2. FLUX ESTIMATION WITH FILTER CONCEPT

Induction motor rotor fluxes are selected to represent the desired and estimated state variable. The following two independent estimators, in the stationary frame, are generally used to derive these rotor fluxes.

2.1 Current Model of Rotor Circuit

The rotor flux estimator can be formed if the stator current and the rotor speed are measured in real time. It can be represented as follows.

$$\dot{\hat{\lambda}}^{s}_{dqr\_cm} = \left(-\frac{1}{\tau_r} I + \omega_r J\right) \dot{\hat{\lambda}}^{s}_{dqr\_cm} + \frac{L_m}{\tau_r} \dot{i}^{s}_{s} \qquad (1)$$

where, $\tau_r = L_r / R_r$, $I \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$,

$i_s = \begin{bmatrix} i_{ds} & i_{qs} \end{bmatrix}^T$,

$\lambda_{dqr\_cm} = \begin{bmatrix} \lambda_{dr\_cm} & \lambda_{qr\_cm} \end{bmatrix}^T$

2.2 Voltage Model of Stator Circuit

The voltage model utilizes the stator voltages and currents, but not the rotor velocity. It is commonly used to implement direct field orientation without speed sensors for low cost drive applications. The rotor fluxes in the stationary d-q reference frame can be obtained.

$$\dot{\hat{\lambda}}^{s}_{dqr\_vm} = \frac{L_r}{L_m}\left\{\left(v^s_s - R_s\, \dot{i}^s_s\right) - \sigma L_s \dot{i}_s\right\} \qquad (2)$$

where, $\sigma = 1 - \frac{L_m^2}{L_s L_r}$, $v_s = \begin{bmatrix} v_{ds} & v_{qs} \end{bmatrix}^T$,

$\lambda_{dqr\_vm} = \begin{bmatrix} \lambda_{dr\_vm} & \lambda_{qr\_vm} \end{bmatrix}^T$

2.3 Rotor Flux Estimation Using Filter Concept

It is well known that the current model is heavily dependent on the parameter accuracy. Similarly, though the voltage model has less sensitivity on the parameter accuracy, the low speed sensitivity is a well acknowledged limitation of this observer due to the stator resistance and the offset problem. Thus, by utilizing the current model in the low speed range and voltage model in the high speed range, more accurate rotor flux can be obtained in wide speed range. In this paper, the filter concept is used to utilize the current model in low frequency region and the voltage model in high frequency region. The resultant rotor flux is obtained from the low pass filtered current model rotor flux and the high pass filtered voltage model rotor flux. The resultant rotor flux observer is written as:

$$\hat{\lambda}^s_{dqr} = [HPF]\hat{\lambda}^s_{dqr\_vm} + [LPF]\hat{\lambda}^s_{dqr\_cm} \qquad (3)$$

where, $\hat{\lambda}^s_{dqr\_vm}$ denotes the voltage model rotor fluxes, $\hat{\lambda}^s_{dqr\_cm}$ denotes the current model rotor fluxes and $[HPF]$, $[LPF]$ denote the high pass filtering operation and the low pass filtering operation respectively. From (3), a flux angle can be detected, which enables direct field oriented control.

The filter can be designed in arbitrary order. For example, second-order filter is used, then

$$[HPF]\frac{s^2}{s^2 + K_p s + K_i} \qquad (4)$$

$$[LPF] = \frac{K_p s + K_i}{s^2 + K_p s + K_i} \qquad (5)$$

The coefficients in (4) and (5) can also be determined by the filter concept. In case of Butterworth filter, the coefficients are related as

$$K_p = \sqrt{2}\omega_c,\ K_i = \omega_c^{\,2} \qquad (6)$$

where denotes the cut-off frequency of the filter. Note that this cut-off frequency is the transition point from current model to voltage model. This flux observer has less parameter dependency in high speed region and has higher immunity to noise and measurement error in low speed region.

