Scalar Speed Control of a dq Induction Motor Model Using Fuzzy Logic Controller

Ramón C. Oros†, Guillermo O. Forte‡, Luis Canali‡

† Departamento de Electrónica, Facultad Regional Córdoba, Universidad Tecnológica Nacional
roros@electronica.frc.utn.edu.ar
‡ Grupo de Investigaciones en Informática para la Ingeniería, F.R. Córdoba, U.T.N.
(gforte, lcanali)@scdt.frc.utn.edu.ar

Abstract: This paper presents a rule-based fuzzy logic controller applied to a scalar closed loop Volts/Hz induction motor (IM) control with slip regulation and its simulation results. They are also compared with those of a PI controller. The IM is modeled in terms of dq-windings, with synchronous frame associated with the frequency ωs of the stator excitation. The results obtained in the simulation are interesting, considering the presence of strong non-linearities in the IM model.

Keywords: Fuzzy logic controller (FLC), induction motor (IM), PI.

1. INTRODUCTION

In the last few years, fuzzy logic has met a growing interest in many motor control applications due to its non-linearities handling features and independence of the plant modeling. The fuzzy controller (FLC) operates in a knowledge-based way, and its knowledge relies on a set of linguistic if-then rules, like a human operator.

The present work consists in the develop and simulation of a controller for a closed loop speed control where the manipulated variable is the volts/Hz relation and, therefore, the slip value. For such applications, the proposed FLC is a suitable way to provide the necessary frecuency variating command signal. The frequency command also generates the voltage command through a volts/Hz function generator, with the low frequency stator drop compensation. For simulation purposes, all values are normalized to per unit (pu).

This paper will focus only on FLC techniques and the comparison with the classical PI controller.

2. FUZZY LOGIC CONTROLLER (FLC)

In the last few years, fuzzy logic has met a growing interest in many motor control applications due to its non-linearities handling features and independence of the plant modeling. The fuzzy controller (FLC) operates in a knowledge-based way, and its knowledge relies on a set of linguistic if-then rules, like a human operator.

2.1 Architecture

The controller architecture includes some rules which describe the casual relationship between two normalized input voltages and an output one. These are:

-Error (e), that is the speed error,

-Change-of-error (Δe), that is the derivative of speed error, and

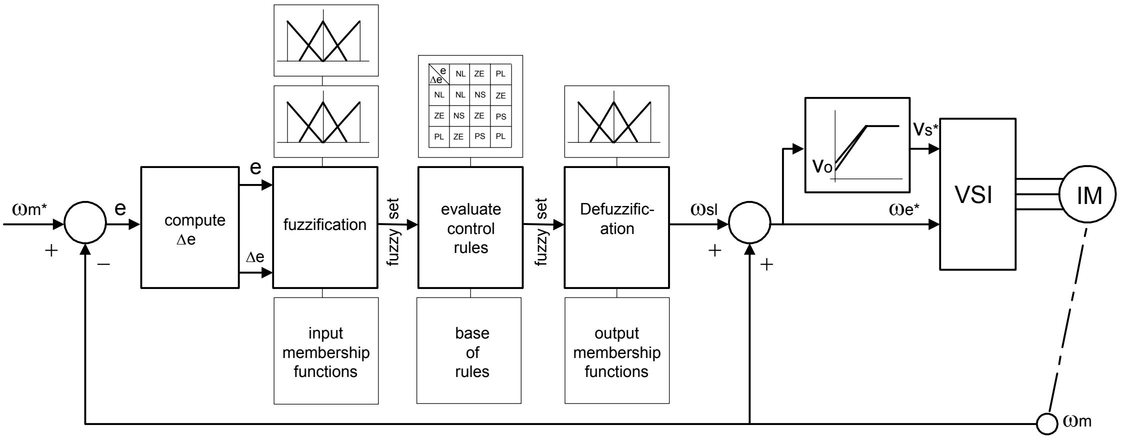


Fig. 1. Block diagram of the scalar IM control with the FLC architecture

-Output, defined as the change-of-control ($\Delta\omega_{sl}$), that added to the motor speed (ω_m) is the input (ω_e^*) to the converter.

