

人工網膜のシミュレーション

梅野正義・野間敏弘*・清水昇市**

共通講座教室

(1984年9月8日受理)

Simulation of Artificial Retina

Masayoshi UMENO, Toshihiro NOMA*, Shoichi SHIMIZU**

Department of Engineering Science

(Received September 8, 1984)

By learning an excellent construction in living body, we have done the computer simulation for approaching the artificial retina. This model has pattern recognition with parallel processing and self-organization. We investigated visual functions, and then pointed out the conditions for working these functions in the hardware system such as with semiconductor devices.

1. 序 論

現在の情報化社会を生み出した最も大きな原動力は、なんといってもそれを支える科学技術の進歩であるといつてよいだろう。様々な分野で科学技術が進歩し、足における車、視覚におけるカメラなどのように、人間の能力を補うべく発達し、ある意味では人間の機能を上まわる能力をそなえるものまで実現している。しかし、まだ未知の分野・工学機械が太刀打ちできない分野も数多く存在する。そのような分野において、生体の優れた機能を学び、その仕組みを工学的に応用していくことは、今後増々重要になってきている。

このような現状の中で我々は、高等動物において最も重要な感覚である視覚に注目し、生体のパターン認識機能を目標とした情報処理システムである「学習を通じて自己のパラメータを変更し、パターン認識能力を獲得していくハードウェアシステム」について考察した。

本研究の特徴としては次の点が上げられる。

- i) 従来のパターン認識で用いられているパターンマッチングのようにあらかじめデータとして与えられているパターンとの比較によりパターンの分類を行なうのではなく、自己パラメータを学習を通じて最適値にすることによりそれを行なっている点。
- ii) デジタル計算機による演算は行なわないで、ハードウェアそのものの特性を利用することにより求める機能を実現しようとしており、それにより高速処理を目指

している点。

- iii) ハードウェア指向のモデルではあるが、学習機能を有するために、適応範囲が広い点。

本研究の最終目標は視覚としての人工網膜であるが、我々の目指す第一到達点としては、モデルのハードウェア化であり、しかもモノリシックなデバイス化である。今回は、そのための、ハードウェア向きのパターン認識モデルの作成を中心に行なった。

2. パターン認識モデルの作成とその動作原理

2.1 モデル作成の条件と仕様

生体の構造・機能を見本にした自己組織モデルについて、システムレベルでの学習モデルの基本となる点としては、可塑シナプスをもつニューロンのシナプス荷重を変化させるということをはじめとし、

- i) 非線形関係と閾値を用いて信号を処理している。
- ii) 受容野内にあるニューロンから信号を受けとり、それを空間加重している。
- iii) 平面的な層構造をなす場合が多い。
- iv) 抑制信号の作用により、あるパターンのクラスは出力層内の一つのニューロンに対応させることが多い。などが挙げられ、これをもとに、仕様を定める。

仕様

機能) パターン認識機能

パターン認識能力は、学習によって強化・形成される。その際、教師信号は用いずに、自己組織の方式によって

行なう。

構造) 階層構造

各層にニューロンモデルを二次元に配置する。最下層を入力層とし、最上層を出力層とする。

処理形態) 並列信号処理

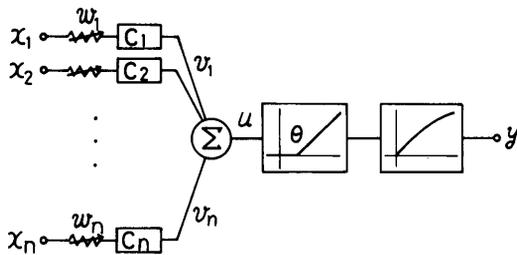
条件

ハードウェア化ということから次の条件になるべく合うものを考えることにする。

- i) 結合が少ない・構造が単純。
- ii) 複雑な処理を必要としない。
- iii) モデルの動作を実現する素子・現象がある。

2.2 モデルの構造

[ニューロンモデルレベル]



