

高集積化をめざした簡略型ニューロンの設計

安藤道則，太田則一，渡邊章弘

Design of Simplified Neuron for High Integration

Michinori Andoh, Norikazu Ohta, Akihiro Watanabe

要 旨

ニューラルネットワーク (NN) をハードウェア化したニューロ LSI は、近年の NN の実用化とともに、数多く開発されている。しかしながら、画像、音声などの分野における大規模な問題を、リアルタイムで処理するためには、集積度、処理速度ともにまだ十分ではない。そこでデジタルニューロ LSI を対象として、集積度を向上させるために、ニューロンモデルを簡略化する方法を検討した。

簡略化したモデルとして、結合係数を 2 のべき乗で表現するニューロンモデルを提案した。このモデルにより、結合係数と入力との乗算をシフト演算に置き換えることができ、回路規模低減が可能

である。

簡略型モデルの有効性を調べるために、文字認識・学習シミュレーションを行った。その結果、文字認識能力に関しては、従来型モデルとほぼ同等の認識能力を持っていることを確認した。学習シミュレーションの結果、収束に要する時間が長くなる問題点がわかった。

そこで認識機能に限定して、簡略型ニューロンを設計した。ニューロン 1 個のサイズを $1.47 \times 1.66\text{mm}^2$ で実現できた。また LSI 内部のネットワーク構造を、柔軟に可変できる構成を考案し、ニューロンの設計に反映させた。

Abstract

Recently, various neural network (NN) LSIs have been developed with practical applications of NN. In real-time handling of large problems in fields such as image and speech signal processing, however, the integration and processing speed of these LSIs are still insufficient. For higher integration of a digital NN LSI, a method of simplifying neuron models was examined.

A new neuron model with weights expressed with power of two has been proposed as a simplified model. In this model, multiplications of the input signal and the weight are replaced with shift operations. Therefore, the circuitry scale reduction is achieved.

Character recognition and learning simulations were carried out to examine the validity of the simplified neuron model. As a result, it was found that the simplified model had the character recognition ability almost equal to these of conventional models. The learning simulation revealed that the amount of time necessary to converge with the simplified model were longer than these with conventional ones.

Finally, a simplified neuron whose application was specified to the recognition was designed. The total pattern area of one neuron was $1.47 \times 1.66\text{mm}^2$. The new structure, by which the network structure of the NN LSI could be changed has been proposed and used for the design of neurons.

キーワード

ニューラルネットワーク，LSI，高集積化，ニューロンモデル，結合係数，2 のべき乗

1. はじめに

近年, NNが認識問題等に多数応用されており, それにともない各種のニューロLSIの開発が活発に行われている。しかしながら, 画像, 音声などの分野における大規模な問題を, リアルタイムで処理するためには, 集積度, 処理速度ともに不十分であり¹⁾, そのようなLSIの実現は, 現状のVLSI技術を用いてもかなり困難である。

これまで様々なニューロLSIが提案されている。その中でアナログ方式は, 高い処理速度を達成しているが²⁻⁴⁾, 回路のオフセット電圧やバラツキ等の精度的な問題があり, 特にNNが大規模になった場合, この問題が顕著になると予想される。これに対して, デジタル方式では精度の問題はなく, 大規模なNNシステムも実現できると考えられる⁵⁻⁸⁾。またアナログの場合, 書き換え可能なメモリの実現は困難であるが, デジタルでは容易に実現できる。

一方, デジタル方式の問題点としては, ニューロンの回路規模が大きくなるため, 集積度が低下することがあげられる。そこで我々は, デジタルニューロLSIを対象として, ニューロンモデルの簡略化により, 集積度を向上させる方法を検討した。

本論文では, まず2章で, ニューロンモデルの簡略化の方法について説明する。次に3, 4章で, 簡略化したモデルの有効性を調べるために, 文字認識, 学習シミュレーションを行った結果について述べる。次に5章で簡略型ニューロンの設計結果についてまとめる。

