

연상학습을 위한 디지털 신경망에 관한 연구

正會員 金 承 煥* 正會員 姜 俊 吉*

A Study of Digital Neural Network for Associative Learning

Seung Whan KIM* June Gil KANG* *Regular Members*

要 約 McCulloch와 Pitts 모델은 VLSI에 의한 설계에 영향을 주었으며 신경회로망 해석에 또한 영향을 주어왔다. 학습훈련 과정에서 가중치를 아날로그 메모리로 저장하는데는 설계상에 많은 어려움이 있다. 따라서 디지털적인 모델을 구현하는데 따른 논리적 분석을 하고, 신경 노드의 훈련에 의한 응답을 수행할 수 있는, 독특한 기능을 정의하며, 크기에 있어 알려진 칩상에서 이루어지게 하는 가중치를 디지털적으로 인출할 수 있는지를 확인하였다.

Connectionist 시스템은 지식을 저장하는 노드(node)로 구성된 세포형식의 네트워크(network)이다. 본 논문에서는 일반적 인 노드 모델을 가정하고 확률적 요소들 가진 논리 진리표에 기초를 두고있다.

PLN 다층연상 메모리를 RAM으로 구현하는데 설계자가 허용해야 하는 일그러짐과 설계규칙을 정하고, 학습 훈련을 시도해 보았다. 훈련결과 확률이 지닐하는 양상으로 상태값이 변화됨을 일장으로 확인하였다. 따라서 본 논문은 확률적 노드모델을 RAM에 의해 다층 PLN을 구현한 것과 훈련을 시도하여 훈련상태가 외부적으로 작용하는지를 확인하는데 의의가 있으며 앞으로 연상학습 및 VLSI 구현의 가능성을 확인하였다.

ABSTRACT The McCulloch and Pitts model still reigns supreme in most schemes of neural computing and influences VLSI makers in their designs. This creates difficulties largely at the level of having to find analogue memories to satisfy the weight designs. Here it is shown that a logical analysis of the neuron leads to a logical model which avoids analogue memory and is centred on the conventional Random Access Memory. Defining the functionality of a neural node as the number of distinct functions that the node can perform in response to training, it is argued that digital attempts at designing weights should be examined closely in comparison with direct logical methods to assess whether sufficient functionality is obtained for the silicon area used.

A Connectionist system is a cellular networks of adaptable nodes that has a natural propensity for storing knowledge. A general node model is assumed in this paper. It is based on a logic truth table with a probabilistic element.

The multi layer associative network described here uses PLNs to provide learnt image to image transformations with design rules and an algorithm which allow the designer to achieve such transformation with a selected degree of accuracy.

I. 서 론

Connectionism은 어떤 의미에서 뇌의 신경망

(neural networks)과 같은 세포 네트워크(cellular networks)에 관한 연구이다. 이들 네트워크를 연결하고 훈련(training)하는 방법에 따라서 나타나는 성질이 있다. 훈련은 내부상태를 만들 수 있도록 구성하였다.

이들 좀 더 고안하여 개발한 자료는 Rumelhart & McClelland 편집한 문헌에 잘표현되어

* 光云大學校 電子工學科
Dept. of Electronic Engineering, Kwangwoon University.
論文番號 : 90-98 (接受1990. 9. 15)

있다. 훈련의 개요는 가중(weight) 또는 임계값(threshold) 변화의 절차이다. 신경 모델링(neural modelling)은 1943년 이후부터 연구되어 왔다. 최근에 와서 상태구조(즉, 상태외 그룹간의 전이)는 훈련 상태를 새로 만들수 있는 모듈(module)이 된다는 새로운 방법임을 알게 되었다. Hopfield¹⁰⁾는 셀 간 상호 연결을 가정하므로써 네트워크가 energetic system으로 모델화될 수 있음을 보여 주었다. 그리고 신경망(neural network)은 에너지 최소(energy minima)를 추구하는 방향으로 동작함을 증명하였다.

신경망은 많은 간단한 processing element의 집합으로, 그 전체적 동작은 개개 processing element의 총체적 특성으로서 표현된다.