$$v_i = x_i \cdot C_i \cdot w_i \quad V_i = x_i \cdot C_i \cdot W_i$$

$$u = \sum v_i = \sum x_i \cdot C_i \cdot w_i \quad U = \sum V_i$$

$$y = f(u - \theta) \quad Y = f(U - \theta)$$

Fig. 1 Newron model of synaps

従来のニューロンモデルとの違いは、入力可塑性によって重み付けされた後、もう一度結合係数倍されるところである。この結合係数 C_i の値は、その C_i が対応している x_i の座標平面の位置によって決まっている定数であり、常に一定の値しかとらない。 $x_i (i=1, \dots, n)$ は入力の値、 $w_i (i=1, \dots, n)$ はシナプス荷重値で、 $a \sim b (0 \leq a < b \leq 1)$ の範囲で一様に分布している。 v_i はシナプス荷重と結合係数により重み付けされた後の入力 x_i の値で、 u は $v_i (i=1, \dots, n)$ の空間加重和で、以後この u の値をポテンシャルと呼ぶ。 u は入力値・シナプス荷重・結合係数の値により一意に決定される。 θ は閾値であり、 u から θ を引いた $(u - \theta)$ が出力関数 f の入力となる。 y は出力である。なお、入力信号はすべて興奮性の信号であり、 $v_i \geq 0$ とした。 w_i, v_i, u, y を確率変数として取り扱うときには、それぞれ大文字で W_i, V_i, U_i, Y_i と表記する。

[システムレベル]

先に示したモデルが2次元面に配置されている。ニューロンの入力、すべて前段の層から入力される。各記号に対する意味は

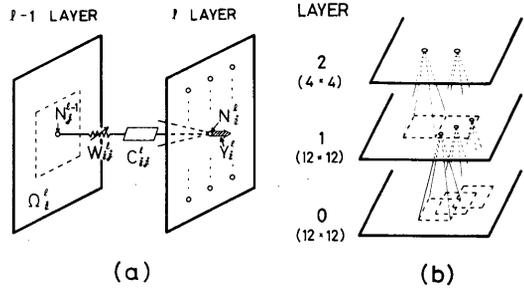


Fig. 2 Connecting area

$N_i(1)$: 1層の平面座標 $i = (i_1, i_2)$ に位置するニューロン

$y_i(1)$: $N_i(1)$ の出力の値

$\Omega_i(1)$: $N_i(1)$ の(1-1)層に対する結合範囲、受容野

$w_{ij}(1)$: $N_i(1)$ と $N_j(1-1)$ を結ぶ可塑性シナプスの値

$C_{ij}(1)$: $N_i(1)$ と $N_j(1-1)$ を結ぶ結合係数値

$$u_i(1) = \sum_{N_j(1-1) \in \Omega_i(1)} y_j(1-1) \cdot w_{ij}(1) \cdot C_{ij}(1)$$

求まる $N_i(1)$ のポテンシャル

$\theta(1)$: 1層のニューロンに共通の閾値

出力は次の関係式で表わされる。

$$y_i(1) = f(u_i(1) - \theta(1)) \quad (1)$$

ここで注意することは、ニューロンレベルではポテンシャルの入力は x_i であったが、システムレベルでは前段の出力を入力とするので、 x_i の代わりにニューロン $N_j(1-1) \in \Omega_i(1)$ の出力 $y_j(1-1)$ を用いる。システムの入力層は0層で表わし、その値を $y_i(0)$ で表わす。同様にシステムの入力層は第L層で、その出力値は $y_i(L)$ である。

Fig. 2(b)は受容野の広がる様子を示したものである。図からもわかるように、あるニューロン受容野は前段・前々段と深くなるに従って広がっていく。

[本モデルの構造的特徴]

構造は他のモデルと比較してかなり単純化してある。ニューロンの入力信号は、必ず前段のしかも、そのニューロンが存在する面内の位置と同じ位置を中心とした限られた範囲内からしか取り込まないよう制限がしてある。また、抑制信号は用いず、興奮性の信号の伝達のみで信号処理を行なう。これらの結果、本モデルにおいては0-1, 1-2層間の結びつきはほとんど同じであるため、同じ構造のものを積み重ねるだけでシステムが構成できるという点、信号の種類が一種類なので信号処理システムが一系統で済むという利点がある。

2.3 モデルの動作原理

本モデルでは次の様に特徴量を定める。

(a) 受容野パターンに対しては、

- (1) ポテンシャルの大きさ,
- (2) 受容野内出力パターン (発火ニューロン位置)
- (b) 入力パターンに対しては,
 - (1) 座標 (シナプス後ニューロンが発火する受容野パターンの位置)

