

신경회로망을 이용한 리니어 펄스 모터의 정밀 제어

Precise Control of a Linear Pulse Motor Using Neural Network

권영건, 박정일
(Young-Kun Kwon and Jung-Il Park)

Abstract : A Linear Pulse Motor (LPM) is a direct drive motor that has good performance in terms of accuracy, velocity and acceleration compared to the conventional rotating system with toothed belts and ball screws. However, since an LPM needs supporting devices which maintain constant air-gap and has strong nonlinearity caused by leakage magnetic flux, friction and cogging, etc., there are many difficulties in improvement on accuracy with conventional control theory. Moreover, when designing the position controller of LPM, the modeling error and load variations has not been considered. In order to compensate these components, the neural network with conventional feedback controller is introduced. This neural network of feedback error learning type changes the current commands to improve position accuracy. As a result of experiments, we observe that more accurate position control is possible compared to conventional controller.

Keywords : linear pulse motor, neural network, force ripple

I. 서론

최근 리니어 모터의 중요성이나 산업 현장에서의 적용 예가 증대되고 있다.[1] [2] [3] 리니어 펄스 모터 (Linear Pulse Motor)는 회전형 스텝 모터를 평면화한 것으로서 직선운동을 얻기 위하여 부가적인 기구가 필요치 않다. 회전형 모터를 이용하여 직선운동을 얻을 때는 볼스크류 등의 부가적인 기구장치가 필요하고, 부가된 장치의 백래시 등에 의해서 정밀제어를 하는데 어려움이 있고, 유지 보수를 하는데에도 더 많은 시간을 투자해야 하는 단점이 있다. 이에 비해 LPM을 이용할 경우 직선변환기구가 필요치 않게 되어 구조가 간단하고, 수리, 유지, 보수가 간편하고 에너지의 손실 또한 적다. 기계적인 백래시(backlash)가 없어 고속·초정밀 위치제어가 가능하며 진동과 소음이 적은 특성을 가지고 있어 고속 전철, 반도체 장비, 디스크 드라이브 등 각종 산업용 장비 및 사무자동화 기기에 널리 사용되고 있다. 이러한 장점이 있는 반면에 개방구조로 되어 있어서 많은 누설자속으로 인해 효율과 역률이 떨어진다. 기존의 제어방식에서는 이러한 누설자속과 마찰력 등 많은 비선형 성분과 모델링 되지 않는 성분을 포함하고 있으므로 고정밀 위치제어를 하기가 어려우며 새로운 제어 알고리즘 및 고정밀 센서가 필요하다.

리니어 펄스 모터의 구동법에는 회전형 펄스 모터에서와 같이 1상여자 방식, 2상여자 방식, 1-2상 여자 방식의 여러 방법이 있다. 최근에는 정밀제어를 위하여 이러한 방법 대신에 한 피치내에서 각 상이 90도의 위상차를 갖는 정현파의 전류를 인가해 주는 마이크로 스텝제어 방식이 사용되고 있다. 그러나 이러한 방법

에서도 기존의 제어기를 사용해서는 비선형 항을 포함한 모델링 되지 않은 부분들이 있기 때문에 힘 리플(force ripple)이 발생하는 것으로 발표되고 있다.[2] [4] [5] 오차 리플을 최소화 하고 비선형 항을 극복하기 위하여 많은 연구가 이루어져 왔는데 그 중 대표적인 것이 정확한 모델링에 의해서 비선형 동특성을 케환에 의해 선형화 시키는 방법과 [5] 학습에 의한 적응제어 방법이다.[2] [6] 정확한 모델링에 의한 방법은 구조가 달라지면 모델링을 다시 해야 하고, 모델링이 잘못되었을 경우 필연적으로 오차가 발생한다. 적응제어 방법은 변수식별에 따른 계산량의 증가로 실시간 제어에 어려움이 있으며, 상태변수가 증가함에 따라 오차도 같이 증가하는 단점이 있다.

리니어 펄스 모터의 정밀 위치제어에 신경망을 적용한 [6]의 논문에서는 본 논문에서 처럼 A,B상의 전류모양을 바꾸는 것이 아니라 제어입력을 수정하여 V/F 변환기를 통하여 A,B상의 주파수가 변화된다. 여기에서는 신경망 제어기의 학습 능력은 탁월하였지만 전체적으로 상당한 위치오차를 항상 가지고 있다. 그래서 이러한 단점을 보완하기 위하여 본 논문에서는 신경회로망을 이용하여 절대오차와 오차 리플을 감소시키는 방향으로 리니어 펄스 모터에 인가되는 전류의 모양을 직접 변화시키는 전류제어기를 설계하고자 한다. 신경회로망은 학습능력이 있으므로 입출력 패턴만으로 제어가 가능하다, 또한 병렬 분산 처리의 원리에 의해 임의의 입출력 데이터 맵핑으로 실시간 제어가 가능하다. 신경회로망을 학습하기 위한 알고리즘으로는 역전파 학습법을 이용한다. [2]의 논문은 리니어 동기 모터의 위치제어에 신경망을 적용한 예이다.