These error inputs are processed by linguistic variables, which require to be defined by membership functions (Ouiguini et al, 1997).

Fig.1 shows a scalar IM control structure with fuzzy knowledge based controller (FKBC). The FLC includes four major blocks: one that computes the error into two input variables, a fuzzification block, an inference mechanism, and the last step is defuzzification. The speed reference control is ω_m^* .

2.2 Knowledge Base Proposed

Fig 2 shows the triangle-shaped membership functions of error (e) and change-of-error (Δe).

The fuzzy sets are designated by the labels: NL (negative large), NM (negative medium), NS (negative small), ZE (zero), PS (positive small), PM (positive medium), and PL (positive large).

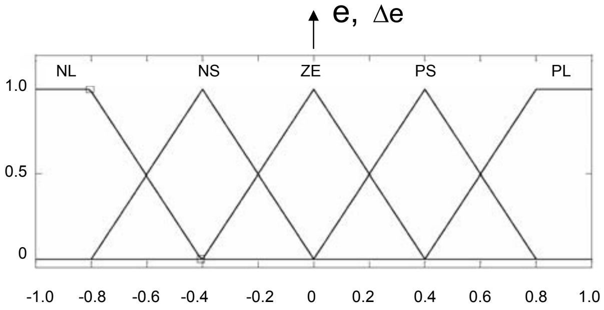


Fig. 2. Triangular membership functions for input variables e and Δe

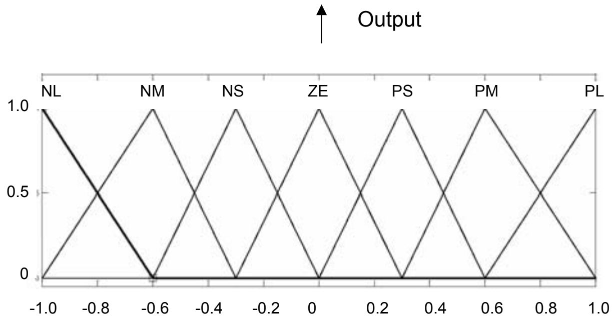


Fig. 3. Triangular membership functions for output variable

Fig. 3 and Table 1 shows the proposed membership functions for output variable and the control rules. The inference strategy used in this system is the Mamdani algorithm, and the center-of-area/gravity method is used as the defuzzification strategy.

Table 1. Linguistic Rule Table

Δe \ e	NL	NS	ZE	PS	PL
NL	NL	NL	NM	NS	ZE
NS	NL	NM	NS	ZE	PS
ZE	NM	NS	ZE	PS	PM
PS	NS	ZE	PS	PM	PL
PL	ZE	PS	PM	PL	PL

According to (Driancov et al, 1993) the equation giving a PI-like FKBC is

$$\Delta\omega_{sl} = k_p \, \Delta e + k_i \, e \qquad (1)$$

The fuzzy if-then statements are symbolically expressed with the form

If e is (…) and Δe is (…) then ω_{sl} is (…)

The output control is, then

$$\omega_e^* = \omega_m + \omega_{sl} \qquad (2)$$

The command signal is obtained from twenty-five rules witch all have the same weight of one, as shown in appendix.

To tune the fuzzy control, it is possible to change the two values k_p and k_i (ec.1).

Fig. 4 shows the control surface extracted from the Matlab® fuzzy logic toolbox surface viewer, when k_p and k_i=1.