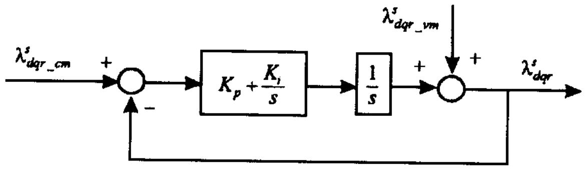


그림 1 전압모델과 전류모델을 함께 사용한 자속 추정기
Fig. 1 Flux estimator with both voltage model and current model

2.4 Estimator eigen value selection

First of all, to select the eigen values of the estimator, that of the controlled system should be predetermined from (1).

$$det(sI - A) = det\begin{bmatrix} s+\frac{1}{\tau_r} & \omega_r \\ -\omega_r & s+\frac{1}{\tau_r} \end{bmatrix} = \left(s+\frac{1}{\tau_r}\right)^2 + \omega_r^{\ 2} \quad (7)$$

$$\therefore S_1,\ S_2 = -\frac{1}{\tau_r} \pm j\omega_r \quad (8)$$

If high eigen values are selected for the estimator, the convergence rate increases but the estimator stability will be diminished. To estimate stable flux quantities, the value of the estimator should be selected according to that of the system. In this paper, variable estimator eigen value is used in designing the filter to satisfy the system requirements: i.e. rapid response and stable estimation. Thus, in this paper, the characteristic root presented by ω_c is determined as a function of the machine speed.

$$\omega_c = f(\omega_r)$$
$$= \frac{1}{\tau_r}\left(\frac{20}{\omega_{r\_\max}}|\hat{\omega}_r| + 5\right) \quad (9)$$

Eq. (9) shows that the estimated speed is used in calculating the cut-off frequency of the filter. In fig. 2, the eigen value determination algorithm is illustrated.

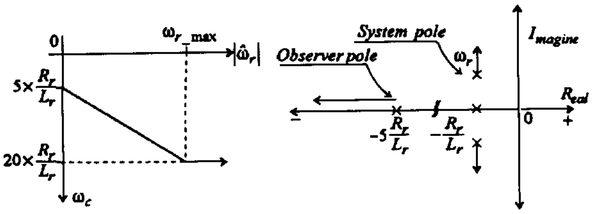


그림 2 자속 추정기의 고유치 선정

Fig. 2 Eigen value selection of the Flux Estimator

3. The Newly Proposed Speed Sensorless Control Algorithm using Neural Network Based on Extended Kalman Filter

The back-propagation algorithm can be summarized as follows.[2]

$$w_{ji}^{\ k-1,\,k}(t+1) = w_{ji}^{\ k-1,k}(t) + \Delta w_{ji}^{\ k-1,k}(t) \quad (10)$$

where, $\Delta w_{ji}^{\ k-1,k}(t) = \eta\delta_j^{\ k} o_i^{\ k-1} + \alpha\Delta w_{ji}^{\ k-1,k}(t-1)$

$\delta_j = \left(t_j - 0_j\right)f'\left(i_j^{\ k}\right)$ for the output layer

$\delta_j = f'\left(i_j^{\ k}\right)\sum_k \delta_k w_{kj}$ for the hidden layer

The Back-propagation training algorithm is an iterative gradient algorithm designed to minimize the mean square error between the actual output of a feed-forward net and the desired output. The algorithm proceeds as follows[3]:

(1) Initialize the weight vectors $w_{ji}^{\ k-1,k}(t)$ randomly.

(2) Run a training pattern through the network.

(3) Evaluate the error signals using step 1 and step 2.

(4) Update the new weight vectors $w_{ji}^{\ k-1,k}(t)$using step 3.

(5) Go to step 2 if the network has not converged.

Fig. 3 illustrates a feedforward net with only forward connections.

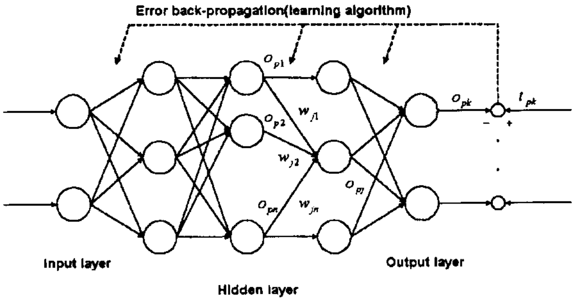


그림 3 다층 신경회로망

Fig. 3 Multi-layer Neural Network

3.1 Learning algorithm via the extended Kalman filter

We have reviewed how the back-propagation algorithm essentially implements gradient descent in sum-squared error. It should be noted, however, that the learning rate is constant, so we may have to consume more time to obtain a sufficiently convergent results, even though we can take into account a momentum term. Our main theoretical contribution here is to show that there is an efficient way of computing a time-varying learning rate.

