A Neural Network Based Speed Control of a Dual Star Induction Motor

Meliani Bouziane, Meroufel Abdelkader
Faculty of Engineer Science, Department of Electrical Engineering, Djilali Liabes University, Algeria

ABSTRACT

This paper proposes the use of artificial neural networks to control the speed of a Double Star Induction Motor driven by a two matrix converter using Venturini modulation algorithm. The advent of the field-oriented with modern speed control technique has partially solved DSIM control problems because it is sensitive to drive parameter variations and performance may deteriorate if conventional controllers are used. Neural network based controller is considered as potential candidates for such an application. In this work the simulations results are provided to evaluate performance of the proposed control strategy.

Keyword:
Dual Star
Field Oriented Control
Induction Machine
Matrix Converter
Neural Network

1. INTRODUCTION

The use of six-phase induction motor for industrial drives presents several advantages over the conventional three-phase drive such as improved reliability, magnetic flux harmonics reduction, torque pulsations minimization, and reduction on the power ratings for the static converter. For these reasons, six-phase induction motors are beginning to be a widely acceptable alternative in high power applications. During the last years, the modeling and control of double star induction machine has been the subject of investigations [1, 2], it is desirable to control the flux and torque separately in order to have the same performances as those of DC motors. One way of doing this is by using the field oriented control. This method assures the decoupling of flux and torque. The vector-controlled DSIM with a conventional PI speed controller is used extensively in industry, because has easily implemented. Alongside this success, the problem of tuning PI-controllers has remained an active research area. Furthermore, with changes in system dynamics and variations in operating points PI-Controllers should be returned on a regular basis. One of the most noticeable control theories is the method using the Adaptive Neural Network. Recently, the neural network (NN) is widely used as a universal approximator in the area of nonlinear mapping and uncertain nonlinear control problems [3]. The NN structure is to be implemented by input output nonlinear mapping models and is constructed with input, output and hidden layers of activation functions. Because the NN can be used for a universal approximator like fuzzy and neural systems, it has been introduced as a possible solution to the real multivariate interpolation problem.

The induction motor drive fed by a matrix converter is superior to the conventional PWM-VS inverter because of the lack of bulky DC-link capacitors with limited life time, the bi-directional power flow capability, the sinusoidal input/output currents, and adjustable input power factor. Furthermore, because of a high integration capability and a higher reliability of the semiconductor device structures, the matrix

Corresponding Author:
Meliani Bouziane,
Faculty of Engineer Science,
Department of Electrical Engineering,
Djilali Liabes University, 2200, Algeria,
Email: melfa06@yahoo.fr.

Copyright © 2014 Institute of Advanced Engineering and Science. All rights reserved.
converter topology is recommended for extreme temperatures and critical volume/weight applications. However, only a few practical matrix converters have been applied to induction motor drive systems because the implementation of the switch devices in the matrix converter is difficult and modulation technique and commutation control are more complicated than the conventional PWM inverter [4, 5].

2. DOUBLE STAR INDUCTION MODELING

Explaining research chronological, including research design, research procedure (in the form of algorithms, Pseudocode or other), how to test and data acquisition [1]-[3]. The description of the course of research should be supported references, so the explanation can be accepted scientifically [2], [4].

The machine studied is represented by two stators windings: \( s_{a1}, s_{b1}, s_{c1} \) and \( s_{a2}, s_{b2}, s_{c2} \), which are displaced by \( \alpha = 30^\circ \) and the rotor phases: \( r_{a1}, r_{b1}, r_{c} \), this is a most rugged and maintenance free machine.

