A Comparative Study of Adaptive Fuzzy Control Schemes For Induction Motor Drives

A. El Dessouky and M. Tarbouchi
Royal Military college of Canada, Electrical and Computer Engineering Department, Canada

Abstract- This paper presents a comparative study of a Fuzzy Model Reference Self-Tuning Controller (FMRSTC) algorithm with two major fuzzy adaptation algorithms. These two algorithms are the Fuzzy Model Reference Control (FMRLC), and the Fuzzy Self-Tuning Control (FSTC). Simulations and real time implementation using induction motor speed control system have been performed for the algorithms performance comparison. The field oriented control technique is used in both simulation and implementation.

Key-Words:- Adaptive fuzzy control, fuzzy self-tuning controllers, induction motor control.

1. INTRODUCTION

Many fuzzy adaptive techniques have been proposed to solve two major problems in the fuzzy control field. The first problem is the selection and optimisation of the fuzzy controller parameters, while the second problem is the adaptation of these parameters as the process parameters change during the real time operation. The optimization of the fuzzy controller parameters in an off-line process has been proposed using both genetic algorithms [1,2] and neural networks [3]. In this paper the focus will be on the analysis of the adaptation mechanism that can compensate for process parameters variation.

Fuzzy Self-Tuning Controllers (FSTC) [4,5] are controllers with adaptation algorithms that tune the output scaling factor of the fuzzy controller. Its structure consists of two parallel fuzzy processors. The first acts as the main fuzzy controller and the second is used as the performance testing and scaling factor (SF) regulator. This structure would improve the controller accuracy with a reduced number of membership functions. However it would not provide the adaptation mechanism that compensates for the parameter variation of the plant.

Another adaptation alternative is to target the rules at the rule base of the fuzzy controller (adaptation of the centres of the output membership function). This adaptation technique (called Fuzzy Model Reference Learning Control (FMRLC)) has the ability to memorize the control surface; which justify the term "learning" in its name. This technique has been proposed and implemented by Passino et al. in [6,7,8,9], Silva el. al in [10] and Mayhan et al. In [11].

In real time implementation, FMRLC may delay in providing the desired control action especially at the start (when the controller has to collect the information about the control surface). FMRLC also requires a long time to adapt to parameter variation. The adaptation of the output-scaling factor has a faster and more significant effect on system performance.

This paper demonstrates that the fuzzy model reference self-tuning controller (FMRSTC) is faster and simpler adaptive control algorithm than the FMRLC. It also provides the controller with an adaptation mechanism not found in STFC.

Analysis and comparison of the techniques and performance between the proposed algorithm (FMRSTC) and the other two algorithms (FMRLC and FSTC) will be introduced.

The paper is organized as follows: Section II introduces the structure and adaptation algorithm for the conventional fuzzy self-tuning controller. Section III discusses the fuzzy model reference learning controller. Section IV is devoted to the description of the newly proposed adaptation algorithm and the advantages of this approach over the previous two algorithms. Section V discusses simulation results of the three algorithms of the controller parameters respectively. The experimental results of the three algorithms applied to speed control of an induction motor are introduced in section VI.

2. FUZZY SELF-TUNING CONTROLLER

The Fuzzy Self-Tuning Controller (FSTC) consists of two parallel fuzzy processors. The first is the main fuzzy controller; the second (which is called fuzzy performance testing and SF regulator) is designed to adapt the output-scaling factor or the input scaling factors of the first fuzzy controller. Tuning the input scaling factors is a complex process and has a less significant effect on the system performance. Figure (1) shows the block diagram of the STFC when the output-scaling factor is the target for the adaptation. Both fuzzy controllers used in this work are PD fuzzy controllers. The inputs of the two fuzzy controllers are given by:

$$e(k) = r(k) - y(k) \tag{1-a}$$

$$c(k) = \frac{e(k) - e(k-1)}{T} \tag{1-b}$$

where $r(k)$ is the set point, $y(k)$ is the output of the process, and T is the sampling time. Triangle membership functions were used for all inputs and outputs of the two fuzzy controllers. The rule-base of the fuzzy controller is built based on the information about the process. The basic structure of such a rule-base is shown in table (1). The rule-base of the fuzzy performance testing is shown in table (2).

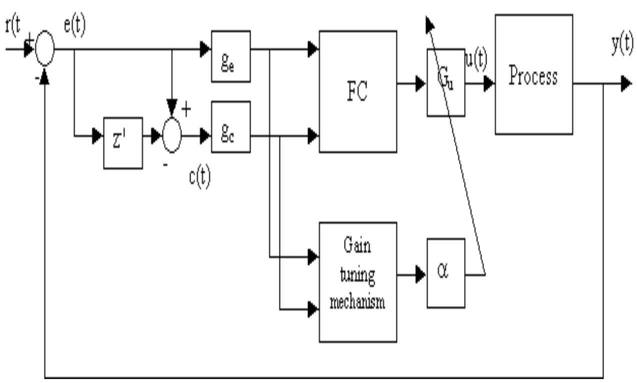


Figure (1) structure of fuzzy self-tuning controller.

