International Journal of Soft Computing 2 (3): 393-400, 2007

Fuzzy Logic Controller of Sensorless Indirect Field-Oriented Induction Machine Drive

[1]M.N. Noaman, [2]J.H. Amjad and [2]R.A. Ahmmad
[1]Department of Computer Engineering, Compuetr Man College for Computer Studies, Khartoum, Sudan
[2]Department of Electrical Engineering AL-Rashed College, University of Technology, Baghdad, Iraq

Abstract: Field orientation control techniques of induction machine have permitted fast transient response by decoupled torque and flux control. Among these techniques, Indirect Field Oriented Control (IFOC) strategy is adopted as an effective one. The results showed that the technique can keep the rotor flux constant even during changes in load torque. This indicates that decoupling control of flux and torque has been obtained. However, field orientation detuning caused by parameter variation is a major difficulty for indirect field orientation control method. The decoupling that IFOC could perform is conditioned by the accuracy of slip calculation. The slip calculation depends on the rotor time constant, which varies continuously according to the operational conditions. A Fuzzy Logic Control (FLC) provides a systematic method to incorporate human experience and implement nonlinear algorithms, characterized by a series of linguistic statements, into the controller. Results of computer simulation showed that FL controller has potential to improve the closed-loop control performance and can outperform conventional PI controllers. An FLC scheme has been designed for an IFO induction machine drive system. Good performance of the IFO drive system is obtained in terms of overshoot, steady-state error, load disturbance rejection and variable speed tracking. The speed measurements are undesirable in a drive because they add cost and reliability problems, besides the need for a shaft extension and mounting arrangements. Therefore, this study has presented a state estimation technique for speed sensorless FOC of induction motors. A stochastically nonlinear state estimator, Extended Kalman Filter (EKF) is suggested for this purpose. Using this observer, the rotor speed and rotor fluxes are estimated simultaneously. The motor model designed for EKF application involves rotor speed, dq-axis stator currents. A number of simulations were carried out to verify the performance of EKF observer. The performance is tested under variable speed tracking, loaded conditions. Also in the implementation of the EKF different of measurement and state noise covariance matrices may be tried to detect the optimum research which increase performance of the EKF.

Key words: FLC, sensorless, EKF, optimum, speed tracking, IFOC

INTRODUCTION

Induction Machines models (IM) are often ill defined. Even if the machine model is well known, there may be a parameter variation problem. The d-q machine model of IM is multivariable, complex and nonlinear. Vector or Field-Oriented Control (FOC) of a drive can overcome this problem, but accurate vector control is nearly impossible and there may be a wide parameter variation problem in the system (Bimal, 2002).

To compensate such parameter variations in the drive, a Proportional Integral PI controller is firstly candidated in the outer speed loop of FOC drive to generate a command current, directly proportional to the required torque. However, the fixed parameter controller is not an intelligent controller and changes in rotor time constant will degrade the speed performance (Denai and Attia, 2002; Zhen and Longya, 2000; Heber *et al*., 1997).

Fuzzy Logic Control (FLC) does not strictly need any mathematical model of the plant and provides a systematic method to incorporate human experience and implement nonlinear algorithms, characterized by a series of linguistic statements, into the controller. Moreover, fuzzy controller is easy to apply and possibly the best adaptive control among the techniques discussed earlier (Zhen and Longya, 2000; Heber *et al*., 1997). In this study, 2 control strategies are considered to adjust the speed of the drive system: PI and FL controller.

Corresponding Author: M.N. Noaman, Department of Computer Engineering, Compuetr Man College for Computer Studies, Khartoum, Sudan

The robustness of these suggested controllers is checked in terms of motor parameter variations (Heber *et al.*, 1997; Cheng and Yeh, 1993).

Controlled induction motor drives without mechanical speed sensors at the motor shaft have the attractions of low cost and high reliability. To replace the sensor, the information on the rotor speed is extracted from measured stator voltages and currents at the motor terminals (Barambones *et al.*, 2002; Denai and Attia, 2002; Bilal, 2003). Therefore, the speed controller is performed under no mechanical speed sensors and speed observer, based on Extended Kalman Filter (EKF), is adopted for this purpose.

