

FNNs의 하드웨어 구현을 위한 학습방안

박진성, 조화현, 채종석, 최명렬  
한양대학교 제어계측공학과

A Learning Scheme for Hardware Implementation  
of Feedforward Neural Networks

Jin-Sung Park, Hwa-Hyun Cho, Jong-Seok Chae, Myung-Ryul Choi  
Dept. of Control & Instrumentation Eng., Hanyang University

**Abstract** - 본 논문에서는 단일패턴과 다중패턴 학습이 가능한 FNNs(Feedforward Neural Networks)을 하드웨어로 구현하는데 필요한 학습방안을 제안한다. 제안된 학습방안은 기존의 하드웨어 구현에 이용되는 방식과는 전혀 다른 방식이며, 오히려 기존의 소프트웨어 학습방식과 유사하다. 기존의 하드웨어 구현에서 사용되는 방법은 오프라인 학습이나 단일패턴 온 칩(on-chip) 학습방식인데 반해, 제안된 학습방식은 단일/다중패턴 온 칩 학습방식으로 다층 FNNs 회로와 학습회로 사이에 스위칭 회로를 넣어 구현되었으며, FNNs의 학습회로는 선형 시냅스 회로와 선형 곱셈기 회로를 사용하여 MEBP(Modified Error Back-Propagation) 학습규칙을 구현하였다.

제안된 방식은 기존의 CMOS 공정으로 구현되었고 HSPICE 회로 시뮬레이터로 그 동작을 검증하였다. 구현된 FNNs는 어떤 학습패턴 쌍에 의해 유일하게 결정되는 출력 전압을 생성한다. 제안된 학습방안은 향후 학습 가능한 대용량 신경망의 구현에 매우 적합하리라 예상된다.

1. 서 론

VLSI 기술의 발전과 인간 신경계에 대한 연구가 진행됨에 따라 포유동물의 신경계를 본따 다양한 신경망 모델을 구현하는 것이 가능해졌다. 그러나, 이러한 신경망 모델은 생물학적 신경망에 비해 크게 간략화된 것이다. 신경계가 모방에 의해 구현되어질 때 인공신경망(Artificial Neural Networks-ANNs)이라 부른다. 인공신경망의 구현은 간단한 계산 구성요소를 대량으로 사용하는데 그 기반을 두고 있다.

많은 연구자들이 많은 종류의 인공신경망 모델을 가지고 신경망의 소프트웨어 혹은 하드웨어 형태의 구현에 대해 연구하고 있다. 소프트웨어 구현은 대개 신경망의 해석과 구조에 기반을 둔 알고리즘을 채용하여 기존의 디지털 컴퓨터 상에서 수행된다(1). 하드웨어 구현은 대개 선형 곱셈기 회로를 가지는 선형 시냅스 회로를 사용하여 이루어진다(2,3,4). 대부분의 하드웨어 구현은 오프 칩(off-chip) 학습이나 단일패턴의 온 칩(on-chip) 학습방식에 기반을 두고 있다(4,5). 본 논문에서는 FNNs의 단일/다중패턴 학습의 하드웨어 구현을 위한 두가지 학습방식을 제안한다. 두 방식 모두 기존의 CMOS 기술에 가지고 구현되었으며, 그 동작은 HSPICE 회로 시뮬레이터로 검증하였다.

2. 본 론

2.1 변형된 학습알고리즘의 구현

그림 1은 학습회로를 내장한 FNNs의 블록도이다. 이 회로는 2x2x1 FNN 회로인데, MEBP 학습규칙을 채용하였으며 두 개의 입력노드와 은닉층에 두 개의 뉴런, 출력층에 한 개의 뉴런을 가지고 있다. 그림 1에서 1D-MUL은 아날로그 신호와 아날로그 시냅스 가중치

를 선형적으로 곱하는 선형 아날로그 곱셈기를 뜻한다. MEBP 학습규칙은 error back-propagation(EBP) 규칙의 안정성과 수렴도를 잃지 않으면서 EBP 규칙에서 시그모이드 도함수 항을 제외한 것이다(6). 이것은 뉴런 k가 출력층에 있으면

$$\bar{w}_{kj} = \eta \sum_p (\tau_{pk} - \bar{y}_{pk}) y_{pj} - \bar{\alpha}_{ji} \bar{w}_{ji}$$

