The University of Osaka
Institutional Knowledge Archive

Title	多層ニューラルネットの産業応用に関する研究
Author(s)	鹿山, 昌宏
Citation	大阪大学, 1995, 博士論文
Version Type	VoR
URL	https://doi.org/10.11501/3106812
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

The University of Osaka

多層ニューラルネットの産業応用に関する研究

鹿　山　昌　宏

①

多層ニューラルネットの産業応用に関する研究

平成７年５月

鹿　山　昌　宏

目　次

第1章　緒　　論 …… 1
第1.1節　現行の産業システムと制御・認識への課題 …… 1
第1.2節　ニューラルネットへの期待 …… 6
第1.3節　ニューラルネット適用上の問題点 …… 9
第1.4節　研究の目的と概要 …… 12
第1.5節　本論文の構成 …… 14
参考文献 …… 16

第2章　ニューラルネットの最適中間層決定方法 …… 17
第2.1節　研究の概要 …… 17
第2.2節　ニューラルネットの構成と学習方法 …… 19
第2.3節　中間層ニューロン数決定方法 …… 25
第2.4節　実験結果および考察 …… 33
第2.5節　まとめ …… 41
参考文献 …… 42

第3章　汎化能力向上を目的としたニューラルネットの学習方式 …… 43
第3.1節　研究の概要 …… 43
第3.2節　学習方法 …… 46
第3.3節　実験結果および考察 …… 55
第3.4節　まとめ …… 70
参考文献 …… 71

第4章　クラスタリング問題における汎化能力の中間層ニューロン数依存性 …… 72
第4.1節　研究の概要 …… 72
第4.2節　中間層ニューロン数と汎化能力 …… 74

第4.3節　最適中間層ニューロン数の決定方法 …… 80
第4.4節　実験結果および考察 …… 82
第4.5節　まとめ …… 87
参考文献 …… 88

第5章　制御モデルのチューニングを高速・高精度に行う
　　アジャスティングニューラルネットの構成と学習方式 …… 89
第5.1節　研究の概要 …… 89
第5.2節　アジャスティングニューラルネットの構成 …… 91
第5.3節　実験結果および考察 …… 106
第5.4節　まとめ …… 112
参考文献 …… 113

第6章　ニューラルネットを搭載した汎用コントローラの高速制御方式 …… 114
第6.1節　研究の概要 …… 114
第6.2節　シーケンス処理の高速演算法 …… 116
第6.3節　プログラム実行方式とトークン管理機能の拡張方法 …… 126
第6.4節　実験結果および考察 …… 130
第6.5節　まとめ …… 136
参考文献 …… 137

第7章　開発技術の実システムへの適用 …… 138
第7.1節　研究の概要 …… 138
第7.2節　車番認識システム …… 140
第7.3節　加熱炉制御システム …… 146
第7.4節　ニューラルネット構築支援システムの構成 …… 155
第7.5節　まとめ …… 162
参考文献 …… 163

第8章　結　　論 …… 164
関連論文 …… 167
謝　　辞 …… 169

第1章　緒　論

第1章　緒　　論

第1.1節　現行の産業システムと制御・認識への課題

鉄鋼，公共，電力等に代表される産業システムでは，近年，システムの複雑化が著しい一方，この中で要求される制御や認識の性能は一段と高度なものになっている。このため制御・認識部を効率的に構築，保守することを目的とした自動化技術や支援技術の重要性が高まっている。本章では，制御・認識部の構築，保守を困難にしている要因を，

（1）要求性能を得るための制御・認識アルゴリズムの複雑化

（2）制御・認識が行なわれる対象そのものの複雑化

に大別して具体的に説明した後，これらを解決する有力なアプローチとしてニューラルネットの適用可能性を議論する。

（1）は要求性能の高まりに同期して，これを実現するための制御・認識アルゴリズムが著しく複雑化していることを意味している。必然的にこれらを作り上げたりメンテナンスするためには，多大な作業量が必要となる。とりわけシステムに対応して，個別に制御・認識アルゴリズムの作り込みを行なう用途では，この問題はソフトウェア生産性と直接関連するため，深刻である。

本論文の具体的な検討対象の一つである車番認識システム[1]を例に，この要因についてもう少し詳しく説明する。図1.1は車番認識システムが行う処理の概要である。車番認識システムは，ＴＶカメラから得た画像情報を用いて車の通過を検知し，この車からナンバプレートを抽出する。そして適当な濃淡処理を施した後，数字や文字を切り出し，これらが何であるかを認識する。一般に認識アルゴリズムは，認識対象や認識装置の使用環境が異なると，その都度作り直す必要がある。このため認識システムを種々の用途に幅広く適用可能とするためには，高度な認識アルゴリズムを短期間に構築する技術を確立する必要がある。認識のアルゴリズムとしてこれまで用いられてきたのは，決定木（デシジョントリー）による手法である。決定木法では，画像情報の中にあらかじめ複数の特徴量を定義しておき，これらを用いた適当な等式や不等

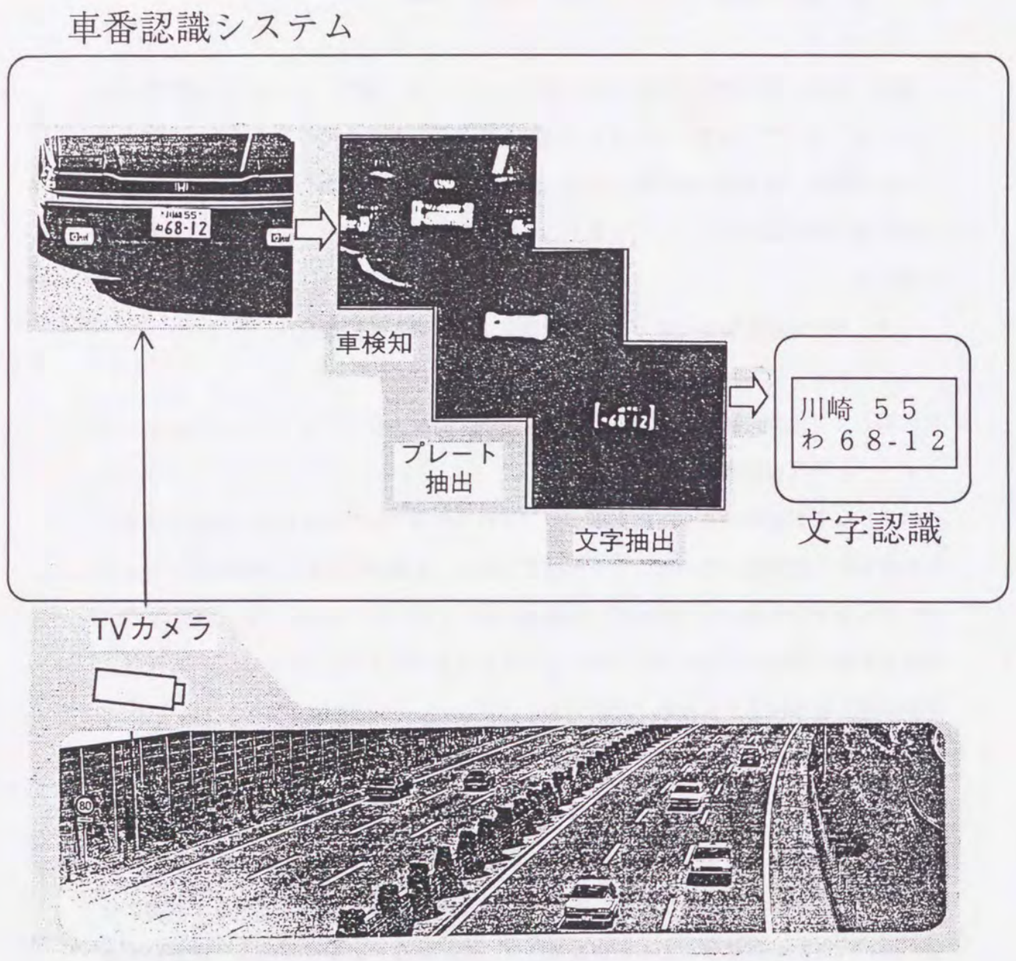


図1.1　車番認識システムの処理の概要

式が成立しているかどうかをトリーの流れにしたがって判定し，カテゴリの可能性を限定する処理を繰り返すことにより，結論を導く。一方この手法では，カメラとナンバプレートの位置，角度のバラツキや，ナンバープレートの汚れ，昼夜の輝度の変化等に対して普遍的な認識を行うためには膨大なサイズの決定木が必要となる。このチューニングは主として人手で行われ，経験豊富なエンジニアをもってしても，数日～数週間を必要としている。この作業は新しい文字が追加されたり，車番認識システムの使用環境が異なる度に必要である。また海外等で，記載されている文字や数字の種類やフォントが異なるシステムを立ち上げる場合には，その都度決定木を一から構築することが必要となる。したがってシステムの構築，拡張時に，迅速に認識アルゴリズムを作り上げる要求に対応できないことが大きな問題である。

対象を認識・制御するための複雑な方法論をいかにして迅速に構築するかという課題は，車番認識システムの他にも，各種認識システム，異常診断・予防保全システム，非線形な制御対象に対して制御系を構築する場合等に存在する。このためこれらの作成を極力簡単化することが望まれている。

（2）の要因は，制御や認識の対象となる装置や物体が複雑化していること，および複雑さゆえにこれまで操作員が手動による制御や目視による認識を行なっており，自動化が困難とされてきた対象を自動化し，省人化，高信頼化を実現したいという要求に起因している。操作員の手動介入が不可欠なプラントとして，熱間圧延プラントの制御を例に説明する。

図1.2は熱間圧延システム[2]の全体構成で，液状の鋼を冷却して鋼材（スラブ）を生成する連続鋳造機，加工に先立って鋼材を1000度～1200度に昇温させる加熱炉，幅出しを行うサイジングプレス，鋼材を大まかに圧延する粗圧延，前後を切り取るクロップシャー，最終的な厚みまで板厚を減少させる仕上げ圧延，圧延材を水冷するストリップ冷却，圧延材をコイルに巻き取るダウンコイラの各プロセスから構成される。各プロセスの制御目的は，鋼板の厚みであったり温度であったりするが，いずれの場合もダイナミックに変化する鋼板の状態量を考慮して制御する必要がある。しかも予熱空気や冷却水の温度，鋼板の塑性特性のバラツキ等の無視できない未知ファクタが存在する。通常，それぞれの工程における温度や，板厚，板幅の制御では，制御対象

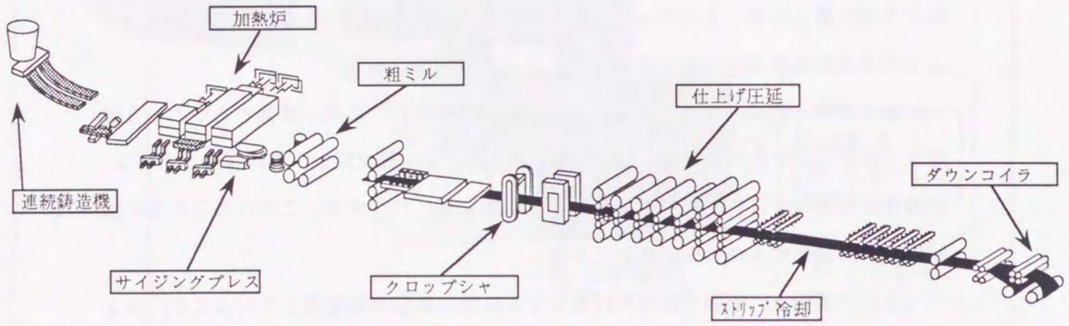


図１.２　熱間圧延システムの全体構成

のモデルを用いた演算で制御の目標値を厳密に算出し（以下，この計算をセットアップと言う），これを指令値としてＰＩ制御等が行われる。したがって制御精度を向上させるためには，高精度なセットアップを行わなければならない。

一例として，本論文で対象とする加熱炉における鋼材の温度制御では，鋼材の熱現象を記述するモデル（鋼材モデル）を基に，鋼材を目標温度に昇温させるのに必要十分な温度指令値を決定する。鋼材モデルは，熱伝達，熱伝導係数をはじめ，鋼材の材質や炉の構造等に依存したいくつかの定数を有しているが，これらの一部は操業条件や材質のばらつき，前述した未知ファクタに依存して非線形に変化する。モデルと実際の制御対象との誤差が無視できなくなると温度の指令値が不適切となり，加熱炉から装出される鋼材の温度に誤差を生じる。このような場合，鋼材モデルをチューニングするとともに，これが制御精度向上に不十分な場合には，システムを監視している操作員が温度指令を手動で変更することになる。しかし熟練操作員の減少により，信頼性の高い手動介入を将来にわたって継続することが難しくなっている。

このように自動化技術の進展に伴い，これまで自動化の範疇でなく，操作員が手介による制御や目視による認識を行ってきた対象を自動化し，制御システムの運転に携わるオペレータの人員をさらに削減したり，高信頼化を実現したいという要求は多い。このような対象では，対象の数学モデルが得られなかったり，得られたとしても複雑すぎて実時間の制御や認識に用いることが難しい。したがってこれらに基づいて厳密なアルゴリズムを構築することは現実的でない。このような理由で熟練操作員の手動介入や目視に拠っている制御・認識部は，前述した圧延制御システムのみならず，産業システムの随所に顕在している。これらを自動化し，省力化，高信頼化を図るとともに，高度な制御動作や判断，認識，運転ノウハウを，将来にわたって蓄積することの必要性が高まっている。

第1.2節　ニューラルネットへの期待

前節で述べた問題点を解決するアプローチは多岐に渡っているが，対象そのものやこれと信号の授受を行う制御・認識装置のインテリジェンスを向上させ，介在する人間の負担を軽減することは有効な解決策の一つである。このための手法として，ＡＩ（Artificial Intelligence）技術の導入が活発で，とりわけ人間の脳細胞の信号伝達に範をとった基本素子であるニューロンをネットワーク構造に結合した，多層ニューラルネット[3]の学習能力と汎化（推論）能力の利用が注目されている。以下，特に断らない限り，本論文でニューラルネットは多層ニューラルネットを意味する。さらにあらゆる連続な非線形関数は3層のニューラルネットで近似できることが証明されおり[4]，また3層構造で立案した手法を，4層構造以上のニューラルネットへ適用することは比較的容易と考えられる。このため本論文では，3層構造のネットワークに限定して検討を進める。

ニューラルネットの構造，学習法に関する詳細は第2章で説明するため，ここでは簡単に述べる。3層ニューラルネットの構造は，一般に図1.3で表されるように，入力信号を受信する入力層，中間層（隠れ層），信号を出力する出力層からなる。生体における各ニューロンが他のニューロンに信号を出力するかどうかの臨界値であるしきい値を，ニューラルネットではシグモイド関数と呼ばれる連続の単調飽和関数で置き換える。これにより任意の連続関数が近似可能，線形分離不可能なカテゴリでもクラスタリング可能といった能力面の向上の他に，しきい値関数を微分可能としたことで最急降下法を模擬したバックプロパゲーション学習法の適用が可能となる。

ニューラルネットは，入力とこれに対応したお手本（教師信号）を用意しさえすれば，学習により入出力関係を自動的に構築できる。このため制御や認識のアルゴリズムを短時間で作ることができる。また対象が複雑に変化しても再学習や追加学習により，アルゴリズムの構造を対象の変化に機械的に追随させることが可能である。能力的にも，前述したように中間層ニューロン数に制限を加えなければ任意の連続関数を近似できることが証明されている。また線形分離できないカテゴリが含まれるクラスタリング問題で識別アルゴリズムを構築するのは容易でない場合が多いが，このよう

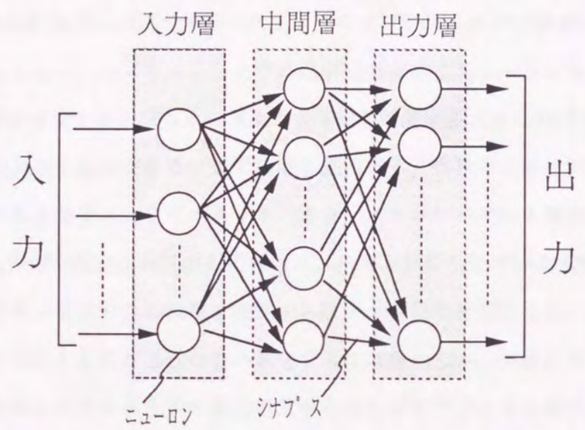


図1.3　多層ニューラルネットの構成

な対象であってもニューラルネットは対処可能であることが示されている。したがって制御や認識のアルゴリズムが複雑であっても，問題の性質をそれほど吟味することなくこれらを構築できる。以上が，制御・認識アルゴリズムを構築，保守する手法としてニューラルネットが注目されている所以であり，前節で述べた要因（1）（2）を解決する有力な手段として期待できる。

第1.3節　ニューラルネット適用上の問題点

ニューラルネットのこのような利点を実際のシステムで有効に発揮させるためには，いくつかの問題を解決しなければならない。とりわけ大きな課題として，

（1）ニューラルネット構築の試行錯誤が多大

（2）ニューラルネットの能力を向上させる手法の検討が不十分

（3）ニューラルネットを実システムで有効に用いるための方法論が未確立

の3点が挙げられる。

（1）はネットワークの構成決定に伴う試行錯誤とネットワークの性能を高めることに要する試行錯誤に大別でき，システム立ち上げ時にこれらに必要とされる時間は，一般に少なくない。システム立ち上げの効率化を目的にニューラルネットの適用を検討する主旨から考えると，ニューラルネット構築時間が多大となれば本末転倒であり，これを極力小さく必要がある。まず前者を説明する。複数のニューロンを層状に結合して構成される多層ニューラルネットの場合，ネットワークの入出力関係は各ニューロンの相互作用で決定され，それぞれのニューロンが果たしている役割は明確でない。したがってネットワークの構成を決定する際に，中間層を何層にし，各層のニューロン数をいくつにするのが最適なのか算定するのは難かしい。一般にニューロン数が少なすぎるとニューラルネットの能力不足から性能が不十分となったり，学習が収れんしない。反対に多すぎるとニューラルネットの応答性が低下する上，入出力関数が描く超曲面が過度に複雑となることの汎化能力（認識能力，推論能力等）に及ぼす影響は不明である。ニューラルネットを制御・認識のようなリアルタイム処理へ適用する場合，応答性の低下は深刻で，可能な限り最少のニューロン数でネットワークを構成する必要がある。3層ニューラルネットに議論を限定すると，決定に時間を要するのは中間層ニューロン数で，設定の自由度が大きい一方，最適数を決める明確な指針は今のところない。最適数を解析的に導くのは容易でないことから，学習済みのニューラルネットの重み行列の固有値に着目して最適数を導く手法[5]や，学習中に中間層のニューロン数を増減させ，最適数を有したネットワークに学習を収れんさせる手法[6]等が提案されているが，実用的にどの程度有意な値が得られるか定かでなく，実シス

テムに適用して良好な成果を得るには至っていない。現状は，ニューロン数を変化させたネットワークを複数構築し，これらの性能評価を基に最適数を決定する試行錯誤的なアプローチに依っており，ニューラルネット応用システム早期立ち上げの障害となっている。

ネットワークの性能を高めることに要する試行錯誤として，学習が収れんしたときの誤差を減少させる目的で学習係数や慣性定数等の学習パラメータを変化させたり，ネットワークの性能向上を目的として教師信号の組み合わせやシナプスの重み係数の初期値を変えて，ネットワークの学習を繰り返すプロセスがある。前者に関しては学習パラメータを学習中に適応修正し，収れん特性を向上させる試み[7]が，また後者に関しても教師信号の品質を学習に先立って検定する手法[8]が提案されている。これらは試行錯誤の回数を低減するのに効果があるものの，依然として最終的なネットワーク構築までに何度かの学習の繰り返しが必要とされている。したがってこのようなアプローチと同時に，学習を終えたネットワークの各教師信号に対する誤差特性や未学習データに対する汎化特性の把握を容易化し，ネットワークの性能を高める作業を適切に支援するニューラルネット構築環境の確立が望まれている。

次に（2）の問題点を説明する。ニューラルネットを用いて自動システムを構築する場合，ニューラルネットの能力を目的に対して十分な値に高めることが重要なのは言うまでもない。自動化の結果，熟練オペレータにより行われて既存の制御・認識より性能が低下した場合，通常，自動化そのものが見送られる。ニューラルネットの学習方法として現在広く用いられている手法はバックプロパゲーションであるが，学習の結果得られたニューラルネットが，その潜在的な能力を最大限発揮することを保証しているわけではない。学習方法について，収れんの高速化[9]や学習の収れん誤差最小化[10]の観点から数多く検討されているが，汎化能力向上を目的とした改善は3章で詳述するように，これまであまりなされていない。したがって要求される制御や認識の性能が高度で，バックプロパゲーションを単純に適用し，前述の試行錯誤を繰り返して実現できる性能を超えている場合，適切な学習方法の改善によりネットワークの能力を十分な値に高めることが，ニューラルネット応用システムを構築するための必要条件となる。

また教師信号の学習をどれほど高精度に行なったとしても，未学習信号に対する誤差を0にすることはできない。したがって特定の入力に対する誤差の値がネットワークの性能を決定づける用途では，これを低減することがニューラルネットの能力向上のために大きな問題となる。このような用途として代表的なのは制御モデルのパラメータチューニングであり，チューニングが終了する領域近傍におけるニューラルネットの出力誤差はチューニングの定常誤差に直接対応し，精度の低下を招く。したがってニューラルネットの学習能力を活かしたパラメータチューニングを実現するためには，チューニング終了領域近傍のニューラルネットの出力誤差を低減させる必要がある。同様の用途は，ニューラルネットをレギュレータ制御の指令発生に用いる場合など多数考えられ，この問題の解決は，ニューラルネットをこの分野で実用化するために必須である。

（3）は，ニューラルネット構築環境や実際の装置へ搭載する場合の方法論が未確立な点を指摘している。ニューラルネットを産業システムで用いる場合，計算時間を要する学習はワークステーションやサーバ等の計算能力の高い上位機器で行い，リアルタイム性が要求される制御や認識の実行はプログラマブルコントローラ等の対象に直結した下位制御機器で実行する，機能分散アーキテクチャが採用されることが多い。このとき問題となるのは，ニューラルネット応用システムの構築を容易化し，前述した試行錯誤を軽減するための上位機器における支援システムの構成と，リアルタイム性を低下させることなくニューラルネットを実行するための下位機器における処理方式である。このようなニューラルネットを実際のシステムで良好に稼働させることに関する検討は，これまでほとんどなされていない。産業システムにおいてニューラルネットの実用化を広範なものとするためには，使い勝手の良さと性能を兼ね備えた実装方法を明らかにし，試作システム等を用いて定量的に評価する必要がある。

第1.4節　研究の目的と概要

産業システムにおける制御・認識技術としてニューラルネットの有効性を高めるためには，前節で指摘した問題点を解決しなければならない。本論文はこれらの解決法を明らかにした上で，ニューラルネットを用いて機器のインテリジェンスを高めることが，複雑な制御・認識アルゴリズムの生産性向上，および複雑な対象の自動制御・認識に有効なことを，実際にいくつかの制御・認識システムをニューラルネットを用いて構築した結果を基に定量化することを目的とする。

ニューラルネット構築の試行錯誤低減に関しては，最適な中間層ニューロン数を簡易に決定する方式の確立を目指す。ここで最適の定義は汎化能力が最大であることに加え，適用分野がリアルタイムシステムであるため，高速演算の観点からできる限り最小の個数でネットワークを実現することである。併せて，この方式を搭載したニューラルネット構築支援ツールを開発し，試行錯誤を種々のレベルで適切に支援する環境を提供する。

ニューラルネットの能力向上については，まず認識アルゴリズムの高性能化を目的に，認識能力最大化の観点から学習方法の改善を目指す。本論文では車番認識システムのように，実機で収集したデータを教師信号として利用する用途を対象にした認識性能の向上手法を議論する。実機で収集したデータを教師信号として利用する場合，教師信号が母集団を代表した性質の良いものである保証はなく，収集コストや時間の制約から量的に不十分な場合もある。車番認識システムにおいても，種々の条件で撮影した画像データをデータベース化して用いるとしても，教師信号が質および量的に十分な場合ばかりではない。したがってこれを前提に，与えられた教師信号で最善の認識性能を得ることが認識アルゴリズム構築の目的となる。以上より，本論文でニューラルネットの性能の指標とする汎化能力は，このようにして構築されたアルゴリズムを，未学習の入力信号に種々のノイズが重畳する実機環境で用いたときの認識率や制御性能，予測精度等と対応する。特に認識アルゴリズムの場合には，教師入力信号を母集団とする各カテゴリに対して，それ以外の入力空間を平等に分割できた度合いと対応する。

ニューラルネットの高性能化に関してさらに，非線形制御モデルのチューニングを対象に，アーキテクチャ改善の観点からこれを少ないチューニング回数で高精度に行うことを検討し，ニューラルネットをこの分野における方法論として確立することを目的とする。熱間圧延システム加熱炉プラントの温度制御に適用した結果を基に，開発手法のチューニング精度およびチューニング回数低減への効果を定量化し，制御精度のばらつきからオペレータの介入が不可避であったこの種のプラントに対して，適切な制御モデル同定とこれを用いたモデルベースト制御により，完全自動制御の可能性を示す。

最後にニューラルネットを制御，認識を行う機器に搭載し，実際の産業システムで良好に稼働させることを目的とした実装方式を明らかにする。まず，ニューラルネットの処理をプログラマブルコントローラ等のＤＤＣ（Direct Digital Controllers）系で，付随する制御演算（シーケンス演算，ループ制御演算等）と混在させて効率的に実行させる演算方式を確立するために，ＤＤＣ系の実行タスクのモデル化方法，およびこのモデル化されたタスクをＤＤＣ系で高速実行するタスク切り換えアルゴリズムを検討し，ＤＤＣ系の高応答化を図る。さらにニューラルネット応用システムの開発期間短縮，メンテナビリティ向上を目的として開発したニューラルネット応用システム開発支援ツールのソフトウェア構成を明らかにする。また本論文で開発したニューラルネットの最適中間層ニューロン数決定手法および学習方法の，支援ツールへの組み込み方法についても示す。さらに実機認識システムとして車番認識システム，制御システムとして加熱炉プラントの温度制御システムをニューラルネットを用いて実際に構築し，開発手法を適用したニューラルネットの，アルゴリズム生産性，制御・認識性能向上への貢献を定量化する。

第1.5節　本論文の構成

前節で述べた目的と対応して，本論文は，以下の各章により構成される。

第2章では，最初に本論文で検討の対象とするニューラルネットの構成を簡単に説明した後，最適な中間層ニューロン数を決定するアルゴリズムを提案する。最初に，ニューロン数が増えるとこれらの挙動の間で線形相関が増大することを明らかにし，線形相関の大きさを冗長なニューロン数に対応づけることにより，必要な中間層ニューロン数の算定が可能なことを示す。次に各ニューロンの出力に対し，線形重回帰分析を再帰的に適用することで冗長なニューロン数を評価する手法を開発する。そして十分な中間層ニューロンを初期設定したニューラルネットを学習しておき，開発手法を用いて算出した冗長なニューロン数を初期設定値から減じることにより，定義に従った最適ニューロン数が求められることを明らかにする。最後に，実機の車番認識および浄水プラントの凝集剤注入制御を行うニューラルネットに開発手法を適用し，最適ニューロンが良好に評価できることを示す。

第3章ではニューラルネットにより構築された認識アルゴリズムの認識能力向上を目的として，学習方法の改善を検討する。まず広範に用いられているバックプロパゲーションによる学習では，クラスタ（カテゴリ）間の境界が，学習の終了時に特定のクラスタに近接する場合が多くあり，これがニューラルネットの認識能力を低下させていることを指摘する。つぎにこの問題を解決する学習方式として，教師入力信号にその都度発生した乱数を重畳してネットワークに提示することが有効なことを示し，認識能力向上のメカニズムおよび最適な乱数の印加方法を明らかにする。

第4章ではニューラルネットの中間層ニューロン数と認識能力の関係をモデル化し，第2章で定義した最適中間層ニューロン数の考え方の一般性を明らかにする。とりわけ最適数が，ニューラルネットが認識するデータに含まれるノイズの大きさに依存することを示し，これに対応して第2章で述べた手法の厳密化を検討する。第3章および第4章で提案した手法は，いずれもアルファベット認識問題および車番認識システムで有効性を確認する。

第5章では，制御モデルのチューニング用に従来のニューラルネットを拡張したアジャスティングニューラルネットを提案する。まず従来のニューラルネットで制御モデルをチューニングすると，ニューラルネットの出力誤差に対応してモデルに定常誤差が残留するため，制御精度が十分に向上しないことを指摘する。次に新しいアーキテクチャのニューラルネット（アジャスティングニューラルネット）の構成と学習方式を示し，熱間圧延の加熱炉プラントを模擬したシミュレーションにより，チューニングされた制御モデルの高精度化，チューニング回数の低減に効果が大きいことを明らかにする。

第6章ではニューラルネットを汎用コントローラへ搭載する場合の性能向上を検討する。このとき実行性能を向上させるためには，第2，第4章で述べた構成最適化によるニューラルネット演算の高速化だけでなく，これに付随する演算（シーケンス処理，ループ制御処理等）の実行を高速に行い，しかも各処理の切り換えを効率化する必要がある。これを実現するために，各処理を離散事象としてモデル化する方法，および処理の切換えをペトリネットのトークン遷移を模擬したタスクスイッチ方法で高速に行うアルゴリズムについて述べる。

第7章で開発手法を用いて実際のシステムを構築した結果を述べる。具体的には，車番認識システムと加熱炉温度制御システムについて，ニューラルネットを適用したシステムの構成と基本動作を示す。また既存の手法に対して，アルゴリズム生産性，制御・認識性能，処理時間等の比較を行う。さらにワークステーションで実現されたニューラルネット構築支援システムのソフトウェア構成を示し，第2～第4章で開発した技術の実際のシステムにおける実現方式を明らかにする。

第8章で本論文の成果を整理し，これをまとめて結言とする。

1章の参考文献

（1）三島·高藤·小林·藤原·柴田，「画像処理を用いた車番認識システムの文献開発」，電学論D, Vol. 109, 5, pp.333 - 338 (1989)

（2）梶原·小山·長島·清水，「生産効率と品質向上を目指す圧延設備の動向」，日立評論, 75, pp4 - 10 (1993)

（3）D.E. Rumelhart et al. : "Parallel Distributed Processing (PDP)" , Vol.1, MIT Press, Cambridge Massachusetts (1986)

（4）K. Funahashi : "On the Approximate Realization of Continuous Mapping by Neural Networks : Neural Networks" , Vol. 2, 3, pp.183-192 (1989)

（5）Q.Xue et al., "Analyses of the Hidden Units of Backpropagation Model by Singular Value Decomposition (SVD)",IJCNN'90 - WASH - DC, pp.739 - 742 (Jan. 1990)

（6）T.Ash, "Dynamic Node Creation in Back Propagation Networks", IJCNN'89 - WASH - DC, pp623 (June. 1989)

（7）N.Iijima, H.Izumi, N.Kamiyama, H.Mitui, M.Yoshida, and M.Sone : "Tuning Learning Rate and Momentum of Back-Propagation", IJCNN'91-Seattle, II A-963 (1991)

（8）S.Abe, M.Kayama, H.Takenaga, and T. Kitamura : "Determing the Convergence of Pattern Classification Neural Networks prior to Learning", IJCNN'92-Beijing, II-799～804 (Beijing, China, 1992)

（9）B.L.Kalman and S.C.Kwasny : "A Superior Error Function for Training Neural Networks", IJCNN'91-Seattle, II-49～54

（10）R.Forsyth : "Neural Learning Algorithms:Some Empirocal Trials", Neuro-Nimes'90, pp.301～317

第2章　ニューラルネットの最適中間層決定方法

第２章　ニューラルネットの最適中間層決定方法

第２.１節　研究の概要

ニューラルネットを実システムに適用する場合の問題の一つは，目的に対応した最適なネットワーク構成が不明な点である。ニューラルネットを車番認識システムのようなリアルタイム認識や，同様にリアルタイム性の高い制御へ応用する場合，応答性やコストミニマムの観点から最小構成が望まれる一方，有用なニューロンを削除したことによる性能低下を回避する必要がある。したがってネットワークの構成は厳密に決定する必要があるが，現状は入出力関係の学習や，ネットワークが未学習入力を汎化するのに十分な個数を，ニューロン数を変化させた種々のネットワークを用いた試行錯誤的な性能比較の結果により決定している。このためニューラルネット応用システムの早期立ち上げの障害になっている。

実際の応用で特に問題となるのは中間層ニューロン数の決定法であり，この指針を与える手法はこれまでにもいくつか検討されている。まずXue [(1)] らは，入力層と中間層を結ぶシナプスの重みを行列化し，この行列のrankの値に着目する方法を提案している。これはＳＶＤ(Singular Value Decomposition)法を用いて縮約した直交行列が有する固有値の数を必要なニューロン数とする手法である。しかし中間層ニューロン数の最大値が，入力層ニューロン数と出力層ニューロン数の小さい方の値となるため，中間層で情報圧縮される用途に適用が限定される問題がある。たとえば $Y=X^2$ を $0 \leqq X \leqq 1$ の領域において適当な精度で近似する，１入力１出力のネットワークの場合，この手法で得られる最適ニューロン数は常に１となり，これが実際の最適数と対応していないことは明らかである。

少数のニューロン数から学習を開始し，ネットワークの出力誤差が収れんした時点でニューロンを追加することで，所望の出力誤差を与える最小ニューロン数を得る手法も報告されている [(2)]。しかし学習の終了までに複数回の収れん計算が必要となるため，ネットワークの規模が大きくなると学習時間が多大になると予想される。またこれら２つ方法で，得られたネットワークが十分な汎化能力を有している保証はない。

学習済みネットワークの汎化能力を高めることに主眼をおいた，ニューロン数の決定方法も提案されている[3][4]。文献３は，ＡＩＣ(Akaike Information Criterion)とＭＤＬＰ(Minimum Description Length Principle)を情報量規準に，入出力の関係を近似する最尤モデルを求め，モデルに必要とされるパラメータ数からニューロン数を決定する手法を述べている。また文献４では，通常，ニューラルネットではローカルミニマム等のため，収れん状態にあっても最尤状態が得られないことを指摘した上で，これを前提にCross Validationを使って情報量規準を推定する手法を提案している。情報量規準により算出した最適モデルを中間層ニューロン数に換算し，最適数に対応づけることの問題点はすでに指摘されている[5]が，ニューラルネットをクラスタリング問題へ適用する場合には，さらに，入力と出力の関係を最大尤度で同定することが，汎化能力の高いネットワークをつくることと直接対応しない[6]ことが問題となる。提案されている内容はこの相違について考慮されていないため，クラスタリング問題において，得られたニューロン数が実際に汎化能力を最大化する値とどの程度対応しているか明らかでない。

本章では以上の研究状況を踏まえた上で，クラスタリング問題に対しても有用な汎用性の高い最適中間層ニューロン数の評価方法を提案する。まず検討の前提である既存のニューラルネットについて，構成と学習方法を簡単に述べた後，検討結果を示す。具体的には，学習を終えたニューラルネットの中間層ニューロンの相互動作に着目し，これらを統計解析して，中間層ニューロンの最適数を算定するアルゴリズムを提案する。提案方法ではまず，十分な中間層ニューロン数を備えた学習済みのニューラルネットを構築する。そしてこれに教師信号を再入力したときの中間層ニューロンの出力に対し，線形重回帰分析を再帰的に適用する手法で非線形関係の大きさを抽出することにより，最適数を決定論的に求める。最後に開発手法を，クラスタリング問題の一例として車番認識システムへ適用し，有効性を明らかにする。さらに非線形関数近似問題として，浄水プラントの薬品注入量制御システム[7]への適用を試み，この場合も有効な最適ニューロン数が評価できることを示す。

第２.２節　ニューラルネットの構成と学習方法

脳の基本素子であるニューロンや，これらを多数結合したニューラルネットワークの構造や情報処理のメカニズムに範をとり，脳の持つ秀れた情報処理能力を計算機上で実現することを目的とした研究は，１９４０～６０年に一時大きく盛り上がった。その後能力の限界等から下火となったが，ネットワークを多層化して能力を高めること，これを学習するバックプロパゲーションと呼ばれるアルゴリズムが提案されたこと，またこれに要する大量の繰り返し演算が計算機技術の進歩によりフィージブルな時間オーダで実行可能となったこともあって，１９８０年代の後半から今日にかけて再び大きなブームとなっている。

ニューロンは多入力―１出力の非線形素子である。図２.１に神経系におけるニューロンと，これをモデル化した形式ニューロンモデルを示す。あるニューロンの入力の数をＮ，各入力の信号の強さを $s_1, s_2, \ldots, s_n$，各シナプスの荷重を $w_1, w_2, \ldots, w_n$ とする。このときニューロンの出力 x は，

$$x(t+1) = 1\{\sum_{i=1}^{N} w_i \cdot s_i(t) - h\} \tag{2.1}$$

ただし　$1\{u\} = 0 \quad u < 0$

$1\{u\} = 1 \quad u \geqq 0$

となり，入力の重み付き荷重和がしきい値 h を越えたとき，ニューロンは１を出力し，それ以外の場合は０を出力する。

ニューロンをネットワーク構造に結合し，多層ニューラルネットとして情報処理に用いる場合の構造は，一般に図２.２で表される。ここであらゆる連続な非線形関数は３層のニューラルネットで近似できることが証明されおり，また３層構造で立案した手法を，４層構造以上のニューラルネットへ適用することは比較的容易と考えられる。このため本論文では，以下，３層構造のネットワークに限定して検討を進める。

