Research Article

Application of Artificial Intelligence Techniques for the Control of the Asynchronous Machine

F. Khammar\textsuperscript{1,2} and N. E. Debbache\textsuperscript{3}

\textsuperscript{1}Electrical Engineering Department, University of Badji Mokhtar of Annaba, P.O. Box 12, 23000 Annaba, Algeria

\textsuperscript{2}Laboratory of Electrical Engineering and Renewable Energy (LEER), University of Mohamed-Cheirf Messaadia-Souk Ahras, P.O. Box 1553, 41000 Souk Ahras, Algeria

\textsuperscript{3}Laboratory of Automatic and Signals-Annaba (LASA), University of Badji Mokhtar-Annaba, P.O. Box 12, 23000 Annaba, Algeria

Correspondence should be addressed to F. Khammar; fkhammar@ymail.com

Received 1 July 2015; Accepted 24 November 2015

Academic Editor: Sing Kiong Nguang

Copyright © 2016 F. Khammar and N. E. Debbache. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The induction machine is experiencing a growing success for two decades by gradually replacing the DC machines and synchronous in many industrial applications. This paper is devoted to the study of advanced methods applied to the command of the asynchronous machine in order to obtain a system of control of high performance. While the criteria for response time, overtaking, and static error can be assured by the techniques of conventional control, the criterion of robustness remains a challenge for researchers. This criterion can be satisfied only by applying advanced techniques of command. After mathematical modeling of the asynchronous machine, it defines the control strategies based on the orientation of the rotor flux. The results of the different simulation tests highlight the properties of robustness of algorithms proposed and suggested to compare the different control strategies.

1. Introduction

Asynchronous machines can currently achieve very high static and dynamic performance thanks to the use of advanced strategies of motor control. For high dynamic performance, one must be able to control torque instantaneously. In the case of the induction motor, the torque comes from a coupling between flux and current. The application of vector control constitutes a revolution for control of induction machines. This technique consists in reducing the behavior of the machine to that of a DC motor, by performing a decoupling between the flux and the electromagnetic torque. The main difficulty in implementing this type of control is the sensitivity to the uncertainties inherent to the system especially when it comes to external disturbances. The rotor resistance is the most difficult parameter to identify accurately [1–8]. The uncertainty in this parameter induces errors on the amplitude and direction of the flux in the machine, causing system instability and greater absorption of the stator current necessary for the given couple [1].

The asynchronous motor has a nonlinear behavior, its dynamic is quick, its parameters vary during operation. All these characteristics make the control of this machine complex. Consequently, the conservation of the nonlinear nature of the machine, the continuation of predetermined trajectories, the robustness to parameter variations, and the rejection of unknown disturbances with a powerful response are objectives to be satisfied during an implementation of a control strategy [9–11].

The artificial intelligence appears as an interesting alternative to control the asynchronous motor and to satisfy the desired requirements. The base element, in a fuzzy command is the fuzzy logic. The fuzzy logic belongs to the family of methods of artificial intelligence. She had a success not only in the modeling but also in the control of nonlinear complex systems and in several areas of electrical engineering [12]. The fuzzy models have the property to approximate any nonlinear function and the other advantage is that it is possible to do without an explicit model of the process. However, a fuzzy system is difficult to apprehend, its control
and its regulating may be relatively long, it is sometimes much more tentative step than a true reflection, then fuzzy logic requires a mean of powerful training to regulate a fuzzy system, and it is the neural networks [2]. The popularity of networks of neurons is due to the capacity to imitate the behavior of nonlinear systems, no parked, and robustness in comparison with disturbances. Most of the commands using a neural network as a controller are distinguished by an identification step and a step of control. The identification is to develop a neural model that is an estimate of the process to be controlled and that through a learning phase, that is to say, to propose a model establishing a relationship between its input and exit and determining from the pair of input-output signals the model behavior [13–15]. The command uses the knowledge acquired during the identification phase and/or online learning to develop the control signals. The joint use of neural networks and fuzzy logic allows taking the advantages of both methods. It allows exploiting the learning capability of neural networks on the one part and the reasoning capabilities of the fuzzy logic on the other part; different combinations of these two artificial intelligence techniques exist and put forward different properties [3, 5, 7].