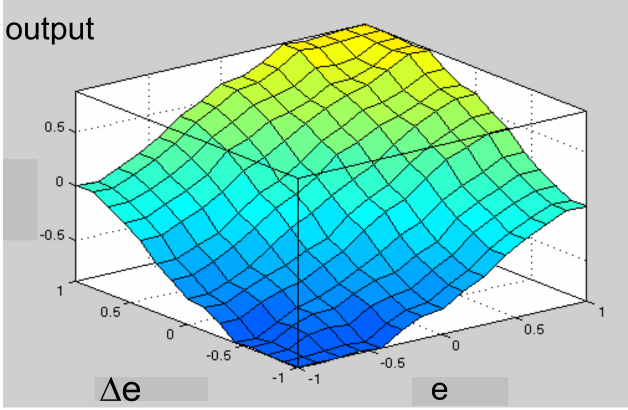


Fig. 4. Three-dimensional plot of control surface

3. IM DYNAMICAL MODEL

The induction motor is modeled with Matlab/Simulink® program running under three-phase sinusoidal symmetrical excitation and is at vectorized form in conformity with state vector formulation. Synchronous frame is used where $\omega_k = \omega_m$ and thetak $= \omega_0$, and where:

ω_0 = base freq.; ω_m = rotor frame freq.; ω_k =dq frame freq.; ω_s = synchronous frame freq.; (rad/sec)

λs=stator flux, λr=rotor flux (pu)

R_s ; R_r =stator and rotor resistance (pu)

$\bar{v}_s$; $\bar{v}_r$ =stator and rotor voltage (pu)

$\bar{i}_s$; $\bar{i}_r$ =stator and rotor current (pu)

L_s ; L_r = stator and rotor inductance (pu)

L_m = magnetizing inductance (pu)

L_{sl} = stator leakage inductance (pu)

L_{rl} = rotor leakage inductance (pu)

T_e = electromagnetic torque (pu)

T_L = load torque (pu)

B_m = viscous friction coefficient. (pu)

d, q=direct and quadrature axis

p=number of poles

H= inertia constant (s)

Operators: $\otimes$ =cross product; $\bullet$ =dot product

3.1 Electrical System Equations

According to (Mohan 2001, 2003), (Leonhard 2001), (Bose 2002), and (Vas 1992), the IM equations in pu are:

$$\bar{v}_s = R_s\,\bar{i}_s + \frac{1}{\omega_0}\left(\frac{d\bar{\lambda}_s}{dt}\right) + \omega_k\,M_{(pi/2)}\,\bar{\lambda}_s \tag{3}$$

$$\bar{v}_r = R_r\,\bar{i}_r + \frac{1}{\omega_0}\left(\frac{d\bar{\lambda}_r}{dt}\right) + (\omega_k - \omega_m)M_{(pi/2)}\,\bar{\lambda}_r \tag{4}$$

And where respective i, v, the space vector λ and the rotational operator M are defined in dq-windings as:

$$\bar{\lambda} = \begin{bmatrix} \lambda_d \\ \lambda_q \end{bmatrix}; \quad \bar{i} = \begin{bmatrix} i_d \\ i_q \end{bmatrix}, \quad M_{(pi/2)} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

3.1 Flux Linkage-Current Relations

On d axis:

$$\bar{\lambda}_{sd} = L_s\,\bar{i}_{sd} + L_m\,\bar{i}_{rd} \tag{5}$$

$$\bar{\lambda}_{rd} = L_m\,\bar{i}_{sd} + L_r\,\bar{i}_{rd} \tag{6}$$

where

$$L_s = L_m + L_{sl} \tag{7}$$

$$L_r = L_m + L_{rl} \tag{8}$$

On q axis:

$$\bar{\lambda}_{sq} = L_s\,\bar{i}_{sq} + L_m\,\bar{i}_{rq} \tag{9}$$

$$\bar{\lambda}_{qr} = L_m\,\bar{i}_{qs} + L_r\,\bar{i}_{qr} \tag{10}$$

3.1 Mechanical System Equations

$$T_e = 2H\frac{d\omega_{mec}}{dt} + B_m\omega_{mec} + T_L \tag{11}$$

where

$$T_e = \bar{\lambda}_s \otimes \bar{i}_s = M_{(pi/2)}\,\bar{\lambda}_s \bullet \bar{i}_s \tag{12}$$

and

$$\omega_{mec} = \frac{2}{p}\,\omega_m \tag{13}$$

3. SIMULATION RESULTS

The response of the controller will be investigated with the Matlab/Simulink® simulation program, the Fuzzy logic, and SimPowerSystems toolbox. See Fig 5.

