

MRAS-NN을 이용한 IPMSM 드라이브의 속도와 위치 추정

이홍균, 이정철, 정택기, 이영실, 정동화
국립순천대학교 정보통신공학부

Speed and Position Estimation of IPMSM Drive using MRAS-NN

Hong-Gyun Lee, Jung-Chul Lee, Taek-Gi Jung, Young-Sil Lee, Dong-Hwa Chung
School of Information & Communication Engineering, Sunchon National Univ.

ABSTRACT

This paper combines the adaption of MRAS with the ability of NN for better modeling of nonlinear system. It presents an MRAS using an NN in the adaption mechanism. The technique is applied to a IPMSM drive. The torque constant and stator resistance variations on the speed and position estimations over a wide speed range has been studied. The NN estimators are able to track the varying parameter of different speeds with consistent performance. The validity of the proposed estimator is confirmed by the operating characteristics controlled by neural networks control.

수 있으며 모델동정과 제어방법의 구성을 일반화하고 간단하게 할 수 있는 강력한 방법이다.

본 연구에서는 IPMSM의 속도와 위치 추정을 위하여 NN에 의한 MRAS를 제시한다. NN 적응방법은 파라미터의 추정기법을 위하여 최적엔진으로 사용한다. NN의 자기학습 특성은 다양한 운전조건에서 비선형 적응설계를 용이하게 처리할 수 있다. 본 연구에서 제시한 알고리즘의 추정성능을 분석하고 그 결과를 제시한다.

2. 토크 정수와 고정자 저항 추정

추정 알고리즘은 그림 1에 나타내며 병렬 MRAS를 사용한다.

1. 서론

전동기 드라이브의 속도와 위치 센서리스 제어를 성취하기 위하여 다양한 추정기법이 개발되었다. 이러한 기법들은 전동기의 파라미터 변동을 추적하는데 필요하다. 따라서 시스템 파라미터의 정확한 인식은 어떠한 추정 알고리즘을 위하여 필요하다.

적용제어는 다양한 현대제어 기법 중에서 가장 유망한 방법으로 각광을 받았다. 이러한 적용제어는 MRAC(Model Reference Adaptive Control)[1], STC(Self Tuning Control)[2] 및 VSC(Variable Structure Control)[3] 등이 있으며 대부분 연구는 전기 드라이브 시스템의 성능을 개선할 수 있는 적응제어기이다. 적응제어는 시스템의 동정 및 상태추정은 일반적으로 시스템 모델의 구성이 설정되었다는 가정에서 수행한다. 그러나 전동기와 부하의 동특성이 이해하기 어려울 경우에는 복잡한 계산을 수행해야 한다.

이러한 문제점을 해결하기 위하여 학습과 제어의 NN(Neural Network)은 미지 또는 부정확한 동특성을 가지는 드라이브 시스템을 위한 제어기의 설계를 위하여 신경구조를 제공한다. NN은 비선형 함수를 근사화시킬

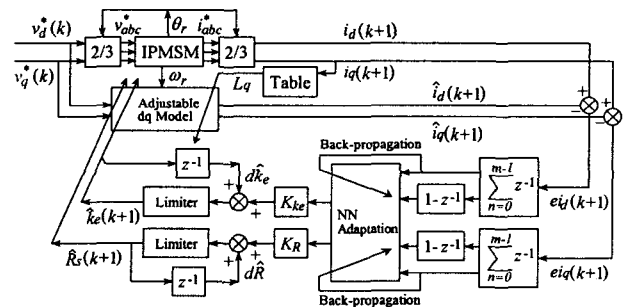


Fig. 1 MRAS-NN based estimator for torque constant and stator resistance

d, q 축의 인덕턴스의 변화는 무시하고 d 축 전류는 일정하게 제어한다. IPMSM으로 된 기준모델을 조절모델과 병렬로 연결한다. 기준모델과 조절모델 사이의 전류 오차는 퍼드포워드 신경회로망의 적응 알고리즘의 입력으로 사용되며 토크정수와 고정자 저항을 추정한다. 전류 오차는 역전과 알고리즘에서 신경망 하중을 학습하는데 사용한다. IPMSM의 $d-q$ 기준축에서 이산 방정식을 구하면 다음과 같다.