In this work the objective is to know what could be the contribution of these methods applied to the identification and the control of the machine asynchronous for the training at variable speed. The paper is structured as follows: the first section displays the vector control of the asynchronous machine. In the second section, the machine is controlled according to two modes which are the fuzzy controller and the neural controller. Then a realization of the two techniques combined, that is, the neurofuzzy controller, is present. Finally, the paper concludes with a phase of tests and simulations to validate the performance and robustness of techniques used with at the end a comparative synthesis of the different control strategies presented.

2. The Vector Control by Orientation of the Rotor Flux

The vector control by orientation of the rotor flux has been developed to remove the internal coupling of the machine and return it to a linear control similar to that of a continuous current machine to separate excitation. For the realization of the vector control, there are two methods: the direct method and the indirect method. This study is based on the indirect method; this approach is not to use the amplitude of the rotor flux but only its position. It eliminates the need for a sensor or an estimator or an observer or stream but requires the use of a speed sensor.

The supply of the asynchronous motor is provided with a voltage inverter powered from a three-phase network. The set of transistors is the three-phase inverter pulse-width modulation (PWM), which imposes the frequency of the rotating field and the amplitude of the current in the machine. The technique of PWM command allows you to define the instants of command of switches. The main objective of this technique is to adjust the amplitude and frequency of the term fundamental and to reject the unwanted harmonics generated by a ripple “full wave” toward the high frequencies.

Equations of tensions and of fluxes of the stator and of the rotor of the vectorial model of the machine in a frame of reference (d, q) turning at a speed \( \omega_s \) in comparison with the stator are [4]

\[
\frac{di_{ds}}{dt} = \frac{1}{\sigma \cdot L_s} \left[ - \left( R_s + \frac{L_m^2}{L_r \cdot T_r} \right) \cdot i_{ds} + \sigma \cdot L_s \cdot \omega_r \cdot i_{qs} 
+ \frac{L_m}{L_r \cdot T_r} \cdot \theta_r + V_{ds} \right],
\]

\[
\frac{di_{qs}}{dt} = \frac{1}{\sigma \cdot L_s} \left[ - \omega_s \cdot \sigma \cdot L_s \cdot i_{ds} - \left( R_s + \frac{L_m^2}{L_r \cdot T_r} \right) \cdot i_{qs} 
- \frac{L_m}{L_r} \cdot \omega_r \cdot \theta_r + \frac{L_m}{L_r} \cdot \omega_r \cdot \theta_q + V_{qs} \right],
\]

\[
\frac{d\theta_r}{dt} = \frac{L_m}{T_r} \cdot i_{dr} - \frac{1}{T_r} \cdot \theta_r,
\]

\[
\frac{d\omega_r}{dt} = \frac{p^2 \cdot L_m}{f \cdot L_r} \cdot \left( \frac{i_{qs} \cdot \theta_{dr} - i_{ds} \cdot \theta_{qr} - F \cdot \omega_r - P \cdot C_r}{f} \right).
\]

If the rotor flux is orientated on the axis of a landmark linked to the field turning at the speed, then they have

\[
\theta_{dr} = \theta_r, \quad \theta_{qr} = 0.
\]
3. Artificial Intelligent Controller

Genetic algorithms, fuzzy logic, and neural networks are all techniques of digital computing in basis of artificial intelligence, which is popular in the area of informatics. But, increasingly, applications based on these new approaches to digital computing develop for practical applications in the fields of science and engineering.

The observers or the estimators based on the techniques of artificial intelligence lead a better dynamic and better accuracy and they are more robust. Their deliveries are very good even for large variations in the parameters of the machine. Nevertheless, the need for the perfect knowledge of the system to adjust or to estimate and the lack of expertise on system limits the current applications to a very specific range [6–8, 12–14].