본 논문의 구성은 I장에서는 본 논문의 연구 배경 및 목적 등을 기술한 서론을, II장에서는 본 논문의 실험 대상인 리니어 펄스 모터의 기본 동작원리 및 모델링에 대해서 설명하고, III장에서는 본 논문에서 제시

접수일자 : 2000. 2. 29., 수정완료 : 2000. 6. 29

권영건 : 삼성전자 생산기술센터 자동화연구소

박정일 : 영남대학교 전자정보공학부

※ 이 논문은 1997학년도 영남대학교 학술연구 조성비에 의해서 이루어졌습니다.

하는 신경회로망을 도입한 제어기에 관해서 다루고 IV 장에서는 수행한 실험 및 측정된 데이터를 이용하여 결과를 고찰한다. 마지막으로 V장에서 결론을 맺는다.

II. 리니어 펄스 모터

리니어모터의 개요, 종류, 특징 등은 참고문헌 [1] [3] 을 참고하기 바란다. 본 장에서는 리니어 펄스 모터의 기본원리와 수학적 모델링에 대해서 알아본다.

1. 리니어 펄스 모터의 기본 원리

본 논문의 제어 대상인 하이브리드 리니어 펄스 모터는 전자석과 영구자석으로 되어있는 가동자와 치(齒)를 이루고 있는 고정자 사이의 일정한 공극을 유지하면서 자력선이 강한 텐션을 가지고 있고, 가능한 짧고 직선적인 형태로 되려고 하는 즉, 자기저항이 최소로 되려고 하는 성질에 의해 발생하는 힘을 이용하여 추력을 발생시켜 직선 운동을 한다.

본 논문에서 사용하는 리니어 모터는 2상이며 다음과 같이 바이폴러 방식으로 구동된다. 모터의 동작 원리를 그림 1을 이용하여 설명하면 다음과 같다. 그림 1의 (a)에 나타난 화살표 방향으로 A상에 전류를 흘려주면 영구자석의 자속과 전류에 의한 자속이 상쇄, 보강을 통하여 그림의 아래 위 화살표 방향으로 자로가 형성되고 자기 저항이 최소가 되는 방향으로 움직여 안정한 평형점에서 정지한다. 그 다음 B상에 화살표 방향으로 전류를 흘려주면 (b)와 같은 자로가 형성되고 같은 원리에 의해 움직여서 이동된 안정점에서 정지하게 된다. 계속하여 A상에 반대방향의 전류를 흘리고, B상에 반대 방향의 전류를 흘려주면 (c), (d)와 같이 움직인다. 이와 같이 한 주기의 전류가 흐를 때 마다 한 피치씩 오른쪽으로 이동한다. 이와 같은 방식은 2상중 항상 한 상이 여자되는 1상여자 방식이다.

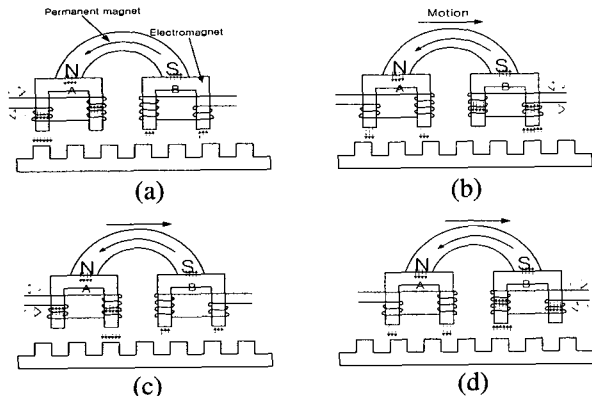


그림 1. 하이브리드형 리니어 펄스 모터의 동작원리.
Fig. 1. The principle of the hybrid type linear pulse motor.

2. 리니어 펄스 모터의 수학적 모델링

모든 시스템을 제어하기 위해서는 주어진 플랜트를 모델링하는 것부터 시작된다. 모델링이란 다루고자 하

는 물리적인 현상을 수학적으로 표현하는 것을 의미한다. 리니어 모터의 모델링시 여러가지의 복잡성을 피하기 위하여 여러 가정을 전제로 기계방정식을 세우면 다음과 같다.

$$M \frac{d^2x}{dt^2} + B \frac{dx}{dt} + F_L = F_e \tag{1}$$

여기서,

- M : 가동자와 부하의 질량
- B : 점성 마찰 계수 (viscous frictional coefficient)
- F_e : 모터에 의해 발생하는 힘 (developed force)
- F_L : 부하에 의한 힘 (load force)
- x : 가동자의 위치

일반적으로 간략화 된 하이브리드형 펄스 모터의 각 상에서 발생하는 힘은 다음과 같다[4].

$$F_e(i_A, i_B, x) = K \left(-\sin\left(\frac{2\pi}{p}x\right) i_A + \cos\left(\frac{2\pi}{p}x\right) i_B \right) \tag{2}$$

