一般化学習ネットワークにおけるファジィルールを用いたパラメータの学習方式

池内, 光雄
九州大学大学院総合理工学研究科エネルギー変換工学専攻 : 修士課程

平澤, 宏太郎
九州大学大学院システム情報科学府電気電子システム工学専攻

大林, 正直
九州大学大学院システム情報科学府電気電子システム工学専攻

胡, 敬炉
九州大学大学院システム情報科学府電気電子システム工学専攻

他

https://doi.org/10.15017/1498361

出版情報：九州大学大学院システム情報科学紀要. 3 (2), pp.225-231, 1998-06-15. 九州大学大学院システム情報科学研究院
バージョン：

九州大学大学院
システム情報科学研究科報告
第3巻　第2号　平成10年9月

Research Reports on Information Science and
Electrical Engineering of Kyushu University
Vol.3, No.2, September 1998

一般化学習ネットワークにおける
ファジィルールを用いたパラメータの学習方式

池内　光雄*・平澤　宏太郎**・大林　正直**・胡　敬炉**・村田　純一**

Learning Method of Parameters for Fuzzy Rules in Universal Learning Network

Mitsuo IKEUCHI, Kotaro HIRASAWA, Masanao OHBAYASHI, Jinglu HU, Junichi MURATA

(Received June 15, 1998)

Abstract: In this paper, a new method which can alter the values of the parameters in neural networks is proposed in order to enhance the representation abilities of the networks. As an example, a fuzzy reference network is used to modify the parameters in this article, even though any kind of networks such as radial basis function networks and neural networks can be adopted to realize varying parameters. From simulations, it is shown that the network using the proposed method is better than the conventional neural networks in terms of representation abilities of the networks.

Keywords: Neural network, Learning, Fuzzy rule

1. はじめに

我々の研究室で研究されている一般化学習ネットワーク(Universal Learning Network; ULN)は，従来のニューラルネットワーク[1]に比べて，様々な一般化・拡張がなされている．一般化学習ネットワークが開発された当初はこれを制御に用いることを主体として考えており，一般化学習ネットワークで制御対象とコントローラを同時に記述することや，高次の微分を行うことで制御系の安定性を上げることなどの研究がなされていた．

しかし，最近では，制御以外の分野にも一般化学習ネットワークを応用しようという試みがなされている．また，制御の分野においても，できるだけ高性能なネットワークを作るべく研究を進めている．

制御以外に一般化学習ネットワークを応用する分野の一例としては，相互に作用している複数の生物種の生態のモデル化が挙げられる．ロトカ・ボルテラ方程式などを用いたモデル化がその一例である．ロトカ・ボルテラ方程式では，各変数が各生物種の個体数に相当し，それらが互いに「食う・食われる」という関係にあるとき，それらの間に相互作用の定数が存在している．これを一般化学習ネットワークで表現しようとすると，一つのノードが一つの未知数，つまり一つの生物種の個体数になる．そして，ノードを結ぶブランチの重み係数が，二つの生物種間の相互作用の定数になる．この定数が，生物種の個体数の変化に伴って変化する方が，より現実世界の生物種の生態をよく表現している，という研究もなされている[2]．

平成10年6月15日受付
*　総合理工学研究科エネルギー変換工学専攻修士課程
**　電気電子システム工学専攻

これを一般化学習ネットワークで表現すれば，ブランチの重み係数をブランチを通した入力に依存して変化させるような機構が必要になる．

そのような機構の一つとして，ファジィルールを用いたものを本論文では検討する．通常，ニューラルネットワークのブランチの重み係数は，学習によってその値を決定するが，本論文で考察するシステムにおいては，重み自体はそのブランチへの入力により一意に決まるものとし，入力から重みの値を算出する過程に関係するパラメータを，学習により決定するものとする．

このような機構を導入した場合のネットワークの性質や性能について，シミュレーションを行って検討する．本来ならば，システムの中での時間を考慮した，ダイナミックシステムでのシミュレーションを行うべきかもしれないが，基本的な性質を調べたいという目的から，スタティックシステムでのシミュレーションを行った．

