NEURO-FUZZY CONTROL OF A DUAL STAR INDUCTION MACHINE

L. BENTOUHAMI  R. ABDESSEMED  Y. BENDJEDDOU  E. MERABET
LEB-Research Laboratory Department of Electrical Engineering, University of Batna-2.
Street Chahid Mohamed El Hadi Boukhlout, 05000. Batna, Algeria, l.bentouhami@univ-bba.dz.

Abstract: This paper presents a new approach to the speed control of dual star induction machine (DSIM). The new technique uses a Neuro-Fuzzy Control (NFC) with an indirect field oriented method. For realize a control of this machine, two voltage source inverters of PWM techniques are introduced. Simulation results are presented for NFC controller, the results are compared with the results obtain from a Proportional Integral (PI) controller. It is observed that the NFC controller give better responses and robustness for the speed control of this machine with introduces load disturbances and parameters variations, such as an increased rotor resistance and moment of inertia.

Key words: Dual star induction machine (DSIM), Indirect field oriented control (IFOC), Voltage source inverter (VSI) and Neuro-fuzzy controller (NFC).

1. Introduction

The inherent disadvantages of DC drives have prompted continual progress in the area of AC drives. Among AC drives, induction machine drives are widely used [1]. Multi-phase machines are AC machines characterized by a stator winding composed of generic number of phases. Actually, electric drive and generation technology multiphase machine has several advantages over the traditional three phase machine such as reducing the amplitude and increasing the frequency of torque pulsation, reducing the rotor harmonic current per phase without increasing the voltage per phase, lowering the dc-link current harmonics and higher reliability, high fault tolerance [2,3,4,5,6]. This characteristic makes them convenient in high power and/or high current applications, such as ship propulsion, aerospace applications, and electric/ hybrid vehicles (EV) [7]. other suitable applications are pumps, fans, compressors, rolling mills, cement mills, mine hoists…[8]. A common type of multiphase machine is the dual star induction machine (DSIM). This machine has been used in many applications since the late 1920s [9]. The main difficulty in the asynchronous machine control resides in the fact that complex coupling exists between the field and the torque. The vector control assures decoupling between the field and the torque is made similar to that of a DC machine [10]. The indirect vector control scheme of the dual stator winding induction machine is based on the assumption that the machine can be treated as two independent induction machines coupled through the same rotor shaft [11,12].

Emil Levi [13] provides a review of the recent developments in the area of multiphase induction motor control. The conventional PI and PID controllers are employed in speed and current controller, but these controllers are very sensitive to machine parameters variation, load disturbances and along with step changes of command speed [14,15,16].

Numerous methods have been proposed to replace PI controller schemes, such as fuzzy logic controller (FLC) [17,18,19] and artificial neural networks (ANN’s) [20,21,22]. A simple fuzzy controller implemented in the motor drive speed control has a narrow speed operation and needs much manual adjusting by trial and error if high performance is wanted [15]. On the other hand, it is extremely tough to create a serial of training data for ANN that can handle all the operating modes [23].

A Neuro-Fuzzy Controller, or simply NFC controller for the induction motor drive, which has the advantages of both FLC and ANN is proposed.

So, in this paper, a NFC for speed control of a DSIM drive system is proposed. Neuro-Fuzzy (NFC) developed in the early 90s by Jang [24,25,26], combines the concepts of fuzzy logic and neural networks to form a hybrid intelligent system that enhances the ability to automatically learn and adapt. An ANN is used to adjust input and output parameters of membership functions in FLC. The back propagation learning algorithm is used for training this network.

The DSIM is preliminary simulated with conventional PI speed regulator in order to establish a term of comparison. The results simulation obtained by using Matlab program is provided to confirm the validity of the proposed techniques.

The paper is organized as follows: the model of machine and IFOC description are given in section 2 and section 3 respectively. Section 4 is devoted to the design of the NFC controller. The simulation results are shown in section 5. The conclusions are given in section 6.
2. Machine Model

A common type of multiphase machine is the dual star induction machine (DSIM), where two sets of three-phase windings, spatially phase shifted by 30 electrical degrees, share a common stator magnetic core as shown in Fig. 1 [17]. The modeling and control of DSIM in the original reference frame would be very difficult. For this reason, it is necessary to obtain a simplified model.