DYNAMIC MODEL OF INDUCTION MACHINE

Induction Machine (IM) equations in arbitrary rotating reference frame can be represented in stator and rotor dq voltage equations (Bimal, 2002; Chee, 1998; Leonhard, 1998).

$$\begin{aligned} v_{qs} &= p\lambda_{qs} + \omega\lambda_{ds} + r_s i_{qs} \\ v_{ds} &= p\lambda_{ds} - \omega\lambda_{qs} + r_s i_{qs} \\ v'_{qr} &= p\lambda'_{qr} + (\omega - \omega_r)\lambda'_{dr} + r'_r i'_{qr} \\ v'_{dr} &= p\lambda'_{dr} - (\omega - \omega_r)\lambda'_{qr} + r'_r i'_{qr} \end{aligned} \quad (1)$$

Where v is voltage; λ is the flux linkage; *i* is the current; ω is the arbitrary speed of the reference frame; r is the resistance and p is the time derivative. The subscript r and s denotes the rotor and stator values, respectively referred to the stator and the subscripts d and q denote the dq-axis components in the arbitrary reference frame.

The equations of the machine in the stationary and synchronously rotating reference frame can be obtained from (1) by setting ω to zero and $\omega = \omega_e$, respectively. To distinguish these 2 frames from each other, an additional superscript will be used; s for stationary frame variables and e for synchronously rotating frame variables. The electromagnetic torque equation can be given by Bimal (2002), Chee (1998) and Dal (2001).

$$T_{em} = \frac{3}{2}\frac{P}{2}(\lambda'_{qr} i'_{dr} - \lambda'_{dr} i'_{qr}) \quad (2)$$

where P denotes the number of machine pole pairs. Using Eq. 1 and 2, one can obtain the state-space model for induction motor developed in stationary reference frame as given below (Bimal, 2002; Bilal, 2003).

$$\begin{bmatrix} i^s_{ds} \\ i^s_{qs} \\ \lambda'^s_{dr} \\ \lambda'^s_{qr} \end{bmatrix} = \begin{bmatrix} -\frac{K_R}{K_L} & 0 & \frac{L_m r'_r}{L'^2_r K_L} & \frac{L_m \omega_r}{L_r K_L} \\ 0 & -\frac{K_R}{K_L} & -\frac{L_m \omega_r}{L'_r K_L} & \frac{L_m r'_r}{L'^2_r K_L} \\ \frac{L_m}{T_r} & 0 & -\frac{1}{T_r} & -\omega_r \\ 0 & \frac{L_m}{T_r} & \omega_r & -\frac{1}{T_r} \end{bmatrix} \begin{bmatrix} i^s_{ds} \\ i^s_{qs} \\ \lambda^s_{dr} \\ \lambda^s_{qr} \end{bmatrix} + \frac{1}{K_L} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V^s_{ds} \\ V^s_{qs} \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} i^s_{ds} \\ i^s_{qs} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} i^s_{ds} \\ i^s_{qs} \\ \lambda'^s_{dr} \\ \lambda'^s_{qr} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V^s_{ds} \\ V^s_{qs} \end{bmatrix} \quad (4)$$

Where $K_L = (L_r L_s - L^2_m)/L'_r$, $K_R = r_s + r'_r (L_m/L'_r)^2$ and $K_M = (3P^2 L_m)/(8JL'_r)$. The parameters L'_r, L'_s, L_m are rotor, stator and main inductances, respectively. $T_r = L_r/r'_r$ is the rotor time constant and is the rotor electrical speed in angular frequency.

INDIRECT FIELD ORIENTATION CONTROL (IFOC)

Figure (1) explains the fundamental principle of indirect vector control with the help of a phasor diagram. The d^s-q^s axes are fixed on the stator, but the d-q axes, which are fixed on the rotor, are moving at speed ω_r. Synchronously rotating d^e-q^e axes are rotating ahead of the d^r-q^r axes by the positive slip angle θ_{sl} corresponding to slip frequency ω_{sl}.

Since the rotor pole is directed on the d^e axis and $\omega_e = \omega_r + \omega_{sl}$, one can write

$$\theta_e = \int \omega_e \, dt = \int (\omega_r + \omega_{sl}) \, dt = \theta_r + \theta_{sl} \quad (5)$$

The phasor diagram suggests that for decoupling control, the stator flux component of current i^e_{ds} should be aligned on the d^e axis and the torque component of current should be on the axis, as shown. For decoupling control, one can make a derivation of control equations of indirect vector control with the help of d^e-q^e dynamic model of IM, i.e., using Eq. (1) with the addition of superscript eto the variables and setting $\omega = \omega_e$. If d^e axis aligned with the rotor field, the q-component of the rotor field, λ'^e_{qr}, in the chosen reference frame would be zero. One can easily shows the following important equations (Bimal, 2002; Chee, 1998; Dal, 2001).