2. 一般化学習ネットワーク

一般化学習ネットワークとは，通常のニューラルネットワークを様々な面で拡張したもので，多重ブランチ，任意の遅れ時間などを許したネットワークである[3]．一般化学習ネットワークに関しては，多階微分計算理論や安定性理論，ロバスト性を高めるコントローラの構成など，多数の論文が発表されている[4),5),6]．しかし，本論文ではそのような理論は用いないので，詳述はしない．

制御の分野で用いる場合，一般化学習ネットワークの最大の特徴は，制御対象とコントローラを同じ枠組で記述できることである．従って，制御対象とコントローラを含めた一つの大きなシステムに対して安定性などを考慮すればよいことになる．

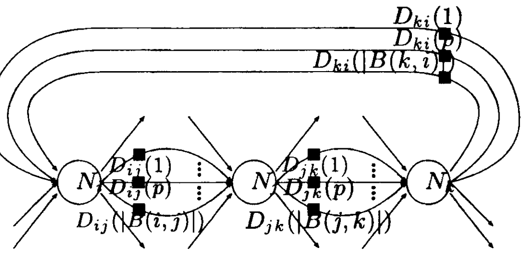


Fig.1 Structure of universal learning network

一般化学習ネットワークの基本式((1)式) 及び構成(**Fig.1**)を示す.

$$h_j(t) = f_j(\{h_i(t\text{–}D_{ij}(p))|i \in JF(j), p \in B(i,j)\},\ \{r_n(t)|n \in N(j)\},\ \{\lambda_m(t)|m \in M(j)\}) \quad (1)$$

$$j \in J, t \in T$$

$h_j(t)$	:	jノードのt時刻の出力値
$r_n(t)$	:	n外部入力変数のt時刻の値
$\lambda_m(t)$	:	mパラメータのt時刻の値
f_j	:	jノードの非線形関数
$D_{ij}(p)$	:	iノードからjノードへのp番目のブランチの遅れ時間
J	:	ノード番号の集合
$JF(j)$	:	jノードへ接続するノード番号の集合
N	:	外部入力変数番号の集合
M	:	パラメータ番号の集合
$M(j)$	:	jノードへ接続するパラメータ番号の集合
$B(i,j)$	:	iノードからjノードへ接続するブランチ番号の集合
T	:	時刻の集合

通常のニューラルネットワークにおいては，ノードが持つ非線形関数として，シグモイド関数$f_j(\alpha_j(t))$を与え((2)式)，その重み$w_{ij}(p)$を学習する.

$$h_j(t) = f_j(\alpha_j(t)) = \frac{1 - e^{-\phi_j \alpha_j(t)}}{1 + e^{-\phi_j \alpha_j(t)}} \quad (2)$$

$$\alpha_j(t) = \sum_{i \in JF(j)} \sum_{p \in B(i,j)} w_{ij}(p) h_i(t\text{–}D_{ij}(p)) + \theta_j \quad (3)$$

$w_{ij}(p)$	:	iノードからjノードへのp番目のブランチの重みパラメータ
θ_j	:	jノードの閾値パラメータ
ϕ_j	:	jノードの勾配パラメータ

一般化学習ネットワークを制御に用いる場合には，ノード関数として，シグモイド関数だけでなく制御対象のシステムを表す式を用いる場合もある．もし，制御対象が微分方程式で表されているならば，それを差分方程式に直してノード関数とする．しかし，本論文では一般化学習ネットワークを制御には用いないので，ノード関数としてはシグモイド関数，あるいは，単に入力の和を出力する関数が使われる.

3. 一般化学習ネットワークへのファジィルールの導入

通常，シグモイド関数における重みは定数であるので，あるノードの出力は定数倍されて次のノードの関数の中で用いられる．しかし，ノード間にもっと柔軟な性質を持たせようとするならば，重みも前段のノード出力に依存して変化するようにする，という方法が考えられる.

3.1 ファジィルールを用いた関数

重みが入力に対して変化する機構には，ファジィルール[7)]を用いている．変化のおおまかな機構は次のようなものである．まず，「入力が $h_i = A_1$ ならば重みは $\lambda_i = B_1$ にする」，「入力が $h_i = A_2$ ならば重みは $\lambda_i = B_2$ にする」，というように，いくつかのルール(ファジィルール)を定めておく．そして，実際の入力 $h_i = h_i^*$ があると，それぞれのファジィルールへの適合度を計算し，それらを統合(非ファジィ化)して，重みの値を決定する．それらの計算には，ファジィ論理が必要となる.

