

## 신경회로망을 이용한 유도전동기의 파라미터 추정

류한민, 김성환, 박태식, 유지윤  
고려대학교 전기공학과 산업전자 연구실

### A Study on the Parameter Estimation of an Induction Motor using Neural Networks

Han-min Ryu, Seong-Hwan Kim, Tae-Sik Park, Ji-Yoon Yoo  
Department of Electrical Engineering, Korea university

#### Abstract

If there is a mismatch between the controller programmed rotor time constant and the actual time constant of motor, the decoupling between the flux and torque is lost in an indirect rotor field oriented control.

This paper presents a new estimation scheme for rotor time constant using artificial neural networks. The parameters of induction motor model organize 2 layer neural to be weight between neuron, which is proposed new in this paper. This method makes networks simple, so it brings not only the improvement in speed but simplification in calculation. Furthermore, it is possible to estimate rotor time constant real time through on-line learning without using off-line learning .

The digital simulation and the experimental results to verify the effectiveness of the new method are described in this paper.

#### 1. 서론

최근 전력 반도체 소자의 발달과 고성능 프로세

서의 등장으로 전력 전자 기술이 비약적으로 발전하여 저 가격, 고신뢰성의 전압 및 주파수 변환 장치들이 개발됨에 따라 유도 전동기가 종전의 직류 전동기의 역할을 대신할 수 있게 되었다.

유도전동기에 대한 만족스런 정밀도, 고속 운전 특성을 얻기 위해서는 고정자에 유입되는 순시 전류의 크기 및 위상을 토크 및 자속 성분으로 분리하여 독립적으로 제어할 수 있는 벡터 제어 방식을 이용하는 것이 가장 적합하다.

벡터 제어에서는 직교 좌표 상에서 기준이 되는 고정자 전류 벡터가 기준이 되는 자속 벡터와 동기될 수 있도록 제어 입력을 인가해야 하므로 자속 위치에 대한 정확한 정보가 요구되며 이를 얻는 방법에 따라 직접 및 간접 벡터 제어로 나눌 수 있다.[1]

직접 벡터 제어는 이론적으로는 바람직한 접근이지만 실제적인 접근에서는 센서 쿼부등으로 인한 높은 가격과 저속에서의 직접 추정 에러가 문제가 되고 있다. 그래서, 실제 많은 플랜트에는 간접 벡터 제어가 적용되고 있다. 간접 벡터 제어의 가장 큰 문제점은 자기적인 포화나 온도의 변화에 의한 모터 파라미터의 변화에 민감한 것이다. 간접 벡터 제어에 핵심적인 슬립 계산 식은 회전자 시정수  $T_r$ 을 포함하고 있는데 이중 회전자 저항  $R_r$ 은 온도 변화에 따라 민감하게 변화하며 회전자 인덕턴스  $L_r$ 은 자속의 포화에

영향을 받아 변화한다. 이에 따라 실제 자속의 위치와의 오차로 벡터 제어 성능이 저하되어 과도상태에서의 진동과 정상 상태 오차를 야기시킨다. 따라서 회전자 시정수  $T_r$ 이나 또는  $R_r$ ,  $L_r$  각각에 대한 정확한 정보가 필요하며 기존에 회전자 저항  $R_r$ 을 추정하는 여러 가지 연구들이 진행중에 있다.

본 논문에서는 신경회로망을 이용하여 회전자 시정수를 실시간으로 추정함으로써 파라미터 변동에 견실한 유도전동기의 벡터 제어를 실현하는 것을 연구 목적으로 한다.

이를 위하여 신경회로망을 이용한 새로운 유도전동기 회전자 시정수 추정 방법을 제안한다. 제안된 회전자 시정수 추정 방법은 유도전동기의 자속 모델을 사용하며 기존 자속과 추정 자속의 오차를 감소시키는 방향으로 신경회로망을 역전파 방법(backpropagation)을 통하여 학습시킴으로써 정확한 회전자 시정수를 추정한다.

또한, 이론적인 분석과 디지털 시뮬레이션에 의하여 제안된 방법을 분석, 고찰하고 고성능 DSP와 2.2kW 유도전동기를 이용한 실험결과로서 제안된 방식의 효용성을 입증한다.

## 2. 신경회로망

생물계에서는 뉴런이라고 부르는 물리적인 세포들의 상호 연결을 통하여 정보를 처리한다. 신경회로망은 이러한 메카니즘을 모방하여 적응학습을 통하여 스스로 지능을 축적해 나가는 기법이다. 가장 일반적으로 사용되는 다층 신경회로망의 학습은 David E. Rumelhart가 개발한 일반화된 델타 규칙을 적용하여 이루어진다.