Table (1) rule-base for the fuzzy controller and inverse model.

Δe/e	NB	NE	ZE	PO	PB
NB	NB	NB	NB	NE	ZE
NE	NB	NB	NE	ZE	PO
ZE	NB	NE	ZE	PO	PB
PO	NE	ZE	PO	PB	PB
PB	ZE	PO	PB	PB	PB

Table (2) rule-base for the fuzzy the fuzzy performance testing and SF's regulator

Δe/e	NB	NE	ZE	PO	PB
NB	PB	PB	NB	PO	ZE
NE	PB	PB	PO	ZE	PO
ZE	PB	PO	ZE	PO	PB
PO	PO	ZE	PO	PB	PB
PB	ZE	PO	PB	PB	PB

The primes membership value is given by:

$$\mu_A^i = \min(\mu_{A_j}^1, \mu_{A_k}^2) \tag{2}$$

where i is the index of the rule, A_j is the linguistic value for the input (the center of the j^{th} input membership function i.e. $A_0 =$ negative large, $A_1 =$ negative … etc), $\mu_{A_j}^1$ is the membership value of the 1st input to the A_j linguistic value. The output of the fuzzy controller is given by the following equation:

$$y^{crisp} = \frac{\sum_{i=1}^{R_1} b_i \sup_y \{\mu_{\hat{B}^i}(y)\}}{\sum_{i=1}^{R_1} \sup_y \{\mu_{\hat{B}^i}(y)\}} \tag{3}$$

similarly, the output of the fuzzy performance testing and SF regulator is given by:

$$\alpha = \frac{\sum_{l=1}^{R_2} b_l \sup_y \{\mu_{\hat{B}^l}(y)\}}{\sum_{l=1}^{R_2} \sup_y \{\mu_{\hat{B}^l}(y)\}} \tag{4}$$

Where y^{crisp} and α are the crisp outputs, b_i and b_l are the centers of the i^{th} and l^{th} output membership functions, R_1 and R_2 are the numbers of the rules at the rule base, (sup (x) denotes supremum value of μ(x) which can be assumed as the upper bound of the chopped output membership function) and $\mu_{\hat{B}^i}$ and $\mu_{\hat{B}^l}$ are the implied fuzzy sets for the i^{th} and l^{th} rules for the fuzzy controller and the fuzzy performance testing and SF regulator respectively. The implied fuzzy set $\mu_{\hat{B}^i}$ is given by:

$$\mu_{\hat{B}^i}(y) = \mu_A^i * \mu_{B^i}(y) \tag{5}$$

where $\mu_{B^i}(y)$ represent the membership function of the output (Triangle membership function was used in this paper). Assuming the use of minimum and maximum function for the inference mechanism then:

$$\sup_y \{\mu_{\hat{B}^i}(y)\} = \mu_A^i \tag{6}$$

and hence equation (3) and (4) would be:

$$y^{crisp} = \frac{\sum_{i=1}^{R_1} b_i \mu_A^i}{\sum_{i=1}^{R_1} \mu_A^i} \tag{7}$$

$$\alpha = \frac{\sum_{l=1}^{R_2} b_l \mu_A^l}{\sum_{l=1}^{R_2} \mu_A^l} \tag{8}$$

The adaptation of the output-scaling factor of the fuzzy controller is given by:

$$g_u = G_u * \alpha \tag{9}$$

where g_u is the output-scaling factor, G_u is the maximum value for the output-scaling factor, and α is the adaptation factor. According to table (2) α has a value between 0 and –1. In this work α was shifted by 0.7 to ensure a minimum value for the output-scaling factor. This would also reduce the steady state error. The control signal (*u*) is given by:

$$u(k) = y^{crisp}(k) * G_u * \alpha \tag{10}$$

Hence $u(k)$ at a given sample can be rewritten as follow:

$$u = G_u * \frac{\sum_{i=1}^{R_1} b_i \mu_A^i}{\sum_{i=1}^{R_1} \mu_A^i} * \frac{\sum_{l=1}^{R_2} b_l \mu_A^l}{\sum_{l=1}^{R_2} \mu_A^l} \tag{11}$$

$$\therefore\ u = G_u * \frac{\sum_{i=1}^{R} b_i \mu_A^i}{\sum_{i=1}^{R} \mu_A^i} \tag{12}$$

where $b_i = b_m * b_n$, $\mu_A^i = \mu_A^m * \mu_A^n$, $i = 1:R$, $m = 1:R_1$, and $n = 1:R_2$. From equation (12) it can be noted that FSTC with the proposed structure in [4,5] can be simplified as a single fuzzy controller with rules at the rule-base $R = R_1 * R_2$ followed by integrator. The total control surface including the effect of the integrator is shown in figure (2).

