

ニューラルネットワークによる
ロボットマニピュレータの
位置と力制御に関する研究

鴫田 正俊

ニューラルネットワークによる
ロボットマニピュレータの
位置と力制御に関する研究

平成7年1月

鴫田 正俊

目次

第1章 序論	1
1.1 研究の背景	1
1.2 研究の状況	4
1.3 本研究の目的と論文の概要	7
第2章 ニューラルネットワークと遺伝子アルゴリズムの概要	11
2.1 ニューラルネットワークモデル	11
2.1.1 ニューロンモデル	11
2.1.2 ネットワークの構成	13
2.1.3 ネットワークの学習アルゴリズム	15
2.2 ニューラルコントローラ	18
2.2.1 ニューラルコントローラの構成	18
2.2.2 ニューラルコントローラの学習	21
2.3 遺伝子アルゴリズム	23
2.3.1 遺伝子アルゴリズムの枠組み	23
2.3.2 遺伝的操作	25
2.4 まとめ	26
第3章 ニューラルネットワークによる対象物の特性を考慮した 適応力制御	27
3.1 制御対象と力制御系	27
3.2 ニューラルネットワークによる力制御系の構成	28
3.3 シミュレーション方法と結果	31
3.3.1 制御対象の数学モデル	31
3.3.2 固定ゲイン方式との比較	34
3.3.3 ニューラルネットワークの構成についての検討	34

3. 4	実験方法と結果	4 3
3. 4. 1	実験装置と実験方法	4 3
3. 4. 2	実験結果	4 3
3. 5	まとめ	4 4

第4章 ニューラルネットワークによる対象物の特性を考慮した

	位置と力の適応ハイブリッド制御	4 9
4. 1	マニピュレータの位置と力のハイブリッド制御	4 9
4. 2	ニューラルネットワークによるハイブリッド制御系の構成	5 1
4. 3	シミュレーション方法と結果	5 3
4. 3. 1	制御対象の数学モデルと制御システム	5 3
4. 3. 2	押し付け動作のシミュレーション結果	5 4
4. 3. 3	倣い制御のシミュレーション結果	5 5
4. 4	実験方法と結果	6 4
4. 4. 1	実験装置と実験方法	6 4
4. 4. 2	押し付け動作の実験結果	6 4
4. 4. 3	倣い制御の実験結果	6 5
4. 5	まとめ	6 6

第5章 ニューラルネットワークによるスタビング制御

5. 1	スタビング制御	7 5
5. 2	ニューラルネットワークを用いたスタビングコントローラ	7 7
5. 3	シミュレーション	7 7
5. 3. 1	制御対象の数学モデル	7 7
5. 3. 2	シミュレーション結果	7 9
5. 4	実験方法と結果	8 3
5. 4. 1	実験装置と実験方法	8 3
5. 4. 2	実験結果	8 3
5. 5	まとめ	8 4

第6章 追加学習型ニューラルネットワークによる適応力制御	88
6.1 シグモイド関数型ニューラルネットワークの問題点	88
6.2 追加学習システムの構成	89
6.2.1 制御対象とシステムの構成	89
6.2.2 追加学習型ニューラルネットワークの構造と学習則	92
6.2.3 追加学習アルゴリズム	95
6.2.4 ニューラルサーボコントローラ	96
6.3 シミュレーション	100
6.3.1 制御対象の数学モデル	100
6.3.2 NNSとNDPの追加学習の比較	100
6.3.3 力制御のシミュレーション結果	101
6.4 まとめ	102
第7章 強化学習によるマニピュレータの最適接近速度の学習	109
7.1 組立作業ロボットへの強化学習の適用	109
7.2 強化学習	110
7.2.1 強化学習の枠組み	110
7.2.2 Q-Learning	111
7.3 制御対象と制御システムの構成	113
7.3.1 制御対象	113
7.3.2 制御システムの構成	113
7.4 シミュレーション方法	117
7.4.1 制御対象の数学モデル	117
7.4.2 シミュレーション方法	118
7.5 シミュレーション結果	119
7.5.1 対象物の動特性が未知の場合	119
7.5.2 対象物の特性が未知で据え付け位置誤差がある場合	119
7.5.3 学習の収束性	120
7.6 まとめ	120

第8章 結論	129
8.1 本研究の成果	129
8.2 今後の課題	131
参考文献	133
謝辞	139

第1章 序論

1.1 研究の背景

近年、産業用ロボットマニピュレータの発展はめざましく、溶接、塗装、組立など、さまざまな作業に応用されている。また一方では、さらにより複雑で高度な作業へのロボットマニピュレータの適用が望まれている。マニピュレータを用いて、組立作業や、ばり取り、研磨作業など、複雑で高度な作業を行うには、マニピュレータの位置の制御だけでなく、エンドエフェクタと作業対象物との相互力をも制御する手法が有効であると考えられている。たとえば、マニピュレータを使った組立作業においては、部品の寸法誤差、位置決め誤差、マニピュレータの制御誤差など、さまざまな不確定要素が存在し、位置制御のみのマニピュレータでは、その調整に多大な時間と労力を必要としてしまう。エンドエフェクタと作業対象物との接触力の情報を用いることにより、これら不確定要素は、ある程度調整可能となるが、さらに積極的に力を制御し位置制御とうまく組み合わせることにより、こうした、さまざまな不確定要素が存在する組立作業に有効となる。

しかし、対象物との相互力を扱う制御系では、その閉ループ内に対象物の動特性が含まれるために、さまざまな対象物を取り扱う場合、あるいは対称物の動特性が未知の場合には、固定された制御システムでは対応できる範囲が限定されてしまう。したがって、対象物の特性変化に対して適応的に制御を行うシステムが必要となる。

また、マイクロマニピュレータで生物の細胞内の組織を破壊せずに細胞膜を破るといったケースや、マニピュレータを使ったドリルによる穴あけといったケースでは対象物の破壊前後において、動特性が大きく異なる。破断から反動抑制への一連の動作をスムーズに行う必要がある、このような非線形特性を有する制御対象には、従来からの制御システムでは適応が困難である。位置情報と力情報を組み合わせ統合的に処理するようなハイブリッド制御システムが必要になる。

このように、さまざまな環境下において複雑な作業を行うとき、環境特性が未知であったり、非線形性、不確定性が存在する。したがって、環境の特性変化に対し

て柔軟に適應できる制御システムが必要とされている。

また、近年は製品の多様化に伴い多品種少量生産が多くなっている。この生産形態に柔軟に対応できる自動組立システムの開発が求められている。フレキシブルな自動組立システムにおいてはロボットの必要性はますます高くなり、求められる機能もますます高度なものとなる。すなわち

- (1) 高速、高精度
- (2) 協調性
- (3) フレキシビリティ、巧妙性
- (4) 自律性、知能

といった機能の拡張が今後の組立作業ロボットに求められる。

将来の生産工場は、図1.1に示すような計算機により管理、制御される統合システムCIM(Computer Integrated Manufacturing)、IMS(Intelligent Manufacturing System)になる。その生産行程において、汎用性の高いロボットの使用は不可欠となるであろう。しかし、人間が動作を細かく教示するような手法では生産能率があがらず統合システムの意味がなくなる。作業に応じてロボット自らが学習して作業の最適化をはかっていく必要がある。

現在広く使用されている産業用ロボットは、動作教示方式を基礎としており、実際の作業環境においてマニピュレータの位置や軌道を人間が細かに教示し、それらをプレイバックすることで作業の実現をはかっている。したがって、作業や環境の変化に対しての教示と調整には、多くの手間と時間を必要とする。このため人間による操作を必要としない、知能を持った自律型のインテリジェントシステムが必要とされている。また、研磨作業や組立作業など複雑な生産工程では熟練作業者のような巧みな動作を必要とするが、それらをロボットに教示するのは容易ではない。熟練作業者は経験を積み重ねることにより位置情報と力感覚を統合し、力と位置を巧みに制御し作業を遂行する。ロボットが熟練作業者のように作業の試行を繰り返し経験を積み重ねるうちに自律的にスキル、行動を獲得していく能力を持たせることができれば実に有効である。

ロボットの適應範囲は、工場内での産業用ロボットに限らず農業、林業、建築、医療、家庭などのあらゆる分野へ広がりは始めている。これらの分野へのロボット

の普及と同時に強く要求されるのが環境適応性である。現在は、環境に応じた専用のロボットを作るか、または、ロボットに合わせて環境を標準化している。しかし、専用ロボットでは高コストになるし、環境の標準化も実際には難しい。家事作業を扱うホームロボットは特に環境適応能力が要求され、標準化が難しい分野であろう。たとえば家の中を部屋から部屋へ移動する場合、ドアがあったり襖のような引戸があったり、それらをスムーズに開閉するためには、その種類（環境）に応じて動作を変え、力感覚を利用して制御しなければならない。あるいは、料理をするロボットは、いろいろな道具を使い、硬い物、柔らかい物、壊れやすい物を切ったり、把持したり、対象に応じて位置と力を巧みに制御する必要がある。環境に応じた動作の選択能力と、位置と力を組合わせ、統合的に制御する能力が必要である。

ロボットが人間と同じ手法で作業を行う必要はないが、ロボットの知能化、自律化を検討する場合、人間の作業を参考にして人間に近い機能を持たせることも重要である。今後、ロボットがより人間に近づき人間に替わって作業を進めることができれば、工場内でより幅広い生産活動に応用されるのはもちろんのこと、宇宙、海中、さらには医療、福祉、家事といった人間社会の中で人間と混在して活動することが可能となる。ロボットの適用範囲がさらに広範囲に拡大される。そのためには、制御性能の向上とともに、力制御を巧妙に実現できる知能ロボットの研究、開発がぜひとも必要と言える。

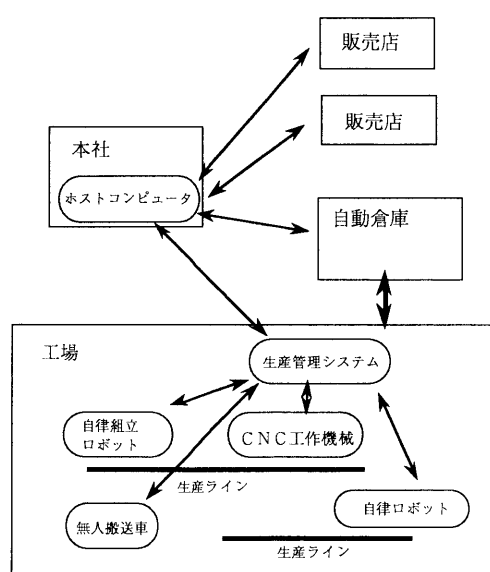


図 1.1 CIM, IMSの構成

1. 2 研究の状況

従来の研究では、組立作業における、部品の寸法誤差、位置決め誤差などの不確定要素に対処する手法として、Hogan[53]によってインピーダンス制御、Raibertら[18]によって位置と力のハイブリッド制御手法が提案されている。ロボットのエンドエフェクタと環境との相互作用力は、慣性力項、減衰力項、剛性力項とからなる機械インピーダンスを用いて表現することができる。インピーダンス制御は、力とマニピュレータの機械インピーダンスの関係を制御する間接的な力制御手法である。また、作業環境によりエンドエフェクタが拘束されている方向には力を制御し、それ以外の方向には、位置と姿勢を制御する方式がハイブリッド制御手法である。しかし、これらの研究では実際の作業において制御系のパラメータをどのように設定するかという点について触れられていないし、制御系の設計においてマニピュレータおよび対象物の動特性は考慮されていない。対象物の動特性も考慮した力制御については、これまでいくつかの研究がある。Anら[43]はマニピュレータおよび対象物を線形なばね-ダンパー-質量でモデル化し制御系の安定性を検討している。系が不安定になる支配的要因として対象物の剛性に対するフィードバックゲインの大きさの関係を挙げており、安定化のためには速度フィードバックゲインの調整が有効であると述べている。また、吉川ら[44]は対象物の動特性を考慮した状態フィードバック制御手法を提案している。対象物の状態量をオブザーバで推定して高精度な制御を実現している。しかし、それらの研究ではいずれも対象物の特性およびマニピュレータの特性が既知である必要があり、その応用性が限られている。

未知な環境や環境特性の変動に対しては、モデル規範型適応制御 (MRAC) [46]やセルフチューニング型適応制御 (STR) [47]といった適応制御手法が提案されている。福田ら[16]は、モデル規範型適応制御を用いた対象物の特性を考慮した力制御系を提案している。しかし、大きな環境変動や非線形な特性を持つ場合には、その適応範囲に限界があるし、複雑な制御では、設計が困難となる[48]。

柏木[55]によりマニピュレータによる研磨、ばり取り作業などの実用化の研究が

進められている。従来の位置制御系に、力誤差フィードバックに基づく修正位置量を与える方式をとっている。現在、多く利用されている位置制御を基本としているマニピュレータの制御プログラムの大きな変更を必要としない点で実用的と言えるが、力と位置の変換ゲインの調整には難しさが伴うようである。

ロボットの知能化、自律化の試みとしては、人間の論理的思考を手本とした人工知能（AI）の手法が研究されてきた [45]。記号処理技術に基づく人工知能により、知識や推論といった人間の高度な情報処理の実現が試みられてきたが、実際に応用する段階で多くの手間を必要とするわりには、それほど賢くない。組立作業や接触動作の作業計画にAI手法を利用した研究も様々なアプローチが試みられてきた [56][57][62]。作業環境内の物体の幾何学的形状や配置を環境モデルとして与え、そのモデルに基づき作業目標を達成する動作の計画を実施する。このアプローチの問題点は、すべての知的機能が集中管理されたモデルに基づいていることから、実際の複雑な環境では、モデルの生成、管理、更新は容易ではなく膨大な時間がかかる。環境特性が変化する場合の適用は困難となる。

AI手法と同じように人間の脳のマクロ的な機能に注目したアプローチとしてファジィ制御 [58] の手法がある。ファジィ理論が従来記号处理的な表現と数値处理的な表現の両面を持ち、論理の世界と現実の世界を結びつける手段として注目され応用研究がさかんである。従来からの制御手法では困難であった系において、オペレータが行っていた制御動作を、その経験則を記述することによりシステム上に実現する手法である。廣田 [59] はロボットマニピュレータの軌道制御やプランニングへ応用し、有効性を示している。不確定要素を含む環境に対する適応能力を持ち、非線形な系に対しても有効であるが、環境の大きな変化に対しては、ルールの変更が必要となる。環境変動を認識する上位レベルの戦略手法が必要である。

一方、人工知能、ファジィ制御のように人間の脳のマクロ的な機能に注目したアプローチでなくミクロ的なメカニズムにヒントを得たニューラルネットワークの研究が近年特に注目されている。理論体系がかなり整備されてきており、パターン認識や制御への応用が数多く報告されている。

Narendraら [24] や、山下ら [63] は、ニューラルネットワークを使用した適応制御を提案しており、ニューラルネットワークの非線形性が制御対象の非線形性を相殺

するように制御系を構成できること、学習により特性が曖昧な制御対象に対しても対応できることを示している。

Albus [49]、川人ら [22] の研究をはじめとして、ニューラルネットワークをロボットマニピュレータの軌道制御に応用し、ニューラルネットワークの学習機能、非線形写像能力を用いた柔軟な制御系を構築する研究も盛んに行われている。Ahmadら[50]はマニピュレータの逆運動学をニューラルネットワークの学習により得る方法を提案している。橋本ら[51]は視覚情報によるマニピュレータの位置・姿勢制御にニューラルネットワークを用いる手法を提案している。

ロボットマニピュレータの力制御系にニューラルネットワークを使用した研究としては、藪田ら[21]が力制御を対象とし、ニューラルネットワークによる逆ダイナミクスの実現の可能性を示している。ニューラルネットワークの非線形写像能力を利用し、対象プラントの逆ダイナミクスを同定する手法で、ニューラルネットワークが完全に逆ダイナミクスの同定に成功した理想状態では、あらゆる目標軌道に対応した制御入力を得ることができる。しかし、有限個のユニットからなるニューラルネットワークの構造やデルタルールに基づいた学習方法では完全な逆ダイナミクスの同定は不可能であり、学習パラメータによっては系の安定化が困難であることも示している。五味ら[54]は人間の腕の筋骨格系の運動と小脳の運動学習に注目し、生体運動制御系における内部モデルの役割、獲得メカニズムのモデル化の研究を進めている。筋骨格系をインピーダンス制御系でモデル化し、運動中のスティッフネスの計測と脊椎反射系の挙動、内部モデルとの関係を考察している。この研究は、直接すぐにロボットマニピュレータの制御に応用できるものではないが、川人[22]らの研究と同様に、神経生理学の研究から制御アルゴリズムへの展開のアプローチして興味深い。Yangら[38]は組立作業の学習に強化学習を用いる方法を提案している。教師を必要としない学習方式である強化学習を用い、ダンピング制御のパラメータを学習し組立作業を実現する手法である。現時点ではごく簡単な作業にしか応用できず、環境の特性変化に対しては適応不可能であるが、強化学習の可能性を示唆しており今後の発展が期待される。また、浅田ら[64]、Linら[52]はばり取り作業において、熟練作業者の持つスキルをニューラルネットワークに学習させ、獲得された作業スキルに基づきロボットを制御する手法を提案している。この手法

では、人間とロボットとは機構的に異なり熟練者の動作がそのままロボットにとって最適である保証がないが、人間の持つスキルを獲得する1つの試みとして興味深い。

AI、ファジイ、ニューラルネットワークといった知能化のための要素技術を個々に使用するのではなく、それらを融合した、ファジイニューラルネットワーク[60]や階層的に組合わせた階層的知的制御[61]の研究が進められ、力制御への応用としては下島ら[60]の研究があり、今後の発展が期待される。

以上、ロボットマニピュレータの力制御、適応制御、AI、ファジイ、ニューラルネットワークを用いた制御の従来からの代表的な研究について述べた。しかし、多くの研究が環境特性が既知であることを前提としているか、または、未知環境に適応可能であっても環境の大きな特性変化に対する適応性についての検討がなされていない。

1. 3 本研究の目的と論文の概要

本研究では、以上のような背景を踏まえ、ロボットマニピュレータの位置と力の制御に、ニューラルネットワークを用いて、環境の特性変化に対して適応的にコントローラを自己生成し、適応範囲を大きく広げることが可能な制御システムの構築を目的とする。

図1. 2に本論文の構成を示す。まず第2章では、本研究で使用しているニューラルネットワークモデルの構成および学習について、その概要を述べる。さらに、ニューラルネットワークをコントローラとして用いるための構成方法と学習方法についても述べる。また、本研究では第6章においてニューラルネットワークの構成を遺伝子アルゴリズムを用いて探索する手法をとっているため、遺伝子アルゴリズムの概要についても述べる。

第3章では、ニューラルネットワークを用いた、未知対象物に対する力制御方式について述べる。対象物との相互作用を伴う制御システムでは、閉ループ内に対象物の動特性が含まれるため、対象物の動特性が既知でないと制御系の設計が可能な

い。また、対象物だけでなく歯車のバックラッシュ、摩擦といったマニピュレータ機構の非線形性、パラメータ推定の困難さがあり、制御系の設計が容易でない。更に、対象物の特性が大きく変化する場合、固定フィードバックゲインによる制御では適応できない。そこで、ニューラルネットワークの学習機能、非線形マッピングを用いることにより、従来の制御方式と比較して厳密な数学モデルを必要とせず、未知対象物に適応でき、制御対象の大幅な特性変化に対しても適応できることを示す。

第4章では、第3章で述べた適応力制御システムを多自由度マニピュレータに拡張する方法について述べる。マニピュレータによるグライディングや刷毛塗りといった作業では、位置と力のハイブリッド制御が有効である。環境によってエンドエフェクタが拘束されている方向には力を制御し、それ以外の方向には位置および姿勢を制御する制御方式である。この場合も、対象物の動特性が、制御の閉ループ内に入るため、対象物の動特性が大きく異なる場合や未知対象物に対して、安定な制御が困難となる。また、多自由度マニピュレータは非線形システムであり、従来のハイブリッド制御システムでは、姿勢変化による不安定性が問題とされている。そこでニューラルネットワークを使用した、対象物の動特性の変化やマニピュレータの姿勢の変化に対してロバストな制御システムの構築について述べる。

第5章では、非線形性の強いスタビング制御にニューラルネットワークが有効であることを述べる。マイクロマニピュレータで生物の細胞内の組織を破壊せずに細胞膜を破るといった制御や、マニピュレータを使ってドリルによる穴あけを行うといったスタビング制御では対象物の破壊前後において、動特性が大きく異なる。破断から反動抑制への一連の動作をスムーズに行う必要がある、このような非線形特性を有する制御対象には、従来からの制御システムでは適応が困難である。位置情報と力情報を組み合わせ統合的に処理するようなハイブリッド制御システムが必要になる。ニューラルネットワークの学習機能、非線形マッピングを用いて、非線形特性を有する対象物の制御を順応的に行うことが可能であることを示す。