Our learning strategy is based on regarding the learning of a network as an estimation (or identification) problem of constant parameters.

The multi-layered neural network is expressed by the following models with non-linear observation equations:

$$\hat{w}_{ji}(t+1) = \hat{w}_{ji}(t) + \zeta(t) \tag{11}$$

$$\begin{aligned} \hat{y}_j(t+1) &= h\left(w_{ji}(t+1)\right) + v(t+1) \\ &= o_j^{M}(t+1) + v(t+1) \end{aligned} \tag{12}$$

where $\{\zeta(t),\ v(t)\}$are mutually independent, zero-mean noise with covariance matrix Q and R regarded as a modeling error. Note that they can be considered pseudo-noises for tuning the gain of the extended Kalman filter. The application of the EKF to (11) and (12) gives the following real-time learning algorithms:

$$w_{ji}(t+1) = w_{ji}(t) + K_{ji}(t)\left[y_j(t) - o_j^{M}(t)\right] \tag{13}$$

$$K_{ji}(t) = \left[\frac{P_{ji}(t+|1)H_{ji}(t)^T}{H_{ji}(t)P_{ji}(t+1|t)H_{ji}(t)^T + R}\right] \tag{14}$$

$$P_{ji}(t+1|t) = P_{ji}(t|t) + Q \tag{15}$$

$$P_{ji}(t+1|t+1) = \left[I = K_{ji}(t)H_{ji}(t)^T\right]P_{ji}(t+1|t) \tag{16}$$

The filtered estimates of $w_{ji}^{k-1,\,k}$, k=M-1,...,2 at t+1 are obtained by the following extended Kalman filter:

$$\hat{w}_{ji}^{k-1,k}(t+1) = \hat{w}_{ji}^{k-1,k}(t) + \eta_{ji}^{k-1,k}(t)\delta_j^{k} o_i^{k-1} \tag{17}$$

where,

$$\eta_{ji}^{k-1,k}(t) = \left[\frac{P_{ji}^{k-1,k}(t+1/t)}{H_{ji}(t)^T P_{ji}^{k-1,k}(t+1/t)H_{ji}(t) + R}\right]$$

$$H_{ji}(t)^T = f'(j_j^{k})o_i^{k-1}$$

$$\begin{cases} \delta_j^{k} = f'(i_j^{k})^{N} \sum_{l=1}^{k+1} w_{jl}^{k,k+1}\delta_1^{k+1} & for\ \ k = M-1,\cdots,2 \\ \delta_j^{k} = f'(i_j^{k})\left(y_j - o_j^{k}\right) & for\ \ k = M \end{cases}$$

with initial conditions $\hat{w}_{ji}^{k-1,k}(0) = \overline{w}_{ji}^{k-1,k}$ and $P_{ji}^{k-1,k}(0|0) = P_{ji}^{k-1,k}(0)$

3.2 Speed sensorless control strategy

Two independent observers are used to estimate the rotor flux vectors: one based on eq. (1) and the other based on eq. (2). Since eq. (1) does not involve the speed ω_r, this observer generates the desired value of rotor flux, and eq. (2) which does involve ω_r may be regarded as a neural model with adjustable weights. The error between the desired rotor flux $\lambda^s_{dqr_vm}$given by eq. (1) and the rotor flux $\lambda^s_{dqr\_cm}$ provided by the neural model eq. (2) is used to adjust the weights, in other words the rotor speed ω_r.

The rotor speed can be derived using the Kalman based on NN. The overall blockdiagram of speed sensorless control is shown in fig. 4.

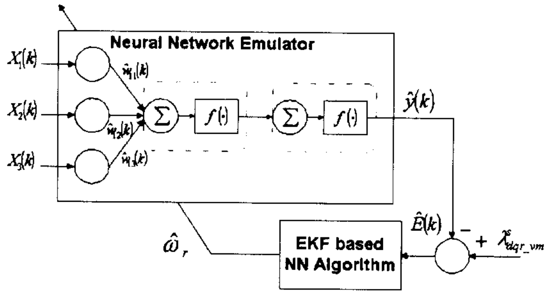


그림 4 $\hat{\omega}_r$ 추정을 위한 칼만필터 신경회로망의 구조

Fig. 4 Structure of Kalman based of NN for $\hat{\omega}_r$ estimation

The weights between neurons are tuned so as to minimize the energy function

$$E = \frac{1}{2}\varepsilon^2 = \frac{1}{2}\left(\hat{\lambda}_{dqr\_vm}(k) - \hat{\lambda}_{dqr\_cm}(k)\right)^2 \tag{18}$$