As a result, the previous analysis shows that the FSTC represent an enhanced fuzzy controller with fewer rules (two parallel fuzzy controller with total rules $= R_1 + R_2$ instead of single fuzzy controller with $R_1 * R_2$ rules) however this controller has no adaptation capabilities. In other words, the control surface that describes the input output mapping is fixed even when the process parameters change (Figure (4)). Accordingly, the system performance would change if the process parameters changes and there would be no guarantee that this performance would be acceptable.

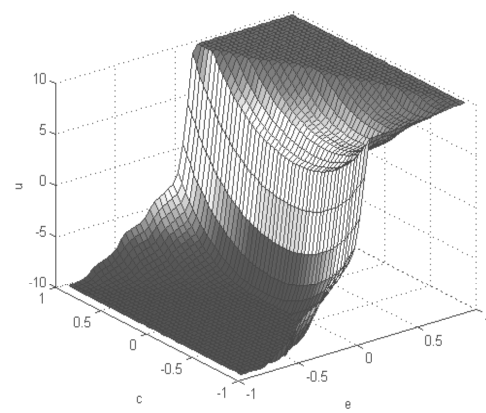


Figure (2) the control surface using FSTC algorithm

3. Fuzzy Model Reference Learning Control

Fuzzy Model Reference Learning Control (FMRLC) is based on the conventional model reference adaptive control (MRAC) algorithm. MRAC shows stability and good performance for non-linear systems. FMRLC was first proposed by Mamdani [12] in 1979.

Figure (3) shows the structure of FMRLC. In this control scheme the system (the plant together with the controller) is asked to follow certain performances given by a reference model. Normally, the reference model is a linear first or second order system. In this work a 2nd order system is used as a reference model given by:

$$\frac{Y_{ref}(s)}{R_{ref}(s)} = \frac{\omega_n^2}{s^2 + 2\eta\omega_n + \omega_n^2} \tag{13}$$

Where $R_{ref}(s)$ is the reference input, $Y_{ref}(s)$ is the output of the reference model, ω_n is the desired nature frequency and η is the desired damping ratio.

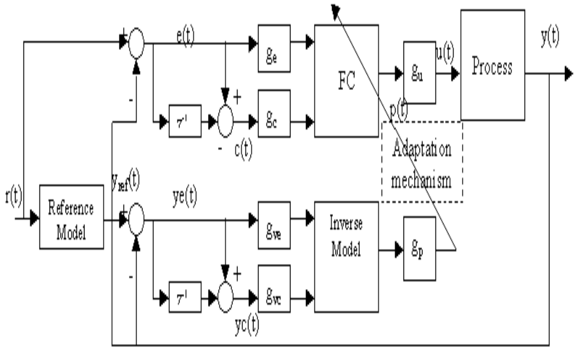


Figure (3) structure of fuzzy model reference learning control.

The learning and adaptation mechanism is responsible for forming the rules at the rule-base of the fuzzy controller. These rules describe the non-linear control surface that compensates and linearizes the overall system to match the reference model. It is also responsible to keep adapting the control surface to compensate according to the process varying parameters. The inputs of the main fuzzy controller (FC) are as in equations (1-a) and (1-b). Triangle membership functions are used for this work for the controller inputs and output for both the main fuzzy controller and the inverse model (The inverse model is a fuzzy controller with dynamics inverse to the process dynamics). The structure of the rule-base and the inverse model are identical to the one used for the main fuzzy controller in the FSTC for simplicity and are shown in table (1). If more information about the process is available it is possible to describe it in the rule-base of the fuzzy controller and the inverse model. The output of the main fuzzy controller is given by equation (2); however, the centres of the output membership function are subjected to the adaptation as follow:

$$b_i(k) = b_i(k-1) + G_p p(k) \tag{14}$$

where G_p is the adaptation gain and $p(k)$ is the output of the inverse model. The inverse model inputs are given by