ファジィ論理においては，通常の論理学において，「真」あるいは「偽」が決定できるような命題に対して「メンバーシップ関数」を定め，その関数で計算される値(それは$[0,1]$の値である)により「どのていど『真』なのか」を表す．例えば，$h_i = A_1$ のメンバーシップ関数 $\mu[A_1](h_i)$ は，h_i がA_1 に近いほど 1 に近付き，h_iがA_1 から離れるほど 0 に近付く．そして，その値が 1 に近いほど，通常の論理学での「真」の度合が高いことになる.

ファジィ論理においても，和，積などの論理演算が定義できる．その定義は唯一ではなく，いくつかの定義がなされているが，本論文では，計算機でシミュレーションを行う関係上，代数積(通常の乗算)を用いた.

また，「A ならば B」という命題は「含意」と呼ばれ，通常の論理学では$A \to B$ と表し，$\sim A \vee B$ (A でないか，あるいは B) と等しいとされている．ファジィ論理においても同様の拡張を行うことができるが，必ずしも$A \to B$ を $\sim A \vee B$ とする必要はない．本論文では，$A \to B$ のメンバーシップ関数を，単純に命題 A のメンバーシップ関数と命題 B のメンバーシップ関数の代数積で定義する.

3.2 各ファジィルールとメンバーシップ関数

ファジィルールとしては，以下のような形のものを考える.

$$\text{if } \ h_i = A_1 \ \text{ then } \ \lambda_i = B_1 \quad (4)$$

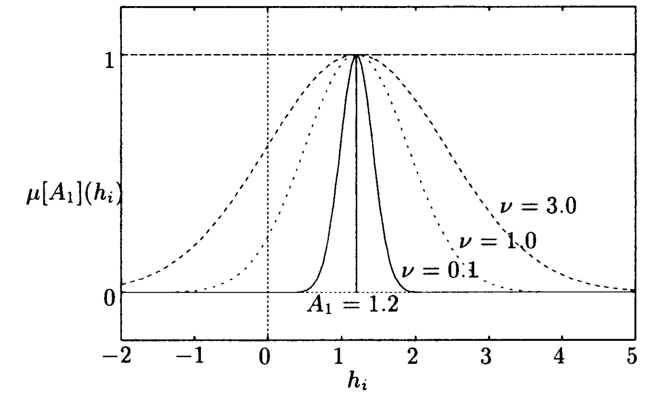


Fig.2 Membership function for $h_i = A_1$

これを日本語にすると,「もし入力 h_i がほぼ A_1 であるならば,重み λ_i はほぼ B_1 になる」となるだろう.ここで,if の節を「前件部」と呼び,then の節を「後件部」と呼ぶ.前件部,後件部の式には,それぞれメンバーシップ関数が付随している.本論文では,(5)式のような形のメンバーシップ関数を用いた.

$$\mu[A_1](h_i) = \exp\left\{-\frac{(h_i - A_1)^2}{\nu}\right\} \tag{5}$$

以下,(5)式中の A_1 を「平均」,νを「擬分散」と呼ぶ.νは,確率の分野での分散に似ているが,メンバーシップ関数においては,確率密度関数と異なり,その面積が 1 にならなければいけないという制限はない.メンバーシップ関数の最大値が 1 であるという条件の方が妥当であるので,(5)式のような形にした.その場合,ν が確率の分野での分散とは異なるので,「擬分散」という呼称を用いる.

(5)式の概形を**Fig.2**に示す.ここで,平均 $A_1 = 1.2$ とし,擬分散 $\nu = 0.1, 1.0, 3.0$ の場合を描いた.後件部に対しても,同様の形式を持つメンバーシップ関数を用いる.

前件部も後件部もメンバーシップ関数を持つが,このファジィルール自体も(真・偽のどちらかを持つ代わりに)メンバーシップ関数を持つ.これは,3.1でも述べたが,前件部のメンバーシップ関数と後件部のメンバーシップ関数の代数積で与える.