일반적인 다층 퍼셉트론의 학습 방법은 다음과 같다. 입력층의 각 유니트에 입력 데이터를 제시하면 이 신호는 각 유니트에서 변환되어 중간층에 전달되고 최종적으로 출력층으로 나오게 된다. 이 출력값과 원하는 출력값을 비교하여 그 차이를 감소시키는 방향으로 가중치를 조정하고 상위층에서 역전파하여 하위층에서는 이를 근거로 가중치를 조정하여 학습이 이루어진다.[2]

최급 강하법을 이용해서 일반화된 델타 규칙을 유도해 보자. 패턴  $p$ 에 대한 실제출력과 목표출력 사이의 오차의 제곱은 다음과 같다.

$$E_p = \frac{1}{2} \sum_j (t_{pj} - o_{pj})^2 \quad (1)$$

$t_{pj}$  :  $p$ 번째 목표출력 패턴의  $j$ 성분,  $j$ 층은 출력층

$o_{pj}$  :  $p$ 번째 입력패턴으로부터 네트워크가 계산한 실제 출력  $j$ 성분

이때, 이 오차의 제곱  $E_p$ 를 감소시키는 방향으로 가중치  $W_{ji}$ 를 조절하기 위해 최급 강하법을 사용한다. 즉,  $W_{ji}$ 의 수정량인  $\Delta W_{ji}$ 로서  $-\frac{\partial E_p}{\partial W_{ji}}$ 를 사용한다. 이때, 가중치  $W_{ji}$ 의 조정을 위하여  $p$ 번째 패턴의 학습으로부터 사용되는 가중치  $W_{ji}$ 는 식 (2)에 의해 수정된다.

$$W_{ji} = W_{ji} + \Delta_p W_{ji} \quad (2)$$

델타 규칙에 의해 가중치의 변화량을 구해보면 연쇄법칙(chain rule)에 의해 식(3)과 같다.

$$\Delta_p W_{ji} = -\frac{\partial E_p}{\partial W_{ji}} = -\frac{\partial E_p}{\partial net_{pj}} \cdot \frac{\partial net_{pj}}{\partial W_{ji}} \quad (3)$$

$$net_{pj} = \sum_j W_{ji} o_{pi}$$

식 (3)을 각각에 대하여 계산해보면 다음과 같다.

$$\frac{\partial net_{pj}}{\partial W_{ji}} = \frac{\partial}{\partial W_{ji}} \left[ \sum_m W_{jm} o_{pm} \right] = o_{pi}$$

$$-\frac{\partial E_p}{\partial net_{pj}} = \delta_{pj} \quad (4)$$

$$-\frac{\partial E_p}{\partial W_{ji}} = \delta_{pj} o_{pi}$$

layer가 출력층의 유니트인 경우엔 통상의 델타 규칙의 경우와 동일하므로 다음과 같이 구할 수 있다.

$$\delta_{pj} = -\frac{\partial E_p}{\partial net_{pj}} = -\frac{\partial E_p}{\partial o_{pj}} \cdot \frac{\partial o_{pj}}{\partial net_{pj}} \quad (5)$$

$$= (t_{pj} - o_{pj}) f'_j (net_{pj})$$

layer가 출력층이 아닌 경우엔 다음과 같이 구해진다.

$$\delta_{pj} = -\frac{\partial E_p}{\partial net_{pj}} = -\frac{\partial E_p}{\partial o_{pj}} \cdot \frac{\partial o_{pj}}{\partial net_{pj}} \quad (6)$$

$$= \left[ \sum_k \delta_{pk} W_{kj} \right] f'_j (net_{pj})$$

식 (5)와 (6)을 이용하여 각 층의 모든 layer에 대한  $\delta_{pj}$ 를 재귀적으로 계산할 수 있다. 그리고 이로부터 모든 유니트에 대한 가중치의 수정량은 각각  $\Delta_p W_{ji} = \eta \delta_{pj} o_{pi}$ 로 구해진다.

### 3. 파라미터 추정기의 구성

제안된 추정 기법은 신경회로망 학습시 오프-라인 학습 방법을 사용하지 않고 회전자 시정수 추정의 시작과 동시에 온-라인 학습을 통하여 회전자 시정수의 실시간 추정이 가능하다. 그림 3-1은 제안된 신경회로망을 이용한 회전자 시정수 추정 시스템의 블록도를 나타낸다.