第6章では、新しい知識を追加したり間違っただけの記憶した知識を修正する追加学習を考慮した、制御システムについて述べる。一般的に、ニューラルサーボコント

ローラには、シグモイド関数を用いた多層型ニューラルネットワークを使用しており、学習にはバックプロパゲイショナルゴリズムを用いているため、新しい知識を追加したり間違っただけで記憶した知識を修正する追加学習ができない。特に、環境（対象物）まで含めた系として扱わなければならない力制御やハイブリッド制御では、動特性の大きく異なる未学習の対象物が新たに制御対象として加わった場合、今まで学習した全ての対象物を含めて再度学習し直す必要があるため非常に効率が悪い。新たなパターンを追加学習するためには、今まで学習したパターンをすべて用意しなければならないし、学習も最初からやり直さなければならず、追加のための学習に多くの手間と時間を必要とする。そこで一種のラジアル・ベース関数を使用したニューラルネットワークと一般的なニューラルサーボコントローラを階層的に組み合わせることにより、追加学習が可能となり、環境の特性変化に対して適応的にコントローラを自己生成し、適応範囲をより大きく広げることが可能な制御システムの構築が可能であることを述べる。

第7章では、マニピュレータの接触動作の最適接近速度の獲得に強化学習（Q-Learning）を用いる方法について述べる。明確な教師がない環境での有効な学習アルゴリズムとして、強化学習がある。強化学習とは、環境から「報酬」あるいは「罰」という形で評価を得ることにより、より良い行動を探索、発見していく学習方法である。組立作業をロボット・マニピュレータで行おうとすると部品の寸法誤差、位置決め誤差、マニピュレータの剛性、制御誤差、さらには環境の未知特性など、さまざまな不確定要素が存在するため、どのような行動をマニピュレータに与えれば、組立を確実に達成できるか明確でない。このようなプロセスに強化学習が有効と考えられる。作業の試行を繰り返すうちに、スキルを獲得し、確実に作業を遂行できるようになる。強化学習による組立作業のスキル獲得をめざし、第7章では、組立作業の基礎となる力制御、特に接触動作をとりあげ、強化学習の可能性について述べる。環境（対象物）の特性が未知で、位置決め誤差が存在する接触動作において、強化学習による最適接近速度の獲得の可能性を検討した。

第8章では、本論文で提案したニューラルネットワークによる位置と力のハイブリッド制御方式についてまとめ、今後の展望について述べる。

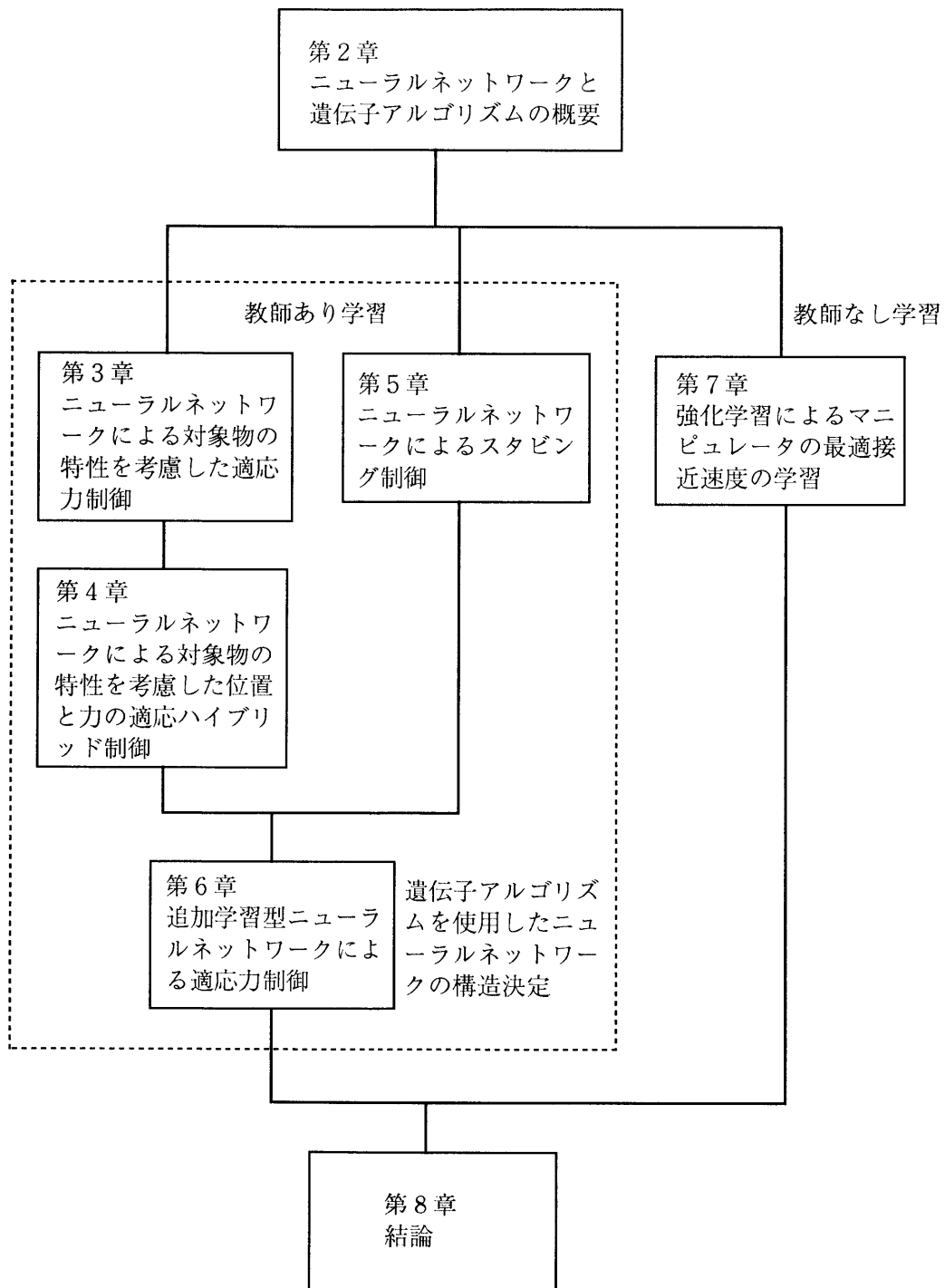


図 1.2 本論文の構成

第2章 ニューラルネットワークおよび 遺伝子アルゴリズムの概要

本章では、本研究で使用している、ニューラルネットワークモデルおよび遺伝子アルゴリズムについて、その概要を述べる。

2.1 ニューラルネットワークモデル

2.1.1 ニューロンモデル

ネットワークを構成する各ニューロンのモデルは、生物のニューロン（神経細胞）を模倣し図2.1のように示される。ニューロンは多入力出力であり、その内部状態は入力信号の荷重和として与えられるれ、さらにニューロンからの出力 $y(t)$ は次式のようになる。

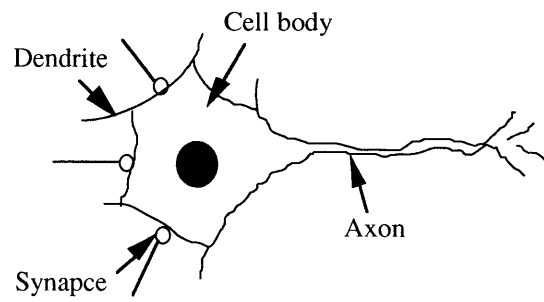
$$y(t) = f\left(\sum_{i=1}^n w_i x_i(t) - \theta\right) \quad (2.1)$$

ニューロンの出力関数 $f(x)$ は、図2.2に示すように、しきい値を用いて1、0の2値関数である場合や連続で非線型であるシグモイド関数を用いる場合が多い。シグモイド関数の1例を次式に示す。

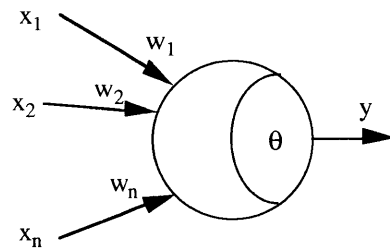
$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.2)$$

$$f(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \quad (2.3)$$

このモデルは、McCulloch-Pitts[1] のモデルを基本としている。実際の神経系ではニューロン間の情報伝達は周波数変調パルスによるが、このモデルでは、そのパルス頻度をアナログ量として捉えてこれを x 、 y で表現する。結合の強さを表す重み係数 w_i はシナプス荷重と呼ばれ、興奮性のときには正の値、抑制性のときには負の値をとる。

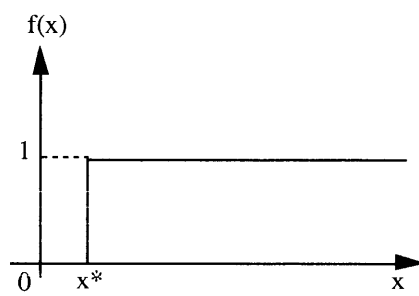


生物のニューロン

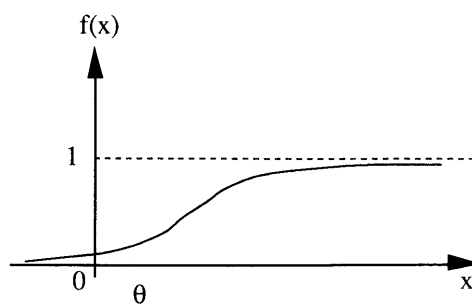


ニューロンモデル

図 2.1 ニューロンモデル



(a) 2 値関数



(b) シグモイド関数

図 2.2 ニューロンモデルの非線形出力関数

2.1.2 ネットワークの構成

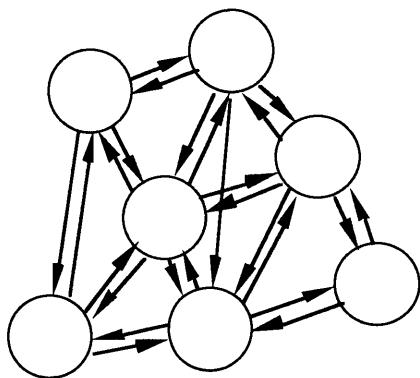
生物の脳においては、膨大な数のニューロンが結合してネットワークを構成し、高度な知能活動が行われている。工学的なニューラルネットワークは、図2.1に示したニューロンモデルによりネットワークを構成する。構成の仕方は、ニューロンが相互に結合している相互結合型と、相互の結合がなく、階層的な構造で信号の流れが一方方向である多層型及び自己回帰する結合があるリカレント型の三つの種類に大別される。図2.3にそれらの構成を示す。

相互結合型ニューラルネットワークは、ある初期状態から出発したネットワークは、ニューロンの状態変化が繰り返されるうちに、ある安定な平衡状態に到達する。いくつかの状態を巡るサイクルに落ち込む。代表的なモデルにHopfieldモデル[2], Kohonen [3], アソシアトロン[4]がある。連想記憶とか最適化問題に利用される。このモデルは、平衡状態に達するまで時間がかかり、リアルタイムな制御には向いていない。

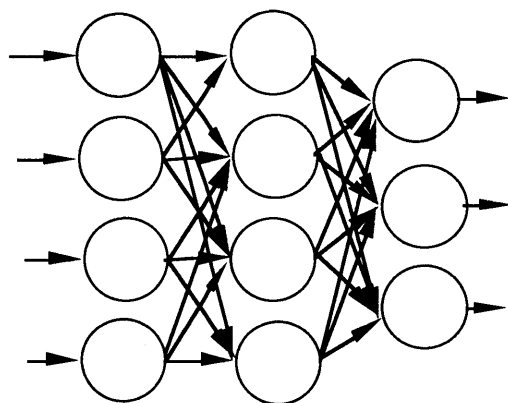
多層型ニューラルネットワークは、相互結合型と異なり、ネットワークの中を情報が入力層から出力層に一方方向に一度流れて状態変化が終わる。代表的なモデルにRosenblattのパーセプトロン[5]、Rumelhart[6]らのバックプロパゲーションモデルがある。パターン認識やコントローラとして用いられ、非線型写像能力がある。

リカレント型ニューラルネットワーク[7]は、多層型ニューラルネットワークが時間的に静的な写像しか表現できないのに対して、ニューロンの出力を自己回帰することにより動的な表現が可能である。コントローラとして用いられているが、学習および初期状態の与え方が難しいといった問題がある。

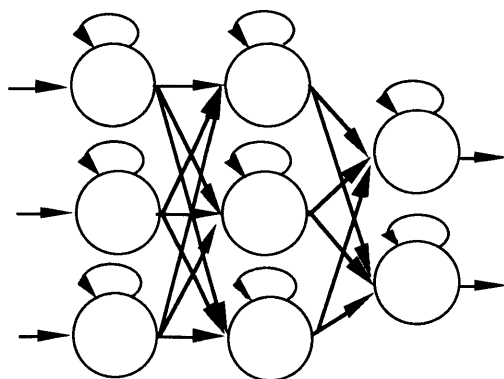
本研究においては、非線型写像能力があり、学習アルゴリズムの比較的簡単な多層型のニューラルネットワークを用いた。



(a) 相互結合型ニューラルネットワーク



(b) 多層型ニューラルネットワーク



(c) リカレント型ニューラルネットワーク

図 2.3 ニューラルネットワークの構成

2.1.3 ネットワークの学習アルゴリズム

ニューラルネットワークの写像関数は、ニューロン間の結合荷重によってきまる。その結合荷重は、ネットワークへの入出力間の写像データを教師信号として学習により決定される。

ニューラルネットワークの学習アルゴリズムには、教師ありと教師なしの2種類があるが、ここでは、本研究に用いた多層型ニューラルネットワークの教師ありの学習アルゴリズム、誤差逆伝播 (Error Back Propagation : BP) 法[6]について述べる。誤差逆伝播法は、最急降下法を基礎としており以下にその手順を述べる。

第 $m-1$ 層第 j ニューロンの出力を y_j 、第 m 層第 i ニューロンの出力を x_i とすると、 w_{ij} を結合荷重として、それらの関係は次式で与えられる。

$$x_i(t) = f(z_i) \quad (2.4)$$

ここで、 z_i は第 i ニューロンの内部状態を表す。

$$z_i(t) = \sum_{j=1}^n w_{ij} y_j(t) \quad (2.5)$$

一方、出力層での教師信号を d_i として、出力自乗誤差を、

$$E = \sum_i \frac{1}{2} (d_i - x_i)^2 \quad (2.6)$$

とすると、結合荷重の修正量は、最急降下法を基に、次のように求める。

$$\Delta w_{ij} \propto -\frac{\partial E}{\partial w_{ij}} \quad (2.7)$$

式(2.7)の右辺を展開すると、

$$-\frac{\partial E}{\partial w_{ij}} = -\frac{\partial E}{\partial z_i} \frac{\partial z_i}{\partial w_{ij}} = \delta_i y_j \quad (2.8)$$

従って、荷重の修正量は、次式のようになる。

$$\Delta w_{ij} = \eta \delta_i y_j \quad (2.9)$$

ただし、 η は定数で学習係数である。

δ_i は第 i ニューロンの学習信号で、出力層では

$$\delta_i = -\frac{\partial E}{\partial z_i} = -\frac{\partial E}{\partial x_i} \frac{\partial x_i}{\partial z_i} = (d_i - x_i) f'(z_i) \quad (2.10)$$

中間層では、第 $m+1$ 層の各ニューロンの学習信号 δ_k を用いて

$$\begin{aligned} \delta_i &= -\sum_k \frac{\partial E}{\partial x_k} \frac{\partial x_k}{\partial x_i} \cdot f'(z_i) \\ &= -\sum_k \frac{\partial E}{\partial x_k} w_{ki} \cdot f'(z_i) \\ &= \sum_k \delta_k w_{ki} \cdot f'(z_i) \end{aligned} \quad (2.11)$$

となる。なお、 f' は f の微分値を示す。

出力層での教師信号と実際の出力値との誤差を入力として、出力層から入力層の方向へ、信号の伝播とちょうど逆の方向に結合荷重で重み付けた和をとりながら伝播させていく。

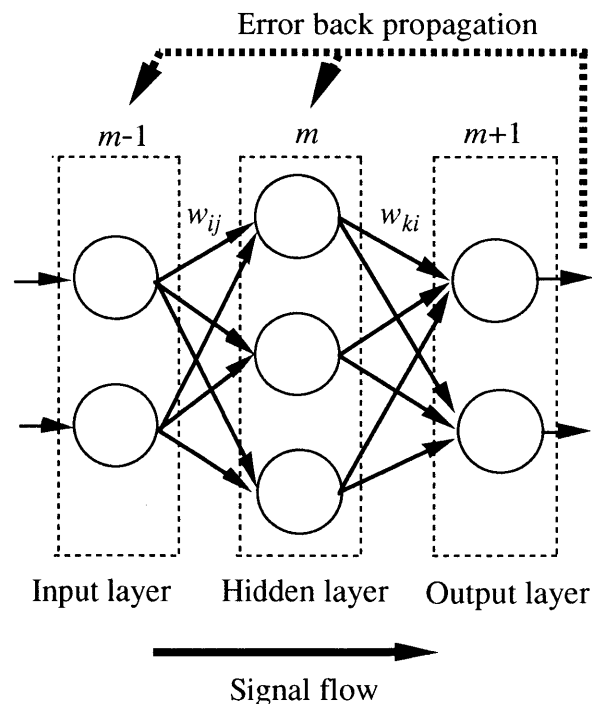


図 2.4 誤差伝播学習アルゴリズム

学習方程式は一般的に、収束度の改善のために (2.9) 式に慣性項をつけ加えた次式が用いられる。

$$\Delta w_{ij}(n+1) = \eta \delta_i y_j + \alpha \Delta w_{ij}(n) \quad (2.12)$$

ただし、 n は学習のタイムステップ、 α は安定化定数である。

学習プロセスは、いろいろな入力信号と対応する出力教師信号のセットに対し繰り返し行う。誤差逆伝播法の学習アルゴリズムは次のようになる。

1. 結合荷重の初期値を設定する。
2. 入力層に入力信号を与え、入力層から出力層へ向かって各ニューロンの状態変化を順次計算し、ネットワークの出力を求め。
3. 学習方程式に従って、出力層から入力層に向かって結合荷重を順次修正していく。
4. すべての教示パターン（入力信号と対応する出力教師信号のセット）に対し、2から3までのステップを繰り返す。
5. 収束判定条件を満たすまで2から4までのステップを繰り返す。

2.2 ニューラルコントローラ

2.2.1 ニューラルコントローラの構成

ニューラルネットワークをコントローラとして用いる場合、図 2.5 に示すように大きく分けて次の 3 種類の構成が考えられる。

(1)直接型コントローラ[8][9]

ニューラルネットワークを直接コントローラに用いる方法で、ニューラルネットワークの非線形性で制御対象の非線形性を相殺するように、制御系を構成することが可能である。

(2)補償型コントローラ[10]

従来の線形フィードバックコントローラの補償器としてニューラルネットワークを使用し、制御対象の未知の特性や非線形性の補償を行うよう制御系を構成する。

(3)間接型コントローラ[11]

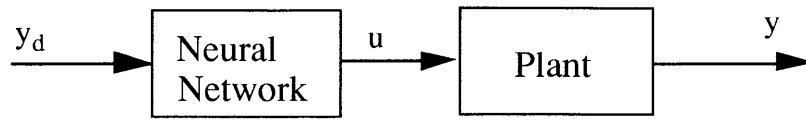
ニューラルネットワークは制御対象の同定、パラメータ推定に用い、その結果を基に従来型のコントローラをチューニングするように制御系を構成する。

制御にニューラルネットワークを用いる利点は、

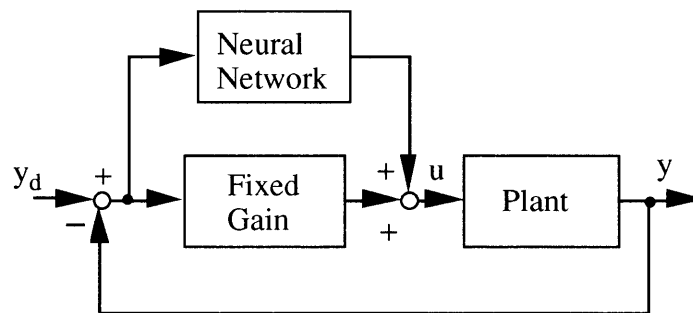
- 1、非線形な制御対象に対して、ニューラルネットワークの非線形性を利用することにより対応が可能である。
- 2、未知特性の制御対象に対して、ニューラルネットワークの学習機能を利用することによりモデリングが可能である。
- 3、汎化能力を持つので、モデル化していない未知パラメータに対してもロバストである。
- 4、情報を分散表現するので、外乱に対して強い構造を持たせることが可能である。
- 5、並列処理により、高速な演算が行える。

ことがあげられる。しかし、ニューラルネットワークの学習のために、制御対象の特性が既知でないと、単純にバックプロパゲーション法を適用できないので、なんらかの工夫が必要である。詳しくは、次節で述べる。