3.3 非ファジィ化

いくつかのファジィルールを持つ重みが付随したブランチに,入力 h_i^* が入ると,各ルールと入力から,メンバーシップ関数が導かれる.ルールと同じ個数のメンバーシップ関数を統合して,一つの値にすることを「非ファジィ化」という.非ファジィ化には,主として二つの方法があり,それぞれ,重心法,面積法と呼ばれている[8].本研究では,計算が簡単になる面積法を用いている.

4. シミュレーションで用いたネットワーク

シミュレーションでは,

- ファジィルールを用いたニューラルネットワーク
- 中間層が一層のシグモイド関数のニューラルネットワーク
- 中間層が二層のシグモイド関数のニューラルネットワーク

の三種類の比較を行った.比較は,まず教師関数を決定し,教師関数の関数関係を対象となるネットワークの入出力関係に学習させ,その結果により行った.教師関数を $f(x)$とすると,学習の際の評価関数は,(6)式のようになる.

$$E = \frac{1}{M}\sum_{i=1}^{M}(y(x_i) - f(x_i))^2 \tag{6}$$
$$i = 1, 2, \cdots, M$$

$y(x)$: ネットワークの出力
$f(x)$: 教師関数
M : 学習により覚えさせたい点の数

4.1 ファジィルールを用いたニューラルネットワーク

ファジィルールを用いたニューラルネットワークは,中間層が一層で,中間層と出力ノードをつなぐブランチの重みをファジィルールで決定している(**Fig.3**).一入力一出力で,出力は(7)式 ~ (10)式に従って計算される.式中の記号は,**Fig.3**中の記号と対応している.また,式中の$\sigma(a)$は,シグモイド関数であり,(11)式で計算される.ファジィルールが用いられているのは,中間層と出力ノードをつなぐブランチにおける重みであり,(10)式の$\lambda_{2i}(h_{1i})$ $(i = 1, 2, \cdots, N)$に相当する.このそれぞれの重みがQ個のファジィルールを持っており,それに従って,h_{1i}から$\lambda_{2i}(h_{1i})$が計算される.

$$h_0 = x \tag{7}$$
$$h_{1i} = \sigma(a_{1i}) \tag{8}$$
$$a_{1i} = \lambda_{1i} h_0 + \theta_{1i} \tag{9}$$
$$i = 1, 2, \cdots, N$$
$$h_2 = \sum_{i=1}^{N} \lambda_{2i}(h_{1i}) h_{1i} = y \tag{10}$$

$$\sigma(a) = \frac{1 - e^{-a}}{1 + e^{-a}} \tag{11}$$

注意すべきは,ファジィルールを用いた重みに関しては,学習パラメータがλではなく,Bの記号(ファジィルー

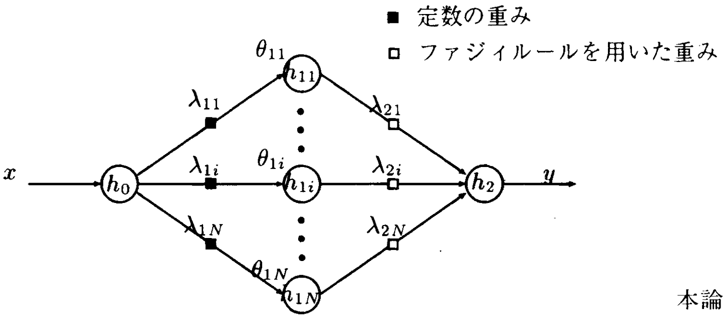


Fig.3　Neural network with fuzzy rules

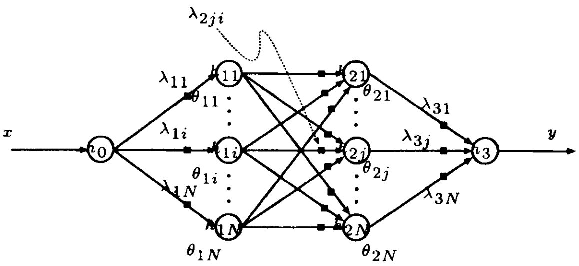


Fig.4　Neural network with two hidden layers for sigmoid function

Table-1　The number of learning parameters included by comparative networks

ファジィルールを用いたニューラルネットワーク	$N(2+Q)$
中間層が一層のシグモイド関数のニューラルネットワーク	N^2+4N

ルの後件部平均)で表されている，ということである．

4.2　中間層が二層のシグモイド関数のニューラルネットワーク

中間層が二層のシグモイド関数のニューラルネットワークの構成を**Fig.4**に示す．図中の重みは全て定数である．ネットワークの計算式は省略する．中間層のノード関数には(11)式のシグモイド関数を用い，出力ノードは線形和である．