![Double stator winding representation](image)

Figure 1. Double stator winding representation

The following assumptions have been made in deriving the machine model:
- Machine windings are sinusoidally distributed
- Machine magnetic saturation and the mutual leakage inductances are neglected
- The two stars have same parameters

The mathematical model of the machine is written as a set of state equations, both for the electrical and mechanical parts, the voltage equation is [2]:

\[
\begin{align*}
[\Phi_{d1}] & = R_{11}[I_{d1}] + \frac{d}{dt}[\Phi_{d1}] - \omega_s[\Phi_{q1}]
\end{align*}
\]

\[
\begin{align*}
[\Phi_{d2}] & = R_{22}[I_{d2}] + \frac{d}{dt}[\Phi_{d2}] - \omega_s[\Phi_{q2}]
\end{align*}
\]

\[
\begin{align*}
[\Phi_{d3}] & = R_{33}[I_{d3}] + \frac{d}{dt}[\Phi_{d3}] - (\omega_s - \omega_r)[\Phi_{q3}]
\end{align*}
\]

(1)

with:

\[
\begin{align*}
[\Phi_{d12}] & = L_{11}[I_{d1}] + L_m(I_{d1} + I_{d2} + I_{d3})
\end{align*}
\]

\[
\begin{align*}
[\Phi_{d3}] & = L_{13}[I_{d3}] + L_m(I_{d1} + I_{d2} + I_{d3})
\end{align*}
\]

(2)

the electrical state variables in the "dq" system are the flux represented by vector \([\Phi] \), while the input variable in the "dq" system are expressed by vector \([V] \).
\[
\frac{d}{dt}[\Phi] = [A][\Phi] + [B][V]
\]

(3)

with:
\[
[\Phi] = \begin{bmatrix} \Phi_{dqr1} & \Phi_{dqr2} & \Phi_{dqr} \end{bmatrix}^T, \quad [V] = \begin{bmatrix} V_{dqr1} & V_{dqr2} & V_{dqr} \end{bmatrix}^T
\]

the equation of the electromagnetic torque is:
\[
T_e = p \frac{L_m}{L_m + L_r} \left[ (I_{d1} + I_{d2}), \Phi_{d1} + (I_{a1} + I_{a2}), \Phi_{a2} \right]
\]

(4)

the equation of flux is:
\[
[\Phi_{dqm}] = L_m(I_{dqr1} + I_{dqr2} + I_{dqr})
\]

(5)

or:
\[
[\Phi_{dqm}] = L_a \left( \frac{\Phi_{dqr1}}{L_1} + \frac{\Phi_{dqr2}}{L_2} + \frac{\Phi_{dqr}}{L_r} \right)
\]

(6)

Where
\[
L_a = \frac{1}{\frac{1}{L_m} + \frac{1}{L_1} + \frac{1}{L_2} + \frac{1}{L_r}}
\]

(7)

the state matrix A and vector B in the d-q axis are:
\[
[A] = \begin{bmatrix}
    a_{11} & a_{12} & a_{13} & 0 & a_{15} & 0 \\
    a_{21} & a_{22} & 0 & a_{24} & a_{25} & 0 \\
    a_{31} & 0 & a_{33} & a_{34} & 0 & a_{36} \\
    0 & a_{42} & a_{43} & a_{44} & 0 & a_{46} \\
    a_{51} & a_{52} & 0 & 0 & a_{55} & a_{56} \\
    0 & 0 & a_{63} & a_{64} & a_{65} & a_{66}
\end{bmatrix}
\]

(8)

\[
[B] = \begin{bmatrix}
    1 & 0 & 0 & 0 & 0 & 0 \\
    0 & 1 & 0 & 0 & 0 & 0 \\
    0 & 0 & 1 & 0 & 0 & 0 \\
    0 & 0 & 0 & 1 & 0 & 0 \\
    0 & 0 & 0 & 0 & 1 & 0 \\
    0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]

(9)

where:
\[
a_{12} = a_{24} = -a_{31} = -a_{42} = \omega, \quad a_{15} = a_{35} = \frac{L_a}{T_1} \frac{T_1 L_r}{T_1 L_1} \\
a_{11} = a_{33} = \frac{L_a}{T_1 L_1} - \frac{1}{T_1}, \quad a_{11} = a_{43} = \frac{L_a}{T_2 L_2}, \quad a_{21} = a_{44} = \frac{L_a}{T_2 L_2} - \frac{1}{T_2}
\]
\[ a_{52} = a_{64} = \frac{L_a}{T_{2}T_r}, a_{55} = a_{66} = \frac{L_a}{T_{1}T_r}, \frac{1}{T_r} \]

\[ a_{32} = a_{46} = \frac{L_a}{T_{2}T_r}, a_{31} = a_{43} = \frac{L_a}{T_{1}T_r}, a_{56} = -a_{65} = \omega_r, T_1 = \frac{L_a}{R_s}, T_r = \frac{L_r}{R_r} \]

and

\[ \omega_r = \omega_a - \omega_m \]