2. 簡略型ニューロン

本論文では, Fig. 1に示すような中間層が1層のみの階層型NNを対象とした。中間層あるいは出力層の各ユニットは, 一つ前の層の全てのユニットからそれぞれ信号 $v_i^{(i)}$, $v_j^{(h)}$ を受けとり, その重み付け総和 $u_j^{(h)}$, $u_k^{(o)}$ を式(1), (2)に従って計算する。

$$u_j^{(h)} = \sum_i a_{ij} v_i^{(i)} + \theta_j^{(h)} \quad (1)$$

$$u_k^{(o)} = \sum_j b_{jk} v_j^{(h)} + \theta_k^{(o)} \quad (2)$$

但し, a_{ij} は, 入力層の*i*番目のニューロンから中間層の*j*番目のニューロンへの結合係数であり, b_{jk} は, 中間層の*j*番目のニューロンから出力層の*k*番目のニューロンへの結合係数である。 $\theta_j^{(h)}$ および $\theta_k^{(o)}$ は, それぞれ, 中間層の*j*番目のニューロンおよび出力層の*k*番目のニューロンのバイアスである。

式(1), (2)により計算した重み付け総和を, 式(3)に示すシグモイド関数により変換し, ニューロンの出力 v_i を計算する。

$$v_i = f(u_i) = 1 / (1 + \exp(-u_i)) \quad (3)$$

式(1)~(3)の演算は各ニューロンで並列に実行する必要がある。この演算をするためのニューロンのブロック構成の一例をFig. 2に示す。入力と結合係数との積和演算は乗算器と加算器を用いて実行される。これをデジタル回路で構成した場合, 回路規模の大部分を乗算器が占めることになる。従ってニューロンの回路規模を縮小するためには, 乗算器を簡略化することが効果的である。

そこで我々は, 乗算器を簡略化する方法として, 結合係数を固定小数点表現から2のべき乗表現に置き換える方法を導入した。2のべき乗表現を用

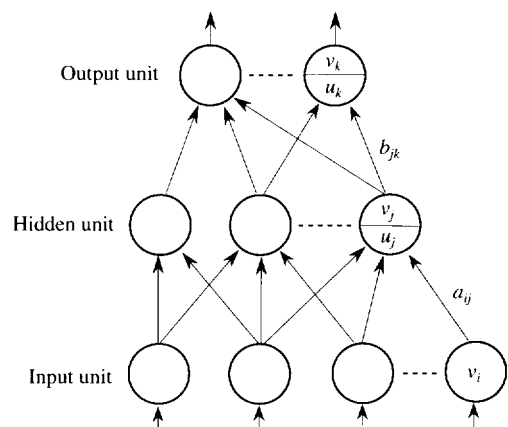


Fig. 1 A multilayer neural network.

いれば、従来の固定小数点表現での乗算を単純なシフト演算に置き換えることができるので、回路規模を縮小することができる。

3. シミュレーションによる文字認識能力の評価

モデルを簡略化することにより、適用対象が限定される可能性があることから、文字認識シミュレーションを行い、モデルの有効性を評価した。

3.1 文字認識シミュレーション

文字認識はFig. 3に示す、0, 1の2値で表現される 7×5 行列の形のアルファベットを対象に実施した。Fig. 3のパターンに対応する教師信号をTable 1に示す。学習はバックプロパゲーション (BP) アルゴリズムに基づき、オフラインで行った。学習後の結合係数を2のべき乗に変換したモデルで認識シミュレーションを行った。

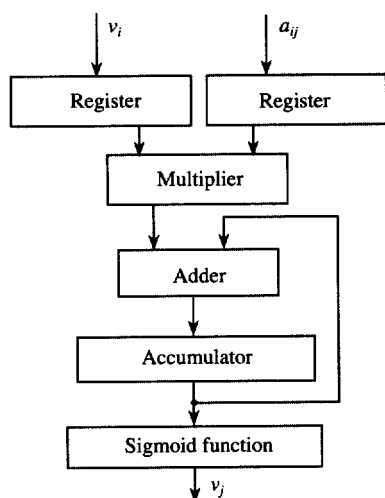


Fig. 2 Example of neuron block diagram.