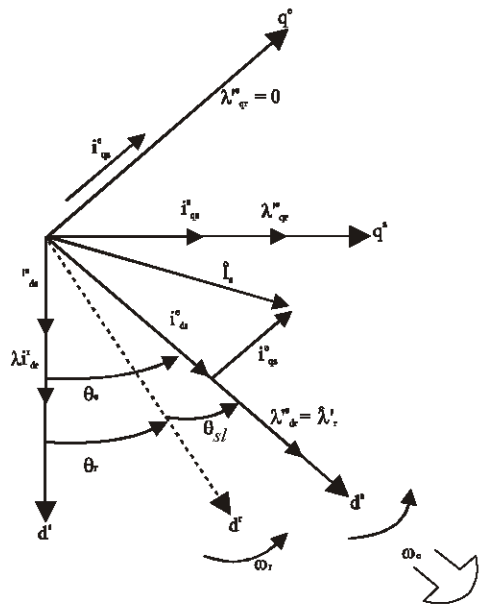

Fig. 1: Phasor diagram explaining indirect vector control

$$T_{em} = \frac{3}{2}\frac{P}{2}\frac{L_m}{L_r}\lambda'^{e}_{dr} i^{e}_{qs} \tag{6}$$

$$\lambda'^{e}_{dr} = \frac{r'_r L_m}{r'_r + L'_r p} i^{e}_{ds} \tag{7}$$

$$\omega^{e}_{sl} = \omega_e - \omega_r = \frac{r'_r}{L'_r}\frac{i^{e}_{qs}}{i^{e}_{ds}} \tag{8}$$

To implement the indirect vector control strategy, it is necessary to use the condition in Eq. 6-8 in order to satisfy the condition for proper orientation. Figure 2 shows an indirect field-oriented control scheme for a current controlled PWM induction machine motor drive.

The command values for the abc stator currents can then be computed as follows

$$\begin{aligned} i^{s*}_{qs} &= i^{e}_{qs}\cos\theta_e + i^{e}_{ds}\sin\theta_e \\ i^{s*}_{ds} &= -i^{e}_{qs}\sin\theta_e + i^{e}_{ds}\cos\theta_e \end{aligned} \tag{9}$$

$$\begin{aligned} i^{*}_{as} &= i^{s*}_{qs} \\ i^{*}_{bs} &= -(1/2)i^{s*}_{qs} - (\sqrt{3}/2)i^{s*}_{ds} \\ i^{*}_{cs} &= -(1/2)i^{s*}_{qs} + (\sqrt{3}/2)i^{s*}_{ds} \end{aligned} \tag{10}$$

FUZZY LOGIC CONTROL OF AN INDIRECT FIELD-ORIENTED INDUCTION MACHINE

The block diagram showing the implementation of the FL controller is illustrated in Fig. 2. The actual inputs to the fuzzy controller are, e_1 and, e^2 which are a scaled version of the speed error (e_i) and the change in speed error (Δe_i). The gains, G_1 and, G_2 can be varied so that there is no saturation on its input universe of discourse. The output gain, G_0, can be tuned to reach the desired performance (Bimal, 2002; Heber *et al*., 1997).

The fuzzy controller observes the pattern of the speed loop error signal and correspondingly updates the output ΔT^*e_m so that the actual speed ω_r matches the command ω^*_r. The output of the fuzzy controller is integrated and fed to command the IFOC IM.

The rules of FL controller are designed to take full advantage of the decoupling of torque and flux such that the actual speed can reach the command speed as quickly as possible without overshoot. The general considerations in the design of the controller are (Bimal, 2002; Cheng and Yeh, 1993).

If both and are zero, then maintain the present control setting $\Delta T^*e_m = 0$.

If e_1 is not zero but approaching this value at a satisfactory rate, then maintain the present control setting.

If e_1 is growing, then the change of $\Delta T^* e_m$ (0,0) will depend on the magnitude and sign e_1 of and e_2 to force e_1 towards zero.

The center of the Table 1, ΔT^*e_m, represents that when the error and change of error are zeros, $\Delta T^*e_m = 0$. Moving away from the center, in any direction, causes the FL controller to increment or decrement T^*e_m. The following rule

IF e_1 is NB and e_2 is PB then ΔT^*e_m is PS

applies for the case when ω_r is much less than ω^*_r and is accelerating quickly toward, ω^*_r, where the linguistic variables PS, PB and NB denote Positive Small, Positive Big and Negative Big, respectively. Most FLs would conclude that ΔT^*e_m should be 0. However, with the IFOC IM application, FL controller decides to slightly increase the torque causing the machine to accelerate even faster.