### 3. MATRIX CONVERTER MODELING

In this section, it is explained the results of research and at the same time is given the comprehensive discussion. Results can be presented in figures, graphs, tables and others that make the reader understand easily [2], [5].

A matrix converter is a variable amplitude and frequency power supply that converts the three phase line voltage directly. It is very simple in structure and has powerful controllability. The real development of the matrix converter starts with the work of Venturini and Alesina who proposed a mathematical analysis and introduced the low frequency modulation matrix concept to describe the low frequency behavior of the matrix converter [1]. In this, the output voltages are obtained by multiplication of the modulation matrix or transfer matrix with the input voltages. The basic diagram of a matrix converter can be represented by Figure 2.

![Figure 2. Basic structure of matrix converter](image)

The existence function provides a mathematical expression for describing switching patterns. The existence function for a single switch assumes a value of unity when the switch is closed and zero when the switch is open. For the matrix converter shown in Figure 2, the existence function for each of the switches is expressed by the following equations:

\[ S_{kj} = \begin{cases} 1, & \text{switch } S_{kj} \text{ closed} \\ 0, & \text{switch } S_{kj} \text{ open} \end{cases} \quad (10) \]

where \( k = (A, B, C) \) is input phase and \( j = (a, b, c) \) is output phase.

The above constraint can be expressed in the following form:

\[ S_{kj} + S_{kj} + S_{kj} = 1 \quad (11) \]

with the above restrictions a 3 X 3 matrix converter has 27 possible switching states.

The mathematical expression that represents the operation of a three phase ac to ac Matrix Converter can be expressed as follows:
where $v_a$, $v_b$ and $v_c$ and $i_A$, $i_B$ and $i_C$ are the output voltages and input currents respectively. To determine the behavior of the MC at output frequencies well below the switching frequency, a modulation duty cycle can be defined for each switch. The modulation duty cycle $M_{kj}$ for the switch $S_{kj}$ in Figure 2 is defined as in equation (14) below.

$$M_{kj} = \frac{t_{kj}}{T_s}$$  \hspace{1cm} (14)

where $t_{kj}$ is the one time for the switch $S_{kj}$ between input phase $k=\{A, B, C\}$ and $j=\{a, b, c\}$ and $T_s$ is the period of the PWM switching signal or sampling period. In terms of the modulation duty cycle, equations 12, and 13 can be rewritten as given below.

$$\begin{bmatrix} v_a(t) \\ v_b(t) \\ v_c(t) \end{bmatrix} = \begin{bmatrix} S_{ab}(t) & S_{ba}(t) & S_{Ca}(t) \\ S_{ab}(t) & S_{ba}(t) & S_{Cb}(t) \\ S_{ab}(t) & S_{ba}(t) & S_{Cc}(t) \end{bmatrix} \begin{bmatrix} V_a(t) \\ V_b(t) \\ V_c(t) \end{bmatrix}$$ \hspace{1cm} (15)

$$\begin{bmatrix} i_a(t) \\ i_b(t) \\ i_c(t) \end{bmatrix} = \begin{bmatrix} S_{ab}(t) & S_{ba}(t) & S_{Ca}(t) \\ S_{ab}(t) & S_{ba}(t) & S_{Cb}(t) \\ S_{ab}(t) & S_{ba}(t) & S_{Cc}(t) \end{bmatrix}^T \begin{bmatrix} i_A(t) \\ i_B(t) \\ i_C(t) \end{bmatrix}$$ \hspace{1cm} (16)