3.1. Fuzzy Logic Controller. Fuzzy logic allows us to formulate mathematically imprecise concepts and to deduce precise actions from fuzzy rules coming from observations. The tool most commonly used in applications of fuzzy logics is the fuzzy rule base. A fuzzy rule has three stages: fuzzification, inference, and defuzzification.

The fuzzy controller consists of two input variables, the error ($e$) and error variation ($de/dt$), and an output variable. The error of the input variable is obtained from the difference between reference model with real speed of the rotor. The membership function of the determination of input and output variables is determined on the basis of experiments on the error of the system [13]. Resolved fuzzy rules are given in Table I. Defuzzification method most used and that of the determination are the center of gravity of the resulting membership function witch converts linguistic variables into specific variables. The abscises of the center of gravity becomes the exit of the regulator and therefore the order of the system, the process depends on the output of the fuzzy

Equations of the couple and of the speed of rotation can be described as follows:

$$T_e = \frac{3}{2} \cdot \frac{P \cdot L_m}{L_r} \cdot \theta_r \cdot i_q,$$

$$\omega_s = \frac{L_m}{L_r} \cdot R_r \cdot \dot{\theta}_{dr} \cdot i_q.$$

3.2. Artificial Neural Network. Neural modeling is chosen to bypass the parametric variations of the mathematical model of the motor. First, the neural network is driven offline; that is, the elements of the network are taken at the end of the application of all the data of the couples of training, for having a fixed network [11].

Then, the training of this one is done online for having an adaptation of the neural model in each moment. This type of training allows to conceive an adaptive order. The neural network can describe the behavior of a dynamic nonlinear system without need to know its parameters.

The neural network used for the command has a multi-layer structure with a hidden layer enabled by the sigmoid function, while the output layer is enabled by a linear function; its programming is carried by a backpropagation algorithm of the gradient with a learning rate adaptive [10]. The general principle of learning algorithms is based on the minimization of a cost function of quadratic of the differences between the outputs of the network and values desired [5–10, 12–14].
The choice of the number of neurons in hidden layer (ten) is determined from the tests to arrive at an acceptable error between the output of the network and expected value. The choice of this error is related to the desired approximation between the network and the system. The inputs of the neural network are the error of the speed (\(e = \Omega_{\text{ref}} - \Omega_r\)) and the derivative of the error (\(de/dt\)); the output is the torque \(C_{em}\).

To fulfill the order by model of networks of neuron it passes through three main steps: first, it is the loading of the two data files (.mat), one for the values of inputs and the other for the values of outputs (desired). After you load the inputs and outputs, it creates a network of neurons to three layers (input layer, output layer, and hidden layer), using the Matlab function "newff." Thus, the learning function "trainlm" was chosen among several functions, because it is the best, from the point of view convergence, speed, and accuracy. Then, it initializes the functions of activations of each layer, as well as the learning options, once the network of neuron is built and their learning has reached satisfactory performance, by using the function of Matlab "train," and simulates the results found with function "sim" of Matlab.

The architecture of this neural network is shown in Figure 2.

The structure of the neuronal control of speed of asynchronous machine is represented by Figure 3.

### 3.3. Adaptive Neurofuzzy Inference System (ANFIS)

The adaptive array ANFIS is a network architecture multilayer formed of a homogeneous combination of a neural network and fuzzy system. It is able to treat nonlinear and complex problems. The ANFIS can achieve optimal results quickly. It contains five layers (rule base, databases, a unit of decision-making, a fuzzification interface, and a defuzzification interface) as shown in Figure 4.

The first hidden layer “fuzzifier,” the variables of entries, and type operators T-norm shall calculate the part premise of rules in the second-hidden layer. The third-hidden layer normalizes the weights of the rules followed by the fourth-hidden layer or the parameters of the parties’ conclusion of rules are determined. The output layer calculates the sum of all the signals coming from the fourth layer [5]. The strengths of the ANFIS consist in the mechanism of distributional inference and the algorithm of adaptive learning. The rules are noninterpretable and the learning is offline.