$$i_d(k+1) = i_d(k) + \frac{T_s}{L_d} [v_d(k) + L_q \omega_r(k) i_q(k) - R_s i_d(k)] \quad (1)$$

$$i_q(k+1) = i_q(k) + \frac{T_s}{L_d} [v_q(k) - L_d \omega_r(k) i_d(k) - L_q k_e \omega_r(k) - R_s i_q(k)] \quad (2)$$

여기서, T_s 는 샘플링 주기이며 $i_d(k+1)$ 과 $i_q(k+1)$ 은 $d-q$ 축 고정자 전류이며 $v_d(k)$ 와 $v_q(k)$ 는 $d-q$ 축 고정자 전압을 나타낸다. 조절모델은 기준모델인 IPMSM에서 R 과 k_e 를 \hat{R} 과 \hat{k}_e 로 대치하면 얻을 수 있다. 동특성 오차는 다음과 같은 형태로 나타난다.

$$e_{id}(k+1) = i_d(k+1) - \hat{i}_d(k+1) = -\frac{T_s}{L_d} (R_s - \hat{R}_s) i_d(k) \quad (3)$$

$$e_{iq}(k+1) = i_q(k+1) - \hat{i}_q(k+1) = -\frac{T_s}{L_q} (k_e - \hat{k}_e) \omega_r(k) - \frac{T_s}{L_q} (R_s - \hat{R}_s) i_q(k) \quad (4)$$

모든 파라미터를 정확하게 추종할 경우 전류의 오차는 영으로 된다. 실시간 구현에서 d 축 전류가 영으로 제어되면 d 축 전류오차는 정확한 추종을 하는데 충분하지 않다. 이러한 문제는 자화전류를 주입하여 해결할 수 있다. 특정한 주입의 악 영향은 동손을 증가시켜 효율과 토크를 저하시키는 결과를 초래한다.

3. 회전자 속도 및 위치 추정

동기 dq 축은 위치 정보를 가지지 못하기 때문에 $\alpha\beta$ 축 또는 고정 기준 축을 사용한다. 그림 2은 기준 $\alpha\beta$ 축에서 전압과 전류에 의해 위치와 속도를 추종하는데 사용하는 NN-MRAS를 나타낸다. 모델은 비선형이므로 여러 가지 뉴론과 히든층에 필요한 NN에 의해 회전자 위치의 이산 추정은 실시간 제어에는 부적절하다. 다른 방법은 $\alpha\beta$ 역기전력을 추정하여 위치를 얻는데 사용할 수 있다.

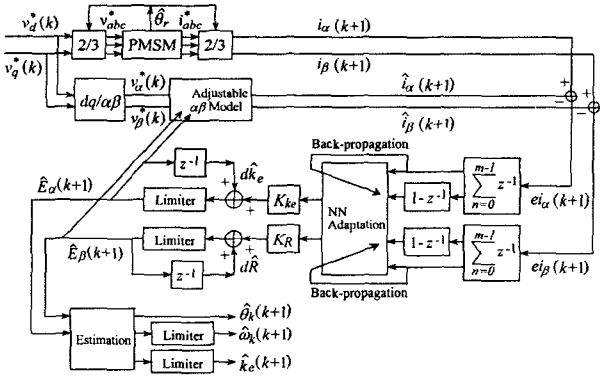


Fig. 2 MRAS NN based estimator for the rotor position and speed

고정 축에서 전압 방정식은 다음과 같다.

$$v_\alpha = R_s i_\alpha + p L i_\alpha - \hat{k}_e \omega_r \sin \theta_r \quad (5)$$

$$v_\beta = R_s i_\beta + p L i_\beta + \hat{k}_e \omega_r \cos \theta_r \quad (6)$$

여기서 $E_\alpha = -\hat{k}_e \omega_r \sin \theta_r$, $E_\beta = \hat{k}_e \omega_r \cos \theta_r$ 로 놓으면 다음 식이 된다.