그리고 은닉층에 뉴런 j가 있을 때

$$\dot{w}_{ji} = \eta \sum_k \bar{w}_{kj} \sum_p (\tau_{pk} - \bar{y}_{pk}) y_{pi} - \alpha_{ji} w_{ji}$$

여기서  $w_{ij}$ 는 시냅스 가중치이며  $y$ 는 출력,  $\tau$ 는 목표치,  $\alpha$ 와  $\eta$ 는 상수이다.

MEBP 규칙은 선형 시냅스 회로, 시그모이드 회로, 그리고 선형 곱셈기로 구현되었다. 회로 구현은 다음의 식으로 설명할 수 있다(7).

$$\begin{aligned} RC \bar{w}_{11} &= k_1 f((\tau - \bar{y}_1) \cdot y_1) - \bar{w}_{11} \\ RC \bar{w}_{12} &= k_1 f((\tau - \bar{y}_1) \cdot y_2) - \bar{w}_{12} \\ RC \dot{w}_{11} &= k_1 g(\bar{w}_{11} \cdot f((\tau - \bar{y}_1) \cdot x_1)) - \dot{w}_{11} \\ RC \dot{w}_{12} &= k_1 g(\bar{w}_{11} \cdot f((\tau - \bar{y}_1) \cdot x_2)) - \dot{w}_{12} \\ RC \dot{w}_{21} &= k_1 g(\bar{w}_{12} \cdot f((\tau - \bar{y}_1) \cdot x_1)) - \dot{w}_{21} \\ RC \dot{w}_{22} &= k_1 g(\bar{w}_{12} \cdot f((\tau - \bar{y}_1) \cdot x_2)) - \dot{w}_{22} \end{aligned}$$

여기서  $x_i$ 는 입력,  $w_{ij}$ 는 시냅스 가중치,  $y_i$ 는 출력,  $\tau$ 는 목표치, 그리고  $k_1$ 는 상수이다.

2.2 제안된 FNN 학습방안의 구현

FNNs의 단일/다중패턴 학습의 하드웨어 구현을 위한 새로운 학습방식이 제안되었다. 제안된 학습방식은 기존의 하드웨어 구현과는 전혀 다르며, 오히려 기존의 소프트웨어적 접근방식과 유사하다.

제안된 학습방식은 그림 2에서와 같이 다층 feedforward 회로와 MEBP 학습회로 사이에 스위칭 제어 회로를 삽입하여 구현된다. 스위칭 회로는 feedforward 회로의 동작과 학습회로의 동작을 분리시켜준다. 따라서 각 학습 단계마다 두 단계를 거치게 된다. 첫 번째 단계에서는 오직 feedforward 회로만이 동작하고 학습회로는 off 상태가 된다. 두 번째 단계에서는 그 반대로 동작한다. 이것은 소프트웨어로 구현된 EBP 규칙에서 미분 방정식을 풀어나가는 방식과 매우 유사하다.

각 스위칭 동작은 feedforward 회로의 출력력 학습 목표와 같아지고, 학습된 시냅스 가중치가 일정한 값으로 수렴할 때까지 계속된다.

전체 회로가 구현되어 HSPICE 회로 시뮬레이터를

통해 하나의 패턴을 학습하는 과정을 모의실험하였다. 그 결과가 표 1에 정리되어 있다. 이하의 표들은 제안된 학습회로의 출력(y)이 상응하는 목표값( $\tau$ )에 수렴하며, 안정된 가중치 값을 가지는 것을 보여주고 있다. 이것은 제안된 학습회로가 주어진 입출력 패턴을 성공적으로 학습하였다는 것을 나타낸다.