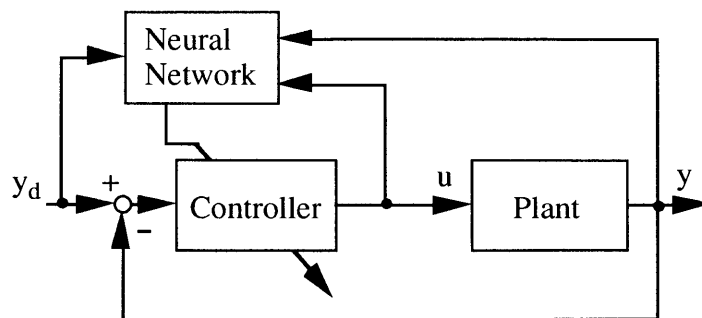
本研究においては、第3章と第4章で、間接型ニューラルコントローラを用いている。ニューラルネットワークは制御対象のパラメータ推定機構及びフィードバックコントローラのゲイン調整機構として働く、一種のセルフチューニング型適応制御系である。従来からの制御系の設計の知識、経験的知識を利用でき、オフラインで学習が可能な利点がある。第5章では、非線形性の強い制御対象に対する制御を実現するために補償型ニューラルコントローラを用いている。第6章では、一種の自律型コントローラの実現を目指しており、ニューラルネットワークの学習機能を利用し、未知特性の制御対象に対してコントローラを自己生成することができる直接型ニューラルコントローラを用いている。



(a) 直接型コントローラ



(b) 補償型コントローラ



(c) 間接型コントローラ

図 2.5 ニューラルコントローラの構成

2.2.1 ニューラルコントローラの学習

本節では、ニューラルネットワークをコントローラとして用いる場合の学習法とその問題点について述べる。

図2.6に示すように、直接型ニューラルコントローラを用いる場合、制御対象の出力 y が、目標値 y_d になるようにするのが制御目的となるから、評価関数 E を

$$E = \frac{1}{2}(y_d - y)^2 \quad (2.13)$$

とすると、ニューラルネットワークの学習は、次式にのようになる。

$$\Delta w \propto -\frac{\partial E}{\partial w} = \frac{\partial y}{\partial u} \frac{\partial u}{\partial w} \quad (2.14)$$

w : ニューラルネットワークの結合荷重、 u : プラントへの入力

ここで $\frac{\partial u}{\partial w}$ は、通常のバックプロパゲーションアルゴリズムで求まるが、 $\frac{\partial y}{\partial u}$ は制御対象の特性が未知の場合求めることが不可能となる。したがって、なんらかの工夫が必要となる。

一般に $\frac{\partial y}{\partial u}$ より小さな定数を $y_d - y$ の掛けた値によって学習をさせる場合が多い [12]。しかし、定数の選び方によっては、学習が収束しない場合がある。Jordan は、図2.7(a)に示すようにニューラルネットワークを使った順モデルを併用し、制御対象の特性をバックプロパゲーションで同定しながら、その時に得られる $\frac{\partial y}{\partial u}$ を $\frac{\partial y}{\partial u}$ の代わりに用いてニューラルコントローラの学習をする手法を提案している [12]。この手法は、順モデルがうまく収束しないと、ニューラルコントローラの学習が発散してしまう問題がある。川人らは、図2.7(b)に示すように、フィードバックの誤差を最小にするように学習を行うフィードバック誤差学習法を提案している [14]。この手法では、あらかじめフィードバックコントローラをうまく調整して目標値誤差がある程度小さくないと、やはり学習が収束しないといった問題がある。ニューラルコントローラの学習に関していろいろな手法が提案されているが、現時点では、対象とする制御系に応じて学習係数（定数）なりフィードバックゲインを調整する必要がある。

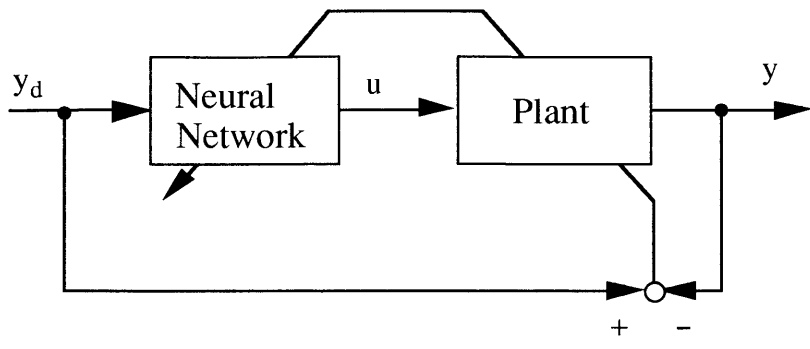


図 2.6 ニューラルネットワークの学習

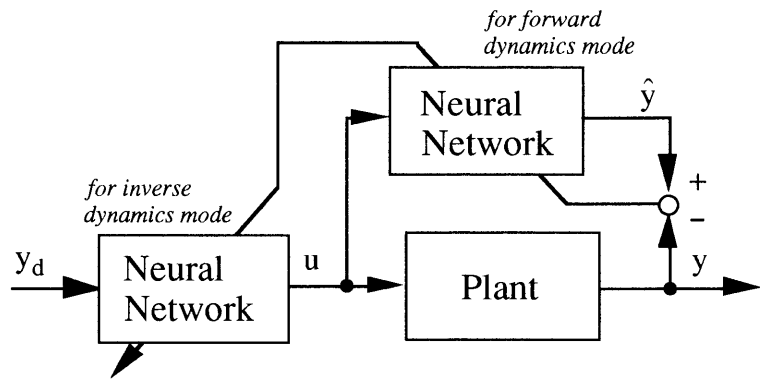


図 2.7 (a) Jordanの一般学習法

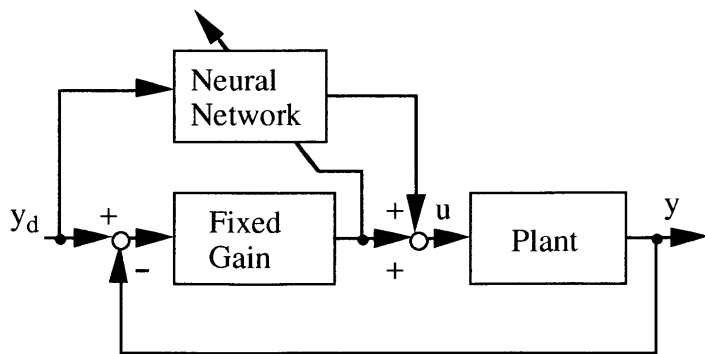


図 2.7 (b) フィードバック誤差学習法

2.3 遺伝子アルゴリズム

2.3.1 遺伝子アルゴリズムの枠組み

遺伝子アルゴリズム (Genetic Algorithm:GA) [15]は、生物の遺伝と進化を模倣するモデルとして考えられた計算アルゴリズムである。そのモデルは、以下のようになっている。

まず個体レベルでは、生物は、生命体の基本要素として遺伝子 (gene) を持っている。遺伝子を取りうる値を対立遺伝子 (alles) という。複数の遺伝子が集まって染色体 (chromosome) を形成する。染色体は普通、文字列 (string) として表現される。各個体は、これらの染色体により形作られる。次に、集団レベルの進化は、まず、ある規模の個体の集団 (population) が存在し、各個体は、生存競争に打ち勝ち、生殖等の過程を経て、新しい世代を生み出していく。新世代は、親の遺伝子の組み合わせにより、新しい遺伝子を持ち、また、生殖の過程で突然変異などの変化が加わり新しい能力を持つことになる。

なお、モデル化に関しては、表現型と遺伝子型という表現形式がある。表現型 (phenotype) とは、染色体によって規定される形質の外部的表現をいう。遺伝子型 (genotype) とは、形質の染色体による内部的表現のことをいう。表現型から遺伝子型へ写像することをコード化 (coding) といい、逆に遺伝子型から表現型へ写像することをデコード化 (decoding) という。

以上の事柄をアルゴリズム化すると次のようになる。

1) 遺伝子型の設定

解こうとする問題に応じてコード化をおこなう。

2) 集団の設定と初期化

ランダムな染色体をもつ個体を m 個生成する。

3) 増殖 (reproduction)

設定された確立で、個体の複製を作る。

4) 交差 (生殖等による遺伝子の組換え) (crossover)

設定された確立、交差回数、交差方法により各個体のペアを親として、染色体の組替えにより、新しい (子の) 個体を生成する。

5) 突然変異 (mutation)

設定された確立、突然変異方法により、各個体ごとに、染色体の選ばれた遺伝子座の値を他の対立遺伝子に置き換える。

6) 個体の環境への適応度の評価と選択 (selection) 淘汰

個体の適応度を考慮して、m個の個体を残し、他を淘汰させる。

図2.8に示すように、以上の3)から6)を有限回(世代数)繰り返すことにより集団を進化させ、評価の高い個体を作り出していく。一種の乱数探索法であるが、交差を行うことにより他点探索が可能となる。

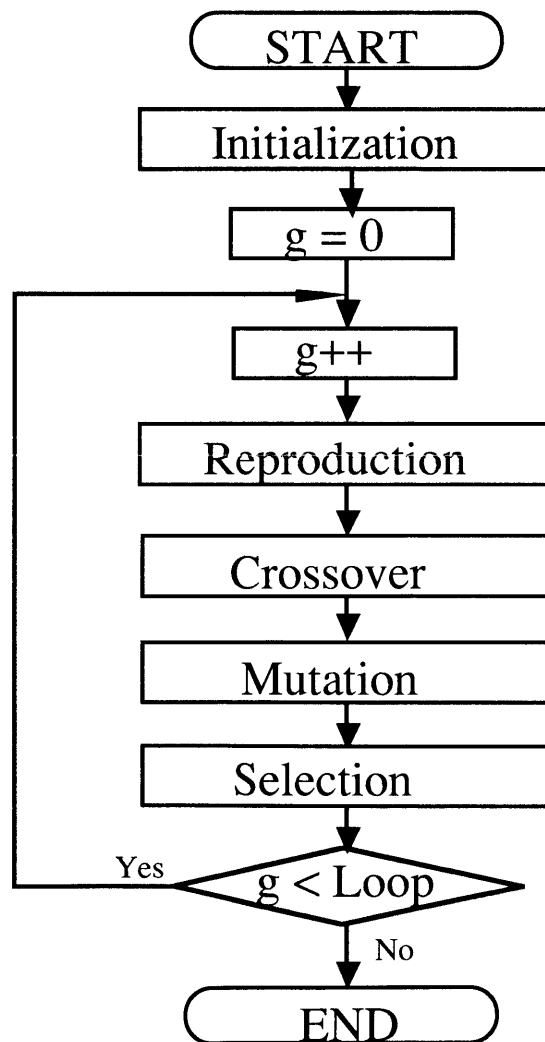


図2.8 遺伝子アルゴリズム

2.3.2 遺伝的操作

遺伝子アルゴリズムの主要な遺伝的操作について、代表的な戦略について述べる。

<選択>

代表的な選択戦略には、ルーレット選択、ランクに基づく選択、エリート保存選択がある。

ルーレット選択は、適応度に比例した確率が各個体に割り当てられており、この確率に従って、個体のペアが順次生成されていく。この方法は、集団内に突出した適応度を持つ個体が存在すると、その個体の複製が多数つくられてしまい初期収束を引き起こす可能性がある。

ランクに基づく選択では、あらかじめ設定された選択確率表に従って、集団内の適応度の順序に対応する選択確率を各個体に割り当て、選択を行う。この方法はルーレット選択の問題点を回避できるが、選択確率表をどのように設定すべきかが問題である。

エリート保存選択は、現世代において適応度の高い個体は、次世代に無条件に残す選択方法で、ルーレット選択、ランクに基づく選択の問題を回避でき、最適化問題に有効な方法である。

<交差>

交差の戦略には、1点交差、多点交差、一様交差がある。

1点交差は、染色体の切断を1箇所とし、2つの個体間で染色体の組替えを行い新しい個体を2つ生成する。最も簡単な手法であるが、交差を行うことにより大域的な探索が可能となる。

多点交差は、染色体の切断を多数行い、切断点ごとに2つの個体間で染色体の組替えを行い新しい個体を2つ生成する。1点交差と比較して何とおりの交差が可能であるが、あまり切断が多いと乱数探索となってしまう収束性が悪くなる。

一様交差は、あらかじめ染色体と同じ長さのマスクを用意しておき、マスクによりどちらの親の遺伝子を子に継承させるかを決定する。特に優先的に継承させたい遺伝子がある場合に有効な手段となるが、あらかじめ用意するマスクの設定が難しいといった問題がある。

<突然変異>

突然変異は、ある確率で選ばれた遺伝子座の遺伝子の値を他の対立遺伝子に置き換える操作である。他に逆位がある。逆位は、遺伝子座のある部分の順序を反転させる操作である。

突然変異の操作を行うことで、局所的な乱数探索を行うことができる。また、集団の多様性を維持し初期収束問題の回避を可能としている。交差で大域的な探索を行い突然変異で局所的な探索を行う。

本研究では、先に述べた問題点を考慮し遺伝的操作の戦略には、エリート保存選択、1点交差を採用した。

2.4 まとめ

本章では、本研究で使用しているニューラルネットワークモデルの構成および学習について、その概要を述べ、ニューラルネットワークをコントローラとして用いるための構成方法と学習方法についても述べた。また、ニューラルネットワークをコントローラとして用いる場合の利点およびニューラルコントローラの学習における問題点を明らかにした。

さらに、遺伝子アルゴリズムの概要についても述べ、本研究で使用している遺伝的操作の戦略と特徴について述べた。

第3章 ニューラルネットワークによる 対象物の特性を考慮した適応力制御

3.1 制御対象と力制御系

ロボット・マニピュレータにより、はめ合い動作やクランク回しなどの高度の作業を行うためには、マニピュレータの位置だけでなく、マニピュレータと対象物との相互力をも制御することが必要となる。しかし、対象物との相互作用を伴う制御システムでは、システム全体のフィードバックの閉ループ内に対象物の動特性が含まれる。

図3.1に、位置制御系に力制御系を加えた場合の一般的な制御系を示す。 $G_1(s)$ は、ロボットマニピュレータの動特性、 $G_2(s)$ は対象物の動特性となる。このように、対象物の動特性がフィードバックループ内に含まれるため、対象物の動特性が変化することにより制御系全体の特性が変化してしまう。したがって、従来の固定フィードバックゲインによる制御では、対象物の動特性が設計値より大きく異なると制御性能が悪化してしまい、適応範囲が限られていた。取り扱う対象物によって適宜フィードバックゲインを変化させる必要がある。モデル規範型適応制御を用いた可変ゲインの制御方式[16][17]が提案されているが、制御系を設計するためには、厳密な数学モデルを必要とし、また、適応範囲もそれほど大きくない。

そこで、本章では、ニューラルネットワークを使用した制御方式を提案する。従来の制御方式と比較して、厳密な数学モデルを必要とせず、制御対象の大幅な特性変化に対しても適用できる。

本章では、図3.2に示すような、平面内で任意の位置と姿勢をとることができる3自由度のマニピュレータによる力制御系を対象とする。手首に設けた歪ゲージ等を利用した力センサで接触力を検出し、ハンドで把持した物体と固定対象物との間の接触力を制御しようとするものである。この制御動作は、ハンドを固定対象物の反作用面に垂直な姿勢に保てば、1自由度の押し付け動作として取り扱うことができる。1自由度マニピュレータの力制御について、

- (1) 力センサ部分の動特性を考慮したシミュレーション
- (2) シミュレーションによる神経回路モデルのユニット数と中間層数などの検討
- (3) 実験による制御性能についての考察

を行った結果について述べる。

3. 2 ニューラルネットワークによる力制御系の構成

ニューラルネットワークを使用した1自由度の力制御系の構成を図3. 3に示す。基本的には、離散値形速度形PID制御則を用い、そのフィードバックゲインをニューラルネットワークにて調整する構成とする。力制御方向におけるマニピュレータの位置と力の変化のパターンから対象物の動特性を推定し、それに応じてフィードバックゲインを変化させ制御性能の悪化を防ごうとするものである。制御系の構成は、一種のセルフチューニング形適応制御系となり、ニューラルネットワークはパラメータ推定器とみなすことができる。

力制御方向に対する位置と力の変化の過程から対象物の動特性を推定する場合、制御対象には、摩擦、歯車減速機構のバックラッシュ等の非線形要素や計測雑音などが含まれるため、その推定は困難である場合が多い。そこで、対象物の動特性推定に、ニューラルネットワークが有効となる。

ニューラルネットワークは、多層フィードフォワード型ネットワークを用い、図3. 3に示すように、入力信号はマニピュレータの位置、対象物との接触力および力の目標値とし、出力信号はPID制御の各ゲイン値とする。ニューラルネットワークは3層とし、各層のユニット数は入力層が3個、出力層が3個、中間層が5個とした。学習はバックプロパゲーションを用い、3層すべてのユニットに対して行い、1サンプリング毎に結合の重みを変化させた。本方式では、既知の対象物にて安定な良好な応答を示すフィードバックゲインを実験的に求めたり、あるいは根軌跡法などの公式的な最適性の規範を用いてフィードバックゲインを決定し、それを教師信号とする方法を用いた。

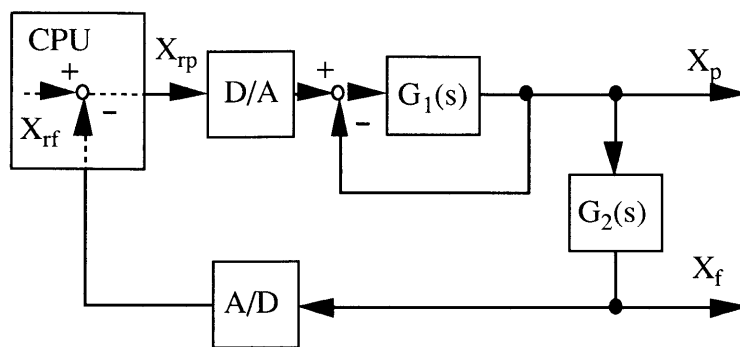


図 3. 1 一般の制御系

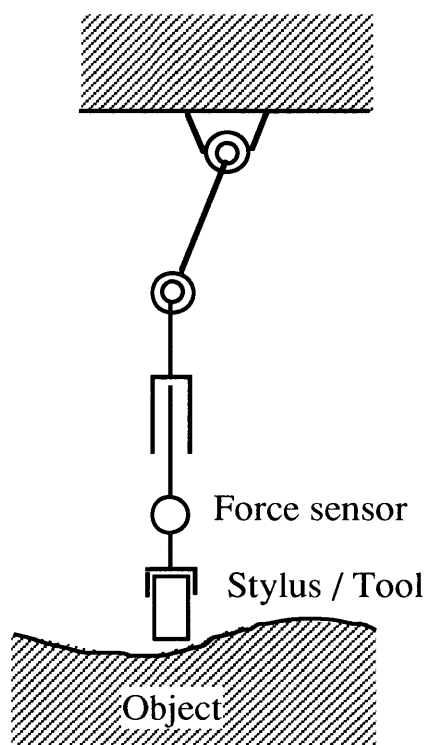
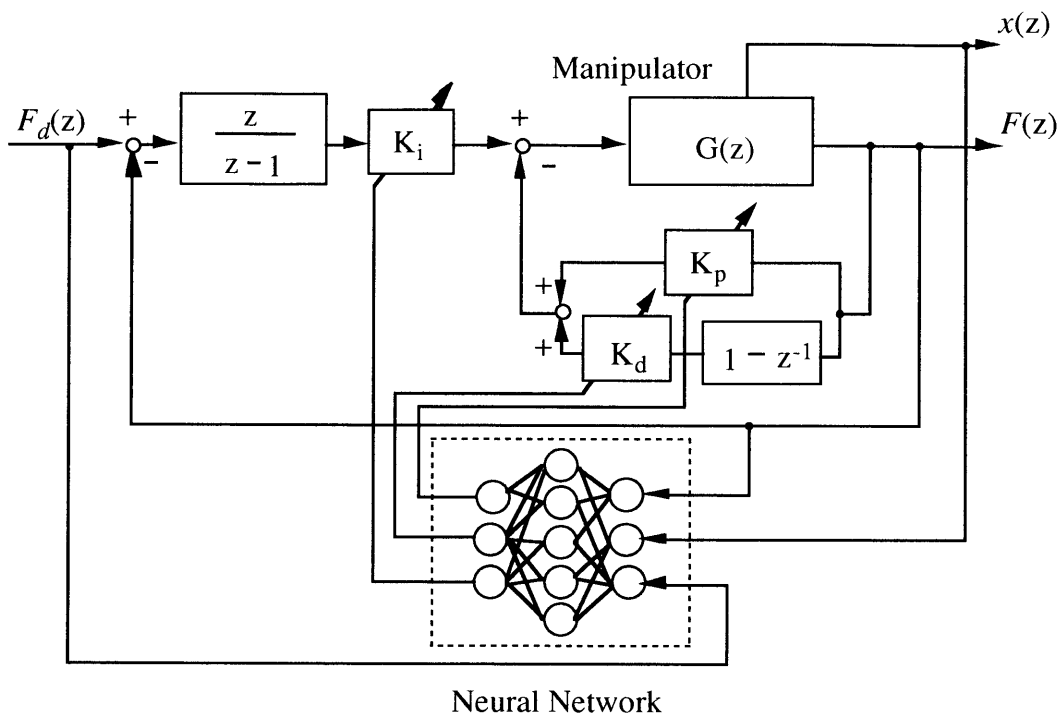