Hinton's의 관점은 훈련과 동작의 개 호환 단계를 사용하였으며, Boltzmann Machine은 Simulated annealing을 사용하여 에러의 양을 줄일 수 있도록 하였다. 그러나 이 방법을 간단하지만 해답에 local minima가 존재할 경우 global minimum을 찾을때 실패할 수 있는 단점이 있다. 따라서 Simulated annealing은 이러한 local minimum에서 빠져나오기 위해 가중치값의 변화를 확률적으로 결정하였다.

Rumelhart 또한 피드백(feed back)이 없이 네트워크에 대한 에러수정 가중변환 알고리즘(error-correction weight changing algorithm)을 개발¹¹⁾하였다.

또한, 노드가 진리표로 기술되는 변수 논리로 나타내는 네트워크(network)가 상기에서 기술한 신경 모델에서 나타난 패턴완성에서 나타나는 성질을 찾아 훈련¹²⁾하였다. 이러한 관찰에서 나타난 결과는 연결과 node 기능에 대한 세부적 사항에 중점을 두지 않는 Connectionist system을 해석적 분석(explanatory analysis)을 증명하기 위함이다.

또한, 훈련 알고리즘에 대한 논리적 구성방법이 정의되어야 할 한다. 이 논문의 저자는 PLN(probabilistic logic node) 노드의 도입을 통해 방법을 서술하고 Connectionist system을 이해하는데 기본이 되며, 패턴인식 분야에 논리 확률적

스택을 RAM 메모리로서 구성할 수 있다.

일반적으로 네트워크를 클램핑이라 하며, 학습시킨 값을 클램핑에 입력시키고, 연산에 의해 생성된 노드값을 이드레스모듈과 패턴모듈에 새로 생성시키고 패턴모듈은 임출력모듈과 연결될 수 있도록 하였다. 이때는 클램프가 제거되어도 네트워크는 선택된 상태로 남아있게 된다. 처리과정을 몇가지 상태에 대해 반복된다. 이러한 연산 방식은 한 패턴의 유일한 부분이 네트워크 클램프되어 다른 상태로 전이되어 가도록 한 것이다.

본 논문에서 구현한 다층연상 메모리는 선형된 정확도를 갖으며, 설계자가 허용해야 하는 알고리즘과 설계 규칙과 함께 학습된 영상을 영상 변화하였다. 자동 연상학습과 학습을 하였을때 입력에 임으로 발생하는 응답 출력에 대한 RAM 모델의 확률적 대안을 영상 변환하여 에너지수지의 상태변화를 통해 학습이 이루어짐을 확인하였다.

II. 본 론

II-1. 커넥션리스트 네트(Connectionist Net)

한 connectionist net는 이분적 요소의 분류에 의해 특징지어질 수 있다. 상위의 수준에서 4개의 요소로 형성되는 집합에 의해 표현된다.

$$C: \langle P, G, Tr, Te \rangle.$$

P는 차례로 2개의 요소로 형성되는 집합에 의해 기술되는 물리적 구조이다.

$$P: \langle B, K \rangle,$$

여기서 B는 노드 $\{x_1, x_2, \dots, x_n\}$ 의 집합이다; K는 불규칙 네트워크에서 노드 집합(노드당 입력수), 혹은 정연한 네트워크에서 연결의 분포를 말하는 연결간의 패턴내어이다.

각 노드의 4개의 가까운 이웃에 연결되는 노드의 2차원 $n \times n$ 행렬이다.

$B: \{x_{ij} / i, j=0, 1, \dots, n-1\}; b=n \times n$
 $k: \text{연결성}=4: x_{ij}$ 에 입력은 $x_i(j-1), x_i(j+1),$
 x_{i-1}, x_{i+1}
 으로 부터 입력된다.
 정의로 돌아가면 G 는 노드를 기술하고 있다. 이것은, 또 3개의 요소로 형성되는 집합에 의해 좀 더 정의될 필요가 있다.

$$G: \langle A, F, D \rangle$$

여기서 A 는 노드의 입출력간 알파벳이다. $A = \{m_1, m_2, \dots, m_a\}$, 분명히 a 는 노드가 허용할 수 있는 message의 수이다. F 는 한 노드가 어떤 한번에 한 기능을 수행하여 주는 함수 집합이다. $F = \{f_1, f_2, \dots, f_v\}$; D 는 F 로 부터 선택을 바꾸기 위해 적용과정의 진보를 위해 선택변화를 주기위한 수단을 의미한다.