図２.２において入力と出力の関係は，

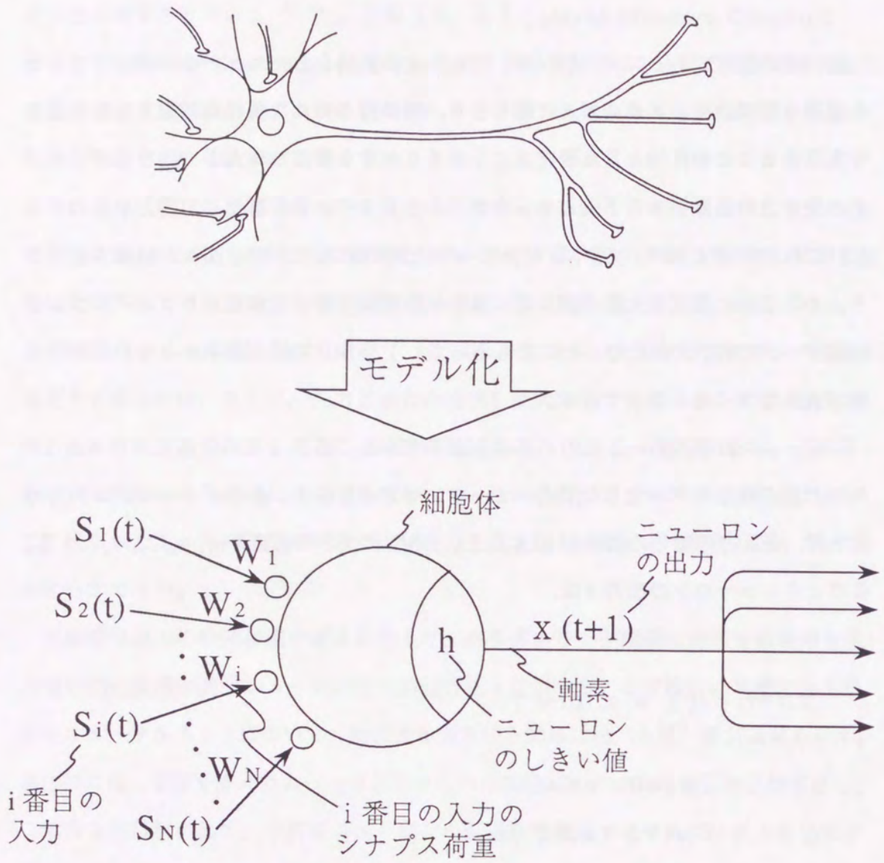


図2.1　形式ニューロンモデル

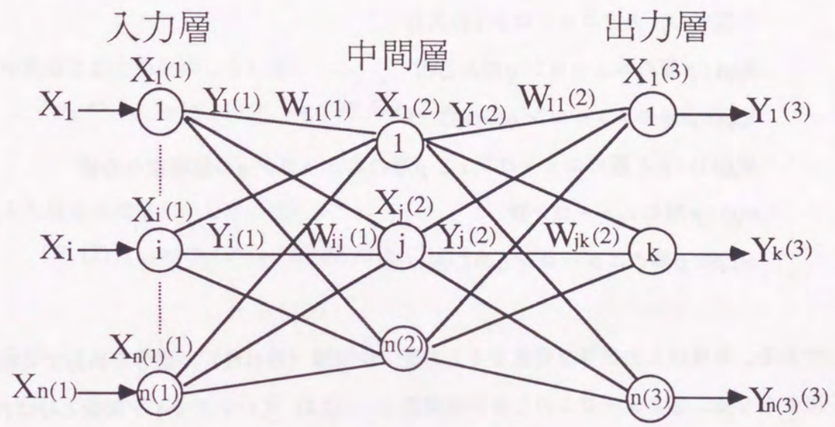


図2.2　3層ニューラルネットワーク

$$Y_i(1) = X_i(1) = X_i \tag{2.2}$$

$$X_q(p) = \sum_{r=1}^{n(p-1)} W_{rq}(p-1)\, Y_r(p-1) + \theta_q(p) \tag{2.3}$$

$$Y_q(p) = f\{X_q(p)\} = 1/[1+exp\{-X_q(p)\}] \tag{2.4}$$

ここで X_i: 入力ニューロン i の入力

$X_q(p)$: p 層のニューロン q の入力和

$Y_q(p)$: p 層のニューロン q の出力

$W_{rq}(p\text{-}1)$: p-1 層のニューロン r と p 層のニューロン q の間の重みの値

$n(p)$: p 層のニューロン数

$\theta_q(p)$: p 層のニューロン q のしきい値

である。各層は入力信号を受信する入力層，中間層（隠れ層），信号を出力する出力層からなる。各ニューロンのしきい値関数は，（2.4）式のシグモイド関数と呼ばれる連続の単調関数で置き換える。これにより任意の連続関数が近似可能，線形分離不可能なカテゴリでもクラスタリング可能といった能力面の向上の他に，しきい値関数を微分可能としたことで最急降下法を模擬したバックプロパゲーション学習法の適用が可能となる。

3 層ネットワークを例にバックプロパゲーション学習法を示す。教師入力信号を入力したときの k 番目の出力ニューロンの値 $Y_k(3)$ に対応した教師信号を T_k とする。このとき各出力ニューロンの誤差 E は，

$$E = (1/2)\cdot \sum_{k=1}^{n(3)} \{T_k - Y_k(3)\}^2 \tag{2.5}$$

となり，学習は E を小さくする方向に進められる。すなわち（2.5）式で算出された誤

差 E を用いて，重み係数及びしきい値は（2.6）～（2.9）式に従って更新される。n は重みの更新回数を表し，α は学習係数である。

（中間層と出力層間の重み係数）

$$\{W_{jk}(2)\}_{n+1} = \{W_{jk}(2)\}_n + \alpha\cdot\{\partial E/\partial W_{jk}(2)\}_k \tag{2.6}$$

（入力層と中間層間の重み係数）

$$\{W_{ij}(1)\}_{n+1} = \{W_{ij}(1)\}_n + \alpha\cdot\{\partial E/\partial W_{ij}(1)\}_j \tag{2.7}$$

（中間層と出力層間のしきい値）

$$\{\theta_k(3)\}_{n+1} = \{\theta_k(3)\}_n + \alpha\cdot\{\partial E/\partial \theta_k(3)\}_k \tag{2.8}$$

（入力層と中間層間のしきい値）

$$\{\theta_j(2)\}_{n+1} = \{\theta_j(2)\}_n + \alpha\cdot\{\partial E/\partial \theta_j(2)\}_j \tag{2.9}$$

ここで

$$\{\partial E/\partial W_{jk}(2)\}_k = (\sigma_{NN})_k\cdot Y_j(2) \tag{2.10}$$

$$\{\partial E/\partial W_{ij}(1)\}_j = (\varphi_{NN})_j\cdot Y_i(1) \tag{2.11}$$

$$\{\partial E/\partial \theta_k(3)\}_k = (\sigma_{NN})_k \tag{2.12}$$

$$\{\partial E/\partial \theta_j(2)\}_j = (\varphi_{NN})_j \tag{2.13}$$

$$(\sigma_{NN})_k = -\zeta_k\cdot Y_k(3)\cdot\{1-Y_k(3)\} \tag{2.14}$$

$$(\varphi_{NN})_j = \sum_{k=1}^{n(3)} (\sigma_{NN})_k\cdot W_{jk}(2)\cdot Y_j(2)\cdot\{1-Y_j(2)\} \tag{2.15}$$

$$\zeta_k = T_k - Y_k(3) \tag{2.16}$$

である。結局，ニューラルネットと教師信号の差分を減少させる方向に各層の$W_{rq}(p\text{-}1)$を変更することを繰り返して学習を進めることになる。学習の終了（収れん）は，ニューラルネットがすべての教師信号に対して一定値以下の差分となったことで判定するのが一般的である。

第2.3節　中間層ニューロン数決定方法

第2.3.1項　最適ニューロン数の定義と算定方法の考え方

本章では図2.2に示す3層ニューラルネットを対象に検討を行う。中間層のニューロン数は，少な過ぎると能力不足からニューラルネットの汎化能力を低下させたり，学習を所望の精度で収れんさせるのが困難となる。逆に多すぎるとネットワークの計算量が増大し，ハードウェアの増大や応答性の低下を招く。中間層のニューロン数と汎化能力の関係についてはこれまでにもいくつか報告されており，教師信号を学習するのに十分な個数が備えられていれば依存関係は少ないという指摘[8]が多いが，汎化能力が最大となるニューロン数の存在を示唆した報告[9]もある。これは学習に用いたデータと評価に用いたデータの性格が一様でないこと，評価に供した中間層ニューロン数が異なっていること等，種々の理由によると思われるが，一般に中間層ニューロン数の増大は，ニューラルネットの性能向上と直接対応するものではないと考えられている。第2.4節で示すシミュレーション結果からも，ニューロン数が少ないときネットワークの汎化能力がやや劣る結果となっているが，一定数以上のニューロンを備えていれば汎化能力はニューロン数に依存せず飽和した一定値となる。

一方，ニューラルネットを制御や認識などのリアルタイム性の制約が大きい用途に用いることを考えた場合，制御応答や認識速度との兼ね合いからニューラルネットは最小構成が望ましい。そこで以下，“最適ニューロン数”を，“最小ニューロン数で最大汎化能力”を実現する値と定義し，これを容易に得る方法を検討する。

まず最適ニューロン数が何で決定されるかを考える。一般的には，入出力データの個数や非線形性，学習の収れん条件等様々な要因が挙げられる。したがってこれらの相互関連や，それぞれのニューロン数への寄与分を数値化し，解析的に値を算定するのは困難と推察される。そこで本論文では，これらの要因を総て含んだ情報を用いた統計的な解析により最適数の算出を行う。以下このような統計量として，収れんに十分な数の中間層ニューロン数を備え学習を終えたニューラルネットに，教師入力信号を再入力したときの中間層ニューロンの出力のふるまいに着目し，これを統計解析することを考える。

一般に中間層ニューロン数が過剰になると，いくつかのニューロンが類似の動作をすることが報告されている[8]。また第2.4節のシミュレーション結果からも，中間層ニューロン数の多いネットワークでは，これらの間の統計挙動に強い線形関係が生じ，結果的に一つのニューロンが果たす役割が減少することが示されている。したがって得られた統計量から，ニューロン間に含まれる線形成分を定量化し，これに相当するニューロン数を設定されているニューロン数から減じることにより，最適中間層ニューロン数が得られると考えられる。

このアイデアを可視化するために，各中間層ニューロンが入出力関係のマッピングにおいて負担している仕事分を，図2.3で模式的に表現する。図はニューロン数が4の場合を示しており，各ニューロンの仕事分はそれぞれの面積で表わされる。中間層における信号変換に本来必要な仕事量は，図2.3の太線の内側で表され，これが最適ニューロン数に対応する。一方，各ニューロンは図のような重複部分（線形関係を有した部分）を持ちながら，その一部を分担していると考えられる。入出力関係をマッピングするのに本来必要な仕事量は，ニューロン数にかかわらず不変のはずである。したがって太線の内側の面積を評価できれば，対応したニューロン数として最適数が得られる。このためには重複部分の面積を定量化し，ニューロン数の初期値から差し引くことが必要である。以下，線形回帰分析[10]を用いた統計的手法により，この処理を行うことを検討する。

第2.3.2項　最適ニューロン数の算出方法

図2.2の多層ニューラルネットにおいて，学習を終えたネットワークに教師入力信号を再入力したときの，各中間層ニューロンの出力値を $Y_1(2)\sim Y_{n(2)}(2)$ とする。これらの一つとして $Y_j(2)$ を抽出し，他の $\{n(2)-1\}$ 個の中間層ニューロンの出力の線形和として，

$$\widetilde{Y}_j(2) = a_0 + a_1 Y_1(2) + a_2 Y_2(2) + \cdots + a_{n(2)} Y_{n(2)}(2) \qquad (2.17)$$

（ただし $a_j Y_j(2)$ を除く）

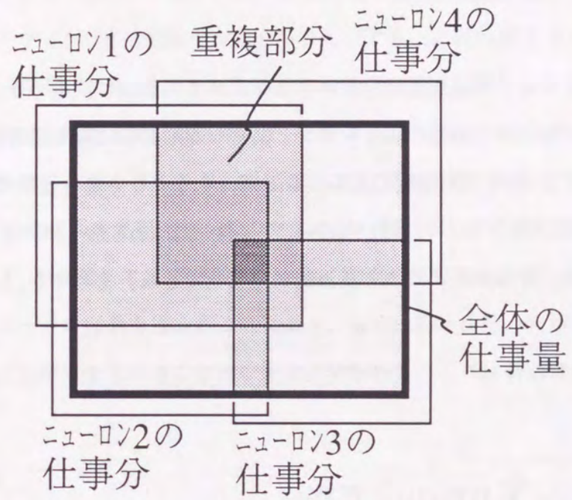


図 2.3　中間層の仕事量模式図

で近似することを考える。係数 $a_0 \sim a_{n(2)}$ は $Y_j(2)$ と $\widetilde{Y}_j(2)$ の２乗誤差

$$S_e = \sum_{l=1}^{N} [\{Y_j(2)\}_l - \{\widetilde{Y}_j(2)\}_l] \tag{2.18}$$

N：教師信号数

$\{Y_j(2)\}_l$：l 番目の教師入力信号に対応した j 番目の中間層ニューロンの出力

$\{\widetilde{Y}_j(2)\}_l$：l 番目の教師入力信号に対応した j 番目の中間層ニューロンの評価値

を最小となるようにする線形回帰分析の手法を用いて一意的に決定できる。

抽出したニューロンが他のニューロンと緊密な線形関係にある場合，(2.17)式の $\widetilde{Y}_j(2)$ は実際の $Y_j(2)$ を良い精度で近似できるが，そうでない場合近似の精度は悪くなる。$Y_j(2)$ が他のニューロンの出力 $Y_1(2) \sim Y_{n(2)}(2)$（$Y_j(2)$ を除く）による線形式で表現できる割合は，寄与率 R_j^2 により統計的に定量化できる。寄与率 R_j^2 は，

$$R_j^2 = 1 - (S_e / S_{yy}) \tag{2.19}$$

ただし

$$S_{yy} = \sum_{l=1}^{N} [\{Y_j(2)\}_l - \overline{Y_j(2)}]^2 \tag{2.20}$$

$$S_e = \sum_{l=1}^{N} [\{Y_j(2)\}_l - \{\widetilde{Y}_j(2)\}_l]^2 \tag{2.21}$$

$$\overline{Y_j(2)} = (1/N) \sum_{l=1}^{N} \{Y_j(2)\}_l \tag{2.22}$$

で与えられる。

R_j^2 は０～１である。R_j^2 が１のとき $Y_j(2)$ と $\widetilde{Y}_j(2)$ はすべての教師信号に対して一致しており，$Y_j(2)$ が他のニューロンの出力の線形和で誤差なく記述できたことを示している。逆に R_j^2 が０のとき $Y_j(2)$ と $\widetilde{Y}_j(2)$ は無相関であり，$Y_j(2)$ は他の中間層の

出力と完全に独立な関係にあることを示している。例えば中間層ニューロン１を例にし R_j^2 の値を用いて，$Y_1(2)$ が，$Y_2(2) \sim Y_4(2)$ の線形和で記述できた割合（以下，$Y_1(2)$ の線形成分）と，記述できなかった割合（以下，$Y_1(2)$ の非線形成分）は，図２.４のように定量化できる。しかし実際には図２.３に示されるように，３つ以上のニューロンが線形関係を有している領域があるため，この部分を評価する適当なアルゴリズムが必要となる。

本論文ではこのために，以下の手順で最適ニューロン数の計算を行う。図２.５に計算手順を模式的に示す。簡単のため中間層が４つのニューロンで構成される場合を例に説明する。まず $Y_1(2)$ を $Y_2(2) \sim Y_4(2)$ の線形和で表し，寄与率 R_j^2 を求める。$1 - R_1^2$ により $Y_1(2)$ の非線形成分を求めることができ，これは図２.５のＡ部の面積で模式的に表される。次に $Y_1(2)$ を統計解析の対象から除き，残りの $Y_2(2) \sim Y_4(2)$ について同様の操作を繰り返す。すなわち $Y_2(2)$ が $Y_3(2)$，$Y_4(2)$ の線形和で記述される割合 $R^2{}_2$ を求める。このとき $1 - R_2^2$ は図２.５のＢ部の面積に相当している。さらに $Y_3(2)$ を $Y_4(2)$ の線形式で記述し，このときの $1 - R_3^2$ としてＣ部の面積を求める。最後に残った $Y_4(2)$ に関しては $R_4^2 = 0$ となり，独立分はＤ部の面積，すなわち１となる。最後にＡ～Ｄの面積を加えることにより，線形成分を排除した中間層ニューロン数（太線内の面積）を求めることができる。すなわち，この場合の最適ニューロン数は，

$$\begin{aligned} &\text{最適ニューロン数} \\ &= (1 - R_1^2) + (1 - R_2^2) + (1 - R_3^2) + 1 \\ &= 4 - (R_1^2 + R_2^2 + R_3^2) \end{aligned} \tag{2.23}$$

で与えられる。ニューロン数として意味を持つのは自然数のため，得られた最適ニューロン数を切り上げて整数化した値を中間層ニューロン数に設定する。最後にネットワークを再学習することにより，最適中間層ニューロン数を備えたネットワークが得られる。

図２.６に以上を一般化したアルゴリズムを示す。最終的に中間層ニューロンの最適

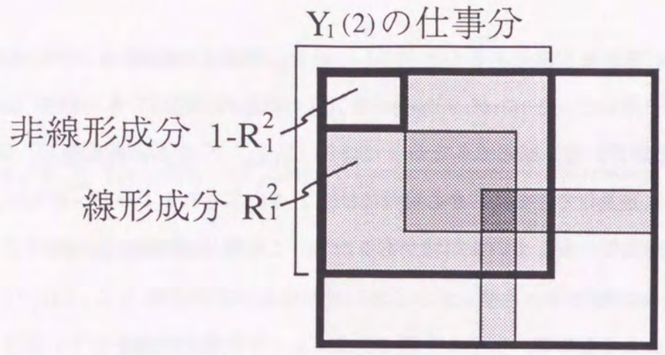


図 2.4 ﾆｭｰﾛﾝ1の線形成分と非線形成分

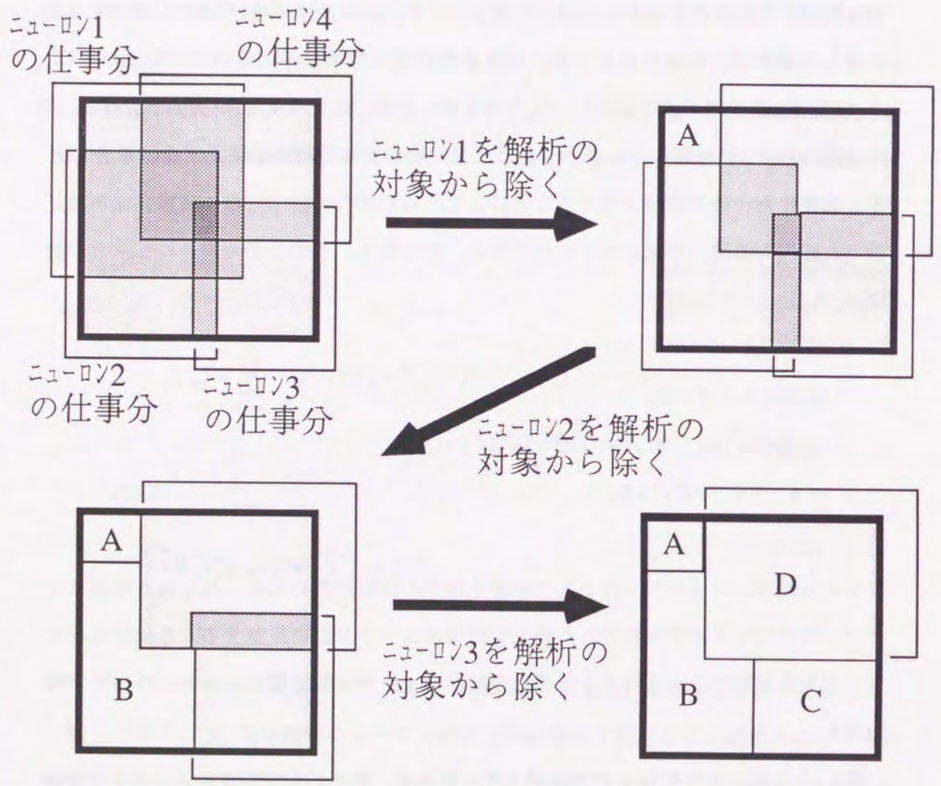


図 2.5 中間層ﾆｭｰﾛﾝの非線形成分計算方法

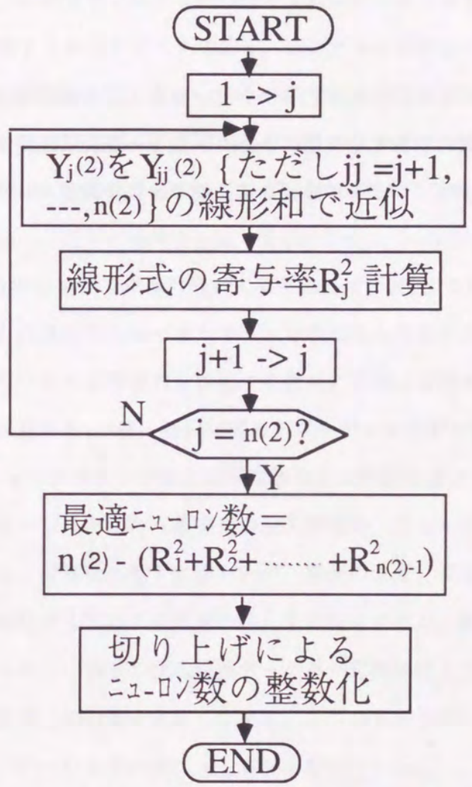


図 2.6 最適中間層ﾆｭｰﾛﾝ数決定ｱﾙｺﾞﾘｽﾞﾑ

数は，最初に中間層に備えられていたニューロン数 $n(2)$ から線形成分の総和

$\sum_{j=1}^{n(2)-1} R_j^2$ を差し引いた値，

$$n(2) - \sum_{j=1}^{n(2)-1} R_j^2 \tag{2.24}$$

により算出される。本手法によれば，ネットワークを１回予備学習する必要はあるものの，これを用いた統計解析で中間層ニューロン数を一意的に決定できる。したがって試行錯誤的なシミュレーションは必要なく，速やかに中間層ニューロン数を設定できる。

第２.４節　実験結果および考察

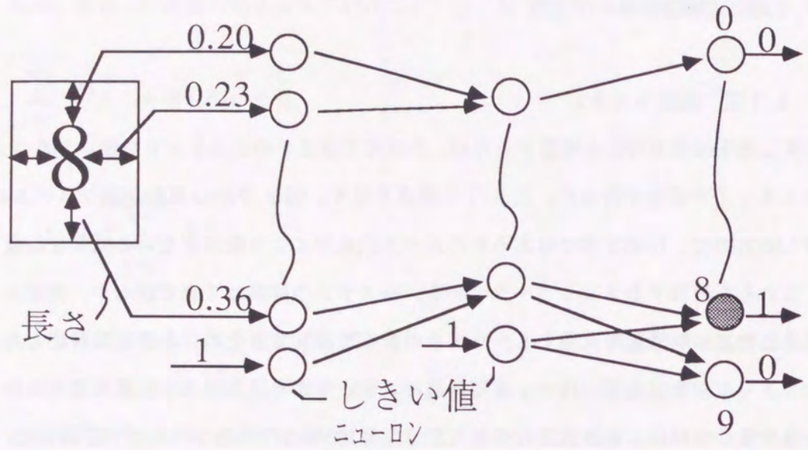
第２.４.１項　実験システム

立案した手法の有効性を確認するため，本研究では２つのシステムを対象にしたニューラルネットで評価を行った。図２.７に構成を示す。図２.７(a)は車番認識システムに適用したもので，印刷文字ではあるが汚れや光沢条件により変形を受ける数字を対象に，これらを認識するネットワークである。システムの詳細は７章に譲るが，画像から抽出した12の特徴量を入力とし，０～９の数字を識別するために各数字に対応した10のニューロンを出力層に持つ，３層構造のニューラルネットである。教師信号には実システムから抽出した各数字について20づつ計200個の代表的なサンプルを用いた。教師出力信号の値は，正解の数字を0.99，他を0.01とし，ニューラルネットの出力がこの値に対して±0.01の範囲を学習の収れん条件とした。200個の教師信号のすべてが収れん条件を満足した時点でニューラルネットを収れんと判定する。また学習により構築されたネットワークの認識能力を評価するため，同様に実機から採取した各数字について教師信号と異なる143個，計1430個のテストデータにより認識率を評価した。

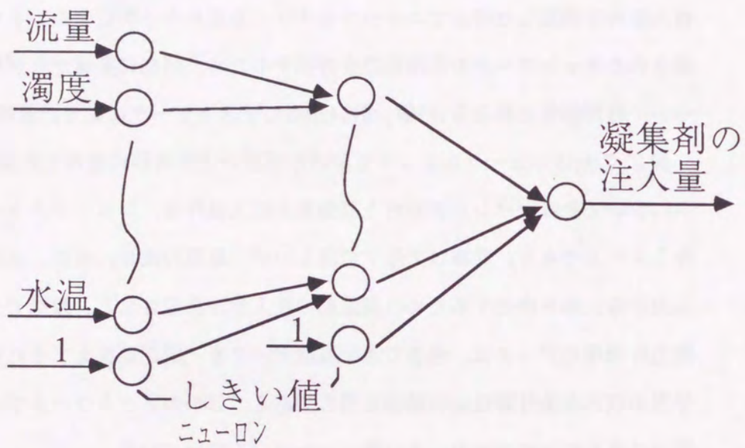
図２.７(b)はニューラルネットを非線形関数の近似問題に適用した例で，浄水プラントにおいて熟練オペレータが行う凝集剤の注入操作を，ニューラルネットで自動化したシステムである。詳細は文献７に詳しいが，原流の流量，濁度，ｐｈなど10種類の入力から，水を浄化するための凝集剤の注入量が決定される。教師データおよび汎化能力評価用のデータは，過去の運転履歴データを一定期間蓄えてそれぞれ構築した。学習の収れん条件等は(a)の場合と同じである。２つのネットワークで，入力層と中間層にはさらに１つづつのしきい値ニューロンを設けている。

第２.４.２項　中間層ニューロンの統計挙動

まず中間層において，ニューロン数が大きくなるとニューロン間の線形関係が大きくなることを示す。図２.８は図２.７(a)のシステムにおいて，学習済みのネットワークに教師入力信号を再入力したときの，中間層ニューロンの出力を統計解析した結果である。各ニューロン数について，それぞれ重みの初期値を変えて学習した３つづつの

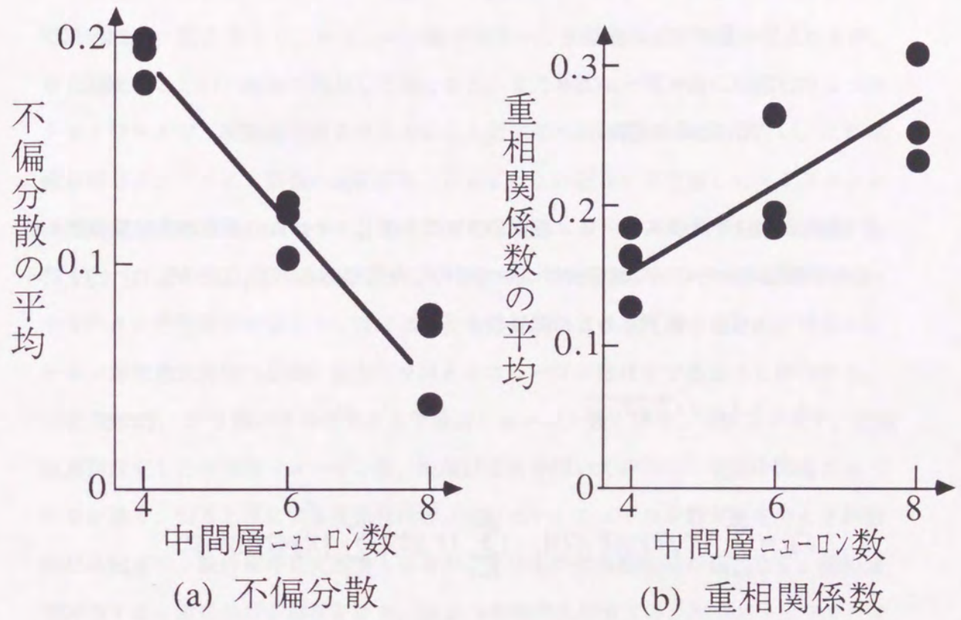


(a) 数字認識システム（クラスタリング問題）



(b) 凝集剤注入システム（非線形関数近似問題）

図２．７　シミュレーションシステム



(a) 不偏分散　　(b) 重相関係数

図２．８　中間層ニューロン数と出力統計量との関係

異なるネットワークで評価した。

図2.8(a)は統計量として不偏分散に着目し，その平均値である $\{1/n(2)\}\sum_{j=1}^{n(2)}(U_j^2)$（但し U_j は j 番目の中間層ニューロンの出力値を従属変数，他のニューロンの出力値を独立変数にした線形近似式から求めた不偏分散の値）を，中間層のニューロン数に対してプロットしたグラフである。不偏分散 U_j は残差平方和 Se と自由度 ϕ を用い(2.25)式で表せ，値が大きいことはニューロン間の線形相関が低いことを示している。

$$U_j^2 = Se/\phi \tag{2.25}$$

ただし　$\phi = N - n(2) - 2$　(2.26)

N：教師信号数

図2.8(b)は中間層ニューロンから2つを取り出し，それらの出力の重相関係数を総ての中間層ニューロンの組合せについて平均した値である。$Y_j(2)$ と $Y_{j'}(2)$（$j, j' = 1, \cdots\cdots, n(2),\ j \neq j'$）の重相関係数 $A_{jj'}$ は

$$A_{jj'} = S_{jj'} / \sqrt{S_{jj} S_{j'j'}} \tag{2.27}$$

ただし

$$S_{jj'} = \sum_{l=1}^{N} \{Y_j(2)\}_l \{Y_{j'}(2)\}_l - [\sum_{l=1}^{N} \{Y_j(2)\}_l \sum_{l=1}^{N} \{Y_{j'}(2)\}_l]/n(2) \tag{2.28}$$

$$S_{jj} = \sum_{l=1}^{N} \{Y_j(2)\}_l^2 - \{\sum_{l=1}^{N} Y_j(2)\}_l^2/n(2) \tag{2.29}$$

$$S_{j'j'} = \sum_{l=1}^{N} \{Y_{j'}(2)\}_l^2 - \{\sum_{l=1}^{N} Y_{j'}(2)\}_l^2/n(2) \tag{2.30}$$

で与えられ，数値が小さいことは同様にニューロン間の線形関係が小さいことを表している。

図2.8よりいずれの統計量においても，ニューロン数が大きくなるとそれらの動作に大きな線形関係が生じている。すなわちニューロン数の冗長度が線形関係の大きさで評価できる可能性を示している。

第2.4.3項　車番認識システムにおける最適中間層ニューロン数

まず図2.7(a)のシステムにおける中間層ニューロン数と認識能力の関係を示す。ニューロン数は4，6，8，10，12の5通りとし，同様に3つづつの異なるネットワークで評価した。図2.9より，ニューロン数が少ないとき認識率はやや低い値となるが，6個以上のとき97～98%で飽和した値となる。また各ニューロン数に対応した3つのネットワークで，初期値を変えたことによる認識率への影響はほとんどない。これは教師信号をシグモイド関数の飽和領域（0および1の近傍）で定義したクラスタリング問題では，出力層のニューロンだけでなく中間層ニューロンの大半も飽和領域で動作する[11]ためで，初期値が変化しても一定の収れん基準のもとで学習を終えたネットワークに性能面でのばらつきはほとんどない。以上より本問題の場合，“最小ニューロン数で最大汎化（認識）能力”を与えるニューロン数は6であることが分かる。

図2.10に，2.3節のアルゴリズムで最適ニューロン数を評価した結果を示す。横軸は初期設定した中間層ニューロン数，縦軸はこれを用いて評価した最適中間層ニューロン数で，切り上げによる整数化は行っていない。ニューロン数が小さいとき評価値はニューロン数に依存して大きくなるが，8以上のとき飽和した値となる。飽和値で評価すると最適値は6前後となり，図2.9の結果と符合する。同様にネットワークが異なることに起因したばらつきはほとんどない。評価値がこのような振る舞いをすることについてさらに定量的な裏付けを必要とするが，図2.10の結果は，図2.3で示した入出力関係のマッピングに必要な全体の仕事量がネットワークサイズに対して普遍的であることと対応しており，本評価手法で全体の仕事量に対応した中間層ニューロン数が決定できていることを示していると考えられる。開発手法では，十分な中間層ニューロン数（本問題では8以上）を初期設定しておけば，本手法により最適中間層ニューロン数が良好に評価できることが分かる。

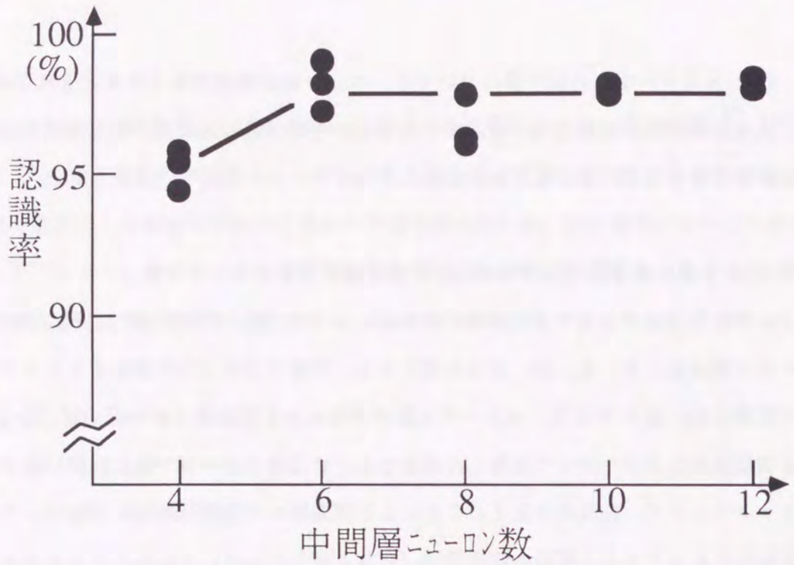


図２.９　中間層ニューロン数と認識率の関係

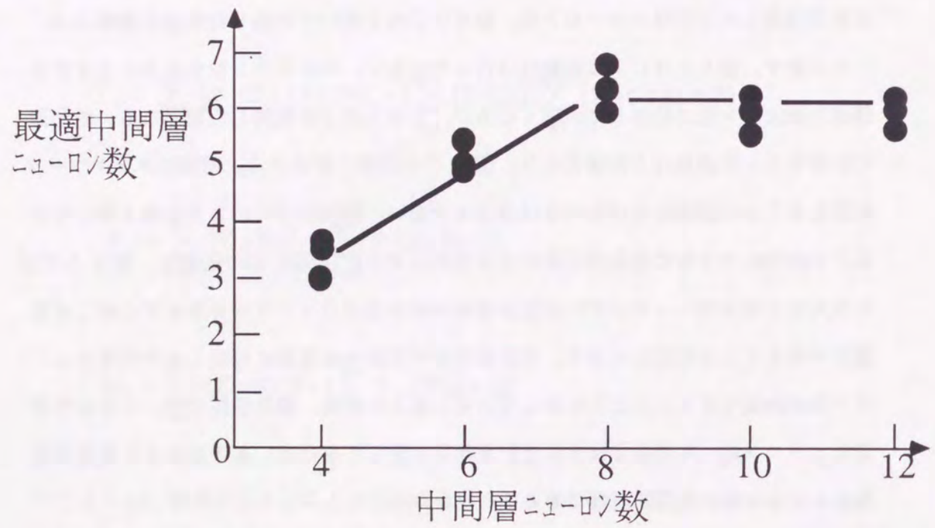


図２.１０　初期ニューロン数と最適ニューロン数の関係

第２.４.４項　浄水プラント凝集剤注入システムにおける最適中間層ニューロン数

前節で示した車番認識システムは，入力を０～９の10カテゴリに分類するクラスタリング問題であったため，出力ニューロンは飽和領域（１または０の近傍）で動作し，学習もこの領域のみを対象に行っている。これに対し図２.７(b)で示した，浄水プラントにおける凝集剤注入システムに適用したニューラルネットは，入出力の非線形関係を同定する問題である。ネットワークの出力である凝集剤の注入量は０～１の間の値で正規化された連続量であり，この点でも車番認識システムと基本的な相違がある。以下では，非線形関係の近似問題においても必要な中間層ニューロン数がニューロン間の非線形挙動として陰に表現されており，前節と同様の手法で評価できることを確認する。

まず図２.11に，中間層ニューロン数と汎化能力の関係を示す。同様に初期値を変えて学習した３つのネットワークで評価した。この場合の汎化能力は，オペレータが実際に操作した注入量と，ネットワーク出力から得た値を比較して求めた２乗誤差で定義した。２乗誤差はややばらついた値となるが，0.015～0.030のバンド幅内に整理でき，中間層ニューロン数に対する依存関係は観察されない。したがってこの問題の場合，学習を収れんさせる最小数である２が，本研究の定義に従った中間層ニューロンの最適数である。

図２.12に，初期設定した中間層ニューロン数と，これを用いて評価した最適中間層ニューロン数の関係を示す。図より，最適数は初期ニューロン数が小さいときやや小さな値となるが，４以上の初期ニューロン数を有していればほぼ一定となる。車番認識システムに適用した場合に比べ，最適数の評価値が小さいにもかかわらずネットワークによる値のばらつきがやや大きいが，この理由は中間層と出力層のニューロンがシグモイド関数全域で動作しているためと考えられる。最適数の評価値は1.5～２であり，本問題のような小規模ネットワークであれば，開発手法を有効に適用できることが確認できる。一方，多数の中間層ニューロンを必要とする大規模ネットワークの構築に本手法を適用する場合には，評価値のばらつきが適切なニューロン数の設定を妨げることも考えられる。詳細な検討は今後の課題であるが，非線形関数近似を行う大規模ネットワークに本手法を適用する場合には，評価に供したネットワークの特性等を考慮する必要があると考えられる。

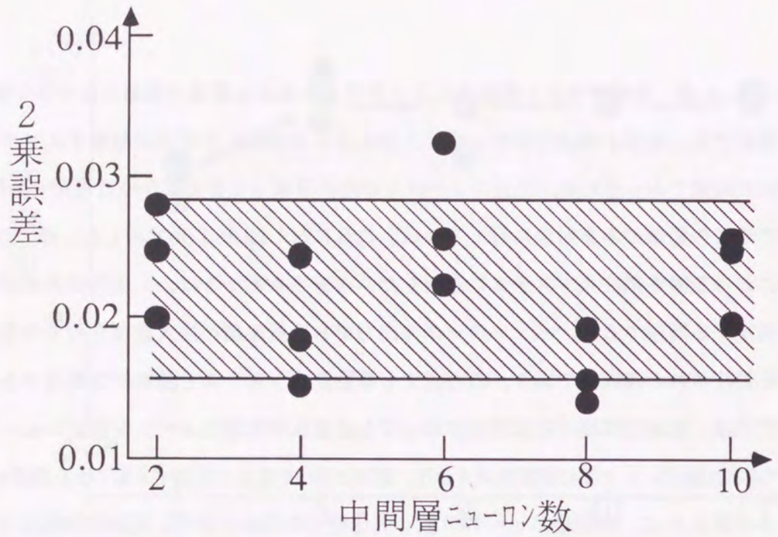


図２．１１　２乗誤差の平均値と中間層ニューロン数との関係

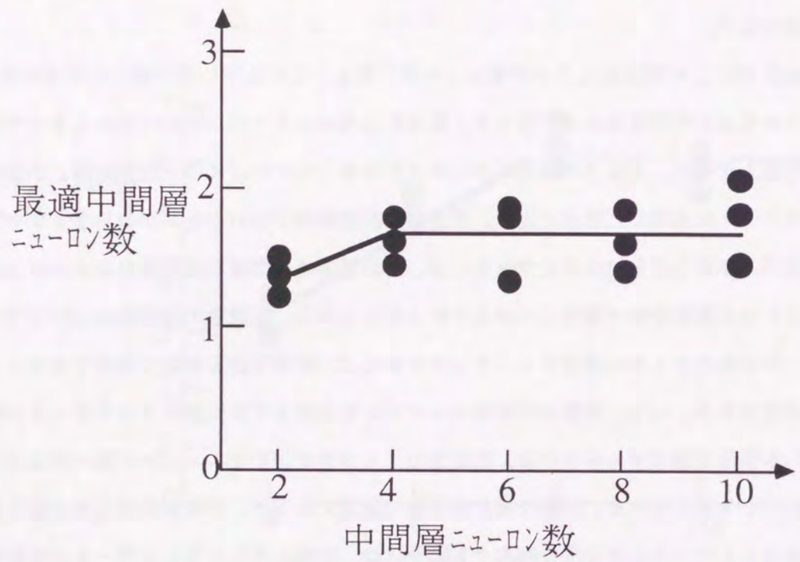


図２．１２　初期ニューロン数と最適ニューロン数の関係

３層ニューラルネットを対象に，中間層ニューロン数の最適数決定手法を検討した。その結果，十分な中間層ニューロン数を備え，学習を終えたニューラルネットに教師入力信号を再入力したときの中間層ニューロンの動作に含まれる線形成分を求め，この値に対応するニューロン数を初期設定した個数から除くことが有効なことを示した。開発手法をクラスタリング問題と非線形関数の近似問題に適用し，有効性を評価した。前者として，印刷文字ではあるが汚れや照明状態によって変形を受ける数字を認識するニューラルネットに，また後者として，浄水プラントの凝集剤注入問題に適用し，それぞれ中間層ニューロン数を６個，２個とすると，汎化能力を低下させることのない最小ニューロン数が得られることを示した。

開発手法を用いれば，中間層ニューロン数を決定するために１回の予備学習は必要になるものの，従来行われていた試行錯誤的なシミュレーションは不要となり，ニューラルネット応用システムの構築時間を短縮できる。