図 3. 2 マニピュレータの動作



$x(z)$: マニピュレータの位置 $F(z)$: 力センサ出力

$F_d(z)$: 力の目標値 z : z 変換の演算子

$G(z)$: 対象物の特性を含めたマニピュレータの伝達関数

K_p : 比例ゲイン K_i : 積分ゲイン K_d : 微分ゲイン

図 3. 3 制御系の構成

3. 3 シミュレーション方法と結果

3. 3. 1 制御対象の数学モデル

本章では、図 3. 4 に示されるような、電圧制御される DC モータの出力を送りねじ機構にてアームの並進機構に変換し、アームにより対象物を押し付ける、1 自由度の押し付け動作を取り上げる。対象物との相互作用を考え、力センサー部分の動特性を考慮に入れるとその数学モデルは図 3. 5 に示すように表される。対象物と力センサは線形なばね、粘性係数でモデル化した。力センサより先端部分の質量 M_p が無視できない場合 K_x でセンシングされた力は、 K^* で作用するものと一致しないし、 $x(t)$ でセンシングされた位置情報は $w_x(t)$ とは一致しない。本方式では、 K_x でセンシングされた力情報、 $x(t)$ でセンシングされた位置情報から対象物の動特性の推定を行うため、力センサ部分の動特性を考慮に入れた検討が必要となる。図 3. 5 のモデルに基づき、動特性方程式を導くと以下のようなになる。

$$\text{増幅器： } V(t) = A_m u(t) \quad (3.1)$$

$$\text{DCモータ： } L_a di(t)/dt + R_a + E(t) = V(t) \quad (3.2)$$

$$\text{逆起電力： } E(t) = K_e \dot{\theta}_m(t) \quad (3.3)$$

$$\text{発生トルク： } T(t) = K_t i(t) \quad (3.4)$$

$$\text{モータ側の釣合： } J_m \ddot{\theta}_m(t) + B_m \dot{\theta}_m(t) = T(t) - T_l \quad (3.5)$$

負荷側の釣合：

質量 M について

$$M\ddot{x}(t) + B_l \dot{x}(t) + C_x \{\dot{x}(t) - \dot{w}_x(t)\} + K_x \{x(t) - w_x(t)\} = 2\pi T_l / p_l \quad (3.6)$$

質量 M_p について

$$M_p \ddot{w}_x(t) + C^* \dot{w}_x(t) + K^* w_x(t) - C_x \{\dot{x}(t) - \dot{w}_x(t)\} - K_x \{x(t) - w_x(t)\} = 0 \quad (3.7)$$

モータも回転角とアームの移動距離の関係：

$$\theta_m(t) = 2\pi x(t) / p_l \quad (3.8)$$

ここで、

A_m : 増幅器のゲイン B_m : モータの摩擦係数

K_t : モータのトルク定数 K_e : モータの逆起電力定数

J_m : モータの慣性モーメント

M : 負荷質量 M_p : 先端質量

B_l : 負荷の摩擦係数 R_a : モータの電機子抵抗

K^* : 対象物のばね定数 C^* : 対象物の粘性係数

K_x : 力センサとマニピュレータのばね定数

C_x : 力センサとマニピュレータの粘性係数

L_a : モータのインダクタンス p_t : 送りねじのリード

T_l : 負荷によるトルク $T(t)$: 発生トルク

$u(t)$: 入力電圧 $V(t)$: モータ端子電圧

$E(t)$: 逆起電力 $i(t)$: モータ電流

$\theta_m(t)$: モータ回転角 $x(t)$: アームの位置

w_x : 力センサのたわみ

モータのインダクタンス L_a 、およびモータの摩擦抵抗 B_m を無視して、各式を整理すると次の様になる。

$$\ddot{x}(t) = -a_1 x(t) - a_2 \dot{x}(t) - a_3 w_x(t) - a_4 \dot{w}_x(t) + b_1 u(t) \quad (3.9)$$

$$\ddot{w}_x(t) = -a_5 x(t) - a_6 \dot{x}(t) - a_7 w_x(t) - a_8 \dot{w}_x(t) \quad (3.10)$$

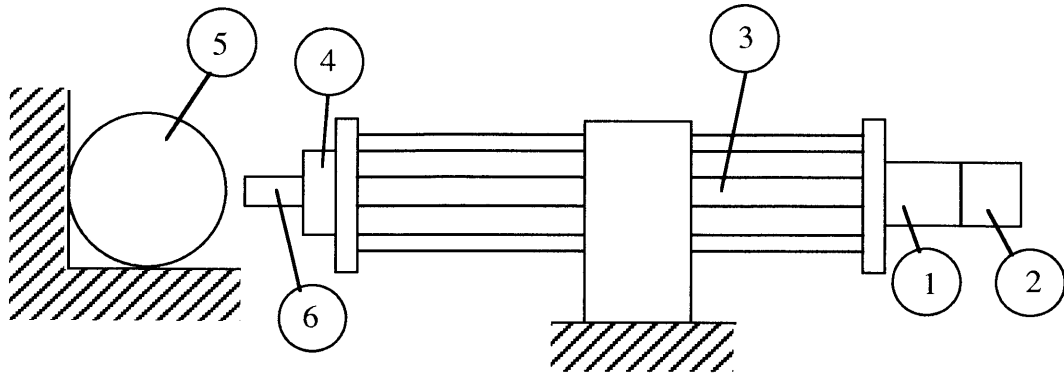
$$a_0 = 4\pi^2 J_m + p_t^2 M, \quad a_1 = p_t^2 K_x / a_0, \quad a_2 = \{p_t^2 R_a (B_l + C_x) + 4\pi^2 K_e K_t\} / (R_a a_0),$$

$$a_3 = -p_t^2 K_x / a_0, \quad a_4 = -p_t^2 C_x / a_0, \quad a_5 = -K_x / M_p, \quad a_6 = -C_x / M_p,$$

$$a_7 = (K_x + K^*) / M_p, \quad a_8 = (C_x + C^*) / M_p, \quad b_1 = 2\pi A_m p_t K_t / (R_a a_0)$$

力 $F(t)$ は、

$$F(t) = C_x \{\dot{x}(t) - \dot{w}_x(t)\} - K_x \{x(t) - w_x(t)\} \quad (3.11)$$



1. Motor 2. Encoder 3. Ball Screw
 4. Force Sensor 5. Object 6. Arm

図 3. 4 1 自由度マニピュレータの構造

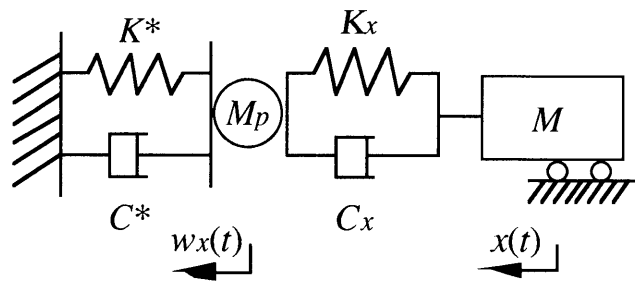


図 3. 5 1 自由度マニピュレータの数学モデル

3. 3. 2 固定ゲイン方式との比較

シミュレーションは、固定フィードバックゲイン方式とニューラルネットワークによる可変ゲイン方式について行った。対象物は柔らかい物としてばね定数 1×10^3 (N/m)、硬い物としてばね定数 3×10^4 (N/m) とした。また、力センサーのばね定数は 2×10^5 (N/m)、質量 M_p は 0.1 (kg) とした。ニューラルネットワークの学習の定数は学習係数 $\eta = 0.01$ 、安定化定数 $\alpha = 0.3$ とした。図 3. 6 に固定ゲイン方式による結果を示す。図は、マニピュレータの位置 x 、力 F 、モータの端子電圧 MV を示している。力 F の図における破線は力の目標値 F_d を示す。図に示すように、ゲインの選び方によっては対象物の動特性の変化に対して適応できず、発振現象を起こしてしまう。図 3. 7 にニューラルネットワークによる可変ゲイン方式の結果を示す。本方式により、対象物の動特性変化に対して適応性が確実に向上することがわかる。

3. 3. 3 ニューラルネットワークの構成についての検討

＜目標値変化に対するニューラルネットワークの応答＞

学習パターンにおける力の目標値変化を 1 N と 5 N の 2 点とした場合、各対象物のばね定数の定常状態での位置、力、力の目標値の関係を表 3. 1 に示す。表から分かるようにばね定数が 1×10^3 N/m と 3×10^4 N/m の 2 点のみの学習では、対象物の推定を位置情報だけで行うことが可能となり、ニューラルネットワークの入出力関係が位置の大きさのみに依存してしまう。従って、図 3. 8 に示すように、目標値の変化に対してニューラルネットワークの出力すなわちフィードバックゲインが大きく変化してしまっている。この現象は、特に未学習の対象物に対して顕著となることから、学習する点を増やすことによりゲインの変化を抑えることができる。図 3. 9 に 4 点 (1×10^3 , 5×10^3 , 1.5×10^3 , 3×10^3 N/m) 学習した場合のゲインの変化の様子を示す。学習する値は、 1×10^3 N/m と 3×10^3 N/m でのゲインの値を結ぶ直線上の値とした。目標値変化に対するゲインの変化は、2 点学習の時に比べて $1/3$ に減少していることが分かる。図 3. 8、図 3. 9 と同様の比較を、学習の点数とニューラルネットワークの出力誤差の最大値、ゲインの変動の最大値の

関係で表したものを図3. 10に示す。図から学習の点数を増やせば誤差、ゲインの変動を抑えられることが分かるが、制御対象のばね定数が大きく変動する場合、少なくとも4点は学習点が必要であると言える。

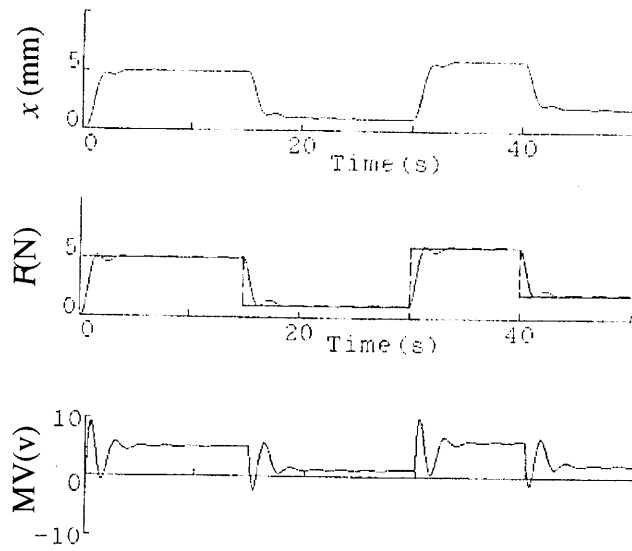
なお、図3. 8から図3. 10は比例ゲイン K_p のみについて示してあるが、積分ゲイン K_i 、微分ゲイン K_d も同様の傾向となった。

<中間層数>

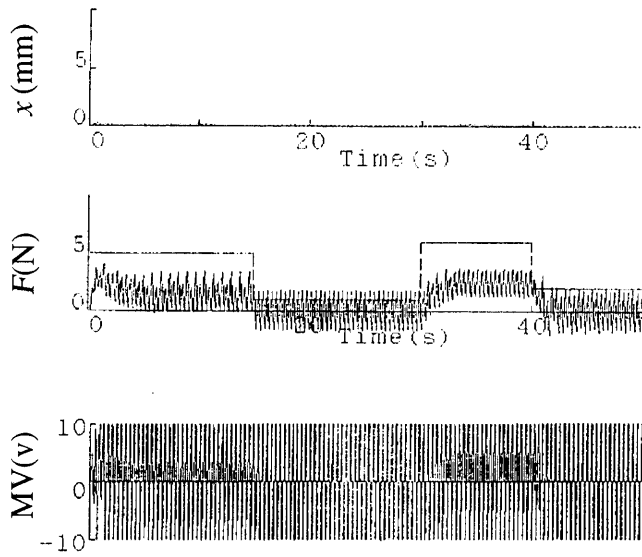
入力層、出力層のユニットを線形出力とし、中間層のユニットをシグモイド出力としたニューラルネットワークで、学習を出力層と中間層のユニットだけに行った場合、図3. 11(a)に示す3層モデル（中間層1層）では、学習が収束せず、4層（図3. 11(b)）必要であった。しかし、図3. 11(c)に示す様な入力層もシグモイド出力とした3層のモデルで、学習を入力層のユニットまで行うことにより図3. 11(b)のモデルとほとんど同じ出力結果を得ることができた。図3. 11(c)のモデルの方が図3. 11(b)のモデルと比較して学習するシナプス結合の数が少なくて済み、学習速度も速い。

<中間層のユニット数>

図3. 11(c)のモデルで中間層のユニットの数を2、5個とした場合のニューラルネットワークの出力を図3. 12、図3. 13に示す。図3. 9と同様に4点学習した。図3. 14にユニット数とニューラルネットワークの出力誤差の最大値、ゲインの変動の最大値の関係を示す。中間層のユニット数が増えるほど誤差が少なくなり、目標値変化に対する変動も少なくなることが分かる。ただしユニット数が5個以上になるとあまり大きな変化は見られないことから、中間層のユニット数は5個程度で良いと言える。

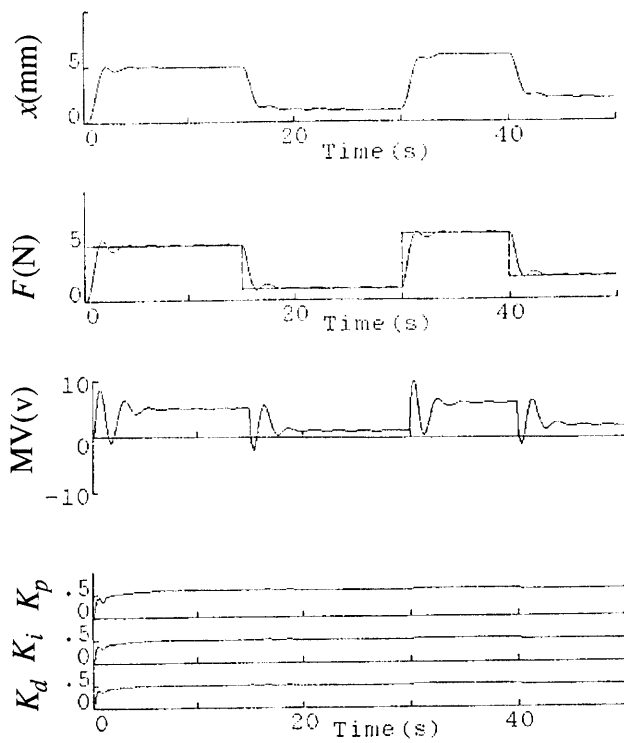


(a) soft object

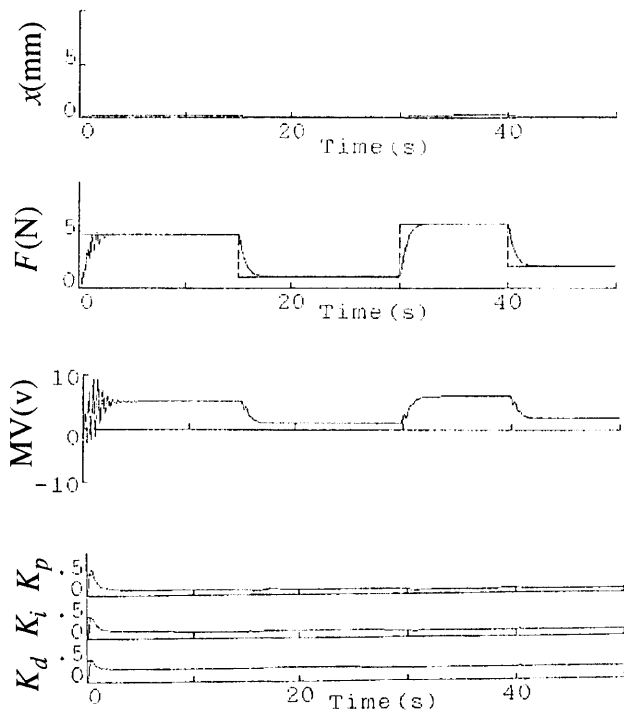


(b) hard object

図 3. 6 固定ゲイン方式のシミュレーション結果



(a) soft object



(b) hard object

図 3. 7 (a) 可変ゲイン方式のシミュレーション結果

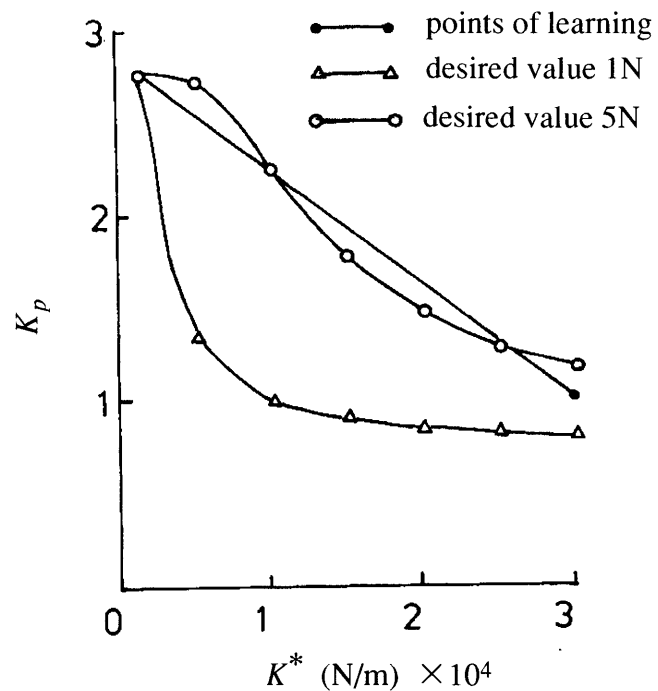


図3. 8 コントローラのゲイン
(学習点の数：2)

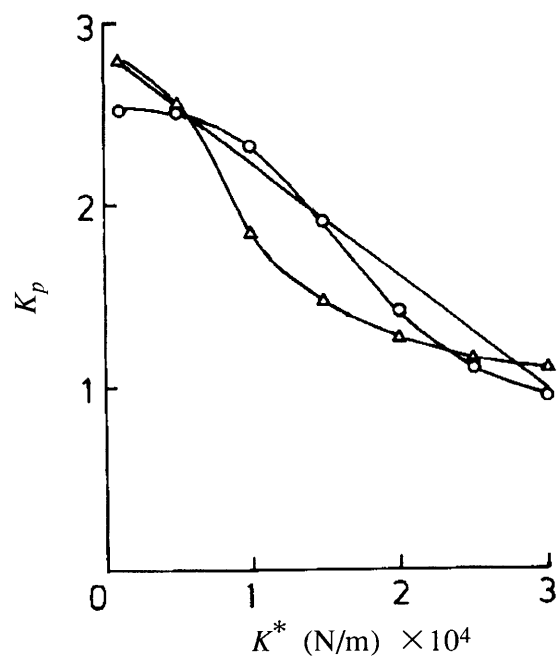


図3. 9 コントローラのゲイン
(学習点の数：4)

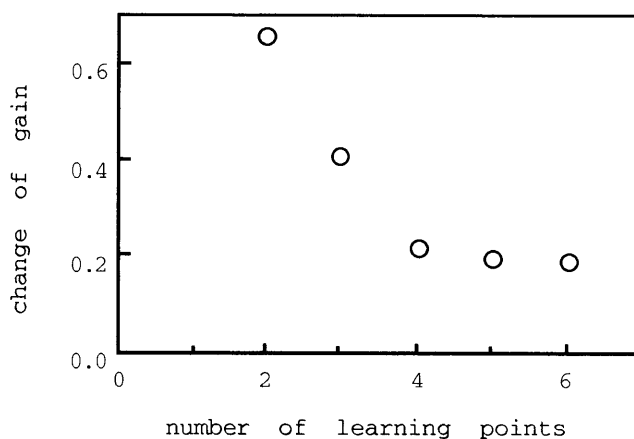
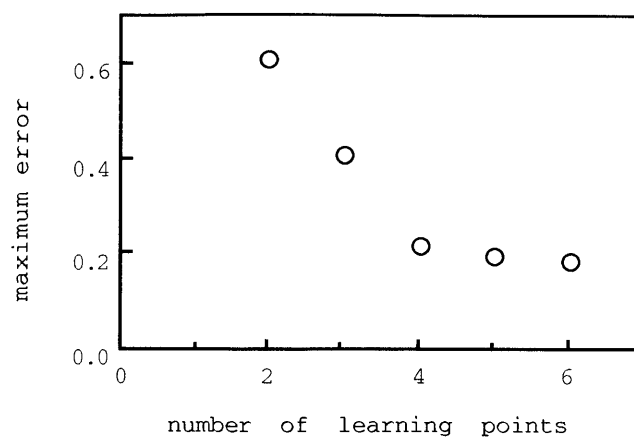


図 3. 1 0 学習点の数と最大誤差の関係

表 3. 1 対象物のバネ定数に対する
マニピュレータの力と位置の関係

Stiffness $\times 10^3$ (N/m)	Position $\times 10^{-3}$ (m)	
	Force 1 (N)	Force 5 (N)
1	1.00	5.00
5	0.20	1.00
15	0.07	0.33
30	0.03	0.17