As consequence, also the mathematical modeling approach of DSIM is similar to the standard induction machine ones and usually it is obtained under the same simplifying assumptions [10] [17]:
- The two stars have same parameters,
- Motor windings are sinusoidally disturbed,
- The mutual leakage inductances are included,
- The magnetic saturation and the core losses are neglected.

The expressions for stator and rotor flux linkages are:
\[
\phi_{s1} = l_s i_{s1} + L_m (i_{s1} + i_{s2} + i_{dr}) \\
\phi_{q1} = l_s i_{q1} + L_m (i_{q1} + i_{q2} + i_{qr}) \\
\phi_{s2} = l_s i_{s2} + L_m (i_{s2} + i_{s1} + i_{dr}) \\
\phi_{q2} = l_s i_{q2} + L_m (i_{q2} + i_{q1} + i_{qr}) \\
\phi_{dr} = l_r i_{dr} + L_m (i_{s1} + i_{s2} + i_{dr}) \\
\phi_{qr} = l_r i_{qr} + L_m (i_{s1} + i_{s2} + i_{qr})
\]

The following voltage equations are written for a DSIM in the synchronous reference frame [9]:
\[
v_{d1} = R_s i_{d1} + p \phi_{ds1} - \omega_s \phi_{q1} \\
v_{q1} = R_s i_{q1} + p \phi_{qs1} + \omega_s \phi_{ds1} \\
v_{d2} = R_s i_{d2} + p \phi_{ds2} - \omega_s \phi_{qs2} \\
v_{q2} = R_s i_{q2} + p \phi_{qs2} + \omega_s \phi_{ds2} \\
v_{dr} = R_r i_{dr} + p \phi_{dr} - \omega_s (\phi_{qdr}) \\
v_{qr} = R_r i_{qr} + p \phi_{qr} - \omega_s (\phi_{qdr})
\]

The mechanical equation is given by:
\[
T_{em} = \frac{l_m}{l_{em}} [\phi_{dr} (i_{q1} + i_{q2}) - \phi_{qr} (i_{q1} + i_{q2})] \\
\frac{1}{P} \omega_r = T_{em} - T_L - \frac{K_f}{\omega_r}
\]

Where \( \omega_s, \omega_r \) speed of synchronous reference frame and rotor electrical angular; \( l_s, l_r \) stator and rotor inductances; \( L_m \) resultant magnetizing inductance; \( P \) number of pole pairs; \( I_m \) moment of inertia; \( T_L \) load torque; \( K_f \) total viscous friction coefficient.

3. Indirect Field Oriented Control of a DSIM

The main goal of a field-oriented control (FOC) is to obtain decoupled control of electromagnetic torque and rotor flux of the motor as in DC machines [10][17]. The indirect field oriented control (IFOC) method is the most commonly used because of its relative simplicity and low cost of implementation. The \( d \)-axis is aligned with the rotor flux space vector. In ideal field oriented control, the rotor flux linkage axis is forced to align with the \( d \)-axis, and it follows that: The direct flux equals to the reference:
\[
\phi_{dr} = \phi_{ds1}^* \\
\phi_{qr} = \phi_{q1}^*
\]

The quadrature component of the flux null:
\[
p \phi_{dr} = \phi_{qr} = 0
\]

The commands/references voltage \( v_{ds1}^*, v_{q1}^*, v_{ds2}^* \) and \( v_{q2}^* \) are derived by substituting the (15) and (16) in (7)-(10):
\[
v_{ds1}^* = R_s i_{ds1} + l_s p i_{ds1} - \omega_s (l_s i_{q1} + T_r \phi_r^* \omega_s^* ) \\
v_{q1}^* = R_s i_{q1} + l_s p i_{q1} + \omega_s (l_s i_{q2} + T_r \phi_r^* \omega_s^* ) \\
v_{ds2}^* = R_s i_{ds2} + l_s p i_{ds2} - \omega_s (l_s i_{q2} + \phi_r^* ) \\
v_{q2}^* = R_s i_{q2} + l_s p i_{q2} + \omega_s (l_s i_{q2} + T_r \phi_r^* \omega_s^* )
\]

The component references of stator current and slip speed \( \omega_s \) can be expressed as:
\[
\omega_s^* = \frac{R_s l_m}{l_{em} + l_{r}} \phi_r^* \\
i_{ds}^* = \frac{1}{l_m} \phi_r^* \\
i_{qs}^* = \frac{1}{l_{em}} \phi_r^* \\
T_r^* = \frac{1}{l_r}
\]
the $d$-$q$ axes currents are relabeled as flux-producing ($i_{dS}^*$) and torque-producing ($i_{qS}^*$) components of the stator current phases, respectively.