前述したように３層ニューラルネットが入出力関係のマッピングに十分な能力を有しており，また実システムで用いられるニューラルネットの大半は３層であることから，本章で述べた手法は大部分のニューラルネット応用システムの構築に適用可能と考えられる。一方，特殊な入出力関係を学習する場合には４層ニューラルネットが適しているという報告[11]もある。中間層が２層以上の場合に開発手法を同様に適用できるかどうかについては，学習対象となる入出力関数との関連から今後検討したい。同時に，大規模ネットワークにおける開発手法の有効性検証も今後の課題である。

2章の参考文献

（1）Q.Xue et al., "Analyses of the Hidden Units of Backpropagation Model by Singular Value Decomposition (SVD)",IJCNN'90 - WASH - DC, pp.739 - 742 (Jan. 1990)

（2）T.Ash, "Dynamic Node Creation in Back Propagation Networks", IJCNN'89 - WASH - DC, pp.623 (June. 1989)

（3）栗田，「情報量規準による3層ニューラルネットの隠れ層のユニット数の決定法」, 信学論 D-II, Vol.J73-D-II, 11, pp.1872 - 1878 (1990)

（4）和田·川人，「新しい情報量基準とCross Validationによる汎化能力の推定」, 信学論D-II, Vol. J74-D-II, 7, pp.955 - 965 (1991)

（5）萩原·戸田·臼井，「階層型ニューラルネットにおける結合重みの非一意性とＡＩＣ」, 信学論D-II, Vol. J76-D-II, 9, pp.2058 - 2065 (1993)

（6）鹿山・阿部，「汎化能力向上を目的としたクラスタリング用ニューラルネットの学習方法」, 信学論 D-II, Vol.J76-D-II, 4, pp.863 - 872 (1993)

（7）圓佛·馬場·松崎·依田，「ニューラルネットを用いたプラント運転ルールの抽出に関する研究」, 電学論D, Vol.111, 1, pp.20-28 (1991)

（8）関·合原·北井·弘津，「ニューラルネットワークによる電力ケーブルの故障診断」, 電学論D, Vol.110, 3, pp.273- 280 (1990)

（9）武長·阿部·高藤·鹿山·北村·奥山，「感度解析を用いたニューラルネットの入力層の最適化とその数字認識への適用」, 電学論D, Vol.111, 1 , pp.36-44 (1991)

(10) 芳賀·橋本，「回帰分析と主成分分析」, 日科技連

(11) S.Abe, M.Kayama, and H.Takenaga : "How Neural Network for Pattern Recognition Can Be Synthesized", 情報処理学会欧文誌(Journal of Information Processing), Vol. 14, No.3, pp.344～350 (1992)

第３章　汎化能力向上を目的とした ニューラルネットの学習方式

第３章　汎化能力向上を目的としたニューラルネットの学習方式

第３.１節　研究の概要

本章ではクラスタリング（パターン認識）問題を対象に，与えられた教師信号の下でネットワークの汎化能力を最大化するための，ニューラルネットの学習方法について検討する。クラスタリング問題における汎化能力の定義は第１章で示したように，教師信号を用いて学習したネットワークを入力信号に種々のノイズが重畳した実機環境で用いたときの認識率と対応する。

ニューラルネットの学習方法として広く用いられているバックプロパゲーション法は，最急降下法を基本とした汎用性の高い手法である一方，問題点として収れんまでに多大な学習時間を必要とすることや，局所解にトラップして大域的な最適解が得られにくいことが指摘されている。このため学習時間の短縮や局所解を回避する方法は数多く検討されている。これに対して，ニューラルネットの汎化能力との関連から学習方式の改善を論じた報告は意外に少ない[(1)~(6)]。この一因として汎化能力を評価する一般的な指標が明確でないことがあるが，主として，ニューラルネットの出力の教師出力に対する誤差（以下，出力誤差）が，汎化能力を含めたニューラルネットの性能を評価する指針と考えられて来たためと推察される。しかし実際には，出力誤差がニューラルネットの性能と直接対応しない場合がある。

一般に，ニューラルネットの適用対象は，非線形関数の近似問題とクラスタリング問題に分けられる。教師信号がすべて正しい値であると仮定した場合，学習における両者の基本的な違いは，出力誤差を小さくする操作が，前者ではニューラルネットで果たすべき本来の目的（近似誤差の最小化）と一応合致しているにの対し，後者では対象の特徴を認識し，弁別するためのヒューリスティクスにすぎない点である。したがってクラスタリング問題ではとりわけ出力誤差が小さくなったとき，この値と汎化能力の関係は不明であり，非線形関数近似の場合ほど両者が相関しないと考えられる。クラスタリング問題にニューラルネットを適用したときの“手段”（学習における出力誤差の最小化）と“目的”（実際のクラスタリング能力の最大化）の整合性の度合

いを定量化する検討が望まれている。

クラスタリングを行うニューラルネットについて，学習方式の改善により汎化能力を高める従来の試みは二通りに大別できる。一つは学習に用いる通常の誤差関数に汎化能力を高める項を追加するものである。文献1では，教師入力の変化に対する出力の変化率を出力誤差に加算し，両方を逆伝播により最小化する方式が提案されている。この手法は教師信号近傍の入力に対するネットワークの出力を安定化させるため，汎化能力向上に効果があると考えられる。しかし学習が複雑になるため，ネットワーク規模やクラスタの数が大きくなった場合収れんに時間がかかること，改善の効果が教師信号のごく近傍のデータを認識する場合に限られることが問題と考えられる。またクラスタを分割する分離超平面に着目し，これを各クラスタからできるだけ離すことを目的とした評価関数も提案されている[2]。しかし適用は単純パーセプトロンに限定されている。クラスタの境界が分離超平面の重ね合わせで形成されるためその位置を陽に特定できない3層以上のニューラルネットでは，この関数は容易に定義できない。

もう一つは教師入力信号に適当なノイズを加算して学習する方法で，文献4では認識時に重畳されるのと同程度のノイズの加算により，その環境下で最も高い汎化能力が得られるとされている。しかし大きさの特定されないノイズが重畳される環境下で汎化能力を最大にする学習方法については言及されていない。

一方，汎化能力には教師信号の量および質が大きな影響を及ぼす。言うまでもなく一般性の高い教師信号が十分に揃えられている場合には，これらを学習することで良い汎化能力が得られる。逆に教師信号が不足している場合には，単に教師信号を学習するだけでは良好な汎化能力が得られない場合がある。第1章で述べたように，実システムでは教師信号を容易に集められる場合もあるが，長時間かかったり，収集に大きな設備を必要とすることも多い。本論文で検討の対象とする車番認識システムにおいても，教師信号の採取には時間がかかる上に，収集した教師信号がどの程度一般性を有しているか判定する指標はない。したがって限られた教師信号を最大限活用して汎化能力を高める学習方法が望まれる。

これらを背景に，本章では，クラスタリング問題を対象に与えられた教師データを

用いて汎化能力を最大化する学習方法を検討する。まず通常のバックプロパゲーションで学習する限り，一定の出力誤差に到達した後さらに学習を進めて誤差を減少させても，ニューラルネットの汎化能力が向上しないことを示す。次にこれを改善するために，教師入力信号に適当な乱数を重畳することが有効であることを述べ，汎化能力向上のメカニズムを明らかにする。さらにこの知見から，与えられた教師データの基で，クラスタリング対象の特性や汎化を行う環境のノイズレベルによらず汎化能力を最大化できる乱数の印加方式を提案する。最後に一般性の高いクラスタリング問題を用いたシミュレーションで，ネットワークの出力誤差と汎化能力の相関挙動の確認、および立案した学習方法の評価を行う。

第3.2節　学習方法

第3.2.1項　クラスタの境界とネットワークの汎化能力

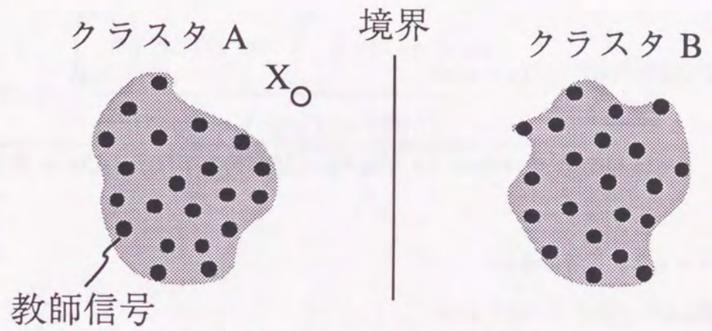
本章で対象とするニューラルネットは，図2.2と同様の3層ニューラルネットである。まずクラスタの境界とニューラルネットの汎化能力の関係について簡単に述べる。一般に，汎化能力を最大化する上で理想的な境界は，それが特定のクラスタに偏たらず，各クラスタのほぼ中間に存在する場合である[7]。図3.1に，クラスタを点，境界を直線で表し，2クラスタの簡単な場合を例に、良い境界と悪い境界の例を示す。（a）ではクラスタA，Bのほぼ中間が境界となっているのに対し，（b）では直線はクラスタAに近づいている。

ここで学習終了後に，入力Xのクラスタを判定する場合を考える。Xは教師信号で定義された領域外のため，本来どちらのクラスタに属するか判断しかねるが、空間距離からはクラスタAに属するデータにノイズが乗った可能性が高い。図3.1で，（a）ではXをクラスタAと判定するのに対し，（b）ではクラスタBと判定するため誤認識する可能性が高くなる。以上より境界線を，教師信号で与えられるクラスタ間の中央位置に近づけることが，認識率（汎化能力）の高いネットワークを得るために必要なことが分かる。同様にクラスタが超領域，境界が超曲面の場合でも，汎化能力を高めるためには，超曲面を特定の超領域に隔たらせないことが必要となる。

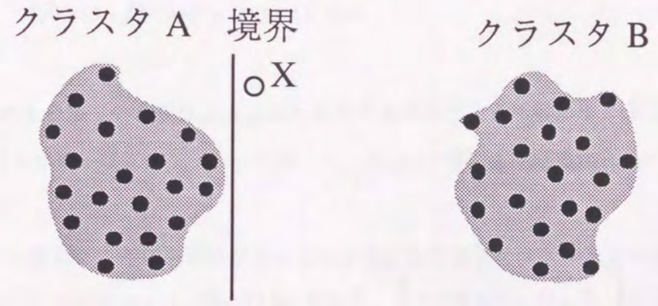
2つのクラスタを1次元上の点A，Bで代表し，同一の頻度で入力される各クラスタの信号を汎化する場合を例に，境界位置が汎化能力に及ぼす影響を定量的に示す。クラスタ間の距離を2 pとし，汎化すべき入力にはA，Bに対し標準偏差 q のガウシアンノイズが重畳している場合を考える。x はA，Bを結ぶ線分上の座標で，さらにAに対応する座標を0，Bに対応する座標を2 pとする。クラスタAに対応した入力の確率密度関数 $f(x)$は，

$$f(x)=\{1/(\sqrt{2\pi}\cdot q)\}\cdot\{exp\ (-x^2/2q^2)\}/2 \qquad (3.1)$$

同様に，クラスタBに対応した入力の確率密度関数 g(x)は，



（a）良い境界の例



（b）悪い境界の例

図3.1　クラスタ間の境界

$$g(x)=\{1/(\sqrt{2\pi}\cdot q)\}\cdot[exp\ \{-(2p-x)^2/2q^2\}]/2 \tag{3.2}$$

である。xに対応した認識率は，

（クラスタＡが正解の場合）

$$R_A(x)=f(x)/\ \{f(x)+g(x)\}$$

$$=\frac{\{1/(\sqrt{2\pi}\cdot q)\}\cdot\{exp\ (-x^2/2q^2)\}}{\{1/(\sqrt{2\pi}\cdot q)\}\cdot\{exp\ (-x^2/2q^2)\}+\{1/(\sqrt{2\pi}\cdot q)\}\cdot[exp\ \{-(2p-x)^2/2q^2\}]} \tag{3.3}$$

（クラスタＢが正解の場合）

$$R_B(x)=g(x)/\ \{f(x)+g(x)\}$$

$$=\frac{\{1/(\sqrt{2\pi}\cdot q)\}\cdot[exp\ \{-(2p-x^2/2q^2\}]}{\{1/(\sqrt{2\pi}\cdot q)\}\cdot\{exp\ (-x^2/2q^2)\}+\{1/(\sqrt{2\pi}\cdot q)\}\cdot[exp\ \{-(2p-x)^2/2q^2\}]} \tag{3.4}$$

となる。ここで境界がＡ，Ｂの中点であるpに存在した場合の，線分上の入力に対応したトータルの認識率R_pは，

$$R_p=\frac{\int_0^p R_A(x)\cdot f(x)dx+\int_p^{2p} R_B(x)\cdot g(x)dx}{\int_0^{2p}\{R_A(x)\cdot f(x)+R_B(x)\cdot g(x)\}dx} \tag{3.5}$$

である。一方，線分上の点x_0（一般性を失わず$0\leqq x_0<p$としてよい）に境界がある場合の，線分上の入力に対応した認識率は同様に，

$$R_{x_0}=\frac{\int_0^{x_0} R_A(x)\cdot f(x)dx+\int_{x_0}^{2p} R_B(x)\cdot g(x)dx}{\int_0^{2p}\{R_A(x)\cdot f(x)+R_B(x)\cdot g(x)\}dx} \tag{3.6}$$

となる。ここで，

$$R_p-R_{x_0}=\frac{\int_{x_0}^{p} R_A(x)\cdot f(x)dx-\int_{x_0}^{p} R_B(x)\cdot g(x)dx}{\int_0^{2p}\{R_A(x)\cdot f(x)+R_B(x)\cdot g(x)\}dx}$$

$$=\frac{\int_{x_0}^{p}\{R_A(x)\cdot f(x)-R_B(x)\cdot g(x)\}dx}{\int_0^{2p}\{R_A(x)\cdot f(x)+R_B(x)\cdot g(x)\}dx}$$

$$>0 \tag{3.7}$$

（なぜならば，$x\leqq x\leqq p$において，$R_A(x)>R_B(x)$，$f(x)>g(x)$）

したがって上述したように，境界がＡ，Ｂの中点にある場合に，トータルの認識率が最大となることが確認される。

第３．２．２項　学習方法の提案

次に汎化能力の高いニューラルネットを構築する手法を述べる。このためには一般的なクラスラリング問題において，各クラスタに対して平等な境界を学習によりどう作るかを考えればよい。ここで教師出力信号が表すベクトルは各クラスタを識別するために便宜上定義された値に過ぎず，理想的なクラスタ間の境界に対応しているわけではない。必然的にネットワークの出力誤差がかなり小さくなったとき，これをさら

に微小にする学習が，境界をクラスタの中央位置へ近づけている保証はない。第3.3節で示すシミュレーション結果からも，ネットワークが一定の出力誤差に到達した後さらに学習を継続して出力誤差を小さくしても，汎化能力の向上は観察されない。以上よりネットワークの出力を教師出力信号に近づける学習だけでは，理想的な境界を構築できないことがわかる。

理想的な境界に直接対応する出力ベクトルを教師出力とし，出力誤差を最小化する方向に学習を進めることができれば一番良いが，前述したように多層ニューラルネットの場合，このような出力ベクトルを実際に得るのは難しい。

一方、教師入力信号に適当な振幅の乱数を重畳する手法は容易に実現できる。この手法の場合，

$$(X_1, \cdots, X_i, \cdots, X_{n(1)})$$

の教師入力信号に対して，実際にネットワークに入力する値はその都度発生させた振幅Rの乱数を重畳し，

$$(X_1 + r_1, \cdots, X_i + r_i, \cdots, X_{n(1)} + r_{n(1)})$$
$$\text{但し} \ |r_i| \leqq R$$

となる。たとえば乱数を一様に発生させる場合を考える。従来の同一入力信号を繰り返し教える方法に対し，ネットワークに入力される信号は，教師入力信号を重心とし一辺の長さが2 Rの超立方体の内側に均等に存在する。このためクラスタの境界がこの空間内を横断する確率を低くできる。すなわちクラスタ境界が，乱数の振幅に対応した値で各クラスタから均等に遠ざかるため，乱数を重畳する学習方法は、汎化能力の向上に有効と考えられる。

課題として，最適な乱数の振幅値の決定方法がある。乱数の振幅Rとクラスタ間の距離2 pにおいて、一般に$R \gg p$であると図3.2（a）のように，入力空間に両クラスタの入力信号が重複する領域が生じる。したがって安定した境界位置が確定できな

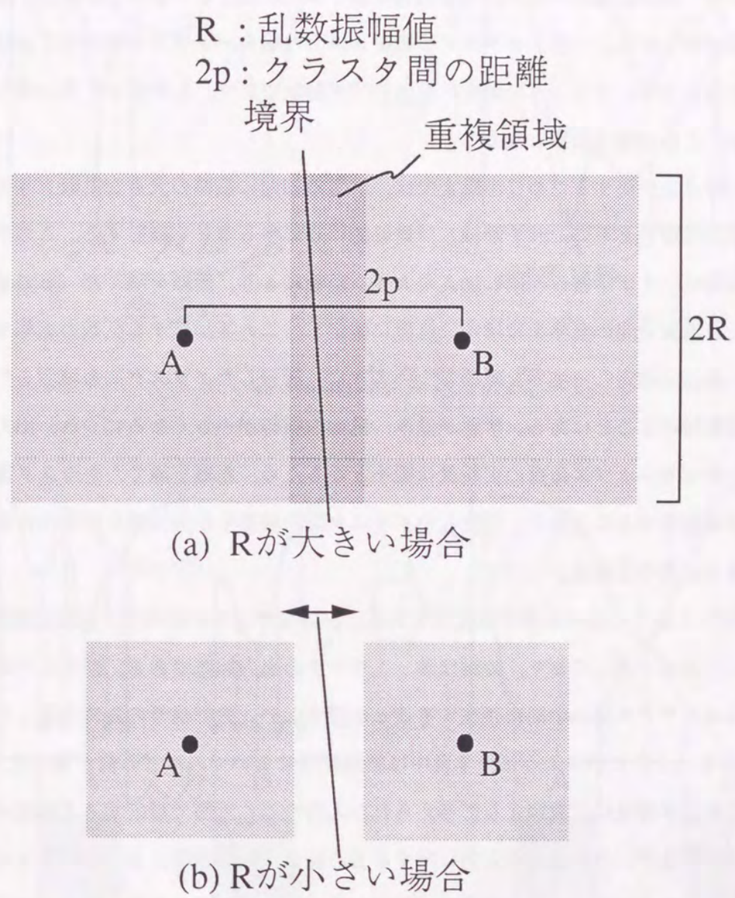


図3.2 Rとpがクラスタ境界に及ぼす影響

いか，確定に多大な学習の繰り返しを必要とする。また$R \ll p$の場合図3.2（b）に示すように，境界は教師信号によって定義されていない空間に高い自由度で存在できるため，理想位置から遠ざかる可能性が高くなる。以上より$R \fallingdotseq p$に設定するのが適当と推察される。一方，クラスタの数が3以上の場合，クラスタ同士の$2p$はさまざまな値となる。このような場合にはpの値を特定できず，したがってRの値も決定できないため問題は困難となる。

この点を解決するために本論文では，学習の初期に振幅の大きな乱数を与え，学習回数の進展とともにこれを漸減し，最終的に0にすることを検討する。この学習方法の目的は，まず学習の初期に加えた大きな乱数により，距離を隔てた（pの値が大きい）クラスタ間の境界をほぼ中央位置に確定し，この後加算する乱数の振幅を小さくし，最初に確定した境界を破壊しないように，近接したクラスタ間の境界を中央位置に位置付けることにある。学習が進み，乱数の振幅が小さくなるにつれ，出力誤差，したがってシナプス荷重の更新量は微小となるため，距離を隔てたクラスタ間の境界を大移動させることなく，近接したクラスタ間の境界を中央位置に位置付けることができると期待される。

図3.3はこの効果を表す模式図である。図は2クラスタの例で，入出力関係を仮想的に2次元で表しており，縦軸はネットワークの出力誤差である。前述したように境界は両クラスタ間の中間に存在することが望ましい。教師信号のみで学習した場合，図3.3（a）に示すように，学習中は教師信号においてのみEの値が最小化される。必然的に学習後は，教師として与えられた入力のごく近傍で対応するEの値が非常に小さくなるが，それ以外の入力に対するEの値は未知である。ネットワークの出力として得られるクラスタ間の境界は，図中に示すように対応するEの曲線が交わるところに存在するため，それが両クラスタの中間付近である保証はない。図3.3（a）のように一方のクラスタに近接する可能性があり，汎化能力低下の原因となる。

本手法では，図3.3（b）に示すように，学習中は乱数の振幅に対応した領域でEを最小化させる。すなわちネットワークの出力を同一の教師出力信号に一致させる。したがって学習結果では，この領域における$\partial E / \partial Xi$（i番目の入力に対する出力誤差の変化率），したがって$\partial Y_k(3) / \partial X_i$（$i$番目の入力に対する$k$番目の出力

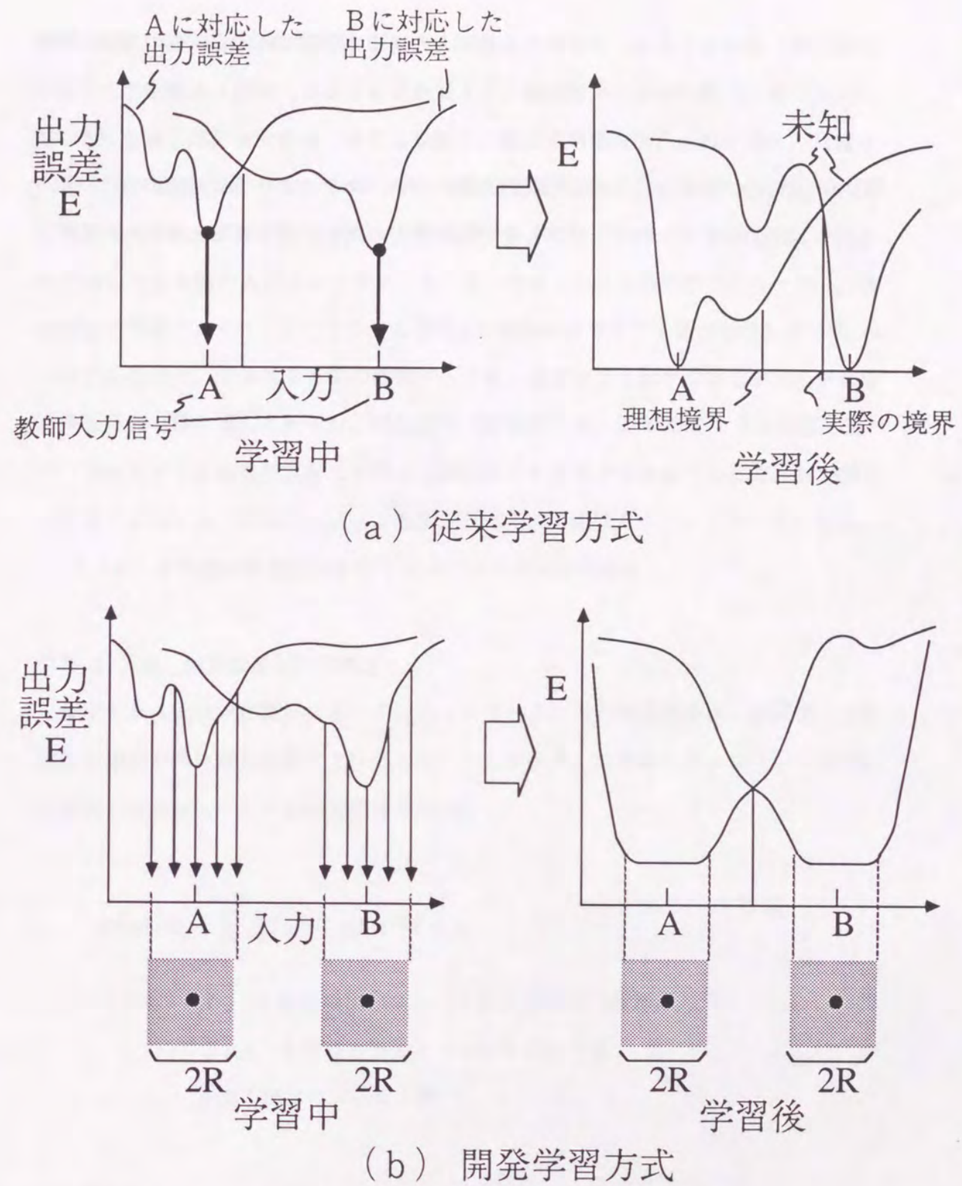


図3.3 教師信号近傍における出力誤差の模式図

の変化率）が小さくなる。すなわち各教師入力信号の近傍で対応するEの曲線が平坦になる。また乱数の振幅を学習回数とともに減少させると，前述した理由で，平坦部は教師入力信号同士の相対距離を反映した領域となる。したがって境界は各クラスタ間の中央付近に位置付けられる可能性を高められ，クラスタリング対象の特性や汎化を行う環境のノイズレベルに依存しない汎化能力の向上が得られると考えられる。

第３.３節　実験結果および考察

第３.３.１項　実験システム

図３.４に実験に用いたニューラルネットの構成を示す。（ａ）は５×５ビット平面に展開されたＡ～Ｚのアルファベットを認識するネットワークで，画像の各ピクセルに対応した２５個の入力ニューロン，Ａ～Ｚに対応した２６個の出力ニューロン，適当数の中間層ニューロンからなる。入力層と中間層にはさらに１個づつのしきい値ニューロンを設けた。学習は２６個の教師データを，各文字２０回ずつネットワークに提示する操作を繰り返して行った。汎化能力（認識率）は，図３.５のように教師入力パターンのビットを適当数反転して作った評価データを各文字あたり５０個，計1300データ作り評価した。反転したビット数を以下ハミング距離という。また（ｂ）は図２.７（ａ）と同様の車番認識を行うニューラルネットである。

第３.３.２項　出力誤差と汎化能力

まずクラスタリング問題において、ネットワークの出力誤差が小さい領域で，出力誤差の減少が汎化能力の向上と対応しないことを示す。以下のシミュレーションで出力誤差Eは各ニューロンの出力誤差を平均し、

$$E^2=(1/2)\cdot[\sum_{k=1}^{n(3)}\{T_k - Y_k(3)\}^2]/n(3) \tag{3.8}$$

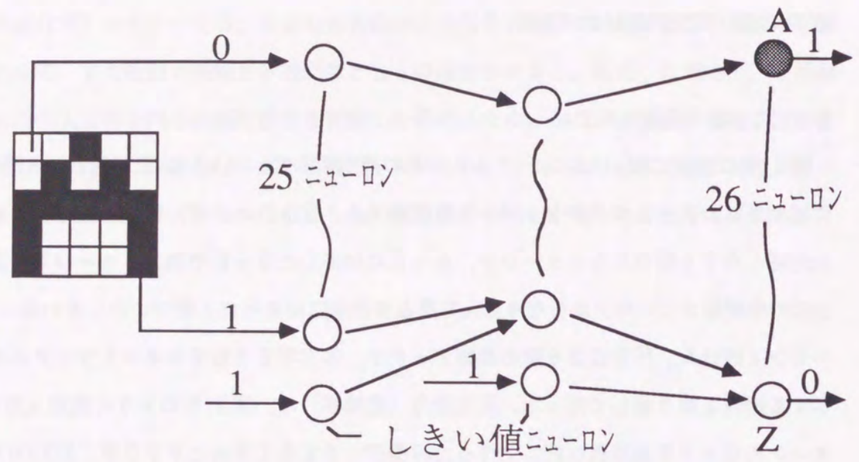
ただし　T_k：k番目の出力ニューロンに対応した教師出力

$Y_k(3)$：k番目の出力ニューロンの出力値

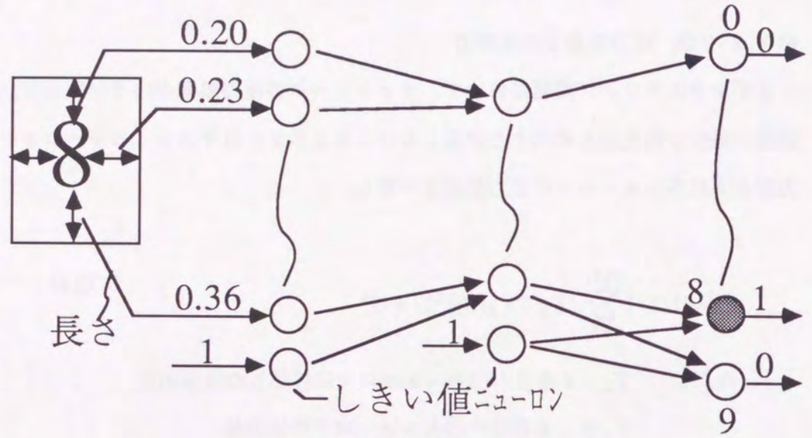
$n(3)$：出力ニューロン数

で計算する。

図３.６に図３.４（ａ）のニューラルネットで評価した，学習完了時のニューラルネットの出力誤差と認識率の関係を示す。各振幅に対して初期値を変えた３つのネットワークを作り認識率を評価した。評価データのハミング距離（反転させるビットの数）

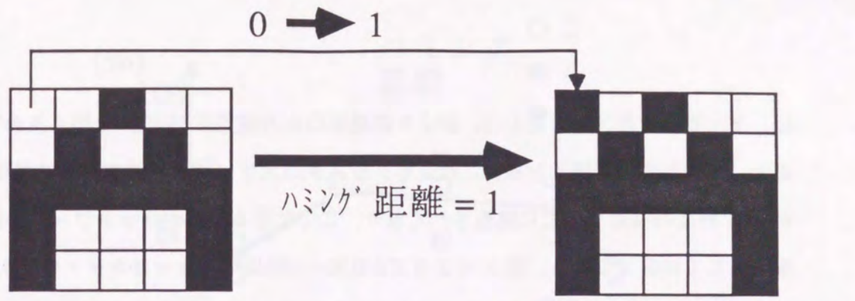


(a) 英文字認識用ニューラルネット



(b)車番認識用ニューラルネット

図3.4　実験に用いたネットワーク



(a) 教師入力信号　　(b) 評価用入力信号

図3.5　評価データ

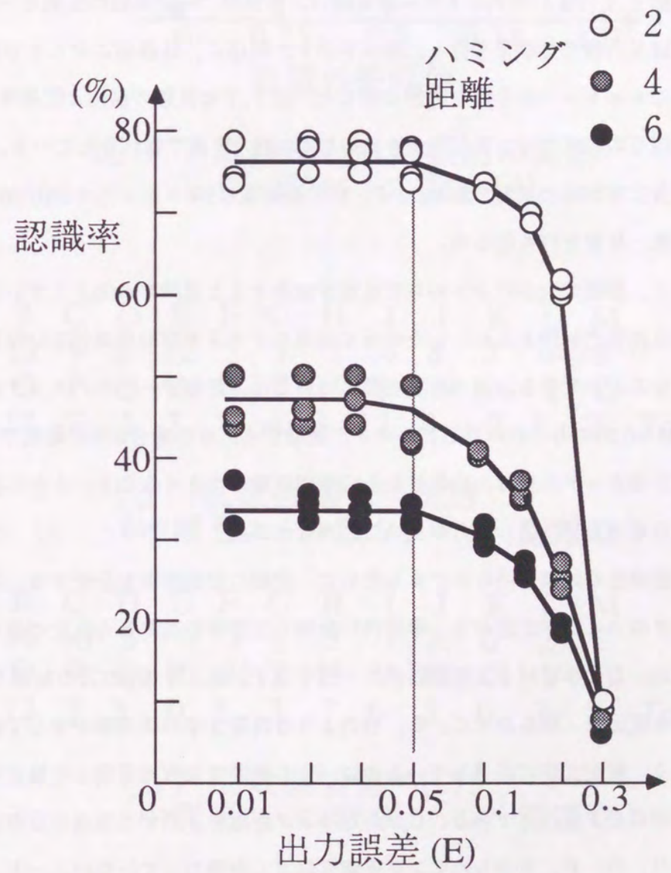


図3.6 認識率と出力誤差の関係
（中間層ニューロン数9個）

は，2，4，6の3通りとした。図より認識率は出力誤差Eが0.03～0.05より小さくなると，飽和して一定値となる。これよりクラスタの大ざっぱな形状や境界位置は，Eが0.05程度の時点ですでに構築されており，ここに至る学習がネットワークの認識率を主として決定している。図3.6よりEが0.03～0.05以下となったネットワークで，これをさらに減少させる学習を行っても認識率は向上しない。したがって過剰な精度でネットワークを学習するのは無意味であり，認識率をさらに向上させるためには学習方法自体を改善しなければならないことが分かる。

第3.3.3項　乱数重畳学習法

まず図3.4（a）のネットワークを用い，学習中，一定振幅の乱数を一貫して重畳した教師入力信号で学習を行った結果を示す。同様に，各振幅に対して初期値を変えた3つのネットワークを作り評価に供した。図3.7は乱数の振幅と認識率の関係である。振幅0の結果では，ネットワークはE＜0.01の精度で収れんしている。他は一定回数（各文字20回の学習を500セット，総学習回数は500×20×26＝260,000回）の繰り返しの後，学習を打ち切った。

図より，振幅の大小にかかわらず乱数を重畳すると認識率が向上している。乱数の重畳は汎化能力を向上させ，したがって境界をクラスタ間の中央付近に位置付けるのに有効なことが分かる。適当な振幅値については，評価データのハミング距離が2のとき0.25～0.50であるのに対し、ハミング距離が4，6の場合0.75が最良であった。これより評価データの教師入力信号からの空間距離が大きくなると，これに相当する大きな乱数を加えて学習した方が良いことが分かる。

次に振幅値の効果を明らかにするために，詳細に認識結果を分析する。図3.8に評価データのハミング距離が2，学習時に教師入力信号に加算した乱数の値が0.25，0.75のときの，各文字に対する認識結果の一例を示す。図より振幅0.75の結果で認識率が低下する原因は、明らかにC，B，Oのような特定文字の認識率が著しく低下するためである。特定文字に共通しているのは，入力空間で比較的近接した地点に，他のクラスタが存在することである。Cからハミング距離を1隔てた地点にOが存在し，BはE，G，O，P，Sからハミング距離5以下しか隔たっていない。一方，加算した

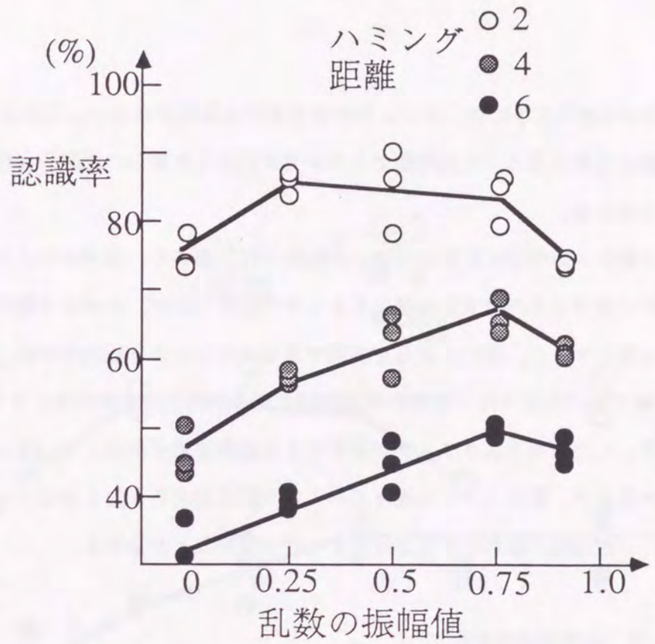


図3.7 認識率と乱数振幅の関係
（中間層ﾆｭｰﾛﾝ数9個）

A	B	C	D	E	F	G	H	I	J	K	L	M	
8	13	9	8	12	5	5	7	4	8	2	4	6	
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
11	12	7	5	5	7	4	3	12	4	0	3	2	Total 166

（a）乱数振幅値＝0.25
（中間層ﾆｭｰﾛﾝ数9，ﾊﾐﾝｸﾞ距離2）

A	B	C	D	E	F	G	H	I	J	K	L	M	
8	36	50	3	9	7	5	5	7	3	0	2	2	
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
3	12	7	3	0	5	3	7	6	3	0	2	2	Total 195

（b）乱数振幅値＝0.75
（中間層ﾆｭｰﾛﾝ数9，ﾊﾐﾝｸﾞ距離2）

図3.8 各文字に対する認識結果
(50テストデータ／文字あたりの誤認識数)

乱数の振幅が0.25のときには，このような特定文字は観察されない。これより振幅の大きな乱数を重畳すると，入力空間でクラスタが近接した場合の認識率が低下することが裏づけられる。

以上より教師入力信号に乱数を付加した効果には２面あり，境界からクラスタを遠ざけるためにはできるだけ大きな値とすることが望ましいが，クラスタ間の距離に対して過大な値とすると，逆にこれらを識別できないことにより認識率が低下する。結局第３.２節で述べたように，振幅の最適値はクラスタ同士の空間距離により相対的に決定される。したがって各クラスタがさまざまな空間距離で混在している一般のクラスタリング対象で，教師入力から種々のハミング距離にある入力をまんべんなく認識する場合，一定振幅の乱数の重畳ではうまく行かないことが分かる。

第３.３.４項　乱数漸減学習法

次に教師入力に加算する乱数の振幅を，学習の進行（学習回数）とともに漸減する手法を検討する。以下学習方法を区別する目的で，乱数を重畳しない通常の学習方法をスタティックＢＰ（ＳＢＰ），提案手法をダイナミックＢＰ（ＤＢＰ）と呼ぶ。

図３.９に漸減方法を変えて学習したニューラルネットの認識率を，図３.７と対応させて示す。いずれも学習回数260,000回の結果で，Ａは振幅を1.0から学習回数に比例して一様に減少させたもの，Ｂは学習回数の平方根に，またＣは２乗に比例した値で振幅を漸減したものである。Ｂ，Ｃの比較から，振幅の大きい乱数の効果を小さくしたとき（Ｂの場合），ニューラルネットはハミング距離の大きな評価データを認識する能力が乏しくなり，逆に振幅の大きい乱数の効果を大きくしたとき（Ｃの場合），ハミング距離の小さな評価データを認識する能力がやや劣る結果となっているが，本解析に関する限り，特定の乱数減少方法が認識率に著しく好影響を及ぼす様子は観察されない。総合的にはＡ，Ｃの結果が秀れており，評価データのハミング距離にかかわらず，図３.７の最大認識率の水準を確保している。したがってＤＢＰ法により前節の問題は解消され，評価データのノイズ値に関係なく汎化能力が最大化される。

次に図３.１０に乱数を一様減少させた場合を例に，学習回数が認識率に及ぼす影響を評価する。図より26,000～260,000回程度のかなり広い範囲で学習回数が変化しても，