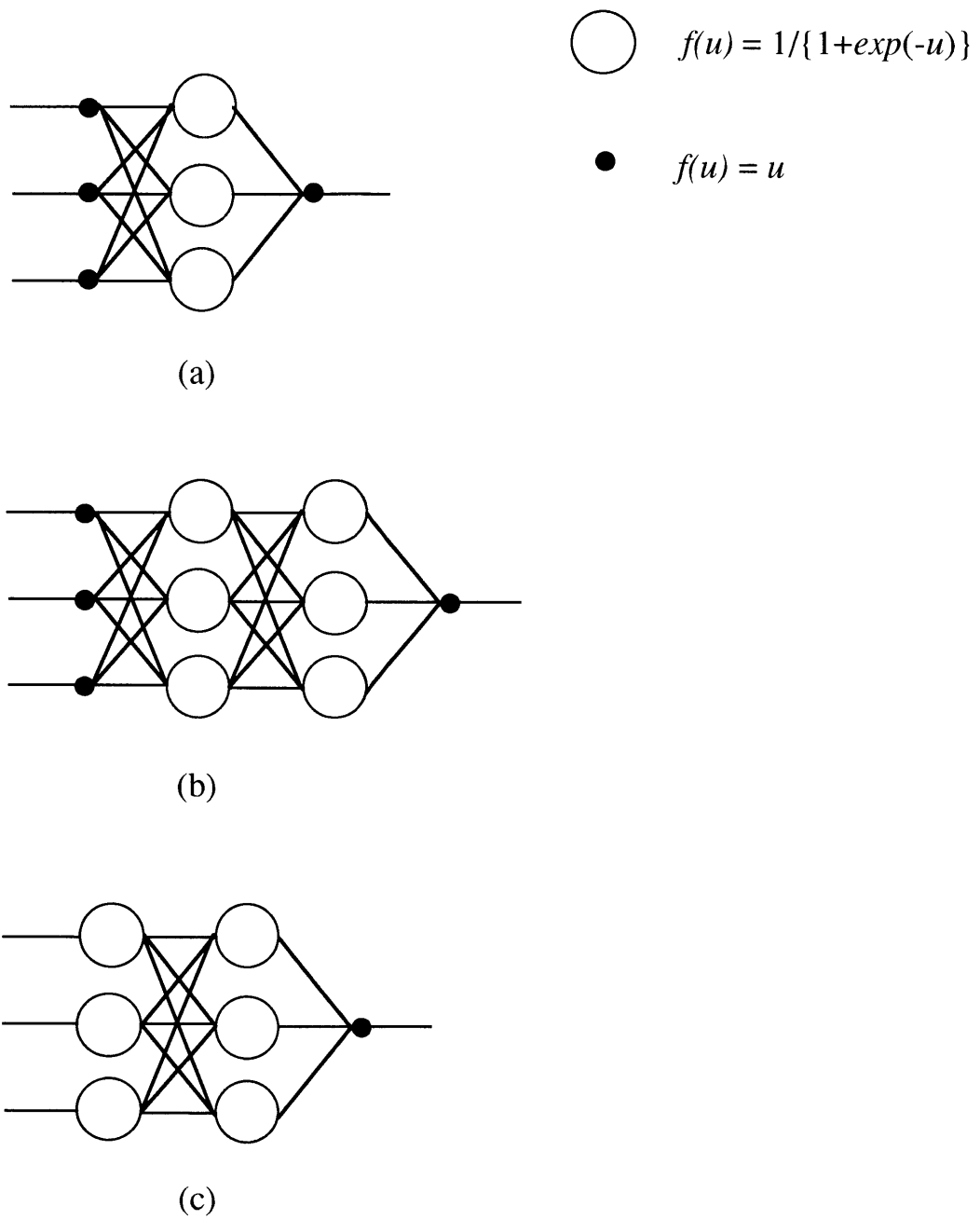


図 3. 1 1 ニューラルネットワークの構造

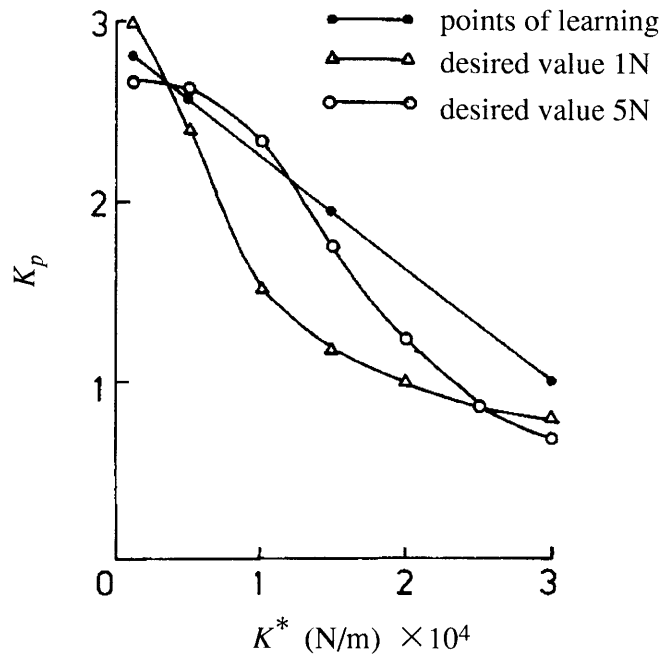


図3. 12 コントローラのゲイン
(中間層のユニット数：2)

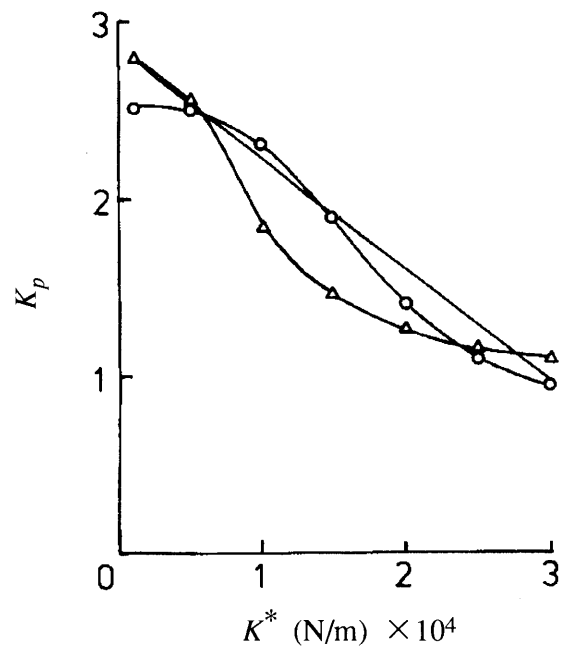


図3. 13 コントローラのゲイン
(中間層のユニット数：5)

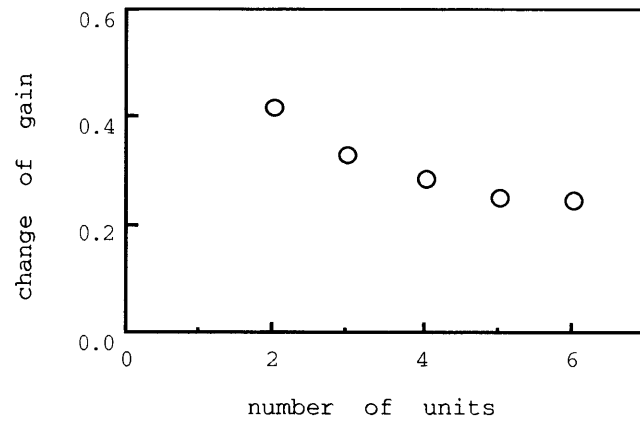
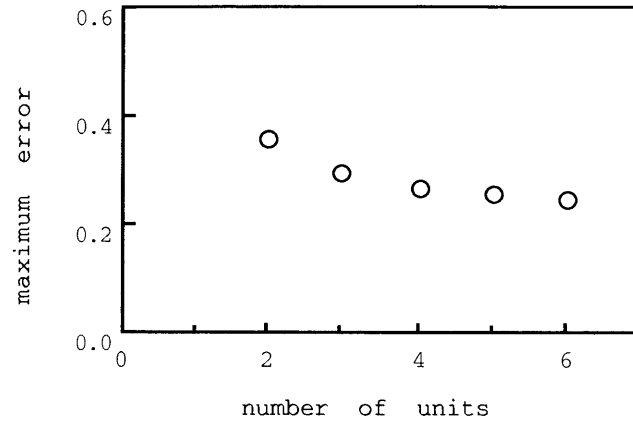


図3. 14 中間層のユニット数と最大誤差の関係

3. 4 実験方法と結果

3. 4. 1 実験装置と実験方法

図3. 15 に実験装置を示す。アクチュエータとしてDCモータを使い、モータ軸に送りねじ（ボールねじ）を付けた回転-並進変換機構とし、送りねじの先端で対象物を押し付ける構造となっている。位置の出力はモータ軸に付けたエンコーダで、力はマニピュレータ先端に取り付けた力センサー（歪ゲージ）で検出する。ニューラルネットワークおよび制御則は、パーソナルコンピュータ（CPU 8086, FPU 8087)のソフトウェアにて実現した。制御のサンプリング時間は20 (ms)とした。また、学習の定数は学習係数 $\eta = 0.01$, 安定化定数 $\alpha = 0.3$ とした。

取り扱う対象物の大きさは未知となるため、まず認識動作を行った。認識動作は、速度制御を行いながら一定速度で対象物に近付け、力センサーの出力が設定値（0.2N）を越えたところで対象物との接触を認識するものとした。その位置から、固定フィードバックゲイン、可変フィードバックゲインによる制御を行った。対象物は、柔らかい対象物として、硬式のテニスボール（ $\phi 64\text{mm}$,ばね定数 $4.9 \times 10^3 \text{ N/m}$ ）、硬い対象物としてガラスのコップ（ $\phi 70\text{mm}$,ばね定数 ∞ ）を用いた。力センサーおよび対象物のばね定数を調べるために、それらの力と変位の関係を測定した結果を図3. 16 に示す。

3. 4. 2 実験結果

図3. 15 の実験装置を使用して、押し付け動作を行った結果を図3. 17 から図3. 21 に示す。図はすべて、認識動作を行った後の出力を示している。それぞれ、位置 x 、力 F 、モータ端子電圧 MV 、ニューラルネットワークの出力（ K_p , K_i , K_d ）を示す。力 F の図における破線は力の目標値 F_d を示す。

図3. 17 と図3. 18 は、同一の固定フィードバックゲインを用いて押し付け動作を行った結果である。柔らかい対象物で図3. 17 のような応答をするフィードバックゲインで、硬い対象物を押し付けると発振現象を起し制御性能が劣化してしまう。このように、対象物の動特性の変化に対し、適応範囲が限られてしまう

ことがシミュレーションと同様、実験でも明かとなった。本章で提案する可変フィードバックゲイン方式を用いて押し付け動作を行った結果が図3.19と図3.20である。柔らかい対象物でも、硬い対象物でも安定した制御動作が実現でき、対象物の変化に対して、適応性が広がることが明かとなった。図3.21は、可変フィードバックゲイン方式による未学習の対象物に対する押し付け動作の結果である。未学習対象物に対しても良好な応答結果が得られている。

3.5 まとめ

本章で提案しているニューラルネットワークを応用した適応力制御について、シミュレーションと実際の実験を行い、次の結果を得た。

(1) 対象物の変化に対して固定フィードバックゲイン制御方式と比較して、安定で、適応範囲が広がり、柔軟に対応できること。

(2) ニューラルネットワークの各種構成につき検討した結果、入力層のユニットの出力関数をシグモイドとし、入力層まで学習すれば、シナプス結合の数を少なくでき、学習速度を速めることができること。

(3) 学習の点数は4点以上、中間層のユニット数は5個程度にすれば、神経回路モデルの出力誤差を少なくし、力の目標値変化に対するゲインの変動を抑えることができること。

また、本章では1自由度マニピュレータを対象として論じたが、力制御の自由度におけるエンドエフェクタの位置と力を測定できれば対象物を推定できるので、多自由度マニピュレータにも拡張することも可能である。多自由度マニピュレータへの本方式の拡張については、次章で述べる。

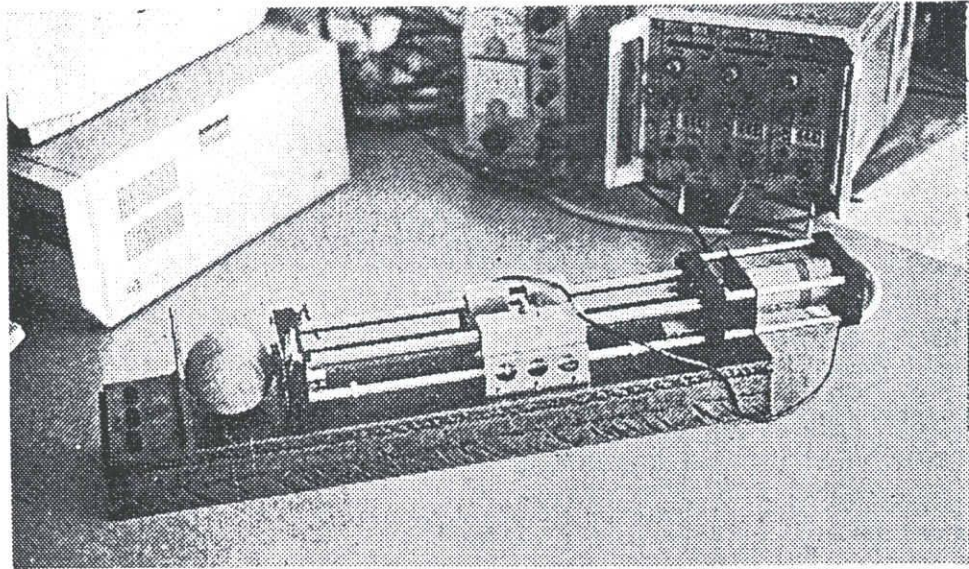


図 3. 15 実験装置

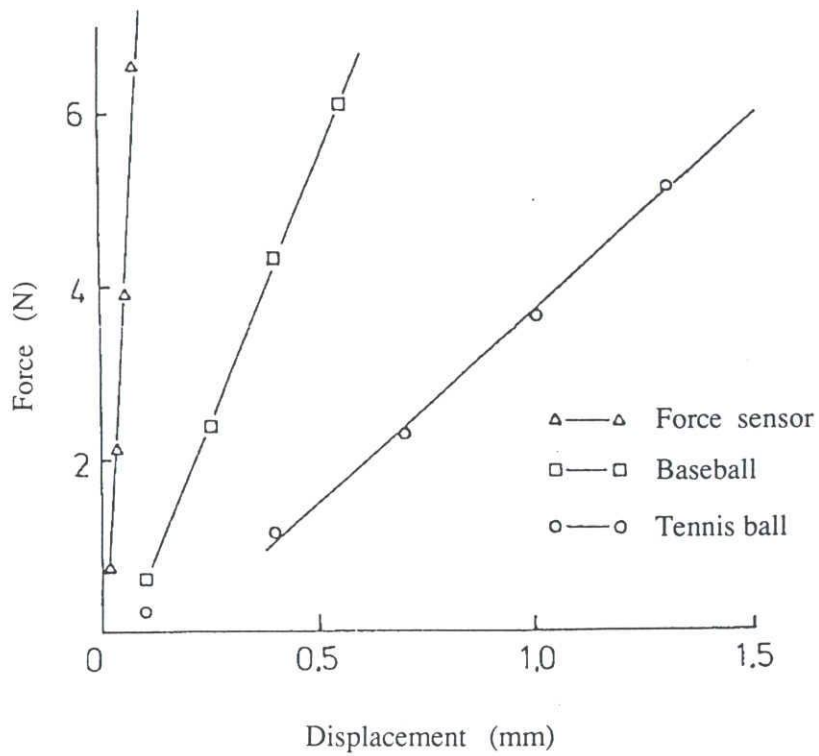


図 3. 16 対象物の力と変位の関係

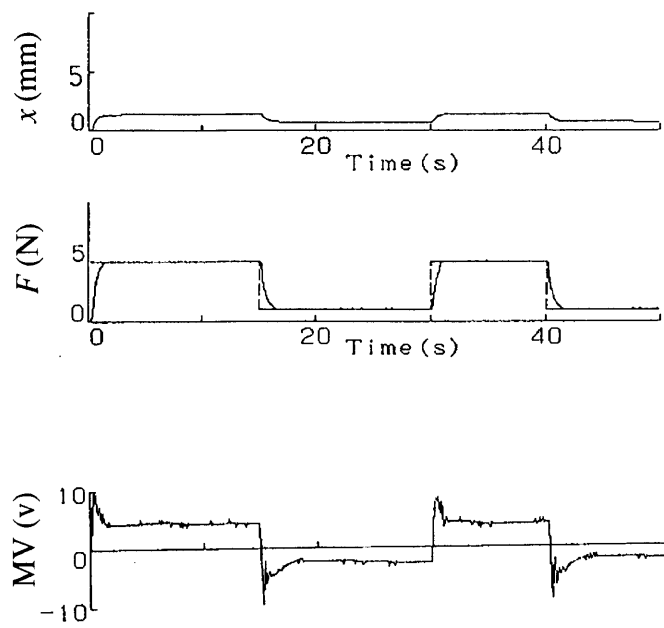


図3. 17 固定ゲイン方式による実験結果
(柔らかい対象物)

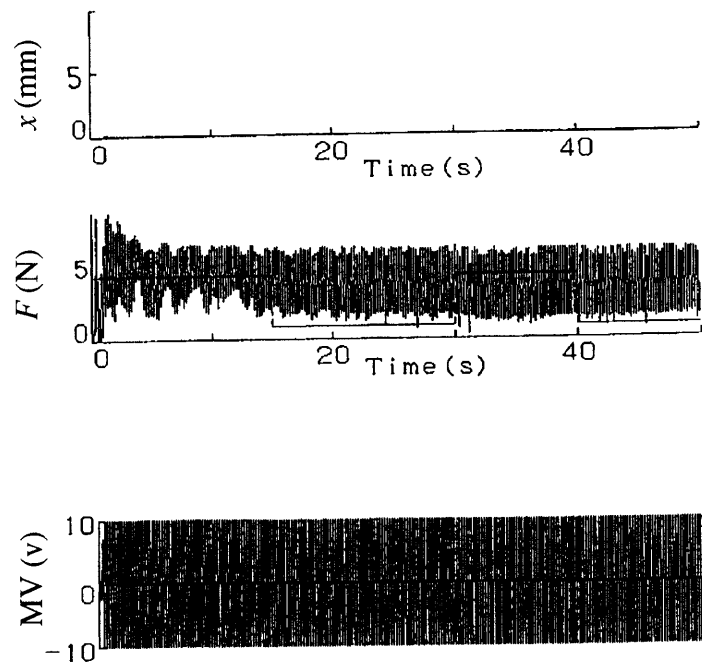


図3. 18 固定ゲイン方式による実験結果
(硬い対象物)

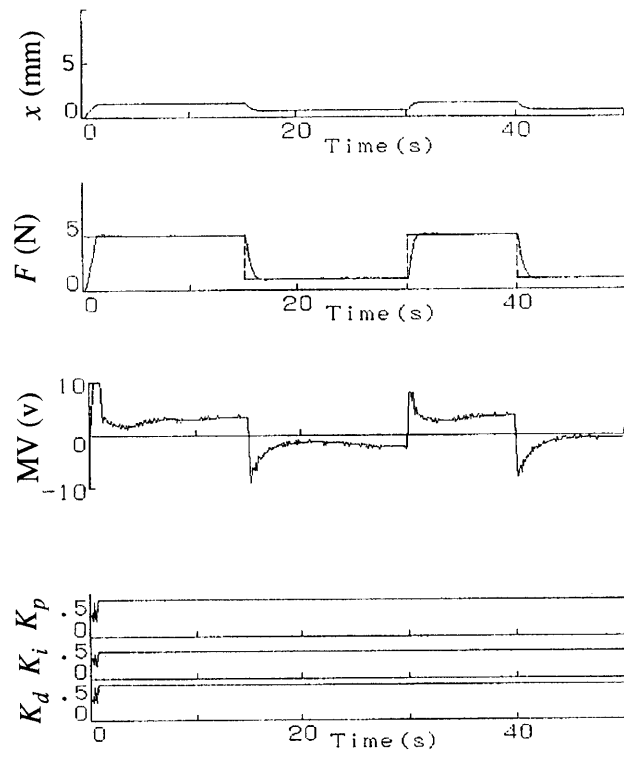


図3. 19 可変ゲイン方式による実験結果 (柔らかい対象物)