예를들어, 신경의 McCulloch와 Pitts⁽³⁾ 모델은 2진수로 만들어지고 $A = \{0, 1\}$, F 는 모두 선형적 분리 함수의 집합이다.

Tr 은 특별한 네트 에 대한 훈련 전략이다. 일반적으로 클램프로 적용되는 훈련 예들의 한 집합 K 의 응용을 의미한다.

$$K = \{C_1, C_2, \dots, C_k\}$$

예를들어, Hinton의 Boltzmann Machine에서, Tr 은 B 로부터 부분집합의 클램프에 포함된다. 가령 $U \subseteq B$ 클램핑은 A 로 부터 U 노드의 출력을 미리 예정된 값으로 유지하는 과정이다. 알고리즘 Tr 은 어떤 D 를 네트에 모든 노드에 적용한다. 그래서 심지어 클램프가 제거된다 해도, 클램프한 패턴은 네트워킹의 안정된 상태로서 남는다.

Te 는 네트를 시험하는 방법이다. 다시 Te 는 부분적 클램프 즉, $V \subseteq U$ 을 적용하고 몇가지 제약에 따라 네트를 동작시키는 것으로 구성된다. 성공적인 동작은 훈련 클램프 $C_j \subseteq k$ 중 한개를 결정하는 전체 U 에 대한 A 로부터 적정 값들의 재창조에 이른다.

II-2. 상태구조

Connectionist systems의 동작은 대개 상태구조의 특징에 의해 논한다.

$$S = \{S_1, S_2, \dots, S_z\}$$

그리고 T 는 S 의 집합이다.

$$T = S \times S \quad (x \text{은 Cartesian product})$$

$t_{jk} \in T$ 는 $S_j \in S$ 에서 $S_k \in S$ 로 전이되는 확률이다. t_{jk} 는 결정하여 특징을 정한 2진수이고 한 확률값에 대한 $0 < t_{jk} < 1$ 의 확률값이다. 그때 상태구조, SS 는 2개의 요소로 형성되는 집합에 의해 정의된다.

$$SS: \langle S, T \rangle.$$

시간은 이산적이라 가정한다: $t=1, 2, \dots$, 그리고 T 에 의해 결정되는 상태 변환은 단지 이 이산시간 값에서만 일어날 수 있다. $K \in S$ 에서 connectionist system의 많은 동작이 K 의 원소를 가하는 확률에 의해 기술됨을 알 수 있다. 실제로, 에너지 우물은 상태는 서로가 혹은 한 주기적인 발견으로 전이되는 K 로 부터 상태 그룹 또는 상태에 의해 논의 될 수 있다.

III. 범용의 확률적 논리 노드(PLN)

III-1. 노드함수의 논리

대부분의 신경 모델 노드가 2진수이기 때문에 그 기능은 자동 조건표(table)에 의해 표현되고 따라서, 간단한 실리콘 RAM으로 구현되어⁽⁴⁾ 있다.

사실, 2진수 제약이 가정되지 않고, 모델이 실수(real number) 그룹(집합 A 의 원소)으로도 시된다. 이것은 또한 진리표에서 수용할 수 있다.⁽⁵⁾ 정식으로 이것은 j 입력을 가진 한 노드가 다음과 같은 표준 Canonical 논리 표현에 의해 기술될 수 있다는 것을 의미한다.

$$\begin{aligned}
 & [i(1,1)\&i(2,1) \cdots i(j,1) \cup [i(1,2)\&\cdots] \cup [i(1,k) \cdots] \rightarrow m_1 \\
 & [i(1,1)\&i(2,1) \cdots i(j,1) \cup [i(1,2)\&\cdots] \cup [i(1,l) \cdots] \rightarrow m_2 \\
 & \hspace{15em} (1) \\
 & [i(1,1)\&i(2,1) \cdots i(j,1) \cup [i(1,2)\&\cdots] \cup [i(1,p) \cdots] \rightarrow m_a
 \end{aligned}$$

여기서 모든 x, y, z에 대한 $i(x,y)$ 와 $m_z \subseteq A$ 는 포함한다. \rightarrow 는 포함함을 의미한다. (\cup 는 'or' &는 'and'라 읽는다.)
 이러한 2진수 시스템을 앞에서 일반화 된것으로 불 대수식의 곱의 합이되며 식(1)으로 표현한다.
 $[\cdots\&\cdots\&]$ 식의 각 항은 minterm을 표현하고 있다. 한 j-입력노드에 대한 완전한, minterms 집합을

$$M = A \times A \times A \cdots (j\text{배}) \cdots \times A \text{라 하자.}$$

이때 노드의 함수는 이 집합의 분할 PM이다.

$$PM = \{M_1, M_2, \cdots, M_a\}$$

여기서 M_p 는 m_p 보다 큰 모든 minterms의 한 구획(block)이고, $p=1, 2, \cdots, a$ 이고 a 는 A에서 원소 번호이다.