$$ye(k) = y_{ref}(k) - y(k) \tag{15-a}$$

$$yc(k) = \frac{ye(k) - ye(k-1)}{T} \tag{15-b}$$

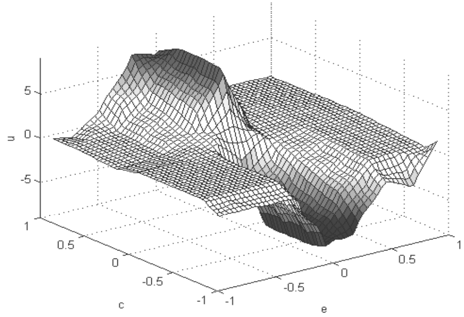
where $y_{ref}(k)$ represent the output of the reference model at the sample *k*. The output of the inverse model ($p(k)$) is similar to the output of the fuzzy controller that is similar to equation (4). To avoid generating control actions exceeding the limits of the process input the following 2 equation are added:

$$b_i(k) = b_{max} \qquad b_i(k) \geq b_{max} \tag{16-a}$$

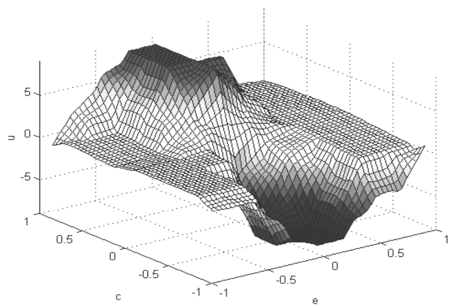
$$b_i(k) = b_{min} \qquad b_i(k) \leq b_{min} \tag{16-b}$$

where b_{min} and b_{max} are the minimum and maximum control action values. Equation (14) shows that the FMRLC algorithm provide both adaptation and learning capabilities. This is because the adaptation process is independent of the inputs of the main fuzzy controller. Also the adaptation process is seeking a certain performance defined by the reference model regardless of the change of the process parameters. Figure (5-a) shows the control surface using FMRLC for the system including the induction motor in speed control loop without load. Figure (5-b) shows the same control system when a load is added to the machine. The figure shows that the control surface has changed as the machine torque load changes. It shows that the algorithm not only adapt to the parameter variation but also memorize these changes in the rule-base of the controller.

However, if the parameters of the process are kept constant and the system faces the same conditions more than once, the performance will be the same without any necessary adaptation. This is only true on the condition where the inverse model is matching exactly the non-linearity of the process. If this condition is satisfied then the only adaptation is to compensate for the process parameter variation. However, if the inverse model does not match the inverse characteristics of the process, the adaptation process will be active indefinitely to adapt and minimise the error between the inverse model and the optimum inverse model. In most cases, the inverse model does not match the inverse characteristics of the process. This is because if the process parameter changes, the inverse model does not describe the inverse characteristics of the process any more. It is required to add identifier to keep changing the inverse model parameters as the process parameters change. This would add complexity to the system mechanism and slow down the adaptation process.



(a)



(b)

Figure (4) shows the control surface using FMRLC (a) machine is unloaded (b) machine loaded.

The FMRLC has a complex and slow adaptation process. This can be clearly noted if the rule-base of the fuzzy controller is set to zero or to certain values that are far from the optimum control surface. However after the system collects the optimum control information, however, it can adapt reasonably fast for process parameter variation if such a variation is limited within certain values (values that will not completely change the system performance).

4. Fuzzy Model Reference Self-Tuning Control

Based on the structure of STFC and FMRLC, a possible combination of the two algorithms can provide an easier and more efficient adaptation algorithm. The proposed structure is shown in figure (5). It can be noted that it is very close to the FMRLC structure in that it includes a reference model that describes the desired performance. However, the target for the adaptation is the output-scaling factor of the fuzzy controller. This will provide the system with a fast adaptation mechanism. The main fuzzy controller is a PI fuzzy controller. The inputs to the fuzzy controller are given by:

$$e(k) = r(k) - y(k) \tag{17-a}$$

$$c(k) = \sum_{i=0}^{k} e(i) \tag{17-b}$$

where $r(k)$ is the set point, $y(k)$ is the output of the process and k is the current sample. The integral output is limited according to the following:

$$c(k) = 2c_{\min} \quad c(k) \geq 2c_{\min} \tag{18-a}$$

$$c(k) = 2c_{\max} \quad c(k) \geq 2c_{\max} \tag{18-b}$$

where $c_{\min}$ and $c_{\max}$ are the centers of the c membership function. The inputs of the inverse model are identical to the ones used in the FMRLC algorithm (given by equation (15)). Triangular membership functions were selected for the inputs and the output of both the fuzzy controller and the inverse model. The rules at the rule-base were built based on rough information about the process under control. The rule-base of the fuzzy controller and the inverse model are similar to the one used by FMRLC and given by table (1).

The output of the fuzzy controller and the inverse model is the same as equation (2). For the proposed algorithm the centers of the output membership function are fixed. The target of the adaptation is the output scaling factor of the fuzzy controller. The output of the inverse model describes the desired correction to the control action to achieve the desired performance.