여기서,

- p : 치 간 거리, K : 상수

III. 제어기 설계

모터의 발생힘 제어는 전체 시스템 제어의 가장 기본이 되는 제어루프이며 발생힘 제어의 성능은 전체 제어 시스템의 성능을 좌우한다고 할 수 있다. 2상여자 방식은 항상 동시에 2개의 상이 여자가 되는 방식이다. 본 논문에서 사용하는 하이브리드 리니어 펄스 모터는 2개의 상(A,B)을 갖고 있고, A, B상의 전류 지령이 음일 때는 구동 방식이 유니폴러 방식이 아닌 바이폴러 방식이므로 \bar{A}, \bar{B} 상(A,B상의 역방향)으로 전류가 흐르게 된다. 2상여자 방식에서의 인가되는 시퀀스는 $AB, \bar{A}\bar{B}, \bar{A}B, A\bar{B}$ 이다. (2)를 이용하여 이와 같은 2상 여자방식이나 그림 1에서 설명한 1상 여자 방식, 또는 1-2상여자 방식의 어느 방법으로 제어 하나 여자 되는 상의 개수 차이만 있을 뿐 발생힘 리플은 여전히 발생한다. 그래서 추력 F_e 를 일정하게 발생시키기 위한 전류명령 i_A, i_B 는 다음과 같이 결정되어 진다. 이와 같은 방법이 마이크로(미세) 스텝 제어라는 개루프 제어방식이다.

$$i_A = -\sin\left(\frac{2\pi}{p}x\right) i_B = \cos\left(\frac{2\pi}{p}x\right) \tag{3}$$

(3)과 같은 정현파 전류를 인가해 줌으로써 (2)에 의하면 힘 리플이 없는 일정한 힘이 발생하며 평형점이 연속적으로 이동할 수 있게 된다. 그러나, 위의 전류 명령도 모델링 과정에서 여러가지의 가정을 도입하여 간략화 된 모델에서 구한 것이므로 필연적으로 추력 리플이 발생하게 되어 위치 정밀도를 떨어뜨리게 된다. 또한 기계적 모델링에서 비선형 마찰은 정확히

모델링되지 못하며 모터의 파라미터 변화에 의한 위치 정밀도도 떨어지게 된다.

1. 신경망 제어기 설계

II장에서 모델링한 플랜트의 수식적 표현은 실제의 시스템과 최소의 오차를 가지고 동작해야 하는데 그 오차가 큰 경우에는 그 모델을 이용하여 설계한 제어기가 제기능을 발휘하지 못하는 수가 있다. 그렇지만 실제의 상황을 모두 고려해서 정확하게 수학적으로 표현하는 것은 매우 어렵다. 그래서 본 논문에서는 그림 2와 같은 신경망 제어기를 사용하였으며, 위치 프로파일에서 생성된 setpoint와 플랜트의 출력을 이용하여 PID제어기를 통하여 전류지령을 생성하고, 전류지령에 따른 모터 구동 전류를 생성한다. 아울러 피드포워드 제어기와 신경망 제어기를 PID제어기와 동시에 동작시키고, 신경망 제어기는 모델의 불확실성 등으로 인한 위치 오차의 절대 크기 및 리플을 최소화 하는 방향으로 학습을 하여 각상의 전류의 모양을 적절히 바꾸어 준다.

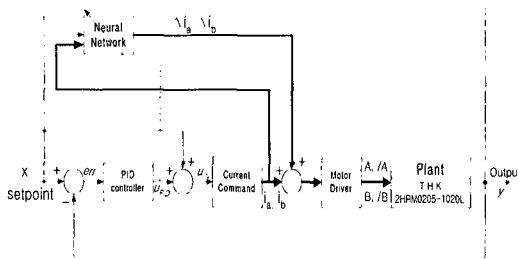


그림 2. 신경망을 사용한 제어시스템의 블록도.
Fig. 2. Block diagram of control system using neural Network.

본 논문에서 제안하고 있는 신경망 제어기는 그림 3에 나타난 바와 같이 입력층과 은닉층 그리고 출력층으로 나누어져 있다.

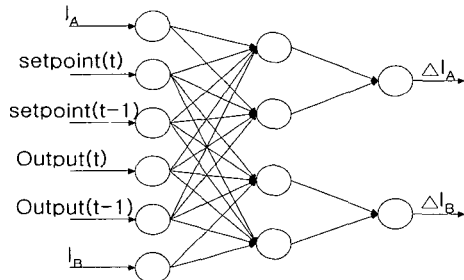


그림 3. 사용한 신경망 제어기 구성.
Fig. 3. The structure of used neural network.

