A Robust Sensorless Vector Control System for Induction Motors

Sung-Hoe Huh*, Ick Choy**, Gwi-Tae Park*

* Department of Electrical Engineering, Korea University SungBuk Ku, Seoul, Korea
** Intelligent System Control Research Center, KIST, Seoul, Korea

Abstract – In this paper, a robust sensorless vector control system for induction motors with a speed estimator and an uncertainty observer is presented. At first, the proposed speed estimator is based on the MRAS(Mode Referece Adaptive System) scheme and constructed with a simple fuzzy logic(FL) approach. The structre of the proposed FL estimator is very simple. The input of the FL is the rotor flux error difference between reference and adjustable model, and the output is the estimated incremental rotor speed. Secondly, the unmodeled uncertainties such as parametric uncertainties and external load disturbances are modeled by a radial basis function network(RBFN).
In the overal speed control system, the control inputs are composed with a norminal control input and a compensated control input, which are from RBFN observer output and the modeling error of the RBFN, repectively.
The compensated control input is derived from Lyapunov unction approach. The simulation results are presented to show the validity of the proposed system.

1.Introduction

In the field oriented speed control systems for induction motors, there are two main issues, which are the speed estimation and the robust performances to the lumped uncertainties, repectively[1].
The accurate speed information is necessary to obtain high performance torque and speed control characteristics. The speed information is obtained from mechanical sensors such as resolvers or pulse encoders which are directly coupled with rotor of induction motors.
However, these sensors are usually expensive, highly sensitive to the experimental environment, bulky and reduce the control performance due to their limitation of the resolutions. Therefore, various speed estimation methods are developed to replace the mechanical sensors in recent years.
Among them, the MRAS based schemes are prefered to any ther approaches because of their simplicity and the proven stability[3]. The MRAS schemes, proposed by C. Schauder[4], F. Z. Peng[5] are based on the idea of comparing two different models. One is reference model which doesn't include rotor speed, and the other is adjustable model which includes estimated rotor speed.
The difference between two outputs is due to the incorrect estimated speed, so that an approapriate adaptiation law can be used to estimate the correct rotor speed. In spite of their simplicity, there are some disadvantages such as incorrect speed estimation in the low speed area, high sensitivity to the motor parameters, and, especially, the estimator gains must be tuned along the speed area and system parameters variation.

The fundamental advantages of the fuzzy logic controllers over the conventional systems are less dependence of the mathematical model and capability of converting a set of linguistic rules into control strategy.
Recently, soft computing methods such as fuzzy logic(FL) and neural networks have been attractive in many fields of industrial applications. [6]-[8]. In this paper, FL speed estimator is proposed to overcome the difficulties of the conventional MRAS scheme. The structre of the proposed FL is very simple. The input of the FL is the rotor flux error difference between reference and adjustable model, and the output is the estimated incremental rotor speed.

Another issue in the field oriented speed control systems is the robust performances to the lumped uncertainties such as external disturbances, nolinearities, and parametric uncertainties. Recently, many researches have been developed to deal with them by applying the adaptive fuzzy logic, fuzzy neural networks, and recurrent fuzzy neural network, etc[1], [9]-[10].

Radial basis function networks(RBFN) is an architecture of the instar-outstar model[11] and constructed with a input, output and hidden layers of normalized Guassian activation functions. Because the RBFN can be used for universal approximator like a fuzzy and neural systems, it has been introduced as one possible solution to the real multivariate interpolation problem. In this paper, the RBFN is used to deal with the lumped uncertainties in the sesorless vector control system. Moreover, considering the modeling error of RBFN, the compensated control input is derived for the Lyapunov function to be stable.
The contents of this paper are as followings,
Firstly, a brief description of the speed control systems is described, and secondly, MRAS speed estimation approach including FL estimator is presented. The next part, RBFN uncertainties observer is described, and the last section, the simulation results are presented to verity the effectiveness and usefulness of the proposed algorithm.

2.Speed Control System

The block diagram of the indirect vector control system is shown in Fig. 1, and its simplified diagram with IP speed controller is shown in Fig. 2. In the sesorless vector control systems, the output of the IP speed controller is a estimated value. From the Fig. 2, the following equations (1), (2) are obtained.

