Speed Control of Vector Controlled Induction Motors with Fuzzy and Posicast Controller

Hamidreza Ghorbani, Siamak Masoudi and Vahid Hajiaghayi

Department of Electrical Engineering, Abhar Branch, Islamic Azad University, Abhar, Iran.

Abstract: The field orientation control of an induction motor has been permitted fast transient response by decoupled torque and flux. The conventional PI controller has been widely used in industrial application due to the simple control algorithm and easy implementation. With help of the Matlab/Simulink, block model of an induction motor drives can be constructed. This paper presents a novel fuzzy controller and POSICAST controller of an indirect field oriented induction motor drive for high performance. A superiority of the proposed fuzzy and POSICAST controller over conventional PI controller in handling nonlinear such as an induction motor has been effectively demonstrated by comparing speed controller with conventional PI controller under varying operating conditions like step change in speed reference and torque reference. The results validate the robustness and effectiveness of the fuzzy logic controller and POSICAST controller for high performance of induction motor drive system.

Key words: Induction Motor Drive, Vector Control, Fuzzy Logic Controller, POSICAST Controller.

INTRODUCTION

Alternating current motors are getting more and more popular for applications in industrial environments. Particularly in speed control systems, ac induction motors are more widely used nowadays due to the characteristics of higher efficiency, less inertia, smaller volume, and lower cost. Moreover, in contrast to dc motors, induction motors can be used for a long time without maintenance because of their brushless structures. The capabilities to operate at higher speeds, higher torques, and larger power ratings make the induction motors more attractive than dc motors for medium and high-power motor drives.

Open-loop control can be used in induction motor speed drives for simple applications. The major problem in open-loop speed control is that the machine is susceptible to speed and flux drifts. Moreover, speed errors will occur due to the disturbances. To overcome these weaknesses, closed-loop controls employing tachometer or encoder feedback are generally used in induction motor speed drives. Different control techniques and methodologies are currently available in literature concerning their applications to power systems. Some use classical and modern control techniques such as PID (Basilio, 2002; Negm, 2006; IEEE conference, 2003; Bounadja, 2007), while others, on the other hand, use control strategies based on computational intelligence methods such as fuzzy logic (Feng and Sheng). Despite having a lot of control-related researches for power systems, there are some still that use basic, non-sophisticated control techniques such as POSICAST control.

Using conventional PI controller it is very difficult and complex to design a high performance induction motor drive system. The fuzzy logic control (FLC) is an attractive approach which can accommodate the motor parametric variations and difficulty in obtaining an accurate mathematical model of induction motor due to rotor parametric and load time constant variations. The FLC is a knowledge-based control that uses fuzzy set theory and fuzzy logic for knowledge representation (L. A. Zadeh.). It has three main characteristics: (i) the FLC is a linguistic controller. it is not necessary to find a precise and accurate mathematical model of the controlled object. (ii) the FLC is a ideal flexible nonlinear type controller. it can overcome the influence of only non-linear variations. (iii) the FLC has strong robustness as it is not sensitive to parametric variations of the controlled process (Feng and Sheng.). Many FLC methods were presented in (R. Ouiguini; Lai, 1996; Liaw and Wang, 1991; Kukolj, 2001; Spiegel, 2003; Nounou and Rehman, 2007), that are suitable for speed control of induction motor drives.

In this paper, we describe the speed control strategies of a vector controlled induction motor. Next, with help of the Matlab/Simulink, we propose the conventional PI, fuzzy and POSICAST controller, which are used to speed control of induction motor drives and explain how they work. Finally we compare their simulation...
results together. The simulation results validate the robustness and reliable of the proposed POSICAST controller for high performance of induction motor drive.

**Vector Control of Induction Motor:**

Vector control is performed in different schemes (Profumo, et al., 1995) and (Santisteban, and Stephan, 2001). However, in all schemes, the machine torque and flux linkage are controlled through stator current vector control. The current vector is decomposed into a torque and flux producing components in a rotating reference frame, e.g. \( i_{ds} \) and \( i_{qs} \), respectively. The former component is along a machine flux linkage vector, and the latter component is perpendicular to the former as depicted in Fig. 1 in a rotor flux reference frame. This decouples the torque control from the flux control as the torque is obtained as;

\[
T_e = \frac{3}{2} n_p \frac{L_m}{L_r} \left( \psi_{dr} i_{qs} - \psi_{qr} i_{ds} \right)
\]  

(1)

where \( n_p \), \( L_m \) and \( L_r \) represent the number of pole pairs, the magnetizing inductance and the rotor inductance. Also, \( \psi_{qr} \) and \( \psi_{dr} \) stand for the quadratic axis and direct axis rotor flux linkage components.