$T_r$ : denotes the rotor time constant.

For generate two sets of command/reference voltage vectors ($v_{dS1}^*, v_{qS1}^*, v_{dS2}^*$ and $v_{qS2}^*$), two independent pairs of PI controllers are introduced.

4. Design of the NFC controller

The proposed NFC incorporates fuzzy logic and a learning algorithm with a four-layer artificial neural network (ANN) structure as depicted in Fig. 2 with back propagation method. It describes the relationship between the speed error $e_ω$, its change $\dot{e}_ω$ and the control signal $T_{emNFC}$ [27]. The goal of the PI controller is to obtain a desired electromagnetic torque $T_{em}^*$.

$e_{tem}$ represents the error between the desired torque ($T_{em}^*$) and the control torque $T_{emNFC}$ (its current value give by NFC controller). The detailed discussions on different layers of the Neuro-fuzzy network fig. 3, are given below.

Layer I: Each input node in this layer corresponds to the specific input variable, the inputs of this layer are given by: $net_1^I = e_ω$ and $net_2^I = \dot{e}_ω$.

The outputs of this layer are given by $y_1^I = f_1^I (net_1^I) = e_ω$ and $y_2^I = f_2^I (net_2^I) = \dot{e}_ω$.

The weights of this layer are unity and fixed.

Layer II: Seven membership function based fuzzy set is utilized to obtain the fuzzy number for each input. In the propose NFC, Gaussian functions are chosen as the membership function as shown in fig. 4.

$$net_{1,j}^II = -\left(\frac{x_{1,j} - m_{1,j}^II}{\sigma_{1,j}^II}\right)^2$$ (27)

$$net_{2,k}^II = -\left(\frac{x_{2,k} - m_{2,k}^II}{\sigma_{2,k}^II}\right)^2$$ (28)

$$y_{1,j}^II = f_{1,j}^II (net_{1,j}^II) = exp (net_{1,j}^II)$$ (29)

$$y_{2,k}^II = f_{2,k}^II (net_{2,k}^II) = exp (net_{2,k}^II)$$ (30)

Where: $m_{1,j}^II$, $m_{2,k}^II$ and $\sigma_{1,j}^II$, $\sigma_{2,k}^II$ are the mean and the deviation of the Gaussian function.

Layer III: Every node in this layer is a fixed node labeled II, for represent a rule base used in the FLC, a product operator is used in each node.
A 49 rules base for the NFC is given in Tab. 1.

\[
\text{net}^{\text{IV}}_{jk} = (x^{\text{III}}_{1,j} \times x^{\text{III}}_{2,k}) y^{\text{III}}_{jk} = f^{\text{III}}_{jk} (\text{net}^{\text{III}}_{jk}) = \text{net}^{\text{III}}_{jk}
\]

(31)

The values of weights between layer III and layer II are unity.

**Layer IV:** The center of gravity method is used to determine the output of NFC. Each node equation is specified as

\[
a = \sum_j \sum_k (w^{\text{IV}}_{jk} y^{\text{IV}}_{jk}), \quad \sum_j \sum_k y^{\text{IV}}_{jk}
\]

\[
\text{net}^{\text{IV}}_0 = \frac{a}{b}, \quad y^{\text{IV}}_0 = f^{\text{IV}}_0 (y^{\text{IV}}_0) = \frac{a}{b}
\]

(32)

(33)

\(w^{\text{IV}}_{jk}\) represent the values of the output membership functions used in the FLC as shown in fig. 5. \(y^{\text{IV}}_0\) is output of the layer IV. \(a\) and \(b\) are the numerator and the denominator of the function used in the center of area method, respectively. In the NFC, the goal of learning algorithm is adjustment the weights\(w^{\text{IV}}_{jk}\), the \(m^{\text{III}}_{1,j}, m^{\text{III}}_{2,k}\) and \(\sigma^{\text{III}}_{1,j}, \sigma^{\text{III}}_{2,k}\). For the learning algorithm we use the supervised gradient descent method. The error E we take for describe the back propagation algorithm.

\[
E(I) = \frac{1}{2} \varepsilon^{\text{rem}}_I
\]

(34)

Where the error value can be written as

\[
e^{\text{rem}}_I = d - y
\]