가중된 노드와 논리적인 것의 차이는 가중된 도해(scheme)로서 일반화의 형태로 그려진다.

예를들어 부등식(2)의 3개 입력 형은 입력 $i(1), i(2), i(3)=111$ 은 1로 입력 : $i(1), i(2), i(3)=000$ 은 0으로 map 시킬 필요가 있고 어떤 매카니즘 0는 가중치가 $w(1), w(2), w(3)=111$ 이 되게하고 임계 값 $T=1.6$ 이라 가정한다.

2개의 기대하는 minterms 집합은 바르게 되었으며, 기대하는 값과 비슷한 minterm도 같은 방법으로 map 된다. 이러한 효과를 노드 일반화라 한다.

예를들어, minterm 0,1,1이 그릇된 집합이고, D가 이것을 조정할 능력이 있다면, $w(2)$ 와 w

표 1. 노드에 대한 진리표

i(1)	i(2)	i(3)	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

(3)을 0.5로 줄이므로써 해를 구할 수 있다. 이것은 자동적으로 minterms 1, 0, 1과 1,1,0을 0으로 map하게 된다. 그러나 진리표 표현은 그런 효과가 없다. m_p 를 mapping 하는 것은 D에 의해 직접 정해지고, 다른 minterms에 영향을 주지 않는다고 가정한다. 이러한 시스템에서는 기술된 일반화의 형식으로 기술되고 훈련 알고리즘에 의해 만들어지고 있음을 알 수 있다.

III-2. 확률노드

대부분의 신경 models에서 잃어버린 한 인자를 도입하는 것은 중요한 것 같다. 노드내에 한 지식, 특별한 minterm이 훈련의 과정 때문에 set 되든 그리고 setting이 일관되는 안되는 한다. 생물학적 신경에 적용전에 신경이 같은 확률로 흥분하거나 흥분하지 않는다는 것은 이미 믿어왔다.

PLN은 어떤 성질을 주기 위해 정의한다.

위의 일반적인 논리식(1)은 군집합 M_i 으로 다시 쓸수 있다.

$$\begin{aligned}
 \text{여기서 } M_1 &= \{w_1 / w_1 \text{은 } m_1 \text{에 속하는 minterm이다}\} \\
 M_2 &= \{w_2 / w_2 \text{은 } m_2 \text{에 속하는 minterm이다}\} \\
 &\vdots \\
 M_a &= \{w_a / w_a \text{은 } m_a \text{에 속하는 minterm이다}\}
 \end{aligned} \tag{2}$$

PLN은 다음과 같이 정의한다.

- (a) $A = \{m_1, m_2, \dots, m_n\}$ 일때 $A' = \{A, u\}$ 라 한다.
- (b) u 는 A 의 원소들이 같은 확률로 내보내는 노드 상태로 정의한다.
- (c) 알고리즘 D 를 적용하기전에 노드는 그것의 모든 minterms이 u 보다 크다고 주장한다.

IV. 실험 및 결과



그림 1. 신경망 실험장치
Neural Network experimental systems

그림 1은 전체 실험장치를 나타내고 있다. 또한 그림 2는 연상 메모리 장치의 블록도를 나타냈으며, 이 장치에 의해 학습을 진행한 과정은 다음과 같다.

3개 논리 노드는 고리(ring) 모양으로 연결한다.

$$P : B = \{x_1, x_2, x_3\}$$

$K : \text{연결성} = 2$

x_1 은 x_2 와 x_3 로 부터 입력을 갖는다.

x_2 은 x_1 와 x_3 로 부터 입력을 갖는다.

x_3 은 x_1 와 x_2 로 부터 입력을 갖는다.

$$G : A = \{0, 1\}$$

F : 서술된 것 같은 논리

D : minterms을 직접 가한다.

Tr : minterms은 상태 111과 000을 안정하게 만들도록 되어있다.