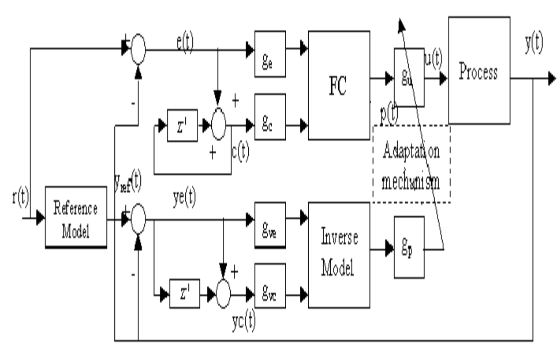


Figure (5) the structure of the proposed FMRSTC algorithm.

The output of the inverse $p(k)$ is given by:

$$p(k) = g_p * y_{Inv}^{crisp} \tag{19}$$

where y_{Inv}^{crisp} represent the crisp output of the inverse model and calculated similar to equation (7) and g_p is the output scaling factor of the inverse model (adaptation gain). The adaptation mechanism changes the output-scaling factor as follow:

$$g_u(k) = g_u(k-1) + \Delta g_u(k) \qquad (20)$$

where $\Delta g_u(k)$ is the incremental amount of the output scaling factor. The calculation of $\Delta g_u(k)$ is related to the output of the inverse model and the current state of the output of the process. Figure (6) shows all possible patterns of the output of the processes with reference to both the set point and the reference model. Consider the pattern shown in figure (7-a) where the error between the output of the process and the set point (e) is positive, the error between the output of the process and the reference model (ye) is positive and the set point is positive. The output of the inverse model ($p(k)$) at this case is positive. It is clear that in this case the process is slower than desired and more control action is required. To magnify the control action, it is necessary to increase the output scaling factor and hence $\Delta g_u(k)$ should be:

$$\Delta g_u(k) = p(k) \qquad (21)$$

To achieve this, a simple fuzzy controller is embedded on the adaptation mechanism to gradually switch between $p(k)$ and $-p(k)$. This fuzzy controller has a single input (e) and five membership functions for both the input and the output. Figure (7) shows the input and the output membership function used for the adaptation fuzzy controller. The output membership function has the centers of three membership functions set to zero. This is to avoid the oscillation of the output of the process around the set point. By adding this fuzzy controller, the adaptation of the output scaling factor will be reduced to zero as the output of the process approaches the set point. It is not possible to plot the total control surface when the adaptation information is not saved (the target of the adaptation is the OSF not the rule-base).

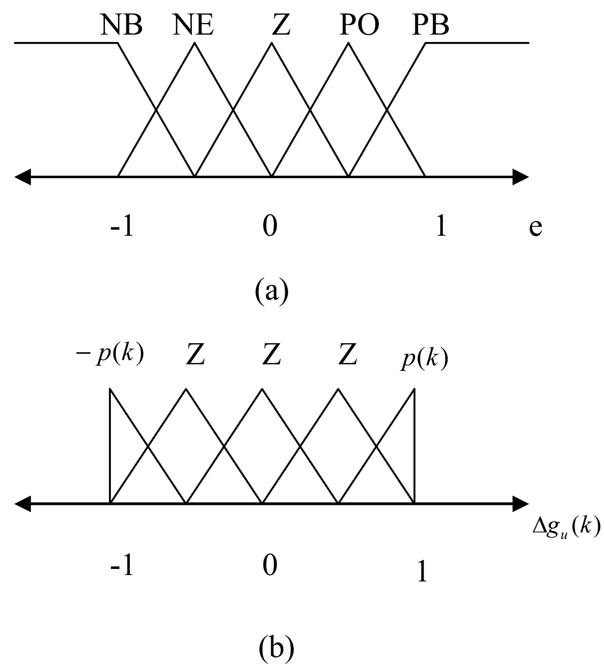


Figure (6) Membership function for (a) the input and (b) the output of the adaptation fuzzy controller

5. SIMULATIONS

A periodic reference speed between –1000 to 1000 rpm was applied to the three algorithms under investigation (FSTC, FMRLC, and FMRSTC). A load was applied after 10.3 sec for duration of one complete period of speed change to study the effect of the load change for both transient and steady state performances. Figures (7), (8) and (9)) show the response of the induction motor when the FMRLC and FMRSTC algorithms were applied respectively. The figures (8) and (9) includes the reference input, the output of the reference model and the speed of the induction motor. In figure (9), the Adaptation Factor (AF) is plotted in figure (OSF is fixed in FMRLC algorithm). In figure (9) both OSF and AF are plotted where the FMRSTC algorithm adapt the OSF.