(35)

4.1. The back propagation algorithm

**Layer IV:** The error expression for the input of this layer \(\sigma^{\text{IV}}_0 = - \frac{\partial E}{\partial y^{\text{IV}}_0} = - \frac{\partial E}{\partial e} \frac{\partial e}{\partial y^{\text{IV}}_0} = e\)

(36)

The variations \(\Delta w^{\text{IV}}_{jk}\) of the weights \(w^{\text{IV}}_{jk}\) to minimize the error can be determined by the generalized delta rule as follows

\[
\Delta w^{\text{IV}}_{jk} = - \frac{\partial E}{\partial w^{\text{IV}}_{jk}} = - \left[ \frac{\partial E}{\partial y^{\text{IV}}_0} \frac{\partial y^{\text{IV}}_0}{\partial a} \frac{\partial a}{\partial w^{\text{IV}}_{jk}} \right] = \frac{1}{b} \sigma^{\text{IV}}_0 y^{\text{IV}}_0
\]

(37)

Finally, the weights \(w^{\text{IV}}_{jk}\) can be updated to minimize the error as follows

\[
w^{\text{IV}}_{jk}(t) = w^{\text{IV}}_{jk}(t - 1) + \mu_w \Delta w^{\text{IV}}_{jk}(t)
\]

(38)
Where $\mu_w$ is learning rate for $w_{jk}^{IV}$.

**Layer III:** The error received by this layer from layer IV is computed as

$$
\delta_{j,k}^{III} = -\frac{\partial E}{\partial t_{jk}} = \left[-\frac{\partial E}{\partial t_{jk}^{IV}} \frac{\partial y_{jk}^{IV}}{\partial t_{jk}} \frac{\partial y_{jk}^{IV}}{\partial t_{jk}} \frac{\partial y_{jk}^{IV}}{\partial t_{jk}} \right] = \frac{1}{b} \delta_0 (w_{jk}^{IV} - y)
$$

**Layer II:** The error received from Layer III is computed as

$$
\delta_{i,j}^{II} = -\frac{\partial E}{\partial t_{ij}} = \Sigma_k \left[-\frac{\partial E}{\partial t_{jk}^{III}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \right] = \Sigma_k \delta_{jk}^{III} \frac{y_{jk}^{III}}{t_{jk}^{III}}
$$

$$
\delta_{i,j}^{II} = -\frac{\partial E}{\partial t_{ij}} = \Sigma_k \left[-\frac{\partial E}{\partial t_{jk}^{III}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \right] = \Sigma_k \delta_{jk}^{III} \frac{y_{jk}^{III}}{t_{jk}^{III}}
$$

In this layer, the changes of $m_{1,j}^{II}$, $m_{2,k}^{II}$ and $\sigma_{1,j}^{II}$, $\sigma_{2,k}^{II}$ are written as

$$
\Delta m_{1,j}^{II} = -\frac{\partial E}{\partial m_{1,j}^{II}} = \left[-\frac{\partial E}{\partial t_{jk}^{III}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \right] = \frac{2(x_{1,j}^{II} - m_{1,j}^{II})}{(\sigma_{1,j}^{II})^2}
$$

$$
\Delta m_{2,k}^{II} = -\frac{\partial E}{\partial m_{2,k}^{II}} = \left[-\frac{\partial E}{\partial t_{jk}^{III}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \right] = \frac{2(x_{2,k}^{II} - m_{2,k}^{II})}{(\sigma_{2,k}^{II})^2}
$$

$$
\Delta \sigma_{1,j}^{II} = -\frac{\partial E}{\partial \sigma_{1,j}^{II}} = \left[-\frac{\partial E}{\partial t_{jk}^{III}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \right] = \frac{2(x_{1,j}^{II} - m_{1,j}^{II})}{(\sigma_{1,j}^{II})^2}
$$

$$
\Delta \sigma_{2,k}^{II} = -\frac{\partial E}{\partial \sigma_{2,k}^{II}} = \left[-\frac{\partial E}{\partial t_{jk}^{III}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \frac{\partial y_{jk}^{III}}{\partial t_{ij}} \right] = \frac{2(x_{2,k}^{II} - m_{2,k}^{II})}{(\sigma_{2,k}^{II})^2}
$$