$$v_\alpha = R_s i_\alpha + p L i_\alpha + E_\alpha \quad (7)$$

$$v_\beta = R_s i_\beta + p L i_\beta + E_\beta \quad (8)$$

$\alpha\beta$ 축에 대한 이산전류를 구하면 다음과 같다.

$$i_\alpha(k+1) = i_\alpha(k) + \frac{T_s}{L} [v_\alpha(k) - E_\alpha(k) - R_s i_\alpha(k)] \quad (9)$$

$$i_\beta(k+1) = i_\beta(k) + \frac{T_s}{L} [v_\beta(k) - E_\beta(k) - R_s i_\beta(k)] \quad (10)$$

여기서 $E_\alpha(k) = -\hat{k}_e \omega_r(k) \sin \theta_r(k)$, $E_\beta(k) = \hat{k}_e \omega_r(k) \cos \theta_r(k)$ 이다. 조절모델은 역기전력과 저항을 제외하면 식 (9)-(10)과 유사하다. 전류 오차는 다음과 같다.

$$e_{i\alpha}(k+1) = i_\alpha(k+1) - \hat{i}_\alpha(k+1) = \frac{T_s}{L} [\hat{E}_\alpha - E_\alpha(k) + (\hat{R}_s - R_s) i_\alpha(k)] \quad (11)$$

$$e_{i\beta}(k+1) = i_\beta(k+1) - \hat{i}_\beta(k+1) = \frac{T_s}{L} [\hat{E}_\beta - E_\beta(k) + (\hat{R}_s - R_s) i_\beta(k)] \quad (12)$$

저항과 인덕턴스가 정확하게 추측되면 전류 오차는 영으로 되고 역기전력은 정확하게 추정된다. 추정 위치, 속도 및 토크 정수는 추정 역기전력으로부터 얻는다.

$$\hat{\theta}(k+1) = -\tan^{-1} \left(\frac{\hat{E}_\alpha(k+1)}{\hat{E}_\beta(k+1)} \right) \quad (13)$$

$$\hat{\omega}_\alpha(k+1) = \frac{\hat{\theta}(k+1) - \hat{\theta}(k)}{p T_s} \quad (14)$$

$$\hat{k}_e(k+1) = \frac{\sqrt{\hat{E}_\alpha^2(k+1) + \hat{E}_\beta^2(k+1)}}{\hat{\omega}_r(k+1)} \quad (15)$$

4. NN 학습과 하중 적응

전류 노이즈의 영향을 해결하기 위하여 전류오차를 감소시킬 수 있는 LPF를 사용한다. 필터링된 전류오차와 이 오차의 변화분은 NN의 입력으로 사용한다. 시스템의 출력은 추정 파라미터의 변화분이다. 전류 LPF는 다음과 같다.

$$e_{id}^\Sigma = \frac{1}{m} \sum_{n=0}^{m-1} z^{-n} e_{id} \quad e_{iq}^\Sigma = \frac{1}{m} \sum_{n=0}^{m-1} z^{-n} e_{iq} \quad (16)$$

여기서 m 은 LPF의 부 사이클의 수이며 z 는 지연 연산자이다. NN에서 다른 두 입력은 전류오차의 변화분이다.

$$ce_{id}^\Sigma = (1 - z^{-1}) e_{id}^\Sigma \quad ce_{iq}^\Sigma = (1 - z^{-1}) e_{iq}^\Sigma \quad (17)$$

오차지수는 다음과 같이 정의한다.

$$E = 0.5 e_{id}^\Sigma \cdot e_{id}^\Sigma + 0.5 e_{iq}^\Sigma \cdot e_{iq}^\Sigma \quad (18)$$

NN 하중적응은 LSM(Least Square Method)에 기초한다.

$$v_{ij}(k+1) = v_{ij}(k) - \gamma \frac{\partial E}{\partial v_{ij}} \quad (19)$$

여기서 E 는 필터링된 전류오차의 지수이며 v_{ij} 는 신경하중의 하나이며 γ 는 학습율을 나타낸다.