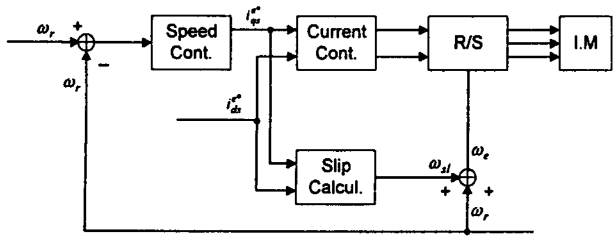

Fig. 1. Indirect Vector Control System.

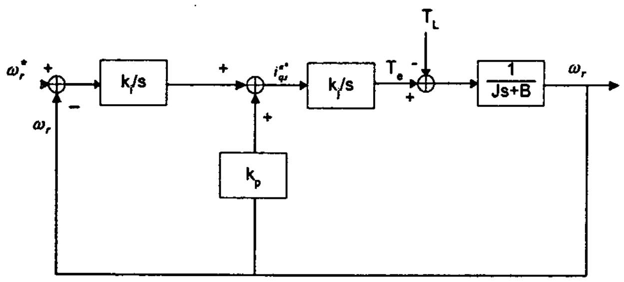

Fig. 2. Simplified IP Speed Control System.

$$\dot{\hat{\omega}}_r = -\frac{B}{J}\hat{\omega}_r + \frac{k_j}{J} i_{qs}^{e*} - \frac{1}{J}T_L \quad (1)$$

It could be expressed by state variable form as following.

$$\dot{X}_n = A_n X_n + B_n U_n + C_n T_L \quad (2)$$

where, $X_n = [\hat{\theta}_r \quad \hat{\omega}_r]^T$, $A_n = \begin{bmatrix} 0 & 1 \\ 0 & -B/J \end{bmatrix}$

$$B_n = \begin{bmatrix} 0 \\ k_j/J \end{bmatrix},\ U_n = i_{qs}^{e*},\ C_n = \begin{bmatrix} 0 \\ -1/J \end{bmatrix}$$

The above equations are expressed by nominal values. But in practical cases, the external disturbance and parametric variation and any other uncertainties could exist.
Considering the unmodeled uncertainties, the following equations could be obtained.

$$\dot{X}_q = (A_n + \Delta A_n)X_q + (B_n + \Delta B_n)U_q + C_n T_L + \varepsilon \quad (3)$$

where the ε is the unmodeled uncertainties.

Equation (3) could be expressed to the following form.

$$\dot{X}_q = A_n X_q + B_n U_q + \delta \quad (4)$$

where $\delta = (\Delta A_n X_q + \Delta B_n U_q + C_n T_L + \varepsilon)$

From equation (4), if we know the exactly the uncertainties, the perfect control input could be obtained as followings.

$$U_q^P = B_n^{-1}[\dot{X}_d - A_n X_q - \delta + K E_x] \quad (5)$$

where $E_x = X_d - X_q$

From the equation (4) with (5),

$$\dot{E}_x + KE_x = 0 \quad (6)$$

3. Speed Estimator

3.1 MRAS Approach

In the MRAS speed estimation method, two models are required, whose outputs are to be compared. One is voltage model (or stator equation) and the other is current model (or rotor equation). Because the voltage model doesn't include rotor speed, it may be regarded as a referecnce model and the other may be regarded as an adjustable model, which includes rotorspeed. The error between two models can be used to derive a suitable adaptation law, which generates the estimated rotor speed for the adjustable model. The equations (7) and (8) are the stator equation (or reference model) and the rotor equation (or adjustable model), respectively. It is convenient to express the stator and rotor equations in stationary frame because the terminal voltages and currents are sensed in the stator.

$$p\begin{bmatrix} \lambda_{dr}^s \\ \lambda_{qr}^s \end{bmatrix} = \frac{L_r}{L_m}\left[\begin{bmatrix} v_{ds} \\ v_{dq} \end{bmatrix} - \begin{bmatrix} R_s + \sigma L_s p & 0 \\ 0 & R_s + \sigma L_s p \end{bmatrix}\begin{bmatrix} i_{ds}^s \\ i_{qs}^s \end{bmatrix}\right] \quad (7)$$

$$p\begin{bmatrix} \lambda_{dr}^s \\ \lambda_{qr}^s \end{bmatrix} = \begin{bmatrix} -1/T_r & -w_r \\ w_r & -1/T_r \end{bmatrix}\begin{bmatrix} \lambda_{dr}^s \\ \lambda_{qr}^s \end{bmatrix} + \frac{L_m}{T_r}\begin{bmatrix} i_{ds}^s \\ i_{qs}^s \end{bmatrix} \quad (8)$$