「受容野パターン」とは、受容野内で発火している (出力 > 0 を出している) シナプス前ニューロンの二次元配置パターンのことで、受容野レベルの比較的狭い領域のパターンのことを指し、シナプス後ニューロンの入力となる。「入力パターン」とは、システムに入力されるパターンを指し、入力層の大きさと同じ大きさをもつ。

パターンが入力されるとこれらの特徴量が計測され、その結果は、ニューロンの発火・不発火で表示される。

a-1) ポテンシャルの大きさ

ポテンシャルの大きさは、 $u = \sum v_i$ である。

ここで、
$$\begin{cases} f(u) > 0 & (u > 0) \\ f(u) = 0 & (u \leq 0) \end{cases}$$

なる出力関係 $f(u)$ を選べば、ニューロンが発火するかしないかは、 u と θ の大小関係で決まり、 $u > \theta$ で発火・ $u \leq \theta$ で発火しない。

今考えているニューロンモデルには、抑制信号がないので、あるニューロン N の受容野パターン p_0 に対するポテンシャル $u(p_0)$ ・受容野パターン p_0 より発火しているニューロンの数が多い受容野パターン p_1 に対するポテンシャルを $u(p_1)$ と表わすと明らかに $u(p_0) \leq u(p_1)$ である。

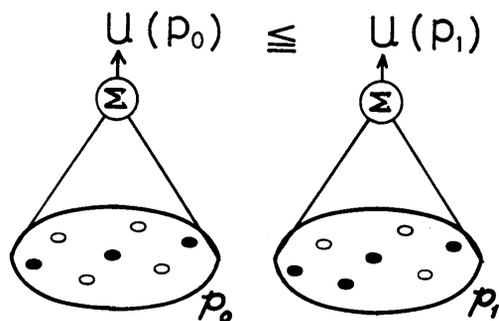


Fig. 3 Relation between Pattern and Potential

a-2) 受容野内出力パターン

ある受容野パターンに対してニューロンが発火するかしないかを最終的に決定するのは、そのパターンに対するポテンシャルの大きさであり、粗くいえば受容野内ニューロン発火率 (発火しているニューロンの数/受容野内ニューロンの数) できまる。しかし、受容野内ニューロン発火率が同じでもパターンの違いによってポテンシャルは異なることがある。これは、シナプス荷重 w や結合係数 c が結びついているニューロンによって異なるからであり、閾値 θ の値いかに同じ発火率の受容野パ

ターンであっても、あるものはシナプス後ニューロンが発火するが、あるものは発火しないといったことが起る。

b-1) 受容野パターンの位置

入力パターンは受容野パターンの和集合と考えられ、その情報処理の結果は、各受容野パターンに対する各々のニューロンの反応を統合したもので表わされる。

[認識方法]

本モデルの認識の方法は、「情報源 I に含まれる入力パターンのクラス $c_k (k=1, \dots, m)$ に対応するニューロン N_k が出力層に存在し、(N_k は一つとは限らない) c_k の要素 p_k がシステムに入力されると N_k が発火する」という形式を用いている。最初から N_k は決まっているのではなく、学習 (パターンの提示) を行なうことにより、出力層上のあるニューロン N をクラス c_k に対応するニューロン N_k にならしめるのであり、この対応づけ (選出) を規定するものが、学習のアルゴリズムである。

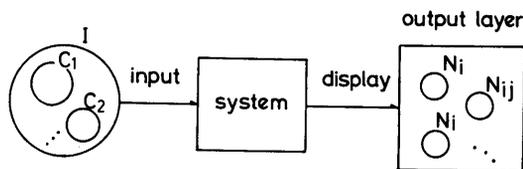


Fig. 4 Schematic diagram of recognition

「パターンが分離された状態」というのは、入力パターンのクラスが異なれば、出力層での発火パターンも異なる状態を指す。これは、各クラス間で出力を比較すると、すくなくとも一ヶ所のニューロンの反応は異なっていることを意味している。

[学習の方法]

入力パターンに対する代表選びは、先に述べた特徴量を計測した結果をもとに行なわれる。入力されるパターンに対する発火条件を内部パラメータの変化によって満たしたニューロンはその入力パターンの属するクラスを代表するニューロンとなる。