The brakes are applied when the linguistic value of error becomes PM by the rule

IF e_1 is NM and e_2 is PB then ΔT^*e_m is NB

The rules near the center are also different from a typical fuzzy controller. In particular, the rules (e_1 = "PS" and e_2 = "NS") and (e_1 = "NS" and e_2 = "PS") would have a zero for the output with a typical FL control. However with this application, this rule base outputs a "PS" or a "NS"

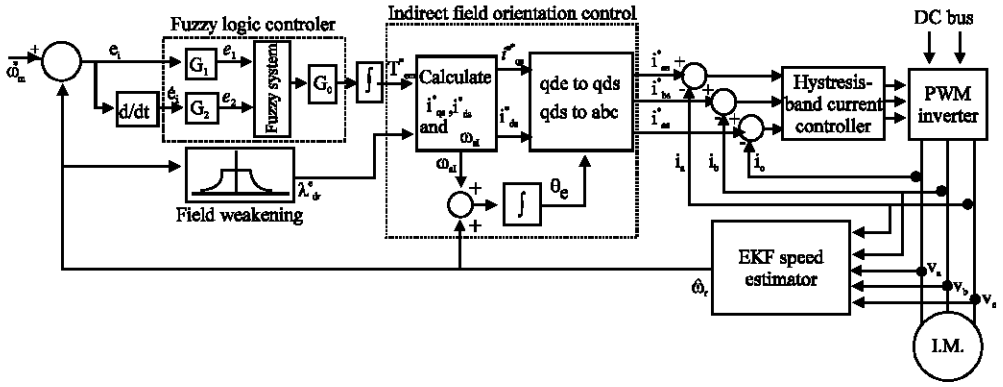

Fig. 2: Block diagram of the FLC for an IFO controlled induction machine drivewith current regulated pwm inverter

Table 1: Knowledge-base array for drive fuzzy model

	e_1						
Δt^*_{em}	NB	NM	NS	ZE	PS	PM	PB
NB	PB	PB	PB	PB	PB	PB	NS
NM	PB	PB	PM	PM	PS	ZE	NM
NS	PB	PM	PM	PS	NS	NM	NM
ZE	PM	PM	PM	ZE	NM	NM	NM
PS	PM	PM	PS	NS	NM	NM	NB
PM	PM	ZE	NS	NM	NM	NB	NB
PB	PS	NB	NB	NB	NB	NB	NB

respectively; a valid justification, as this will change the current command just enough to drive the error to zero faster than a zero output would. Moreover, if any disturbance occurs, the rules near the center quickly change the current to keep the speed at the reference speed. Consequently, these rules near the center reduce the error more effectively and then improving the steady-state performance. Following the above reasoning for the other possible situations, the knowledge base of direct FL controller can be stored in Table 1.

Thus, the rule base of direct FL controller is designed to have large changes in the current command when the error and/or the change of error are large. This makes a better use of the torque capabilities of the FOC drive.

A fuzzy set is defined by assigning the grade of membership values to each element of the universe of discourse. There are many types of Membership Functions (MF), e.g., the bell-shaped, the triangular-shaped, the trapezoidal-shaped, etc. For simplicity, uniformly distributed the triangular-shaped MFs are used for FL controller.

The defuzzification strategy is aimed at producing a nonfuzzy control action that best represents the possibility distribution of an inferred fuzzy control action. Many strategies can be used for performing the defuzzification. The Center-of-Gravity method (COG) is adopted in this study.

SPEED ESTIMATION USING EKF

The standard KF is a recursive state estimator capable of use on a multi-input/output system with noisy measurement data and process noise (stochastic plant models). It uses the plant's inputs and output measurements, together with a state space model of the system, to give optimal estimates of system states (Bilal, 2003; Wade *et al*., 1997).

To estimate the rotor speed, it must be treated as a state and a nonlinear model is formed with the states consisting of the parameter to be estimated and the original states (Wade *et al*., 1997; Ouhrouche, 2000). The new model is formed after augmented with the new state and discretized to give

$$X(k+1) = f(x, u, k) + V(k) \qquad (11)$$
$$Y(k) = CX(k)+W(k)$$

where,

$$X(k) = \left[i^s_{ds}(k)\, i^s_{qs}(k)\ \lambda^s_{dr}(k)\, \lambda^s_{qr}(k)\, \omega_r \right]^T$$

is the combined state and parameter matrix, f (x, u, k) is the nonlinear state function. V (k) and W (k) are zero-mean, white Gaussian noise vectors of X(k) and Y(k), respectively.