### 2.3 다중 학습패턴의 구현

표1의 결과는 제안된 학습회로가 주어진 입출력 패턴을 성공적으로 학습하였다는 것을 보여준다. 그러나, 여러 개의 서로다른 입출력 패턴을 학습하기 위해 한번에 하나의 입출력 패턴을 인가하면 학습회로는 전체적으로 이전 학습패턴을 유지하지 못한다는 것을 알게되었다. 이것은 시냅스 가중치가 새로 인가된 학습패턴만을 학습하기 위해 갱신되기 때문이다.

다중 입출력 패턴을 학습하기 위해 각 입력-목표 패턴이 상응하는 학습단계동안 연속적으로 인가된다. 예를 들어, 두 개의 입출력 패턴을 학습하는 경우, 첫 번째 패턴은 첫 번째 학습단계동안 인가되며 다른 패턴은 다음 단계에서 인가된다. 그 다음에는 학습이 끝날 때까지 이러한 동작이 반복된다.

HSPICE 시뮬레이션으로 하나의 목표를 가지는 2-, 3-, 4-패턴 학습을 수행하였다. 이 시뮬레이션 결과가 표 2와 표 3에 각각 나타나있다. 이 표에서 제안된 학습회로의 출력(y)은 목표( $\tau$ )에 수렴하며, 안정된 가중치 값을 가지는 것을 알 수 있다. 이러한 간단한 예들은 비록 그 구현이 모든 강력한 학습 능력을 보여주기에 부족하지만 제안된 학습방식에 의한 학습 동작이 이루어지는 것을 보여준다. 위에서 서술한 바와 같이 각 시냅스 가중치 값은 각각 수렴하며 학습회로에서 얻을 수 있다. 학습된 시냅스 가중치는 디지털 메모리 회로에 저장되어 사용될 수도 있다. 그러나, 아날로그 시냅스 가중치의 완벽한 사용을 위해서는 capacitor와 같은 아날로그 저장 장치를 사용하여야 한다.

### 3. 결 론

본 논문에서는 새로운 학습방안을 채용한 MEBP 학습규칙을 채용하여 FNNs를 구현하였다. 제안된 학습방안은 FNNs의 단일패턴 학습과 다중패턴 학습의 하드웨어 구현에 적합하다. 이 학습방안은 기존의 하드웨어 구현과는 전혀 다르며 오히려 기존의 소프트웨어적 접근방식과 유사하다. 학습방안은 다층 feedforward 회로와 MEBP 학습회로 사이에 스위칭 회로를 넣는 것으로 구현되었다.

제안된 학습방안은 2x2x1 feedforward 신경 회로를 위해 구현되었으며, HSPICE 회로 시뮬레이터로 그 동작을 검증하였다. 그 결과, 구현된 FNN 회로는 각각의 단일 입출력패턴과 다중 입출력 패턴을 성공적으로 학습하였다. 제안된 학습방안은 향후 학습기능을 가진 대규모 신경망의 구현에 적합하다.

본 연구는 과학기술부의 G7 뇌연구개발사업 지원을 받아 수행되었습니다.

#### [참 고 문 헌]

[1] D.E.Rumelhart, J.L.McClelland, and the PDP Research Group Eds., "Parallel Distributed Processing - Explorations in the Microstructure of Cognition", vol.1, Foundations, Cambridge, MA:MIT Press, 1986.  
 [2] M.R.Choi, "Analog Vector Multipliers for Feedforward Neural Networks", Journal of Engineering & Technology, Vol.2, No.1, Hanyang University, Korea, pp.12-21, 1993

[3] B.Gilbert, "A high performance monolithic multiplier using active feedback", IEEE J. Solid State Circuits, SC-9, pp.364-373, 1974.  
 [4] F.Kub et al., "Programmable Analog Vector Matrix Multipliers", IEEE J. Solid State Circuits, SC-25, pp.207-214, 1990.  
 [5] M.R.Choi and F.Salam, "Implementation of Feedforward Artificial Neural Networks with Learning using Standard CMOS Technology", ISCAS '91, Singapore, pp.1509-1512, 1991.  
 [6] M.R.Choi, "Implementation of Feedforward Neural Network with Simple Nonlinear Circuits", J. Hanyang Research Institute of Industrial Science, vol.2, Hanyang University, Korea, pp.15-25, 1999.