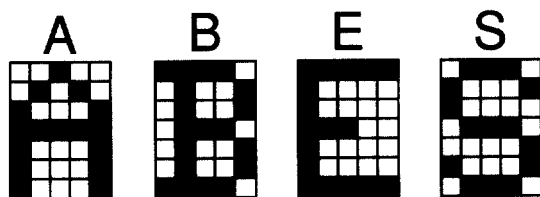


Fig. 3 Character patterns.

3.2 結果

Table 2, 3に認識結果の一例を示す。シミュレーションに用いたNNのニューロン数は入力層35, 中間層16, 出力層4とした。

Table 2は、オフラインで学習した結合係数をそのまま用いて認識した結果である (従来型モデル)。Table 3は、オフラインで学習した結合係数を2のべき乗に置き換えた簡略型モデルにより認識した結果である。簡略型モデルによる認識は、従来型モデルによる認識とほぼ同等な結果が得られた。

次にFig. 3に示す学習パターンに対して、ノイズが混入した文字パターンの認識能力を評価した結果について述べる。学習パターンに対してそれ

Table 1 Teacher signals.

Input Pattern	Output			
	A	B	E	S
A	1	0	0	0
B	0	1	0	0
E	0	0	1	0
S	0	0	0	1

Table 2 Recognition results (conventional model).

Input pattern	Output			
	A	B	E	S
A	<u>0.996</u>	0.002	0.003	0.003
B	0.003	<u>0.995</u>	0.003	0.004
E	0.003	0.003	<u>0.996</u>	0.003
S	0.003	0.004	0.003	<u>0.995</u>

Table 3 Recognition results (simplified model).

Input pattern	Output			
	A	B	E	S
A	<u>0.969</u>	0.016	0.016	0.016
B	0.016	<u>0.945</u>	0.008	0.023
E	0.008	0.008	<u>0.961</u>	0.031
S	0.016	0.008	0.016	<u>0.969</u>

ぞれランダムに3ドットノイズが混入した場合の入力パターンをFig. 4に、認識結果をTable 4, 5に示す。Table 4が従来型モデルによる認識結果、Table 5が簡略型モデルによる認識結果である。簡略型モデルでも、この程度のノイズ量であれば十分認識可能であることが確認できた。

次にアルファベット26文字を簡略型モデルにより認識するシミュレーションを行った。シミュレーションに用いたNNのニューロン数は入力層35、中間層20、出力層26とした。認識結果の例をTable 6に示す。他の文字についても同程度の認識能力であり、簡略型モデルでもアルファベット26文字を認識できることを確認した。

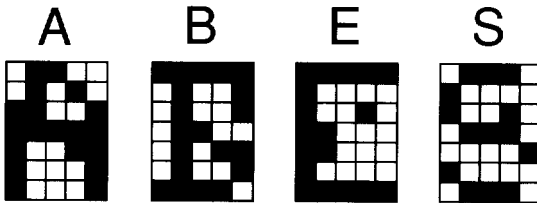


Fig. 4 Character patterns corrupted by random noise.

Table 4 Recognition results (conventional model).

Input pattern	Output			
	A	B	E	S
A	<u>0.992</u>	0.009	0.006	0.001
B	0.002	<u>0.994</u>	0.010	0.003
E	0.003	0.007	<u>0.996</u>	0.002
S	0.004	0.001	0.020	<u>0.983</u>

Table 5 Recognition results (simplified model).