The parameters describing the electrical and electromechanical system are expressed in per unit, pu.

Stator resistance (*pu*)= 0.01

Rotor resistance (*pu*)= 0.02

Stator leakage inductance (*pu*)= 0.10

Rotor leakage inductance (*pu*)= 0.10

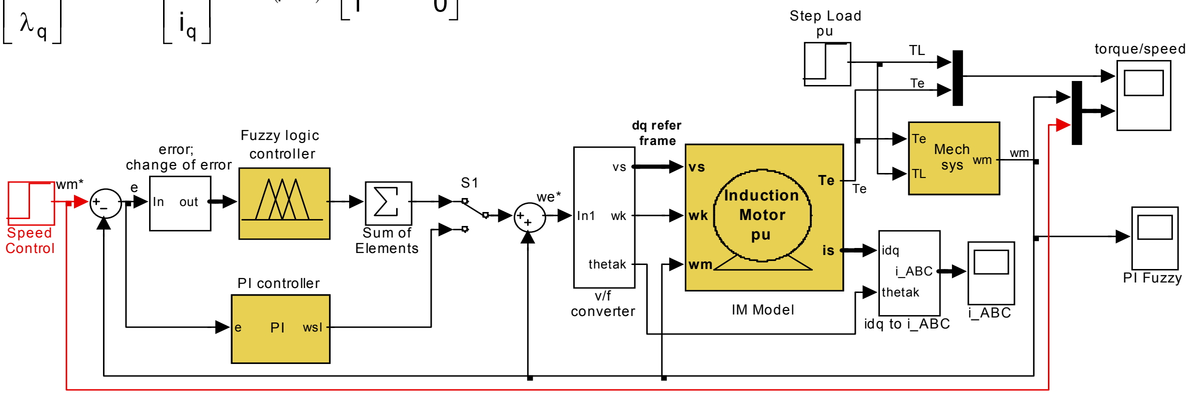
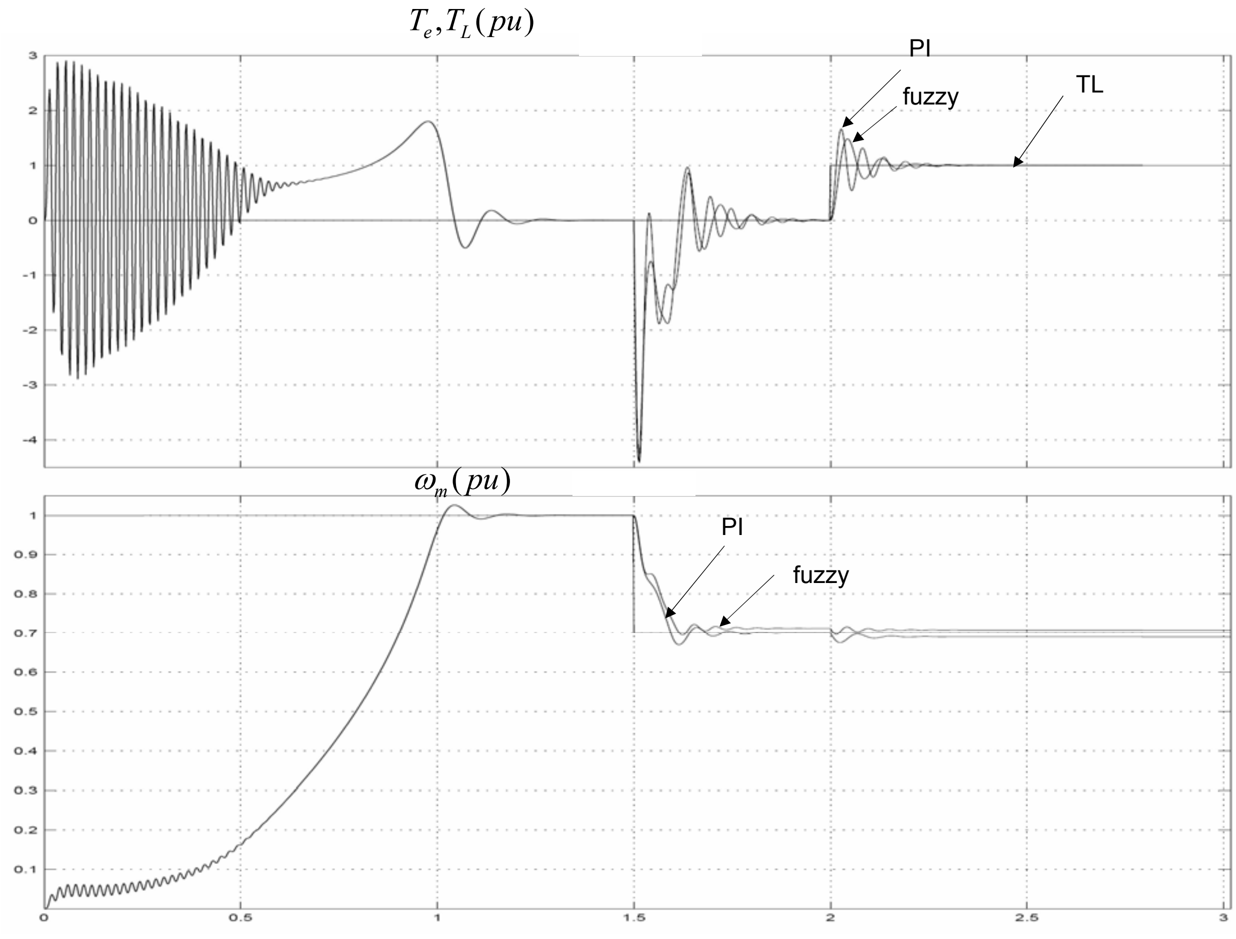