신경망 제어기는 그림 3과 같이 입력층은 6개의 뉴런으로 이루어져 있으며, 각 상의 전류지령과, 입출력 값, 이전 입출력값을 입력으로 받는다. 리니어 모터는

코킹 때문에 위치에 종속되어 외란의 영향을 받는다. 그래서 신경망의 입력으로 setpoint 위치와 실제 출력의 위치를 사용하였다. 은닉층의 뉴런의 수는 4개로 하였다. 각 상의 전류를 변화 시킬 $\Delta i_a, \Delta i_b$ 를 출력층을 통해 출력하는데 각각 1개의 출력으로 나누어져 있다. 이 출력은 PID제어기에 의해서 계산된 전류지령 i_a, i_b 에 더해진다. 학습신호로는 PID의 출력을 사용하였다. 각각의 뉴런들은 시냅스로 서로 연결되어 있으며, 시냅스의 연결강도는 학습에 의해 적절하게 변화된다. 시냅스의 연결강도를 변화하는 알고리즘으로는 오차 역전파 알고리즘을 사용한다.

2. 신경망 학습

학습 신경회로망으로는 다층 역전파 신경회로망(Back Propagation Neural Network) [7]을 사용하며 입력층, 은닉층, 출력층으로 이루어진다. 하나 이상의 은닉층을 가지는 다층 역전파 신경회로망은 임의의 비선형 함수를 근사화시킬 수 있다고 알려져 있다[7].

전이 함수는 양극성 출력을 내기 위해 양극성 전이 함수를 사용하였다. 또한 빠른 수렴을 위하여 모멘텀 항을 사용한다. 초기 가중치들은 $-0.2 \sim +0.2$ 사이의 난수 값을 사용하고 학습률은 0에서 1사이의 값으로 초기화한다.

오차 학습 제어기에서는 신경회로망의 학습 신호로 목표 출력값 setpoint와 실제 출력과의 차이 즉 위치 오차를 사용하며, 신경회로망은 피드백 제어기의 출력이 최소화되도록 학습이 이루어진다.

입력에 대한 신경회로망의 출력은 다음과 같다

$$y_i = f(\psi_i); \quad \psi_i = \sum_{j=0}^Q w_{ij}h_j; \quad i=1,2,\dots,M \quad (4)$$

여기서, h_j 는 은닉층의 j 번째 신경소자의 출력으로서 은닉층이 1개인 경우에는 다음과 같다.

$$h_j = g(\psi_j); \quad \psi_j = \sum_{k=0}^N v_{jk}x_k; \quad j=1,2,\dots,Q \quad (5)$$

결선강도의 학습에 이용될 신경회로망의 출력과 목표 출력 사이의 제곱오차는 다음과 같다.

$$E = \frac{1}{2} \sum_{i=1}^M (U_{PID})^2 \quad (6)$$

출력층 결선 강도 w_{ij} 에 대한 오차의 Gradient는 다음과 같다.

$$\frac{\partial E}{\partial w_{ij}} = -(U_{PID}) \frac{\partial f(\psi_i)}{\partial \psi_i} h_i = -\delta_{yi} h_j \quad (7)$$

여기서, δ_{yi} 는 출력층 i 번째 신경 소자의 일반화된

오차이다.

$$\delta_{yi} = (U_{PID}) \frac{\partial f(\psi_i)}{\partial \psi_i} \quad (8)$$

은닉층 결선강도 v_{jk} 에 관한 오차의 Gradient는 다음과 같이 구해진다.

$$\frac{\partial E}{\partial v_{jk}} = - \sum_{i=1}^M \delta_{yi} w_{ij} \frac{\partial g(\phi_j)}{\partial \phi_j} x_k = -\delta_{hj} x_k \quad (9)$$

여기서, δ_{hj} 는 은닉층 j 번째 신경소자의 일반화된 오차로서 다음과 같이 나타내어진다.

$$\delta_{hj} = \sum_{i=1}^M \delta_{yi} w_{ij} \frac{\partial g(\phi_j)}{\partial \phi_j} \quad (10)$$

최종적으로 출력층과 은닉층의 결선 강도는 (11), (12)에 의하여 조정된다.

$$w_{ij}(n+1) = w_{ij}(n) - \eta \frac{\partial E}{\partial w_{ij}} \quad (11)$$

$$= w_{ij}(n) + \eta \delta_{yi}(n) h_j(n)$$

$$v_{jk}(n+1) = v_{jk}(n) - \eta \frac{\partial E}{\partial v_{jk}} \quad (12)$$

$$= v_{jk}(n) + \eta \delta_{hj}(n) x_k(n)$$

(11)과 (12)의 출력층과 은닉층의 결선강도 들은 (6)으로 정의된 E의 함수 값이 줄어드는 방향으로 조정되며 이 때 η 는 결선강도가 변해가는 양을 조절하는 학습률을 나타낸다. (6)의 E의 함수는 플랜트로 인가되는 PID제어입력의 제곱이며 신경망은 이 값이 줄어드는 방향으로 학습이 된다. PID제어 입력이 0이 된다는 것은 정확히 출력이 setpoint를 추종한다는 것을 의미하므로 이것을 신경망을 학습시킬 오차 값으로 사용하였다. 이러한 방법을 피드백 오차 학습방법이라고 한다.