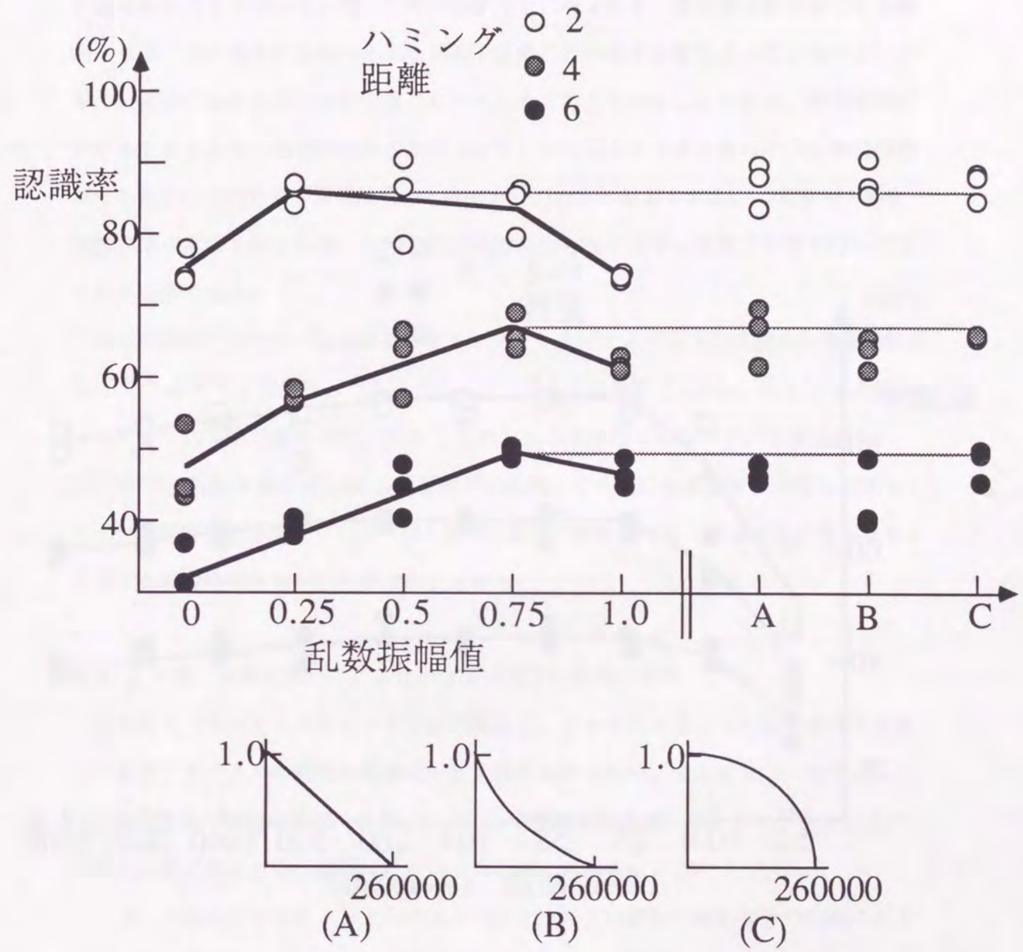


図３.９ 乱数減少方法と認識率の関係

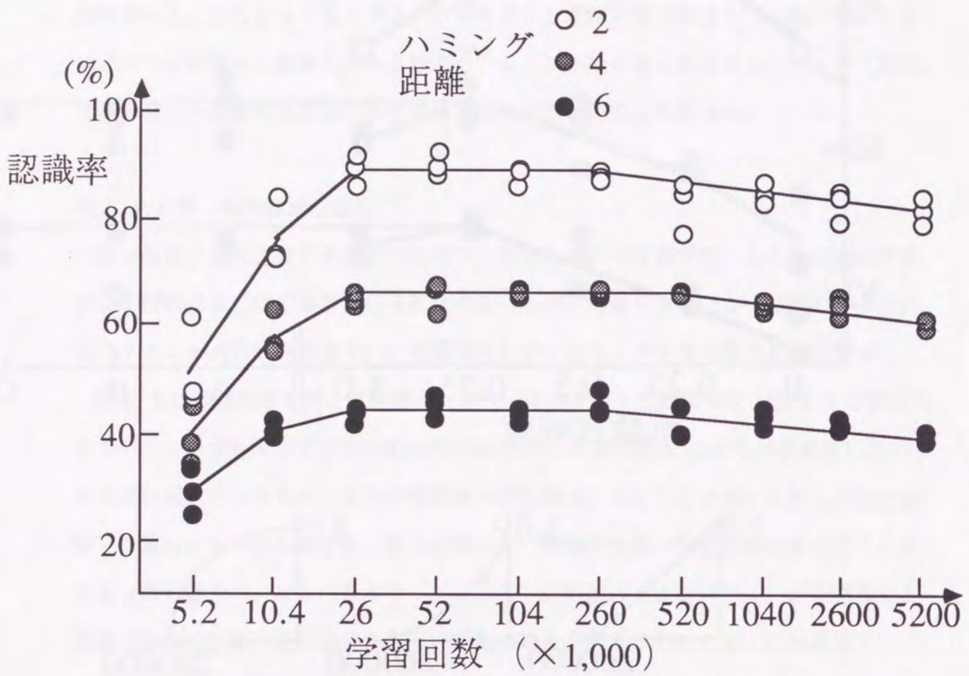


図３．１０ 認識率と学習回数の関係

認識率は安定した最大値を示している。これは学習回数設定の任意性が高いことを示すとともに，少ない回数で素早く学習を打ち切っても，ネットワークに十分な性能を与えられることを示している。学習回数がさらに増えると，認識率はやや低下する傾向となる。これは学習終期の小さな乱数を重畳した学習を必要以上に行ったことにより，全体的に乱数重畳の効果が減じられたためと考えられる。とりわけ，学習初期における大きな乱数の振幅の効果が希薄となり，いったんクラスタ間の中央近傍に位置づけられていた境界が，理想位置から遠ざかったためと推察される。本学習法の場合，適当振幅の乱数を加えた後，この効果が希薄にならないような回数で学習を打ち切ることが必要である。

次に中間層ニューロン数が異なるネットワークに対する学習方法改善の効果を示す。図３．１１はＳＢＰで学習したネットワーク（収れん条件は$E<0.01$）における中間層ニューロン数と認識率の関係，図３．１２はＤＢＰで学習した結果で，学習回数は260,000回，乱数の振幅を1.0から学習回数に比例して一様に減少させて学習したネットワークで解析した。図よりニューロン数が広範囲で変化しても，学習方法の改善はネットワークの認識率を15～20%程度向上させる。

第３．３．５項　実機認識システムにおける学習方法改善の効果

前節までで述べたアルファベット認識問題は，１クラスタを１つの教師信号で代表しており，クラスタの形状や境界は一切定義されていない。またクラスタ同士は１つの分離超平面で確実に分離できた。したがって教師信号は質，量ともに不十分であり，乱数の加算が最も有効な適用対象であったと考えられる。

一方，一般のアプリケーションでは１つのクラスタは複数の教師信号で代表されるため，ある程度クラスタの形状が明確になっているのが普通である。またクラスタの形状やクラスタ間の相対位置関係が複雑になると，各クラスタを複数の分離超平面の重ね合わせでしか分離できない場合が生じる。これらより実機における乱数加算の効果は，前節までと多少異なることが予想される。本節では車番認識を行う図３．４（ｂ）のネットワークを例に，開発手法を実システムに適用した場合の評価を行う。

図３．１３に教師信号の数と認識率の関係を示す。すべての結果は，重みの初期値を

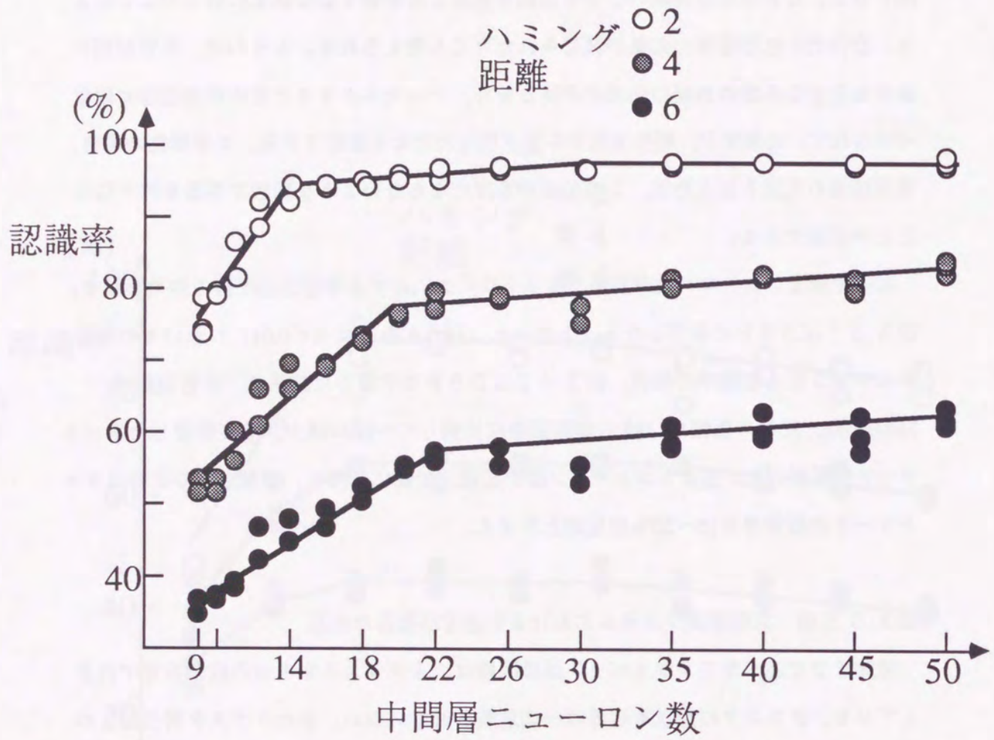


図３．１１　ＳＢＰ法で学習したﾈｯﾄﾜｰｸにおける
中間層ﾆｭｰﾛﾝ数と認識率の関係

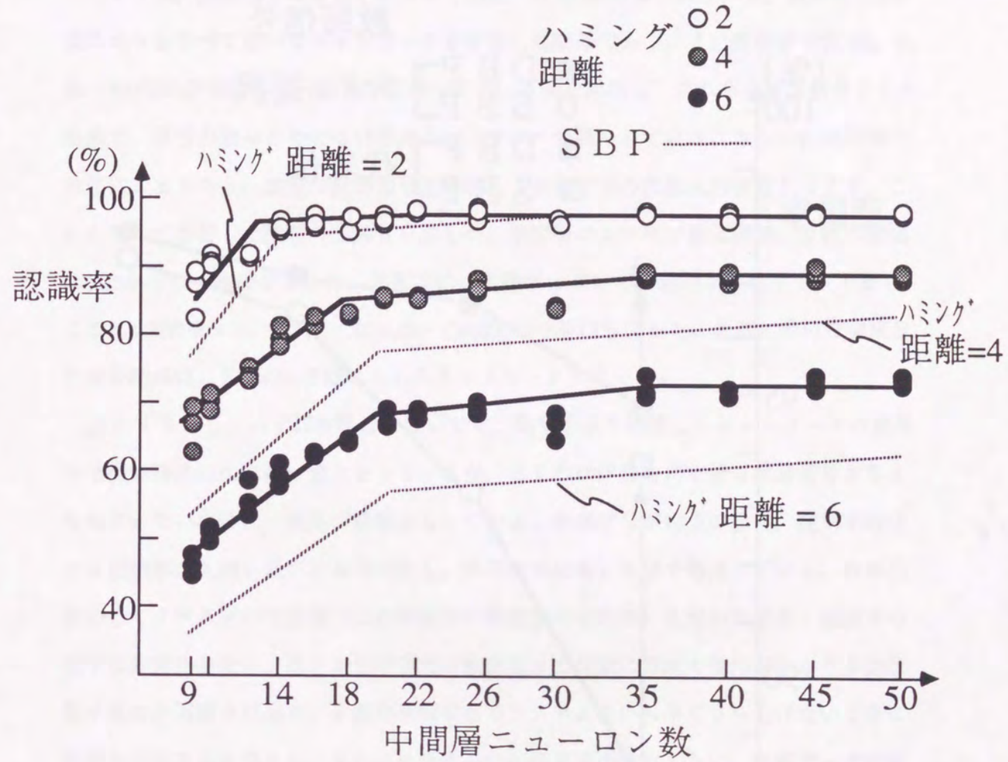


図３．１２　ＤＢＰ法で学習したﾈｯﾄﾜｰｸにおける
中間層ﾆｭｰﾛﾝ数と認識率の関係

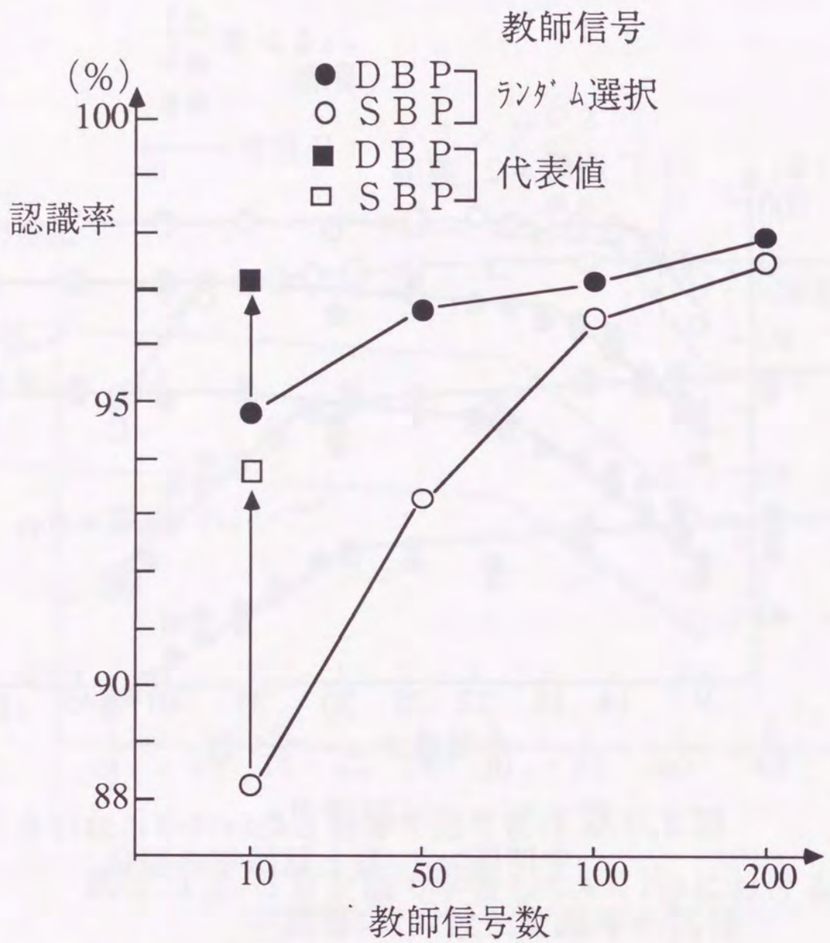


図３．１３ 認識率と教師信号数の関係
（中間層ﾆｭｰﾛﾝ数１０個）

変えた15個の異なるネットワークで評価した値を平均したものである。第２．３節で述べたように，教師信号は各数字について20個，計200個用意されている。教師信号200はこれらをすべて用いてネットワークを学習した結果である。また教師信号数100，50は，200個の教師信号から各数字について10，５個を抽出し，これらを教師信号とした結果で，信号の組み合わせは15個のネットワークで異なっている。さらに10は同様の方法によるものと，20個の教師信号を平均して各数字毎の代表入力信号をつくり，これを用いて学習した結果の２通りを示した。中間層ニューロン数は10個，乱数の振幅は0.25からの一様減少とした。学習回数は各数字について20回の学習を１セットとしてこれを500セット，計500×20×10＝100,000回の後打ち切った。比較に用いた従来ＢＰ法の結果は，Ｅ＜0.01で収れんしたネットワークを用いた。

図３．１３より，いずれの場合においても，開発手法で学習したネットワークの認識率は従来結果に比べ高い値となっているが，とりわけ学習時に十分な教師信号が与えられていない場合に，差異が顕著になっている。教師信号が10個の場合，従来学習法では認識率が大幅に低下するのに対し，本手法では高い水準を維持している。教師信号としてクラスタの代表値（この解析では母集団の平均値）を用いた場合、認識率の低下は非常に少ない。以上より本手法は教師信号の採取に時間や費用がかかり多数収集するのが困難な用途や、少数の教師信号でシステムをいち早く立ち上げたいときに有効な学習方法と言える。またノイズ含みの印刷文字認識のように，代表データが確実に定義できる場合であれば，これに対応した教師信号のみを用いた学習で，かなりの性能を発揮するネットワークを構築できる。反面、教師信号が十分に用意され，クラスタの形状が明確になっている場合には，あえて乱数を重畳しても認識率向上の効果は希薄となる。

図３．１４に中間層ニューロン数が変化した場合の結果を示す。教師信号は200個全てを用いた。中間層ニューロンが十分備えられているとき（図では６個以上），乱数を加算した結果は従来法に比べ認識率を向上させるが，中間層ニューロン数が４個のときには逆に認識率を低下させる。これは中間層が入出力信号をマッピングする上での，自由度の違いに起因していると推察される。自由度が大きいとき（ニューロン数が多いとき），乱数の重畳はシナプスの重み空間に多数ある可能な解の中で，汎化能

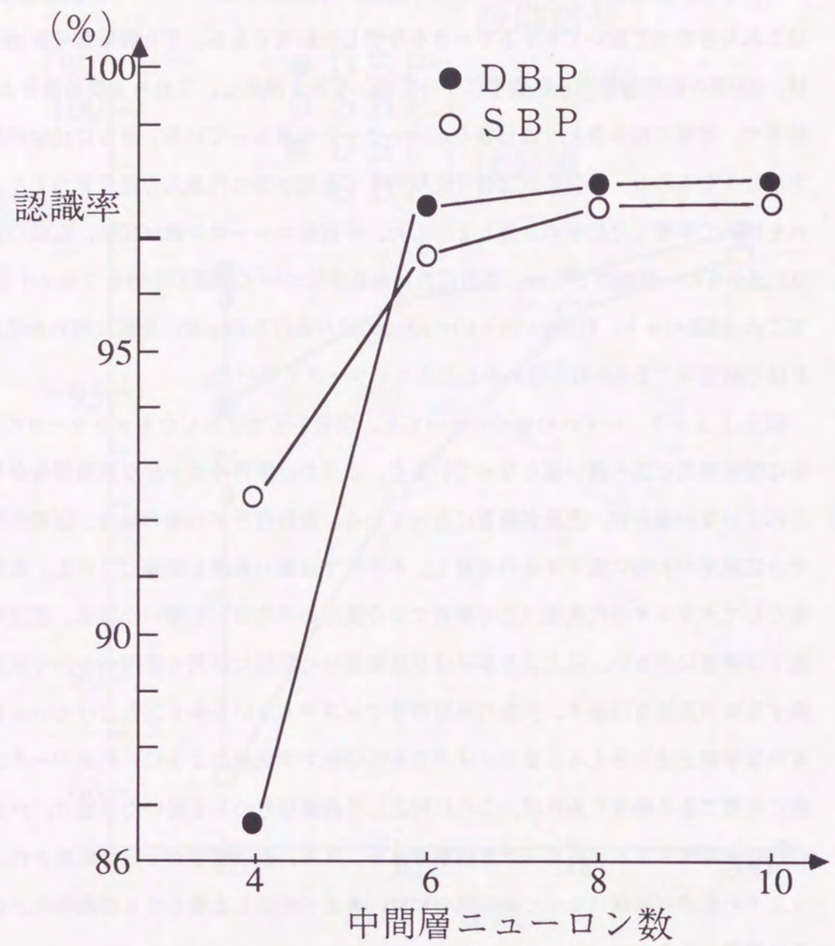


図３．１４　認識率と中間層ﾆｭｰﾛﾝ数の関係

力最大化の観点から理想に近い解にネットワークを収れんさせる。一方，自由度が小さいとき（ニューロン数が少ないとき），乱数の重畳は限られた解にネットワークが収れんするの逆に妨げることになる。本手法を適用する場合には中間層ニューロン数にやや余裕を持たせることが必要である。

第3.4節　まとめ

与えられた教師信号の基で汎化能力を最大化することを目的に，クラスタリングを行うニューラルネットの学習方法を検討した。このためには教師入力信号に適当な振幅の乱数を重畳し，クラスタの形状を見かけ上拡大することが有効であるが，とりわけ振幅値を学習の進行とともに減少させ、最終的に0にすると，クラスタ同士の相対距離や，汎化すべきデータに加わっているノイズの大きさに依存しない，高性能なネットワークが得られることが明らかとなった。開発手法をアルファベット認識問題に適用した結果，従来学習法に比べ認識率が15％～20％向上した。実機問題として開発手法を車番認識システムに適用し，認識率の向上を確認した。とりわけ教師信号が十分に揃えられていない場合に，従来学習法からの改善が顕著であり，本手法は少数の代表データのみを用いて素早く認識システムを立ち上げたい場合，あるいは多数の教師信号を収集するのが時間的，コスト的に困難な場合に特に有効であることが明らかとなった。

反面，異なったクラスタに属する教師入力信号が入力空間の一部で交わりを持つ場合のクラスタリングでは，上述した主旨から開発手法の汎化能力向上への効果は減じられると考えられる。同様に各クラスタが線形分離不能な場合等の，複雑なクラスタリング問題における開発手法の有効性評価，および汎化能力向上手法の確立が今後の課題である。

3章の参考文献

（1）Y. Kimura, "Distorted Handwritten Kanji Character Pattern Recognition by a Learning Algorithm Minimizing Output Variation", IJCNN-Seattle, 1, pp.103 - 106 (1991)

（2）大堀·西村·道見·林·渡邊，「追加学習による単純パーセプトロンの汎化」，信学論 D-II, Vol.J74-D-II, 10, pp.1472 - 1874 (1991)

（3）J. Sietsma, et. al. "Creating Artificial Neural Networks That Generalize", ISSN, Journal Neural Networks, 4, 1, pp.67 - 79 (1991)

（4）宮武·大西，「雑音環境下でのニューラルネットを用いた単語音声認識」，日本音響学会講演論文集, pp.29-30 (1991)

（5）小川，「逆問題としてのニューラルネットの理論」，信学誌, 73, 7, pp.690 - 695 (1990)

（6）H. Ogawa, J. Oja, "Optimally Generalizing Neural Network", IJCNN-Singapore, pp.2050 - 2055 (1991)

（7）S.Abe, M.Kayama, and H.Takenaga："How Neural Network for Pattern Recognition Can Be Synthesized", 情報処理学会欧文誌(Journal of Information Processing), Vol. 14, No.3, pp.344～350 (1992)

第４章　クラスタリング問題における
汎化能力の中間層ニューロン数依存性

第４章　クラスタリング問題における汎化能力の中間層ニューロン数依存性

第４.１節　研究の概要

第２章において，最適中間層ニューロン数を“最小ニューロン数で最大汎化能力”を実現する値と定義し，学習済みのニューラルネットに教師入力信号を再入力したときの中間層ニューロンのふるまいを統計解析することで，この値を決定論的に評価できることを示した。具体的には，ニューロン間に含まれる線形成分の大きさを評価し，これに相当するニューロン数を中間層に初期設定されている値から差し引けば，得られた値は初期設定数に依存せず，ニューロン数決定の指標となることを述べた。さらに線形成分を抽出する手法として線形回帰分析を再帰的に適用するアルゴリズムを開発し，シミュレーションで有効性を確認した。

しかしながら求められたニューロン数が最適値として一般的に意味を持つかどうかに関する検証は必ずしも十分でなかった。同様に，他の研究者により提案されている中間層ニューロン数決定に関するいずれの手法[(1)～(6)]においても，ネットワークの汎化能力と中間層ニューロン数の関係を明らかにした上で議論を展開した報告はない。ニューロン数の最適値が一般的に存在することの検証，および第２章で提案したアルゴリズムの妥当性確認のために，両者の関係を一般的に記述するモデルの立案が望まれる。

本章では車番認識システムのようなクラスタリング問題を対象に，ニューラルネットの汎化能力と中間層ニューロン数の関係を定量化することを検討する。まず，クラスタリング問題における中間層ニューロン数とネットワークの汎化能力の関係を明らかにし，これを記述する簡易なモデルを提案する。そして，最適ニューロン数を“ネットワークの汎化能力を最大化できる最小数”と定義することの一般性を明らかにする。とりわけ最適ニューロン数が，ニューラルネットが識別するデータに含まれるノイズの大きさに依存することを示し，評価を厳密化するためには，この効果を定量化して最適ニューロン数決定アルゴリズムに反映することが必要なことを述べる。次に開発

済みの線形重回帰分析を応用した手法を拡張し，このようなニューロン数を簡易に評価するアルゴリズムを提案する。同時に，教師信号の数が少ない場合や，教師入力信号が十分にカテゴリを網羅していないとき，第２章で提案したアルゴリズムでは評価値の精度が低下する場合があることを示した上で，これを補う手法として，乱数を重畳した教師信号を適当数生成し，統計解析用のデータベースに追加することが有効なことを明らかにする。最後にシミュレーションで，提案したモデルの妥当性，および開発したアルゴリズムで最適ニューロン数が良好に評価できることを確認する。

第４.２節　中間層ニューロン数と汎化能力

まずクラスタリング問題における，中間層ニューロン数と汎化能力の関係を検討する。問題を簡単にするために，以下の仮定をおく。

[仮定１]　入力空間（入力ニューロンの創るベクトル空間）において各クラスタは特定の単一座標で表現される。したがって任意の２つのクラスタは高々１つの超平面で分離可能である。

一般の問題で多くの場合，クラスタは形状を有した領域となるが，以下ではクラスタの形状に依存しない検討を行なう。各クラスタが凸形状の領域である限り，任意の２つのクラスタは単一の超平面で分離可能であり，このような場合，仮定１で問題の一般性が失われることはない。ここで超平面は入力空間において，中間層ニューロン数と対応した $n(2)$ 個存在し，$X_1(2), \cdots, X_j(2), \cdots, X_{n(2)}(2)$ を０とおくことにより，次式で定義できる[7]。

$$
\begin{cases}
\sum_{i=1}^{n(1)} W_{i1}(1)X_i(1)+\theta_1(2)=0 \\
\vdots \\
\sum_{i=j}^{n(1)} W_{ij}(1)X_i(1)+\theta_j(2)=0 \\
\vdots \\
\sum_{i=1}^{n(1)} W_{in(2)}(1)X_i(1)+\theta_{n(2)}(2)=0
\end{cases}
\tag{4.1}
$$

まず中間層ニューロン数が不十分なとき，ニューラルネットが十分な汎化能力を有することができないことを，模式図を用いて示す。図４.１では入力空間に，特定の座標で表現される４つのクラスタを想定する。入力空間を模式的に平面で表し，各クラスタが図のように配置されているとする。これらの座標ベクトルに対応した４つの教

師信号を用いて学習されたネットワークが，教師座標ベクトル近傍の未学習データをクラスタリングする場合を考える。

第３章で示したように，汎化能力（クラスタリング能力）を最大化するためにクラスタ間の境界線として理想的なのは，境界線が特定のクラスタに隔たらず，これらを均等に区分している場合である[8]。図４.１(a)にこのような境界線の一例を示す。学習の結果このような境界線が入力空間に形成されたとき，ネットワークの汎化能力は最大となる。

次に，このために必要な超平面（中間層ニューロン）の数について検討する。図４.１(a)の４つのクラスタは，明らかに２つの超平面で分離可能である。図４.１(b)に，これらを２つの超平面で分離する場合を示す。図より超平面の数が不十分なため，これらをどのように配置しても明らかに超平面とクラスタが近接した状態となる。したがって教師入力信号にわずかなノイズが重畳した場合でも，この入力はいずれかの超平面を越境し，他のクラスタに属する入力として認識されるため，誤認識が発生する可能性が高くなる。実際のクラスタ境界は分離超平面の重ね合わせで構築されるが，図４.１(b)より明らかなように，中間層ニューロン数が小さいときクラスタ境界位置には大きな制約が加わるため，学習方法にかかわらずネットワークに十分な汎化能力を与えられないことが分かる。一方，中間層ニューロン数が大きくなると，超平面の増加に対応してクラスタ境界位置の自由度も大きくなる。このため理想的なクラスタ境界の周辺にも学習を収れんさせる解が多く存在し，このような境界が形成される確率が高くなる。したがって中間層ニューロン数の増大は，基本的には汎化能力を向上させると考えられる。

次に，学習を完了したネットワークが汎化の対象とする入力信号に対して，以下を仮定する。

[仮定２] 汎化すべき入力信号は教師入力信号に対して，常に一定半径 r の超球の内側に存在する。

このような仮定は，特定のノイズ環境下における音声や画像の認識など，多くの実

問題と対応する。汎化対象となる信号が存在する入力空間中の領域を，汎化対象領域と称し，図４.２にこれとクラスタ境界との関係を示す。図から直感的に分かるように，この場合の誤認識は，超球が境界を越境するときに限り発生する可能性を有する。これは境界が超球の外側に位置していさえすれば，ネットワークの汎化能力には影響を及ぼさないことを意味している。

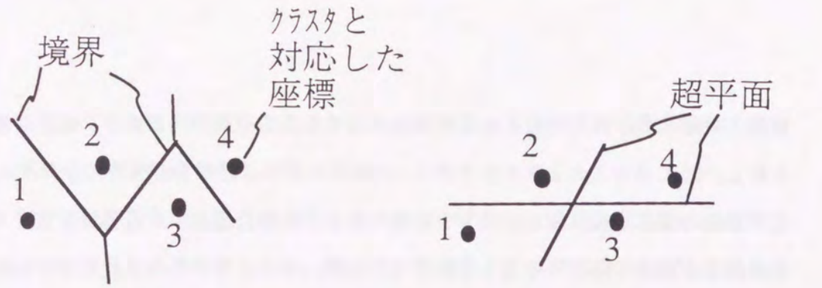
以下，超球同士に重複部分がない場合を考える。このとき，すべてのクラスタ境界が超球の外側にある状態を，クラスタ境界の理想状態と定義する。またある超平面が２つの超球の中間位置に存在するとき，この超平面は２つの超球を理想的に分離していると言う。このとき仮定１，２の基で，空間内に単連結の n 個のクラスタが存在するとき，これらの理想状態を得るのに必要な超平面の数に関して，次の定理が導ける。

[定理１] クラスタ数が n のとき，高々 $n(n-1)/2$ の超平面により，クラスタ境界を理想状態にできる。

[証明] （４.１）式より各超平面は，入力空間において任意の位置に存在できる。一方，唯一の超平面で任意の２つの超球を理想的に分離でき，このような超球のペアは入力空間に ${}_nC_2$ 通り存在する。 [証明終]

定理１より，汎化能力向上のために必要な中間層ニューロン数には上限が存在する。また上限値以下であっても，すでに多数の中間層ニューロンが備えられたネットワークでは，汎化能力に及ぼすニューロン追加の効果は小さいと考えられる。以上より，中間層ニューロン数の増加に対応して汎化能力が向上する割合はニューロン数の増加とともに小さくなり，定理１で示されるニューロン数以下の値で飽和する。

さらに以下のように考えると，実際に汎化能力を飽和させるニューロン数が r に依存することが導ける。すなわち r が大きいとき，誤認識を避けるためには図４.３(a)に示すように，境界は２つの超球の間の極めて狭い領域に存在しなければならない。この場合，各超球のペアを独立した超平面で厳密に分離する必要があり，上限付近の多くの超平面を必要とする。これは r が大きいとき汎化能力を飽和させる中間層ニューロ



(a) クラスタ境界の理想状態　　(b) 2つの超平面による境界

図４．１　クラスタの境界と超平面

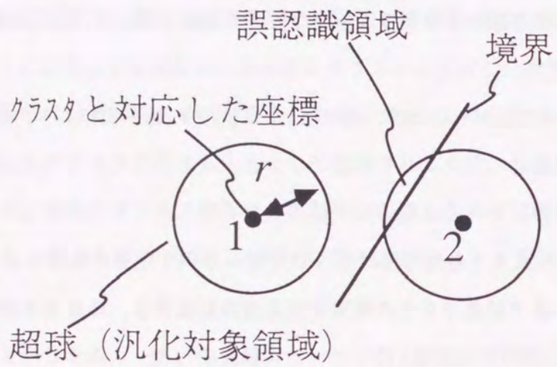
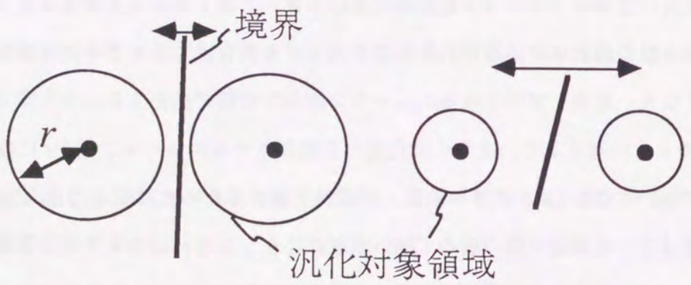


図４．２　入力空間における汎化対象領域



(a) r が大きい場合　　(b) r が小さい場合

図４．３　境界位置に及ぼす汎化対象領域の大きさの影響

ン数が大きくなることを示している。rが小さいときには，図４．３(b)のように超球間の距離が大きくなるため境界の存在制約は緩くなり，単一の超平面で３つ以上の超球を分離できる場合が多くある。したがって少数の中間層ニューロンでクラスタ境界の理想状態を形成可能であり，この場合，汎化能力を最大化するために過剰な中間層ニューロンは必要ないことを示している。これよりrが小さいとき，汎化能力を飽和させる中間層ニューロン数は小さくなる。

以上の検討結果から，仮定１，２の基で，汎化能力と中間層ニューロン数の関係は図４．４のモデルで一般化できる。モデルには２つの特徴的な挙動が存在し，

（１）汎化能力は中間層ニューロンの増加とともに増大し，やがて飽和する。

（２）汎化能力が飽和するニューロン数は，汎化の対象となるデータの r の値とともに大きくなる。

このモデルの示す挙動は第４．４節で行なったシミュレーションにより実験的に裏付けられる。また問題が仮定１，２を満足しない場合でも定性的にこれらは成立し，とりわけ(1)の挙動は，多くの研究者によりこれまで報告されている実験結果によっても確認される[9][10]。最適ニューロン数を“最小ニューロン数で最大汎化能力を実現する値”と定義した場合，図４．４より，rの値に依存して汎化能力が飽和するニューロン数 $Nopt\ (r)$ が対応する。以下，この値を評価するためのアルゴリズムを検討する。

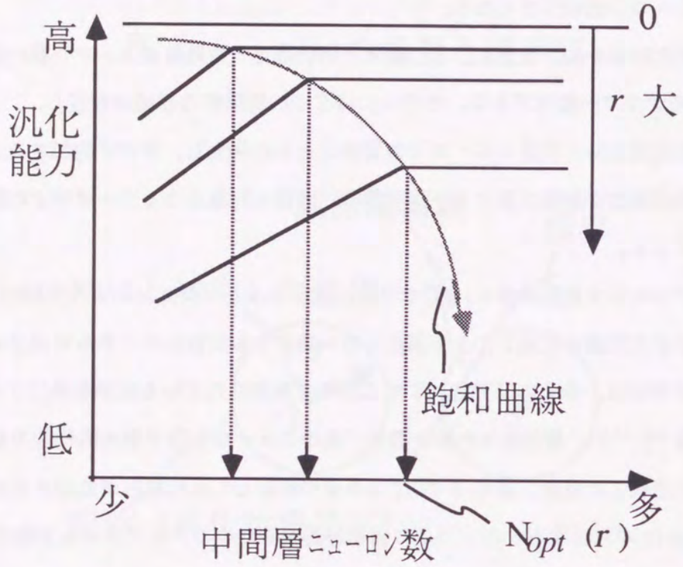


図4.4　中間層ニューロン数と汎化能力の関係模式図

第4.3節　最適中間層ニューロン数の決定方法

最適中間層ニューロン数は，収れんに十分な数の中間層ニューロン数を備え学習を終えたネットワークに，入力信号を再入力したときの中間層ニューロンの出力に着目して評価される。アルゴリズムの骨子は第2章と同様で，統計解析に用いるデータの生成時に汎化対象領域の大きさを考慮する点が新たに付加された特徴である。まず第4.2節の仮定1，2がともに成立している場合を示す。このとき本手法では図4.5に示すように，教師入力信号に加えて，新たに生成した汎化対象領域の信号をネットワークに入力し，対応した中間層ニューロンのふるまいに着目して，統計解析用のデータベースを構築する。第4.2節で述べたように入力空間全体のうち，ネットワークの汎化能力に関与するのは汎化対象領域に属する空間のみである。したがってこの空間のみに対応した中間層ニューロンのふるまいの統計量から，最適ニューロン数を決定できる。汎化対象領域の信号は，教師信号に乱数を重畳する等の手法で容易に生成できる。最適ニューロン数決定の基本アルゴリズムは第2章と同様のため省略する。

仮定2が成立していない場合には，汎化対象領域として教師信号近傍のかなり広い領域を設定することにより同様の手順で最適ニューロン数を評価できる。この場合最適中間層ニューロン数は，汎化対象領域が小さい場合に比べやや大きな値になると考えられる。また仮定1が成立しない場合には，教師信号がクラスタ全体に渡って十分に備えられていれば，教師信号が網羅する領域と汎化対象領域が近似的に等しくなるため，教師信号のみに対応した中間層ニューロンの統計量から，最適ニューロン数を高精度に評価できる。さらに教師信号が不十分な場合には，評価値は真の値より小さくなると考えられるが，汎化対象領域を想定し，この領域に属する入力信号を適当数追加すれば，評価値の精度を向上させられる。

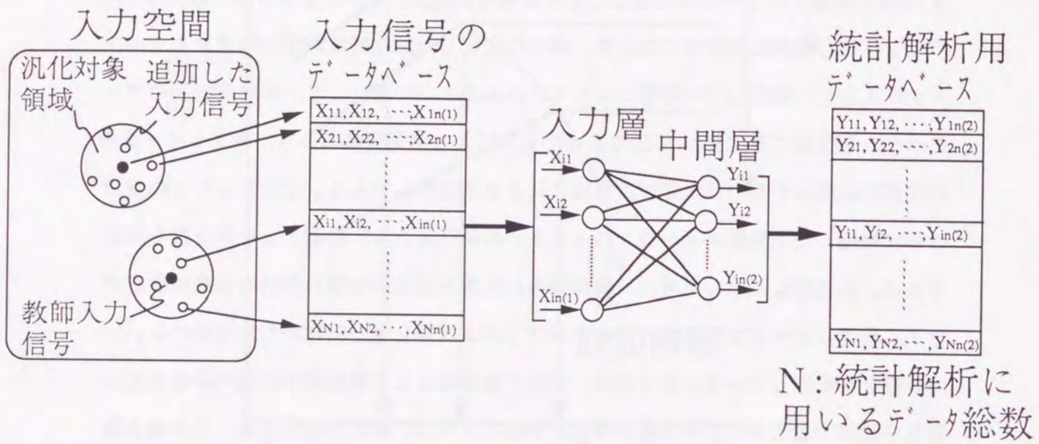


図4.5　統計解析に用いるデータの作成法

第4.4節　実験結果および考察

第4.4.1項　中間層ニューロン数と認識率の関係

シミュレーションに用いたネットワークの構成は，第3章の図3.4で示した，アルファベット認識ネットワークと車番認識ネットワークである。まず中間層ニューロン数と認識率の関係が，図4.4のモデルと一致することを確認する。図4.6に，アルファベット認識ネットワークで評価した，中間層ニューロン数と認識率の関係を示す。同一の中間層ニューロン数で，重みの初期値を変えた3つのネットワークを構築し，評価した。評価データのハミング距離は2，4，6の3通りである。図より認識率は中間層ニューロン数とともに向上し，やがて飽和する。さらに図中の飽和曲線から明らかなように，認識率が飽和するニューロン数は，評価データのハミング距離が大きくなるほど大きくなる。以上の挙動は図4.4のモデルと符合する。図4.6より本報の定義にしたがった最適中間層ニューロン数は，評価データのハミング距離2，4，6に対して，それぞれ約14，18，20である。

車番認識ネットワークについては第2章より，認識率はニューロン数が4のときやや低い値となり，6以上のとき97～98%の飽和した値となるため，最適ニューロン数は約6であることが明らかになっている。この挙動もまた提案したモデルに符合している。

第4.4.2項　最適中間層ニューロン数の評価

図3.4(a)のアルファベット認識問題では，第4.2節の仮定1，2を満足しているため，まずこの問題について検討する。学習を終えたネットワークに，教師信号に加えて汎化対象領域に属する信号を適当数ネットワークに入力してデータベースを構築する。汎化対象領域の大きさを半径ハミング距離2，中間層ニューロン数の初期値を30とし，追加する入力信号は乱数を用いて決定したビットを該当数反転して生成した。図4.7に汎化対象領域のハミング半径距離が2の場合を例に，追加した入力信号数と最適ニューロン数の関係を示す。図より，教師信号のみで評価したとき最適ニューロン数は約10であるが，ネットワークに入力する信号数を増やすとこの値は増大し，14