For the purpose of deriving an adaptation mechanism, the following error equations can be obtained from (7) and (8). In general, rotor speed in (8) is a time varying value and therefor, the adjustable model is a linear time varying system. However, we treat the rotor speed as a constant during the adaptation processing.

$$p\begin{bmatrix} \varepsilon_d^s \\ \varepsilon_q^s \end{bmatrix} = \begin{bmatrix} -1/T_r & -\omega_r \\ \omega_r & -1/T_r \end{bmatrix}\begin{bmatrix} \varepsilon_d^s \\ \varepsilon_q^s \end{bmatrix} + \begin{bmatrix} -\hat{\lambda}_q^s \\ -\hat{\lambda}_q^s \end{bmatrix}(\omega_r - \hat{\omega}_r) \quad (9)$$

where, $\varepsilon_{dq}^s = \lambda_{dq}^s - \hat{\lambda}_{dq}^s$

The equation (9) can be expressed as following.

$$p[E] = AE - W \quad (10)$$

The object of the above error equations is the derivation of an adaptation mechanisim for (9) to be stable. For the matrix A is SPR(Strictly Positive Real) Hermitian, an adaptive law can be derived to satisfy the following *Popov*'s criterion which required a finite negative limit on the inner product of the input and output of the nonlinear feedback systems.

$$\int_0^{t_1}[E]^T[W]dt \ge -r^2, \qquad \forall t_1 \ge 0 \quad (11)$$

If we let the adaptation law as (12),

$$\hat{\omega}_r = \Phi_2 + \int_0^t \Phi_1 d\tau \qquad (12)$$

The following solutions can be derived.

$$\Phi_1 = k_1(\lambda^s_{qr}\hat{\lambda}^s_{dr} - \lambda^s_{dr}\hat{\lambda}^s_{qr}) \qquad (13)$$

$$\Phi_2 = k_2(\lambda^s_{qr}\hat{\lambda}^s_{dr} - \lambda^s_{dr}\hat{\lambda}^s_{qr})$$

3.2 Fuzzy Logic Approach

In general, the MRAS speed estimation approaches are more simple than any other strategies. However, there are some difficuties in the scheme, which are strong sensitivity to the motor parameters variations and necessity to detune the estimator gains caused by different speed area. In this paper, the fuzzy logic(FL) speed estimator is proposed to reduce the difficulties. The fundamental advantage of the fuzzy logic approach over the conventional control strategies is a less dependence of the mathematical model and capability of converting a set of linguistic rules into the control strategy as known widely. The structre of the proposed FL is very simple. The input of the FL is the rotor flux-error difference between reference and adjustable models, and the output is the estimated incremental rotor speed. The rule bases are as followings.

$$R_j : if\ \Phi\ is\ A_1^j\ and\ \dot{\Phi}\ is\ A_2^j,\ then\ \Delta\hat{\omega}_r\ is\ C_j \qquad (14)$$

where, $\Phi = \lambda^s_{qr}\hat{\lambda}^s_{dr} - \lambda^s_{dr}\hat{\lambda}^s_{qr}$

Then, the estimated rotor speed could be obtained as following.

$$\hat{\omega}_r(k) = \hat{\omega}_r(k-1) + \Delta\hat{\omega}_r \qquad (15)$$

Practically speaking, because the change of flux error linearly changes the speed estimation value, the fuzzy rules (14) could be constructed.

Table 1. Rule table.