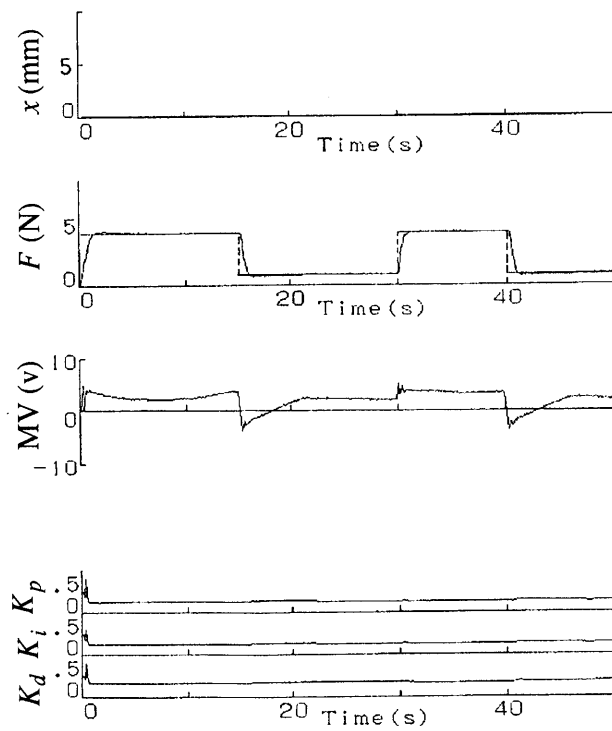


図3. 20 可変ゲイン方式による実験結果 (硬い対象物)

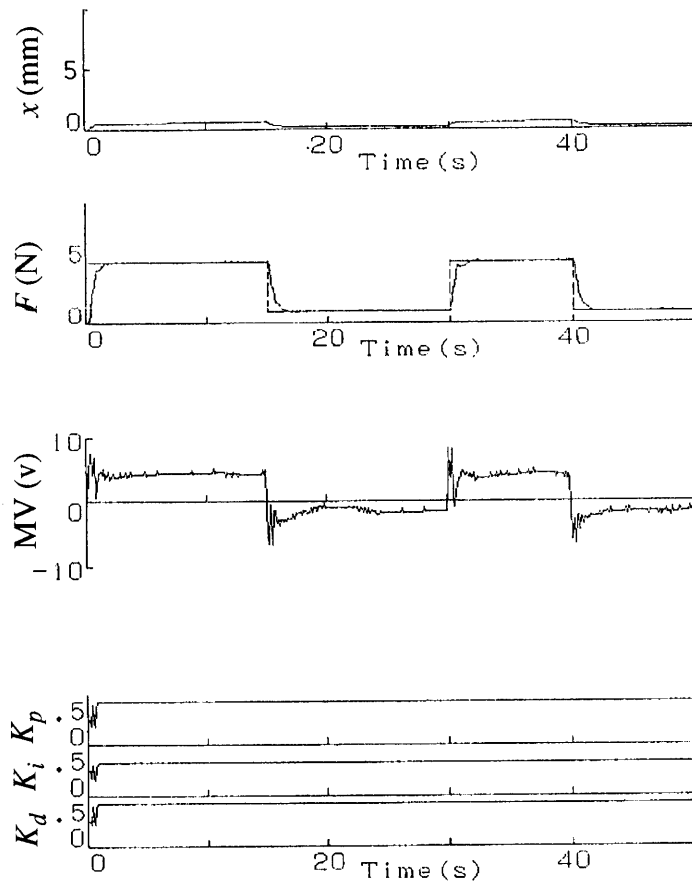


図3. 21 可変ゲイン方式による実験結果
(未学習対象物：野球ボール)

第4章 ニューラルネットワークによる 対象物の特性を考慮した位置と力の 適応ハイブリッド制御

本章では、前章で述べたニューラルネットワークを用いた力制御方式を多自由度マニピュレータの位置と力のハイブリッド制御系に拡張する場合について述べる。

4.1 マニピュレータの位置と力のハイブリッド制御

図4.1に示すような、多関節型のロボットマニピュレータによる位置と力のハイブリッド制御系を対象とする。壁に沿う方向には位置と姿勢を制御し、壁に垂直な方向に対しては、手首に設けた力センサで接触力を検出し、ハンドで把持した物体と固定対象物との間の接触力を制御しようとするものである。マニピュレータによるグライディングや刷毛塗りといった作業に適用できる。

ハイブリッド制御の基本的な考えはRaibertとCraigによって提案され[18]、作業環境によってエンドエフェクタが拘束されている方向には力を制御し、それ以外の方向には、位置と姿勢を制御する制御方式である。制御系は力制御フィードバックループと位置制御フィードバックループを重ね合わせた構成となっている。したがって、前章でも述べたように、対象物の動特性が閉ループ内に入るため、対象物の動特性が大きく異なる場合や未知対象物に対して、安定な制御が困難となる。また、多自由度マニピュレータは非線形システムであり、従来のハイブリッド制御方式における姿勢変化による不安定性が議論されている[19][20][21]。そこで本章では、ニューラルネットワークを使用して、対象物の動特性の変化や姿勢変化に対してロバストなハイブリッド制御システムの構築が可能であることを述べる。

そして、2自由度マニピュレータを対象に、提案したニューラルネットワークによるハイブリッド制御系のシミュレーションおよび実験を行い、対象物の動特性の変化に対するロバスト性、マニピュレータの姿勢変化に対するロバスト性について評価し、本方式の有効性、実現性について示す。

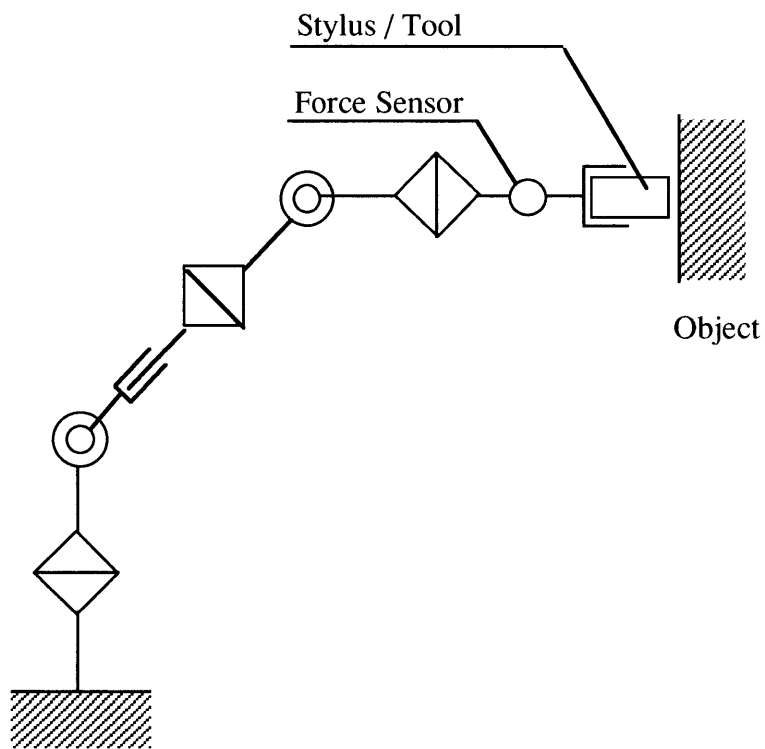


図 4. 1 多関節型マニピュレータによるハイブリッド制御

4. 2 ニューラルネットワークによるハイブリッド制御系の構成

ニューラルネットワークモデルを使用した制御系の構成を図4. 2に示す。本方式では、ニューラルネットワークで直接ロボットを制御するのではなく、ニューラルネットワークは、制御対象のパラメータ推定器およびフィードバックゲインの設定機構として働く。なお、本方式によるハイブリッド制御系は、図4. 2に示すように、RaibertとCraigの提案によるハイブリッド制御則[18]を用い、そのフィードバックゲインをニューラルネットワークにて調整する構成とする。本図におけるマニピュレータの各関節の駆動トルク τ は次式のようにになる。

$$\tau = \tau_p + \tau_f \quad (4.1)$$

$$\tau_p = K_{pp}q_e(t) + K_{pi} \int q_e(t)dt + K_{pd}\dot{q}_e(t) \quad (4.2)$$

$$\tau_f = K_{fp}\tau_e(t) + K_{fi} \int \tau_e(t)dt \quad (4.3)$$

$$q_e(t) = J^{-1}(I - S)\{X_d(t) - \Lambda(q(t))\} \quad (4.4)$$

$$\dot{q}_e(t) = J^{-1}(I - S)\{\dot{X}_d(t) - J\dot{q}(t)\} \quad (4.5)$$

$$\tau_e(t) = J^T S\{F_d(t) - TF(t)\} \quad (4.6)$$

ここで、

K_{pp} 、 K_{pi} 、 K_{pd} ：位置フィードバックゲイン、

K_{fp} 、 K_{fi} ：力フィードバックゲイン、

J ：ヤコビアン行列、 S ：コンプライアンス選択行列、

Λ ：座標変換行列、 T ：力変換行列、

X ：ワーク座標系の位置ベクトル、

q ：関節座標系の位置ベクトル、 F ：力ベクトル、

t ：時間。

ニューラルネットワークは、図4. 3に示すような、3層のフィードフォワード型ネットワークを用い、入力信号はマニピュレータの各関節の角度（変位）、ワーク座標系における力制御方向のエンドエフェクタの位置、対象物との接触力および力の目標値とし、出力信号はPID制御の各ゲイン値とする。学習はバックプロパゲイションを用い、3層すべてのユニットに対して行い、1サンプリング毎に結合

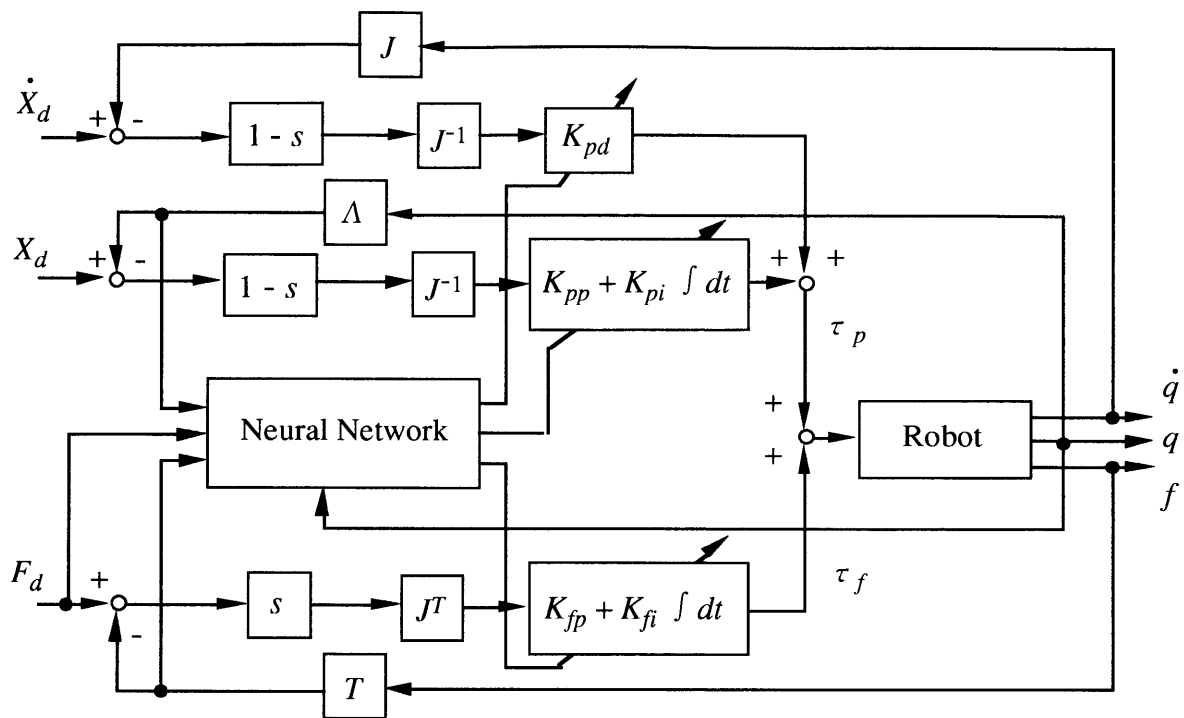


図4. 2 制御系の構成

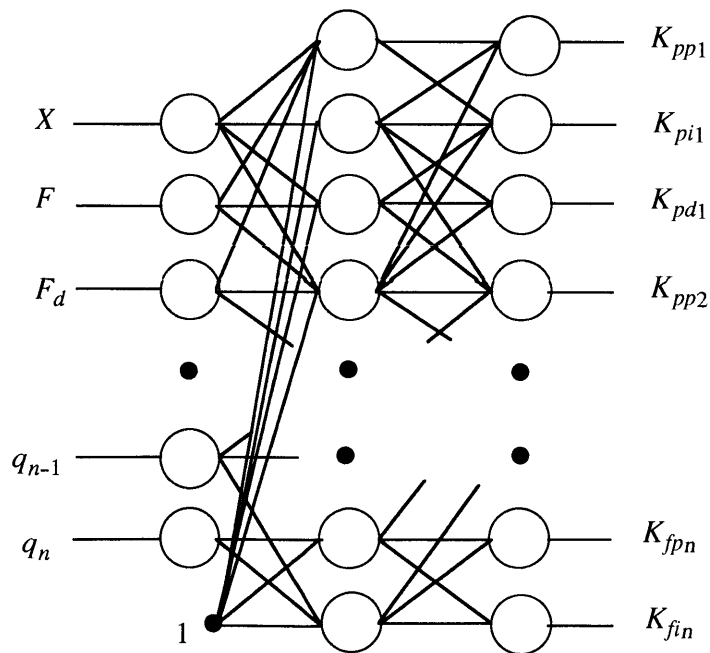


図4. 3 ニューラルネットワークモデル

の重みを変化させる。教師信号には、既知の対象物にて実験的に求めた各フィードバックゲイン値を用いる。

4. 3 シミュレーション方法と結果

4. 3. 1 制御対象の数学モデルと制御システム

シミュレーションは図4. 4に示す2自由度のマニピュレータを対象とする。水平面内で関節1は回転運動、関節2は並進運動を行う機構とする。対象物の拘束面は、y軸に平行に置く、従ってx軸方向が力制御、y軸方向が位置制御となる。対象物と力センサー部分の動特性は図4. 4に示すような線形の粘弾性モデルで近似する。なお、マニピュレータのエンドエフェクタと対象物との摩擦は考慮していない。このモデルに基づき運動方程式を導くと次のようになる。

$$\ddot{q}_1 = (\tau_1 - C_1 \dot{q}_1 + q_2 K_w w_2 + l K_w w_1) / I_1 \quad (4.7)$$

$$\ddot{q}_2 = (\tau_2 - C_2 \dot{q}_2 + K_w w_1) / (M_2 + M_3) \quad (4.8)$$

$$\ddot{w}_1 = \{-K_w w_1 - C_w \dot{w}_1 - f_x \cos(q_1)\} / M_3 \quad (4.9)$$

$$\ddot{w}_2 = \{-K_w w_2 - C_w \dot{w}_2 - f_x \sin(q_1)\} / M_3 \quad (4.10)$$

$$f_x = K_r X_c + C_r \dot{X}_c \quad (4.11)$$

$$X_c = (q_2 + w_1) \cos(q_1) + (l - w_2) \sin(q_1) - X_r \quad (4.12)$$

ここで、

q_1 、 q_2 ：関節座標で表示したマニピュレータの位置、

w_1 、 w_2 ：力センサーのたわみ、

τ_1 ：関節1の駆動トルク、 τ_2 ：関節2の駆動力、

C_1 ：関節1の粘性係数、 C_2 ：関節2の粘性係数、

I_1 ：関節1における慣性モーメント、

C_w ：力センサーとマニピュレータの粘性係数、

K_w ：力センサーとマニピュレータの剛性、

M_3 ：先端質量、 l ：リンクのオフセット、

f_x ：対象物からの反作用力、 K_r ：対象物の剛性、

C_r ：対象物の粘性係数、 M_2 ：関節2の質量、

x_c : 対象物の変位、 x_r : 対象物表面の X 方向の位置。

これらのシミュレーションに用いた値を表 4. 1 に示す。また、2 自由度マニピュレータの制御システムにおける(4.1)式から(4.6)式の各値は、以下のようになる。

$$\begin{aligned} X &= [x \quad y]^T & q &= [q_1 \quad q_2]^T \\ \tau &= [\tau_1 \quad \tau_2]^T & F &= [f_x \quad f_y]^T \end{aligned}$$

$$\Lambda(q) = \begin{bmatrix} q_2 \cos(q_1) + l \sin(q_1) \\ q_2 \sin(q_1) - l \cos(q_1) \end{bmatrix} \quad T = \begin{bmatrix} \cos(q_1) & -\sin(q_1) \\ \sin(q_1) & \cos(q_1) \end{bmatrix}$$

$$J = \begin{bmatrix} l \cos(q_1) - q_2 \sin(q_1) & \cos(q_1) \\ q_2 \cos(q_1) + l \sin(q_1) & \sin(q_1) \end{bmatrix} \quad S = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}$$

ニューラルネットワークは、各層のユニット数を、入力層が 5 個、中間層、出力層がそれぞれ 10 個とし、学習の定数は、学習係数 $\eta = 0.01$ 、安定化定数 $\alpha = 0.3$ とした。また、学習した点は、柔らかい対象物と硬い対象物のそれぞれにおいて、図 4. 5 に示す 1/4 円をマニピュレータの動作範囲とし、範囲内の六ヶ所の点で行った。

4. 3. 2 押し付け制御のシミュレーション結果

押し付け制御のシミュレーションは、固定フィードバックゲイン方式とニューラルネットワークによる可変ゲイン方式について行った。対象物は柔らかい物としてばね定数 8×10^2 (N/m)、硬い物としてばね定数 5×10^4 (N/m) とした。図 4. 6 に固定ゲイン方式による結果を示す。図は、マニピュレータの力 F 、力制御方向の変位 ($x-x_0, x_0$: マニピュレータが対象物に接触したときの位置)、位置制御の偏差 ($y-y_d, y_d$: y_d 目標位置) を示している。図に示すように、ゲインの選び方によっては対象物の動特性の変化に対して適応できず、発振現象を起こしてしまう。図 4. 7 に本章で提案するニューラルネットワークによる可変ゲイン方式の結果を示す。本

方式により、対象物の動特性変化に対して適応性が確実に向上することがわかる。表 4. 2 に学習した 2 種類の対象物の中間のばね定数を持つ対象物に対して、その応答を自乗誤差にて比較した結果を示す。表は、ばね定数が 8×10^2 (N/m) の対象物における固定フィードバックゲイン方式による結果を 1 として正規化してある。固定フィードバックゲイン方式では対象物が硬くなるにつれて発振現象を起こすため誤差が増大しているのに比較して、可変フィードバックゲイン方式では誤差に大きな変化は見られない。未学習の対象物に対してもニューラルネットワークの補間機能により、安定な制御動作が可能であることがわかる。

次に、マニピュレータの姿勢変化に対するシミュレーション結果を図 4. 8 と図 4. 9 に示す。姿勢は、関節 1 の対象物拘束面に対する角度を変えず、関節 2 を縮めた場合と伸ばした場合について示してある。図 4. 8 は、固定フィードバックゲインを用いた押し付け制御の結果を示した場合であり、マニピュレータの姿勢変化に対して不安定になっていることがわかる。他方、図 4. 9 はニューラルネットワークによる可変ゲイン方式を用いた同様の結果を示す。固定フィードバックゲインの場合に比較して本章で提案するニューラルネットワークを用いた可変ゲイン方式の方が、マニピュレータの姿勢の大きな変化に対してロバストであることがわかる。

4. 3. 3 倣い制御のシミュレーション結果

倣い制御のシミュレーション結果を図 4. 10、図 4. 11 に示す。図 4. 10 は、対象物の動特性の変化に対する固定フィードバックゲイン方式の結果を示す。対象物の動特性の変化に対して不安定になることがわかる。図 4. 11 は、本章で提案する可変フィードバックゲイン方式の結果を示す。この図から倣い制御においても、対象物の動特性の変化に対して、ニューラルネットワークによる可変フィードバックゲイン方式が有効であることがわかる。