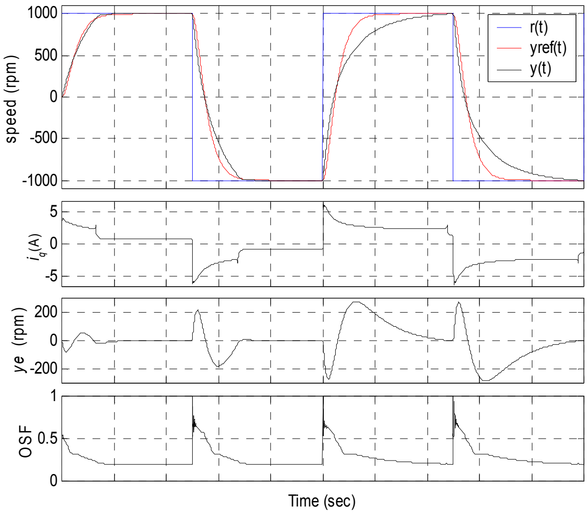


Figure (7) Control system response using optimized FSTC

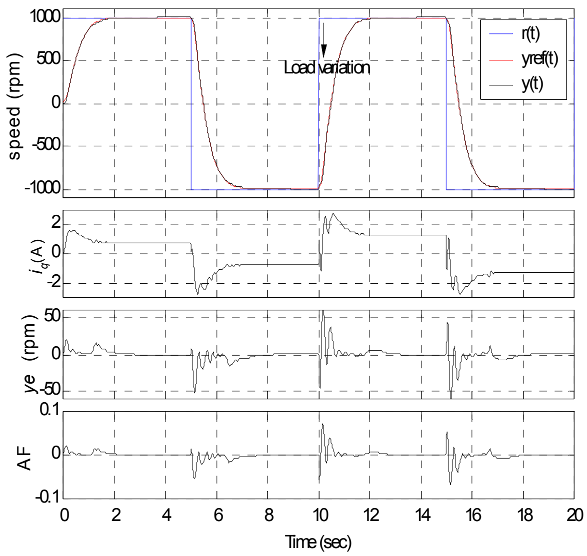


Figure (8) Control system response using FMRLC

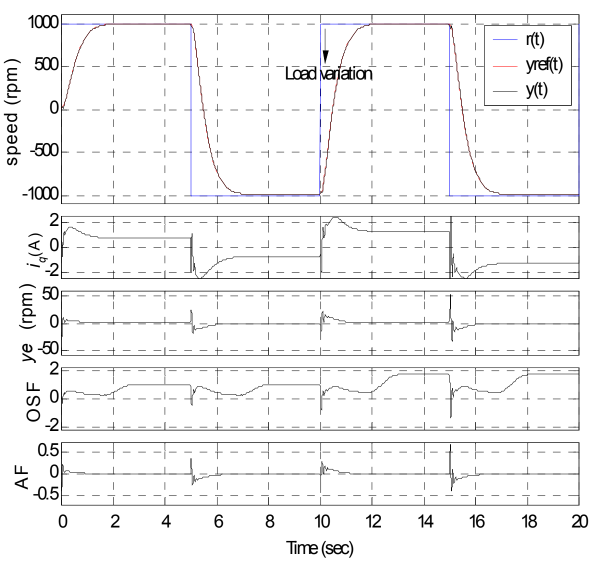


Figure (9) Control system response using FMRSTC

The speed curves of figures (8) and (9) show that the newly developed algorithm has faster adaptation capabilities. It can be noted also that the AF for the FMRLC is active indefinitely, which indicates that the learning process is not 100% efficient. This is due to the mismatch between the inverse model and the inverse characteristics of the induction motor. However, this does not affect the performance of the system but, in fact, indicates less effectiveness of the learning process.

6. Experimental Results

In addition to simulations, experiments have been conducted to verify the proposed algorithm and to compare it with the other two algorithms (FSTC and FMRLC). An induction motor speed control system was used. The motor parameters are given in table (3). Indirect field oriented control scheme was used. The controller was applied in the speed control loop. Figure (10) shows the control test bench where DSPACE DS1102 was used as the control board. The speed operating range was set to 1000 rpm to avoid the saturation of the inverter.

Figures (11), (12) and (13) show the response for the system when FSTC, FMRLC, and FMRSTC are applied respectively. The measured results show very close agreement with the ones obtained by simulations.

Figures (12) and (13) show that the FMRSTC has less error than FMRLC. The maximum speed error between the machine and the reference model is less than 60 rpm while in FMRLC it was less than 100 rpm. The OSF pattern changed at transient after adding the load to keep the transient performance constant. The FMRLC has similar performance but much slower which results in some oscillation around the reference model at transient.

Table (3) induction motor parameters.