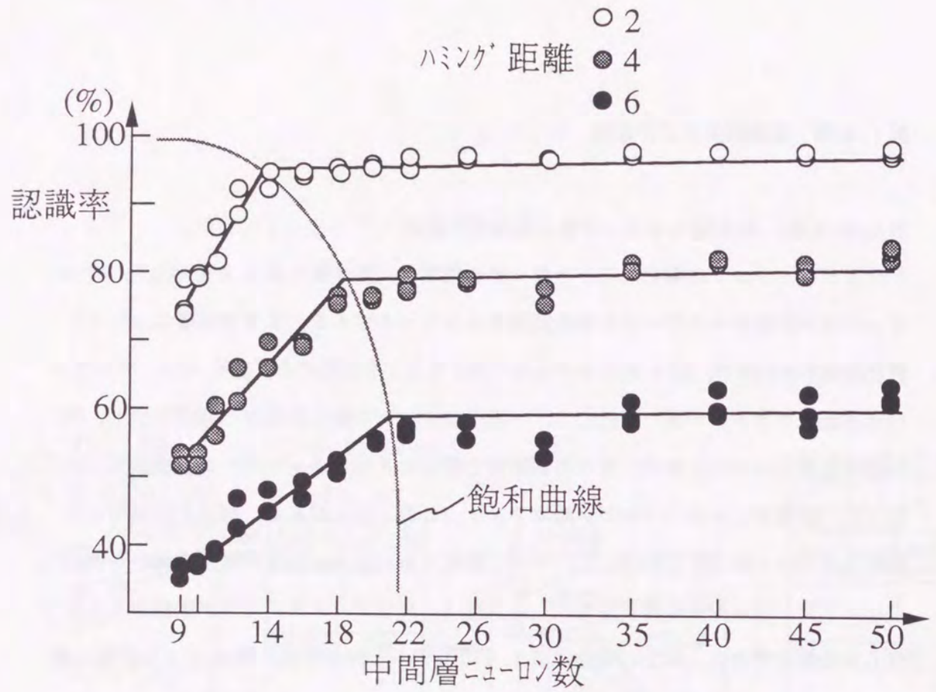


図４．６　中間層ﾆｭｰﾛﾝ数と認識率の関係

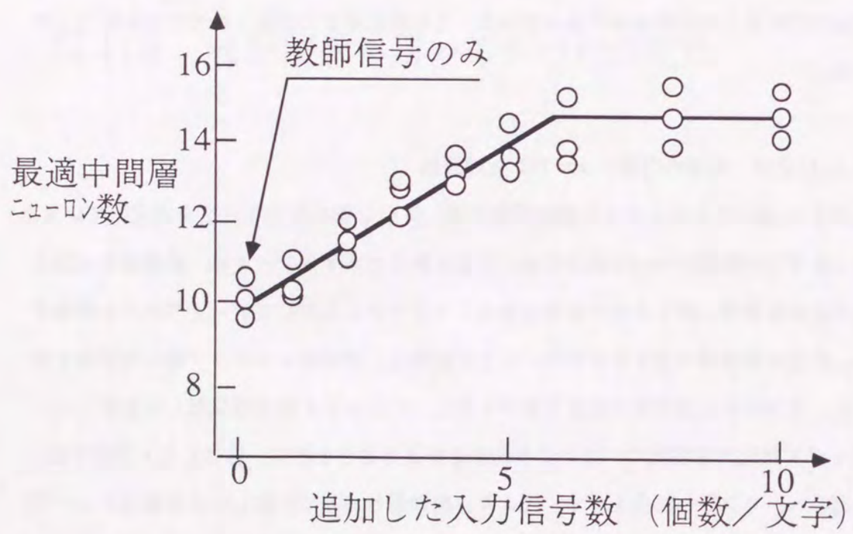


図４．７　入力信号数と最適中間層ﾆｭｰﾛﾝ数の関係
（初期ﾆｭｰﾛﾝ数３０，ﾊﾐﾝｸﾞ距離２）

程度で飽和する。結果は省略するが，この挙動は汎化対象領域の半径ハミング距離にかかわらず同一で，最適中間層ニューロン数の評価値は各アルファベットについて６～７個程度の信号の追加で飽和する。したがって，教師信号のみに対応したデータベースで統計解析を行ったとき評価値は真の最適値より小さくなり，求められたニューロン数の精度の低下が確認される。統計解析の精度からは，汎化対象領域の信号を６～７個追加して評価用のデータベースを構築すれば十分なことが分かる。以下，追加する信号の数をアルファベットあたり10個として，最適中間層ニューロン数の評価を行なう。

図４．８は，図２．６のアルゴリズムで評価した最適中間層ニューロン数を，初期ニューロン数に対してプロットしたグラフである。初期ニューロン数が小さいとき，最適ニューロン数は初期ニューロン数とともに増加し，やがて飽和する。値を飽和させる初期ニューロン数は汎化対象領域の大きさに依存して異なり，ハミング距離２，４，６に対してそれぞれ約25，30，35である。飽和値は同様に，14.4，16.9，18.0であり，図４．６より得た値を良好に近似している。したがって本方式によれば，初期ネットワークの中間層ニューロン数として十分な値を設定しておけば，最適ニューロン数が精度良く求められる。同時にニューロン数の最適値が比較的大きい場合であっても，図２．６のアルゴリズムが有効に動作することも確かめられる。

図４．９に，車番認識ネットワークの結果を示す。この問題では，仮定１，２は満足されていないが，各クラスタ（数字）について20個の教師信号が備えられており，これはクラスタを十分に網羅している。本問題の場合，入力信号を追加しなくても，最適ニューロン数を高精度に評価できることを示す。汎化対象領域は，1430の評価信号の存在領域で近似的に定義できる。(a)は第２章で示した結果で，200個の教師入力信号をネットワークに再入力し，対応した中間層ニューロンの挙動から最適中間層ニューロン数を評価した。(b)は1430個の評価信号から任意抽出した200個を加えた，計400個の入力信号で評価した。(a)，(b)で類似の傾向が得られており，最適ニューロン数は初期ニューロン数とともに増大し，やがて飽和する。飽和値では(b)の方がやや大きな値となるが，最適ニューロン数の評価値はいずれも約６である。教師信号が十分に備えられていれば，汎化対象領域を考慮しなくても，最適ニューロン数を良好に評価できることが分かる。

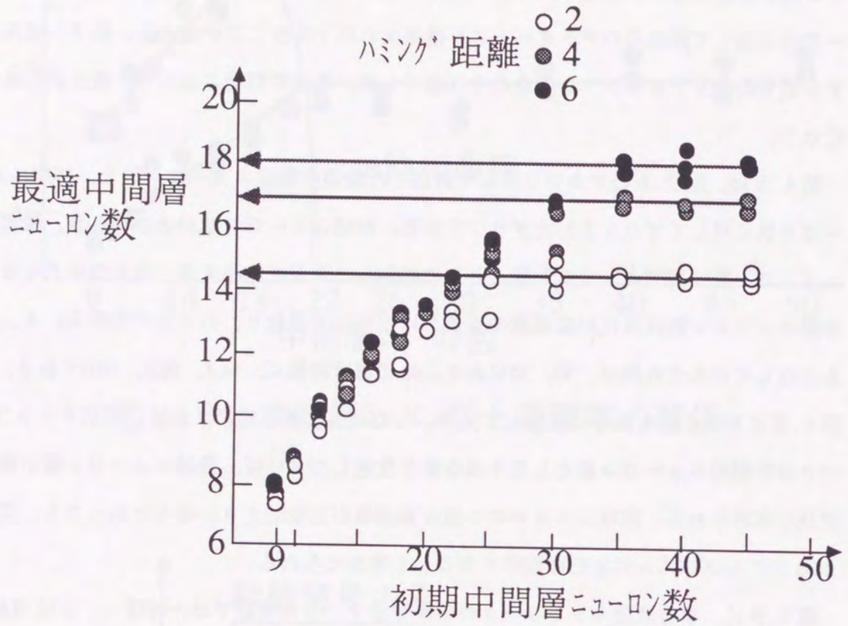
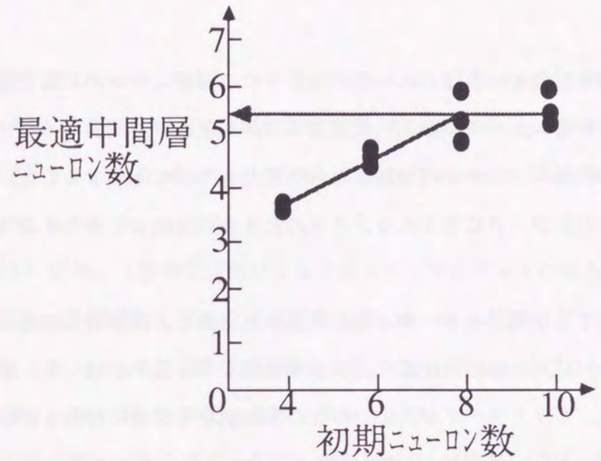
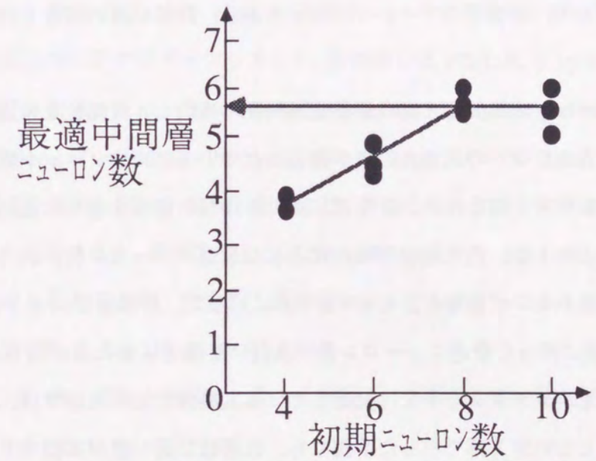


図４.８　最適中間層ﾆｭｰﾛﾝ数と初期ﾆｭｰﾛﾝ数の関係



(a) 教師信号のみによる評価



(b) 教師信号に評価信号を加えた評価

図４.９　最適中間層ﾆｭｰﾛﾝ数と初期ﾆｭｰﾛﾝ数の関係

第4.5節　まとめ

クラスタリング問題を対象に，多層ニューラルネットの中間層ニューロン数決定方法を検討した。まず中間層ニューロン数と汎化能力の関係を記述する簡易なモデルを提案し，汎化能力が中間層ニューロンの増加とともに増大しやがて飽和すること，飽和するニューロン数は汎化データに含まれるノイズの大きさに対応して大きくなることを示した。

次に汎化能力が飽和する中間層ニューロン数を最適数と定義し，提案済みの線形回帰分析を応用したアルゴリズムの拡張により，これを算定する手法を検討した。本手法では入力空間のうち，ネットワークが汎化の対象とする領域の信号に対応した中間層ニューロンのふるまいから，最適ニューロン数を決定する。とりわけ教師信号が不十分で，カテゴリおよびその近傍を網羅していない場合には，適当な大きさの乱数を重畳した教師信号を加えて，解析用のデータベースを構築し，評価結果の精度を向上させる。

開発手法をアルファベット認識問題と実機車番認識問題に適用し，有効性を確認した。教師信号として，各カテゴリの代表点のみが備えられているアルファベット認識問題では，不十分な教師信号を補うため，カテゴリあたり10個の信号を新たに追加して統計解析を行った。このとき，汎化対象領域に対応した最適ニューロン数，14, 18, 20（それぞれ領域の半径ハミング距離が2, 4, 6の場合）に対して，評価値は14.4, 16.9, 18.0であり，本報の定義に添った最適ニューロン数が良好に評価できることが明らかとなった。また教師信号がカテゴリを十分に網羅している実機数字認識問題では，汎化対象領域を特に考慮しない第2章で示した手法でも，最適数に近い値が算出できることを確認した。

4章の参考文献

（1）Q.Xue et al., "Analyses of the Hidden Units of Backpropagation Model by Singular Value Decomposition (SVD)",IJCNN'90 - WASH - DC, pp.739 - 742 (Jan. 1990)

（2）T.Ash, "Dynamic Node Creation in Back Propagation Networks", IJCNN'89 - WASH - DC, pp.623 (June. 1989)

（3）栗田，「情報量規準による3層ニューラルネットの隠れ層のユニット数の決定法」, 信学論 D-II, Vol.J73-D-II, 11, pp.1872 - 1878 (1990)

（4）和田·川人，「新しい情報量基準とCross Validationによる汎化能力の推定」, 信学論D-II, Vol. J74-D-II, 7, pp.955-965(1991)

（5）萩原，「淘汰機能を有するバックプロパゲーション」, 信学論D-II, J74-D-II, 6, pp.812 - 819 (1991)

（6）松永·中出·山川·村瀬，「階層型ニューラルネットの中間層素子を自動削減する誤差逆伝搬学習アルゴリズム」, 信学論 D-II, J76-D-II, 8, pp.863 - 872 (1993)

（7）S. Abe et al., "How Neural Networks for Pattern Recognition Can be Synthesized ", Journal of Information Procesing, Vol. 14, 3, pp.344 - 350 (1992)

（8）鹿山・阿部，「汎化能力向上を目的としたクラスタリング用ニューラルネットの学習方法」, 信学論 D-II, Vol.J76-D-II, 4, pp.863 - 872 (1993)

（9）狩野・金田・金丸，「推論型ニューラルネットによる送電線の故障区間特定」, 電学論 C, Vol.110, 7, pp.420 (1990)

（10）林・坂田・中尾・大橋，「分散法による英数字の認識能力」, 電学論 C, Vol.110, 3, pp.156 (1990)

第５章　制御モデルのチューニングを高速・高精度に行うアジャスティングニューラルネットの構成と学習方式

第５章　制御モデルのチューニングを高速・高精度に行うアジャスティングニューラルネットの構成と学習方式

第５.１節　研究の概要

ニューラルネットを制御に応用する[1]場合，適用先の多くは，非線形な制御対象の同定である。圧延材の板厚，張力，温度制御，電力供給量の制御，あるいは道路交通の信号制御等，プラント制御の多くでは，制御対象をモデル化し，これに基づいて制御対象の状態を予測したり，動作点の決定や制御指令値の算出を行う。したがって制御対象を高精度にモデル化できていればいる程，これを用いた制御の精度を高めることができる。一方，上記プラントでは制御対象の特性が種々の要因で変化するため，精度を維持するためにはこれに追従してモデルを修正なければならない。逆にこの修正がうまく行かないと，制御精度が低下し保守員の介在等を余儀なくされる場合がある。非線形な制御対象を適応的に同定する手法として，自己組織化機能を有するニューラルネットが注目されている理由の一つである。

制御対象の同定にニューラルネットを用いる場合，制御対象の順方向あるいは逆方向の入出力関係を直接ニューラルネットに蓄え，モデル化する[2][3]手法が一般的であるが，制御対象のダイナミクスを記述する微分方程式等が先見的に得られている場合には，これに含まれる未知パラメータにモデル誤差を集約し，これをチューニングすることで，モデルの精度を高める手法の方が，学習の効率化，モデル誤差低減の点で優れている。

本章では以下，後者を検討の対象とし，モデル誤差とパラメータ修正量の関係を獲得したニューラルネットを用いてモデルチューニングを高速，高精度に行う手法を検討する。ここで扱うニューラルネットは，実際に制御対象から検出した制御量とモデルを用いて算出した値との差分と，これに対応した望ましいパラメータ修正量の関係を近似しており，差分を検出した場合にはニューラルネットの出力にしたがってモデルのパラメータを修正する。一般に，差分と最適なパラメータ修正量の関係は非線形

であるため，適切な修正量の算出は容易でない。実際のプラントでモデルチューニングを自動化する場合には，両者の関係を本来の関係より緩やかな勾配に設定した安全な線形式で近似し，チューニングを行うことが多いが，収れんまでのチューニング回数が多大となることが問題となる。

本章ではモデルチューニングの中でも，差分と最適なパラメータ修正量の関係に制御対象の状態量が多様な影響を及ぼすため，適切なパラメータの修正が困難とされていた複雑なチューニングを良好に行なう手法を開発する。このような高度なモデルチューニングが必要なプラントは，本章の中で述べる加熱炉（温度誤差と総括熱吸収係数の修正量の関係に，各炉帯の温度や鋼材の挿入温度，板厚等が影響を及ぼす）の他にも，温度制御や各種化学プラントの反応速度制御等で一般的に存在する。本章では，まずこのような制御モデルを従来のニューラルネットを用いてチューニングした場合，目標点近傍の出力誤差により，チューニング精度が低下することを示す。次にこれを解決するために従来のニューラルネットを拡張したアジャスティングニューラルネット（以下，ＡＪＮＮ）を提案する。そしてこれを用いたモデルチューニングシステムの構成，およびＡＪＮＮの性能を最大化する学習方式を検討する。最後に熱間圧延における加熱炉温度制御を模擬したシミュレーションで，開発手法の有効性を検証する。

第５.２節　アジャスティングニューラルネットの構成

第５.２.１項　加熱炉制御システムとチューニングの問題点

以下，熱間圧延システムにおける加熱炉プラントを例に説明する。加熱炉の役割は，圧延工程に先立ってスラブと呼ばれる鋼材を目標温度に昇温させることである。近年，板の品質を含めた高精度な圧延を実現するため，加熱炉から抽出される鋼材の温度を目標値に対して高精度に一致させる要求が高まっている。図５.１に加熱炉と，これを制御する加熱炉制御システムの構成を示す。加熱炉は通常，複数の炉帯（図５.１では予熱帯，第１加熱帯，第２加熱帯，均熱帯）からなり，各炉帯はバーナで高温に保たれている。加熱炉制御システムは，炉温指令値算出部，鋼材モデル，パラメータ修正部により構成される。炉温指令値算出部では，鋼材の初期温度，材質，熱特性等のパラメータを用い，鋼材モデルを用いて鋼材を目標温度に昇温にするのに必要な炉温指令値を算出する。しかしながら鋼材モデルに誤差があると，目標温度と実際に抽出される鋼材の温度が一致しない。パラメータ修正部では温度誤差と，制御対象の状態量（炉温や板厚等）を取り込み，鋼材モデルのパラメータを修正することで，鋼材モデルを実際の特性と一致させる。鋼材モデルは，以下の熱輻射，熱伝達方程式で記述されることが知られている[4]。

熱輻射方程式

$$Q = 4.88\Phi_{cg}\ [\{(T+273)/100\}^4 - \{(\theta_s+273)/100\}^4] \tag{5.1}$$

熱伝達方程式

$$(c_s\rho/k)(\partial\theta/\partial t) = (\partial^2\theta/\partial x^2)+(\partial^2\theta/\partial y^2)+(\partial^2\theta/\partial z^2) \tag{5.2}$$

但し，Q：雰囲気中から鋼材が吸収する熱量

Φ_{cg}：総括熱吸収係数（熱伝導係数）

T：炉温

θ_s：鋼材の表面温度

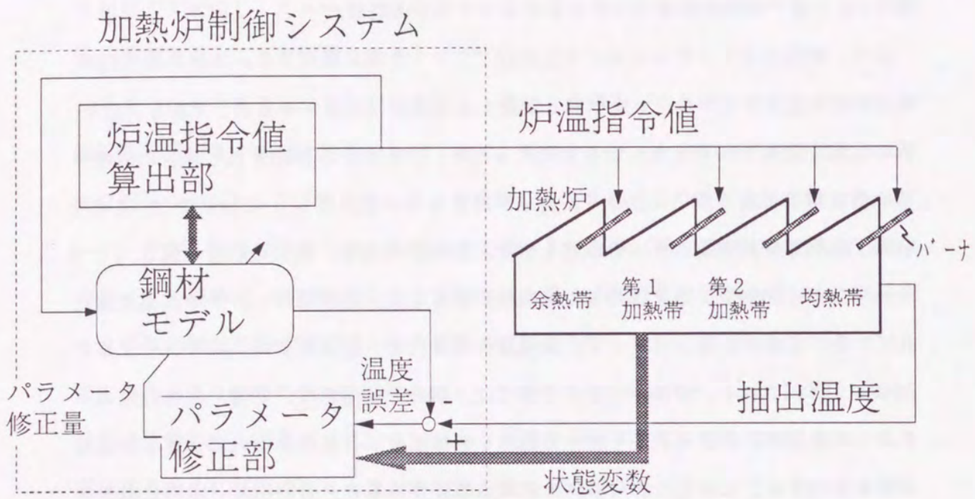


図5.1　加熱炉制御システムの構成

θ：鋼材の内部温度

c_s：比熱

ρ：密度

k：熱伝達係数

加熱炉における鋼材の昇温が(5.1)，(5.2)式で示した熱方程式に従って行なわれると仮定すると，鋼材モデルの誤差は総括熱吸収係数Φ_{cg}，熱伝達係数k，比熱c_sが実際とどの程度隔たっているかに集約される。これらのうちkおよびc_sは，さまざまな温度において繰り返し実験された実測値がデータベース化[4]されており，このテーブルを逐次参照して値を決定しているため，一定の精度が保証されている。したがって本論文では，鋼材モデルの誤差をΦ_{cg}の誤差が代表していると仮定して以下の検討を進める。Φ_{cg}は炉の構造特性や，操業状態により変化する炉温に依存して変化するため，あらかじめ精度良く算出することは困難である。このためパラメータ修正部で，実際の加熱炉から得た鋼材の抽出温度をモデルを用いて算出した値と比較し，この偏差からΦ_{cg}を修正する。

本チューニングを困難にしている要因は，温度誤差（$\delta\theta_{out}$）とパラメータ修正量（$\delta\Phi_{cg}$）の関係が非線形である上に，この関係に制御対象の状態量が複雑な影響を与える点である。このため温度誤差から簡単な線形式等でパラメータ修正量を算出する従来手法では，線形式の勾配により，チューニングが発散したり，収れんまでに多大なチューニング回数を要していた。したがって良好なチューニングを行うためには，非線形な関係を高精度に近似する手法でパラメータ修正部を構築する必要がある。関数近似の手法はいくつかあるが，チューニングの結果蓄えられる$\delta\theta_{out}$と望ましい$\delta\Phi_{cg}$の関係を追加的に用いて，近似式を高精度化したり，プラントの特性変化に伴う$\delta\theta_{out}$と$\delta\Phi_{cg}$の関係の変化に近似式を追従させる目的で，ニューラルネットの自己組織化機能に着目した。以下ではチューニング問題との対応から，ニューラルネットの出力を－1～1で正規化して考える。すなわち$\delta\theta_{out}$が0でΦ_{cg}に修正の必要がない場合にニューラルネットは0を出力し，0～1は正の修正量，－1～0は負の修正量と対応する。

ここでプリミティブなニューラルネットを鋼材のモデルチューニングに用いると，次の問題が生じる。ニューラルネットの出力を

$NN_{out}(\delta\theta_{out}, x_1, x_2, \cdots, x_n)$

ただし，$x_1 \sim x_n$は$\delta\theta_{out}$と$\delta\Phi_{cg}$の関係に影響を及ぼすプラントの状態変数

加熱炉プラントでは鋼材の板厚や各炉帯の炉温等に対応する

とすると，$\delta\theta_{out}$が0のときΦ_{cg}の修正量は0であることから，近似された非線形関数は明らかに，

$$NN_{out}(0, X_1, X_2, \cdots, X_n) = 0 \tag{5.3}$$

ただし，$x_1 \sim x_n$は任意

を恒等的にに満足する必要がある。すなわちニューラルネットの入出力超曲面は，入出力ベクトル空間の２つの超曲面である，

$$\delta\theta_{out} = 0 \tag{5.4}$$

$$NN_{out}(\delta\theta_{out}, x_1, x_2, \cdots, x_n) = 0 \tag{5.5}$$

の重ね合わせによる構成される，次式の超曲面を包含しなければならない。

$$\delta\theta_{out} = NN_{out}(\delta\theta_{out}, x_1, x_2, \cdots, x_n) = 0 \tag{5.6}$$

しかしながら(5.6)式の超曲面上の総ての点を教師信号で定義することは不可能なため，学習をどれほど高精度に行ったとしても，(5.3) 式に対して実際の出力は誤差を含んでいる。一方，モデルチューニングの精度を決定するのは，チューニングが収れんする

領域である(5.6)式近傍のニューラルネットの出力精度であり，この領域の出力誤差は，チューニングの定常誤差と直接対応する。以上より，ニューラルネットを本問題のようなモデルチューニングに用いる場合，(5.6) 式近傍の出力精度を高める必要がある。以下，この問題を解決するために従来の多層ニューラルネットの拡張を検討した内容を次節で述べる。本論文で提案する拡張ニューラルネットを以下，アジャスティングニューラルネット（ＡＪＮＮ）と呼ぶ。

第５.２.２項　アジャスティングニューラルネットの構成と入出力曲面

図５.２にＡＪＮＮの構成を示す。ＡＪＮＮは通常のニューラルネット（通常ＮＮ）に，これと同一の構成，重みを有した誤差算出用ＮＮを並列に設けたことが特徴である。ＡＪＮＮは，Φ_{cg}の真の値に対応したニューラルネットの出力誤差が，プラントの状態変数が既知であればその都度正確に算出できることに着目して開発した。すなわち$\delta\theta_{out}=0$に対応したニューラルネットの出力誤差は$NN_{out}(0, x_1, x_2, \cdots, x_n)$で与えられる。したがって$x_1 \sim x_n$が分かっていれば，誤差の値を正確に計算することができる。ＡＪＮＮでは，$NN_{out}(\delta\theta_{out}, x_1, x_2, \cdots, x_n)$（プリミティブなのニューラルネットの出力）に対して，図５.２のように$NN_{out}(0, x_1, x_2, \cdots, x_n)$（誤差算出用ニューラルネットの出力）を並列に設け，$NN_{out}(\delta\theta_{out}, x_1, x_2, \cdots, x_n)$から$NN_{out}(0, x_1, x_2, \cdots, x_n)$を減じた値を最終的な出力とする。すなわちＡＪＮＮの出力$AJNN_{out}(\delta\theta_{out}, x_1, x_2, \cdots, x_n)$は，

$$AJNN_{out}(\delta\theta_{out}, X_1, X_2, \cdots, X_n) = NN_{out}(\delta\theta_{out}, X_1, X_2, \cdots, X_n) - NN_{out}(0, X_1, X_2, \cdots, X_n) \tag{5.7}$$

で与えられる。第１項は通常ＮＮの出力，第２項は誤差算出用ＮＮの出力で，ＡＪＮＮからは両者の差分が出力される。ＡＪＮＮを用いることにより，チューニング精度を向上させる上で最も重要な$\delta\theta_{out}=0$近傍における出力精度が向上する。しかも，

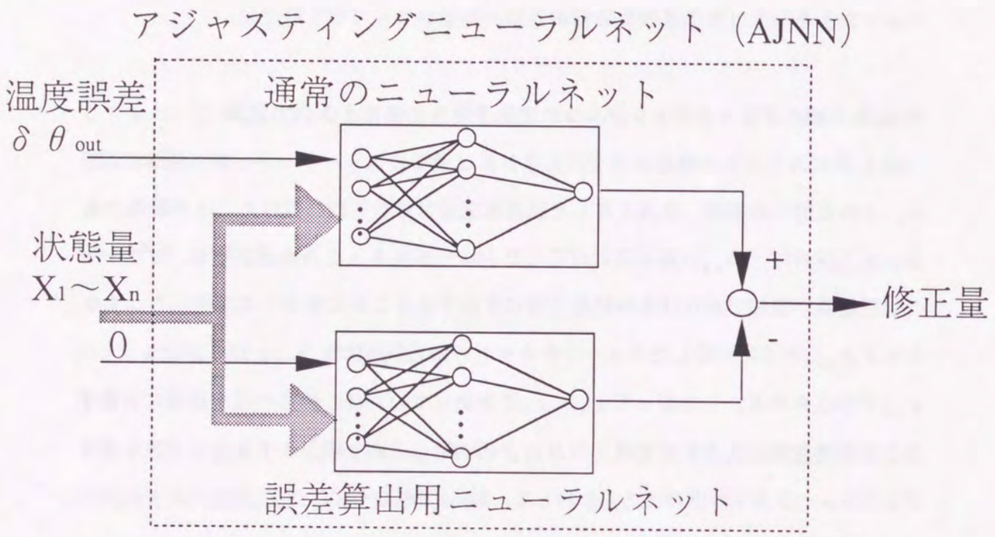


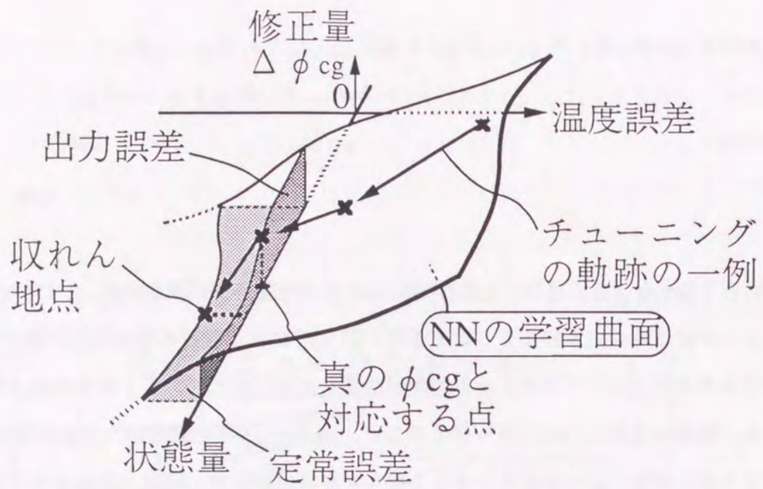
図5.2　アジャスティングニューラルネットの構成

$$AJNN_{out}(0, X_1, X_2, \cdots, X_n) = NN_{out}(0, X_1, X_2, \cdots, X_n) - NN_{out}(0, X_1, X_2, \cdots, X_n) = 0 \tag{5.8}$$

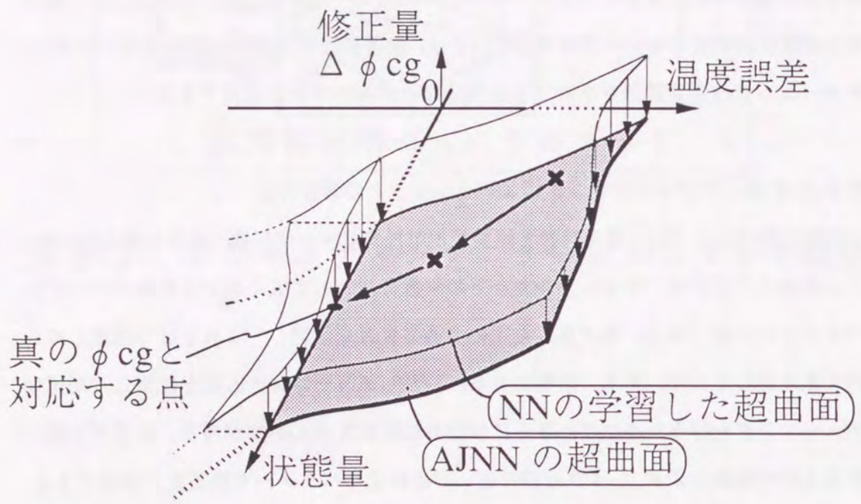
であることから，ＡＪＮＮの入出力関数は(5.3)式を恒等的に満足する。したがってチューニング終了時の定常誤差を０にできる。図５.３にＡＪＮＮの学習曲面の模式図を，プリミティブなニューラルネットの学習曲面と比較して示し，ＡＪＮＮの利点を説明する。簡単のためx_1～x_nを一軸で表した。$\delta\theta_{out}$と状態変数が入力されたときのプリミティブなニューラルネットおよびＡＪＮＮの出力が，縦軸で表わされている。理想的な学習曲面は，(5.6)式で表される$\delta\Phi_{cg}=0$と$\delta\theta_{out}=0$が交叉した超曲面（図では状態量の軸に対応した直線）を含む必要があるが，図５.３(a)ではこのような学習曲面は実現できない。チューニングの軌跡は一例として図のようになり，定常的な温度誤差が生じる。一方図５.３(b)では，学習曲面が前述した直線を含んでおり，チューニングの定常誤差を０にでき高精度なチューニングが実現できる。

第５.２.３項　アジャスティングニューラルネットの学習方法

前節で述べたように，通常ＮＮと誤差算出用ＮＮは同一の構成，重みを有している。このためＡＪＮＮは，ニューラルネットの学習方法として広く用いられるバックプロパゲーション法（以下，ＢＰ法）を用いて通常ＮＮを学習し，これを２つ用意して並列に組み合わせれば，簡単に構築できる。しかしチューニング回数を低減した高速なチューニングを行うためには，さらに学習方法を改善する必要がある。ＡＪＮＮは，構造上の特徴から$\delta\theta_{out}=0$近傍において高精度なパラメータ修正量を算出できるものの，$\delta\theta_{out}=0$から隔たった入力値では誤差算出用ＮＮの出力が通常ＮＮの出力誤差と対応しない。このため誤差算出用ＮＮの出力が大きいと，通常ＮＮからこの値を減算する処理によりＡＪＮＮの出力精度が逆に低下する場合がある。この結果，これらの入力値におけるＡＪＮＮの出力精度にばらつきが生じ，チューニング回数が増大する場合がある。



（a）ﾌﾟﾘﾐﾃｨﾌﾞなﾆｭｰﾗﾙﾈｯﾄの学習曲面



（b）AJNNの学習曲面

図5.3　学習曲面の比較

これを解決するために，ＡＪＮＮの学習では教師信号抽出後，まず$\delta\theta_{out}$の値を検出し，これに依存した以下の２つの学習則を繰り返すことにより行なう。

［1］$\delta\theta_{out}\neq 0$のとき　；

通常ＮＮには$\delta\theta_{out}$及び状態変数を，誤差算出用ＮＮには$\delta\theta_{out}$の代わりに０を入力して順方向の演算を行なう。この結果得られたＡＪＮＮの出力値と教師信号から誤差を算出し，後述する拡張ＢＰ法により誤差が減少する方向に通常ＮＮの重み係数を更新した後，この重み係数を誤差算出用ＮＮの重み係数にコピーする。

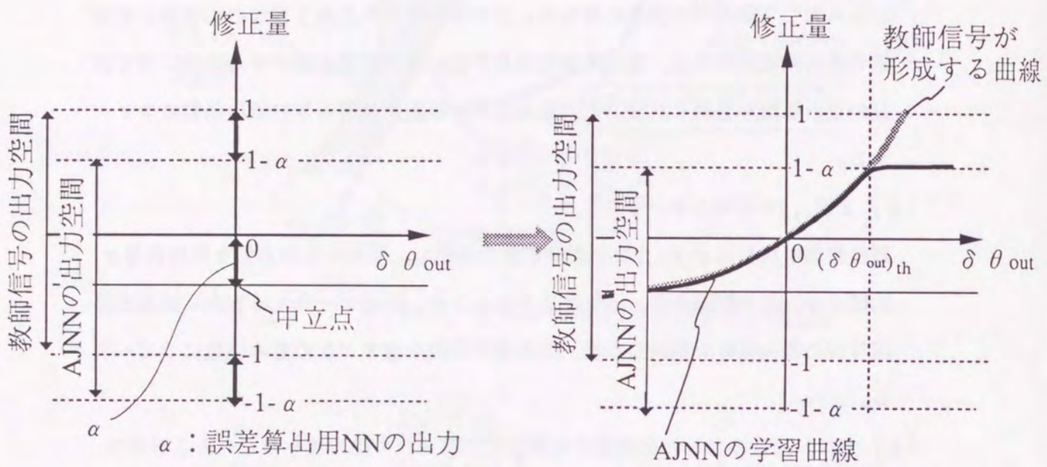
［2］$\delta\theta_{out}=0$のとき　；

誤差算出用ＮＮに$\delta\theta_{out}$及び状態変数を提示し，得られた出力値と教師信号を比較する。この結果から，通常のＢＰ法により，両者が一致する方向に誤差算出用ＮＮの重み係数を更新した後，この重み係数を通常ＮＮの重み係数にコピーする。

［1］では，通常のＮＮの出力値から誤差算出用ＮＮの出力値を減じたＡＪＮＮの最終出力値を教師信号と比較し，これらを一致させることにより$\delta\theta_{out}$の全領域でＡＪＮＮの出力精度を高精度化する。このため後述するように，通常のＮＮと誤差算出用ＮＮからなる並列ネットワーク用にＢＰ法を拡張する。

［2］では，通常のＢＰ法を用いて誤差算出用ＮＮの出力を小さくする学習を行ない，誤差算出用ＮＮが大きな出力値を持つ場合に生じる，以下の問題を回避する。

図5.4に，ＡＪＮＮの出力空間と学習曲線の様子を，入出力空間が２次元の場合を例に示す。横軸は$\delta\theta_{out}$，縦軸は$\delta\Phi_{cg}$の修正量にそれぞれ対応している。ＡＪＮＮが０を出力するときの誤差算出用ＮＮの出力値をαとすると，このときＡＪＮＮの出力空間の中立点は$-\alpha$に設定される。したがって，図5.4(a)に示すように，ＡＪＮＮの出力空間全体もαだけ負側にシフトされる。この結果，(b)に示すように，教師信号で定義される出力空間（$-1\sim1$）とＡＪＮＮの出力空間（$-1-\alpha\sim1-\alpha$）の間に不一致が生じるため，正負いずれかの空間がαに対応した$(\delta\theta_{out})_{th}$以上の領域でクリップされてしまい，この領域で教師信号との誤差が大きくなる。このように，誤差算出用ＮＮの出力値はチューニングの０点と対応するため，この値が



(a) AJNNの出力空間　　(b) AJNNの学習曲線

図５．４　AJNNの出力空間と学習曲線

０から隔たると，ＡＪＮＮの出力空間がいずれか一方の符号で縮退し，ＡＪＮＮが教師信号を学習できない場合が生じる。［２］の学習は，誤差算出用ＮＮの出力値を０に近づけることで空間の縮退を回避することに寄与する。

上記［１］，［２］の学習則のブロック図をそれぞれ，図５.５(ａ)，(ｂ)に示す。［１］の操作により，$\delta\theta_{out}=0$ 以外の教師信号に対して，ＡＪＮＮの最終出力値と教師信号が一致する方向に学習され，(5.3) 式の超曲面近傍のみならず，教師信号で与えた領域全体で良好な出力精度が期待できる。また［２］により，誤差算出用ＮＮの出力が０に近づくため，出力空間の縮退を最小化したネットワークが実現できる。

拡張ＢＰ法に用いる誤差関数及び各重み係数更新方法を以下に示す。通常ＮＮの出力を $NN_{out}(\delta\theta_{out}, x_1, x_2, \cdots, x_n)$，誤差算出用ＮＮの出力を $ENN_{out}(0, x_1, x_2, \cdots, x_n)$，ＡＪＮＮの出力を $AJNN_{out}(\delta\theta_{out}, x_1, x_2, \cdots, x_n)$ とする。拡張ＢＰでは，教師信号とＡＪＮＮの出力値との誤差を小さくする学習を行うため，誤差関数を以下のように修正する。T は，入力値（$\delta\theta_{out}, x_1, x_2, \cdots, x_n$）に対応した教師信号（望ましい Φ_{cg} の修正量）であり，以下ではＡＪＮＮが１出力の場合について示す。

$$E = (1/2)\cdot\{T - (NN_{out} - ENN_{out})\}^2 \tag{5.9}$$

(5.9) 式で算出された誤差Ｅを用いて，重み係数及びしきい値は (5.10)～(5.13) 式にしたがい更新される。n は重みの更新回数を表し，α は学習係数である。

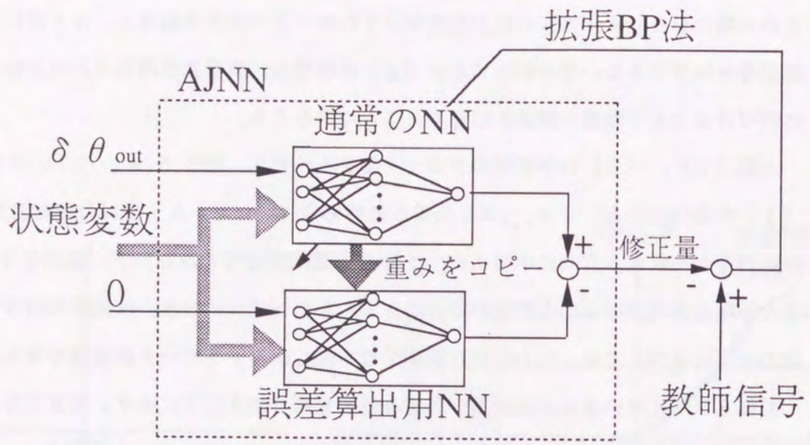
（中間層と出力層間の重み係数）

$$\{W_{j1}(2)\}_{n+1}=\{W_{j1}(2)\}_n+\alpha\cdot\{\partial E/\partial W_{j1}(2)\} \tag{5.10}$$

