is established ($e$ is ZE and $e_c$ is ZE) so impose a small gain $\Delta Tem$ is ZE.

Table 5 represents the inference rules.
6. Simulation results and discussion
The results were obtained using a PSO algorithm programmed and implemented in MATLAB. The parameters of the DSIM are presented in Table 6 (Appendix). To illustrate the performances of the DTC of the DSIM we replaced the classical PID controller by a fuzzy-PSO technique in Figure 8. The simulation is carried out under the following conditions: the hysteresis band of the torque comparator is set to $\pm 0.25$ Nm and that of the flux comparator to $\pm 0.5$ Wb.

Figure 9 depicts the waveforms of the improved performances of speed control. It can be noticed that the use of the fuzzy-PSO controller allows the speed to judiciously follow its reference value of 100 rad/s despite the presence of a load torque of 14Nm at $t = 0.6$ s. In fact, this behaviour represents a clear improvement in dynamic response with a hybrid controller.

<table>
<thead>
<tr>
<th>$\Delta T_{em}$</th>
<th>NB</th>
<th>NM</th>
<th>NS</th>
<th>$e_c$</th>
<th>ZE</th>
<th>PS</th>
<th>PM</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E$</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
</tr>
<tr>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
</tr>
<tr>
<td>NS</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
</tr>
<tr>
<td>ZE</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
</tr>
<tr>
<td>PB</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
</tr>
</tbody>
</table>

Table 5. Inference rules.

<table>
<thead>
<tr>
<th>Rated Power</th>
<th>4.5KW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stator Resistance $R_s$</td>
<td>3.72$\Omega$</td>
</tr>
<tr>
<td>Rotor Resistance $R_r$</td>
<td>2.12$\Omega$</td>
</tr>
<tr>
<td>Stator Inductance $L_s$</td>
<td>0.022H</td>
</tr>
<tr>
<td>Rotor Inductance $L_r$</td>
<td>0.006H</td>
</tr>
<tr>
<td>Mutual Inductance $L_m$</td>
<td>0.3672H</td>
</tr>
<tr>
<td>Pole Pairs $P$</td>
<td>1</td>
</tr>
<tr>
<td>Machine Inertia $J$</td>
<td>0.0662 kg.m$^2$</td>
</tr>
<tr>
<td>Viscous Friction Coefficient $f_r$</td>
<td>0.001 kg.m$^2$/s</td>
</tr>
</tbody>
</table>

Table 6. DSIM parameters [12].

Figure 8. Block diagram of the proposed DTC-fuzzy-PSO tuning speed controller.
that is adjusting strictly the values of the parameters by increasing the constant of integration without an overshoot at the level of the dynamic response of the speed, contrary to a drive with a standard DTC-PID where the speed has underwent slightly rejected.

Performance with each controller is also analyzed through these of Integral Squared Error (ISE), Integral Absolute Error (IAE) and Integral Time Squared Error (ITSE), and the results described in Table 7 confirm the improved performance with the fuzzy-PSO algorithm.

In Figure 10 the electromagnetic torque produced by the DSIM controlled by DTC-PID and DTC-fuzzy-PSO is presented. In this figure, it can be noticed that the ripple is not the same for the two techniques. It is clear that the classical DTC-PID present two problems, steady state error and high torque ripples. On the other hand, the DTC-fuzzy-PSO corrects the steady state error and reduces the torque ripples.

<table>
<thead>
<tr>
<th>Controllers</th>
<th>IAE</th>
<th>ISE</th>
<th>ITSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>0.5473</td>
<td>0.1498</td>
<td>0.1348</td>
</tr>
<tr>
<td>Fuzzy-PSO</td>
<td>0.2072</td>
<td>0.0215</td>
<td>0.0193</td>
</tr>
</tbody>
</table>

Table 7. Comparison of performance index.

Figure 9. Comparison of the rotor speed regulation of the standard DTC and hybrid DTC.

Figure 10. Electromagnetic torque comparison of the two strategies.
In Figure 11, it can be observed that the currents are sinusoidal and current ripples have also a notable reduction in fuzzy-PSO controller compared to the standard controller. Figure 12 shows the trajectory of stator flux for the standard DTC and the hybrid DTC. It can be seen that this hybrid strategy has less ripple. Figure 13 summarizes the evolution of the fitness function with respect to the number of iterations.

7. Conclusion
In this paper, a comparative study between the conventional DTC of the DSIM with PID controller and DTC-fuzzy-PSO has been presented for a speed controller. From the simulation studies, hybrid controller produced better performances in terms of a fast rise time, a small overshoot, reduced torque and flux ripples. Therefore very satisfactory performances have been achieved. Furthermore, the effectiveness of the proposed algorithms is evaluated and justified from performance indices IAE, ISE and ITSE. According to the yielded simulation results one can conclude that this algorithm is suitable for applications requiring a high
tracking accuracy in presence of external disturbances. In future, the work can be extended by the applications of the intelligent hybrid techniques like neuro-fuzzy-GA, neuro-fuzzy-PSO, neuro-fuzzy-ACO.

**Abbreviations**

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSIM</td>
<td>dual star induction motor</td>
</tr>
<tr>
<td>FLC</td>
<td>fuzzy logic controller</td>
</tr>
<tr>
<td>PSO</td>
<td>particle swarm optimization</td>
</tr>
<tr>
<td>DTC</td>
<td>direct torque control</td>
</tr>
<tr>
<td>PID</td>
<td>proportional integral derivative</td>
</tr>
<tr>
<td>IAE</td>
<td>the integral of absolute value of the error</td>
</tr>
<tr>
<td>ISE</td>
<td>the integral of square error</td>
</tr>
<tr>
<td>ITSE</td>
<td>the integral of time multiply square error</td>
</tr>
</tbody>
</table>

**References**


Appendix
See.

**Corresponding author**
Sebti Belkacem can be contacted at: belkacem_sebti@yahoo.fr

For instructions on how to order reprints of this article, please visit our website: [www.emeraldgrouppublishing.com/licensing/reprints.htm](http://www.emeraldgrouppublishing.com/licensing/reprints.htm)
Or contact us for further details: permissions@emeraldinsight.com