Input pattern	Output			
	A	B	E	S
A	<u>0.953</u>	0.047	0.023	0.008
B	0.008	<u>0.930</u>	0.031	0.016
E	0.008	0.016	<u>0.961</u>	0.023
S	0.016	0.000	0.070	<u>0.930</u>

4 . シミュレーションによる学習能力の評価

4 . 1 方法

簡略型ニューロンモデルでもBP学習が実行できるように、従来のBPアルゴリズムを修正する。最初に従来のBPアルゴリズムについて説明した後、次に修正BPアルゴリズムを説明する。

4 . 1 . 1 従来のBPアルゴリズム

ある入力に対し、式(1)～(3)の順方向の計算を行った後、学習過程では、第1番目に出力層ユニットの誤差 $\delta_k^{(o)}$ を、出力層ユニットの実際の出力 $v_k^{(o)}$ と教師データ t_k から式(4)に従って計算する。

$$\delta_k^{(o)} = \eta (t_k - v_k^{(o)}) f' (v_k^{(o)}) \quad (4)$$

ここで η は学習定数と呼ばれるパラメータである。

第2番目に中間層ユニットの誤差 $\delta_j^{(h)}$ を式(5)に従って計算する。

$$\delta_j^{(h)} = (\sum_k \delta_k^{(o)} b_{jk}) f' (v_j^{(h)}) \quad (5)$$

第3番目に結合係数の更新を行う。結合係数の更新量 Δa_{ij} 、 Δb_{jk} が式(6)、(7)によって計算され、結合係数 a_{ij} 、 b_{jk} は式(8)、(9)のように Δa_{ij} 、 Δb_{jk} だけ更新される。

$$\Delta a_{ij} (n) = \delta_j^{(h)} v_i^{(i)} + \alpha \Delta a_{ij} (n-1) \quad (6)$$

$$\Delta b_{jk} (n) = \delta_k^{(o)} v_j^{(h)} + \alpha \Delta b_{jk} (n-1) \quad (7)$$

Table 6 Recognition results (simplified model).

Input pattern	Output					
	A	B	C	D	E	F
A	<u>0.963</u>	0.000	0.000	0.000	0.002	0.002
B	0.000	<u>0.923</u>	0.000	0.016	0.002	0.000
C	0.000	0.000	<u>0.924</u>	0.000	0.013	0.000
D	0.000	0.002	0.000	<u>0.945</u>	0.000	0.000
E	0.000	0.020	0.000	0.000	<u>0.960</u>	0.000
F	0.006	0.000	0.000	0.000	0.015	<u>0.975</u>

$$a_{ij}(n) = a_{ij}(n-1) + \Delta a_{ij}(n) \quad (8)$$

$$b_{jk}(n) = b_{jk}(n-1) + \Delta b_{jk}(n) \quad (9)$$

式(6), (7)の右辺第二項は学習時の振動を減らし, 学習の収束を加速するための項であり, 慣性項と呼ばれる。 α は慣性係数である。

学習手順は以下になる。

1. 結合係数に乱数を用いて初期値を設定する。
2. ある入力パターンを提示し, 出力を計算する。
3. 各層での誤差を計算し, 結合係数を修正する。
4. 2, 3を全てのパターンについて繰り返す。
5. 学習が収束したか否かを, ある誤差評価関数を用いて全てのパターンについて判定する。
6. 2~5を一回の学習とし, 学習が収束するまで繰り返す。

以上が従来のBPアルゴリズムである。

4. 1. 2 修正BPアルゴリズム

修正BPアルゴリズムでは, 式(4)~(9)の演算を簡略型モデルで実行するための改良を行う。まず, 式(4)~(7)における乗算をシフト演算に置き換えるために, 各パラメータを以下のように, 2のべき乗表現に置き換える。

1. $\eta f'(v_k^{(o)})$ を2のべき乗の定数とする。(式(4))
 2. $f'(v_j^{(h)})$ を2のべき乗の定数とする。(式(5))
 3. α を2のべき乗の定数とする。(式(6), (7))
 4. $\delta_k^{(o)}$ を2のべき乗表現に置き換える。(式(7))
- ただし $\delta_k^{(o)}$ は, 式(4)で算出した結果を2のべき乗表現に置き換えて, 式(7)で用いることになる。