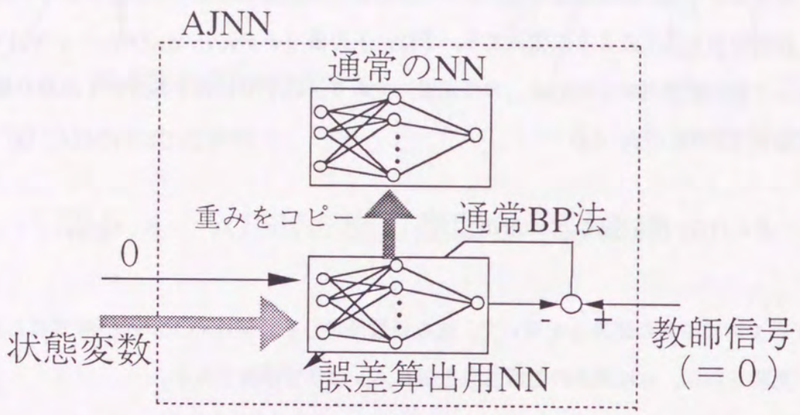
（入力層と中間層間の重み係数）

$$\{W_{ij}(1)\}_{n+1}=\{W_{ij}(1)\}_n+\alpha\cdot\{\partial E/\partial W_{ij}(1)\}_j \tag{5.11}$$

（中間層と出力層間のしきい値）



（a） δ θ out≠０のときの学習



（b） δ θ out＝０のときの学習

図５.５　ＡＪＮＮの学習法

$$\{\theta_1(3)\}_{n+1}=\{\theta_1(3)\}_n+\alpha\cdot\{\partial E/\partial\theta_1(3)\} \tag{5.12}$$

（入力層と中間層間のしきい値）

$$\{\theta_j(2)\}_{n+1}=\{\theta_j(2)\}_n+\alpha\cdot\{\partial E/\partial\theta_j(2)\}_j \tag{5.13}$$

ここで

$$\partial E/\partial W_{j1}(2)=\sigma_{NN}\cdot Y_j(2)-\sigma_{ENN}\cdot YE_j(2) \tag{5.14}$$

$$\begin{aligned}\{\partial E/\partial W_{ij}(1)\}_j &= (\varphi_{NN})_j\cdot Y_i(1)-(\varphi_{ENN})_j\cdot Y_i(1) && (i\neq 1)\\ &= (\varphi_{NN})_j\cdot Y_i(1) && (i=1)\end{aligned} \tag{5.15}$$

$$\partial E/\partial\theta_1(3)=\sigma_{NN}-\sigma_{ENN} \tag{5.16}$$

$$\{\partial E/\partial\theta_j(2)\}_j=(\varphi_{NN})_j-(\varphi_{ENN})_j \tag{5.17}$$

$$\sigma_{NN}=\zeta\cdot NN_{out}(1-NN_{out}) \tag{5.18}$$

$$\sigma_{ENN}=\zeta\cdot ENN_{out}(1-ENN_{out}) \tag{5.19}$$

$$(\varphi_{NN})_j=\sigma_{NN}\cdot W_{j1}(2)\cdot Y_j(2)\cdot\{1-Y_j(2)\} \tag{5.20}$$

$$(\varphi_{ENN})_j=\sigma_{ENN}\cdot W_{j1}(2)\cdot YE_j(2)\cdot\{1-YE_j(2)\} \tag{5.21}$$

$$\zeta=T-(NN_{out}-ENN_{out}) \tag{5.22}$$

である。また $YE_j(2)$ は誤差算出用ＮＮの j 番目の中間層ニューロンの出力値である。

図５.６に拡張ＢＰの学習アルゴリズムの詳細を示す。抽出した教師信号のペアに対

して$\delta\theta_{out}$の値を検出し，0でないならば［1］の処理を，0ならば［2］の処理を，学習が収れんするまで繰り返す。

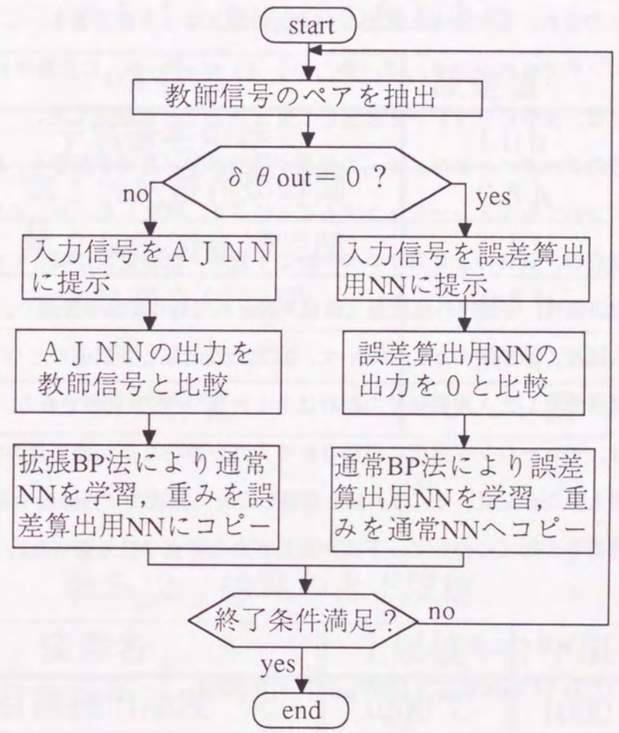


図5.6　ＡＪＮＮの学習アルゴリズム

第5.3節　実験結果および考察

第5.3.1項　実験条件

シミュレーションに用いたＡＪＮＮには，温度誤差$\delta\theta_{out}$の他に，状態変数として各炉帯の温度$T_1 \sim T_4$，鋼材の厚みH，鋼材の挿入温度θ_{in}，モデルに設定されているΦ_{cg}が入力され，これらから算出したΦ_{cg}の修正率αを出力する。したがって一回のチューニングでモデルのΦ_{cg}は，Φ_{cg}から（$1+\alpha$）$\cdot\Phi_{cg}$に更新される。加熱炉の操業条件は，実際のプラントを想定して表5.1のように設定した。

教師信号のデータベースは，Φ_{cg}の偏差が温度誤差に及ぼす影響を，種々の偏差，状態変数についてシミュレーションすることにより，構築した。Φ_{cg}の偏差については0および正側，負側の１０通りづつの計２１通り，状態変数に関しては表5.2に示す，目標抽出温度，目標内外温度差（鋼材の表面と内部の許容温度差），鋼材の厚み，鋼材の挿入温度，加熱炉のΦ_{cg}について，操業上の上限と下限値の２つづつに対応した値を教師信号とした。教師信号の総計は$21\times2^5=672$個である。実機に適用する場合には，チューニングの度に獲得される入力と適切なΦ_{cg}修正量の関係を教師信号として蓄え，これを用いてＡＪＮＮを構築したり，逐次的に再構成することになる。学習は教師信号の総てについて，下式が満足された時点で打ち切った。

$\delta\theta_{out}\neq0$のとき

$$(1/2)\cdot\{T-(NN_{out}-ENN_{out})\}^2<0.0001 \tag{5.23}$$

$\delta\theta_{out}=0$のとき

$$(1/2)\cdot(ENN_{out})^2<0.0001 \tag{5.24}$$

第5.3.2項　実験結果

まずＡＪＮＮ，通常のニューラルネット，および現行のプラントでΦ_{cg}のチューニングに広く用いられている線形式について，チューニング結果の一例を示し，精度およびチューニング回数の比較を行なう。Φ_{cg}の真の値を0.6，モデルのΦ_{cg}の初期値を

表5.1　加熱炉の操業条件

変数名	設定値
予熱帯在炉時間	1.0 h
第１加熱帯在炉時間	0.5 h
第２加熱帯在炉時間	0.5 h
均熱帯在炉時間	0.5 h
室　温	30 ℃
鋼材材質	リムド鋼

表5.2　操業の上下限値

変数名	上限値	下限値
目標抽出温度	1200 ℃	1000 ℃
目標内外温度差	50 ℃	30 ℃
鋼材の厚み	0.2 m	0.1 m
鋼材の挿入温度	500 ℃	30 ℃
加熱炉のΦ_{cg}	0.8	0.5

0.7とし，線形式はチューニングが発散しないことに配慮した，

$$\alpha = 0.005 \cdot \delta\theta_{out} \tag{5.25}$$

を用いてチューニングを行った結果を一例として示した。表5.3中のΦ_{cg}，$\delta\theta_{out}$は，それぞれ修正されたΦ_{cg}，およびこれを用いて算出された温度誤差である。チューニングが収れんしたかどうかの判定は次式に従った。

$$|\{\Phi_{cg}\}_n - \{\Phi_{cg}\}_{n-1}| \leq 0.01 \tag{5.26}$$

nは試行回数であり，上式を満たすnが存在したとき，チューニングはn回目で収れんしたと定義する。表より，線形式によるチューニングでは何度か振動した後，6回のチューニングで真のΦ_{cg}に収れんしている。通常のニューラルネットでは2回のチューニングで収れんしているが，Φ_{cg}の値は0.56であり，約14℃の定常温度誤差が残っている。これに対して，ＡＪＮＮによるチューニングでは，2回のチューニングで真のΦ_{cg}に収れんしている。

次に評価結果の一般性を確認するため，状態変数の組合わやモデルのΦ_{cg}の設定値を変えた３０通りについてシミュレーションを行った結果を示す。表5.4に線形式，通常ＮＮ，さらにＡＪＮＮのそれぞれについて，チューニングの収れん率，回数，精度の比較を行なった結果を示す。表中の収れん率はチューニング精度を考慮にいれない成功率であり，チューニング精度は収れん状態における真のΦ_{cg}に対する誤差の割り合いである。ＡＪＮＮによるチューニングは，線形式よりも収れん率が向上し，チューニング回数も低減されている。また，通常のニューラルネットに比べ，チューニング精度が向上し，1℃以下の温度誤差を実現している。以上より，ＡＪＮＮでは線形式より少ない回数で，また通常のニューラルネットよりも高精度にチューニングが行なえることが分かる。

次に学習方法改善の効果を明らかにする。表5.5は，従来のＢＰ法により構築したネットワークを並列に組み合わせたＡＪＮＮと，前節で開発した学習方法により構築

表5.3　チューニング結果の一例

	線形式		通常のﾆｭｰﾗﾙﾈｯﾄ		AJNN	
ﾁｭｰﾆﾝｸﾞ回数	$\delta\theta$ out	ϕ cg	$\delta\theta$ out	ϕ cg	$\delta\theta$ out	ϕ cg
0	-32.939	0.70	-32.939	0.70	-32.939	0.70
1	25.027	0.53	9.314	0.57	-9.597	0.63
2	-9.041	0.63	14.688	0.56	-1.932	0.60
3	5.678	0.58	14.864	0.56	-0.685	0.60
4	-2.975	0.61	14.710	0.56	-0.090	0.60
5	1.664	0.59	14.726	0.56	-0.012	0.60
6	-0.826	0.60	12.346	0.56	-0.029	0.60
7	0.463	0.60	19.557	0.56	-0.004	0.60
ﾁｭｰﾆﾝｸﾞ回数	6		2		2	
ﾁｭｰﾆﾝｸﾞ精度	0.0%		6.7%		0.0%	

表5.4　チューニング結果の比較

	収れん率	ﾁｭｰﾆﾝｸﾞ精度	温度誤差	平均ﾁｭｰﾆﾝｸﾞ回数
線形式	70.0%	0.46%	0.64℃	4.14 回
通常ﾆｭｰﾗﾙﾈｯﾄ	100%	10.78%	17.99℃	2.40 回
AJNN	100%	0.44%	0.81℃	2.20 回

表5.5　学習方法の違いによるチューニング結果

ﾈｯﾄﾜｰｸ番号	従来学習法	開発学習法
1	2.55	2.47
2	3.90	2.27
3	2.77	2.20
4	2.70	2.47
5	2.83	2.80
平均	2.95	2.44

したＡＪＮＮについて，重み係数の初期値を変えた５つのネットワークでチューニング回数の比較を行なった結果である。いずれも，状態変数を変えた３０通りのネットワークで評価した平均値である。従来学習法によるＡＪＮＮでは，チューニング回数がネットワークに依存してばらついており，とりわけネットワーク２ではチューニング回数が３.９回と大きな値になっている。開発学習法によるＡＪＮＮでは，いずれのネットワークでも少ない回数で安定したチューニング結果が得られている。以上から開発手法では，ばらつきの少ないチューニング特性が実現できることがわかる。

第5.4節　まとめ

制御モデルのパラメータチューニングを高速・高精度に行うことを目的に，従来のニューラルネットを拡張したアジャスティングニューラルネット（ＡＪＮＮ）を提案した。ＡＪＮＮは，通常のニューラルネットに加え，これと同一の構成および重みを有した誤差算出用ニューラルネットを並列に結合したアーキテクチャを備えており，通常のニューラルネットの出力から誤差算出用ニューラルネットの出力を減じた値を出力する。さらに，バックプロパゲーションに用いる誤差関数を並列ネットワーク向けに修正し，これを用いた並列ネットワークの学習と，誤差算出用ニューラルネットの出力を０に低減する学習を組み合わせた，ＡＪＮＮの学習方式を提案した。

ＡＪＮＮを加熱炉温度制御における鋼材モデルのパラメータチューニングに適用したシミュレーションで，現在広く用いられている線形式によるチューニングや従来のニューラルネットを用いたチューニングに比べ，少ない回数で定常温度誤差の小さいチューニングが行えることを示した。

開発手法は，モデルを用いて指令を生成する制御系の制御モデルチューニングをはじめ，レギュレータ制御の指令発生をニューラルネットで行う場合などに，広く適用可能な手法である。今後種々の応用へ適用を試み，応用分野の拡大を検討していく。

5章の参考文献

（1）W. T. Miller, III, R. S. Sutton, and P. J. Werbos, "Neural Networks for Control", The MIT Press, Cambridge, Massachusetts, London, England (1990)

（2）横山·幸田·井上，「混合構造型ニューラルネットワークによるプラントの同定と最適制御のシンセシス」, 計測自動制御学論 , Vol.29, 3 (1993)

（3）堀·陰山·土谷，「多層ニューラルネットワークを用いた運動量変化逆モデル学習によるマニピュレータ軌道制御」, システム制御情報学論 , Vol.7, 1 (1993)

（4）鉄鋼協会編，連続鋼片加熱炉における伝熱実験と計算方法(1971)

第６章　ニューラルネットを搭載した
汎用コントローラの高速制御方式

第６章　ニューラルネットを搭載した汎用コントローラの高速制御方式

第６.１節　研究の概要

ニューラルネットを実際の生産ラインにおける制御や認識に適用する場合，計算時間を要する学習はワークステーション等の計算サーバで行い，学習後のネットワークを用いた演算は応答特性を高める目的で，ＤＤＣ（Direct Digital Controller，以下，コントローラ）と呼ばれる，制御対象に直接対応した末端の汎用コントローラで実行される場合が大半である。このような学習（オフライン系）と推論（リアルタイム系）を切り分けたアーキテクチャでは，必然的にリアルタイム系のニューラルネット演算は，コントローラにおいてニューラルネット以外の処理と並行して実行される。ニューラルネット以外の処理の大部分を占めるのはシーケンス処理であり，したがってニューラルネットを搭載したコントローラの性能を高めるためには，第２章および第４章で検討したニューラルネットの構成最適化の他に，シーケンス処理の高速化を達成する必要がある。

シーケンス演算の処理速度向上の要求は，制御対象の拡大や末端の制御機器であるリレーやセンサの性能向上とともに年々高まっており，ハード，ソフト両面から種々の工夫がなされている。ハード面では高速マイクロプロセッサの導入や，シーケンス制御に特有の多数の論理処理を高速処理する専用ＣＰＵの開発[1]が挙げられる。また，この種の制御装置の多くでは，入力信号を走査的に検出し，対応した出力を決定するための論理演算を順に実行する，いわゆる全スキャン方式で制御が行われている。このため，ソフト面からのアプローチとしてプログラムの中で高速応答を必要とする部分を別走査とし，タイマ割込で定周期実行する方法がある。さらに全スキャン方式では，本来変化するはずのない入力に対しても周期的な検出処理や，それに伴う論理演算，出力処理を行うことが本質的な実行上の無駄として存在している。この無駄を省略するために，コントローラの一部では，プログラムをサブルーチンで定義し，各サブルーチンの起動・終了を管理することでこれらを選択的に走査する方法が用いら

れている。

一方，シーケンスシステムを構成する各作業が非同期・並列で次々と起動される様子と，トークンの移行によって表現されるペトリネット[2]の状態遷移との親和性の良さが指摘されており，これを実システムの設計[3]やシミュレーション[4]へ適用する試みが報告されている。また，ペトリネットの解析性の高さを利用した診断システム[5]，可視性を利用したコントローラのプログラミングシステム[6]への適用例も報告されている。

これらの研究に対し本章では，シーケンス制御を高速に実行する目的でペトリネットの応用を試み，ニューラルネットの応答性を高めることを検討する。具体的にはペトリネットのトークン遷移を利用したタスク管理の導入で，全スキャン方式の実行上の無駄を積極的に排除し，コントローラを高速化する手法について述べる。まずニューラルネット演算をはじめとした生産ラインで行うべき種々のタスクを実時間処理に適した単純な形でモデリングする方法を示す。次にペトリネットの状態遷移を模擬したトークン管理により実行すべきタスクの高速抽出を行うアルゴリズムを提案する。最後に試作システムを用いて演算速度を評価し，全スキャン方式に比べて処理速度を向上できることを明らかにする。

第6.2節　シーケンス処理の高速演算法

第6.2.1項　ペトリネットとこれを利用したシーケンスシステムのモデリング

ペトリネットは1962年にPetriにより提案されて以来，離散システムのモデリング手法として，計算機システムを始めさまざまなシステムのモデリングに適用が検討されている。ペトリネット構造はプレースの集合P，トランジションの集合T，入力関数I，出力関数Oからなり，さらにPtはマーキングベクトルと呼ばれ，トークンの存在するプレースの集合を表す。これらは等価なペトリネットグラフで表される。図6.1(a)にペトリネット構造およびマーキングベクトルの一例を示す。また図6.1(b)にこれと対応したペトリネットグラフを示す。図でI(ti)はトランジションTiから見た入力プレースを，O(ti)は同様にトランジションTiから見た出力プレースを特定しており，Ptは，プレースの2，5，7，9に一つずつ，4に2つのトークンが存在することを示している。

ペトリネット構造により生産ラインの順序，並列処理の構造が対応よく表現でき，トークンの遷移で加工物の移動が模擬できることから，緒言で述べたように，シーケンスシステムの分野においても，ペトリネットの種々の性質（表現能力，解析能力，ペトリネットグラフの可視性）に着目した適用が検討されている。図6.2にペトリネットを生産ラインのモデル化に適用した例を示す。プレースに各作業内容を割り付け（ボックスに対応），作業順序にしたがってゲート（トランジションに対応）を介してアークにより接続する。図6.2では，B1でインタロックが解除された後，ロボットのアクチュエータであるサーボ系のX軸とY軸をそれぞれA点，B点に位置決めし，さらにその後，ニューラルネットで所望の加工物の到着を検知し，ロボットのハンドをオンする処理を行うことを意味している。各ボックスで定義した作業はトークンの入力により起動され，ゲートによりトークン移行の可否が判定される。ゲートは以下のゲート条件，

＜ゲート条件＞

・入力側のすべてのボックスにトークンがある

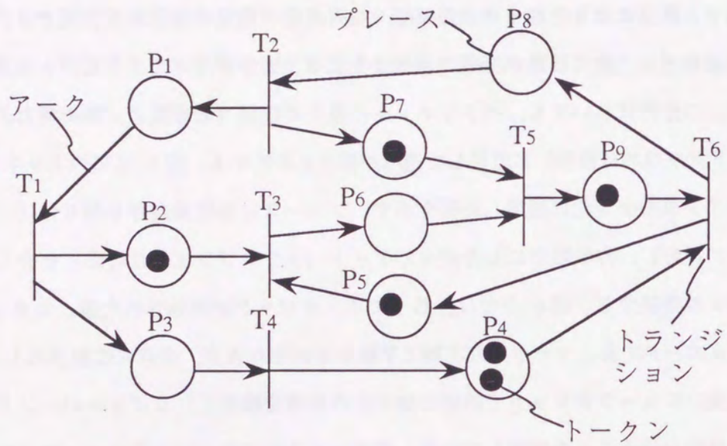
$$C = (P,\ T,\ I,\ O)$$

$$P = \{P_1, P_2, \cdots\cdots, P_8, P_9\}$$
$$T = \{T_1, T_2, \cdots\cdots, T_5, T_6\}$$
$$I = \{I(t_1), I(t_2), \cdots\cdots, I(t_5), I(t_2)\}$$
$$O = \{O(t_1), O(t_2), \cdots\cdots, O(t_5), O(t_2)\}$$

$I(t_1)=\{P_1\}$	$O(t_1)=\{P_2, P_3\}$
$I(t_2)=\{P_8\}$	$O(t_2)=\{P_1, P_7\}$
$I(t_3)=\{P_2, P_5\}$	$O(t_3)=\{P_6\}$
$I(t_4)=\{P_3\}$	$O(t_4)=\{P_4\}$
$I(t_5)=\{P_6, P_7\}$	$O(t_5)=\{P_9\}$
$I(t_6)=\{P_4, P_9\}$	$O(t_6)=\{P_5, P_8\}$

$$P_t = \{0, 1, 0, 2, 1, 0, 1, 0, 1\}$$

（a）ペトリネット構造とマーキングベクトル



（b）ペトリネットグラフ

図6.1　ペトリネット構造とペトリネットグラフ

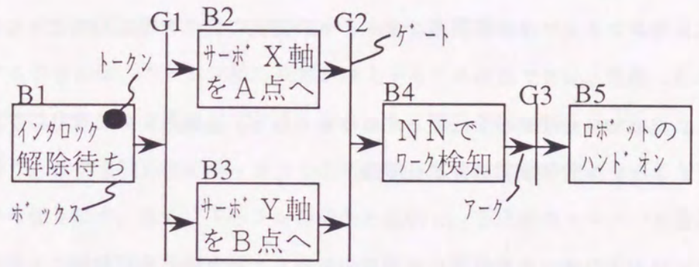


図6.2　ペトリネットを用いた生産ラインのモデリング

・出力側のどのボックスにもトークンがない

が成立したとき，入力側のすべてのボックスのトークンを消失させ，出力側のすべてのボックスにトークンを出現させる。

本章で以下検討するのは，このような制御対象の構造を模擬したモデルから制御対象を直接制御するためのデータベースを構築し，これに基づいて制御を行うことである。トークンの存在するボックスの作業が現在実行すべき作業であるため，トークンに着目することにより，現在行うべき作業のみを対象とした制御を行うことができる。これによりコントローラの演算量を全スキャン方式に比べ大幅に削減できることが期待される。課題として，

（1）連続的な時間処理を必要とする作業（以下，連続系）のモデリング方法

（2）トークン管理演算の簡単化手法

がある。

（1）はペトリネットが時間の表現能力を有していないことに起因した問題点で，ペトリネットの利用により生産ラインの状態遷移をトークンの移行により対応良く表現できるが，トークンの移行に要する時間や，トークンが移行してからの経過時間等を表わせない。一方，生産システムには出力処理等の一度の実行で演算が終了する離散的な処理の中に，タイマ処理のような時間演算を必要とする処理や，入力待ちのような連続的な検出演算を必要とする処理が混在している。このため図6.2のモデルにどのような方法で時間の概念を導入し，連続処理を必要とする対象に対して，一定間隔以内の周期的な実行を保証するかが問題となる。

また（2）については，従来の全スキャン方式に比べ，ペトリネットを適用した方式では，本来の制御演算に加えてトークンの移行を管理する演算が新たに付加される。したがってこの演算時間を縮減することが，提案手法の利点を生かし，リアルタイム制御への適用を可能にするために重要である。以下，2つの課題について検討した結果を述べる。

第6.2.2項　連続系のモデリング方法

時間の経過を伴う離散システムをモデル化する場合，プリミティブなペトリネットでこれを表現できないことの不都合はすでに指摘されている。これを解決するために提案されている一般的な手法は，ボックスの機能を生産ラインへの出力処理に限定し，入力処理等の連続系の作業を，新たに定義した許可枝，禁止枝を用いてゲート条件とすることにより，トークン移行条件に組み入れる方法である。この手法の場合，入力の検出はゲート条件を確認する演算に含めて行うことになるため，必然的にゲートの周期的な走査が必要となる。またゲートにおける条件判定の演算が複雑になる。このためオフラインシミュレーションへ応用する場合には問題ないが，リアルタイム制御に適用する場合には，トークン移行演算が増大するため適当でない。

本方式では，以下のようにして，ペトリネットに構造上の拡張を施すことなく連続処理を扱う。まず連続処理をボックスの作業内容として独立して定義することでゲート条件から切り離す。さらにトークンの概念を拡張し，2種類のトークンを用いて連続処理ボックスの状態を識別する。2種類のトークンとは作業の実行中を示す実行トークンと，作業は終了したが，ゲート条件が真でないためにボックスに停留している終了トークンである。例えばサーボの位置決め制御の場合，図6.3に示すように，位置決め実行中は実行トークンの状態を保持し，位置決め制御を完了すると終了トークンに変わる。このとき先に述べたゲート条件は，

＜ゲート条件＞

・入力側のすべてのボックスにトークンがあり，終了トークンである

・出力側のどのボックスにもトークンがない

の様に修正され，この条件が共に成立したとき，トークンの移行が行われる。トークンにこのような属性を新たに設け，コントローラが行うボックスの作業とゲートの判定処理の実行タイミングを，次節に示すようにトークンの属性にしたがって決定することにより，連続系の処理やゲート条件の判定処理を，離散的な出力処理の中に混在させて適切に時分割処理できる。

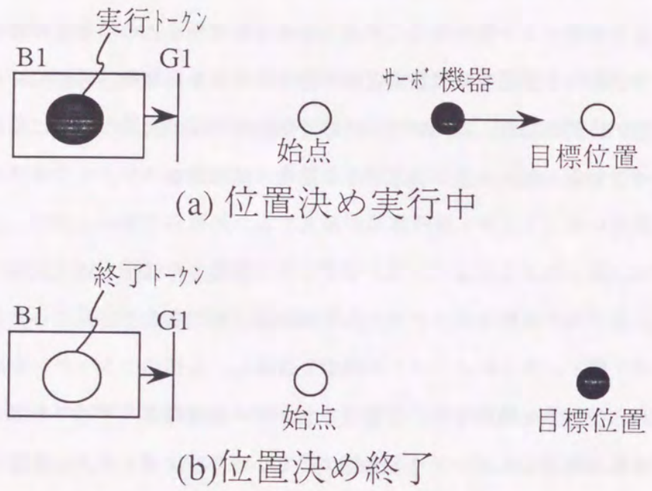


図6.3　実行トークンと終了トークン

第6.2.3項　トークン管理演算の簡単化手法

図6.2のモデルにしたがって制御を行うためには，まずこのモデルを制御実行に適した形で，データ化する必要がある。一般にボックスとゲートの接続関係は，それと等価な接続マトリクスで表される。これを利用するとたとえば図6.2の内容は，図6.4(a)(b)に示す接続マトリクスと，各ボックスの作業内容を示す情報とに分離してデータ化することができる。ここで接続行列の－1はゲートから見て入力のボックスを，また1は出力のボックスを示している。さらに0は未接続であることを示している。トークンのマーキングは一般に，図6.5に示すような接続マトリクスに基づいた演算を実行することで求めることができる。発火ゲート列において，1はゲート条件が真となったゲート番号と対応しており，図ではG1のゲート条件が真となってトークン遷移が可能となっていることを示している。しかしトークンが移行するたびに行列演算を行い，トークンの移行先を決定するのは，多大な演算量を必要とする。また発火ゲート列を求めるにあたり，各ゲートについて順にゲート条件をチェックするのは，無駄が多い。このため本方式では，図6.4のように接続マトリクスを1，－1の部分のみを抽出することで縮約してデータ化した。すなわち各ボックスの出力ゲートと，各ゲートの入出力ボックスを特定し，図6.4(c)のような接続テーブルとして編集した。コントローラは図6.4(b)(c)をプログラム実行のためのデータとして制御を実行する。

図6.6にトークン管理のアルゴリズムを示す。開発した方式は従来の全スキャン方式に対して，部分スキャン方式とでも呼ぶべき手法で，その特徴は，トークンの存在するボックスを一括してＦＩＦＯスタックに登録し，ゲート条件の判定結果に従ってスタックを更新することで，実行すべきボックスの抽出を能率的に行う点にある。

まずこのスタックにより実行すべきボックスを決定する。次にボックスのトークンが実行トークンなのか終了トークンなのかを判定し，実行トークンの場合には図6.4(b)によりボックスの処理内容を認識した後，実行する。そして実行トークンを終了トークンにするかどうかを判定する。“ロボットのハンドをオンする”といった命令の場合は一度の演算で実行トークンは終了トークンになるが，連続演算を必要とする“サーボＸ軸の位置決め”や“ニューラルネットでワーク検知”，“インタロック解除待ち”のような命令の場合には，実行トークンの状態が継続される場合がある。実行

$$
\begin{array}{c} \\ B1 \\ B2 \\ B3 \\ B4 \\ B5 \end{array}
\begin{array}{c} \begin{array}{ccc} G1 & G2 & G3 \end{array} \\ \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \end{array}
$$

(a) 接続マトリクス

B1：ｲﾝﾀﾛｯｸ解除待ち
B2：ｻｰﾎﾞ X軸をA点へ
B3：ｻｰﾎﾞ Y軸をB点へ
B4：ﾆｭｰﾗﾙﾈｯﾄでﾜｰｸ検知
B5：ﾛﾎﾞｯﾄのﾊﾝﾄﾞｵﾝ

(b) ボックスの作業内容

ボックス	出力ゲート
B1	G1
B2	G2
B3	G2
B4	G3
B5	

ゲート	入力ボックス	出力ボックス
G1	B1	B2,B3
G2	B2, B3	B4
G3	B4	B5

(c) 接続テーブル

図6.4　制御演算に用いるデータ構造

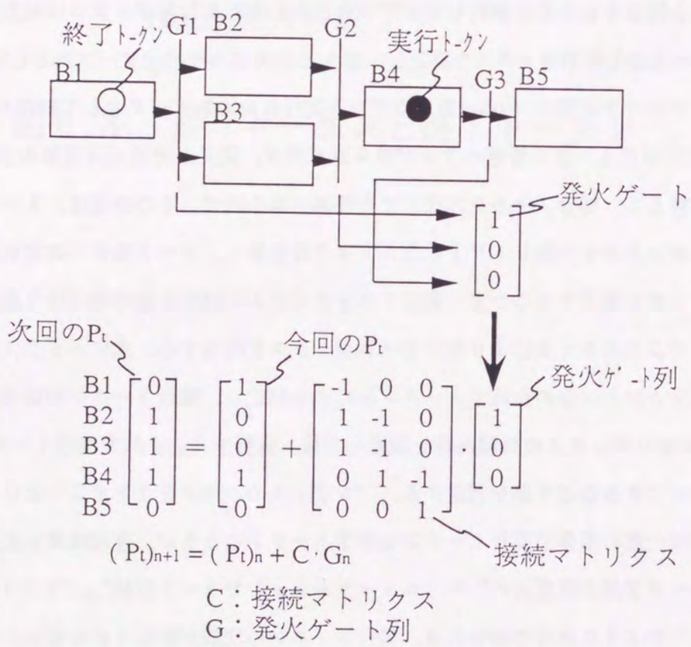


図6.5　接続マトリクスによる
トークン遷移演算

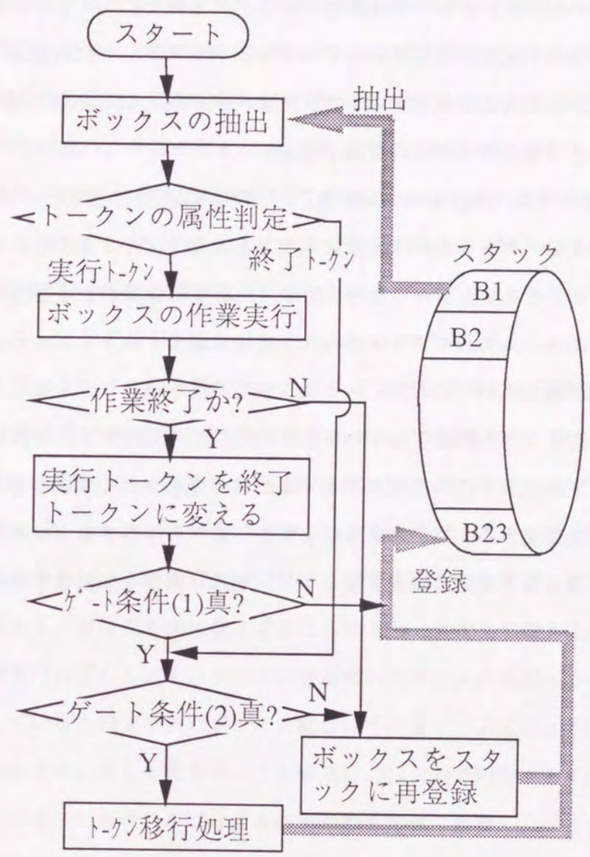


図6.6　トークン管理アルゴリズム

トークンの状態が継続された場合には，このボックスを再びスタックに登録する。また終了トークンに変化した場合には，図6.4(c)のテーブルにより対応するゲートを抽出し，ゲート条件(1)(2)の判定を行う。いずれも真と判定された場合には，トークンの移行が許可されたので，ゲートの出力側のボックスを新たにスタックに登録する。ゲート条件(1)が偽と判定された場合にはそのボックスをスタックに再登録し，次回の処理へ進む。ゲート条件(1)は真であるがゲート条件(2)が偽と判定された場合には，抽出しているボックスをスタックに再登録する。

図6.6に示すように，実行トークンが終了トークンに変化した時をゲート条件(1)の判定タイミングとし，ゲート条件(1)が真と判定された時をゲート条件(2)の最初の判定タイミングとした。そして，ゲート条件(1)が成立し，ゲート条件(2)が成立していないゲートに関してのみ，入力側のボックスのうち最後に終了トークンに変わったボックスをスタックに登録しておくことで，スタックから終了トークンがを有したボックスが抽出されたタイミングを関連するゲート条件判定処理の実行タイミングに対応させた。この結果，ゲート条件(2)の判定処理をボックスで定義された制御演算の実行に組み入れて時分割実行できる。以上のようにしてトークンの存在するボックスの処理のみに着目し，不要な演算を極力除去することで，制御に必要な演算量を最小化した。

次に図6.4(b)(c)のデータ構造に基づいたコントローラの実行方式を検討する。コントローラのプログラム実行方式としては一般にインタプリタ方式とコンパイル方式がある。高速性を考えると基本的にはコンパイル方式で実行することが望ましいが，現場での使用を考えると，制御定数の変更等はコンパイルなしに簡易なツールで行なえる方が操作性が良い。そこで図6.4(b)のボックスの作業内容を汎用化した命令コードと定数を含む一定サイズの中間語表現とし，図6.4(b)(c)のデータ構造を中間語としてプログラムを実行する方式とした。これらは最終的にコンパイルされた形で実行されるが，中間語における定数格納領域のリロケータブルアドレスとサイズを固定しておくことにより，定数の値の変更はコンパイル後のオブジェクトに対して直接行なうことができ，再コンパイルの必要はない。図6.7にプログラム実行部の構成を示す。制御実行部は図6.6のアルゴリズムを実行するトークン管理部と，図6.4(b)(c)のデータ構造に対応する駆動データ，さらに処理アルゴリズム群により構成する。駆動データの中間語群には，ボックスの作業内容がそれぞれ命令コードと定数のセットに翻訳された形で格納されている。処理アルゴリズム群には各命令コードに対応して実行するアルゴリズムを，定数部を未知数とすることにより一般化して備える。

次に制御実行の流れを示す。トークン管理部のスタックから抽出され，実行トークンを保持していると判定されたボックスについて，まずパス①により処理内容を示す中間語を検索する。そして命令コードを解読し，パス②で対応する処理アルゴリズムを抽出する。そして処理アルゴリズムの未知数を定数で置換し，これを実行する。連続系の処理を行なうアルゴリズムでは，入力検出や論理演算を1回だけ実行する。そして処理アルゴリズムの最後で，命令コードで定義された処理が終了したかどうかを判定し，判定結果をコード化した後，パス③でトークン管理部に復帰する。トークン管理部では，このコードの値により実行トークンを終了トークンに変えるかどうかを判定する。

さらに図6.6のアルゴリズムは，順序，並列，多重制御に関して十分な能力を備えているが，実際の制御を高機能に行なうためは，分岐，繰返し制御を始めとして，さ

らにトークン管理機能の拡張が必要な場合がある。またコントローラの制御機能の上限がこのアルゴリズムの能力で決まってしまうことを避けなければならず，このためトークン管理機能の拡張方法を検討する必要がある。ここでは競合ゲートの調停演算を例に，本論文で提案する機能拡張の考え方を説明する。競合ゲートの調停演算とは図6.8(a)に示す競合ボックスの，複数のトークン排出ゲートから一つを選択する処理である。本方式では，例外処理の付加による基本アルゴリズムの煩雑化を避けるため，この演算をボックス処理として定義し，図6.6のアルゴリズムから分離した。そしてこのボックスの出力アークを，図6.8(b)のように仮想的に未接続とし，ゲートの探索処理をボックスの処理アルゴリズムの中で実行する。さらにトークンの存在しているボックスが制御の対象となるため，タスクの単位はできるだけ小さく設定することが望ましい。そこで競合ボックスは作業に対応したタスクと，競合ゲートの調停演算を行うタスクに分割して図6.8(b)の形でモデル化した。

競合ゲートの調停演算を行なうボックスを，図6.9(a)の中間語で表現し，命令コードCNFに対応して図6.9(b)のアルゴリズムを定義する。そしてこのボックスに対応した中間語表現の定数部には移行先のゲート番号を格納しておく。処理アルゴリズムでは，まず移行先ゲートを決定する。そしてそのゲートについてゲート条件(2)の判定を行ない，その結果によってトークン管理部の所定の部分に復帰する。移行先ゲートの決定条件は，交互，優先等，いろいろ考えられるが，図6.9(b)のアルゴリズムの内容に組み込むことにより図6.8の基本アルゴリズムに影響を及ぼすことなく対処できる。条件分岐命令や繰返し命令についても，図6.9(b)のアルゴリズムの“移行先ゲート決定”の条件として，論理演算の結果やループカウンタの値を用いることにより同様の形で実行でき，多様なトークン管理を実現することができる。