Te : 사용하지 않는다. 상태 구성 SS 전체에 생기는 변화에 관련된다.

우선 노드는 확률적인편을 무시하고 논리 진리표를 가정한다.

다음과 같이 그것들은 minterm mappings를 무순으로 선택하였다.

상태구조 결과는 그림 3-(a)에 나타냈다. 훈련은 상태를 000으로 클램프하고 결과 minterms

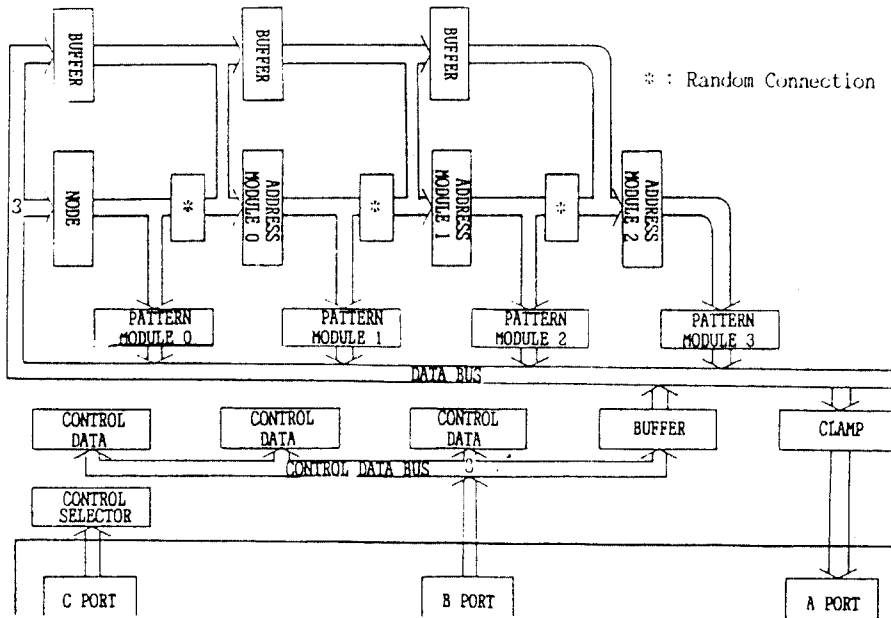
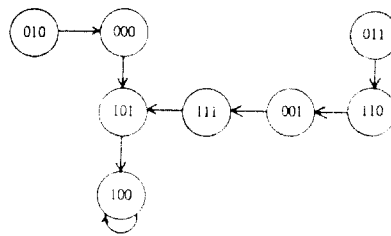
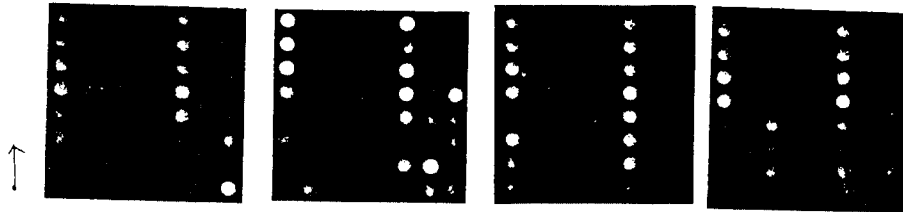


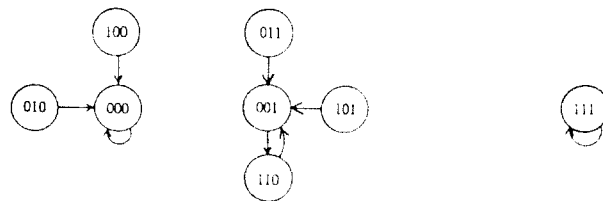
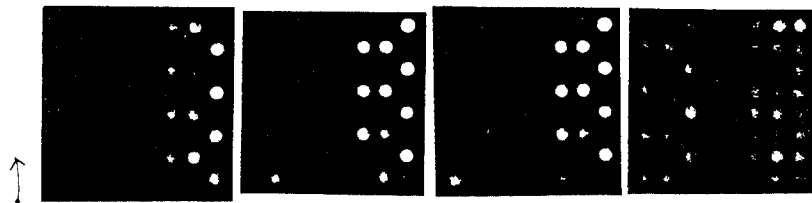
그림 2. 연상학습을 위한 디지털 신경망 블록도
Block Diagram of Digital Neural Network for Associative Learning

노드 X1	노드 X2	노드 X3
X2 X3 : X1	X1 X3 : X2	X1 X2 : X3
0 0 : 1	0 0 : 0	0 0 : 1
0 1 : 1	0 1 : 1	1 0 : 0
1 0 : 0	1 0 : 0	1 0 : 0
1 1 : 1	1 1 : 0	1 1 : 1

을 000로 mapping 한다. 이것은 111에 대해 반복되고, 결과적으로 다음 진리표가 된다.