1) 閾値 θ の設定の仕方

閾値 θ は発火に対しての感度を制御する値であり、設定にあたっては、まず、どのようなパターンに対してニューロンを発火させるのかを決める必要がある。それは、作成者の意図するところにより異なる。つまり、有意パターン (意味を持つパターンのこと) と無意パターン (雑音など) のそれぞれのポテンシャルを u_a, u_n とすると、閾値 θ を

$$u_n < \theta < u_a \tag{2}$$

になるように設定すればよい。

2) ポテンシャルの計算方法

ポテンシャル u は次式で表わされる。

$$u = \sum c_i \cdot x_i \cdot w_i \tag{3}$$

C ₁	C ₂	C ₃
C ₄	C ₅	C ₆
C ₇	C ₈	C ₉

Fig. 5 Connect Coefficient

今回作ったモデルでは、受容野として Fig. 5 のように 3 × 3 のものを考えた。図の小さな四角形一つにシナプス前ニューロン一つが対応する。

$$\begin{cases} c_5 > c_i & (i = 1 \sim 9, i \neq 5) \\ c_i = c_j & (i, j = 1 \sim 9, i, j \neq 5, i \neq j) \end{cases} \quad (4)$$

の関係が成り立つように設定してある。

入力層における信号強度は、発火している時に 1、そうでない時に 0 とした。

w_i は、初期時には a から b の間に一様分布 (0 ≤ a < b ≤ 1) している。

w_i が確率変数であるからその係数倍の和も確率変数となり、これを U と記す。すると、(3)は、

$$U = \sum c_i \cdot x_i \cdot w_i$$

と書き換えられる。U の最大値 u_{max}、最小値 u_{min} は次のようになる。

$$\begin{cases} u_{max} = \min_x (x) \\ F_U(x) = 1 \\ u_{min} = \max_x (x) \\ F_U(x) = 0 \end{cases} \quad (6)$$

(6) は、u_{max} が F_U(x) = 1 を満たす x のうちで最小のものであり、u_{min} が F_U(x) = 0 を満たす x のうちで最大のものであることを示している。U の分布関数が求まると、それと θ との関係は Fig. 6 で示すようになる。この分布関数が有意パターンのポテンシャルのものだとすると、

$$u_{min} < \theta < u_{max} \quad (7)$$

が成り立つように θ を選ぶ必要がある。

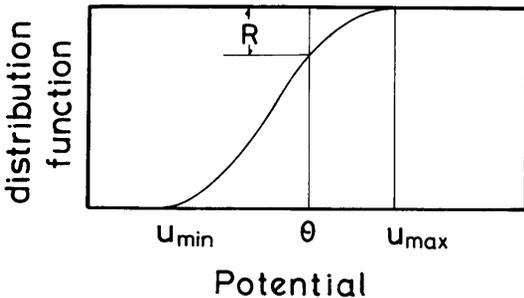


Fig. 6 Distribution function of potential

閾値 θ を設定したときのそれぞれのパターンに対する発火率 R は、

$$R = (1 - F_U(\theta)) \quad (0 \leq R \leq 1) \quad (8)$$

で表わされる。

3) シナプス荷重 w に関して、

学習を通じて変化する内部パラメータの中で最も重要な項が、このシナプス荷重であるが、w に関しては、次の3点が重要である。

① 変更をする際の条件

連合作用説による方式で変更を行なう。

i) X ∩ Y 型: X (入力) の発火かつ Y (出力) の発火の際、その間のシナプス荷重を強化していく。つまり、その入力が出力に対して役に立つものとみなし、その信号の伝達を良くする。

ii) X ∩ Y 型: X の不発火かつ Y の発火の際、その間のシナプス荷重の働きを弱める。つまり、その入力が出力に対して役に立っていないものとみなし、その伝達を悪くする。

② 変更する量

w の変更については、

$$w_i(t+1) = (1-c)w_i(t) + dr_i(t) \quad (9)$$

と表わすが、学習信号 r_i(t) は w の変更条件により

$$dr_i(t) = \begin{cases} \xi x_i(t) - \eta & (y > 0 \text{ のとき}, \xi > \eta > 0) \\ 0 & (y \leq 0 \text{ のとき}) \end{cases} \quad (10)$$