4.3　各ネットワークの学習パラメータ数

ここで，上記三種類のネットワークの学習パラメータ数を**Table–1**に記しておく．

5.　シミュレーションに用いた学習（探索）法

ニューラルネットワークでいう学習とは，ある評価関数を用意し，その関数ができるだけ小さくなるような(あるいは大きくなるような)挙動をネットワークが示すように，ネットワーク内のパラメータを決定する，ということに他ならない．

Table-2 Simulation condition

学習回数	100000
学習係数の初期値	0.01
モーメント法の定数μ	0.9
適応的学習係数法の定数η_1	1.05
適応的学習係数法の定数η_0	0.7
教師関数	(**Fig.5**)

本論文で用いたシミュレーションには，勾配法を用いた．本節では，この探索法について，簡単に述べる．

勾配法は，評価関数を学習パラメータで微分し，求めた微係数を元に学習パラメータを変更する，という手法である((12)式).

$$\lambda \leftarrow \lambda - \gamma \frac{\partial^{\dagger} E}{\partial \lambda} \tag{12}$$

E　:　評価関数
λ　:　学習パラメータ
γ　:　学習係数

ただし，本論文のシミュレーションでは，単純な勾配法よりも学習効率の良い，「モーメント法」と「適応的学習係数法」を併用した．

実際にシミュレーションを行うにあたっては，ランダムサーチのシミュレーションも行ったが，勾配法で良い結果を得ることができたので，本論文に掲載している結果は，全て勾配法で学習したものである．

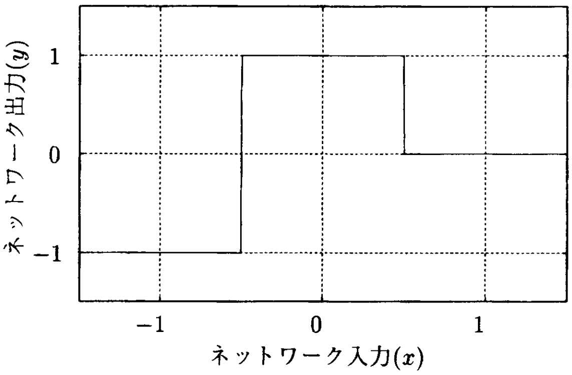
6.　シミュレーション

6.1　ノード数とルール数の組み合わせによるネットワークの性能の違い

本論文のメインとなるシミュレーションとして，学習パラメータの数を等しくした場合に，4.で述べた2つのネットワークを比較する，というシミュレーションを行うが，それに先立ち，ファジィルールを用いたネットワークのみで，その特性を明らかにするためのシミュレーションを行う．具体的には，「学習パラメータの数を一定としたときに，『ノード数』と『ルール数』の兼ね合いをどのようにすれば，ネットワークが最もよく学習するか」という問題である．

4.で述べたシミュレーションに使用したネットワークの中のファジィルールを用いたネットワークは，中間層のノード数をN，各重み毎のファジィルールの数をQとすると，学習パラメータの数は$N(Q+2)$個となる．学習パラメータの個数を24個とすると，NとQの可能な組み合わせは$(N,Q)=(1,22),(2,10),(3,6),(4,4),(6,2),(8,1)$の6通りになる．

学習曲線を**Fig.6**に示す．(a-1)~(a-6)は，中間層のノード数と重み毎のルール数によって分けて示した学習



[-1,1]の範囲を50等分した点を学習対象の点とした．従って，学習した点の数は51個である．

Fig.5 Function to be realized

曲線である．それらの各々の中で最も良い評価を得たものをまとめて(b)に示している．

これらの結果より，ノード数とルール数との間には，ネットワークの能力が最も良くなる関係があり，そのとき，ノード数とルール数は，どちらも極端に大きかったり小さかったりはしないことが分かる．