The current model can be represented as a neural model.

$$\hat{\lambda}_{dqr\_cm}(k) = x(k)w^T(k) \tag{19}$$

where,

$$x(k) = \left[\hat{\lambda}^s{}_{qr\_cm}(k)\ \ \hat{\lambda}^s{}_{dr\_cm}(k)\ \ i^s{}_{qs}(k)\right]$$

$$w(k) = \left[1 - \frac{1}{\tau_r}T_s\ \ \hat{\omega}_r T_s\ \ \frac{L_m}{\tau_r}T_s\right]$$

The weight variation is given by

$$\Delta\omega_r(k) \propto \frac{\partial E}{\partial \omega_r} = \frac{\partial E}{\partial \hat{\lambda}_{dqr\_cm}}\frac{\partial \hat{\lambda}_{dqr\_cm}}{\partial \omega_r} \tag{20}$$

The estimated rotor speed $\omega_r(t)$applied by Kalman based on NN is computed as follows

$$\omega_r^{k-1,k}(t+1) = \omega_r^{k-1,k}(t) + \eta_{ji}^{k-1,k}(t)\delta_j^{k} o_i^{k-1} \tag{21}$$

4. SIMULATION RESULTS

A 22kW 4-pole induction motor is used for the simulation and experiment simultaneously. The

표 1 유도전동기의 제정수
Table 1 Induction Motor Parameters

Rated Power	22kW	L_s	43.75mH
Rated Speed	2000rpm	L_r	44.09mH
Rated Torque	120Nm	L_m	42.1mH
R_s	0.115Ω	J_M	0.1618kgm^2
R_r	0.0821Ω	P	4

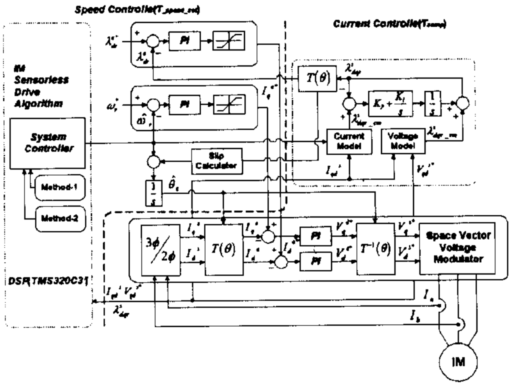

Speed Controller(T_speed_ctrl)
Current Controller(T_current)
IM Sensorless Drive Algorithm
System Controller
Method-1
Method-2
DSP TMS320C31
PI
T(θ)
Current Model
Voltage Model
Slip Calculator
3φ/2φ
T⁻¹(θ)
Space Vector Voltage Modulator
IM

그림 5 전체 제어 알고리즘의 블록선도
Fig. 5 The block diagram of the overall control algorithm

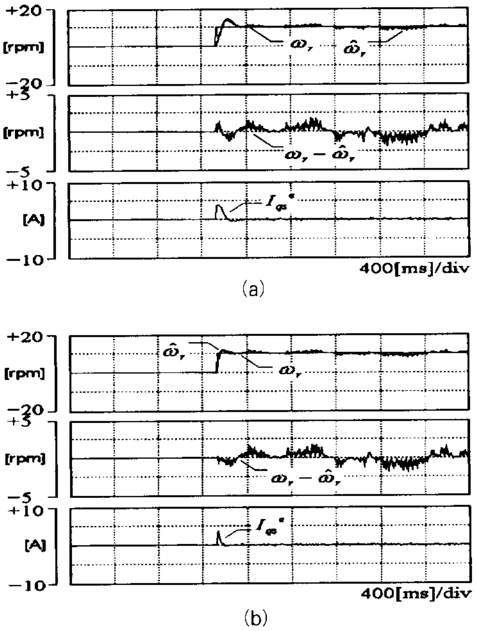


그림 6 저속영역에서 속도 추정 특성 비교
Fig. 6 The characteristics comparison of speed estimation in the low speed region. (0 → +10[rpm], 20,000 iterations)
(a) [Method-1] BP-based NN algorithm
(b) [Method-2] EKF-based NN algorithm

nominal parameters used for the simulations are given table I.