IV. 실험 및 결과 고찰

1. 실험장치의 구성

그림 4에 실험장치 구성도를 나타내었으며, 실험은 TMS320C31이 내장된 dSPACE사의 DS1102 DSP 보드를 PC에 장착하여 제어기를 꾸몄으며, 제안된 이론에 따라서 전류명령을 생성하여 전류 드라이버로 명령을 전달하면 전류 드라이버에서 리니어 펄스 모터를 구동하기 위해서 전류지령이 양일 경우에는 A, B상의 전류를 음일 경우에는 /A, /B상의 전류 파형을 생성한다. 이동한 거리측정은 모터와 직접 연결되어 있는 리

니어 스케일을 사용하였으며, 스케일에서 발생한 펄스를 카운터에서 검출하여 이동한 거리를 알아낸다.

리니어 펄스 모터는 THK사의 2HRM0205-1020L이며 피치는 0.8mm이며, 정격 전류는 각 상당 1.4A이다. 리니어스케일의 분해능은 20 μm 이며 이것을 DSP에서 4채배하여 5 μm 의 분해능을 가지도록 하였다. 한 주기의 위치 프로파일을 사용하여 실험을 하였으며, DSP는 host PC로부터 프로그램을 다운 받아 제어를 시작하고 제어가 종료되면 메모리에 저장하였던 결과 값을 host PC로 전송한다.

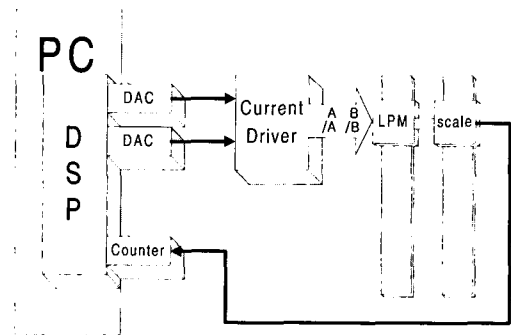


그림 4. 실험장치의 구성도.

Fig. 4. Schematic diagram of experimental equipment.

2. 전류 드라이버

전류 드라이버의 블록도를 그림 5에 나타내었다. DSP 보드에서 나온 전류명령과 홀센서를 통해서 피드백 된 모터의 실제전류와의 차이를 아날로그적으로 OP Amp.로 PWM시켜 전압의 변화를 펄스폭의 변화로 바꾸어 준다. 스텝모터 드라이브인 L298N에서 리니어 펄스 모터 구동신호인 A, B, /A, /B를 출력한다.

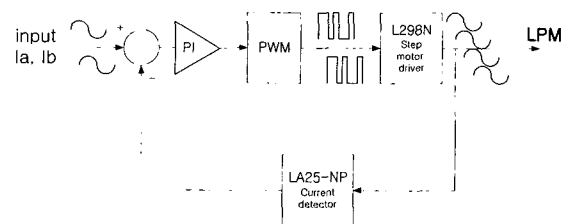


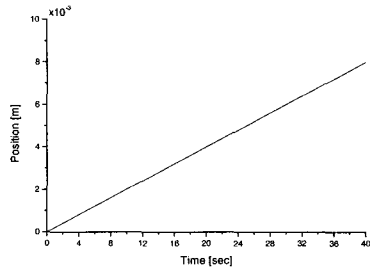
그림 5. 제작한 전류 드라이버의 블록도.

Fig. 5. The block diagram of current driver.

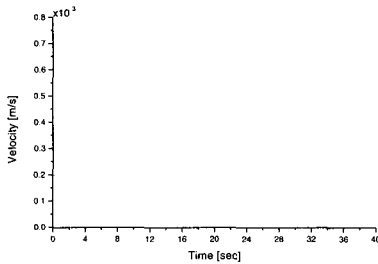
3. 실험결과 및 고찰

신경망제어기의 학습률은 0.1, 모멘텀항은 0.9, 초기 가중치는 -0.1~ +0.1사이의 값으로 초기화를 하였다. 리니어 펄스 모터의 1피치는 0.8mm이다.

그림 6과 그림 7은 실험에서 사용한 저속의 경우와 고속의 경우 각각에 대한 기준위치와 기준속도를 나타낸다. 샘플링 시간은 2.5msec이다.

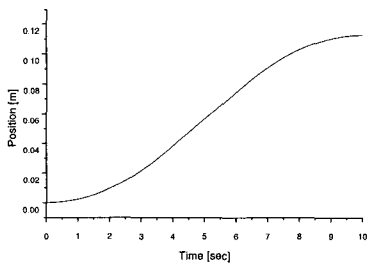


(a) 기준위치

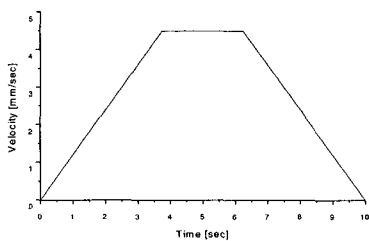


(b) 기준속도

그림 6. 저속실험에서의 기준위치와 기준속도.
Fig. 6. Desired position and desired velocity at low velocity.