Description	**Value**	**Units**
Rated Power	0.5	Hp
Rated current	1.3	Amp
Rated speed	1500	rpm
Rated line voltage	208	Volts
Stator inductance	0.397416	H
Rotor inductance	0.378417	H
Mutual inductance	0.372084	H
Stator resistance	9.652065	Ω
Rotor resistance	0.378417	Ω
Moment of inertia	0.00439811	kg-m^2
viscous damping	0.00028587	N-m-s
Number pole pairs	2	

The oscillation was a result of the high adaptation gain. To reduce the oscillation, lower adaptation gain should be used. Lower adaptation gain reduces the system performance at the transient.

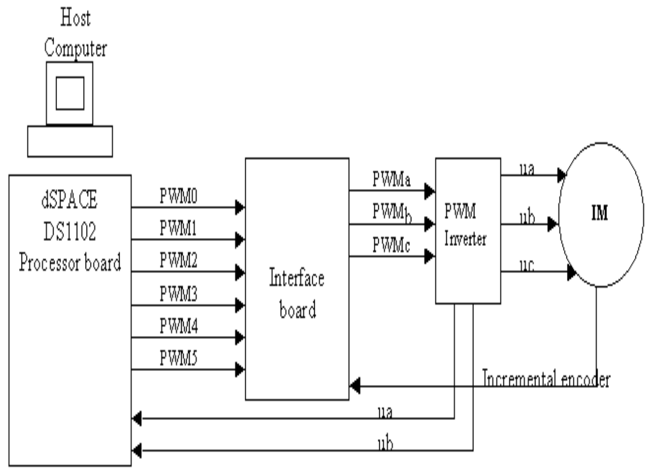


Figure (11) control test bench.

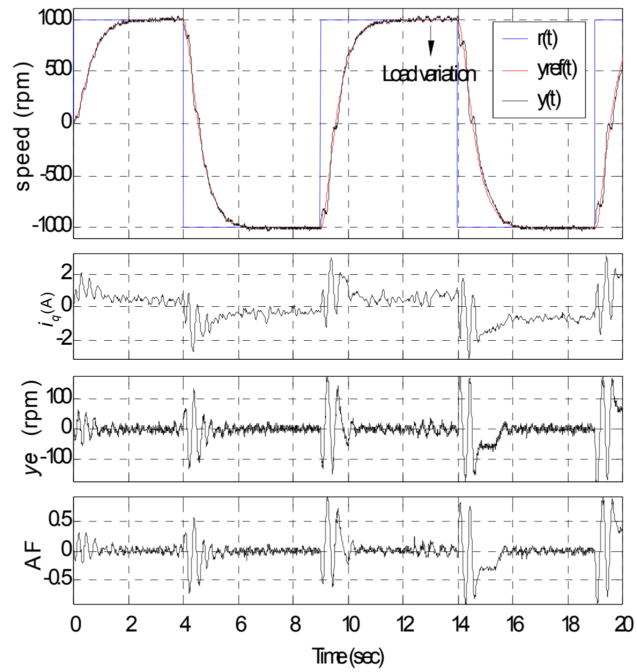


Figure (12) Control system response using FMRLC

7. CONCLUSION

An algorithm simpler and more effective than FMRLC and FSTC adaptation algorithms was developed and presented. A comparison between the newly developed algorithm and the other two algorithms has been introduced. The results show that the newly developed algorithm has a faster adaptation performance than FMRLC as well as simpler adaptation algorithm. The results for both the simulation and real time implementation confirm the superiority of the FMRSTC.