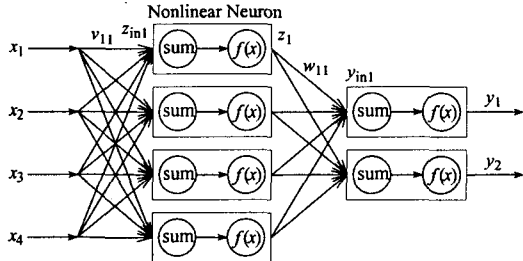


Fig. 3 Neural network weights updating

예를 들면, v_{11} 은 그림 3에서 나타낸 첫 입력신경에서 첫 히든신경까지 하중이다. 이 하중에 대한 오차지수의 미분은 다음과 같다.

$$\begin{aligned} \frac{\partial E}{\partial v_{11}} &= \frac{\partial E}{\partial e_{iq}^{\Sigma}} \frac{\partial e_{iq}^{\Sigma}}{\partial v_{11}} + \frac{\partial E}{\partial e_{id}^{\Sigma}} \frac{\partial e_{id}^{\Sigma}}{\partial v_{11}} = e_{iq}^{\Sigma} \frac{\partial e_{iq}^{\Sigma}}{\partial \hat{\omega}_r} \frac{\partial \hat{\omega}_r}{\partial v_{11}} + e_{id}^{\Sigma} \frac{\partial e_{id}^{\Sigma}}{\partial \hat{R}} \frac{\partial \hat{R}}{\partial v_{11}} \\ &= e_{iq}^{\Sigma} \frac{\partial e_{iq}^{\Sigma}}{\partial \hat{\omega}_r} \frac{\partial \hat{\omega}_r}{\partial y_{in1}} \frac{\partial z_1}{\partial z_{in1}} \frac{\partial z_{in1}}{\partial v_{11}} + e_{id}^{\Sigma} \frac{\partial e_{id}^{\Sigma}}{\partial \hat{R}} \frac{\partial \hat{R}}{\partial y_{in2}} \frac{\partial z_1}{\partial z_{in1}} \frac{\partial z_{in1}}{\partial v_{11}} \\ &= e_{iq}^{\Sigma} f'(y_{in1}) w_{11} f'(z_{in1}) x_1 + e_{id}^{\Sigma} f'(y_{in2}) w_{12} f'(z_{in1}) x_1 \end{aligned} \quad (20)$$

여기서 $\partial e_{iq}^{\Sigma} / \partial \hat{\omega}_r$, $\partial e_{id}^{\Sigma} / \partial \hat{R}$ 는 시스템의 Jacobean 함수이며 이는 신경회로망에서 $\partial \hat{\omega}_r / \partial e_{iq}^{\Sigma}$, $\partial \hat{R} / \partial e_{id}^{\Sigma}$ 에 의해 계산할 수 있다. 간단하게 $\partial e_{iq}^{\Sigma} / \partial \hat{\omega}_r = 1$, $\partial e_{id}^{\Sigma} / \partial \hat{R} = 1$ 을 놓고 사용한다. $f'(y_{in1})$ 은 초기 비선형 함수의 미분이며 z_{in1} 은 첫 번째의 히든 신경의 합성출력이다. 그리고 z_1 은 첫 번째의 히든신경의 출력이며 y_{in1} , y_{in2} 는 첫째와 둘째 출력신경의 합성출력이고 x_1 은 첫 번째의 입력신경의 입력이다. 마지막으로 적용법칙을 다음과 같이 표현할 수 있다.

$$\begin{aligned} v_{11}(k+1) &= v_{11}(k) - \gamma \frac{\partial E}{\partial v_{11}} \\ &= v_{11}(k) - \gamma \left[e_{iq}^{\Sigma} f'(y_{in1}) w_{11} f'(z_{in1}) x_1 + e_{id}^{\Sigma} f'(y_{in2}) w_{12} f'(z_{in1}) x_1 \right] \end{aligned} \quad (21)$$

5. 시스템의 성능결과

그림 4는 PI 제어기에 의한 파라미터 응답특성이며 0.2[sec]에서 기동하여 지령속도를 500[rpm]로 스텝으로 증가시켜 운전하고 있다. 추정 파라미터를 정격 값의 50[%]로 설정하여 500[rpm] 운전 중, 0.5[sec]에서 토크 정수와 고정자 저항의 추정특성을 나타내고 있다. 그림 4(a)는 지령속도와 실제속도를 나타내며 그림 (b)는 9축 전류를 나타낸다. 그리고 그림 (c)는 토크정수를 나타내고 그림 (d)

는 고정자 저항을 나타낸다.