また式(6)の右辺第一項の乗算は, 入力 $v_i^{(o)}$ を, 0, 1に限定しているため省略できる。

結合係数の更新を行う式(8), (9)の演算では, 右辺第二項の Δa_{ij} , Δb_{jk} が2のべき乗ではないので, そのまま右辺第一項, 第二項を加算した場合, 演算結果が2のべき乗にはならない。そこで, 次の様に式(8), (9)の演算を修正する。

$$\Delta \Delta a_{ij}(n) = \Delta a_{ij}(n) + \Delta \Delta a_{ij}(n-1) \quad (10)$$

$$\Delta \Delta b_{jk}(n) = \Delta b_{jk}(n) + \Delta \Delta b_{jk}(n-1) \quad (11)$$

式(6), (7)において算出した毎回の更新量 Δa_{ij} ,

Δb_{jk} を式(10), (11)により積算する。積算する過程で毎回, 式(10), (11)の積算量の絶対値と結合係数の絶対値を比較し, 積算量の絶対値が大きいならば, 結合係数を修正する(2のべき乗表現)。結合係数修正後は積算量を一旦ゼロに設定し, 再度積算を繰り返す。以上のような方法を繰り返すことにより結合係数を修正し, 学習を行う。

4. 2 結果

4.1.2において説明した修正BPアルゴリズムの有効性を検証するため, 従来からしばしば取り上げられている EXCLUSIVE OR の学習シミュレーションを行った。EXCLUSIVE OR の学習には, 結合係数のビット精度がかなり必要といわれており, 簡略型モデルを用いた学習で収束可能かどうかを調べた。

Fig. 5 にシミュレーション結果を示す。点線が従来のBPアルゴリズムであり, 実線が修正したBPアルゴリズムによるシミュレーション結果である。学習が収束するまでの学習回数としては, 点線が425回, 実線が2282回であり, 修正BPアルゴリズムでは, 約5倍の学習回数を必要とする。この原因としては, 結合係数の修正部分に問題が

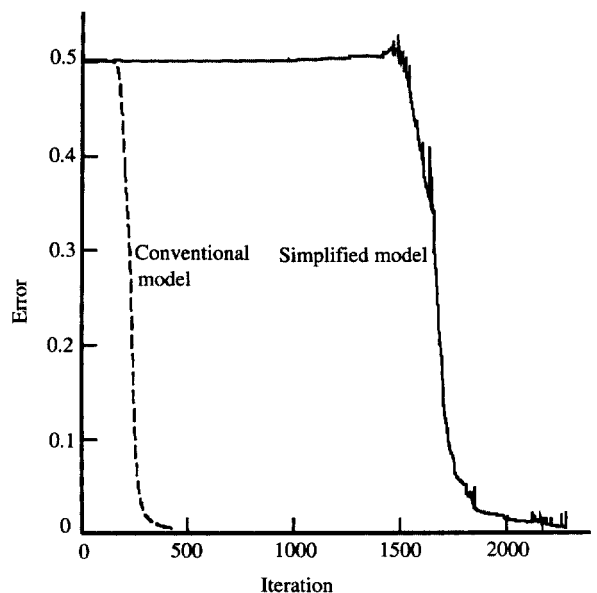


Fig. 5 Learning curves.