DE \ E	NS	ZE	PS
NS	NS	NS	ZE
ZE	NS	ZE	PS
PS	ZE	PS	PS

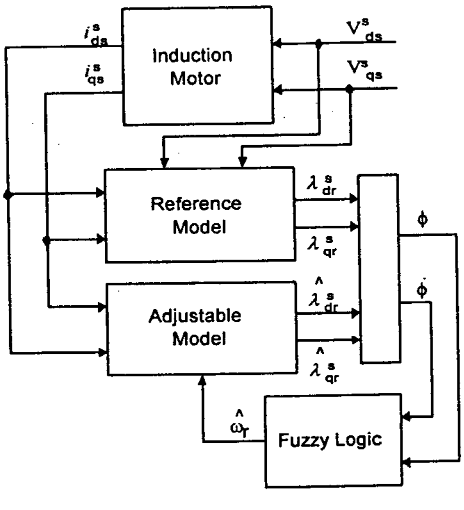

Fig. 3. The Proposed FL Speed Estimator

4. Uncertainty Observer

4.1 Radial Basis Function(RBFN)

Radial basis function networks(RBFN) is an architecture of the instar-outstar model[1] and constructed with a input, output and hidden layers of normalized Guassian activation functions. Because the RBFN can be used for universal approximator like fuzzy and neural systems, it has been introduced as one possible solution of the real multivariate interpolation problem. The RBFN is basically trained by the hybrid learning rule: unsupervised learning in the input layer and supervised learning in the output layer. The weights in the output layer can be updated by using the gradient descent method etc. The RBFN is based on the concept of the locally tuned and overlapping receptive field structure. Fig. 2 is a schematic diagram of a simple type of the RBFN which consists of one input, one output and single hidden layer.

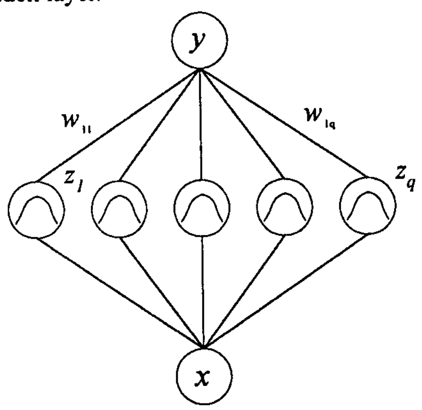

Fig. 4. Structure of the RBFN

The hidden nodes in the RBFN have normalized Gaussian activation function:

$$z_q = \xi_q(\mathbf{x}) \triangleq \frac{\phi_q(\mathbf{x})}{\sum_k \phi_k(\mathbf{x})} = \frac{\exp\left[-\left|\mathbf{x}-\mathbf{m}_q\right|^2/2\sigma_q^2\right]}{\sum_k \exp\left[-\left|\mathbf{x}-\mathbf{m}_q\right|^2/2\sigma_k^2\right]} \qquad (16)$$

where $\mathbf{x}$ is the input vector, $\mathbf{m}_q$ is the center, and σ_q is the width of radial-basis function. Hidden node q gives a maximum response to input vectors close to $\mathbf{m}_q$. Each hidden node q is said to have its own receptive field $\phi_q(\mathbf{x})$ in the input space, which is a region centered on $\mathbf{m}_q$ and σ_q are the mean and variance of the qth Gaussian function. Gaussian functions are a particular example of radial-basis functions. The output of the RBFN is simply the weighted sum of the hidden node output. In this paper, RBFN output has a simple form, multiplication of weights and hidden layer output:

$$y_i = a_i\left(\sum_{q=1}^{l} w_{iq} z_q + \theta_i\right) \qquad (17)$$

where $a_i(\cdot)$ is the output activation function and θ_i is the threshold value. The equation (17) could be expressed to simple form without θ_i, and $a_i(\cdot)$ set to 1 for single input, single output.

$$y = \sum_{q=1}^{l} w_{1q} \cdot z_q$$

$$= [w_{11}\ w_{12}\ \dots\ w_{1l}] \begin{bmatrix} z_1 \\ z_2 \\ \dots \\ z_l \end{bmatrix} = W^T \cdot Z \qquad (18)$$

The unsupervised part of the learning involves the determination of the receptive field centers $\mathbf{m}_q$ and widths σ_q, $q = 1, 2, \dots, l$. The proper centers $\mathbf{m}_q$ can be found by unsupervised learning rules such as the vector quantization approach, competitive learning rules, or simply the Kohonen learning rule:

$$\Delta \mathbf{m}_{closest} = \eta(\mathbf{x} - \mathbf{m}) \qquad (19)$$

where $\mathbf{m}_{closest}$ is the center of the receptive field closest to the input vector $\mathbf{x}$ and the other centers are kept unchanged. Then, once the receptive field centers $\mathbf{m}_q$ have been found, the widths σ_q are usually determined by an adhoc choice such as the mean distance to the first few nearest neighbors $\mathbf{m}$ (the γ-nearest-neighbors heuristic). In the simplest case, the following first-nearest-neighbor heuristic can be used:

$$\sigma_q = \frac{|\mathbf{m}_q - \mathbf{m}_{closest}|}{\gamma} \qquad (20)$$

where $\mathbf{m}_{closest}$ is the closest vector to $\mathbf{m}_q$.