**Fig. 1:** Principle of vector control.

Since the \( d \) axis flux vanishes, the torque equation is simplified to

\[
T_e = \frac{3}{2} n_p \frac{L_m}{L_r} \psi_{dr} i_{qs}
\]

(2)

where

\[
\psi_{dr} = L_m i_{ds}
\]

(3)

By taking \( i_{ds} \) = Constant, the torque linearly depends on \( i_{qs} \), providing a torque response as fast as the current \( (i_{qs}) \) response. The following stator and rotor voltage and flux linkage equations also hold for an induction machine.
\[ V_s = R_s i_s + \frac{d\psi_s}{dt} \]  (4)

\[ 0 = R_s i_s + \frac{d\psi_r}{dt} - j\omega_m \psi_r \]  (5)

\[ \psi_s = L_s i_s + L_m i_r \]  (6)

\[ \psi_r = L_m i_s + L_i i_r \]  (7)

Fig. 2 shows a block diagram of a typical vector controlled induction motor drive.

Conventional speed controller is based on a PI regulator, shown in Fig. 3. The output of this regulator is a torque set point applied to the vector control block.

\[ \Delta e(k) = \omega_{ref} - \omega_r \]  (8)

\[ \Delta e(k) = e(k) - e(k-1) \]  (9)

where, \( \omega_{ref} \) is the reference speed and \( \omega_r \) is the actual rotor speed. The reference current \( i_{ref}^* (k) \), that is applied to the vector control system, is calculated as:

Fig. 3: speed controller block with PI controller.

where, \( N' \) and \( N \) are motor reference and actual speed.

**Fuzzy Logic Controller in Induction Motor:**

Different schemes of Fuzzy logic control methods are used widely in induction motor control (R. Ouiguini; Lai, 1996; Liaw and Wang, 1991; Kukolj, 2001; Spiegel, 2003; Nounou and Rehman, 2007) The Fuzzy Logic Toolbox based controller architecture is shown in Fig. 4.

Fig. 4: Fuzzy Controller Architecture.

The design and synthesis of a fuzzy controller will be presented using fuzzy logic toolbox with Matlab/Simulink. In this paper, fuzzy logic controller employs speed error and change of speed error as inputs, the changes in torque component of current that drives the induction motor is output.
\begin{equation}
\dot{i}_{qs}^e(k) = \dot{i}_{qs}^e(k-1) + C_i^e(k)
\end{equation}

In the first stage, the crisp variables \(e(K)\) and \(Ae(K)\) are converted into fuzzy variables. The fuzzification maps the error, and the error changes to linguistic labels of the fuzzy sets. The proposed controller uses following linguistic labels: \{NB(Negative Big), NM(Negative Medium), NS(Negative Small), ZE(Zero), PS(Positive Small), PM(Positive Medium), PB(Positive Big)\}. Each fuzzy label has an associated membership function. The membership functions of triangular type as shown in Fig.5.

Knowledge base involves defining the rules represented as statements governing the relationship between input and output variables in term of membership functions. The control rules are represented as a set if then rules. The fuzzy rules of proposed controller for speed control of induction motor are presented in Table1.

![Fig. 5: Membership Function of Input, Output variables.](image)

<table>
<thead>
<tr>
<th>Error change</th>
<th>error</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB</td>
<td>NB</td>
</tr>
<tr>
<td>NM</td>
<td>NB</td>
</tr>
<tr>
<td>NS</td>
<td>NB</td>
</tr>
<tr>
<td>ZE</td>
<td>NB</td>
</tr>
<tr>
<td>PS</td>
<td>NM</td>
</tr>
<tr>
<td>PM</td>
<td>NS</td>
</tr>
<tr>
<td>PB</td>
<td>ZE</td>
</tr>
</tbody>
</table>

The present paper uses Mamdani's Max-Min algorithm for inference mechanism.