(a) 기준위치



(b) 기준속도

그림 7. 고속실험에서의 기준위치와 기준속도.
Fig. 7. Desired position and desired velocity at high Velocity.

3.1 저속에서의 실험

개루프 제어만으로 제어했을 때의 수행결과를 그림 8에 나타내었다. 개루프 제어란 그림 2에서 피드백 경

로와 피드포워드 경로, 신경망 제어기를 사용하지 않고 제어하는 방법이다. 기준위치를 정확히 따라가지 못하고 항상 얼마간의 일정한 오차가 존재하였으며, DC 오프셋 오차가 존재하며 오차 리플 크기도 0.1mm 정도로 많이 큼을 알 수 있다.

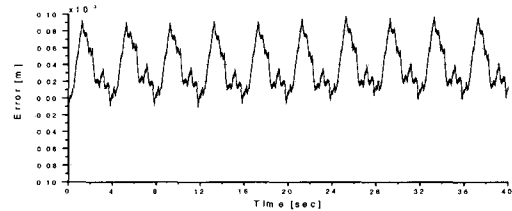


그림 8. 개루프만으로 제어했을 때의 위치오차.
Fig. 8. The position error with only open loop control.

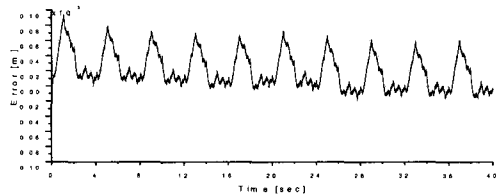


그림 9. 피드포워드와 PI(FF+PI) 제어기를 사용한 경우의 위치오차.
Fig. 9 The position error with feedforward and PI (FF+PI) controller.

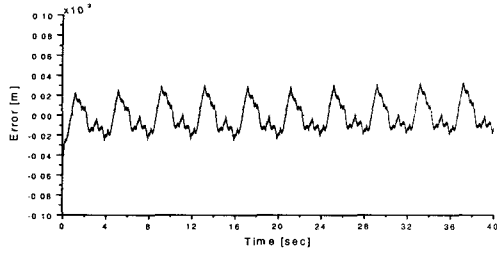
PI제어기와 피드포워드 제어기만을 사용하였을 때의 결과를 그림 9에 나타내었다. 피드포워드 제어기는 그림 2에서 나타낸 것처럼 단위 피드포워드 경로가 연결되어 이 출력이 PI의 출력과 합해져서 인가된다. DC 오프셋 오차가 점차적으로 음의 방향으로 드리프트가 되는 것을 볼 수 있다. 그러나 리플의 크기는 0.075mm 정도로 별로 줄어들지 않았다.

PI+FF제어기와 신경망 제어기를 동시에 사용한 결과를 그림 10에 나타내었다. DC 오프셋과 리플의 크기가 0.05mm정도로 많이 줄어들었음을 알 수 있다. (b)의 그림을 보면 7번째와 8번째 학습 과정사이에서 루트 평균제곱오차(root mean squared error :RMSE)가 급격히 줄어들음을 알 수 있다. 이 때의 RMSE는 (13)과 같이 계산된다. 이 그림의 가로축은 매 샘플링마다 on-line으로 학습되어서 10,000개의 샘플링 구간 즉, 40초 동안(샘플링 시간 : 2.5ms)에 수행된 전체의 루트 평균제곱 오차를 한번의 학습iteration으로 나타낸 것이다.

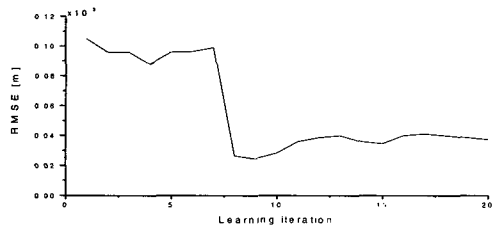
$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (error_i)^2} \quad (13)$$

지금까지 수행한 측정 결과를 표 1에 정리하였다. 표 1의 마지막 RMSE값은 최적으로 학습된 결선강도를 이용하여 40초 동안 수행한 값을 나타낸다. 신경망 제어기가 동작하면 RMSE값이 상당히 개선됨을 알 수

있으며, 특이한 것은 매번 한 피치당 같은 모양의 오차가 반복됨을 알 수 있다.



(a) 위치오차



(b) 학습에 의한 RMSE값의 변화

그림 10. PI+FF제어기와 신경망 제어기를 사용하였을 경우의 위치오차와 RMSE값.

Fig 10. The position error and RMSE value when the neural network and PI+FF controller are used.

표 1. 저속에서의 오차비교.

Table 1. Comparison of position error at low velocity.