Fig. 5. Block diagram of scalar IM FLC in Simulink®


$T_e, T_L (pu)$

$\omega_m (pu)$

Fig. 6. Performances of classical PI and fuzzy due to torque
and speed changes

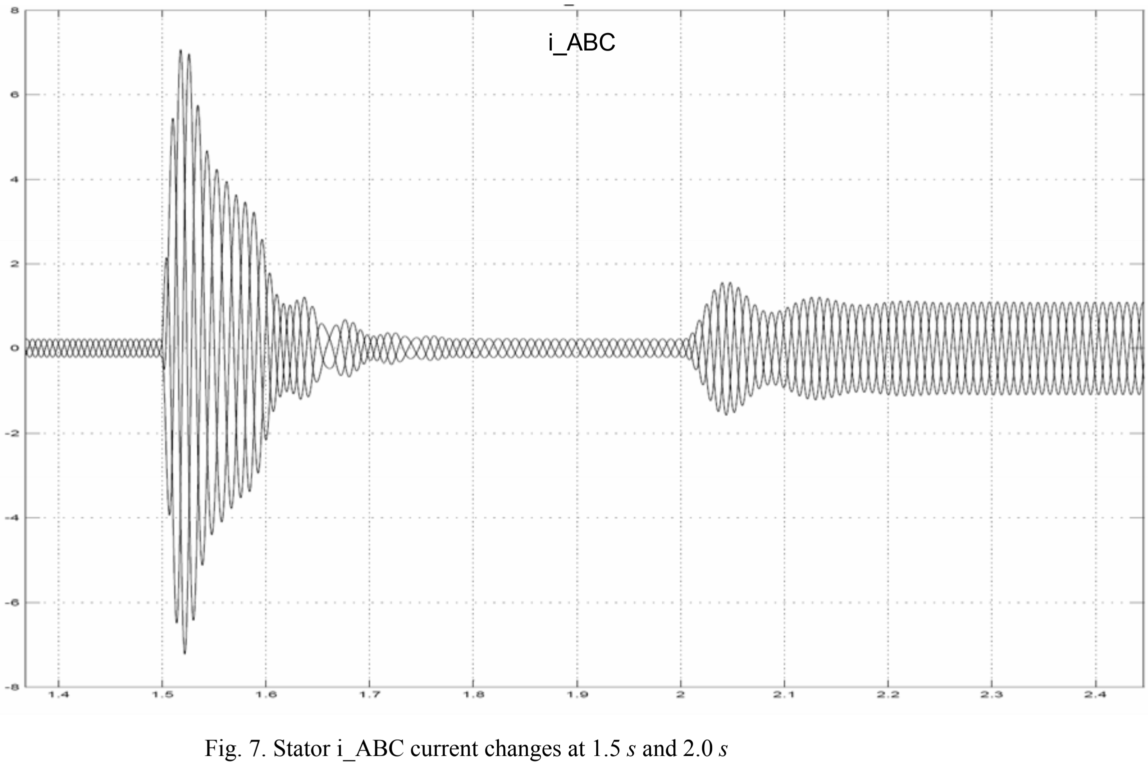


Fig. 7. Stator i_ABC current changes at 1.5 *s* and 2.0 *s*

Magnetizing inductance (*pu*)= 4.5
Base frequency (rad/s)= 2*pi*50
p=2; or $\omega_{mec} = \omega_m$
H=0.3 (s)
B_m (pu)=1e-5

The parameters of fuzzy speed controller are:
Kp=0.3; Ki=1

The parameters of PI speed controller are:
Kp=3; Ki=5; saturation limit (*pu*)=0.5

The tuning method used for the classical PID controller was based on the optimization of the absolute integral error using the best approximation of the system with a first order transfer function with delay. The FLC controller parameters where chosen based on the parameters of the classical PID (Ramón Ferreiro García et. al.), then the tuning was empirically improved.

The three-phase sinusoidal excitation can be adjusted in both amplitude and frequency.
In the simulation all the initial conditions are assumed to be zero.
The motor is started without load at rated voltage and frequency until t=1.5*s*. After this time, and after reaching the steady-state conditions, voltage and frequency are both changed suddenly to 0.7 *pu*, up to the end.
At t=2.0*s* a full load step function is applied.

Figure 6 shows the comparison between fuzzy controller and classical PI controller.