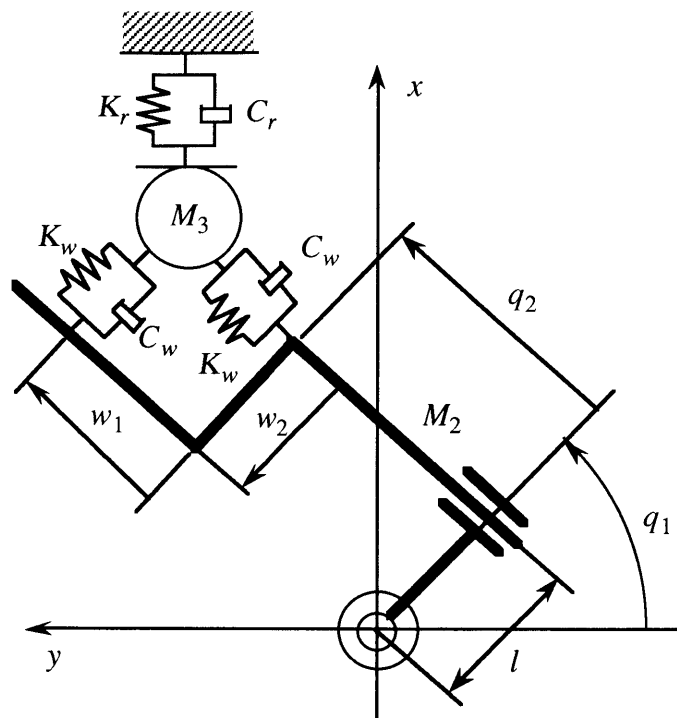


図 4. 4 マニピュレータの数学モデル

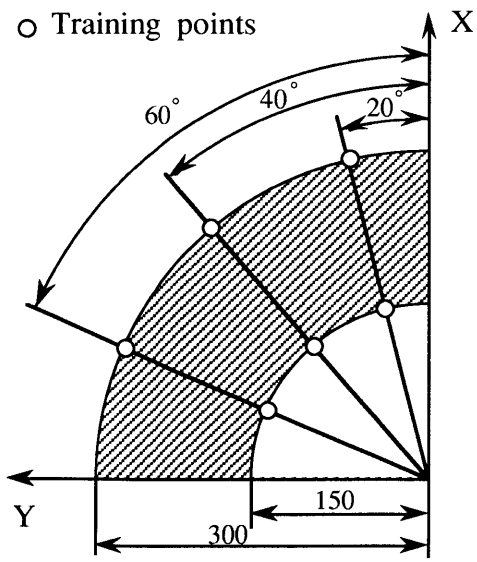
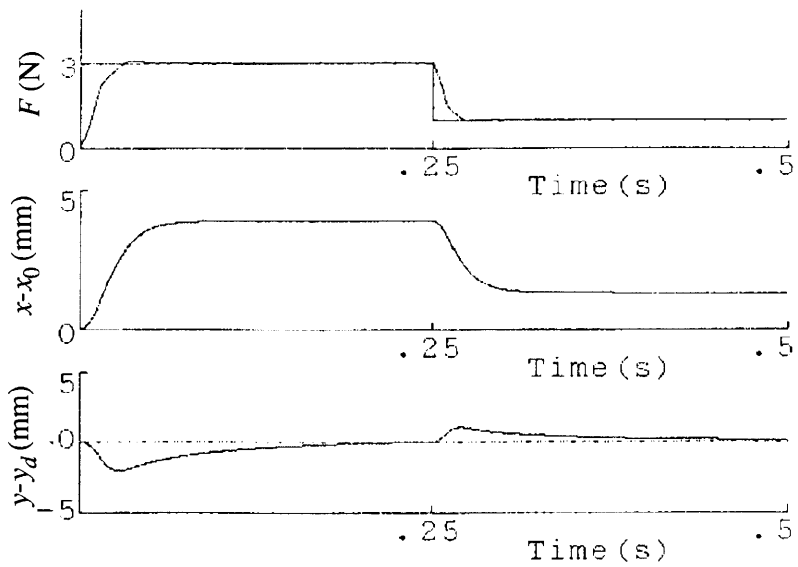
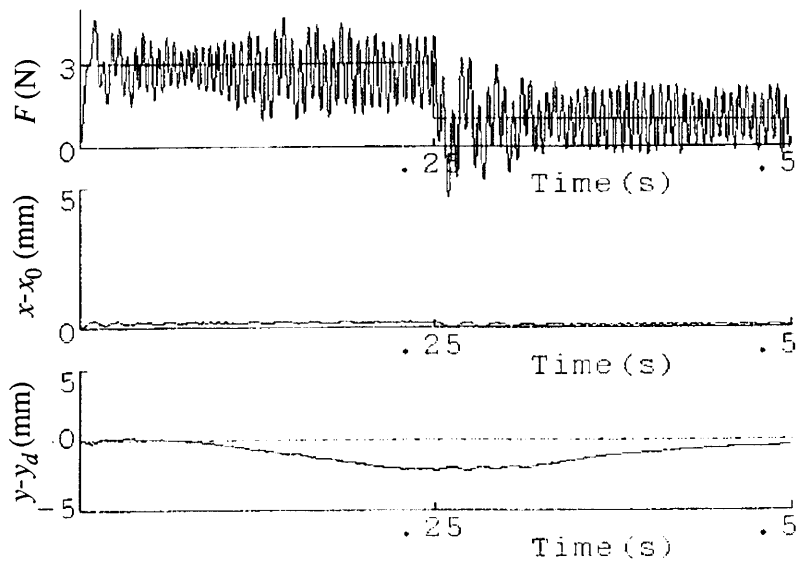


図 4. 5 マニピュレータの動作範囲と学習点

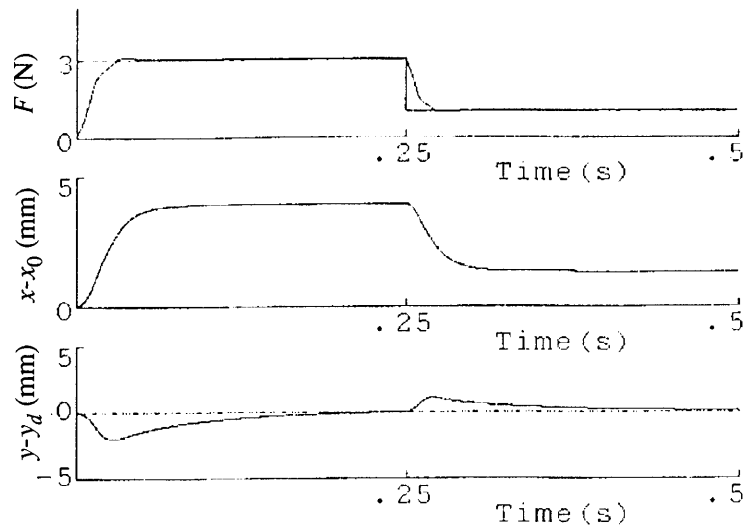


(a) soft object

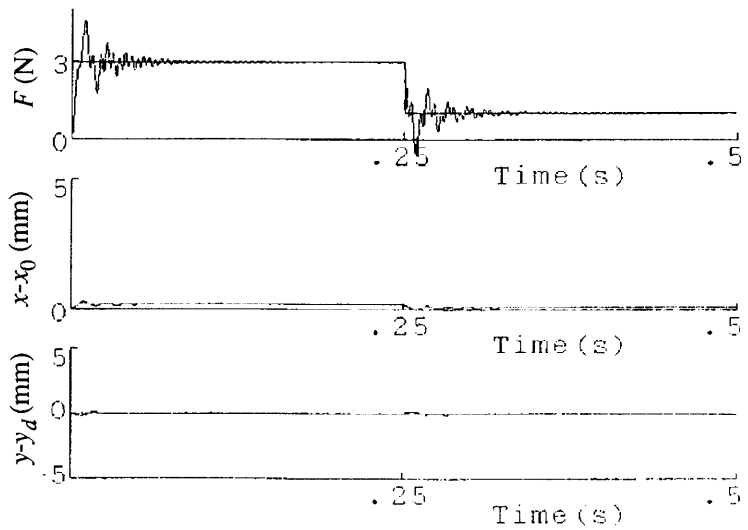


(b) hard object

図 4. 6 固定ゲイン方式による押し付け動作のシミュレーション結果
(対象物の動特性の変化の影響)

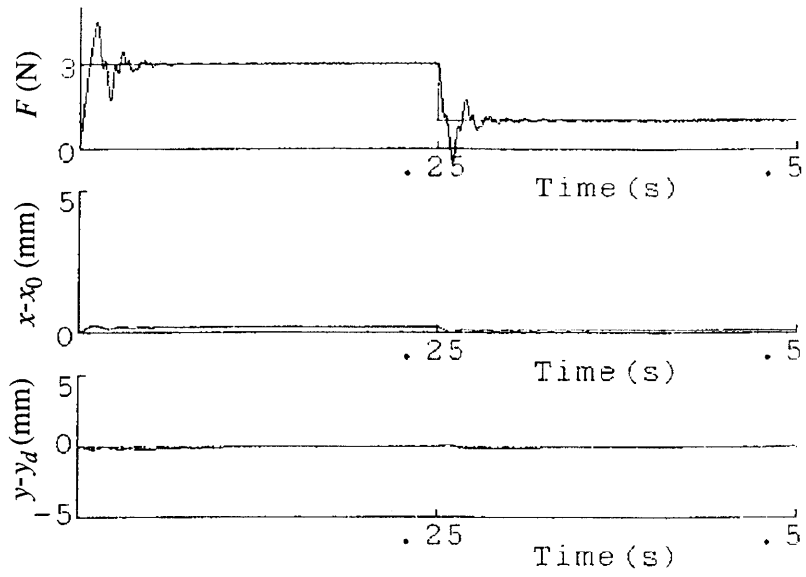


(a) soft object

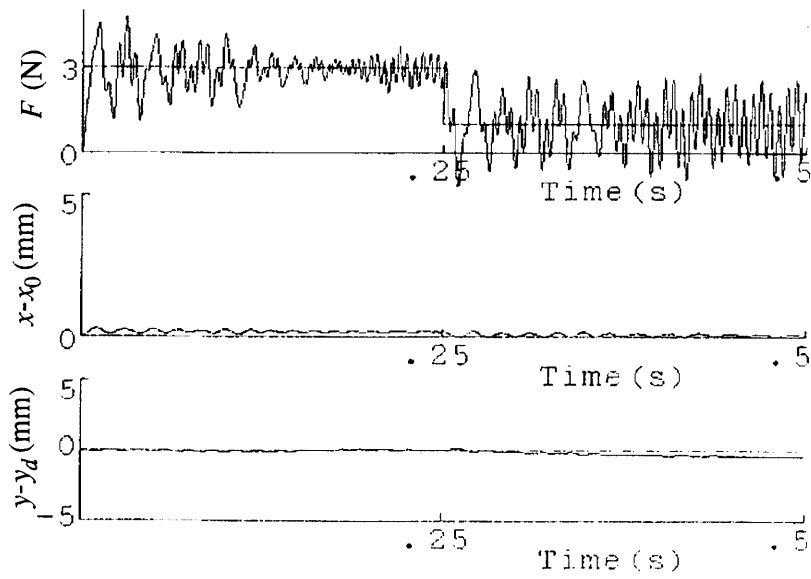


(b) hard object

図4. 7 可変ゲイン方式による押し付け動作のシミュレーション結果
(対象物の動特性の変化の影響)

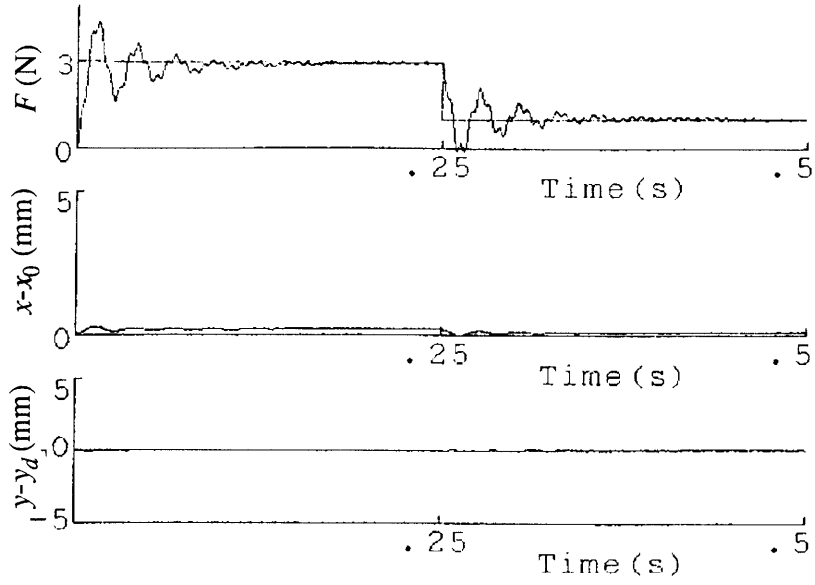


(a) $q_1 = 45^\circ$ $q_2 = 150 \text{ mm}$

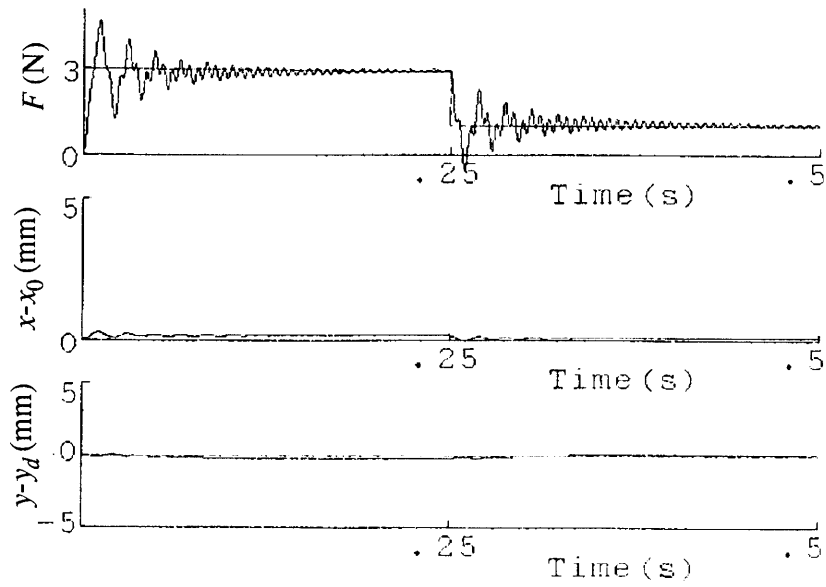


(b) $q_1 = 45^\circ$ $q_2 = 300 \text{ mm}$

図4. 8 固定ゲイン方式による押し付け動作のシミュレーション結果
(マニピュレータの姿勢変化の影響)

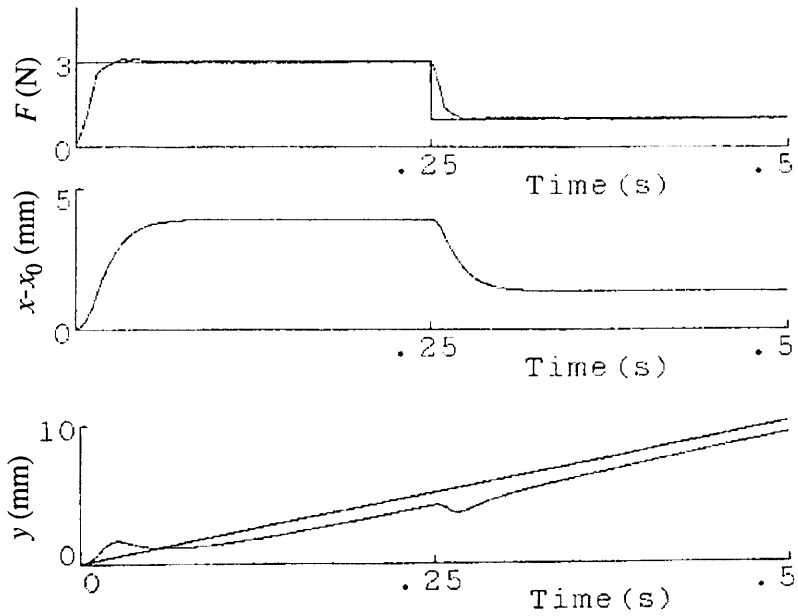


(a) $q_1 = 45^\circ$ $q_2 = 150$ mm

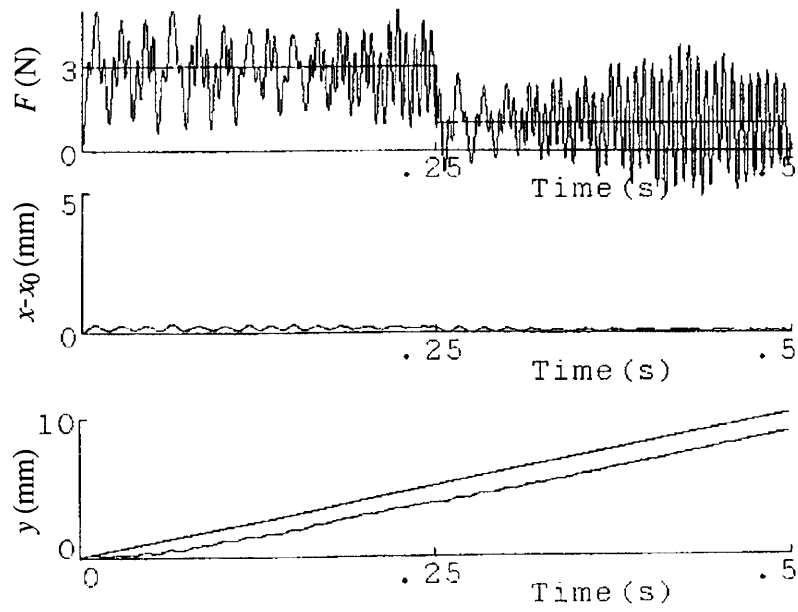


(b) $q_1 = 45^\circ$ $q_2 = 300$ mm

図4. 9 可変ゲイン方式による押し付け動作のシミュレーション結果
(マニピュレータの姿勢変化の影響)

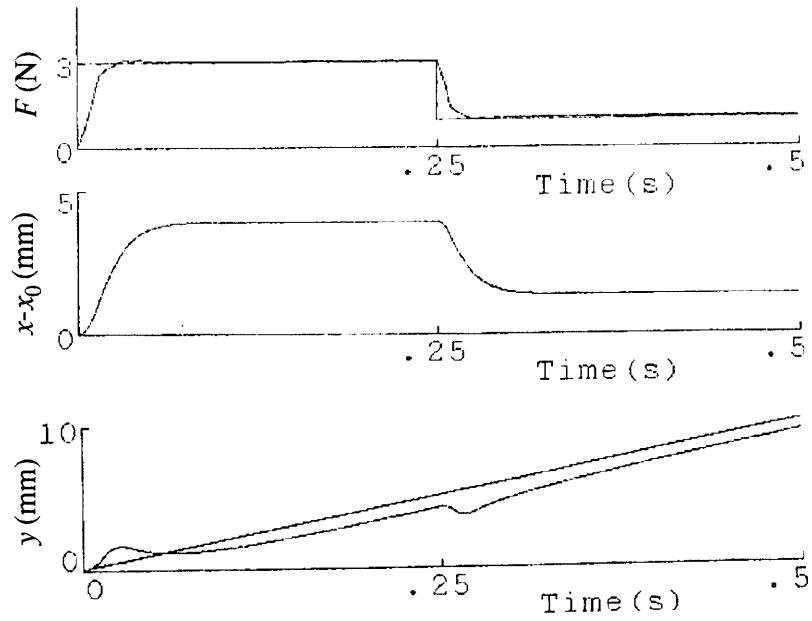


(a) soft object

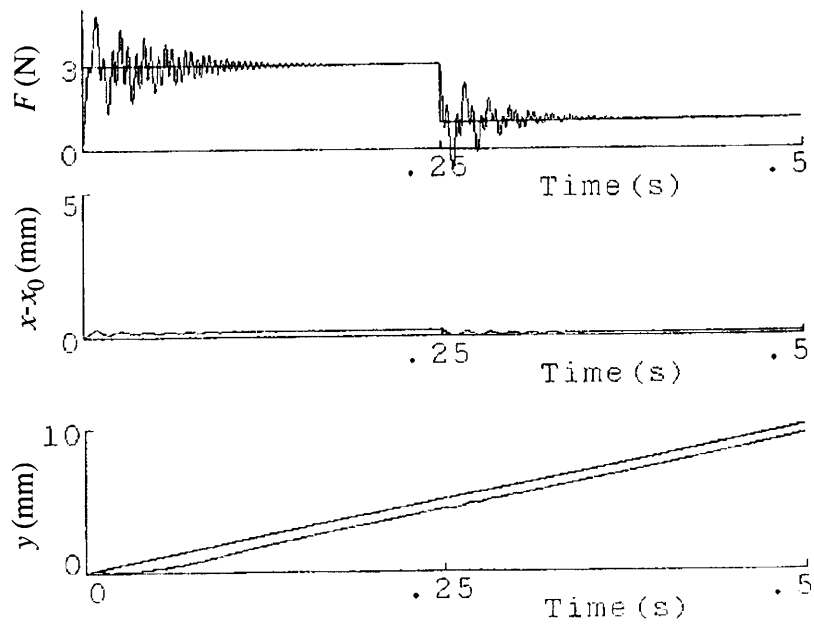


(b) hard object

図4. 10 固定ゲイン方式による倣い制御のシミュレーション結果



(a) soft object



(b) hard object

図4. 11 可変ゲイン方式による倣い制御のシミュレーション結果

表4. 1 シミュレーションに用いたマニピュレータのパラメータ

M_2	3	kg
M_3	0.1	kg
C_1	0.11	Nms
C_2	22	Ns/m
l	0.05	m
K_w	3×10^4	N/m
C_w	3.75	Ns/m
I_1	$0.0324 + 2 \times (q_2 - 0.18)^2$	kgm ²
K_r (soft)	8×10^2	N/m
C_r (soft)	11.1	Ns/m
K_r (hard)	5×10^4	N/m
C_r (hard)	10.5	Ns/m

表4. 2 対象物のばね定数にの变化に対する出力自乗誤差

K_r $\times 10^3$ (N/m)	Fixed Gains		Adaptive Gains	
	Force	Position	Force	Position
0.8	1.000	1.000	1.011	0.805
15	0.621	0.043	0.517	0.013
30	0.474	0.016	0.435	0.005
50	5.015	3.573	0.566	0.005
60	49.916	14.989	1.267	0.068