제 어 기	오차 표준 편차 (mm)	RMSE값 (mm)
Open loop	0.0264	0.0455
PI + FF	0.0233	0.0308
PI + FF + NN	0.0142	0.0143

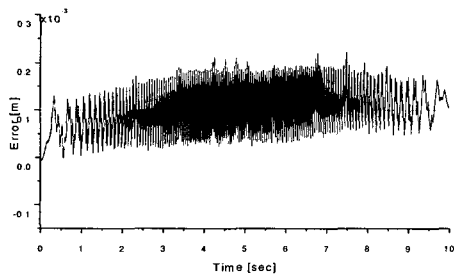


그림 11. 개루프만으로 제어했을 때의 위치오차.

Fig. 11. The position error with only open loop control.

3.2 고속에서의 실험

고속인 경우(그림 7의 프로파일을 사용)에 대하여

위치오차의 절대 크기와 리플의 크기를 실험적으로 고찰하여 보았다.

개루프 제어만으로 제어 했을 때의 결과를 그림 11에 나타내었다. 속도가 빠른 부분(3.5초에서 6.5초 사이)에서 리플의 크기가 큼을 알 수 있으며 저속인 경우에 비해서 오차 리플의 크기가 약 2배 정도 커졌음을 알 수 있다. 또한 기준위치를 정확히 따라가지 못하고 항상 얼마간의 오차가 존재하고 있음을 알 수 있다.

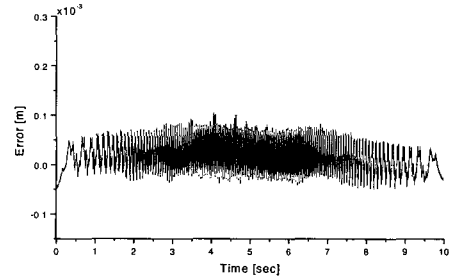
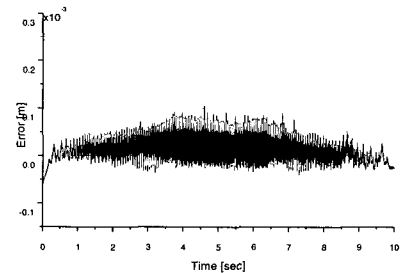


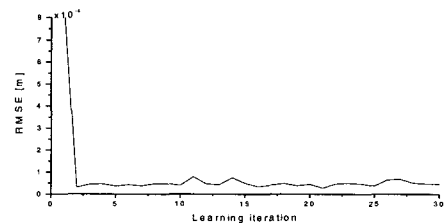
그림 12. 피드포워드와 PI제어(FF+PI)를 수행한 경우의 위치오차.

Fig. 12. The position error with PI+FF controller.

신경망 제어기를 사용하지 않고 PI제어기와 피드포워드 제어기만을 사용하였을 때의 결과를 그림 12에 나타내었다. 속도가 일정한 구간에서는 개루프 제어결과에 비해서 I 제어의 영향으로 DC 오차가 음의 방향으로 약간 드리프트됨을 알 수 있다. 오차리플의 크기도 개루프 제어결과에 비해서 조금 줄어들었음을 알 수 있다.



(a) 위치오차



(b) 학습에 의한 RMSE값의 변화

그림 13. PI+FF제어기와 신경망제어기를 동시에 사용하였을 경우의 위치오차

Fig 13. The position error and RMSE value when the neural network and PI+FF controller are used.

표 2. 고속에서의 오차비교.

Table 2. Comparison of position error at high velocity.

제 어 기	오차 표준 편차(mm)	RMSE값 (mm)
Open loop	0.0424	0.1110
PI + FF	0.0277	0.0325
PI + FF + NN	0.0231	0.0265

PI+FF제어기와 신경망 제어기를 동시에 사용하였을 때의 결과를 그림 13에 나타내었다. 오차의 크기는 상당히 개선되었지만 여전히 리플은 많이 남아 있음을 알 수 있다. (b)의 그림을 보면 저속인 경우와는 달리 신경망이 첫 학습부터 오차를 줄이도록 바로 학습을 시작함을 알 수 있다. 고속인 경우에도 저속과 마찬가지로 신경망 제어기를 동시에 사용한 경우가 가장 좋은 결과를 얻었다. 지금까지 수행한 고속에서의 결과를 표 2에 나타내었다.