(a)



(b)

그림 3. 결정된 상태 구조 : (a) 훈련이 없는 상태 (b) 훈련된 상태
 Deterministic state structures : (a) untrained (b) trained

노드 X1 X2 X3 : X1	노드 X2 X1 X3 : X2	노드 X3 X1 X2 : X3
0 0 : 0	0 0 : 0	0 0 : 0
0 1 : 1	0 1 : 1	0 1 : 0
1 0 : 0	1 0 : 0	1 0 : 0
1 1 : 1	1 1 : 1	1 1 : 1

대응하는 상태 구조는 그림 3-(b)와 같다. 안정한 재 입력 상태는 훈련에 의해 만들어진다. 이것이 분명하다. 그러나, 관련없는 순환에도 존재여부를 주시할 필요가 있다. 사실, 잡음 발생 후, 순환 에너지는 다시 확률의 역으로 재입력된다고 할때, 이 에너지는 순환내 관련된 상태의 모든 수에 역 비례한다. 따라서 훈련후 만들어진 3개 순환과 관련된 에너지는:

순환 주기	에너지
000	3
111	1
110-001	4

그것은 가장 낮은 에너지 레벨에 있는 잘못된 순환이다.

이것은 잡음의 목적을 무너뜨리는 함정(trap)으로 작용할 수 있다. 확률 노드는 이러한 문제를 피하는 것으로 생각할 수 있다. 위에서 확률 노드는

$$A = \{0, 1, u\},$$

로서 논의된다. 여기서 u는 노드가 같은 확률을 갖고 0 혹은 1을 내보내는 미지의 minterm mapping 이다. 초기의 각노드에서 모든 minterms가 u로 mapping 된다. 그러나 훈련후 각노드는 다음과 같은 진리표를 갖는다.

in 1	in 2	out
0	0	0
0	1	u
1	0	u
1	1	1

따라서 시스템에 대한 전체 상태 천이 표는 다음 진리표가 되므로

현 상태	차기상태
000	000
001	uu0
010	u0u
011	1uu
100	0uu
101	u1u
110	uu1
111	111

그림 4에 나타난 확률 상태 구성이된다.

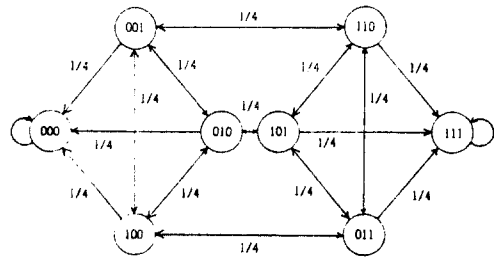


그림 4. 확률적 훈련구조 천이도. Trained, probabilistic trained structure.

천이는 한 "u"가 0이나 1이 되는 50/50 결정을 의미하는 기초위에 상태 표로 부터 계산된다. 그래서 가령 011로 부터 1 u u의 mapping은 100, 101, 110와 111의 각각에 0.25 확률 천이로 해석된다. t시간에 상태 $S(0 \leq S \leq 7)$ 인 확률은 이것이 반복 계산 되는 $P(s,t)$ 이다.

$$P(s,t) = \sum_{j=0}^7 P(j,t-1) * t_{js}$$

는 상태 j에서 상태 s로 천이 확률임.

각 상태에 대해 초기 확률을 1/8이라 가정하고 $p(s,t)$ 가 $s=000$ 과 $s=111$ 에 대해 t를 갖고 다른 상태 모두에 대해 0이되어 0.5가 되는 경향이 있다.

이 경우에 이러한 두 상태에 대해 훈련은 실제 에너지 우물(well)을 만들고, 노드의 확률 성질은 시스템에 우물에 대한 탐색 능력을 부여한다고 결론지을 수 있다.