となり、すると y > 0 の場合 (9) より

$$w_i(t+1) = w_i(t) + \xi x_i(t) - \eta \quad (11)$$

y = 0 の場合は、w_i は変化せず、

$$w_i(t+1) = w_i(t) \quad (12)$$

ここで c = 0 としたのは、a ≤ w_i ≤ b (0 ≤ a < b ≤ 1) と設定してあるため c > 0 としなくても、w_i が発散することがないからである。

③ 初期設定

w の初期値の設定がポテンシャル u の値の分布を決定する。端的に言えば、|b - a| が大きいときつまり分散が大きくなる時は u の分散も大きくなり、小さいときはその逆になる。シナプス荷重 w の値が広い範囲で分散しているということは、(11) で表わされる w の変更が何回が行なわれた状態と等価の状態になっているとみることが出来る。つまり、学習を行なう前からあるパターンに対しては発火しやすく、あるパターンに対しては発火しにくくなっている。この条件下では、特定パターンに対する指向性が高まっておりパターンの違いによる反応の違いが大きくでるが、そのことは最初から反応するパターンが固定化されることを意味し、そのため有意パターンであってもポテンシャルが非常に低くなる場合が起こる。その結果有意パターンが不発火となり、必要な情報が欠落する恐れが生じる。この逆に、狭い範囲で分散しているときには限られた範囲で θ を設定し、発火率を制御できるが、一方で個々のパターンに対する反応に差が

出なくなってしまう。

4) θ の変更に関して

我々の希望としては、①ニューロンに特定パターンへの指向性を与えたい。②有意パターンの情報の欠落は避けたい。ということがあるが、そもそもこの二つの事象は相反し、各々の最適状態を同時に与えることはできない。そこで、全体の最適状態を求めるために、まず①の最適状態を定め、除々に②の最適状態の方向に変化させることを考える。①の最適状態は w の初期幅を大きくとり θ を u_{\max} に近い値にすることにより得られるが、このままでは②の最適状態の方向に変化しない。この問題に対する解決策として θ を時間の関数として変化させる方法を考える。具体的な θ の変更は、まず u_{\max} と u_{\min} の間を等間隔に n 分割する。 $u_{\max} = \theta_0$, $u_{\min} = \theta_n$ とし、その他の各分割点を θ_i とする。 ($i = 1, \dots, n-1$) ある学習サイクルの時間を ΔT としたとき、学習を開始して $(i-1) \times \Delta T$ ほど経過した直後の θ の値を θ_i にする。この様に $\theta(t)$ を設定しパターン分離がされた時点で、学習が完了したものとして θ の変更も終了する。 i が小さい時はポテンシャルの大きい限られたパターンだけに発火するのでパターンの選択性は高い状態にある。この時パターン分離が完了していれば出力層で発火しているニューロンの数は少なく一つのパターンクラスに対応して発火するニューロンも少なくすむ。しかし、運悪くパターンの分離が完了しなかった時は θ の値を上げて発火するニューロンを増やしてやる。これを繰り返し、パターン分離が完了するまで行なう。繰り返し回数が少ないほど一つのパターンクラスに対応して発火するニューロンは少なく、パターン分離がうまくいっていることになる。いずれにしても、 θ を時間関数として変化させることにより、

有意パターンに対しニューロンを発火させることは保障される。

3. モデルシミュレーションおよびハードウェア化の条件

作成したモデルをコンピュータによりシミュレーションさせ、その動作が所期の目的通りになるか検証してみる。

3.1 シミュレーションの方法

シミュレーションで学習させたパターンは、“A”、“B”、“C”、“D”、“E”、“F”の6つの文字パターンであり、用いた値は Table. 1 に示す。シナプス荷重値はパターンに対する指向性を高めるために最も分散の大きい0.0~1.0とした。有意パターンとして受容野の中央のものを含んで3つ以上発火しているものを設定した。すると、 $u_{\max} = 0.4$, $u_{\min} = 0$ が得られる ($w_{\max} = 1$, $w_{\min} = 0$, $x = 1$, $c_i = 0.2$ or 0.1)。すると、 $\theta_0(1) = 0.4$ としなければならないが、数回シミュレーションした結果 $\theta(1) = 0.3$ ぐらいまでは有効な出力が出てこないため、 $\theta(1) = 0.30$, $\theta(2) = 0.10$ とした。 ΔT すなわち θ_i が一定である学習サイクルの時間は、パターンを21サイクル学習させる間とし、この間に強化すべき w が1になるよう $\xi = 0.05$ とした。 η は $\xi \gg \eta$ にとるが、今回は $\xi = 10\eta$ になるようにし、 $\eta = 0.005$ とした。