4. 결과고찰

기존의 개루프 제어나 PI+FF제어기에 신경망 제어기를 동시에 사용한 경우가 저속에서나 고속에서 모두 RMSE 값이 확연히 줄어들음을 알 수 있었다. RMSE 값을 보면 저속의 경우 개루프 제어의 결과에 비해서 PI+피드포워드 제어기는 67.69%, PI+피드포워드+신경망 제어기의 경우는 RMSE값이 31.4% 크기로 줄어들었다. 고속의 경우는 개루프의 결과에 비해서 각각 29.27%, 23.8% 크기로 줄어들었다. 이 결과로부터 신경망 제어기의 효과는 특히 저속에서 크게 나타나며 고속의 경우에는 신경망 제어기의 효과가 크게 개선되지 않음을 알 수 있다. 그러나 고속의 결과 중 속도가 낮은 부분에서는 오차 리플의 크기가 신경망을 사용했을 때가 가장 작음을 알 수 있다. 그리고 오차의 표준편차 측면에서도 고속보다 저속에서의 효과가 더 큼을 알 수 있다. 전체적인 결과를 볼 때 신경망 제어기를 도입하였지만 오차 리플의 크기를 줄이는 데는 아직 미흡한 점을 알 수 있다. 이러한 리플의 문제는 본 논문에서 제시한 신경망 제어기 이외에도 전류 드라이브 자체의 전기적인 시정수, 샘플링 시간과 주기적으로 나타나는 리플의 모양 등을 고려한 총체적인 제어기를 설계하는 방법도 고려해야 될 것으로 생각된다.

V. 결론

신경망 제어기를 리니어 펄스 모터의 정밀 모션 제어에 적용하였다. 사용된 신경망 제어기는 오차의 크

기가 줄어들도록 리니어 펄스 모터에 인가되는 전류지령의 모양을 변화시켰으며, 이 결과 위치오차의 리플 크기와 DC 오프셋 오차 크기가 줄어들음을 확인할 수 있었다. 적용된 신경망 제어기의 RMSE값이 학습이 진행됨에 따라서 줄어드는 것을 알 수 있었으며 그것으로 잘 동작함을 알 수 있었다. 제안한 신경망 제어기의 효과는 특히 저속에서 크게 나타나며 고속의 경우에는 신경망 제어기의 효과가 약간 밖에 개선되지 않음을 알 수 있었다. 그러나 아직도 위치 오차에 큰 리플이 존재하는 데, 이는 리니어 펄스 모터의 많은 비선형 항 등의 모델링 되지 않은 부분을 신경망 제어기가 충분히 극복하지 못한 때문이라 생각된다. 신경망 제어기의 많은 값들이 최적의 조건으로 맞추어 지지 않았으며, 실험적으로 최적으로 맞추는데도 어느 정도 한계가 있었으며, 이를 해결하기 위한 한 방법으로 리플의 모양을 보면 피치 마다 같은 모양의 리플이 반복해서 나타나므로 이러한 주기적 리플에 대한 외란 관측기를 구현하거나 시정수가 아주 짧은 고성능 전류 드라이브의 제작 등을 포함한 다른 신경망의 구성법을 찾아야 할 것으로 생각된다.

참고문헌

- [1] S. A. Nasar and I. Boldea, Linear electric motors: Theory, design, and practical applications, *Prentice-Hall, Inc.*, 1987.
- [2] Gerco Otten, Theo J. A. De Vries, Job van Amerongen, Adrian M. Rankers, and Erik W. Gaal, "Linear motor motion control using a learning feedforward controller," *IEEE/ASME Trans. on Mechatronics*, vol. 2, no. 3, pp. 179-187, September, 1997.
- [3] 박정일, 이석규, 이기동, 이재형, "리니어모터와 제어기술," 대한전자공학회지, 제26권, 제10호, pp. 53-65, 1999.
- [4] T. Kenjo and A. Sugawara, *Stepping Motors and their Microprocessor Controls*, Oxford, 1985.
- [5] 추상훈, 강준혁, 임희선, 하인중, "하이브리드 스텝 모터의 궤환 선형화 제어," 대한전기학회 논문집, 제43권, 제9호, pp. 1572-1583, 1994.
- [6] C. S. Moon, J. I. Park, K. D. Lee, S. G. Lee, and J. H. Lee, "Design of position controller of linear pulse motor using neural network," *Proceeding of American Control Conference*, pp. 990-991, 1998.
- [7] C. T. Lin and C.S. George Lee, *Neural Fuzzy systems*, *Prentice-Hall Inc.*, 1996



권영건

1971년 5월 29일생. 1998년 2월 영남대학교 전자공학과 졸업(공학사). 2000년 2월 영남대학교 대학원 전자공학과 졸업(공학석사). 2000년 1월~현재 삼성전자 생산기술센터 자동화 연구소 근무. 관심분야는 지능제어, 자동화.



박정일

1958년 4월 8일생. 1981년 2월 경북대학교 전자공학과 졸업(공학사). 1983년 2월 서울대학교 대학원 전자공학과 졸업(공학석사). 1989년 2월 서울대학교 대학원 전자공학과 졸업(공학박사). 1983년 4월~1985년 12월 삼성전자 종합연구소 연구원. 1989년 4월~1992년 2월 삼성종합 기술원 전자기기 연구소 선임연구원. 1992년 3월~현재 영남대학교 전자정보공학부 부교수. 1998년 1월~1999년 1월 University of Illinois at Urbana-Champaign 방문교수. 관심분야는 지능제어, 리니어·서보모터 제어, 메카트로닉스, 적응학습 시스템.