あると考えられる。修正が毎行われるのではなく、修正量を積算した結果を基に修正されるので、それだけ収束が遅れることになる。

以上のシミュレーションから得られた結果をまとめる。

1. 文字認識の結果は比較的良好な結果が得られた。
2. 学習アルゴリズムに関しては、改良の必要がある。

5. 簡略型ニューロンの仕様・構成・設計

学習アルゴリズムに関しては、改良の必要があることから、今回は認識機能に限定して、簡略型ニューロンを実際に設計した。

5.1 簡略型ニューロンの仕様

簡略型ニューロンの設計仕様をTable 7に示す。入出力データ、内部データはそれぞれ8ビット、16ビットとした。結合係数は仮数部、指数部それぞれ符合を含んで2ビット、4ビットとした。指数部の数値表現は2の補数表現を採用したので、+7～-8となる。これらのビット数は、先に述べた認識シミュレーションの結果を参照して、このビット数であれば十分認識できるとして、決定した値である。シナプス数は、最大64とし、ニューロン間の結合はバスを用いる構成とした。

5.2 簡略型ニューロンの構成

5.2.1 ニューロン間の結合

階層型NNを構成する場合、適用例に応じて、各層を構成するニューロン数を柔軟に変化させる必要があるため、チップ上のネットワークにおいても柔軟に可変できることが望まれる。

そこで今回の設計ではFig. 6に示すように、チ

ップ内部においては、階層型ではなく全てのニューロンを同じバスに接続し、各ニューロンが中間層、出力層、どちらのニューロンにもなり得る構成とした。ユーザが演算前に各ニューロンをプログラムすることにより、階層型と比較してより柔軟なネットワークを構築することができる。またバス結合にすることにより、ニューロン結合のためのハードウェア(主に配線)量の増加を抑えることができる。

5.2.2 シグモイド変換部

シグモイド変換部は非線形変換になるので、ROMテーブルを用いる構成とした。変換部の入出力ビット数は、ニューロンの入出力ビット数と同じ8ビットとした。各ニューロンが変換部を持っている構成と、複数ニューロンで変換部を共有する構成が考えられるが、今回の設計では、回路規模の削減を優先するために、後者の構成を採用した。

5.2.3 演算部

主にシフト演算部、加算部より構成される。シフト演算部は、任意のビットシフトを高速で実行できるように、バレルシフタを用いた。

5.2.4 演算動作

ネットワークの演算動作についてFig. 6を基に説明する。初期設定として、各ニューロンを中間

Table 7 Specifications of the simplified neuron.

Input resolution	8 bits
Output resolution	8 bits
Internal data resolution	16 bits
Weight resolution	Exponent 4bits Mantissa 2bits
Number of synapses	64
Internal network	Bus network

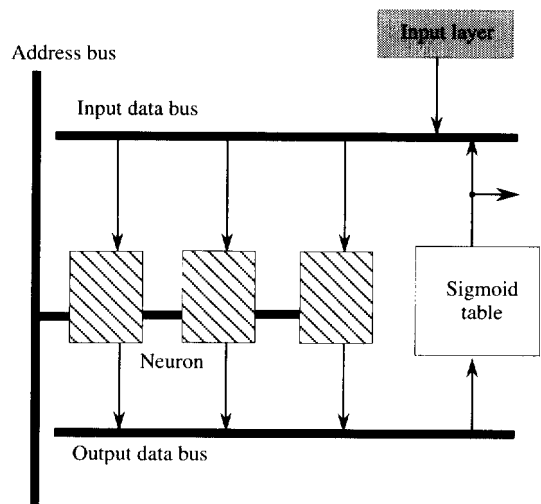


Fig. 6 Network diagram on LSI.