The RBFN can also be trained by the error backp ropagation algorithm and becomes a purely supervised learning network. According to the chain rule, the supervised learning rule for the RBFN can be derived as

$$\Delta w_{iq} = \eta_w (d_i - y_i) z_q$$
$$\Delta \mathbf{m}_q = \eta_m \sum_i (d_i - y_i) \frac{\partial y_i}{\partial \mathbf{m}_q} \qquad (21)$$
$$\Delta \sigma_q = \eta_\sigma \sum_i (d_i - y_i) \frac{\partial y_i}{\partial \sigma_q}$$

where the derivatives $\partial y_i / \partial \mathbf{m}_q$ and $\partial y_i / \partial \sigma_q$ can be obtained using the chain rule on Eqs. (16), (17).

4.2 Modeling the Uncertainties using RBFN

As we seen from (1) – (5), the exact information of the uncertainties makes the overal system stable. In this paper, the unkown uncertainties are modeled by RBFN, which have been widely used as a nolinear function interpolator. Considering the modeled uncertainties and the error from approximator, the overal control input *Uq* could be designed as following [2],

$$U_q = U_n + U_C \qquad (22)$$

where the *Un* is the norminal control input with the RBFN observer and *Uc* is a compensated control input for the modeling error of the RBFN. Using the RBFN, *Un* could be designed as

$$U_n = B_n^{-1}[\dot{X}_d - A_n X_q - \hat{\delta} + K E_x] \qquad (23)$$

For deriving the compensated control input, *Uc*, defining a Lyapunov function as followings:

$$V_e(t) = \frac{1}{2} E_x^T E_x + \frac{1}{2\eta}(W - W^*)^T (W - W^*) \qquad (24)$$

The compensated control input can be derived for the derivative of the Lyapunove function to be a negative value. The derivative of the error equation is as following.

$$\dot{E}_x = \dot{X}_d - \dot{X}_q$$
$$= \dot{X}_d - (A_n X_q + B_n (U_n + U_c) + \delta)$$
$$= \dot{X}_d - (A_n X_q + B_n (B_n^{-1}[\dot{X}_d - A_n X_q - \hat{\delta} + K E_x] + U_c) + \delta) \qquad (25)$$

Take the time derivative of the Lyapunov equation, then

$$V_e(t) = E_x^T \dot{E}_x + \frac{1}{\eta}(W - W^*)^T \dot{W}$$
$$= -K E_x^2 + E_x[(W - W^*)^T Z + e_W - B_n U_C]$$
$$+ \frac{1}{\eta}(W - W^*)^T \dot{W}$$
$$= -K X_e^2 + X_e[e_W - B_n U_C] + E_x (W - W^*)^T Z$$
$$+ \frac{1}{\eta}(W - W^*)^T \dot{W} \qquad (26)$$

From the above two equations(25),(26), the following results can be derived.

$$\dot{E}_x = -K E_x - B_n U_C + (W - W^*)^T Z \qquad (27)$$

And the following compensating control input and updated law could be derived.

$$U_C = B_n^{-1} k \operatorname{sgn}(E_x) \qquad (28)$$

$$\Delta w_{1i} = n_w (y_d - y_1) \cdot z_i$$
$$w_{1i}(k+1) = \Delta w_{1i} + w_{1i}(k) \qquad (29)$$

where

$$z_i = \frac{F_i(x_1)}{\sum_{q=1}^{l} F_q(x_1)}$$
$$(1 \le i \le l)$$

5. Simulation Results

Computer simulations for a 2.2[kW] induction motor using the proposed fuzzy logic estimator and RBFN uncertainties observer are presented.

Table. 2. 2-pole Induction Motor Parameters

Rated volt.	150[V]	Rs	0.385[ohm]
Rated freq.	50[Hz]	Rr	0.342[ohm]
Rated curr.	14[A]	Ls	0.03257[H]
Rated torq.	14[Nm]	Lr	0.03245[H]
Base speed	1500[rpm]	Lm	0.03132[H]
		J	0.0088[Kgm]

Fig. 5 shows the resultant performance of the FL speed estimation at 22 rpm(electrical speed).