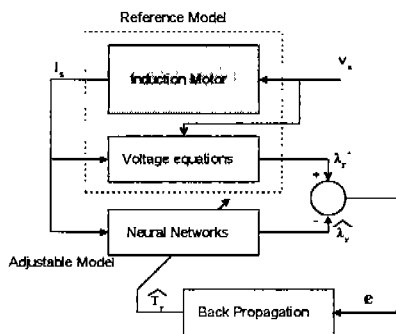


그림 3-1 : 신경망을 이용한 회전자 시정수 추정기 구조

유도전동기의 회전자 자속 모델은 다음과 같으며 유도 전동기의 전압과 전류는 고정자측에서 측정되므로 모델을 고정자측으로 표현하는 것이 편리하다.

$$p \begin{bmatrix} \lambda_{rd} \\ \lambda_{rq} \end{bmatrix} = \frac{L_r}{L_m} \begin{bmatrix} v_{sd} \\ v_{sq} \end{bmatrix} - \begin{bmatrix} R_s + \sigma L_s p & 0 \\ 0 & R_s + \sigma L_s p \end{bmatrix} \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} \quad (7)$$

$$p \begin{bmatrix} \lambda_{rd} \\ \lambda_{rq} \end{bmatrix} = \begin{bmatrix} -1/T_r & -\omega_r \\ \omega_r & -1/T_r \end{bmatrix} \begin{bmatrix} \lambda_{rd} \\ \lambda_{rq} \end{bmatrix} + \frac{L_m}{T_r} \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} \quad (8)$$

회전자 시정수  $T_r$ 을 포함하고 있는 식(7)의 출력을 회전자 자속의 기준값으로 선택하고  $T_r$ 을 포함하지 않는 식(8)은 가중치를 조절하기 위한 신경회로망의 모델로 선택한다. 유도 전동기의 파라미터들이 정확하고, 센싱한 전동기의 속도가 정확하다면 두 모델의 출력은 부합해야한다. 그러나, 신경회로망의 모델에 들어가는 파라미터와 실제 유도 전동기의 파라미터 사이에 오차가 존재한다면 두 모델의 출력 역시 오차가 있게 된다. 그 출력 오차를 가지고 신경회로망의 가중치, 즉 유도 전동기의 회전자 시정수 값을 조절하는 것이다. [3]

가중치의 학습은 역전파 알고리즘으로 행해지며 그러기 위해서 식 (8)을 backward difference method를 이용하여 sampled-data model로 변환하면 다음과 같다

$$\begin{bmatrix} \hat{\lambda}_{rd}(k) \\ \hat{\lambda}_{rq}(k) \end{bmatrix} = W_1 \begin{bmatrix} \hat{\lambda}_{rd}(k-1) \\ \hat{\lambda}_{rq}(k-1) \end{bmatrix} + W_2 \begin{bmatrix} -\hat{\lambda}_{rd}(k-1) \\ \hat{\lambda}_{rd}(k-1) \end{bmatrix} + W_3 \begin{bmatrix} L_m i_{ds}(k-1) - \hat{\lambda}_{rd}(k-1) \\ L_m i_{qs}(k-1) - \hat{\lambda}_{rq}(k-1) \end{bmatrix} \quad (9)$$

$$W_1 = 1, \quad W_2 = T_s \cdot \omega_r, \quad W_3 = \frac{T_s}{T_r}$$

$T_s$  : 샘플링 타임

$\omega_r$  : 회전자의 전기적 속도

벡터  $X$ 를 정의하여 다시 쓰면

$$\hat{\lambda}_r(k) = W_1 X_1 + W_2 X_2 + W_3 X_3 \quad (10)$$

가 되고 이를 그림으로 나타내면 다음과 같다.