$f(x, u, k) = A_dX(k)+B_dU(k) = (12)$

$$\begin{bmatrix} a_{11}x_1(k) + a_{13}x_3(k) + a_{14}x_5(k)x_4(k) + bu_1(k) \\ a_{11}x_2(k) - a_{14}x_5(k)x_3(k) + a_{13}x_4(k) + bu_2(k) \\ a_{31}x_1(k) + a_{33}x_3(k) + a_{34}x_5(k)x_4(k) \\ a_{31}x_2(k) - a_{34}x_3(k)x_5(k) + a_{33}x_4(k) \\ x_5(k) \end{bmatrix}$$

where,

$$a_{11} = 1 - \frac{K_R}{K_L}T_s,\ a_{13} = \frac{L_m r'_r}{L_r^2 K_L}T_s,\ a_{14} = \frac{L_m}{L_r K_L}T_s$$

$$a_{31} = \frac{L_m}{T_r} T_s,\ a_{33} = 1 - \frac{T_s}{T_{r\,s}},\ a_{34} = -T_s$$

Also, A_d and B_d matrices are discretized system and input matrices, respectively. T_s is the sampling time

$$B_d = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, C_d = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}, U(k) = \begin{bmatrix} V_{ds}^{s}(k) \\ V_{qs}^{s}(k) \end{bmatrix} \quad (13)$$

To use a nonlinear model with the standard KF, the model must be linearized about the current operating point, giving a linear perturbation model

$$F(x, u, k) = \frac{\partial f(x,u,k)}{\partial x}\bigg|_{\hat{x}(k)+u(k)} =$$

$$\begin{bmatrix} a_{11} & 0 & a_{13} & a_{14}x_5(k) & a_{14}x_4(k) \\ 0 & a_{11} & -a_{14}x_5(k) & a_{13} & a_{14}x_3(k) \\ a_{31} & 0 & a_{33} & a_{34}x_5(k) & a_{34}x_4(k) \\ 0 & a_{31} & -a_{34}x_{34}(k) & a_{33} & -a_{34}x_3(k) \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

Figure 3 shows the block diagram of the EKF algorithm. The EKF algorithm uses the full machine dynamic model, where the speed ω_r is considered a parameter as well as a state. Both V (k) and W (k) are independent of X(k) and Y(k), respectively. The statistics of noise and measurements are given by 3 covariance matrices, Q, R and P, where Q = system noise vector covariance matrix (5×5), R = measurement noise vector covariance matrix (2×2) and P = system state vector covariance matrix (5×5).

The sequence of the EKF algorithm implementation by a flow diagram is shown in Fig. 4, which also includes the basic computational expressions. Basically, it has 2 main stages: Prediction stage and filtering stage. In prediction stage, the next predicted values of states X^* (k+1) are calculated by the machine model and the previous values of estimated states. In addition, the predicted state covariance matrix P^* (k+1) is also calculated using the covariance vector Q. In the filtering stage, the next estimated states $\hat{X}^*(k+1)$ are obtained from the predicted states X^* (k+1) by adding the correction term e_k where e = Y (k+1)-$\hat{Y}$ (k+1) and K = Kalman gain. The Kalman gain is optimized for the state estimation errors. The EKF computations are done in recursive manner so that e approaches 0.

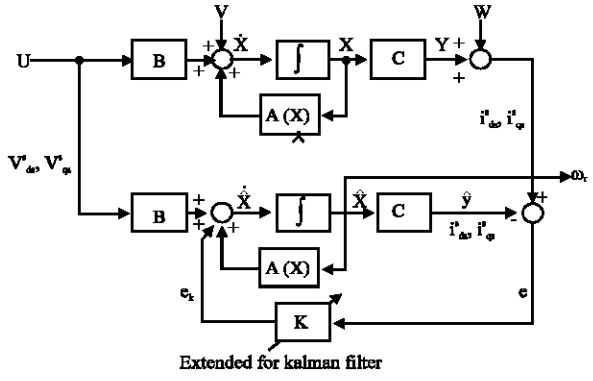

Fig. 3: EKF for estimation of speed

RESULTS AND DISCUSSION

In the simulation, a 20 Hp motor is used, whose parameters are listed in Table 2. In the first part of simulation, the state estimation performance of EKF is assessed. The simulation of Fig. 1 is implemented with Simulink. In this simulation, input voltages and measured currents in stationary reference frame are produced by IFOC block. The EKF algorithm Is developed as a S-function and then inserted to Simulink in the form of S-function block.