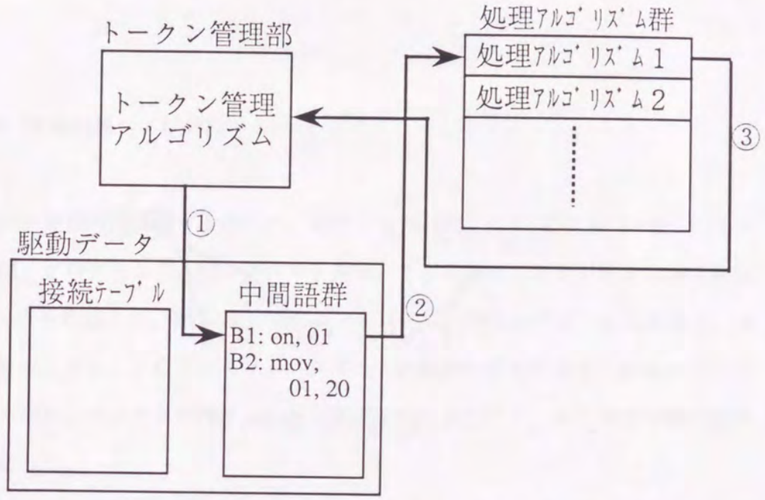
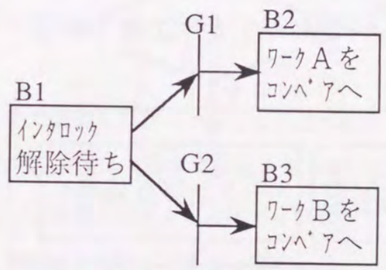
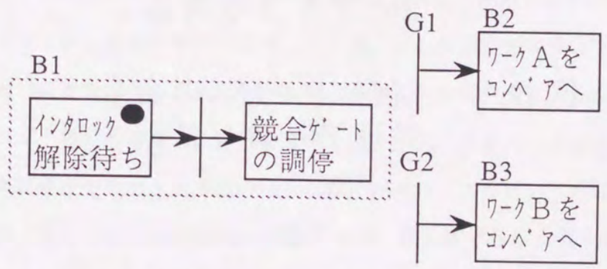


図6.7　プログラム実行部の構成

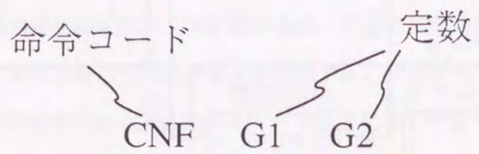


(a) 競合ボックス

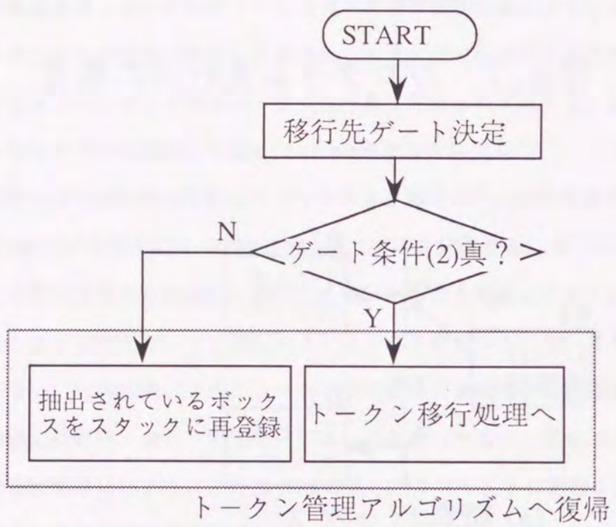


(b) 競合ボックスのモデル化方法

図6.8　競合ボックスのモデル化方法



(a) 命令コード



(b) アルゴリズム

図6.9　競合ボックスの命令コードと処理アルゴリズム

開発手法を定量的に評価する目的で，16ビットの汎用ＣＰＵ(*i*80186, 8 MHz)とシステムＲＯＭ，プログラムＲＡＭ，データＲＡＭ，Ｉ／Ｏ接点，タイマ等からなる試作コントローラを作成した。図6.10に試作コントローラの概観を示す。また図6.11，6.12にＣＰＵボードとＩ／Ｏインタフェースボードの概観を写真で示す。評価システムの簡単化のため，ボックスの作業は信号出力，信号の入力待ち，タイマ待ち等の基本命令に限定した。

図6.6のトークン管理アルゴリズムにおいてクリティカルパスに相当するのは，スタックから実行トークンを抽出し，ボックスの作業を実行後，ゲート条件がいずれも真となった場合である。この演算を試作コントローラは約140μsで実行できる。処理実行に要する時間はボックスの作業内容に依存しているが，定義した基本命令の範囲では，最大60μsであった。このとき基本アルゴリズム１サイクルの演算時間，すなわちスタックに登録されているボックス１つについて必要な演算時間は，最大200μsと見積られる。

次に全スキャン方式で制御を行なった場合との処理速度の比較を示す。シミュレーションは，図6.13のペトリネット表現で記述された並列度５，多重度２の生産システムを想定して行った。大規模システムに対応づける目的で，入出力接点数としてのべ1000点を想定した。等価なシステムをラダー図で記述し，上述したハードウェア上で実行した場合の走査時間は一定で，プログラムの記述方法により変化するが，40ms程度である。図6.14にこれと本方式による演算時間を比較した結果を示す。本方式では，演算時間はスタックに登録されているボックス数に対して直線的に増大する。ボックス数は制御の対象となっている加工物が並列，多重にいくつ存在するかに依存し，これにインタロックのためのボックスに存在するトークンの数を足した値となる。同時刻にスタックに存在するボックス数の最大値はインタロックボックスの記述の仕方により異なるが，図6.13の記述では10～20個である。最大値である20個の場合４msで走査が完了することになる。以上より想定した生産システムでは，本方式は全スキャン方式に比べ制御を10倍以上に高速化できる可能性があり，このとき制御の遅れ時間は

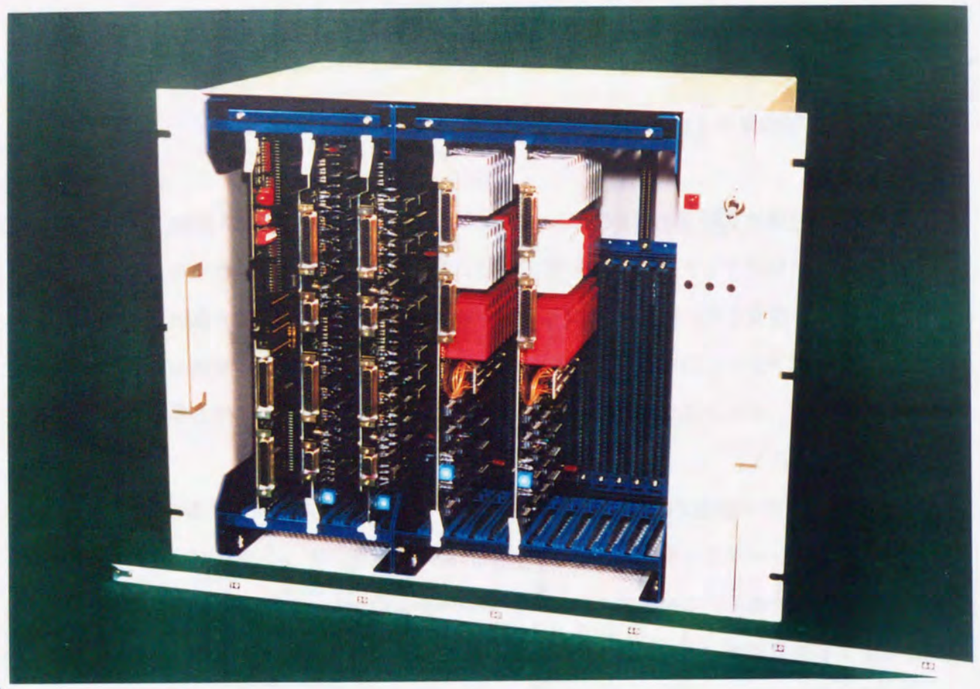


図6.10　試作コントローラ概観

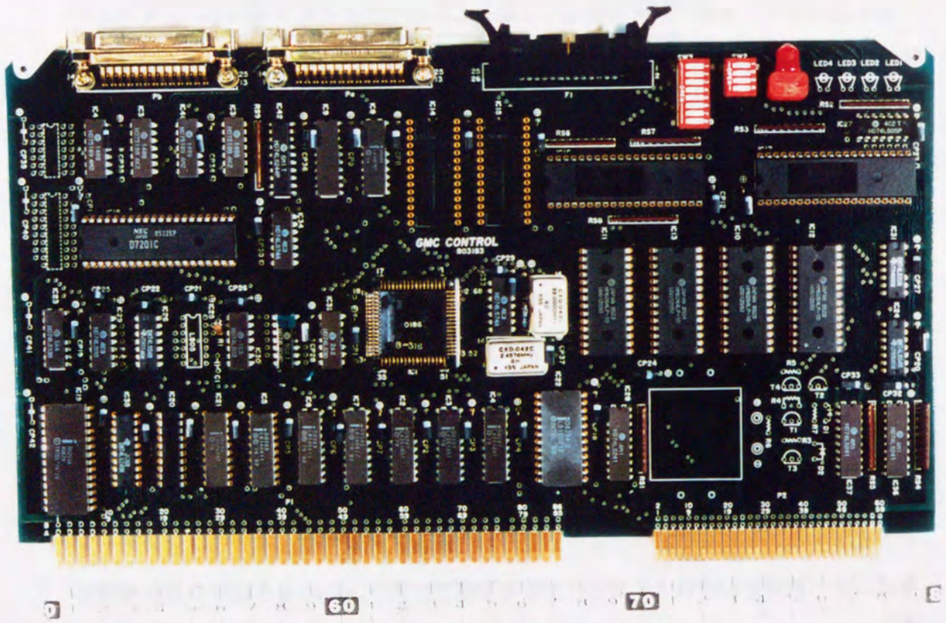


図6.11　CPUボード

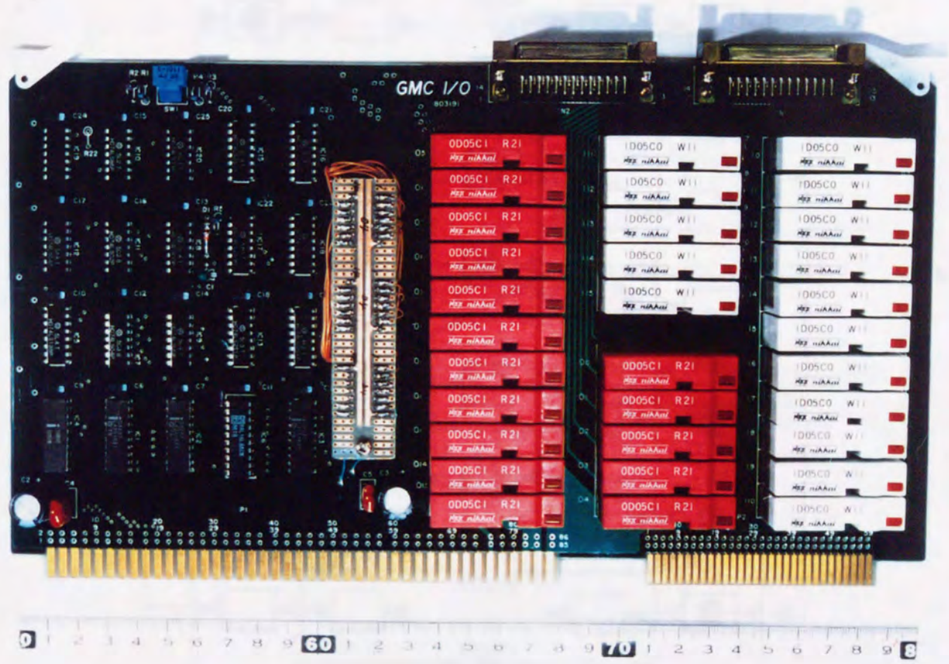


図6.12　I／Oインタフェースボード

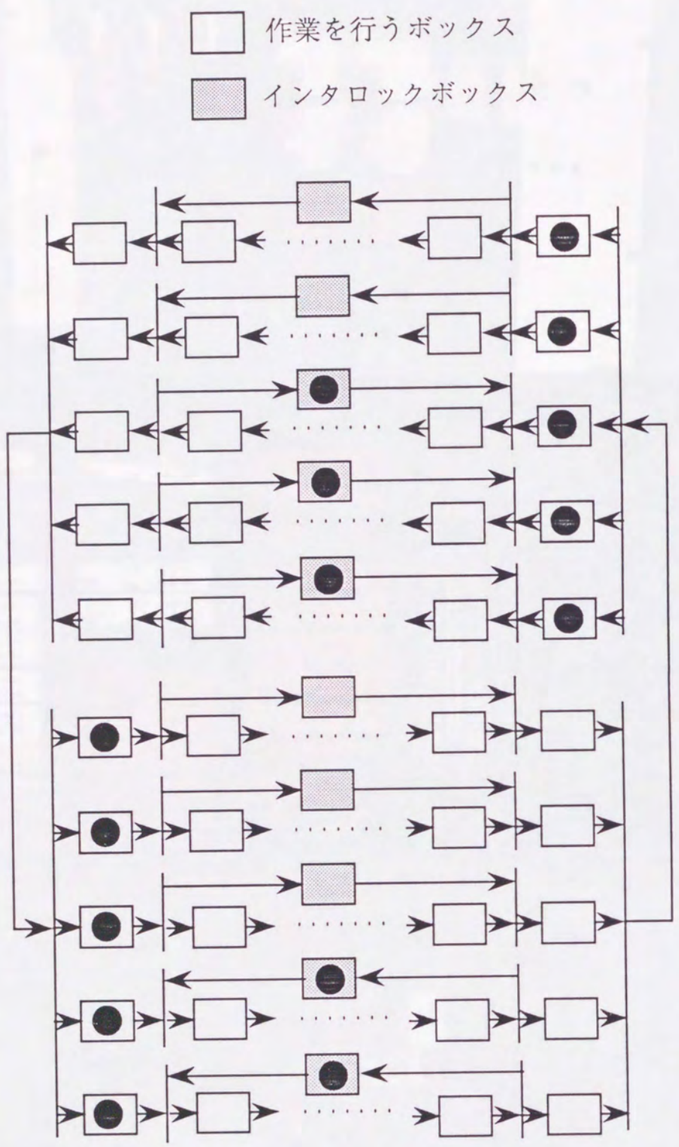


図6.13　シミュレーションに用いたシーケンスシステム

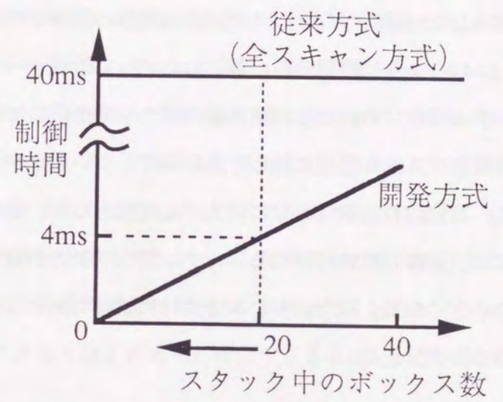


図6.14　各ルーチンの処理時間比

1／10以下に短縮される。実システムを対象とした評価が今後の課題である。

全スキャン方式における一周期の制御速度は，走査するプログラムのステップ数で決定される。したがって制御対象の規模が大きくなると記述に要するプログラム量が増加し，制御速度が遅くなることが問題となる。一方，開発方式では図6.14から分かるように，スタックに登録されているボックス数が制御速度の指標となる。この値は制御対象が並列性，多重性の高いシステムであるかどうかにより異なり，制御実行中にもトークンの分裂，統合により変化するが，システム規模に対して直接依存しないことが特徴である。さらに第6.2.3節で述べたように，保有しているトークンが終了トークンに変わったボックスについては，出力ゲートに関して最後に変化したものを除いてスタックには登録しない方式として，スタックに実際に登録されているボックスを最小化している。これにより，同期待ちが多くしたがって管理すべきトークン数が多いシステムを制御する場合においても，演算量の増大を最小限にできる。

開発方式では制御時間がシステム規模に対して直接依存しないことから，制御対象の規模が拡大したとき，高速化の比率はさらに増大する傾向となる。逆に小規模システムを制御する場合には，全体の演算に占めるトークン管理演算の割合が大きくなるためこの値は必然的に小さくなる。以上より，本方式は大規模システムの制御方式として適した手法であることが分かる。

第6.5節　まとめ

ニューラルネット演算とこれに付随するシーケンス演算，さらにタイマ処理やサーボ制御等の連続系演算を，汎用的に実行するコントローラの高応答化について検討した。まず高応答化のためにはコントローラの演算量を削減するとともに，コントローラが実行するタスクの切り替えを高速に行なう必要があることを述べた。そして前者に関しては，応答性の低下が特に問題となる大規模システムの制御に適した方式として，ペトリネットのトークン遷移を模擬した手法でタスク抽出を行い，現在実行すべきタスクのみを制御の対象とすることでコントローラの演算量を削減し，しかも応答性が制御対象のシステム規模に依存しない方式を開発した。また後者に関しては，ペトリネットのトークンの概念を拡張し，新たに実行トークンと終了トークンを定義した上で，これらを有するボックスが一律に登録されたスタックを走査することによる高速タスク切り替えアルゴリズムを開発した。具体的には，実行トークンを有したボックスのスタックからの排出をボックスで定義された作業の，また終了トークンを有したボックスの排出を関連するトークン移行判定処理の実行タイミングに対応させ，制御対象に対する各種の演算やトークン移行演算を行うタイミングを効率的に設定することで，タスク切り替えのオーバヘッドを最小化し，コントローラの応答を高めることができる。

さらに例外的なトークン管理はタスク処理として扱い，トークン管理の基本アルゴリズムから分離した。基本アルゴリズムの処理を固定したことで，アルゴリズムをハードウェアで置き換えることが容易であり，さらなる高速化が実現できる。また例外処理をボックスの作業として新たに定義したり削除することにより，コントローラの制御能力を柔軟に変えることができる。

開発したシステムを試作システムで評価し，限られた場合についての評価ではあるが，従来の制御方式である全スキャン方式に比べて，コントローラの演算量を一桁以上削減できることが可能なことを示した。

6章の参考文献

（1）山下·榊原·川島，「プログラマブルコントローラ専用ＣＰＵの一構成法」，信学論D-I, J72-D-I, 8, pp.583-591 (1989)

（2）J. L. Peterson, "Petri Net Theory and the Modeling of Systems", Prentice Hall (1981)

（3）辻·熊谷·児玉·黄，「シーケンス制御系のペトリネットによるモデル化と動作検証」，信学技法, CAS pp.85 - 98 (1985)

（4）長谷川·高橋·増田·大野，「非連続システム制御のためのマーク流れ線図の提案」，計測自動制御学論, Vol. 20, 2, pp.122 (1984)

（5）角·児玉·松本·武田，「電力系統におけるスイッチ開閉シーケンスのペトリネットによる解析」，システムと制御, Vol. 25, pp.605 (1983)

（6）薦田·村田，「ペトリネットのＦＡ制御への適用」，システムと制御, 30, pp.42 (1986)

第７章　開発技術の実システムへの適用

第７章　開発技術の実システムへの適用

第７.１節　研究の概要

本章では，ニューラルネットを実システムに適用する場合の方法論，システム構成，従来手法と比べた利点等について，いくつかの実機システムへの適用した結果を基に述べる。

まず走行中の車のナンバプレートを自動的に認識する車番認識システムの認識アルゴリズムをニューラルネットで実現した例を示す。車番認識システムは，ナンバプレートの画像情報に対して適当な濃淡処理を施し，数字や文字の切り出しを行った後，これらを認識する。認識のアルゴリズムとして，従来，画像情報から算出した複数の特徴量を入力とする決定木（デシジョンツリー）が用いられてきた。しかしカメラとナンバプレートの角度のバラツキやナンバープレートの汚れ，昼夜の輝度の変化等に対して普遍的な認識を行うためには膨大なサイズの決定木が必要で，このチューニングは経験豊富なエンジニアをもってしても，数日～数週間を必要としていた。今回開発したのは，数字認識の部分にニューラルネットを用いたシステムで，従来手法とアルゴリズム生産性，認識性能の比較を行い，ニューラルネットを用いた場合の利点を明らかにする。また第２章，第４章で示した最適中間層ニューロン数決定手法の認識アルゴリズム生産性向上に対する寄与についても概算する。

次に熱間圧延システムの加熱炉温度制御にニューラルネットを適用した例を示す。ニューラルネットは温度指令生成時に用いる鋼材の熱モデルの特性を，実際の鋼材と一致させるためのチューニング手段として用いる。加熱炉は圧延工程に先立ち，鋼材を目標温度に精度良く昇温させるプロセスで，実加熱炉から装出される鋼材の温度と加熱炉モデルを用いて算出された温度を比較し，この偏差を減少させる方向にモデルパラメータのチューニングを行う。まず構築した加熱炉制御システムについて説明した後，第５章で開発したアジャスティングニューラルネットを学習するための教師信号の獲得方法，およびチューニングシステムの構成を明らかにする。

最後にワークステーション上で実現した，ニューラルネット開発支援システムの構成を示す。上記２つのシステムを含め，産業システムは複雑・大規模化しており，こ

れらの中で処理の一部にニューラルネットを適用する場合，学習のパラメータを変更したり，教師信号を変更して学習した複数のネットワークで性能の比較を行う等の試行錯誤を必要とする場合が多い。ニューラルネット開発支援システムは，ユーザがこれらを行いやすい環境を整え，ニューラルネット応用システムの立ち上げ時間を短縮することを目的に開発した。とりわけ第２章，第４章で示した最適中間層ニューロン数決定アルゴリズムを組み込み，構造決定に要する試行錯誤を低減した。また第３章で示した学習方法を従来のＢＰと選択的に使用できる環境を提供し，高性能なニューラルネットの構築を容易にした。本章では支援システムの機能，ソフトウェア構成，およびこれを組み込んだ制御システムのアーキテクチャを概説する。

第７.２節　車番認識システム

第７.２.１項　システム構成

開発した車番認識システムでニューラルネットが認識の対象とするのは数字の部分のみであるため，以下，数字認識に限定して説明する。車番認識システムが行う処理の概要は図１.１にも示したように，ＩＴＶカメラにより取り込んだ車の画像からナンバプレートを基に，各数字が何であるかを認識する。今回実用化したのは，認識したナンバプレートを盗難車のナンバプレートが格納されているデータベースと照合するシステムであるが，今後は，建設省により整備が進められているフリーウェイ情報提供システム（車がある区間をどの程度の時間で走行したかを検出し，この結果を基にこれから走行する車がその区間を通過するのに必要な時間を算出，表示するシステム）や，駐車場における自動管理・課金システム等への適用を検討中である。

図７.１にシステム構成図を示す。車番認識システムは，カメラと接続され，濃淡処理による２値化や後述する特徴量の抽出，およびこれを入力とし数字の認識を行う画像認識装置（日立HIDIC - IP/200）と，教師信号の生成，編集，およびこれを用いたニューラルネットの学習を行うワークステーション（同 2050/32）からなる。このような機能分散型のシステム構成により，認識は画像処理専用のハードウェアで，また学習は汎用演算に関して能力の高いワークステーションで，共に高速に実行できる。ワークステーションは，輝度や変形を変えた画像から抽出した特徴量の組みを学習用データとして画像認識装置から取り込み，これと正解の数字をペアにして編集したデータベースを教師信号として学習した後，ネットワークの重みを画像認識装置にセットする。製品化したシステムではワークステーションに上記の処理を行うソフトウェアを新規に作成したが，その後開発した第７.４節で述べるニューラルネットシステム構築支援システムを搭載すれば，さらに作業効率を高めることができる。

ニューラルネットの入力としては，画像を展開した各画素を取り込むことも考えられるが，入力数が膨大となり学習時間，認識時間が増大すること，画像のズレに対する認識率の低下が問題になることから，特徴量を抽出して入力要素とした。特徴量の選定はこれまでの経験，および恣意的にならないことに留意して図７.２で示した12個

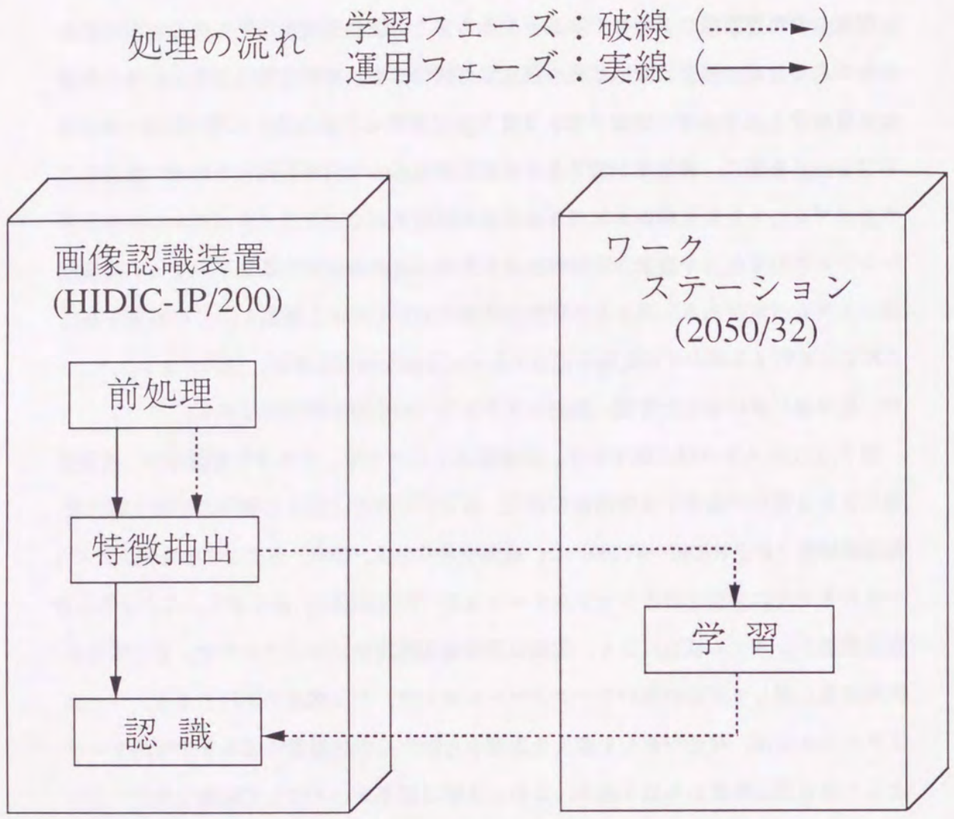
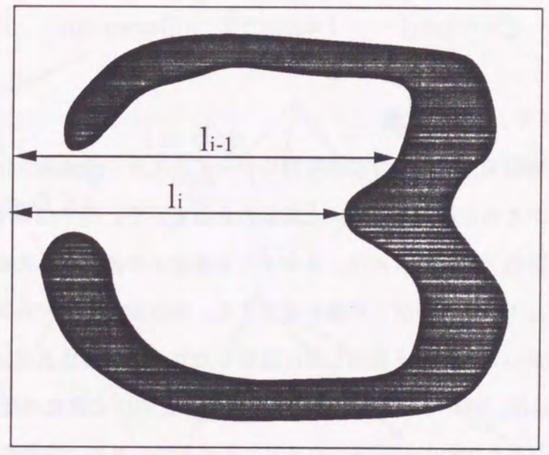


図7.1　システム構成



特徴量	意 味	
(1) hnum	穴の数	
(2) hg08	穴の位置。０と８の区別を行う	
(3) l-c-dep	左中央部の深さの変曲点	Max (l_i - l_{i-1})
(4) t-c-dep	上中央部の深さの変曲点	
(5) lu-dep	左上部の深さの変曲点	
(6) ld-dep	左下部の深さの変曲点	
(7) ru-dep	右上部の深さの変曲点	
(8) rd-dep	右下部の深さの変曲点	
(9) luw-2	左上部の深さの最大値	Max (l_i)
(10) ldw-2	左下部の深さの最大値	
(11) ruw-2	右上部の深さの最大値	
(12) rdw-2	右下部の深さの最大値	

図7.2　ニューラルネットの入力となる特徴量一覧

を用いている。

第7.2.2項　システム性能の比較

アルゴリズム開発期間，認識率，認識時間について，ニューラルネットを用いた場合と，チューニングされた決定木を用いた場合の比較を示す。図7.3に決定木の一部分を示す。従来の認識アルゴリズムは，このような決定木の流れにしたがってカテゴリの可能性を限定していき，最後に結論を導出する。決定木のチューニングとは，各分枝における特徴量のしきい値を最適な値に設定したり，木の構成決定，誤認識を起こした場合の枝の追加，削除等を行うことに対応する。表7.1に性能の比較表を示す。アルゴリズム開発期間と認識率の評価は対で考えるべきで，ともに特徴量が決定され，教師データ（本評価では各数字143個づつの計1430セット）が与えられた状態から，一定水準の認識アルゴリズムが構築できるまでの期間と，このときの認識性能を示している。認識性能は教師とすべきデータに含まれない合計7800セットの評価データで行った。決定木の構築は従事する人の熟練度により大きく変化する。さらに特徴量そのものの追加をチューニングの対象とすると，アルゴリズム開発期間はさらに増大するが，認識率をさらに高められる可能性もある。またニューラルネットに関しても，認識率は教師信号の品質に大きく依存する。さらに誤認識したデータを新たに教師信号に加えて再学習することを繰り返せば，アルゴリズム開発期間は長くなるものの，認識率の更なる向上が可能である。また計算機の演算速度により値は大きく変わる。したがって評価としては必ずしも定量的でなく，性能の一つの目安と考えられる。

認識アルゴリズムとして決定木を用いた場合，本条件下では7～14人·日で，98.6%の認識率が達成されている。これに対しニューラルネットを用いた場合，アルゴリズム開発期間はワークステーションにより学習が収れんするまでの時間である。まず予備学習を行い，第2章で述べた手法で，最適中間層ニューロン数を算出し，この値である6を中間層ニューロン数にして再学習を行う。以上の処理が完了するのに約8時間かかった。またこのときの認識率は99.8%であった。ワークステーションとして用いた2050/32の演算速度は5 MIPS（Mega Instruction per Second）程度であることから，100MIPSクラスの最新ワークステーションを用いれば，30分程度でアルゴリズム開発

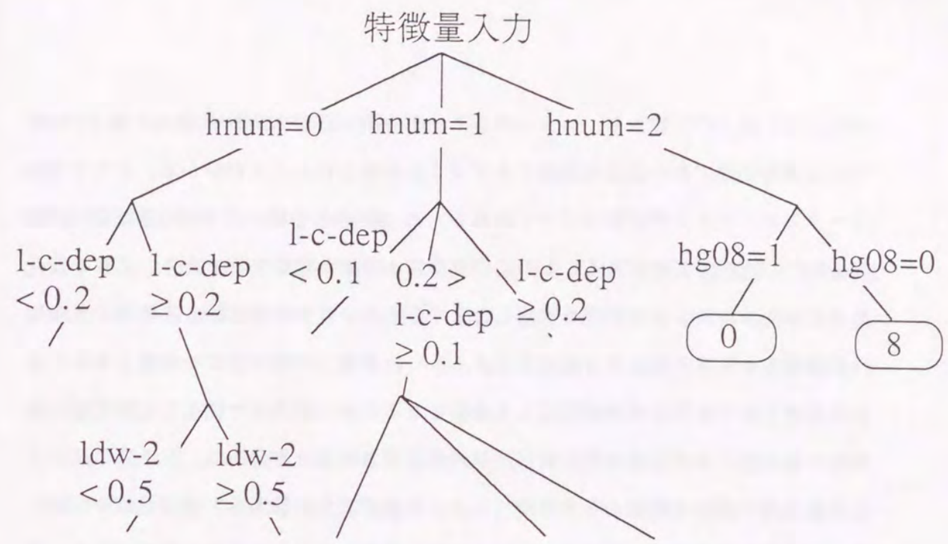


図7.3　決定木の一例

表7.1　ニューラルネットと従来手法の性能比較

手法	デシジョンツリー（従来手法）	ニューラルネット
入力データ	１２種類の特徴量	
アルゴリズム開発期間（人・日）	７～１４	０.３３
認識率	９８．６％	９９．８％
処理時間	５００ms	６６０ms

が行える。以上の比較から，ニューラルネットを用いたシステムでは，少ないアルゴリズム開発期間で高い性能の認識アルゴリズムが得られることが分かる。また学習はワークステーションで自動的に行なわれるため，決定木チューニングの熟練者は必要なくなり，人手を省略できる。さらに中間層ニューロン数が未知であり，これを変化させてシミュレーションで値を決定したとすると，一度の学習にかかる時間約4時間（中間層ニューロン数により値は変化する）に，構築したネットワーク数を乗じた多大な値がアルゴリズム開発期間として必要となるため，本論文で開発した中間層ニューロン数決定アルゴリズムが，実用的に効果の大きいことが分かる。

また処理時間は，画像からナンバプレートを抽出した状態から，濃淡処理を行い，単一の数字を切り出し，特徴抽出を行った後，数字を認識するまでの時間である。決定木法では，認識時間は決定木の深さで決まるが，値はごく僅かである。これに対してニューラルネットを用いた場合，各ニューロンの入出力演算を順に演算するため，処理時間が大きくなる。HIDIC-IP/200 の数値演算能力が低いこともあり，処理時間の増加分は160ms となっているが，この値が実用上問題となる場合には，高速のマイコンを搭載したり，専用のハード回路を用いて各層の演算を並列に行うことにより，高速化を図ることができる。

以上の比較より，ニューラルネットを用いて認識システムを構築する場合，最大のメリットはアルゴリズム生産性の向上である。ただし100%に限りなく近い認識率を要求される用途に適用する場合には，時間をかけた決定木のチューニングで誤認識をことごとく排除していく方が確実な場合もある。したがってニューラルネットは後処理により誤認識を補える用途や，限られた数字の組み合わせのみを対象とした認識を行う用途等で，システムを短時間に立ち上げるのに適した手法と考えられる。例えばフリーウェイ情報提供システムでは，区間の入り口と出口で認識結果が一致した場合のデータのみを用いて情報提供すればよく，しかも何台かの試行から結論を導ける。また駐車場における自動管理・課金システムでは，あらかじめ登録されたナンバプレートの車を対象に，駐車場の専有時間の記録，課金内容の出力を行えばよい。開発手法は，このようなシステムの認識アルゴリズムをすみやかに構築する場合に特に適した方法と言える。

第7.3節　加熱炉制御システム

第7.3.1項　加熱炉と鋼材の熱現象

まず想定した加熱炉と加熱炉制御システムを説明する。図7.4に加熱炉と加熱炉温度制御システムの構成を示す。加熱炉は，通常，温度の異なる4つ程度の炉帯から構成される。加熱炉温度制御システムは，与えられた鋼材の目標装出温度を基に鋼材の熱現象を記述するモデル（鋼材モデル）を参照して各炉帯への炉温指令値を算出する炉温指令発生部，炉温指令と実際に炉帯から検出した温度の偏差から，バーナの燃料流量の増減を制御する燃料流量制御部からなる。鋼材モデルは，熱伝達，熱伝導係数をはじめ，鋼材の材質や炉の構造等に依存したいくつかの定数を有しているが，これらの一部は操業条件や材質のばらつき，種々の検出できないファクタに依存して非線形に変化する。モデルと実際の制御対象との誤差が無視できなくなると温度の指令値が不適切となり，加熱炉から装出される鋼材の温度に誤差を生じる。このような場合，鋼材モデルをチューニングするとともに，システムを監視している操作員が温度指令を手動で変更することになる。

次に，加熱炉内部で想定した熱現象を示す。加熱炉から鋼材への熱伝達要素は，炉壁とスラブ間の輻射伝熱と，対流しているガスからの対流伝熱である。一般に連続鋼片加熱炉では輻射伝熱が支配的となるため，加熱炉から鋼材へ伝わる熱量としては輻射伝熱のみを考慮すればよく，これは次のステファンボルツマン方程式（熱輻射方程式）で記述される。

熱輻射方程式

$$Q = 4.88\phi_{cg}\ [\{(T+273)/100\}^4 - \{(\theta_s+273)/100\}^4] \qquad (7.1)$$

但し，Q：雰囲気中から鋼材が吸収する熱量

Φ_{cg}：総括熱吸収係数（熱伝導係数）

T：炉温

θ_s：鋼材の表面温度

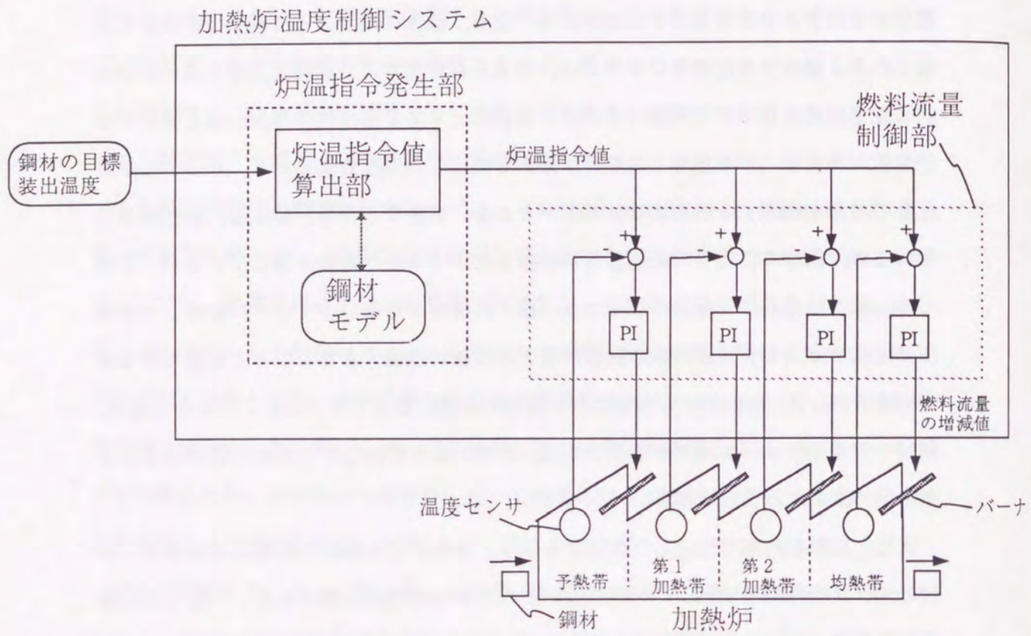


図7.4　加熱炉温度制御システムの構成

また鋼材内部の熱伝達は，以下の熱伝達方程式で記述される。

熱伝達方程式

$$(c_s\rho/k)(\partial\theta/\partial t) = (\partial^2\theta/\partial x^2)+(\partial^2\theta/\partial y^2)+(\partial^2\theta/\partial z^2) \qquad (7.2)$$