3.2 シミュレーション結果

Fig. 7 に学習の進行状況を示す。各パターンごとに、3層の出力が左から順番に表示されている。数字は出力

Table. 1

number of patterns	6
layer's width	0: 12×12 1: 12×12 2: 4×4
receptive field's width	1: 3×3 2: 3×3
connect coefficient	$C_5 = 0.20$ $C_i = 0.10$ ($i = 1 \sim 9, \neq 5$)
initial value of weight	0.0 ~ 1.0 (uniform distribution)
Δ weight	ξ : 0.05 η : 0.005
initial value of threshold (θ_1)	1: 0.30 2: 0.10
Δ threshold ($\theta_{i+1} - \theta_i$)	1: 0.030 2: 0.015
output function	$2 \times \arctan(2 \times u)$
learning times / ΔT	$21(\text{cycle}) \times 6(\text{pattern}) = 126$

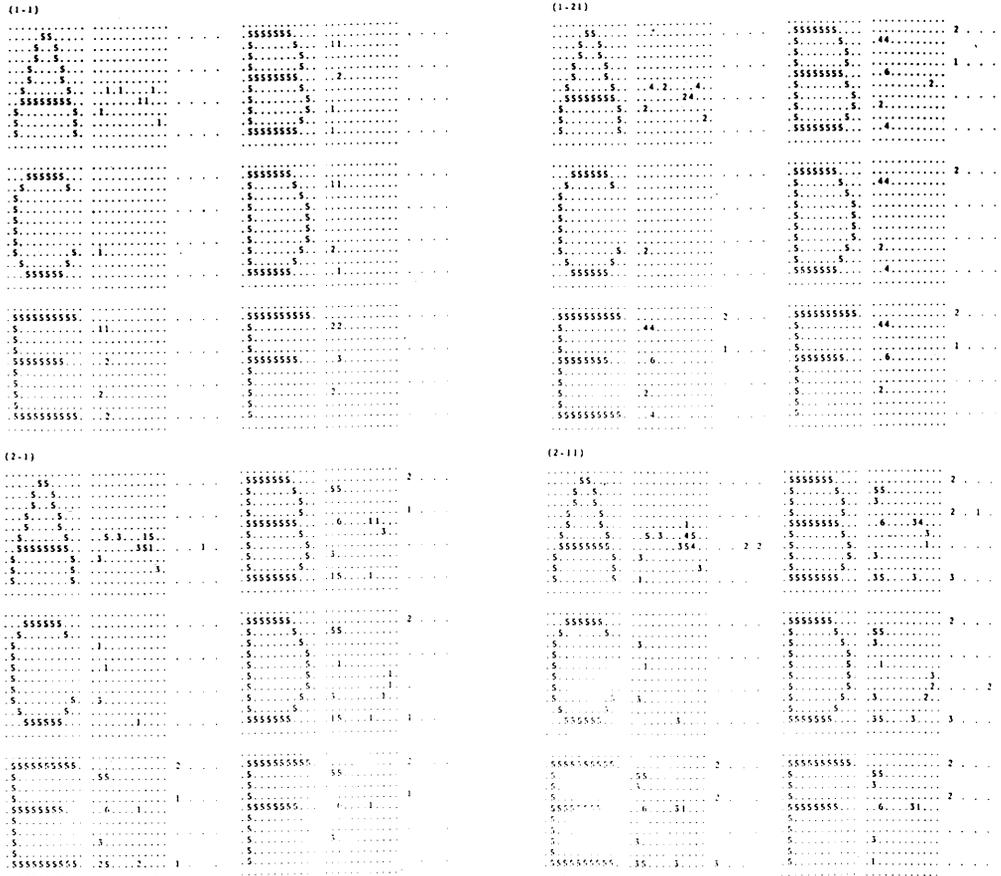


Fig. 7 Result of output

強度を表わし、実際の値を5倍したものより小さくない最小の整数が表示されている。パターン“A”の左上にある数字が θ_i の i の数字と θ_i においてパターンが何サイクル提示されたかという学習回数を表わす。