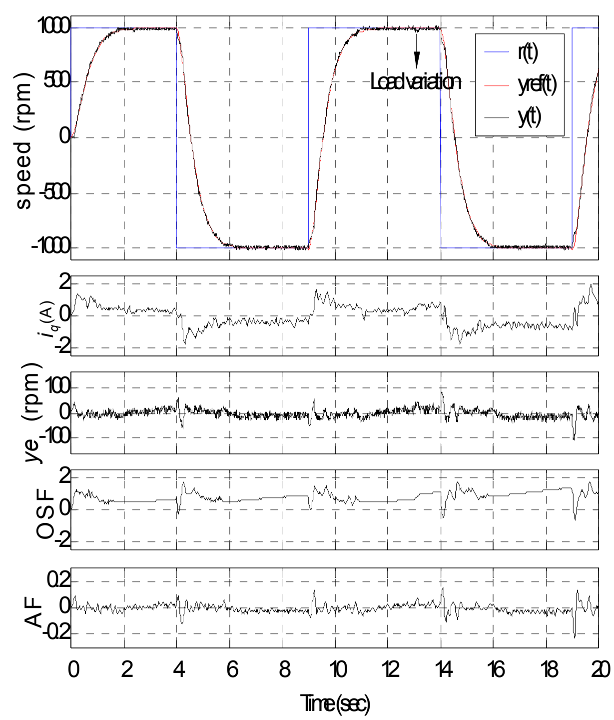


Figure (13) Control system response using FMRSTC

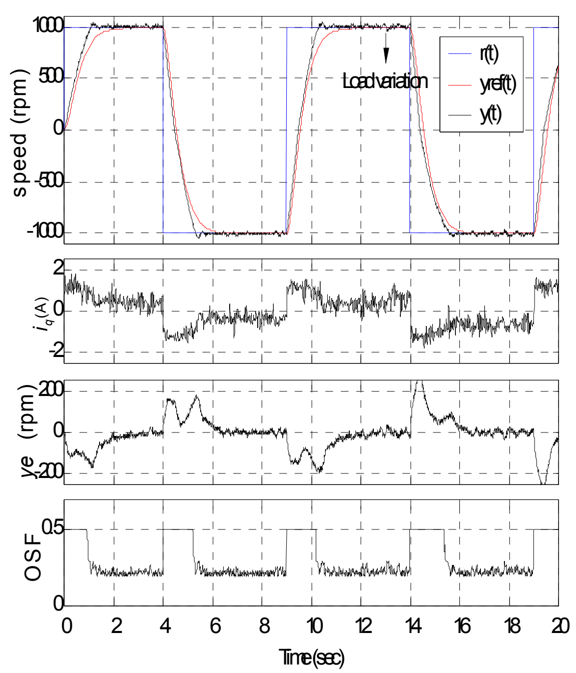


Figure (11) Control system response using FSTC

References

[1] Chia-Ju Wu, Guan-Yu Liu, “Design of Fuzzy Logic Controllers Using Genetic Algorithms” IEEE Internationa Conference On Systems, Man, and Cybernetics, Vol. 6, 1999, pp. 104-9.

[2] Seihwan Park, H. Lee-Kwang, “Designing Fuzzy Logic Controllers by Genetic Algorithms Considering Their Characteristics” Congress on Evolutionary Computation, Vol. 1, 2000, pp. 683-90.

[3] Faa-Jeng Lin, Rong-Jong Wai, “Hybrid Control Using Recurrent Fuzzy Neural Network for Linear-Induction Motor Servo Drive” IEEE Transaction on Fuzzy Systems, Vol. 9, No. 1, Feb 2001, pp. 102-9.

[4] Want Xiucai, Xiang Shiwu, Zhang Yong, Leng Jing “ The Design of a Self- Adaptive Fuzzy Control system” IEEE International Conference on Intelligent Processing Systems, Vol. 1, Oct. 1997, pp. 305-308.

[5] Rajani K. Mudi and Nikhil R. Pal “ A Robust Self-Tuning Scheme for PI- and PD-Type Fuzzy Controller” IEEE Transaction on Fuzzy Systems, Vol. 7, No. 1, Feb 1999, pp. 2-16.

[6] Layne, J., Passino K. M “Fuzzy Model Reference Learning Control for Cargo Ship Steering” IEEE Control Syst. Mag., Vol. 13, No. 6, Dec. 1993, pp. 23-34.

[7] Lavne J., Passino K., Yurkvoich S. "Fuzzy model learning control for antiskid braking systems," IEEE Trans. Contr. Syst. Techno., vol. 1, No. 2, June 1993, pp. 122-129.

[8] Layne J. and Passino K., "Fuzzy model reference learning control" In Proc. 1st IEEE Conference Control Applications, Dayton, OH, May/June 1995, pp. 624-632.

[9] Raul Ordonez, Jon Zumberge, Jeffrey T. Spooner, Kevin Passino “Adaptive Fuzzy Control: Experments and Comparative Analyses” IEEE Transaction on Fuzzy Systems, Vol. 5, NO. 2, May 1997, pp. 167-187.

[10] Silvaneto JL., Le-Huy H. “A Fuzzy Rule Changing Algorithm for an Adaptive Fuzzy Controller” ISIE 97: Proceedings of the IEEE International Symposium on Industrial Electronics: Vol. 3 Jul. 1997, pp. 1157-1161.

[11] Mayhan P., Washington GN. “Fuzzy Model Reference Learning Control: A new Control Paradigm for Smart Structures” Smart Materials and Structures, Vol. 7, Aug 1998, pp. 874-884. Conference, 1999, pp. 637-639.

[12] Procyk TJ., Mamdani EH. “A Linguistic Self-Organizing Process Controller” *Automatica,* Vol. 15, 1979, pp. 15-30.

[13] Holland J. H. “Application in Natural and Artificial Systems” Ann Arbor, MI: Michigan Press, 1975.