그림 5는 본 연구에서 제시한 신경회로망에 의한 결과로서 그림 4와 같은 조건에서 추정성능을 나타낸다. 본 연구방법이 종래의 PI 제어기 보다 추종성능이 매우 우수하게 나타낸다.

그림 6은 토크 정수와 고정자 저항의 추정성능을 비교한다. 그림 (a)와 (b)는 PI 제어기에 의한 파라미터이며 그림 (c)와 (d)는 본 연구에서 제시한 신경회로망에 의한 파라미터이다. 본 방식이 종래의 PI 제어기 보다 우수한 추정성능을 나타내고 있다.

그림 7은 본 논문에서 제시한 신경회로망에 의한 회전자 속도 추정 성능의 결과이다. 지령속도를 0.1[sec]에서 1500[rpm]로 스텝으로 증가시켜 운전 중 0.5[sec]에서 500[rpm]으로 변화시켰을 경우의 응답특성을 나타낸다. 그림 (a)는 지령속도와 실제속도를 그림 (b)는 지령속도와 추정속도를 나타낸다. 추정속도는 실제속도에 양호하게 추종되고 있다. 그림 (c)는 실제 역기전력을 그림 (d)는 추정 역기전력을 나타낸다. 지령속도의 변화에도 추종 역기전력은 실제 역기전력에 양호하게 추종되고 있다.

6. 결론

본 연구에서는 토크정수와 고정자 저항과 같은 파라미터를 위하여 신경회로망에 기초하여 추정자를 제시하였다. 신경회로망을 이용한 추정자는 다양한 동작조건에서 파라미터의 변동을 아주 정확하게 추정할 수 있었다. 그리고 본 연구에서는 IPMSM의 속도와 위치 추정을 위하여 NN에 의한 MRAS를 제시하였다. NN 적용방법은 파라미터의 추정기법을 위하여 최적엔진으로 사용하였다. 본 연구에서 제시한 알고리즘의 추정성능을 분석하고 성능을 비교하여 본 연구의 타당성을 입증하였다.

참고 문헌

- [1] S. R. Bowes and J. Li, "New robust adaptive control algorithm for high performance ac drives," IEEE Trans. IE, vol. 47, pp. 325-336, 2000.
- [2] B. Zhang, Y. Li and Y. Zuo, "A DSP based fully digital PMSM servo drive using on-line self tuning PI controller," in Proc. PIEMC 2000, vol. 2, pp. 1012-1018, 2000.
- [3] I. C. Baik, et al., "Robust nonlinear speed control of PM synchronous motor using adaptive and sliding mode control techniques," Proc. IEE Elect. Power Applicat., vol. 145, no. 4, pp. 369-376, 1997.