**Speed Control with Posicast Controller:**

**Posicast Controller:**

In the late 1950’s, POSICAST is a feedforward control method that dampens oscillations in systems whose other transient specifications are otherwise acceptable. When properly tuned, the controlled system yields a transient response that has deadbeat nature. Consider a system having a lightly damped step response as shown in Fig. 6(a). The overshoot in the response can be described by two parameters (Sugiki and Furuta, 2006). First, the time to the first peak is one half the under-damped response period \(T_d\). Second, the peak value is described by \(1+d\), where \(d\) is the normalized overshoot, which ranges from zero to one. Zero overshoot corresponds to critical damping. POSICAST splits the original step input command into two parts, as illustrated in Fig. 6(b). The first part is a scaled step that causes the first peak of the oscillatory response to precisely meet the desired final value. The second part of the reshaped input is full scale and time-delayed to precisely cancel the remaining oscillatory response, thus causing the system output to stay at the desired value. Such is the idea behind “half-cycle POSICAST,” which can be modeled using just the two parameters \(d\) and \(T_d\). The resulting system output is sketched in Fig. 6(c).

One block diagram interpretation of the half-cycle POSICAST controller is shown in Fig. 7(a). The model has two forward paths. The upper path is that of the original, uncompensated command input. In the lower path, a portion of the original command is initially subtracted, so that the peak of the response will not overshoot the desired final value. Precisely a half cycle later, the command is fully restored to cancel oscillations and maintain the final value. The transfer function is given by the function \(1+P(s)\), where \(P(s)\) is given by:

\[P(s) = \frac{d}{1+d}\left(-1+\exp\left(-\frac{T_d}{2}\right)s\right)\]

\[P(s) = \frac{d}{1+d}\left(-1+\exp\left(-\frac{T_d}{2}\right)s\right)\]
a. A lightly damped transient response

b. POSICAST command

c. System output

Fig. 6: Natural response, POSICAST command and resulting output.
Classical POSICAST generally suffers from sensitivity to modeling errors. The sensitivity problem can be reduced if POSICAST compensation is applied within a feedback system rather than in the classical feedforward configuration (Hung, 2007) and (Hung, 2000). A block diagram explaining the control method is shown in Fig. 7(b). Whereas the classical applications placed POSICAST before the lightly damped system, recent work suggests that POSICAST be used within a feedback system. The proposed control method is a significant departure from classical POSICAST.

**POSICAST Controller in Induction Motor:**

Since the aim is controlling the induction motor reference torque, $I+P(s)$ and $C(s)$ are being put in the path of the $(N-N^*)$ and $Te^*$ like Fig. 8. For $Te^*$, $T_d$ and $d$ parameters are counted and $P(s)$ is being formed.

**Comparative Evaluation:**

The proposed control system is applied to the induction motor with parameters as in the Appendix. The motor performance is then simulated, and the results are compared with those obtained by the motor performance with PI, FLC and POSICAST speed controller. Fig. 9 and 10, shows the efficiency of POSICAST control method in comparison with usual PID and FL controllers.

**Conclusion:**

Invented 50 years ago, POSICAST was originally designed as a feedforward compensator for lightly damped systems. The first reported applications were for mechanical structures. The technique is also closely related to input pre-shaping control, which has been widely studied and reported in the robotics research.
community. More recently, POSICAST has been proposed for use within feedback loops, to take advantage of its superior damping qualities while also reducing POSICAST’s sensitivity to modelling error. Three methods for speed control of a vector controlled induction motor have been proposed in this paper. A comparison between the conventional PI, fuzzy logic and POSICAST controller reveals the superiority of the third one.

Fig. 10: speed response of motor with step variation of load torque.

Appendix:
The parameters of the simulated induction motor are presented in Table 2.

<table>
<thead>
<tr>
<th>Table 2: Simulated motor parameters.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated power</td>
</tr>
<tr>
<td>Rated voltage (line-line)</td>
</tr>
<tr>
<td>Pair of poles</td>
</tr>
<tr>
<td>Stator resistance</td>
</tr>
<tr>
<td>Rotor resistance</td>
</tr>
<tr>
<td>Stator leakage inductance</td>
</tr>
<tr>
<td>Rotor leakage inductance</td>
</tr>
<tr>
<td>Magnetizing inductance</td>
</tr>
</tbody>
</table>

REFERENCES