The proposed sensorless control of induction motor is shown in fig. 5.

Fig. 6 shows the step response of the conventional BP algorithm. The step response of the proposed sensorless algorithm is shown in fig. 7 when the speed reference is changed from 0[rpm] to +10[rpm]. As shown in fig. 7, we can know that the speed error is limited by 1.0% of the rating speed. Also, the proposed learning algorithm usually converges in a few iterations and the error is comparable to that of the well-known back-propagation algorithm.

Both algorithms were started from exactly the same initial weights with values ranging from 0 through 1 and the traing patterns were selected in exactly the same order. The learning rate, covariance, and momentum terms were selected to try and maximize the performance of each algorithm. For the back-propagation algorithm, the learning rate and momenterm term was 0.7 and 0.1, respectively. The proposed algorithm used R=1.0, Q=0.005 and P(0)=10.5.

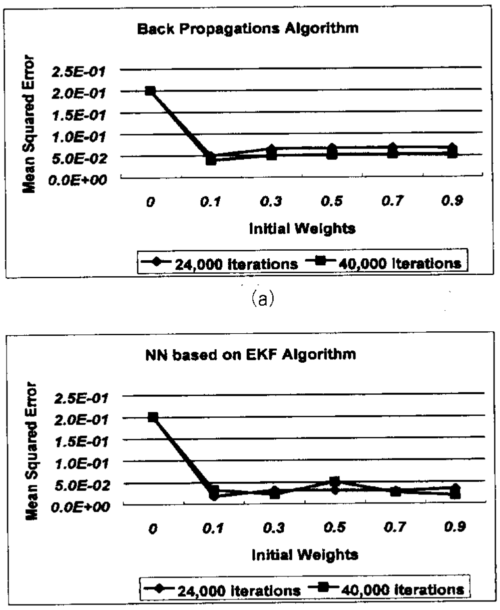


그림 7 평균제곱오차 대 초기 결선강도의 변동의 결과
Fig. 7 Results of mean squared error vs. the deviation of the initial weights.
(a) [Method-1] BP-based NN algorithm
(b) [Method-2] EKF-based NN algorithm