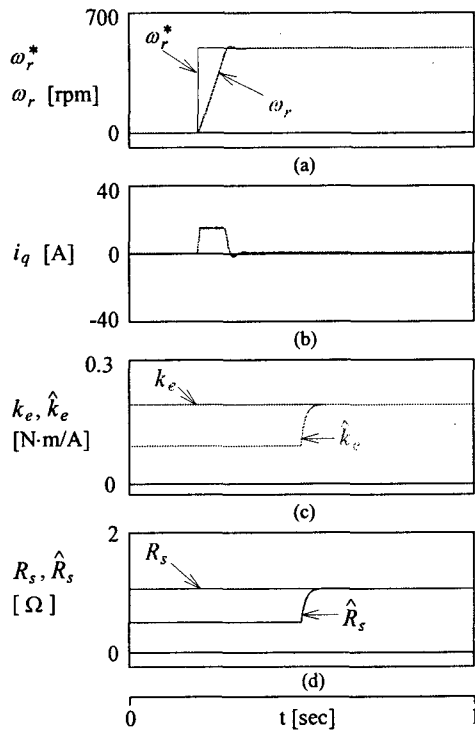


Fig. 4 Response characteristics of Parameter with PI controller

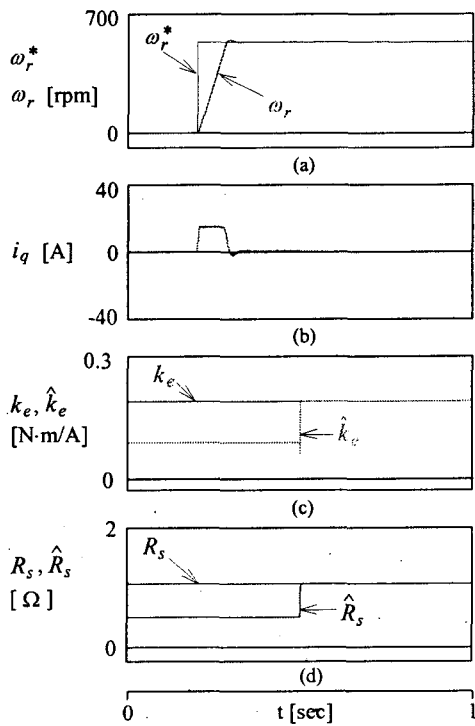


Fig. 5 Response characteristics of Parameter with Neural Networks

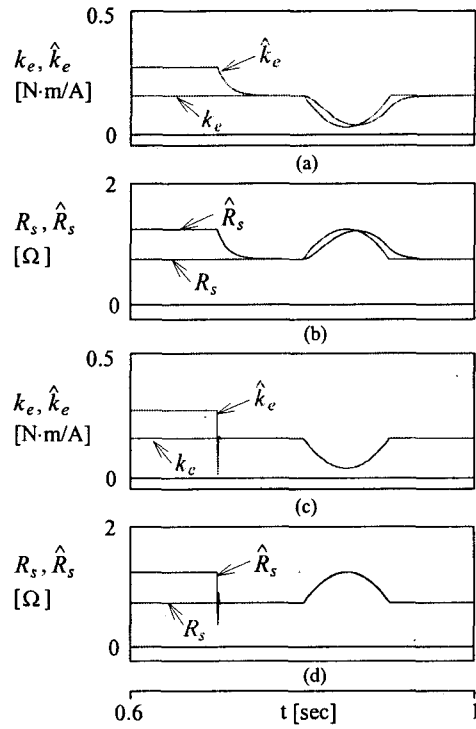


Fig. 6 Comparison of estimated characteristics in torque constant and state resistance

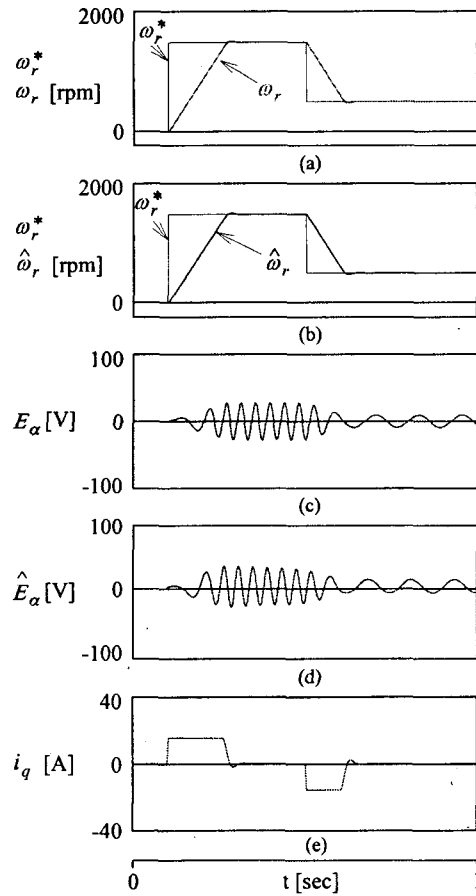


Fig. 7 Response characteristics with step command speed.