The high-frequency synthesis technique introduced by Venturini and Alesina in [4-5] allows the use of low frequency continuous functions, referred to as the modulation matrix $m(t)$, to calculate the existence functions for each switch of the matrix converter. Thus, the aim when using the Alesina and Venturini modulation method is to find a modulation matrix which satisfies the following set of equations.

$$v_i(t) = m(t) \cdot v_i(t)$$ \hspace{1cm} (17)

$$i_i(t) = m(t)^T \cdot i_i(t)$$ \hspace{1cm} (18)

where the input voltages $v_i(t)$ are given by the following set of functions

$$v_i(t) = V_in \begin{bmatrix} \cos(\omega_ct) \\ \cos(\omega_ct - \frac{2\pi}{3}) \\ \cos(\omega_ct + \frac{2\pi}{3}) \end{bmatrix}$$ \hspace{1cm} (19)

and the desired output voltages $v_o(t)$ are
output currents \( i_o(t) \) can be expressed as:

\[
i_o(t) = I_o \begin{bmatrix}
\cos(\omega_o t + \phi_o) \\
\cos(\omega_o t - \frac{2\pi}{3} + \phi_o) \\
\cos(\omega_o t + \frac{2\pi}{3} + \phi_o)
\end{bmatrix}
\]

(21)

where \( \phi_o \) is the phase angle of the linear load.

finally, the desired input current has an arbitrary phase \( \phi_i \). This angle can be set to 0 to obtain unity input power factor of the matrix converter.

\[
i_i(t) = I_i \begin{bmatrix}
\cos(\omega_i t + \phi_i) \\
\cos(\omega_i t - \frac{2\pi}{3} + \phi_i) \\
\cos(\omega_i t + \frac{2\pi}{3} + \phi_i)
\end{bmatrix}
\]

(22)

The elements of matrix \( m(t) \) that satisfy equations 17 and 18 are given by

\[
m_o(t) = \frac{1}{2} \cdot \alpha_1 \left[ 1 + \left( \frac{1}{3} \right) \cos \left( \omega_o (i-j) + \frac{2}{3} \pi (2-i-j) \right) \right]
\]

\[
+ \frac{1}{3} \cdot \alpha_2 \left[ 1 + \left( \frac{1}{3} \right) \cos \left( \omega_o (i-j) + \frac{2}{3} \pi (2-i-j) \right) \right]
\]

(23)

where \( \alpha_1 = \frac{1}{2} \left[ 1 + \frac{\tan(\phi_i)}{\tan(\phi_o)} \right] \), \( \alpha_2 = 1 - \alpha_1 \), \( q = \frac{V_o}{V_i} \)

With the following restrictions \( \alpha_1 \geq 0 \), \( \alpha_2 \geq 0 \), \( 0 \leq q \leq \frac{1}{2} \)

4. SPEED CONTROL OF THE DSIM WITH NEURAL NETWORK

Feedforward artificial neural networks (ANN's) are universal approximators of nonlinear functions [7]. As such, the ANN's use a dense interconnection of neurons that correspond to computing nodes. Each node performs the multiplication of its input signals by constant weights, sums up the results, and maps the sum to a nonlinear function (activation function), the result is then transferred to its output. The mathematical model of a neuron is given by

\[
y = \phi(\sum w_i x_i - b)
\]

(24)

where \( x_1, x_2, ..., x_N \) are inputs from the previous layer neurons, \( w_1, w_2, ..., w_N \) are the corresponding weights, and \( b \) is the bias of the neuron.

For a logarithmic sigmoidal activation function, the output is given by...
A feedforward neural network is organized in layers: an input layer, one or more hidden layers, and an output layer. No computation is performed in the input layer: the signals are directly supplied to the first hidden layer. Hidden and output neurons generally have a sigmoidal activation function. The knowledge in an ANN is acquired through a learning algorithm, which performs the adaptation of weights of the network iteratively until the error between the target vectors and output of the network falls below a certain error goal. The most popular learning algorithm for multilayer networks is the backpropagation algorithm, which consists of a forward and backward action. In the first, the signals are propagated through the network layer by layer. An output vector is thus generated and subtracted from the desired output vector. The resultant error vector is propagated backward in the network and serves to adjust the weights in order to minimize the output error. The backpropagation training algorithm and its variants are implemented by many general-purpose software packages such as the neural-network toolbox from MATLAB. The structure of NN controller selected in this paper is shown in Figure 2. The NN controller consists of three neurons in the input layer, seven neurons in the hidden layer and a neuron in the output layer.