結果として θ_2 での11回目のパターンの提示が終了した時点で、6つのパターンの出力パターンがすべて違ったパターンを示した。ただし、“C”に対しては、第2層のニューロンは、一つも発火していない。

予想されたように、左上の部分“B”、“D”、“E”、“F”などで共通している部分を受容野にもつ2層目のニューロンが最初に発火した。その後、各パターン独自の特徴点に対応する部分で発火が起きている。

まとめると、モデルは所期の目的通りほぼ動作することが確認された。しかし、まだ雑音や発火強度等で問題があり、これに対してはパラメータの設定の仕方に一考を要する。

3. 3 ハードウェア化の条件

今回作成したモデルはハードウェア化を目指すという

ことのでかなりシンプルなものを目指したが、それにしてもまだハードウェア化に関しては多くの課題があり、今後モデルの改良と同時に、ハードウェアの方からのアプローチが必要である。

[モデルのハード化に際して必要とされる機能]

まず、システムを大きく四つに分ける。

①入力部：外部から入力されてきた信号を適当な信号形態に変換して信号処理部に渡す。一般的に、外部からのパターン信号は光信号であることが多いと考えられるので、ここには光-電変換素子・光-光増幅器などを使うことが考えられるのであろう。

②表示部：役割としては、入力信号のパターンのクラスを表示するわけであるが、信号処理部で処理された信号をシステム外に出力する機関である。処理した内容を人が見て利用するとなれば光パターンによる表示等が考えられるが、他のシステムの入力につなぐとしたら電気信号による出力も考えられる。

③信号処理部：システムにおいてメインとなる部分である。モデルにおいて行なわれている学習動作を実現する

ためには、

- ・荷重変更機能として→信号伝達制御機能
- ・荷重値の保存のための→メモリー機能
- ・学習を行うか否か調べるための→条件検出・判断機能、

信号の形態としては、電気信号・光信号・それらのハイブリット等いろいろなものが考えられる。

④電源部：システム各部に電力を供給する。

以上のうち、①②④に関しては光受容素子としてフォトダイオード、CCD、フォトトランジスタ等、発光素子として半導体レーザ、LED等があり、電力供給とあわせあまり問題はない。一番の問題は、③の信号処理部にある。

[ハード化における課題]

ハード化における課題の中で「並列化」「結合の空間的広がりの実現」の2点が特に難しいといえよう。

テレビジョンなどの信号処理と同じように時系列信号により処理を行えば、結線も少なくすむが、並列処理を今の段階で実現しようとする膨大な数の結線が必要となる。また、同様の技術的な難しさは、「結合の空間的な広がりの実現」にもあるといえる。この問題の解決策として、隣り合う同士を結びつけてしまう方法なども考えられるが、特性の問題も含め、今後の課題である。

この他にもハード化を実現する上で多くの困難があるが、現在の超LSI作製における微細加工・超格子および面発光デバイスなどを用いた新しい半導体デバイスの技

術開発等々の技術発展は、やがてこれらの問題の解決の糸口を与えてくれるように思われる。

4. 結 論

本研究で得られた結果を以下に要約する。

1. 従来の自己組織モデルと比較して、非常に単純な構造をもった自己組織モデルを作成した。
2. ポテンシャルの概念を用いることにより、パターンに対するニューロンの発火、不発火の説明を明瞭にした。
3. パターンの選択性を高く保ちながら、情報の欠落を起こさせないための方法として、閾値を学習回数により変化させればよいことを示した。
4. 受容野パターン同士の関係づけを行ない、それとモデルの出力との関係を明らかにした。
5. モデルのコンピュータシミュレーションを行ない、その結果、与えたパターンが分離されることを確認した。
6. ハード化における必要な機能を明らかにし、さらに今後の課題を指摘した。

なお、本研究を行なうにあたり、御支援いただきました情報工学科教授(現学長)太田正光先生に感謝いたします。