6.2 学習曲線からの考察

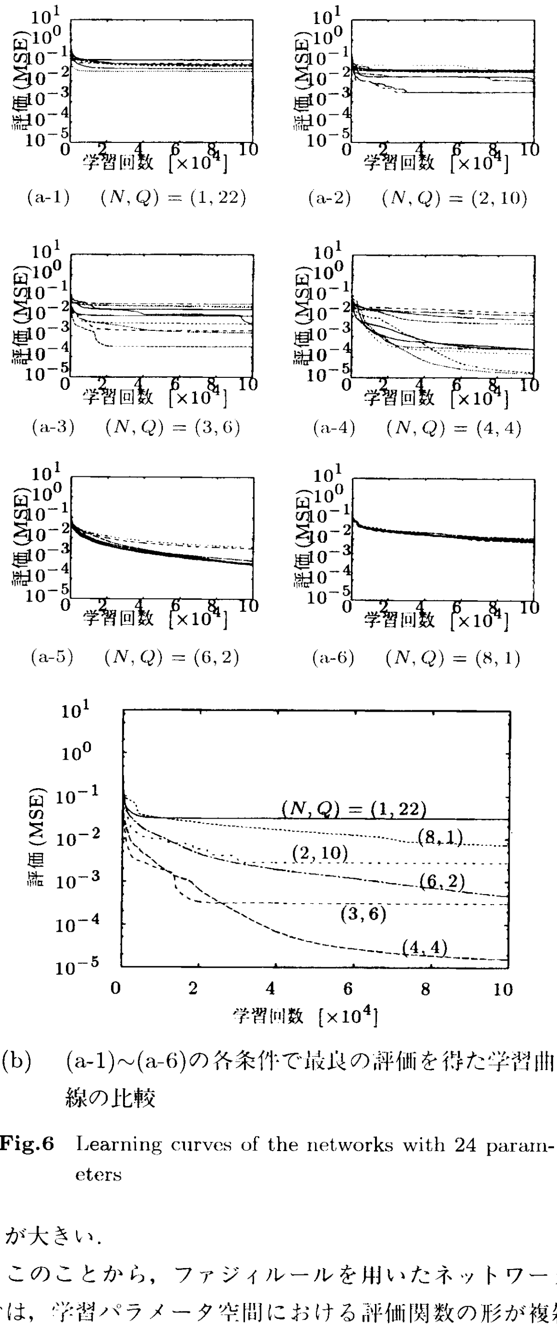
本項では，ファジィルールを用いたニューラルネットワークと，シグモイド関数を用いた通常のニューラルネットワークとで，得られた学習曲線を比較した結果について述べる．

比較の基準としては，学習パラメータの個数が等しいという条件を用いた．学習パラメータが45個の場合に，2つのニューラルネットワークとも端数なくネットワークを構成できるので，その学習パラメータ数でシミュレーションを行った．

ファジィルールを用いたニューラルネットワークでは，学習パラメータの個数が45個の場合，ノード数とルール数の組み合わせが，$(N,Q)=(1,43),(3,13),(5,7),(15,1)$の4通り存在する(**Table–1**参照)．しかし，前述のように，これらの組み合わせの中の適切なものを選べばよいので，全ての組み合わせでシミュレーションを行う必要はない．実際には，ノード数かルール数が一つの場合を除外して，$(N,Q)=(3,13),(5,7)$の二通りでシミュレーションを行った．

シミュレーション1と同様に，同じ条件で10通りの乱数系列を用いてシミュレーションを行った．

学習パラメータが45個で勾配法を用いて学習した場合の，学習曲線を**Fig.7**に示す．この図を見ると(縦軸のスケールと範囲は揃えてある)，図(a)のファジィルールを用いたニューラルネットワークでは，図(b)に比べて学習曲線に滑らかでない部分が見られる．また，明らかにばらつ



(b) (a-1)〜(a-6)の各条件で最良の評価を得た学習曲線の比較

Fig.6 Learning curves of the networks with 24 parameters

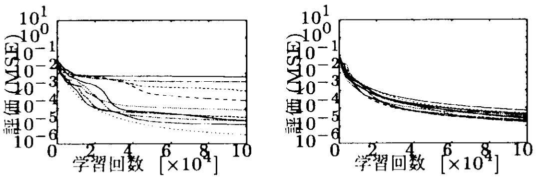
きが大きい．

このことから，ファジィルールを用いたネットワークでは，学習パラメータ空間における評価関数の形が複雑で，ローカルミニマムに陥りやすいといえる．また，10通りの乱数系列の中で評価値が最小のものは，ファジィルールを用いたネットワークの方が小さい．このことは，ファジィルールを用いたネットワークの方が高い写像能力を持つことを示している．