The number of neurons in each layer is equal to the number of rules. There are two methods of learning: the learning hybrid method and the backpropagation learning method. The output variable is obtained by applying fuzzy rules to all input variables; it uses fuzzy rules derived from human expertise to model the dynamic behavior of the system [12, 14].

These rules are as follows:

Rule 1: “If \(x\) is \(A_1\) and \(y\) is \(B_1\), then \(f_1 = p_1 \cdot x + q_1 \cdot y + r_1\)”

Rule 2: “If \(x\) is \(A_2\) and \(y\) is \(B_2\), then \(f_2 = p_2 \cdot x + q_2 \cdot y + r_2\)”

\(x\) and \(y\) are input variables, \(A_1, A_2, B_1,\) and \(B_2\), of fuzzy sets, \(y_i\) is the outputs of all neurons defuzzification, and \(p_i, q_i,\) and \(r_i\) are parameters of the consequent of rule \(i\) determined during the process learning [12, 14].

The structure of ANFIS is shown in Figure 4.

The structure of a neurofuzzy controller is similar to a multilayer neural network. This is controller to an input layer, output layer, and three hidden layers which represent the membership function and the fuzzy rules.

### 4. Implementation

In order to highlight the benefits of intelligent approaches discussed above, simulations are performed under the same conditions. These simulations were performed with Matlab Simulink.

The simulations presented in the first phase are made based on a fuzzy controller. The structure adopted is that of Figure 1. The controller is of Mamdani type with a fuzzification Min-Max method and a defuzzification method of centroid type. Membership functions of fuzzy sets for the variables chose triangular and trapezoidal shape; gains standards are applied as input \((k_c\) and \(k_{de})\) and output \((k_c)\) to adjust the operation and vary the sensitivity range of the controller.
In the second phase, the fuzzy controller speed is replaced by a neural controller according to Figure 3. In this case, two operations occur; the first is the development of the model which means a learning phase of the algorithm retro error propagation; to fulfill the order by model of networks of neuron it passes through three main steps. First, it is the loading of the two data files (.mat), one for the values of inputs and the other for the values of outputs (desired). After you load the inputs and outputs, it creates a network of neurons to three layers (input layer, output layer, and hidden layer), using the Matlab function “newff,” Thus, it chose the learning function “trainlm,” among several functions, because it is the best, from the point of view convergence, speed, and accuracy. Then, it initializes the functions of activations of...
each layer, as well as the learning options, once the network of neuron is built and their learning has reached satisfactory performance; the second operation is the command that uses the information acquired during the learning phase to develop control signals.

In the last part of work, we apply Adaptive Neurofuzzy Inference System (ANFIS); it is a hybrid system using fuzzy inference of Takagi Sugeno. It is implemented in the toolbox “neurofuzzy” of Matlab. In this case, a hybrid learning rule which combines a gradient descent algorithm with an estimate by least squares is used. Retain the same configuration of Figure 3 but replace neural block by neurofuzzy controller.

5. Simulation Results and Interpretation

See Figures 5, 6, 7, and 8.

5.1. Validation of Models by Varying Parameters. See Figures 9, 10, and 11.

5.2. Interpretation of Simulation Results. The simulations are performed for a simulation time of \( t = 3 \) sec with the following:

Applying a torque-load \( c_r = 10 \) N-m for \( 1 < t < 1.5 \) sec. Inversion-rotation at time \( t = 2 \) sec.

5.3. Comparison of Results Obtained with the Techniques Applied. Figure 8 shows the shape of the speed for the three control techniques. It is found that the neurofuzzy controller is faster compared to neural and fuzzy controllers, the response time is faster even in the case of reversing the...
direction of rotation, and the neurofuzzy controller rapidly followed the desired value. Table 2 shows a summary of the results by type of control and parametric variations.

Table 2: Response time of the machine with three types of regulator during the variation parametric.