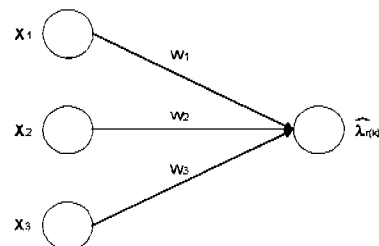


그림 3-2 : 2 layer 신경망 회로

회전자 자속의 기준값과 신경망회로의 출력사이의 에러는  $\varepsilon(k) = \lambda(k) - \hat{\lambda}(k)$ 이 되며 뉴런 사이의 가중치는 에너지 함수  $E = \frac{1}{2} \varepsilon^2(k)$ 를 최소화하는 방향으로 조절된다. 식 (9)에서 알 수 있듯이  $W_1$ 은 상수이며  $W_2$ 는 에러와 상관없이 조절되는 값이기 때문에  $W_3$ 만을 조절해나가면 된다. 식 (5)로부터

$$\delta_3 = -\frac{\partial E}{\partial W_3} = -\frac{\partial E}{\partial \hat{\lambda}_r} \cdot \frac{\partial \hat{\lambda}_r}{\partial W_3} \quad (11)$$

$$= (\lambda_r - \hat{\lambda}_r)^T \cdot X_3$$

이 되어, 결국 새로운 가중치는

$$W_3(k) = W_3(k-1) + \eta \cdot \delta_3 \quad (12)$$

$\eta$  : 학습률

이 된다. 결과적으로 추정하고자 하는 회전자 지정수는

$$\hat{T}_r = \frac{T_s}{W_3} \quad (13)$$

가 된다. 또한, 기준모델 내의 순수적분기 성분에 의하여 발생하는 문제를 해결하기 위하여 모델을 다음과 같이 수정하였다.[4]

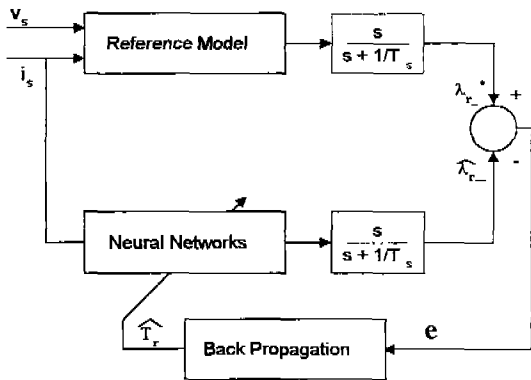


그림 3-3 : 수정된 시스템의 블록도

#### 4. 디지털 시뮬레이션

제시된 알고리즘의 타당성을 고찰하기 위해 실험에 앞서 컴퓨터 시뮬레이션을 하였다. 유도기의 기계적, 전기적 시정수 및 위치 검출기의 정도를 고려하여 속도 및 자속 제어, 전류 제어의 제어 주기를 각각 1ms, 100μs로 하여 디지털 시뮬레이션을 하였고 자속 및 파라미터 추정기의 추정 주기 역시 100μs로 설정하였다. 시뮬레이션에서의 미분방정식 계산은 Runge-Kutta 방식을 사용하여 행하였으며 미분방정식을 포함한 모든 수식 연산은 BORLAND C++로 프로그래밍 하였다. 이와 함께 그래프들은 BORLAND C++로 제작된 GRAPH.A.hpp를 이용하여 도시하였다. 시뮬레이션은 각각 기준속도 1000, 500, 50[rpm]에서 행하였으며, 그림 4.1은 500[rpm]에서의 결과이다.

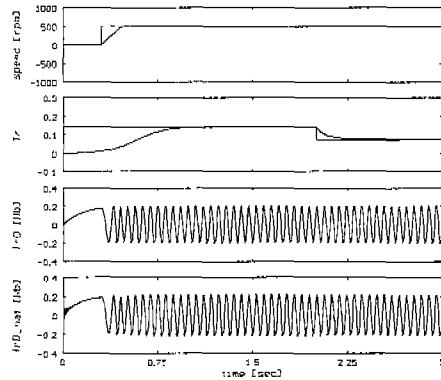


그림 4.1.  $T_r$ 의 추정 500[rpm]  
(a) 속도 응답 (b) 실제  $T_r$ 과 추정된  $T_r$   
(c) 기준 자속의 크기 (d) 추정 자속의 크기

#### 5. 실험결과 및 분석

본 논문에서 사용되는 전체 제어 알고리즘은 많은 양의 복잡한 계산을 실시간 처리해야 하기 때문에 필수적으로 고속의 프로세서를 요구한다. 따라서 본 연구에서는 고속 및 다량 연산에 있어서 최적의 프로세서인 DSP(Digital Signal Processor) TMS320C31을 하드웨어 시스템의 주 연산 및 제어기로 사용하였고, 이와 함께 전력 변환부의 고집적화를 통한 인버터의 신뢰성 향상을 실현하기 위해 IPM(Intelligent Power Module)을 이용하여 전력 변환부를 구성하였다. 시스템의 전체 구성도는 다음과 같다.