As shown in Fig. 6, the proposed FLC reacts rapidly when there is a change in the speed command ω_m^* or in the TL. The effects of line voltage and load variation are shown too. Figure 7 shows the stator current due to speed command and load changes on the FLC at steady state.

4. CONCLUSION

A new FLC that improve the performance of scalar IM speed drives has been proposed. The method uses a new linguistic rule table in FKBC to adjust the motor control speed, and this FLC can achieve a good system performance of the IM scalar drive, and it is possible to implement a PI fuzzy logic controller instead the traditional PI controller.

ACKNOWLEDGMENTS

We want to thank Professor Mahmoud Riaz from the Department of Electrical and Computer Engineering, Minnesota University for permission of using his dq IM model at Simulink® file "im_3frames.mdl" available at
http://www.ece.umn.edu/users/riaz/macsim/electricdrives.zip

REFERENCES

Afonso, J.L., Fonseca, J., Martins, J.S., Couto, C.A., "Fuzzy logic techniques applied to the control of a three-phase induction motor", *Industrial Electronics, ISIE '97, Proceedings of the IEEE International Symposium on*, **3**, Guimaraes, Portugal, 1179-1184, (1997).

Bose Bimal K., *Modern Power Electronics and AC Drives*, Prentice Hall PTR, Upper Saddle River, (2002).

Cupertino, F., Lattanzi, A., Salvatore, L., "A new fuzzy logic-based controller design method for DC and AC impressed-voltage drives", *Power Electronics, IEEE Transactions on*, **15**, 974 – 982, (2000).