6.3 ファジィルールを使わないネットワークとの比較

そこで，更に複雑な教師関数を用いて，シミュレーションは，学習パラメータ数が12個と45個の二通りの場合で行った．教師関数としては，**Fig.8**に示す関数をを用いた．

Fig.9を見ると，明らかにシグモイド関数の二層のネットワークは，教師関数の概形を学習できていない．これ



(a)　ファジィルールを用いたネットワーク ((N,Q)=(5,7))

(b)　中間層が二層のシグモイド関数のニューラルネットワーク

Fig.7　Comparison with neural networks with varying parameters and conventional neural networks

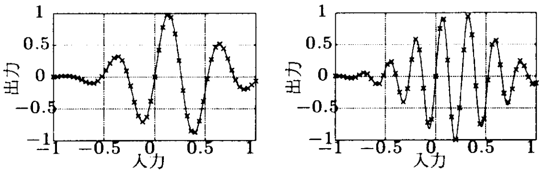
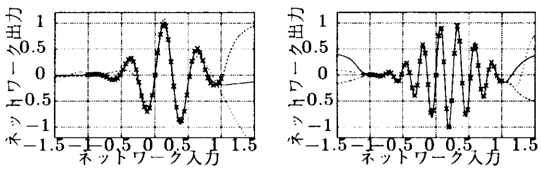
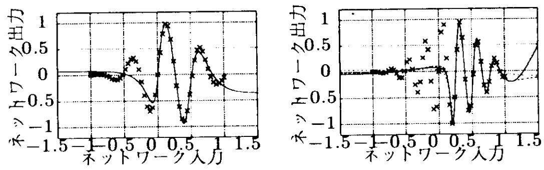


Fig.8　Complex functions to be realized



(a)　ファジィルールを用いたネットワーク(学習パラメータ12個)

(b)　ファジィルールを用いたネットワーク(学習パラメータ45個)



(c)　シグモイド関数を用いた二層のネットワーク(学習パラメータ12個)

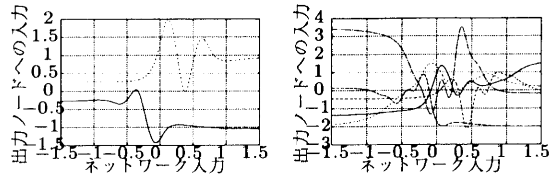
(d)　シグモイド関数を用いた二層のネットワーク(学習パラメータ45個)

図中で示している点は学習した点である.

Fig.9　Nonlinear functions produced by learning

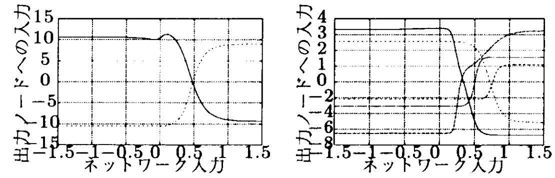
は，出力ノードで足し合わされるシグモイド関数の個数が，教師関数の起伏を表現できるのに必要な個数より小さいためと思われる．これに対してファジィルールを用いたネットワークは教師関数の概形をほぼ学習している．その大きな原因としては，中間層から出力ノードにつながるブランチのファジィルールによって，その重みが適切に変化するように学習がなされたためであることが考えられる．

シグモイド二層のネットワークは，出力ノードへの入力が，シグモイド関数のような形をしていて，それらの組み合わせで教師関数を表現しようとしているので，ノード数が足りずにいくつかの起伏を表現できないようなネットワークになってしまっている．ファジィルールを



(a)　ファジィルールを用いたネットワーク(学習パラメータ12個)

(b)　ファジィルールを用いたネットワーク(学習パラメータ45個)



(c)　シグモイド関数を用いた二層のネットワーク(学習パラメータ12個)

(d)　シグモイド関数を用いた二層のネットワーク(学習パラメータ45個)