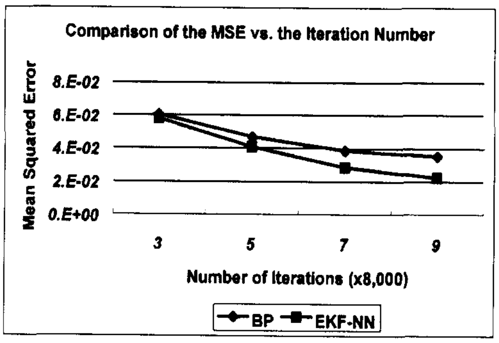


그림 8 평균제곱오차 대 반복회수의 비교

Fig. 8 Comparison of mean-squared error vs. the number of iterations

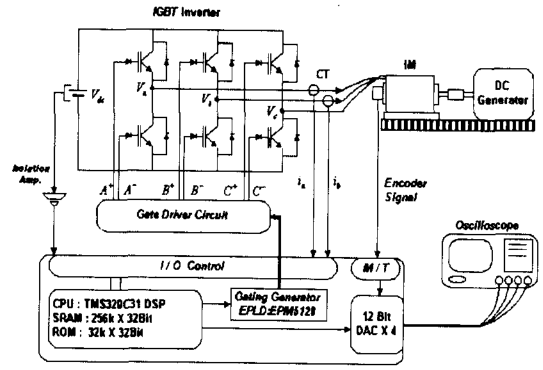


그림 10 전체 유도전동기 구동 시스템

Fig. 10 The overall IM drive system

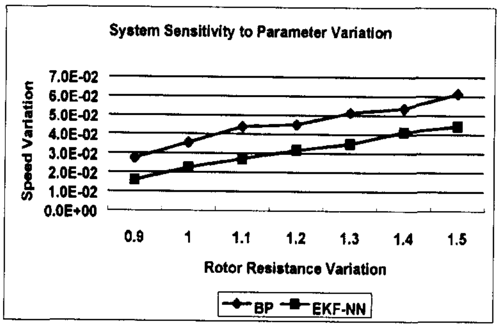
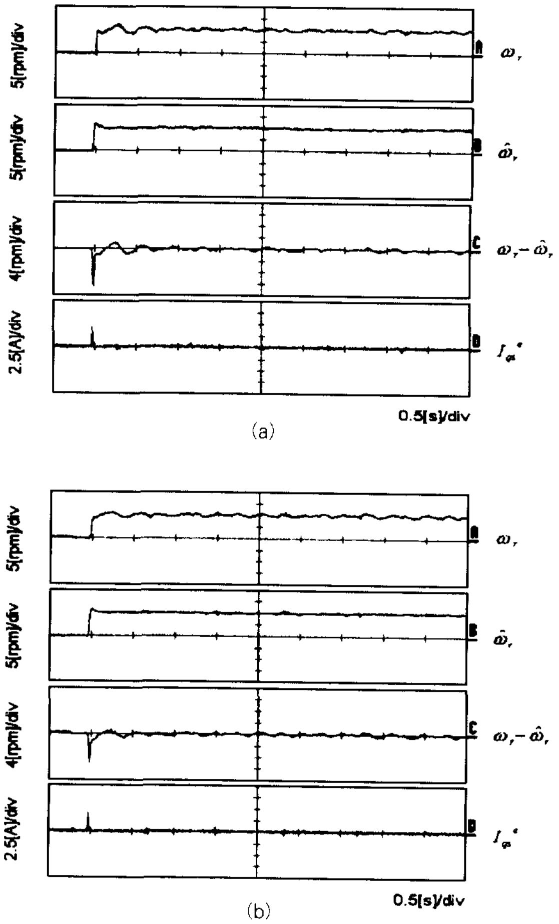


그림 9 파라미터 변동에 대한 시스템 민감도
: 속도변동 대 회전자 저항 변동

Fig. 9 System sensitivity to parameters variation
: speed variation vs. rotor resistance variation



(a)

(b)

그림 11 저속영역에서 속도추정 특성 비교

Fig. 11 The experimental waveforms of characteristics comparison of speed estimation in the low speed region (0 → +10[rpm], 20,000 iterations).
(a) [Method-1] BP-based NN algorithm
(b) [Method-2] EKF-based NN algorithm

Fig. 8 shows the mean-squared error versus the iteration number for both algorithms during training. The proposed algorithm remains below a mean-squared error of 0.07 after 5 iterations as opposed to the back-propagation algorithm at 10 iterations.

5. EXPERIMENTAL RESULTS

For the high performance IM drives, the overall IM drive system in fig. 10 is implemented with a TMS320C31 DSP control board and a PWM IGBT inverter.

For actual load emulation, the DC generator is coupled to the IM. Actual rotor speed is measured from an incremental encoder with 4096[ppr] resolution for monitoring. The sampling time of current controller loop is 250[μs] and that of the outer voltage regulating loop and speed loop is 2.5[ms]. The control algorithm

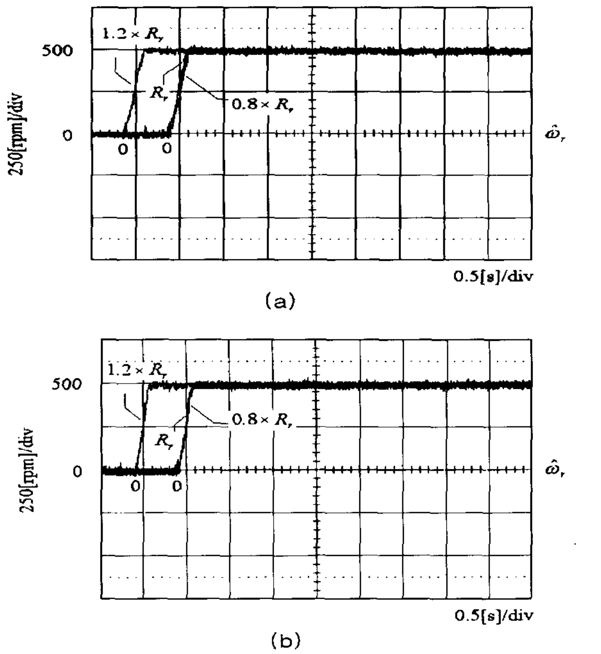


그림 12 회전자 저항의 변동에 따른 속도응답 특성

Fig. 12 The experimental waveforms of speed response characteristics according to the rotor resistance variation (0 → +500[rpm], TL=0).
(a) [Method-1] BP-based NN algorithm
(b) [Method-2] EKF-based NN algorithm