層，出力層にそれぞれ設定し，同時に結合係数を各ニューロンに書き込む。その後，以下のようなフローにより，演算が実行される。

1．入力層からデータが入力データバスへ出力される。

2．データ出力と同時に，データを出したニューロンのアドレスがアドレスバスに出力される。

3．各ニューロンではアドレスをデコードし，そのデコード信号に基づいて，中間層ニューロンは演算を実行する。（出力層ニューロンは演算を実行しない。）

4．次に中間層ニューロンが出力データバスへデータを出す。

5．データはシグモイド変換され入力データバスへ出力される。

6．同様に各ニューロンではアドレスをデコードし，そのデコード信号に基づいて，出力層ニューロンは演算を実行する。（中間層ニューロンは演算を実行しない。）

7．最後に出力層ニューロンからデータが出力されシグモイド変換後，外部へ出力される。

5.3 簡略型ニューロンの設計

以上の検討を基に，簡略型ニューロンを設計した。ニューロンのブロック構成をFig. 7に示す。

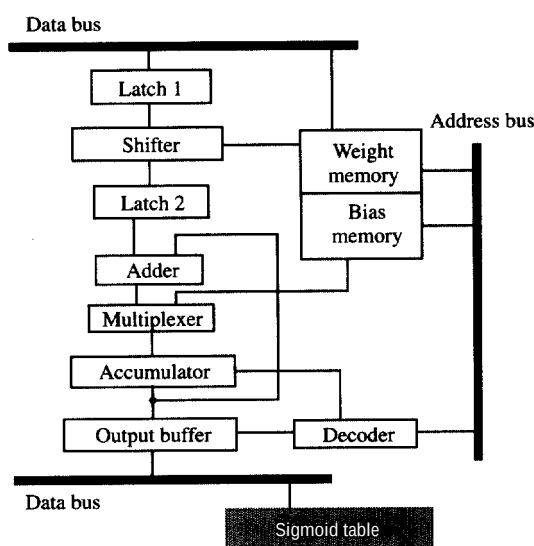


Fig. 7 Simplified neuron block diagram.

設計にはシリコンコンパイラGENESILを用いた。レイアウト設計結果をFig. 8に示す。設計ルールは $1\mu\text{m}$ CMOSルールを用いた。

簡略型ニューロンの回路規模は約6600トランジスタであり，メモリ部，演算部がほぼ同数の約3300トランジスタで構成され， $1.47 \times 1.66\text{mm}^2$ で実現できた。

簡略型モデルの特徴として，乗算器がシフタに置き換わることに，結合係数メモリも削減することができる。そこでトランジスタ数を用いて，従来モデルとの簡単な比較を行った。結果をTable 8に示す。

従来モデルの演算部トランジスタ数は，約6000トランジスタであった。また固定小数点表現（16ビット）により，結合係数メモリ部を構成した場合，約8500トランジスタであった。演算部，メモリ部を合計した従来モデルのトランジスタ数は，

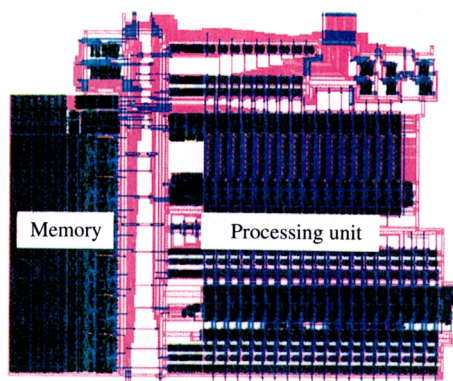


Fig. 8 Layout of the neuron.

Table 8 Comparison of number of transistors (automatic design).

	Conventional model	Simplified model
Processing unit	6000Tr	3300Tr
memory	8500Tr	3300Tr
Total	14500Tr	6600Tr

約14500トランジスタとなった。従って、簡略型モデルの回路規模は従来モデルの約1/2になった。

今回の設計は回路規模を簡単に見積もるために、自動設計システムを用いた。この結果マニュアルで設計する場合と比較して、回路規模が大きくなる。特に演算部の差が大きくなると考えられる。そこでこの影響を調べるために、演算部のみをマニュアルで設計して、回路規模を比較した。この結果をTable 9に示す。

従来モデルの演算部の回路規模が約5200トランジスタ、簡略型モデルの演算部の回路規模が約1400トランジスタという結果になった。従ってそれぞれメモリを加えた結果を比較すると、簡略型モデルの回路規模は、約4700トランジスタであり、従来モデルの約13700トランジスタと比較して、約1/3になった。

6. まとめ

本論文では、デジタルニューロLSIの集積度向上を目的として、簡略型ニューロンモデルの構成を検討した。さらに認識機能に限定して、簡略型ニューロンのレイアウトを設計し、回路規模の評価を試みた。以下に結果をまとめる。