「出力ノードへの入力」の和が，出力ノードの出力(すなわち，ネットワーク出力)になる．図中で示している点は学習した点である．

Fig.10　Inputs to output node

用いたネットワークは，学習パラメータが12個の場合は，出力ノードへの二つの入力が，うまく教師関数の山の部分を分担し合って学習していることが分かる(**Fig.10**参照)．学習パラメータが45個の場合は，幾分冗長な部分(互いに上に凸と下に凸の部分がキャンセルし合っている部分)が見られるものの，結果的に教師関数の概形を学習している．

ファジィルールの特長としては，「人間にとって直観的に分かりやすい」というものがあり，一般に解析しにくい複雑なネットワークの挙動を見るために有用ではないか，という期待もあってファジィルールを導入したという経緯があった．その点を鑑みるに，解析しにくいような結果が出たのは，一考を要する問題である．これを解決するには，例えば，一つの重みを決定するために使われるファジィルール同士がなるべく大きな違いを持たないようにする，などの新しい評価指標を導入する，という方法も考えられる．こうすれば，ファジィルールを持った重みに余計な起伏が生ぜずに，解析しやすいようにネットワークが学習する可能性がある．

次に，学習範囲外でのネットワークの挙動について考察する．

学習は，ネットワーク入力が$[-1,1]$の範囲で行われているのだが，学習をしていない範囲では，ファジィルールを用いたネットワークもシグモイド関数を用いた二層のネットワークも，ネットワーク出力は平坦ではないが，それらの要素である出力ノードの入力は，比較的平坦である．その根拠は定かではないが，一つ考えられることとしては，たまたま今回のシミュレーションで用いた教師関数が，学習範囲の外側に近付くにつれて0に近付くよう

なものだったため，ということがある．しかし，全ての教師関数がそのような条件を満たしているので，完全に原因が分かったわけではない．しかし，もしどんな教師関数であろうとも学習範囲以外ではネットワーク出力が平坦になるような原因があるとすれば，それを知ることによって学習していない範囲での安定性を改善できるかもしれない．

直接今回のシミュレーションとの関係はない話になるかもしれないが，学習の対象になっている範囲とそうでない範囲との境界におけるネットワーク出力の傾きも，一つの安定性の指標になりうるだろう．学習範囲である程度の学習がなされていて，ネットワークが目的に適うような反応をしていたとしても，学習範囲とそれ以外の範囲との境界に近付くにつれて誤差が大きくなる，という場合も考えられるからである．

7. む す び

本論文では，これまでのネットワークで定数として扱われていたパラメータ(重みパラメータ)をファジィルールにより可変とすることにより，ネットワークの関数写像能力が増すことを示した．問題は，写像能力が高いために評価関数の形状が複雑になり，過学習が起こりやすいという点である．しかし、過学習の原因がファジィルールを用いた関数の起伏の多さであるので、過学習が起こらないような評価関数の設定は比較的容易であると考えている．

論文中では，シミュレーションにスタティックシステムを用いたため，関数写像能力についてしか論じることができなかったが，今後はこの機構のダイナミックシステムへの適用を行う予定である．

参 考 文 献

1) 西川，北村: システム制御情報ライブラリー ニューラルネットと計測制御，朝倉書店 (1995)
2) 巌佐 庸: シリーズ・ニューバイオフィジックス 数理生態学，共立出版 (1997)
3) 平澤，大林，藤田，古賀: 一般化学習ネットワーク理論，電気学会論文誌 C,116 巻，7 号 (1996)
4) 平澤，大林，古賀: フォワードプロパゲーション一般化学習ネットワーク理論，電気学会論文誌 C,116 巻,6 号 (1996)
5) 平澤，大林，古賀，村田，楠見: 一般化学習ネットワークの高次微分の計算理論，電気学会論文誌 C,115 巻,12 号 (1995)
6) 平澤，大林，古賀: 一般化学習ネットワークの安定性理論，電気学会論文誌 C,116 巻,8 号 (1996)
7) 坂和 正敏，石井 博昭，西崎 一郎: ソフトコンピューティングシリーズ ソフト最適化，朝倉書店 (1995)
8) 広田 薫:知能工学シリーズ 1 知能工学概論，昭晃堂 (1996)
9) 東郷 和幸，平澤 宏太郎，古賀 勝，大林 正直: 集中化・多様化を用いたランダム探索法，第 35 回計測自動制御学会学術講演会 予稿集 pp.237-238 (1996)