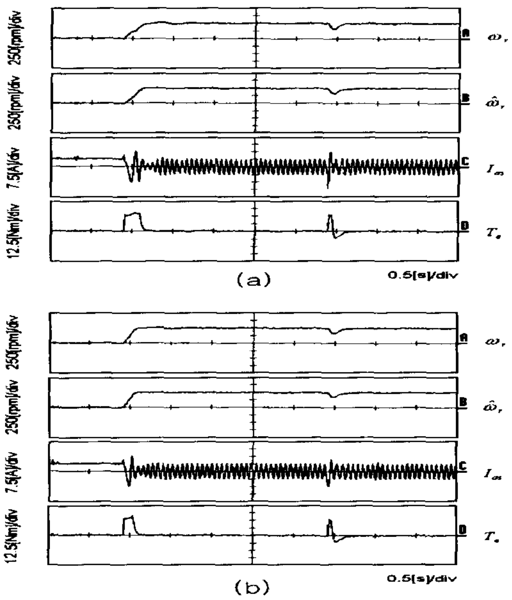


그림 13 부하토크 인가시 속도응답 특성 비교

Fig. 13 The experimental waveforms of characteristics comparison of speed response when applied to the load torque(0 → +500[rpm], TL=1[p.u.]).
(a) [Method-1] BP-based NN algorithm
(b) [Method-2] EKF-based NN algorithm

including the proposed scheme was fully implemented with the software.

Experiments are conducted to evaluate the performance of the new speed sensor elimination algorithm based on the NN. The speed step response of the proposed sensorless algorithm is shown in fig. 10 and 11 when the speed reference is changed with no load torque. As shown in the figure, the proposed algorithm works well in spite of the variation of the machine speed. It shows that the estimated speed is tracking the real one with good accuracy. Fig. 12 shows the characteristics of load torque response. As shown in those figure, the proposed algorithm works well in spite of the load torque variation.

6. CONCLUSION

We have studied learning algorithm for multi-layered feedforward type neural networks and proposed a new back-propagation algorithm that uses extended Kalman filters to identify the connection weights of the network. The simulation and experimental results can be summarized as following two points.

(a) The proposed method assures faster learning than the BP algorithm

(b) Unlike the conventional method, it doesn't require a large number of learning step.

REFERENCES

[1] Youji Iiguni, Hideaki Sakai and Hidekatsu Tokumaru, "A Real-Time Learning Algorithm for a Multilayered Neural Network Based on the Extended Kalman Filter", IEEE, Trans. on Signal Processing, vol. 40, No. 4, pp.959~966, Apr. 1992.

[2] Keigo Watanabe, Toshio Fukuda and Spyros G. Tzafestas, "Learning Algorithms of Layered Neural Networks via Extended Kalman Filters", Int. J. Systems Sci., vol. 22, No. 4, pp.753~768, 1991.

[3] Robert S. Scalero and Nazif Tepedelenlioglu, "A Fast New Algorithm for Training Feedforward Neural Networks", IEEE, Trans. on Signal Processing, vol. 40, No. 1, pp.202~210, Jan. 1992

〈저 자 소 개〉

김윤호(金倫鎬)

1949년 6월 20일 생. 1974년 서울대 공대 전기공학과 졸업. 1980년 NewYork 주립대 졸업(석사). 1987년 미국 Texas A&M 대학 졸업(박사). 현재 중앙대 공대 전기공학과 교수. 당 학회 부회장.

최원범(崔源範)

1960년 9월 22일생. 1982년 중앙대 전기공학과 졸업. 1984년 중앙대 대학원 전기공학과 졸업(석사). 1992년 중앙대 대학원 전기공학과 졸업(박사). 1994~1995년 중앙대 전기공학과 강사. 1996~현재 여주대학 자동차과 조교수.

국윤상(鞠潤相)

1972년 2월 26일생. 1994년 중앙대 전기공학과 졸업. 1996년 중앙대 대학원 전기공학과 졸업(석사). 1999년 동 대학원 졸업(박사).