이것은 PLN이 connectionist system에 대해 self-annealing 면을 증명하고 있음을 설명할 수 있다.

V. 결 론

본 논문은 신경컴퓨터를 구현에 따른 VISI 기술로 장치화하는데 중점을 두고 McCulloch 와 Pitts가 제안한 형식으로 실리콘 칩상에 신경 모델을 구현할 수 있는지를 알아보고, 훈련하는 방법과 설계규칙을 정하여 보았다. 설계시 RAM 으로 PLNs를 구성하고 훈련을 시켜보았다. PLNs 노드는 디지털적으로 처리하고 학습을 시킬 수도 있는 모델로서 생각할 수 있으며, 신경노드의 상태변화를 논리적 분석과 실험을 통해 확인할 수 있다. PLNs 다층 메모리를 훈련 시 설계규칙을 정하고 훈련을 한 결과 노드가 확률적으로 변화됨을 확인하였다.

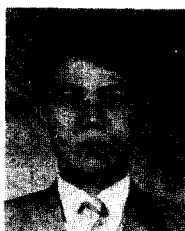
따라서, 훈련에 의해 상태주기가 훈련상태로 변화하였으며, 최종적으로 어느한 상태에 정착됨을 알수 있었다.

본 실험에서는 실리콘으로 구현하기전에, RAM 을 다층으로 구성하고 연결상태를 선으로 연결한후 디지털 컴퓨터에 의한 훈련을 실시하여 사전에 신경컴퓨터로서의 가능성을 확인하는데 그의의가 있으며, 실험결과 디지털적으로 확률적 논리노드를 실리콘상에 RAM으로 구현이 가능함을 확인 하였다. 노드수의 규모가 적은 관계로 패턴인식등의 성능을 확인할 수 없었으며 다른 모델과의 성능상의 비교를 할 수 없었다. 앞으로

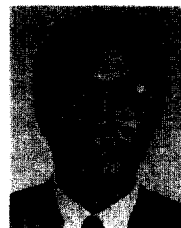
노드 수가 확장된 VISI에 의한 PLNs가 구현이 되었다면 패턴 또는 영상을 학습 및 인식을 시도해 볼필요가 있다. 또한 이에따른 학습 알고리즘개발과 훈련전략, 잡음철가등에 대한 인식실험이 기대될 수 있으며, 다른 모델과의 성능을 비교할 수 있겠다.

참 고 문 헌

1. Rumelhart, D.E & McClelland, J.L., Parallel Distributed Processing, Vol. I:Foundations (Cambridge Mass:MIT Press, 1986)
2. Hopfield, J.J Neural networks and physical systes with emergent computational abilities, Proceedings of the National Academy of sciences, USA, Vol.79, pp. 2554-2558 (1982)
3. McCulloch, W.S & Pitts, W.A logical calculus of the ideas immanent in nervous activity, Bull. Math. Biophys. 5, 115-133 (1943)
4. Aleksander, I,Bram cell to microcircuit, Electronics and power 16,48-51 (1970)
5. Rumelhart, D.E., Hinton, G.E. & Williams, R.J. Learning internal representations by error propagation. In rumelhart, D.E & McClelland, J.L.(eds.) Parallel Distributed Processing, Vol.I (Cambridge Mass:MIT Press, 1986)
6. Hinton, G.E., Sejnowski, T.J. & Ackley, D.H,Boltzmann Machines: Constraint Satisfaction Networks that Learn (Tech. Rep., CMU CS 84-119, Carnegie Mellon University, Pittsburgh, 1984)
7. Aleksander, I. The logic of connectionist systems, Neural computing Architectures, 133-142 (Cambridge, Mass:MIT Press, 1989)



金 丞 煥(Seung Whan KIM) 正會員
1955年 4月 5日生
1980年 2月: 光云大學校 電子工學科 卒業
1985年 2月: 光云大學校 大學院 電子工學科 (工學碩士)
1991年 2月: 光云大學校 大學院 電子工學科 (工學博士)



姜 俊 吉(June Gil KANG) 正會員
1945年 2月 15日生
1980年 2月~1981年 2月: 美國 MIT 電子工學科 客員 研究員
1982年 9月~現在: 國際 技能올림픽委員 審査委員
1984年 3月~現在: 光云大學校 電子工學科 教授

1985年 9月~現在: 光云大學校 電子技術研究所 所長