<table>
<thead>
<tr>
<th>Controller</th>
<th>Variation parametric</th>
<th>$T_r$ (sec) ($\omega_{ref} = 150$ rad/s)</th>
<th>$T_r$ (sec) ($\omega_{ref} = -150$ rad/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy</td>
<td>$R_r = R_{rm}$</td>
<td>0.17</td>
<td>2.275</td>
</tr>
<tr>
<td></td>
<td>$J = I_n$</td>
<td>0.2</td>
<td>2.323</td>
</tr>
<tr>
<td></td>
<td>$R_r = 1.5R_{rm}$</td>
<td>0.23</td>
<td>2.37</td>
</tr>
<tr>
<td></td>
<td>$R_r = -50%R_{rm}$</td>
<td>0.155</td>
<td>2.265</td>
</tr>
<tr>
<td>Neural</td>
<td>$R_r = R_{rm}$</td>
<td>0.223</td>
<td>2.41</td>
</tr>
<tr>
<td></td>
<td>$J = I_n$</td>
<td>0.28</td>
<td>2.585</td>
</tr>
<tr>
<td></td>
<td>$J = 1.5I_n$</td>
<td>0.3</td>
<td>2.615</td>
</tr>
<tr>
<td></td>
<td>$R_r = -50%R_{rm}$</td>
<td>0.195</td>
<td>2.35</td>
</tr>
<tr>
<td>ANFIS</td>
<td>$R_r = R_{rm}$</td>
<td>0.13</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td>$J = I_n$</td>
<td>0.1675</td>
<td>2.255</td>
</tr>
<tr>
<td></td>
<td>$J = 1.5I_n$</td>
<td>0.197</td>
<td>2.32</td>
</tr>
<tr>
<td></td>
<td>$R_r = -50%R_{rm}$</td>
<td>0.12</td>
<td>2.25</td>
</tr>
</tbody>
</table>

Figures 9, 10, and 11 show the responses of the system, speed, and flux in the case that the model of the machine is affected by the parametric variations. The results show that the performance of prosecution of speed and flux is very satisfactory with the neurofuzzy controller in comparing with the other two control laws according to these results. The changes are noticed in the shape of the flux during the inversion of the direction of rotation of the machine, but the regulators react and arrive to stabilize the flux at its desired value. In addition, we see from the speed graphs that the response time increases for $R_r = 1.5R_{rm}$, but for $R_r = -50\%R_{rm}$ we see an improvement in the time of stabilization,
which expresses the influence of parametric variation on the speed, but the neurofuzzy regulator remains the fastest, which confirms that the ANFIS control greatly reduces the sensitivity of the indirect vector control to uncertainties $R_r$ because the response speed is faster.

6. Conclusion

The asynchronous machine is a multivariable, not linear system, subjected to parametric variations and to unknown disturbances. Study accomplished in this job concerns order at variable speed of an asynchronous engine. Various algorithms of control were approached, that is, vector control, control by the techniques of artificial intelligence. This study reveals the motivation to realize research in the field of control by artificial intelligence techniques to the asynchronous motor drive.

Uncertainties in the rotor resistance, however, are the main problem, because it is the parameter most difficult to estimate with precision. By means of the study of sensitivity, it is possible to choose the strategy which strongly reduces...
the parametric sensitivity, while obtaining a satisfactory dynamics.

The different techniques proposed in this work to analyze and develop various controls contribute to refine the selection of a control strategy for a given application; it is advisable to examine experimentally the efficiency of the strategy ANFIS to confirm the results found by this study.

Appendix

The parameters of the machine are $U_n = 220$ V, $I_n = 3.64$ A, nominal speed $w_n = 154$ rd/s, and $f = 50$ Hz; 3-phase, 2 pairs of poles, $R_s = 4.85 \Omega$, $L_s = 0.274$ H, $R_r = 3.805 \Omega$, $L_r = 0.274$ H, $L_m = 0.258$ H, and $J = 0.031$ Kg\cdot m^2; coefficient of friction $K_f = 0.001136$ kg\cdot m^2/s.
Conflict of Interests
The authors declare that there is no conflict of interests regarding the publication of this paper.

References