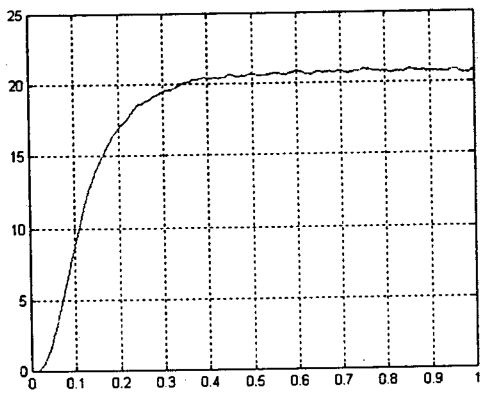

Fig. 5. Speed Estimaion(at 22rpm: electrical speed)

The speed response of the proposed control system with RBFN uncertainties observer is shwon in Fig. 6. In this simulation, the following conditions are occered at 0.5s.

J = 5 x J and **B** = 5 x B

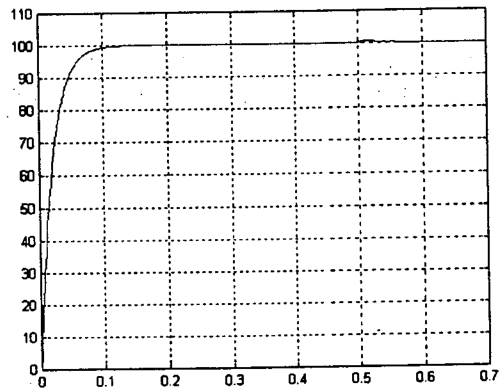

Fig. 6. Speed Control Response.

5. Conclusions

In this paper, a robust sensorless vector control system using a fuzzy logic(FL) speed estimator and a radial basis function networks(RBFN) observer for unmodeled uncertainties is proposed and verified with a computer simulation. Basically, the proposed FL estimator is MRAS based approach, and the performance of the estimator is verified over the wide speed range, and could be used in sensorless vector control system effectively. The overal control input is composed of norminal control input and compensated control input. The norminal contol input is computed with the RBFN observer, and the other is compensated control input for the modeling error of the RBFN, which is derived by Lyapunov satbility approach. The RBFN observer successfully observe the modeled uncerties. However, because the terminal voltages and currents are directly used in the estimtor, additional control strategy has to be required for compensating the noise effect. Moreover, the experimental verifications have to be required for the practical case.

References

[1]R.J. Wai, "Hybrid Control for Speed Sensorless Induction Motor Drive", *IEEE Trans. on Fuzzy Systems, Vol. 9, No. 1,* pp. 116-138, 2001.

[2] Jehudi Maes, "Speed-Sensorless Direct Torque Control of Induction Motors Using an Adaptive Flux Observer", *IEEE Trans. on I.E* vol. 36, no.3, pp. 778-785, 2000

[3]T.H.Chin, "Approaches for Vector Control of Induction motor without Speed Sensor", *IEEE IECON*, pp. 1616-1620, 1994

[4]C. Schauder, "Adaptive Speed Identification for Vector Control of Induction Motors without Rotational Transducers", *IEEE Trans. on I.A* vol. 28, no.5, pp. 1054-1061, 1992

[5]F.Z.Peng, "Robust Speed Identification for Speed Sensorless Vector Control of Induction Motors", *IEEE Trans. on I.A* vol. 30, no. 5, pp. 1234-1240, 1994

[6]P.Vas, Artificial-Intelligence-Based Electrical Machines and Drives, Oxford Univ. Press, 1999.

[7]S.A.Mir, D.S,Zinger, "Fuzzy Controller for Inverter Fed Induction Machines", IEEE Trans. on Industry Applications, Vol. 30, No. 1, Jan., pp. 78-84, 1994

[8]B.Heber, L.Xu, "Fuzzy Logic Enhanced Speed Control of an Indirect Field-Oriented Induction Machine Drive", IEEE Trans. on Power Electronics, Vol. 12, No. 5, Sep., pp. 772-778, 1997

[9]L.X. Wang, "Stable Adaptive Fuzzy Controllers with Application to Inverted Pendulum Tracking", IEEE Trans. on Sys. Man & Cyber. B, Vol. 26, No. 5, Sep., pp. 677-691, 1996

[10]F.J. Lin, W.J. Hwang, "A Supervisory Fuzzy Neural Network Control System for Tracking Periodic Inputs", IEEE Trans. on Fuzzy Systems, Vol. 7, No. 1, Feb., pp. 41-52, 1999

[11]C.T. Lin, C..S. Lee, Neural Fuzzy Systems, Prentice *Hall*, 1996.