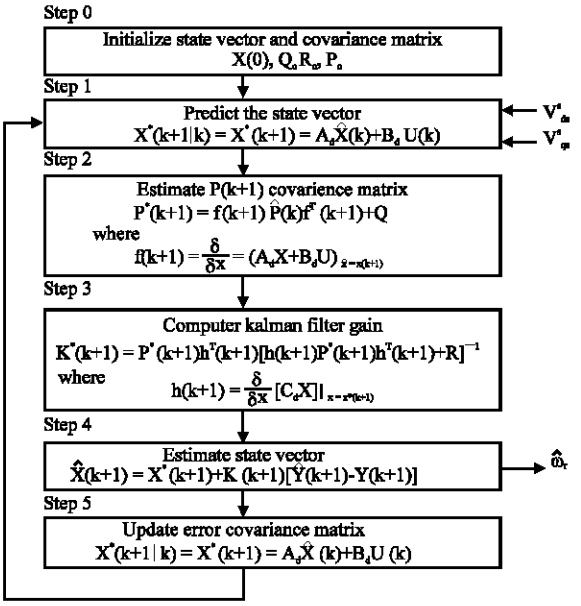

Fig. 4: EKF algorithm flow diagram

Table 2: Induction motor parameter

Rated power	20 hp
Rated line-line voltage	200 V
Rate torque	81.5 Nm
Number of poles (P)	4
Stator resistans (rs)	0.106 Ω
Stator inductance (L_m)	8.67 mH
Rotor resistance (r_s)	0.076 Ω
Rotor inductance (L_s)	9.15 mH
Moment of inertia (J)	2.5 kg m^2
Base excitation frequency (f)	60 Hz

In Fig. 5 speed reversal at no-load is given with reference speed. The estimated speed and the reference speed are plotted together. The estimated speed near steady-state is shown in Fig. 6. The figure illustrates that the estimated speed does not overlap with reference speed at steady state.

A zero steady state error could be reached by using different settings of measurement and state covariance matrices, but this leads to poor transient speed estimation. In the case of Fig. 7 simulation, state covariance is decreased; the algorithm begins to behave such that the state space model gives more accurate estimates compared to measure values so it assigns less importance to the measurements. This causes a decrease in Kalman gain, which reduces the correction speed of the currents. In the extra time used for current correction the algorithm finds opportunity to decrease the steady-state error.

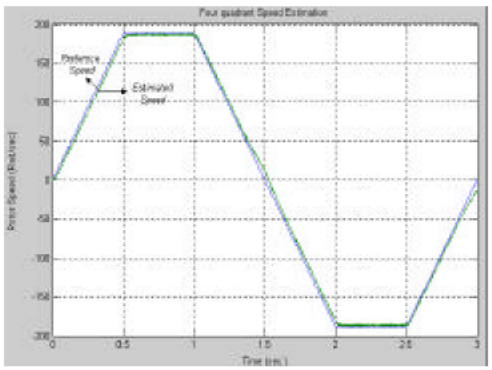

Fig. 5: High Speed, no-load, four quadrant speed estimation with EKF

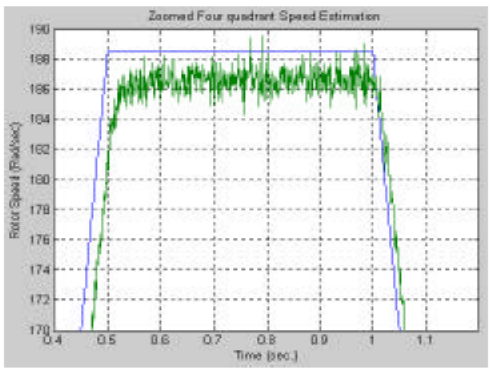

Fig. 6: (Zoomed at steady state) high speed, No-load, four quadrant speed estimation at steady state

Low speed estimation performance of the EKF is also quite satisfactory and close to reference speed as shown in Fig. 8 and 9.

To verify the performance of EKF under loaded conditions, rated mechanical load is applied to the motor between 0.75-1.5 sec as shown in Fig. (10). The EKF estimator works properly even under fully loaded case. One may decrease steady-state error to very low levels with appropriate state covariance's optimized for steady state.

In the second part of simulation, the performance of FL controller is assessed in terms of speed tracking and load rejection capability. The performance of the FL controller is compared to that of PI controller. In this case, the speed controller receives the estimated speed from the EKF block and then generates the required torque to obtain the required speed. The estimated speed is also fed into the IFOC block to generate the required slip angle for proper filed orientation.