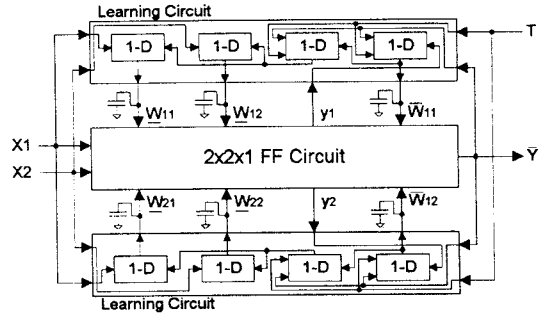


그림 1. 학습회로를 가진 2x2x1 FNNs

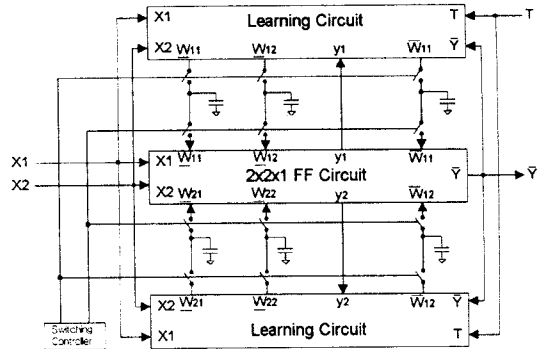


그림 2. 제안된 학습방안을 적용한 2x2x1 FNNs

표 1. 단일패턴 모의실험 결과

입력	x1	0	0	5	5	0	0	5	5
x2	0	5	0	5	0	5	0	0	5
목표치	$\tau$	0	5	5	0	5	0	0	5
출력	y	1.64	4.67	4.47	0.697	4.67	0.697	1.64	4.47
시냅스 가중치	$w_{11}$	2.74	2.52	2.48	2.42	2.52	2.58	2.26	2.48
	$w_{12}$	2.74	2.48	2.52	2.42	2.52	2.42	2.74	2.48
	$w_{21}$	2.74	2.52	2.48	2.42	2.52	2.58	2.26	2.48
	$w_{22}$	2.74	2.48	2.52	2.42	2.52	2.42	2.74	2.48
	$\bar{w}_{11}$	3.24	2.24	2.30	3.07	2.24	3.07	3.24	2.30
	$\bar{w}_{12}$	3.24	2.24	2.30	3.07	2.24	3.07	2.24	2.30

표 2. 2-패턴 모의실험 결과

입력	$x_{11}$	0	0	0	0
	$x_{12}$	0	5	0	5
	$x_{21}$	5	5	5	5
	$x_{22}$	0	5	0	5
목표	$\tau$	0	5	5	0
출력	$\bar{y}$	0.797	4.66	4.66	0.797
가중치	$w_{11}$	2.5	2.5	2.5	2.5
	$w_{12}$	2.5	2.5	2.5	2.5
	$w_{21}$	2.5	2.5	2.5	2.5
	$w_{22}$	2.5	2.5	2.5	2.5
	$\bar{w}_{11}$	3.1	2.25	2.25	3.1
	$\bar{w}_{12}$	3.1	2.25	2.25	3.1

표 3. 3-패턴, 4-패턴 모의실험 결과

입력	$x_{11}$	0	0	0	0
	$x_{12}$	0	5	0	5
	$x_{21}$	5	5	5	5
	$x_{22}$	0	5	0	5
	$x_{31}$	0	0	0	0
	$x_{32}$	5	5	5	5
	$x_{41}$	-	-	5	5
	$x_{42}$	-	-	0	0
목표	$\tau$	0	5	5	0
출력	$\bar{y}$	1.82 -2.0	4.66	2.3	3.43
가중치	$w_{11}$	2.59	2.5	2.5	2.5
	$w_{12}$	2.41	2.5	2.5	2.5
	$w_{21}$	2.59	2.5	2.5	2.5
	$w_{22}$	2.41	2.5	2.5	2.5
	$\bar{w}_{11}$	3.26	2.30	3.26	2.85
	$\bar{w}_{12}$	3.26	2.30	3.26	2.85