The three inputs signals \( e(k) \), \( e(k-1) \), \( isq1(k-1) \), and the torque \( (Tem^*(k)) \) output are exported to the MATLAB Workspace \((e(k) \) is the speed error and \( e(k-1) \) previous speed error\). The following MATLAB code trains the Neural Network. The first section of code generates the 'cell array'. The cell array combines the 3 different inputs into 1 input vector. The activation functions of the hidden and output neurons are Hyperbolic tangent sigmoid and linear, respectively. The learning of NN controller is done using the Levenberg-Marquardt back-propagation algorithm [7]. The training parameters for the Levenberg-Marquardt algorithm (trainlm) are:

- Maximum number of epochs to train \( (\text{net trainParam epochs}=400) \)
- Performance goal \((\text{net trainParam goal} = 1e-5) \)
- Epochs between displays \((\text{net trainParam show} = 5) \)

The off-line learning process of NN controller is shown in Figure 3. The data training is taken from the input and output values of the PI controller by simulating it under normal and disturbance conditions, (the fuzzy logic system is used on-line to generate the PI controller parameters), the learning rate were taken equal to \( 0.2 \). The electromagnetic torque from PI controller and the electromagnetic torque from NN controller are compared to obtain desired error goal [8,9].
5. SIMULATION RESULTS

The SIMULINK model for indirect FOC of the 4.5 Kw cage motor DSIM associated with adaptive FLC-PI controller is shown in Figure 4. The machine is fed by a matrix converter. The parameters of the induction motor are summarized in Appendix.

The first test concerns a no-load starting of the motor with a reference speed $\omega_{\text{ref}} = 288$ rad/sec. and a nominal load disturbance torque (14N.m) is suddenly applied between 1 sec and 2 sec, followed by a consign inversion (-288 rad/sec) at 2.5 sec. At 4.5 s, a -14Nm load disturbance is applied during a period of 2 s. This test has for object the study of controller behaviors in pursuit and in regulation.

The test results obtained are shown in figure 5. The speed of the motor reaches $\omega_{\text{ref}}$ at 0.2 s with almost no overshoot. It then begins to oscillate inside a 0.4% error strip around $\omega_{\text{ref}}$. The neural network controller rejects the load disturbance very quickly with no overshoot and with a negligible steady state error.

In order to test the robustness of the used method we have studied the effect of the parameters uncertainties on the performances of the speed control. To show the effect of the parameters uncertainties, we have simulated the system with different values of the parameter considered and compared to nominal value (real value). The Figure 6 and Figure 7 show respectively the behavior of the DSIM when $R_r$ is 10% increased of its nominal value and $J$ is increased and decreased 10% of its nominal value. An increase of the moment of inertia gives best performances, but it presents a slow dynamic response. The figures show that the proposed controller gave satisfactory performances thus judges that the controller is robust.

![Figure 3. Learning process of NN controller](image-url)

![Figure 5. Simulated results of neural network controller for DSIM](image-url)
6. CONCLUSION

In this paper a control strategy which incorporates the neural network for control of non linear system is described and used to demonstrate the effectiveness of the neural network for control of non linear system is described and used to demonstrate the effectiveness of the neural network for control the speed of dual star induction motor based on the indirect FOC. The machine is fed by a matrix converter. Simulation results show that the designed neural controller realizes a good dynamic behavior of the motor, with a rapid settling time, no overshoot, almost instantaneous rejection of load disturbance, a perfect speed tracking and it deals well with parameter variations of the motor. It seems to be a high-performance robust controller.
REFERENCES