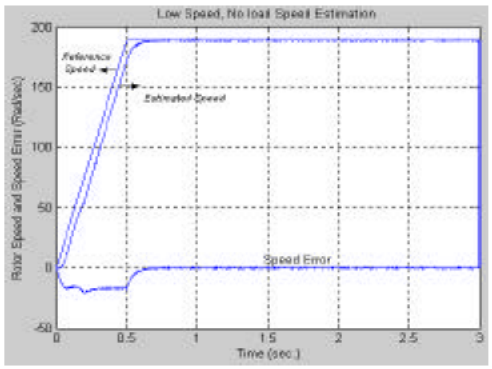

Fig. 7: High Speed, no-load, speed estimation steady state performance optimized

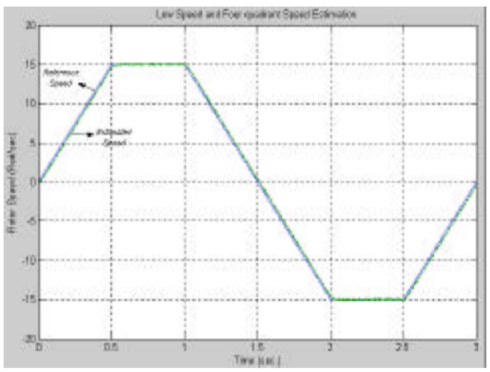

Fig. 8: Low speed, no-load, four quadrant speed estimation

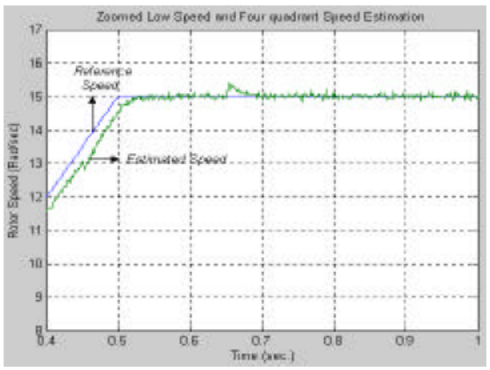

Fig. 9: (Zoomed) Low Speed, No-Load, Speed Estimationat Steady State

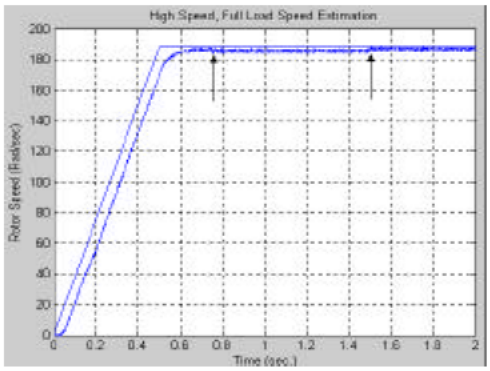

Fig. 10:High Speed, Full-Load Speed Estimation

Figure 11 and 12 show the speed tracking performance, under no load, for both FL and PI controllers, respectively. It can be shown that PI controller has difficulty in following the command because of the current limit and the time needed to build up the flux once the flux is established. At the same time, the FL controller shows almost perfect overlap of the command speed with the estimated speed.

To improve the speed tracking performance of PI controller, the gains of the PI controller have to be retuned, such as increasing the integration gain. However, overshoot and oscillation are usually associated with the increase of the gain. Therefore, there is a serious conflict in the speed performance.

Figure 13 shows the responses of PI and FL Controllers with stepped load torque of four times the nominal inertia (4J). The load is exerted to motor between 0.75-1.5 sec. The robustness in the response is self-evident.