- ・結合係数を2のべき乗で表現するモデルを提案した。

- ・簡略型モデルを用いた文字認識シミュレーションにより、認識能力が従来モデルと同等であることを確認した。

- ・学習シミュレーションの修正により、簡略型モデルでも学習できることを確認した。ただし従来法と比較して、学習が収束するまでの時間が長くなる問題がある。

- ・LSI内部の構成として、柔軟なネットワーク

を構築できる方式を考案した。

- ・ニューロンのレイアウトを設計し、ニューロン1個を約1.5mm角で実現できた。

- ・簡略型モデルの回路規模は従来モデルと比較して、全て自動設計した場合、約1/2、演算部をマニュアル設計した場合、約1/3になることがわかった。

今後の課題として、簡略型モデルによる学習時に、収束時間を短くできるようにアルゴリズムをさらに改良する必要がある。

謝 辞

ニューロンモデル検討に際して、当所システム1部機械認識研究室原田義久室長に協力して頂きました。

参 考 文 献

- 1) Ramacher, U. : "SYNAPSE-X: A General-purpose Neurocomputer Architecture", IJCNN'91, Singapore, vol.3(1991), 2168
- 2) Morishita, T., Tamura, Y., Otsuki, T. and Kano, G. : "A BiCMOS Analog Neural Network with Dynamically Updated Weights", IEICE Trans. Electron, E75-C-3(1992), 297
- 3) Arima, Y., Murasaki, M., Yamada, T., Maeda, A. and Shimohara, H. : "A Refreshable Analog VLSI Neural Network Chip with 400 Neurons and 40k Synapses", ISSCC Dig. of Tech. pap., (1992), 132
- 4) Satyanarayana, S., Tsividis, Y. P. and Graf, H. P. : "A Reconfigurable VLSI Neural Network", IEEE J. Solid State Circuits, 27-1(1992), 67
- 5) Yasunaga, M., Masuda, N., Yagyu, M., Asai, M., Yamada, M. and Masaki, A. : "Design, Fabrication and Evaluation of a 5-inch Wafer Scale Neural Network LSI Composed of 576 Digital Neurons", IJCNN'90, San Diego, (1990), 527
- 6) Griffin, M., Tahara, G., Knorpp, K., Pinkham, R. and Riley, B. : "An 11-Million Transistor Neural Network Execution Engine", ISSCC Dig. of Tech. pap., (1991), 180
- 7) Uchimura, K., Saito, O. and Amemiya, Y. : "A High-Speed Digital Neural Network Chip with Low Power Chain-Reaction Architecture", IEEE J. Solid State Circuits, 27-12(1992), 1862
- 8) Kondo, Y., Koshiba, Y., Arima, Y., Murasaki, M., Yamada, T., Amishiro, H., Shinohara, H. and Mori, H. : "A 1.2GFLOPS Neural Network Chip Exhibiting Fast Convergence", ISSCC Dig. of Tech. pap., (1994), 218

Table 9 Comparison of number of transistors.

	Conventional model	Simplified model
Processing unit memory	5200Tr	1400Tr
	8500Tr	3300Tr
Total	13700Tr	4700Tr

Processing unit : manual design
Memory : automatic design

著 者 紹 介



安藤道則 Michinori Andoh
生年：1960年。
所属：機械認識研究室。
分野：コンピュータビジョンに関する研究開発。
学会等：電子情報通信学会会員。
1995年 第26回画像工学コンファレンス優秀ポスタ賞受賞。



太田則一 Norikazu Ohta
生年：1963年。
所属：回路設計研究室。
分野：CMOSアナログ回路の設計。
学会等：電子情報通信学会会員。



渡邊章弘 Akihiro Watanabe
生年：1966年。
所属：機械認識研究室。
分野：認識アルゴリズムの研究。
学会等：情報処理学会会員。