但し，c_s：比熱

ρ：密度

k：熱伝達係数

次に炉内の熱収支と，各炉帯のバーナの燃料流量の算出方法を示す。図7.5にI番目の炉帯における熱の流入出の様子を示す。炉帯Iの熱収支はエネルギ保存の法則から次式で与えられる。

$$Q_f(I)+Q_A(I)+Q_g(I+1)-Q_g(I)-\Delta Q_s(I)-Q_L(I)=0 \qquad (7.3)$$

但し，$Q_f(I)$：燃料発熱量

$Q_A(I)$：燃料とともに炉帯に投入される空気の熱量

$Q_g(I+1)$：水蒸気，排ガスの形で炉帯$(I+1)$から流入する熱量

$Q_g(I)$：水蒸気，排ガスの形で炉帯$(I-1)$へ流出する熱量

$\Delta Q_s(I)$：炉帯Iでスラブが吸収した熱量

$Q_L(I)$：炉壁，扉，スキッド等から漏れる損失熱量

であり，それぞれ以下のように定式化される。

$$Q_f(I)=V_f(I)\cdot H_L \qquad (7.4)$$

$$Q_A(I)=V_f(I)\cdot A_r\cdot(t_{pre}-t_{room})\cdot C_A \qquad (7.5)$$

$$Q_g(I)=V_g(I)\cdot(t_I-t_{room})\cdot C_g+V_w(I)\cdot(t_I-t_{room})\cdot C_w \qquad (7.6)$$

$$\Delta Q_s(I)=\{\theta_I-\theta_{I-1}\}C_s\cdot M(I) \qquad (7.7)$$

$$Q_L(I)=\eta_L\cdot t_I+\xi_L \qquad (7.8)$$

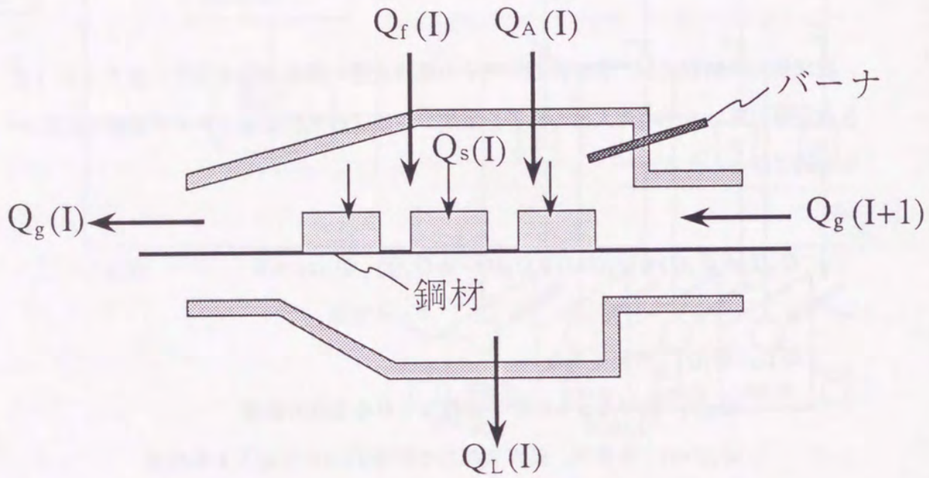


図７.５　炉帯Ｉにおける熱収支

ここで，H_L：単位熱量発熱量

A_r：空然比

$V_g(I)$：炉帯から流出する排ガスの体積

$V_w(I)$：炉帯から流出する水蒸気の体積

t_{pre}：燃焼に供する余熱空気温度

C_g：排ガスの比熱

C_w：水蒸気の比熱

C_s：スラブの比熱

$M(I)$：全スラブ重量

η_L：炉帯熱損失係数

ξ_L：排ガス温度係数

また単位燃料あたりの燃焼によって新たに生じる気体の体積をG_0，水蒸気の体積をW_0，燃焼に伴って消費される空気体積をA_0とすると，流入ガス中の排ガスの体積，水蒸気の体積は次の様になる。

$$V_g(I)= V_g(I+1)+ V_f(I)\cdot(G_0 - A_0 + A_r) \tag{7.9}$$

$$V_w(I)= V_w(I+1)+ V_f(I)\cdot W_0 \tag{7.10}$$

ここで右辺第１項は流入してきた排ガスおよび水蒸気の体積，右辺第２項は炉帯Iで新たに発生したこれらの体積である。加熱炉が４帯の炉で構成されている場合であれば，$V_g(I)$および$V_w(I)$が０になることを考慮して（７.４）～（7.10）式を（７.３）式に代入すると，バーナで消費される全燃料流量Jは次式で与えられる。

$$J = \sum_{I=1}^{4} v_f(I) = \frac{\sum_{I=1}^{4} \Delta Q_s(I) + \sum_{I=1}^{4} Q_L(I)}{H_L + A_r(t_{pre} - t_{room})C_A - C_{gw}(t_1 - t_{room})} \tag{7.11}$$

ここで明らかにした全燃料流量 J は，次節で述べる炉温指令値決定に用いられる。

第7.3.2項　炉温指令値設定方法

各炉帯には，仕様として与えられた内外温度差（鋼材の表面と中央の温度差）の基で鋼材を目標温度まで適切に昇温させるための炉温指令値が設定され，各バーナの燃料流量はこれを指令値として制御される。加熱炉が複数の炉帯で構成される場合，目標を満足させる炉温指令値の組は多数存在するため，前節で述べた J の最小化を新たに目標として設定し，これらに対応した炉温指令の組を最適炉温指令と定義した。すなわち炉温指令最適パターンは，最小の燃料流量で鋼材を目標温度へ昇温させる組み合わせである。最適炉温指令を決定する問題は，制約付き最適化問題に定式化できる。すなわち非線形関数である（7.11）式を評価関数とし，以下を制約条件として解くことができる。

$$(T_1)_{min} \leqq T_1 \leqq (T_1)_{max} \tag{7.12}$$

$$(T_2)_{min} \leqq T_2 \leqq (T_2)_{max} \tag{7.13}$$

$$(T_3)_{min} \leqq T_3 \leqq (T_3)_{max} \tag{7.14}$$

$$(T_4)_{min} \leqq T_4 \leqq (T_4)_{max} \tag{7.15}$$

$$(\theta^*_{out})_{min} \leqq \theta^*_{out} \leqq (\theta^*_{out})_{max} \tag{7.16}$$

$$\Delta\theta_{out} \leqq (\Delta\theta_{out})_{max} \tag{7.17}$$

$$|T_I - T_{I-1}| \leqq \zeta_I \tag{7.18}$$

$$0 \leqq V_f(1) \leqq (V_f(1))_{max} \tag{7.19}$$

$$0 \leqq V_f(2) \leqq (V_f(2))_{max} \tag{7.20}$$

$$0 \leqq V_f(3) \leqq (V_f(3))_{max} \tag{7.21}$$

$$0 \leqq V_f(4) \leqq (V_f(4))_{max} \tag{7.22}$$

ここで，$(T_I)_{min}$：炉帯 I の温度下限値

$(T_I)_{max}$：炉帯 I の温度上限値

$(\theta^*_{out})_{min}$：目標抽出温度の下限許容値

$(\theta^*_{out})_{max}$：目標抽出温度の上限許容値

$\Delta\theta_{out}$：内外温度差

$(\Delta\theta_{out})_{max}$：内外温度差の許容値

ζ_I：炉帯 I と炉帯(I-1)の温度差の上限値

$(V_f(I))_{max}$：炉帯 I の燃料流量の上限値

このような制約付き非線形最適化問題の解法はいくつかあるが，本システムではコンプレックス法で解を求めた。コンプレックス法は以下の手順で計算が進められる。まず T_1～T_4 からなる4次元のベクトル空間を定義し，この中で制約条件を満足する炉温の組を複数個用意する。次に各点で J を計算し，最悪点を抽出する。そして最悪点を残りの点の重心の方向に更新し，上記の操作を繰り返す。更新した点がやはり最悪点であったり，境界条件を満足しない場合には，更新量を変えて同様の操作を繰り返す。最終的に複数の炉温ベクトルが互いに近傍に集まった時点で演算を終了し，得られた炉温を温度指令値として各炉帯に出力する。

第7.3.3項　鋼材モデルのパラメータチューニングシステム

前節で述べた温度指令値は，各炉帯における鋼材の温度上昇を（7.1），（7.2）式の鋼材モデルに従って計算しているため，高精度な温度指令算出には鋼材モデルの精度を高めることが必要である。そこで加熱炉炉温指令発生システムには，第5章で述べたＡＪＮＮを核とした鋼材モデルのパラメータチューニングシステムを設けた。本システムでは第5章で述べた理由で，モデル誤差を Φ_{cg} に集約し，抽出温度誤差が減少する方向に Φ_{cg} を修正する。図7.6に加熱炉炉温指令発生システムの構成を示す。パラメータ修正部は抽出温度誤差と抽出温度が検出されたときの状態変数から，鋼材モデルの Φ_{cg} 修正量を出力する。状態変数として板厚，鋼材の挿入温度，各炉帯の温度を考慮した。パラメータ修正部として，シミュレーションベーストパラメータ修正部と実プラントベーストパラーメータ修正部を並列に設けたことが特徴で，どちらもＡＪＮＮにより構成されている。前者は加熱炉モデルを用いたシミュレータを用い，種々の場合についての鋼材の温度誤差と Φ_{cg} 修正量の関係をデータベース化した後，

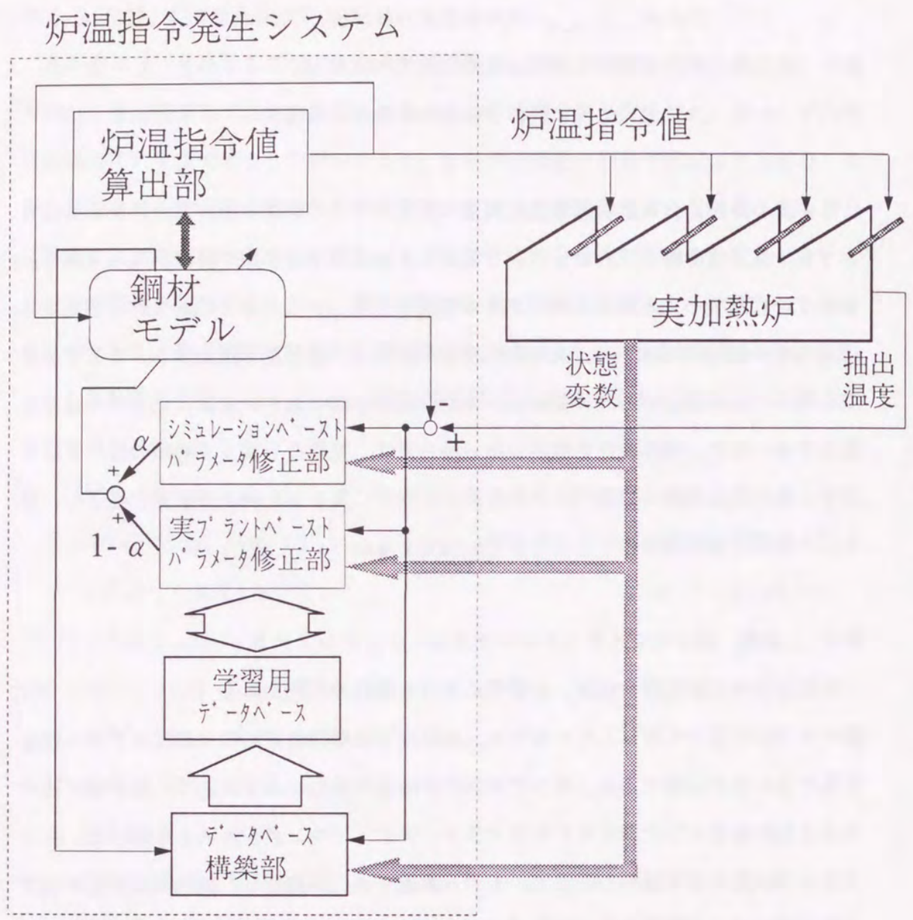


図７.６　加熱炉制御システムの構成

これを教師信号としてシステムの立ち上げに先立って学習したネットワークが備えられている。後者は実際のプラントから得た抽出温度誤差と状態変数を入力，チューニングの最終結果として得られたΦ_{cg}を対応した教師出力とした学習用データベースに基づいたＡＪＮＮで処理が行なわれる。両者の出力を重み付け加算することにより，最終的なΦ_{cg}修正量が算出される。学習用データベースには，プラントが稼働し，チューニングが行なわれる度に，一つないし数個づつ教師信号が蓄えられる。実プラントベーストパラーメータ修正部は，これを学習してネットワークの出力精度を高める。このため，プラントの立ち上げ初期はαを１にし，シミュレーションベーストパラメータ修正部の出力により，Φ_{cg}を修正する。一方，シミュレーションベーストパラメータ修正部の出力はシミュレータの加熱炉モデルと実加熱炉の不一致による本質的な誤差を有しており，これは不変である。このため学習用データベースに多様な実機データが蓄えられた後は，これを用いて構築した実プラントベーストパラーメータ修正部の出力の方が信頼性が高い。このためαを小さくし，この影響を高める処理を行なう。実際には，実加熱炉からの入力に類似のデータが学習用データベースに含まれる場合に，αを小さくしてΦ_{cg}修正量の精度を高める。さらにシミュレーションベーストパラメータ修正部と実プラントベーストパラメータ修正部のαが一定以上隔たっていた場合には，シミュレーションベーストパラメータ修正部の出力値を最終的なΦ_{cg}の修正量とすることで，誤った値が出力されることを防いでいる。

以上述べた２つのパラメータ修正部を備えたモデルチューニングのアーキテクチャにより，プラントの立ち上げ初期にはシミュレータが実加熱炉を模擬している範囲で高精度な，またプラントが稼働した後は，チューニングの実績を反映したΦ_{cg}の修正量が算出でき，信頼性の高いモデルチューニングを行うことができる。

第7.4節　ニューラルネット構築支援システムの構成

第7.4.1項　ニューラルネット構築支援システムのコンセプト

ニューラルネット構築支援システムは，圧延，電力，化学，交通，公共等の産業プラントにおいて，ニューラルネットを組み込んだシステムを構築する場合に，ワークステーションベースでこれを支援するツールである。支援システムは，以下の4点をコンセプトとして開発した。

（1）生産性向上の支援：ニューラルネット応用システムの構築時間短縮を目的とした支援機能を設けた。具体的には，本論文で述べた中間層ニューロン数決定アルゴリズムをサポートし，この値を決めるのための試行錯誤を省略させた。また同一の教師信号に対応して，異なった構造や重みの初期値で構築したニューラルネットの汎化能力や汎化特性を，グループ化して一つのファイルにまとめる機能を備え，過去の試行を容易に検索できる構成とした。

（2）信頼性向上の支援：ニューラルネットの信頼性を定性的にではあるが評価できる環境を提供した。すなわち誤出力する可能性をチェックする機能として，認識を行うニューラルネットでは，誤認識した場合にどのクラスをどのクラスに間違えたかを一覧として表示する機能を備えた。また関数近似を行うニューラルネットでは，誤差が一定値以上の入力データを列挙する機能を備え，ネットワークの特性を観察できるようにした。

（3）性能向上の支援：複数の学習アルゴリズムを備え，これらを選択的に使用可能とした。学習方式として，通常のバックプロパゲーションの他に，本論文の第3章で述べた教師入力信号に乱数を重畳させて入力する方式，後述する共役勾配法を応用した学習でニューラルネットの出力誤差を低減させる方式等を備え，問題に応じてこれらを選択的に用いる環境を提供した。

（4）拡張性向上の支援：上述したプラントで頻繁に用いられる，ファジイ処理や知識処理，数値解析処理等とリンクした実行系の構築を容易化した。すなわちニューラルネットの処理ルーチンをサブルーチン化し，その他の処理からの呼び出しが可能な構成とした。またデータベースのフォーマットを，これらの処理の間で統一した。

第7.4.2項　ニューラルネット構築支援システムのソフトウェア構成

ニューラルネット構築支援システムのソフトウェア構成を図7.7に示す。本論文で開発した内容は網掛部分であり，他のニューラルネット構築装置に対し本支援システムを特徴付けるための中核機能となっている。

ネットワークおよびデータに対して，学習エンジンは図中の種々の学習方法のいずれかにより，重みの更新を行う。通常のＢＰ法に対して，ＤＢＰは第3章記載の学習方法で，教師信号に適当な乱数を重畳することにより，クラスタリングを行なうニューラルネットの認識性能を高める方式である。共役ＢＰは重みの更新を総ての教師信号の誤差に対して共役勾配方向に更新する方式で，ニューラルネットが学習する対象により，学習回数の低減や教師信号の近似精度向上が実現できる。また適応ＢＰは，（2.6）～（2.9）式の学習係数（誤差と重みの更新量の割合を与える定数）を学習の進行（あるいは誤差の減少）と共に適切な割合で低下させることにより，やはり教師信号の近似精度を高める方式である。

実行エンジンは，ある入力に対してニューラルネットの演算を行い，出力を算出する。同時にニューラルネットを，層数と各層のニューロン数で構成される構造データと重みデータからなる実行モジュールとしてサブルーチン化する。さらに次節で述べるように，必要に応じてこのサブルーチンをＣのソースに変換する。

設計エンジンは，学習済みのニューラルネットを用いて，本論文の第2章で述べた最適中間層ニューロン数決定アルゴリズムを実行し，算出された最適中間層ニューロン数を提示する。さらに文献1で示されている統計解析や，文献2の感度解析結果を基に，不要な入力ニューロンを提示し，ニューラルネットのコンパクトな設計を支援する。評価エンジンは，教師信号に対する学習誤差や，未学習信号に対する汎化能力を評価する。さらに前述したように，誤認識した場合にどのクラスをどのクラスに間違えたかを一覧として表示したり，誤差が一定値以上の入力データを列挙する機能を備え，ニューラルネットの特性が観察できる機能を設けた。

構築支援ファイルとしては代表的なもののみを示す。組み込み実績ファイルには，ネットワークの構造や重みの初期値，学習プロセス（適用した学習方法，学習に要した回数，学習パラメータの値，誤差の減少過程等）が，アプリケーション毎に記憶さ

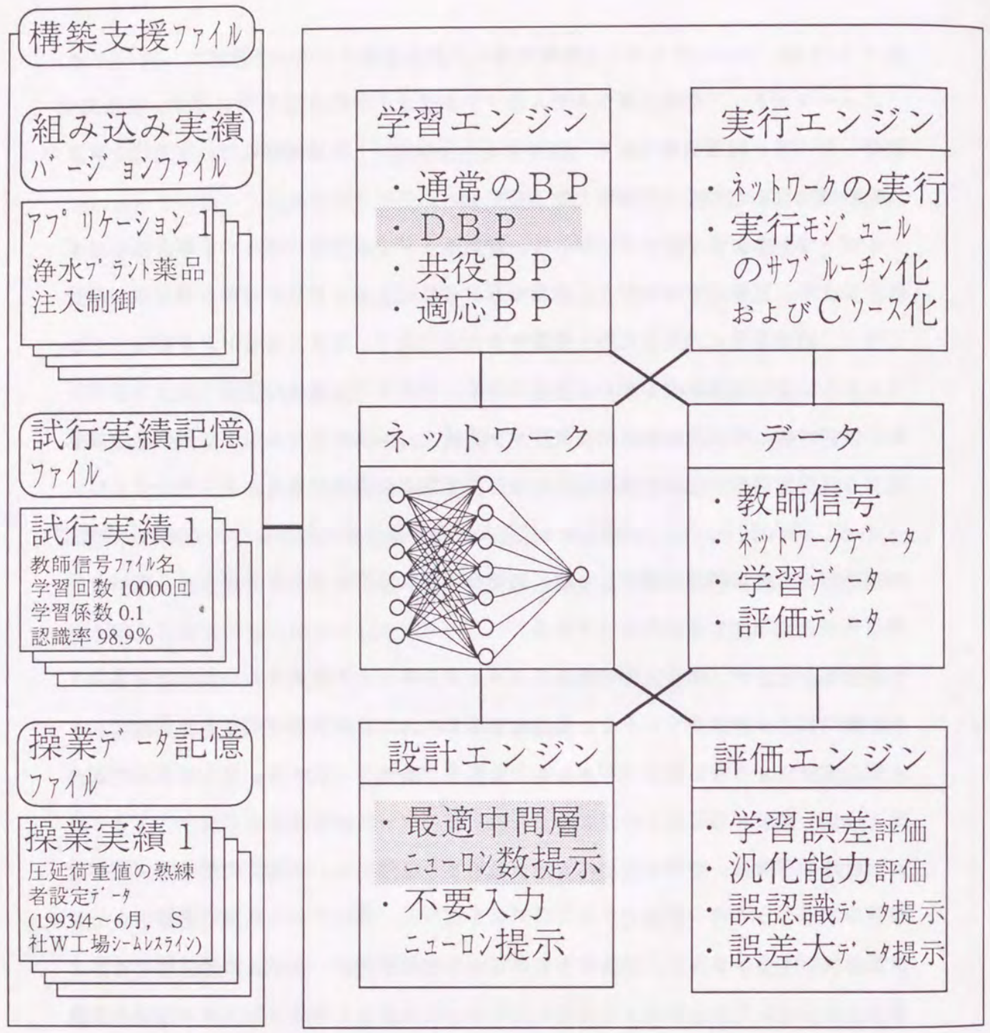


図7.7　支援システムのソフトウェア構成

れている。この情報を基に対象で稼働しているネットワークを忠実に再現したり，システム更新時に前回の情報を検索することが容易になる。試行実績記憶ファイルは，構築されたネットワークに対して，教師信号を特定する情報や，ネットワークの構造，学習プロセス，ネットワークの性能等をひとまとめにして記憶している。同一の教師信号に対し種々の条件でネットワークを構築し性能評価を行う場合に，これまでの試行の検索を容易にし，試行の重複や欠落を避けることができる。操業データ記憶ファイルは，現場から採取した操業のデータを，操業条件や操業時期，操業場所との対応で記憶する。この他にもユーザ定義により任意の支援ファイルが構築できる。

以上の機能に対し，ユーザは Motif 上のグラフィックなユーザインタフェースでアクセスでき，ニューラルネット応用システムの立ち上げや保守が効率化される。

第7.4.3項　ニューラルネット構築支援システムを組み込んだ制御システムの構成

図7.8に，ニューラルネット構築支援システムが制御システムに組み込まれた場合のシステム構成の一例を示す。制御システムは，上位の基幹制御ＬＡＮと下位の制御ＬＡＮを核として構成される。支援システムは，制御対象のシミュレーションやこの結果を基に圧延材の形状や温度情報をトラッキングするプロセスコンピュータや，コントローラのプログラミング・モニタ装置であるＰＯＣ（Programming Operating Console）とともに，基幹制御ＬＡＮに接続される。一方，下位の制御ＬＡＮには，制御演算を実行したり制御対象と信号の授受を行うコントローラやＩ／Ｏ（Input / Output）が接続されている。

このような制御システムの中で効果的に稼働させるための機能として，支援システムには以下の機能を備えた。

（1）基幹制御ＬＡＮへの接続機能：学習したネットワークの情報（構造，重み）をコントローラへ転送したり，Ｉ／Ｏから得た制御対象の操業データを前述した操業データ記憶ファイルに格納することを，容易でしかも高速に行うことを目的として，基幹制御ＬＡＮへの接続を可能とした。したがってこれらの処理をＬＡＮを介して高速に実行できる。

（2）学習済みのニューラルネットをＣコードへ変換しパッケージ化する機能：学

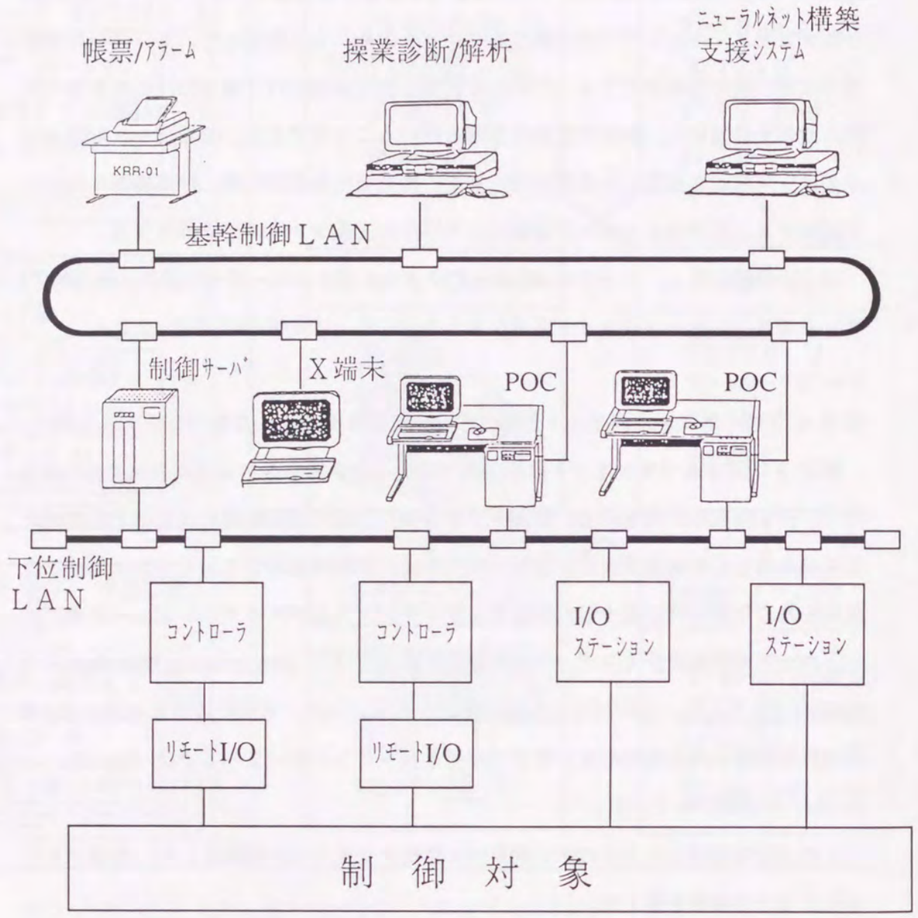


図７.８　ニューラルネット構築支援システムが組み込まれた制御システム

習結果を図７.７の実行エンジンでＣソース化する機能と対応しており，ニューラルネット演算をＰＯＣで作成する制御プログラムに埋め込み，他の制御処理と関連づけて実行することが容易となる。図７.９は，ニューラルネット演算を行うＣプログラムのパッケージを制御演算とリンケージする機能の模式図である。図のように，第６章で述べたペトリネットベースの制御プログラム，および現行広く用いられているラダープログラムのいずれの場合も，リンケージは容易である。リンケージされた結果は，基幹制御ＬＡＮ，下位制御ＬＡＮを介して，コントローラに転送され，実行される。これによりニューラルネット演算の実行タイミングを，シーケンス演算の中で自由に設定でき，ニューラルネット演算結果をシーケンス制御演算に反映させることも，特に工夫なく行える。

（３）出力先のパッケージに対して重みのデータのみを差し替える機能：再学習等によりニューラルネットの重みが変更される度に，ネットワークの転送，シーケンスプログラムとのリンケージを行う手間を省略することを目的としており，パッケージの転送先にアクセスし，重みのデータのみの差し替えを行う。実際には，支援システムは転送先のコントローラにおけるパッケージの先頭アドレスを，ＰＯＣからの返信で把握しておき，これを起点にニューラルネットのプログラムが占有するエリア中で，重みデータに相当する部分の差し替えを行う。これによりプラントの特性が変化した場合等においても，同一構造のニューラルネットで重みのみを変更する場合であれば，コンパイルやリンク演算なしで，短時間にニューラルネットの演算内容を変更できる。

以上述べた機能の大半を備えたニューラルネット構築支援システムは，ＥＵＲＩＫＡ Ⅲ（Electronic Understanding and REasoning by Knowledge Activation）の名で製品化され，圧延プラント，化学プラント等で稼働中であり，ニューラルネット応用システムの構築，保守の効率化に大きな効果を発揮している。

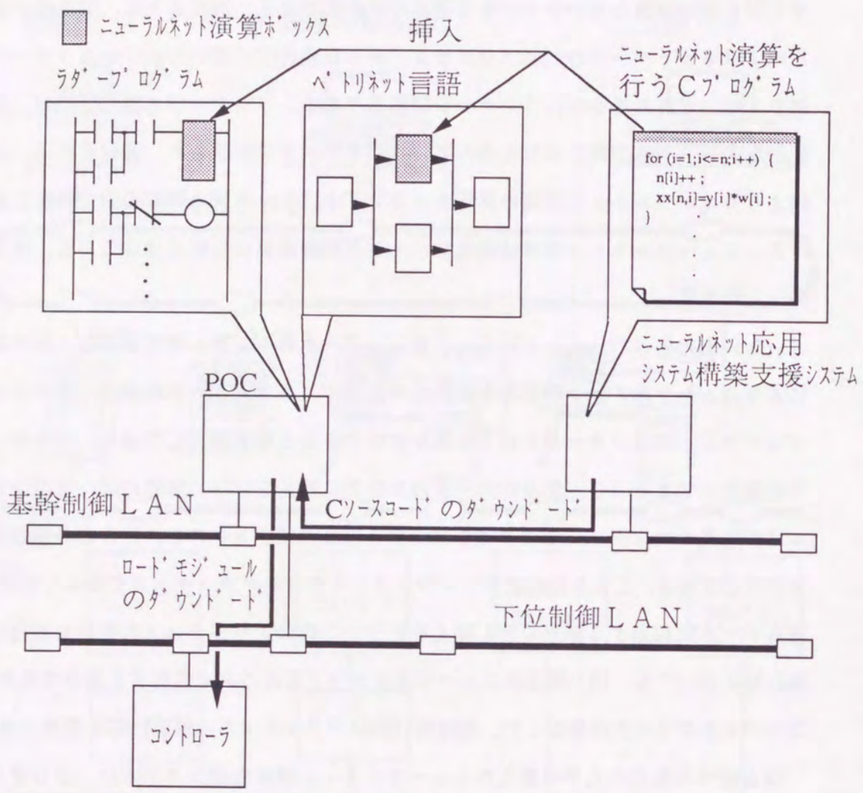


図７.９　ニューラルネット演算プログラムと
制御プログラムのリンケージ

第７.５節　まとめ

本章では，本論文で提案した技術を現実のシステムに実装する場合の方法論を，車番認識システム，圧延制御システム，ニューラルネット開発支援システムを例に示し，開発技術の有用性を明らかにした。

ニューラルネットは，車番認識システムにおいてはソフトウェア生産性の向上に，また圧延制御システムのモデルチューニングにおいては高速なチューニングの収れんに威力を発揮する。第２章，第４章で開発した技術を適用すれば中間層ニューロン数の決定が迅速化されるため，ネットワーク構築の試行錯誤を軽減でき，システム立ち上げの労力が軽減される。さらに車番認識システムでは，第３章で述べた教師信号に乱数を印加する学習方式により認識率の向上が，モデルチューニングでは，第５章で示したＡＪＮＮにより定常誤差を低減した高精度なチューニングが実現できる。さらに汎用のニューラルネット開発支援システムに本論文で開発した技術を適用した場合には，汎化能力を高めることを目的とした学習の支援，迅速な構造決定の支援が可能となり，使用者に大きな利点を与えることができる。

7章の参考文献

（1）M.Kayama, et. al., "Constructing Optimal Neural Networks by Linear Regression Analysis", Neuro Nimes'90, pp.363～376 (1990)

（2）武長·阿部·高藤·鹿山·北村·奥山，「感度解析を用いたニューラルネットの入力層の最適化とその数字認識への適用」，電学論D, Vol.111, 1 , pp.36-44 (1991)

第8章　結　　論

第8章　結　　論

本論文では，産業システムの制御・認識に多層ニューラルネットを適用する場合の方法論を，ネットワークの構成の最適化，学習方法，制御装置における実行方式の観点から検討した。また検討結果を用いて実際の制御・認識システムを構築した例として，車のナンバプレートを自動認識する車番認識システム，圧延プラントにおける加熱炉温度制御システムを示し，システム構成，従来技法との比較結果を明らかにした。さらにニューラルネット構築支援ツールの機能，ソフトウェア構成，および開発技術の適用形態について述べ，本論文で提案した技術が開発した支援ツールを特徴あるツールとして位置づける上で大きく貢献していることを示した。本論文で得られた主な結論を以下に要約する。

（1）多層ニューラルネットの中間層の最適ニューロン数を“最大汎化能力を実現する最少数”と定義し，簡易な手法でこれを決定するアルゴリズムを開発した。具体的には，学習済みのニューラルネットに教師入力信号を再入力したときの中間層ニューロンの動作に含まれる線形成分を求め，この値に対応するニューロン数をニューラルネットに初期設定されているニューロン数から除くことが有効なことを示した。さらにクラスタリング問題を対象に，中間層ニューロン数と汎化能力の関係を記述する簡易なモデルを提案し，汎化能力が中間層ニューロンの増加とともに増大しやがて飽和する挙動の一般性を明らかにした。

開発手法を用いれば，中間層ニューロン数を決定するために1回の予備学習は必要になるものの，従来行われていた試行錯誤的なシミュレーションは不要となり，ニューラルネット応用システムの構築時間を短縮できる。

（2）パターン認識を行うニューラルネットの汎化能力最大化を目的に，学習方法の改善を提案した。教師入力信号に，その都度発生させた適当な振幅の乱数を重畳することが認識能力向上に有効なことを述べ，そのメカニズムを明らかにした。さらに振幅値を学習の進行とともに減少させ、最終的に0にすると，クラスタ同士の相対距離

や，認識すべきデータに加わっているノイズの大きさに依存しない，高性能なネットワークが得られることを示した。開発手法では，教師信号が十分に揃えられていない場合に従来学習法からの改善が顕著であり，少数の代表データのみを用いて素早く認識システムを立ち上げたい場合，あるいは多数の教師信号を収集するのが時間的，コスト的に困難な場合に特に有効な方法である。

（3）制御モデルのパラメータチューニングを高速・高精度に行うことを目的に，従来のニューラルネットを拡張したアジャスティングニューラルネット（ＡＪＮＮ）を提案した。ＡＪＮＮは，通常のニューラルネットに，これと同一の構成および重みを有した誤差算出用ニューラルネットを並列に結合したアーキテクチャを備えており，通常のニューラルネットの出力から誤差算出用ニューラルネットの出力を減じた値を，最終的に出力する。誤差算出用ニューラルネットの出力は制御モデルが正しいときの通常ニューラルネットの出力誤差に対応しており，通常ニューラルネットの出力からこの値を減じる処理によりチューニングの定常偏差を０にできる。さらに，バックプロパゲーションに用いる誤差関数を並列ネットワーク向けに修正した誤差関数を新たに定義し，これを用いた並列ネットワークの学習と，誤差算出用ニューラルネットの出力を０に低減する学習を組み合わせた，ＡＪＮＮの学習方式を提案した。

ＡＪＮＮを加熱炉温度制御における鋼材モデルのパラメータチューニングに適用したシミュレーションで，現在広く用いられている線形式によるチューニングや従来のニューラルネットを用いたチューニングに比べ，少ない回数で定常温度誤差の小さいチューニングが行えることを示した。

（4）ニューラルネットを搭載したコントローラの性能を総合的に高めることを目的に，ニューラルネットに付随する処理（シーケンス制御演算，ループ制御演算）を高速化し，しかも各処理の切り換えを効率化することを検討した。具体的には，ペトリネットのトークン遷移を模擬した手法でタスクの抽出を行い，現在実行すべき処理のみを制御の対象とすることで，コントローラの演算量を削減する手法を開発した。開発方式ではペトリネットのトークンの概念を拡張し，新たに実行トークンと終了トー

クンを定義する。そしてトークンが存在するボックスをスタックに登録し，実行トークンを有したボックスの抽出をボックスの処理の，また終了トークンを有したボックスの抽出を関連するトークン移行判定処理の実行タイミングに対応させる。これにより，トークン移行処理に要する演算を最小化できるとともに，ニューラルネット演算を含む各種処理の実行タイミングやトークン移行処理のタイミングを時分割の中で効率的に設定できる。

開発手法を試作システムで評価し，従来の全スキャン方式に比べてシーケンス制御の応答時間を一桁以上高速化できることが可能なことを示した。

（5）本論文で提案した技術を現実のシステムに実装する場合の構成を，車番認識システム，圧延制御システム，ニューラルネット開発支援システムを例に示し，従来技術との比較を通して開発技術の有用性を明らかにした。

ニューラルネットは，車番認識システムにおいてはソフトウェア生産性の向上に，また圧延制御システムのモデルチューニングにおいては高速なチューニングの収れんに威力を発揮する。開発技術の適用により中間層ニューロン数の決定が迅速化されるため，ネットワーク構築の試行錯誤を軽減できる。さらに車番認識システムでは，教師信号に乱数を印加する学習方式により認識率の向上が，モデルチューニングでは，ＡＪＮＮにより定常誤差を低減した高精度なチューニングが実現される。また汎用のニューラルネット開発支援システムに開発技術を適用した場合には，汎化能力を高めることを目的とした学習の支援，迅速な構造決定の支援が可能となり，使用者に大きな利点を与えることができる。

本論文で提案した技術により，ニューラルネットを応用展開する場合の課題のいくつかが解決できた。ニューラルネットの適用分野は，論文中に述べた制御，認識装置から，電力やエネルギの需要，株価等の予測装置，プラントの異常診断装置，家電品等，産業の各分野に広がっている。本論文の成果はこれらを開発する場合の期間短縮，性能向上に大きく貢献できるものと考えている。

関 連 論 文

本論文主題に関する著者の主要な発表論文

2　章

（1）鹿山，阿部，武長，諸岡：多層ニューラルネットの最適中間層ニューロン数決定方法，電気学会論文誌D，112巻11号(1992)

（2）M.Kayama, S.Abe, H.Takenaga, and Y.Morooka："Constructing Optimal Neural Networks by Linear Regression Analysis", Neuro Nimes'90, pp.363～376 (Nimes, France, 1990)

（3）鹿山，阿部，武長，諸岡：線形回帰分析による多層ニューラルネットの中間層ニューロン数決定方法，情報処理学会第41回（平成2年後期）全国大会，pp.2-107～108 (1990)

3　章

（1）鹿山，阿部：汎化能力向上を目的としたクラスタリング用ニューラルネットの学習方式，電子情報通信学会論文誌D-II，Vol.J76-D-II, No.4(1993)

（2）鹿山，阿部：クラスタリング用ニューラルネットの学習方式，情報処理学会大44回（平成4年前期）全国大会，pp.2-215～216(1992)

（3）鹿山，阿部，諸岡：多層ニューラルネットの汎化能力評価指標，平成5年電気学会全国大会 (1993)

（4）鹿山，阿部，諸岡：パターン認識を行うニューラルネットの汎化能力最大化学習方式，電気学会産業システム情報化研究会 (1993)

4　章

（1）鹿山，阿部，武長，諸岡：多層ニューラルネットにおける汎化能力の中間層ニューロン数依存性，電気学会論文誌D，113巻3号(1993)

（2）M.Kayama, S.Abe, H.Takenaga, and Y.Morooka："Modeling Generalization Capability for a Multi-layered Neural Network Classifier and Optimizing Its Number of Hidden Units", IJCNN'92-Beijing, I-429～434 (Beijing, China, 1992)

（3）M.Kayama, S.Abe, H.Takenaga, and Y.Morooka："Constructing Optimal Neural Network Classifier for Real-time Recognition Systems",Annual Conference of IEE-Japan, E.87～92 (1992)

5　章

（1）M.Kayama, Y.Sugita, Y.Morooka, and Y.Saito："Adjusting Neural Networks for Accurate Control Model Tuning",FUZZ-IEEE/IFES'95, (Yokohama, Japan, 1995)

6　章

（1）鹿山，長瀬：ペトリネットのトークン遷移を適用した自動機械システムの高速制御，電気学会論文誌D，109巻7号，p463～469(1989)

（2）M.Kayama, H.Nagase："A High Performance Programmable Controller for CIM Systems Based on Petri Net Theory", IECON'89,p805～810 (Philadelphia, USA, 1989)

（3）鹿山，長瀬：ペトリネットを適用した自動機械システムの制御方式，昭和63年電気学会全国大会，pp.2094～2095 (1988)

（4）鹿山，長瀬，諸岡：ペトリネットを適用したプログラマブルコントローラの制御機能拡張方法，平成元年電気学会全国大会，pp.1726～1727 (1989)

謝　辞

謝　辞

本論文をまとめるにあたり，終始熱心に御指導下さいました大阪大学教授宮崎文夫博士を始め，同教授小倉敬二博士，同教授小坂田宏造博士に深甚なる謝意を表する。

また，本研究は（株）日立製作所日立研究所および電力・電機開発本部における業務の一環として行ったもので，研究の機会を与えられ研究の推進にあたって種々御指導いただいた日立研究所副所長川上潤三博士，電力・電機開発本部主管研究員諸岡泰男博士，同電機第１部長松田靖夫博士，日立研究所主管研究員阿部重夫博士，同システム第２部長谷藤真也氏，研究開発推進本部大前力博士，水戸工場長瀬博博士，愛知工大植田明照博士，九州産業大学木脇久勝博士に深く感謝する。さらに，本研究の過程において，日立研究所武長寛主任研究員，北村忠明研究員，井上健士氏，電力・電機開発本部杉田洋一氏には有益な御討論をいただいた。ここに厚く御礼申し上げる。

また，本研究の一部はカリフォルニア大学ロサンゼルス校（ＵＣＬＡ），Computer Science Department における研究の一環として行ったものである。有意義な御指導と御討論を頂いたＵＣＬＡ教授 W. J. Karplus 博士，リサーチアシスタントE. R. Tisdale 博士，Han-Sen Dai 博士，香港科学技術大学Michael Stiber 博士に厚く御礼申し上げる。

さらに，本研究の一部は（株）日立製作所大みか工場の製品開発に関連して行った。研究の機会を与えていただくとともに，有益な御助言を賜わった大みか工場副技師長林利弘氏，同電機制御システム設計部部長今井浩一氏，同情報システム部森清三主任技師，ほか多数の方々に厚く御礼申し上げる。