D'Angelo, G., Lo Presti, M., Rizzotto, G., "Fuzzy controller design to drive an induction motor", *Fuzzy Systems, IEEE World Congress on Computational Intelligence, Proceedings of the Third IEEE Conference on*, **3**, 1502 - 1507 (1994)

Driankov D., Hellendoorn H., Reifrank M., *An Introduction to Fuzzy Control*, Second Edition, Springer-Verlag, (1993).

Leonhard Werner, *Control of Electrical Drives*, 3rd edition, Springer-Verlag, Berlin, (2001).

Mao-Fu Lai, Chen Chang, Wen-Yuh Chiou, "Design of fuzzy logic controller for an induction motor speed drive", *SICE '97. Proceedings of the 36th SICE Annual Conference,* 1071 – 1076, Tokushima, (1997).

Martins, J.E., Branco, P.J.C., Pires, A.J., Dente, J.A., "New trends in recognizing experimental drives: fuzzy logic and formal language theories" *Fuzzy Systems, IEEE Transactions on,* **9**, 68 – 87, (2001).

Mohan N., *Electric Drives: An Integrative Approach*, MNPERE, Minnesota (2003).

Mohan N., *Advanced Electric Drives: Analysis, Control and Modeling using Simulink®,* MNPERE, Minnesota (2001).

Ouiguini, R., Djeffal, K., Oussedik, A., Megartsi, R., "Speed control of an induction motor using the fuzzy logic approach", *Industrial Electronics, ISIE '97, Proceedings of the IEEE International Symposium on*, **3**, Guimaraes, Portugal, 1168 – 1172, (1997).

García, Ramón Ferreiro, Martinez, Xoan Pardo, Vidal Paz, José, "De los reguladores convencionales a los reguladores basados en lógica difusa".

Vas Peter, *Vector Control of AC Machines*, Clarendon Press-Oxford, U.K., (1990).

Vas Peter, *Electrical Machines and Drives: A space Vector Theory Approach*, Oxford University Press, U.K., (1992).

Zhao Jin, Bose, B.K., "Evaluation of membership functions for fuzzy logic controlled induction motor drive", IECON 02 [Industrial Electronics Society, IEEE 2002 28th Annual Conference of the] , **1** , 229 – 234, (2002).

APPENDIX

The if-then rules are defined as follow

1. If(error is NL) and (change-error is NL) then (output is NL) (1)
2. If(error is NS) and (change-error is NL) then (output is NL) (1)
3. If(error is ZE) and (change-error is NL) then (output is NM) (1)
4. If(error is PS) and (change-error is NL) then (output is NS) (1)
5. If(error is PL) and (change-error is NL) then (output is ZE) (1)
6. If(error is NL) and (change-error is NS) then (output is NL) (1)
7. If(error is NS) and (change-error is NS) then (output is NM) (1)
8. If(error is ZE) and (change-error is NS) then (output is NS) (1)
9. If(error is PS) and (change-error is NS) then (output is ZE) (1)
10. If(error is PL) and (change-error is NS) then (output is PS) (1)
11. If(error is NL) and (change-error is ZE) then (output is NM) (1)
12. If(error is NS) and (change-error is ZE) then (output is NS) (1)
13. If(error is ZE) and (change-error is ZE) then (output is ZE) (1)
14. If(error is PS) and (change-error is ZE) then (output is PS) (1)
15. If(error is PL) and (change-error is ZE) then (output is PM) (1)
16. If(error is NL) and (change-error is PS) then (output is NS) (1)
17. If(error is NS) and (change-error is PS) then (output is ZE) (1)
18. If(error is ZE) and (change-error is PS) then (output is PS) (1)
19. If(error is PS) and (change-error is PS) then (output is PM) (1)
20. If(error is PL) and (change-error is PS) then (output is PL) (1)
21. If(error is NL) and (change-error is PL) then (output is ZE) (1)
22. If(error is NS) and (change-error is PL) then (output is PS) (1)
23. If(error is ZE) and (change-error is PL) then (output is PM) (1)
24. If(error is PS) and (change-error is PL) then (output is PL) (1)
25. If(error is PL) and (change-error is PL) then (output is PL) (1)