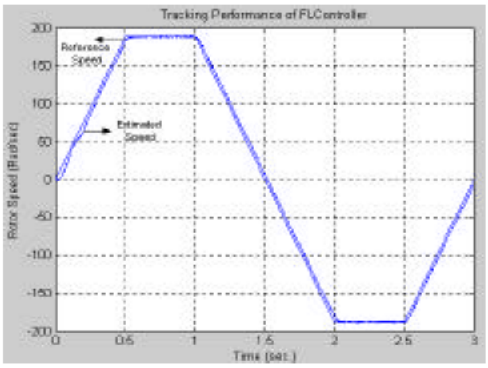

Fig. 11: Speed tracking performance of the PI-based IM

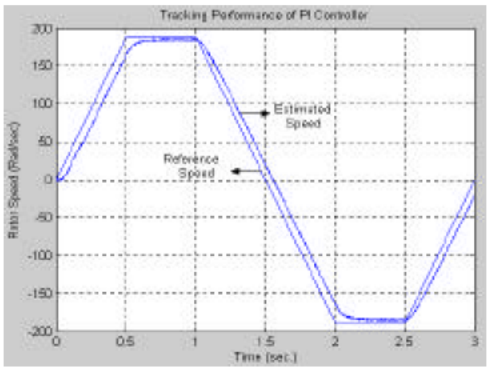

Fig. 12: Speed tracking performance of the of the PI-based IM FMRLC-based IM

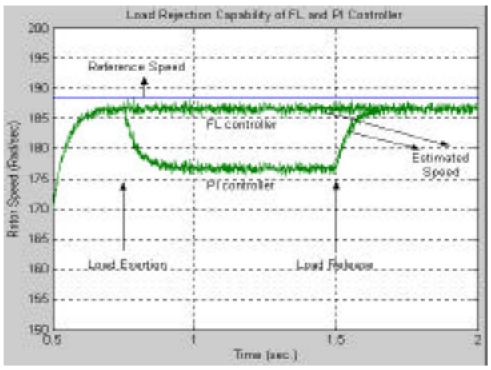

Fig. 13: Torque disturbance rejection capability of the PI and FL based IM

When the rotor's resistance is doubled between 1-1.25 sec., the speed becomes oscillatory indicating that the flux and torque current commands are no longer decoupled. Despite the loss of decoupling, the FL

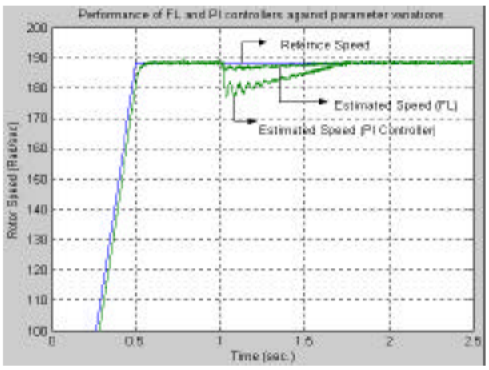

Fig. 14: Detuning Effects of FL and PI Controller-Based IM

controller can manage to return the speed to settling point faster than the case with PI controller. Moreover, the maximum dip in the response of FL controller is much less than that in case of PI controller as shown in Fig. 14. One can deduce that the FL controller is more robust than the PI controller against the variation of IM parameters (rotor resistance).

CONCLUSION

The following points can be deduced from the results associated with speed estimation using EKF technique:

- The EKF shows high tracking performance for both high and low speed estimations and close to reference speed. The high performance is verified for four-quadrant speed.
- The performance of EKF has been verified under loaded conditions. The EKF works properly even under fully loaded case.
- The steady-state error may be decreased to very low levels with appropriate state covariance's optimized for steady-state case.

It has been shown that a properly designed direct fuzzy controller can outperform conventional PI controllers. Based on simulation results, the following conclusions are made:

- The FC can be tuned to a single setting such that the speed will track trapezoidal commands with zero steady-state error.
- FC is more robust than the PI controller when load disturbances occurred.
- FC is more robust than the PI controller when detuning effect (rotor resistance variation) is encountered.

REFERENCES

Barambones, O., A.J. Garrido and F.J. Maseda, 2002. A sensorless robust vector control of induction motor drives, University of Pais Vasco.

Bilal Akin, 2003. State estimation techniques for speed sensorless field oriented control of induction machine, MSC. Thesis, The Middle East Technical University.

Bimal, K. Bose, 2002. Modern power electronics and ac drive, University of Tennessee, Knoxville, Prentice Hall.

Chee-Mun Ong, 1998. Dynamic simulation of electric machinery using Matlab/Simulink, Purdue University, Prentice Hall, PTR.

Cheng, F.F. and S.N. Yeh, 1993. Application of fuzzy logic in the speed control of ac servo systems and an intelligent inverter, IEEE. Trans. Ener. Conv., 8: 312-318.

Dal, Y., Ohm, 2001. Dynamic model of induction motors for vector control, drivetech, Inc., Blacksburg, Virginia.

Denai, M.A. and S.A. Attia, 2002. Fuzzy and Neural Control of an Induction Motor, Int. J. Applied Math. Comput. Sci., 12: 221-233.

Denai, M.A. and S.A. Attia, 2002. Intelligent control of an induction motor. Int. J. Electric Power Components and System. Taylor and Francis, 30: 409-427.

Heber, B., L. Xu and Y. Tang, 1997. Fuzzy logic enhanced speed control of an indirect field-oriented induction machine drive, IEEE. Trans. Power Elect., 12: 772-778.

Leonhard, W., 1997. Control of electrical drives, springer press, Birlin.

Li Zhen and Longya Xu, 2000. Fuzzy learning enhanced speed control of an indirect field-oriented induction machine drive, IEEE.

Ouhrouche, M.A., 2000. EKF on-line tuning of rotor time constant in an induction motor vector control, International Journal of Power and Energy System, Vol. 20.

Wade, S., M.W. Dunnigan and B.W. Williams, 1997. Modeling and simulation of induction machine vector with rotor resistance identification, IEEE. Transaction Power Elecronic, Vol. 12.