Membership parameters adaptation is obtained through the following:

$$
m_{1,j}^{II}(t) = m_{1,j}^{II}(t - 1) + \mu_m \Delta m_{1,j}^{II}(t)
$$

$$
m_{2,k}^{II}(t) = m_{2,k}^{II}(t - 1) + \mu_m \Delta m_{2,k}^{II}(t)
$$

$$
\sigma_{1,j}^{II}(t) = \sigma_{1,j}^{II}(t - 1) + \mu_\sigma \Delta \sigma_{1,j}^{II}(t)
$$

$$
\sigma_{2,k}^{II}(t) = \sigma_{2,k}^{II}(t - 1) + \mu_\sigma \Delta \sigma_{2,k}^{II}(t)
$$

Where $\mu_m$ and $\mu_\sigma$ are learning rates, respectively, for $m_{1(2),j}^{II}$ and $\sigma_{1(2),j}^{II}$.

4. **Simulation results under proposed structure**

Simulations results for FLC of DSIM based on IFOC are presented as follows:

The Fig. 6 presents the responses with the application of a rotor speed inversion with an increased moment of inertia ($\Delta J \% = +50\%$) on no load.

In Fig. 7, the system is simulated for a step reference speed, with an increased rotor resistance ($\Delta Rr \% = +50\%$) followed by applying load torque (14N.m). The figure (8) presents responses to many reference speeds for an increased rotor resistance ($\Delta Rr \% = +50\%$) after $t = 1.5$s on no load.

Table 2 shows the parameters of the DSIM used for simulation whose general specifications are 4.5kW, 2753rpm, 220/380V, 2 poles.

<table>
<thead>
<tr>
<th>Motor parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stator resistance</td>
<td>R_s=3.72 Ω</td>
</tr>
<tr>
<td>Rotor resistance</td>
<td>R_r=2.12 Ω</td>
</tr>
<tr>
<td>Stator leakage inductance</td>
<td>L_s=0.022 H</td>
</tr>
<tr>
<td>Rotor leakage inductance</td>
<td>L_m=0.006 H</td>
</tr>
<tr>
<td>Resultant magnetizing inductance</td>
<td>L_w=0.3672 H</td>
</tr>
<tr>
<td>Moment of inertia</td>
<td>J=0.0662 kg.m²</td>
</tr>
<tr>
<td>Viscous friction coefficient</td>
<td>K_w=0.001 kg.m²/s</td>
</tr>
</tbody>
</table>

Fig. 6, shows the reference and simulated values of rotor speed, electromagnetic torque, direct and quadratic rotor flux respectively.

It can be seen in Fig. 6(a), that the rotor speed follows its reference (2500rpm) in a good manner when the load torque is zero, the PI controller for the rotor speed has the fast response time in transient state. The actual speed tracks the negative reference speed (~2500 rpm) for both controllers, after 2.5s the inertia variation ($J=1.5J_0$) increases the inversion time of rotating direction. The starting electromagnetic torque reaches until (66N.m) with PI and (62N.m) for NFC controllers, Fig. 6(b). This figure shows also that the electromagnetic torque decreases to stabilize.
b) Electromagnetic torque response of PI and NFC controllers.

c) Direct and quadratic rotor flux.

Fig. 6. Simulated responses to inverse of the rotor speed at $t = 2s$ with an increased moment of inertia ($\Delta J \% = +50\%$) at $t = 2.5s$ with no load.

a) The rotor speed with PI and NFC Controllers.
The performances of the proposed NFC controller has been also tested for many reference speeds (1000, 2000 and 2500 rpm) with different values of the rotor resistance, Fig. 8.

In Fig. 8(a), the rotor speeds reach their references (1000, 2000 and 2500 rpm), but they have important oscillations in the case of PI controller especially for 2500 rpm mostly with Rr = 1.5Rr. The same thing is observed for the electromagnetic torque when this latter oscillates until +22 and -57N.m with Rr = 1.5Rr and 2500 rpm, Fig. 8(b). Also, Fig. 8(c) shows the same situation for the direct and quadratic rotor flux.

This comparison shows clearly that the NFC controller gives better performances and robustness than the PI controller.

5. Conclusion
In this paper, a new approach NFC control is established where four layer NN are used to adjust input and output parameters of membership functions in NFC. The simulation results show that the NFC give better performances compared to a traditional controller PI.

Also, the robustness tests show too that the NFC controller is more robust than the PI controller when load disturbances occurred, and when some motor parameters (i.e. rotor resistance and moment of inertia) were increased.

6. References