4. 4 実験方法と結果

4. 4. 1 実験装置と実験方法

図4. 12に本研究で用いた実験装置の概要を図4. 13に概観を示す。アクチュエータとしてDCモータを使い、関節1は、モータ軸にハーモニックドライブを減速装置として取り付けた回転機構とし、関節2はモータ軸に送りねじを付けた回転-並進変換機構とし、送りねじの先端で対象物を押し付ける構造となっている。位置の出力はモータ軸に付けたエンコーダで、力はマニピュレータ先端に取り付けた力センサ（歪ゲージ）で検出する。ニューラルネットワークおよび制御則は、パーソナルコンピュータ（CPU 8086, FPU 8087）のソフトウェアにて実現したため、制御のサンプリング時間は20 ms となった。

取り扱う対象物の大きさは未知となるため、まず認識動作を行う。認識動作は、速度制御を行いながら一定速度で対象物に近付け、力制御方向における力センサの出力が設定値（0.2N）を越えたところで対象物との接触を認識するものとする。その位置から、固定フィードバックゲイン方式と、ニューラルネットワークを用いた可変フィードバックゲイン方式による制御を行った。なお、ニューラルネットワークの学習点は、シミュレーションと同様に図4. 5に示す動作範囲内の六ヶ所にて行った。学習方法は、あらかじめ学習すべきフィードバックゲインにて制御動作を行い、その時の各関節の角度、変位、ワーク座標系における力制御方向のエンドエフェクタの位置、対象物との接触力の各データをサンプリングしておき、オフラインにて学習を行った。

4. 4. 2 押し付け動作の実験結果

図4. 12の実験装置を使用して、対象物の動特性が変化した場合の押し付け動作の安定性について評価した結果を図4. 14と図4. 15に示す。図はすべて、認識動作を行った後の出力を示している。対象物は、柔らかい対象物として、軟式のテニスボール（ $\phi 64\text{mm}$,ばね定数 $2 \times 10^2 \text{ N/m}$ ）、硬い対象物として鉄板（厚さ 20 mm ばね定数 ∞ ）を用いた。図4. 14は、固定フィードバックゲインを用いて押し付け動作を行った結果である。柔らかい対象物で図の様な応答をする

フィードバックゲインで、硬い対象物を押し付けると発振現象を起し制御性能が悪化してしまう。このように、対象物の動特性の変化に対し、適応範囲が限られてしまうことがシミュレーションと同様、実験でも明かとなった。本章で提案するニューラルネットワークによる可変フィードバックゲイン方式を用いて押し付け動作を行った結果が図4.15である。柔らかい対象物でも、硬い対象物でも安定した制御動作が実現でき、対象物の変化に対して、適応性が広がることが明かとなった。

マニピュレータの姿勢変化に対する押し付け動作の安定性について評価してみた結果を図4.16と図4.17に示す。対象物には、鉄板を用いた。図4.16は固定フィードバックゲイン方式による結果を示す。シミュレーションと同様に関節1の角度を変えず、関節2を縮めた場合と伸ばした場合について示してある。マニピュレータの姿勢の変化によって応答が不安定になっていることがわかる。他方、図4.17は、ニューラルネットワークによる可変ゲイン方式を用いた同様の結果を示す。固定フィードバックゲインの場合に比較して、マニピュレータの姿勢の変化に対して動作が安定していることがわかる。動作範囲内での姿勢変化に対する応答を自乗誤差にて比較した結果を図4.18に示す。図は、関節2を伸ばした場合と縮めた場合における、関節1の拘束面に対する角度を変えた姿勢での力制御の誤差について示してある。学習した六ヶ所以外の姿勢においても、ニューラルネットワークの補間機能により安定した制御が可能であることがわかる。

4.4.2 倣い制御の実験結果

図4.19と図4.20に、対象物の動特性が変化した場合における倣い制御の実験結果を示す。図4.19は、固定フィードバックゲイン方式の結果を示す。対象物の動特性の変化に対して不安定になることがわかる。図4.20は、本章で提案する可変フィードバックゲイン方式の結果を示す。この図から倣い制御においても、対象物の動特性の変化に対して、ニューラルネットワークによる可変フィードバックゲイン方式が有効であることがわかる。

4. 5 まとめ

本章では、ニューラルネットワークを使用したマニピュレータの位置と力の適応ハイブリッド制御系の構成方法及びその有効性について示した。

そして、シミュレーション及び実験により、本方式が固定フィードバックゲイン制御方式と比較して、

- 1、対象物の動特性の変化
- 2、マニピュレータの姿勢の変化

に対して、安定で、適応範囲が広がり、柔軟に対応できることを示した。

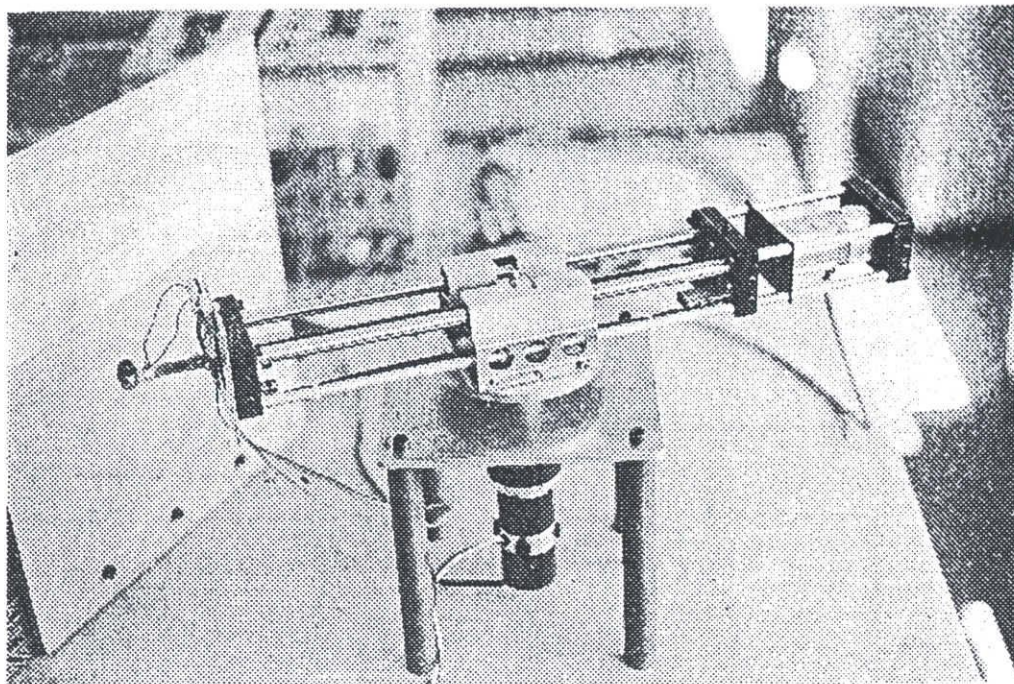
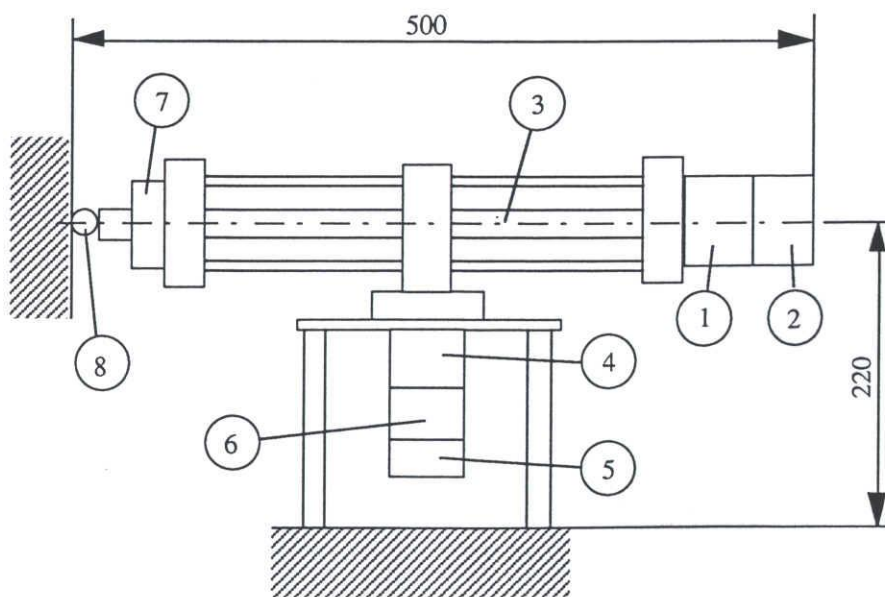


図 4. 1 2 実験装置



- | | |
|-------------|-----------------|
| 1. DC Motor | 5. Encoder |
| 2. Encoder | 6. DC Motor |
| 3. Screw | 7. Force Sensor |
| 4. Gear | 8. Ball Bearing |

図 4. 1 3 実験装置の概観

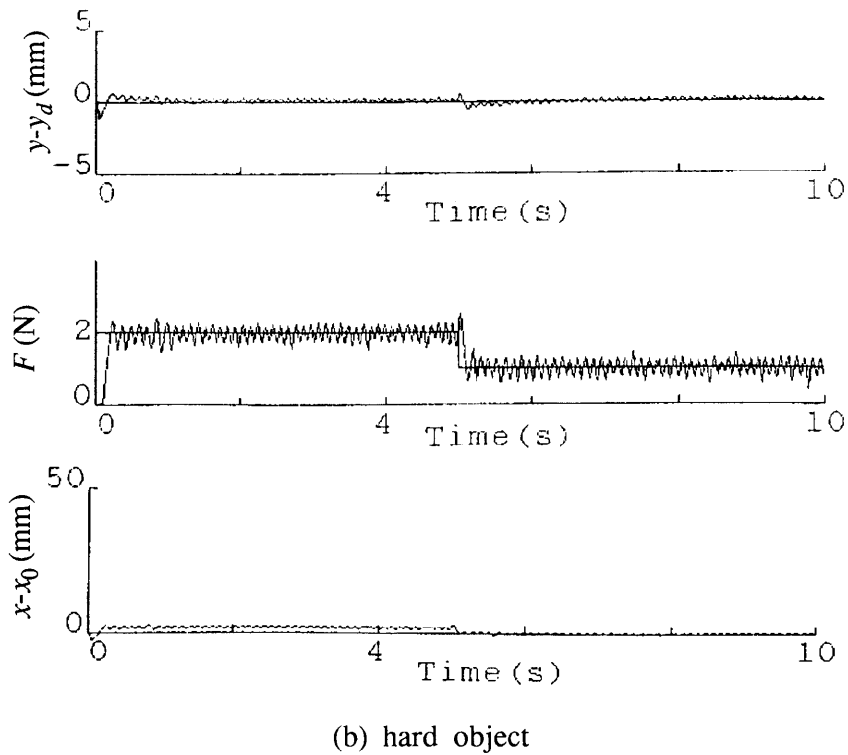
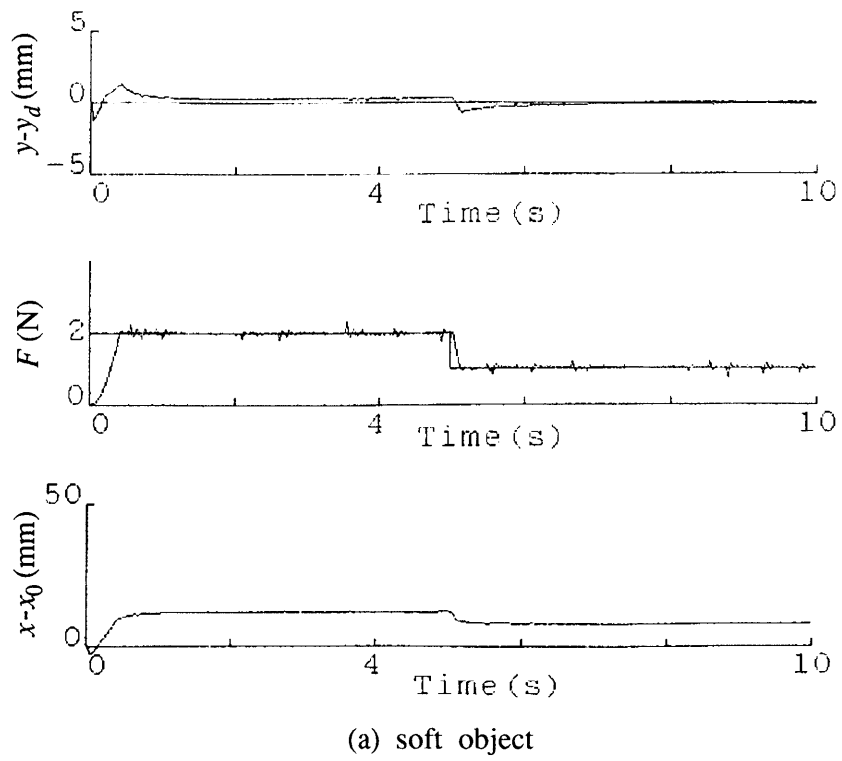
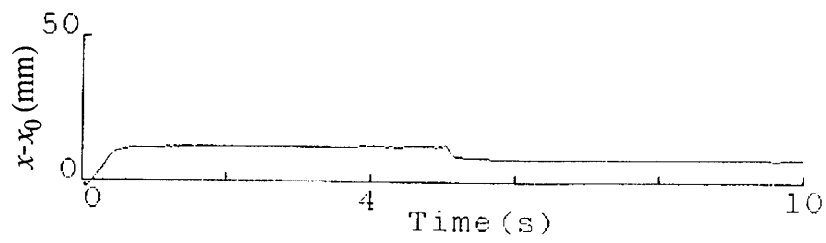
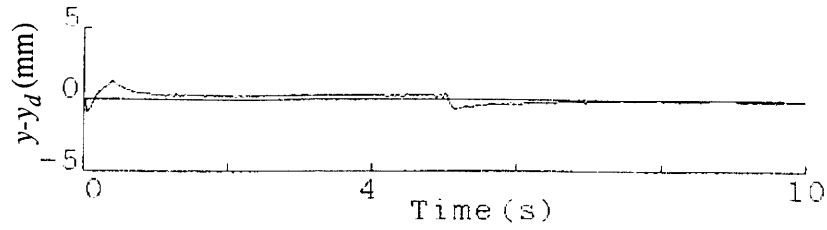
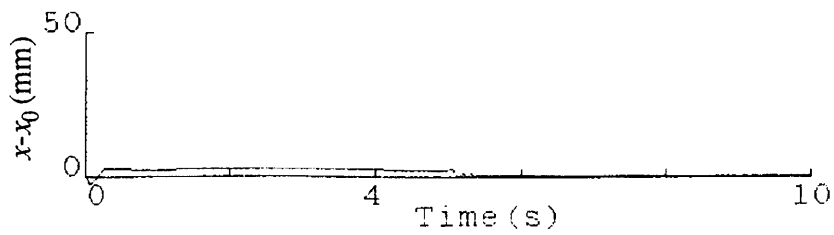
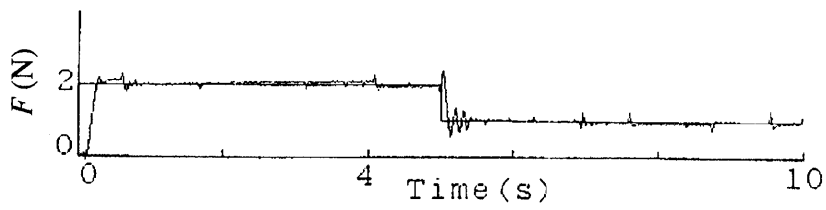
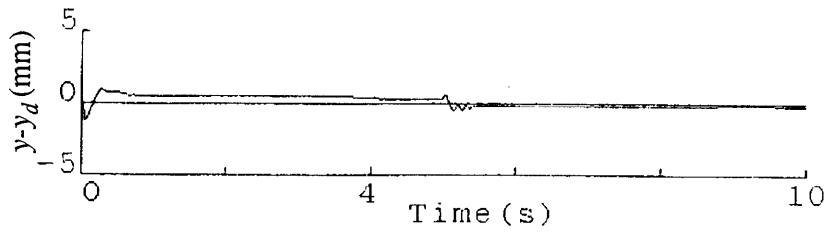


図4. 14 固定ゲイン方式による押し付け動作の実験結果
(対象物の動特性の変化の影響)

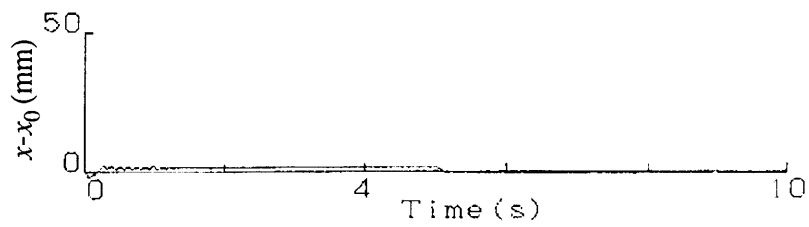
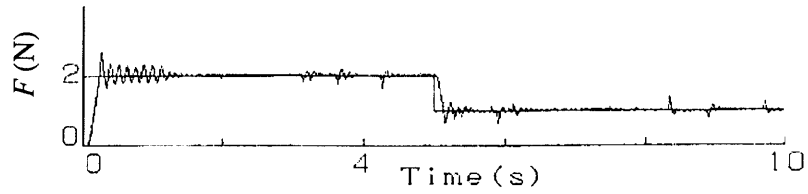
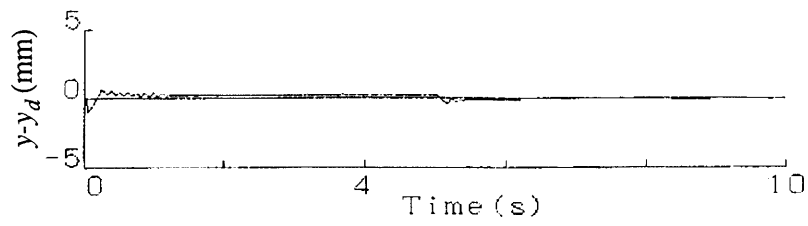


(a) soft object

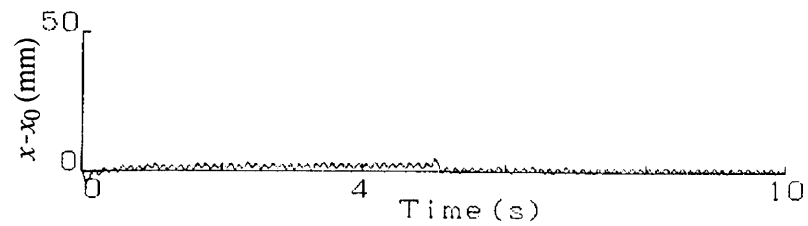
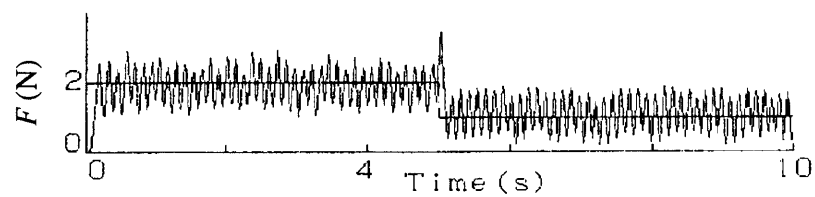
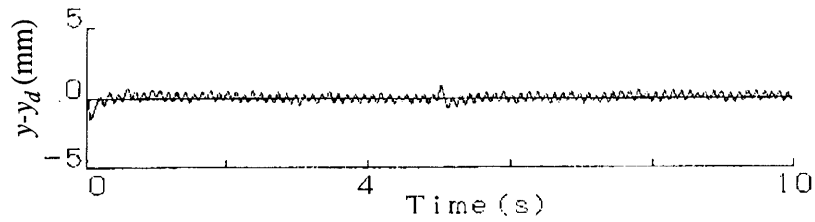


(b) hard object

図4. 15 可変ゲイン方式による押し付け動作の実験結果
(対象物の動特性の変化の影響)

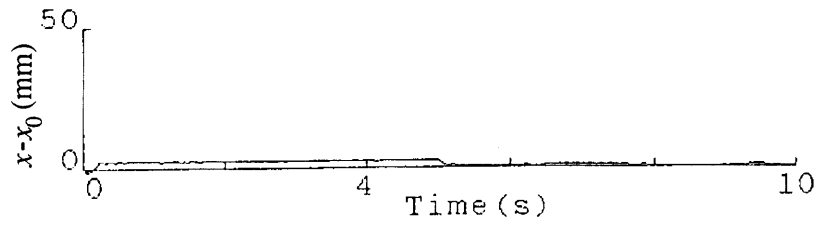
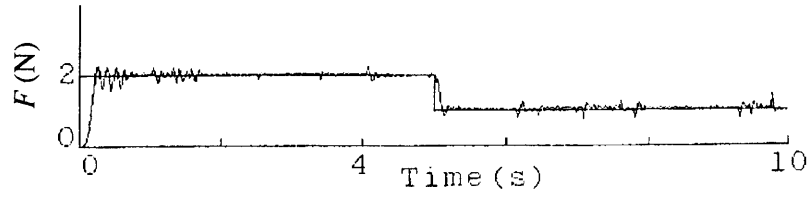
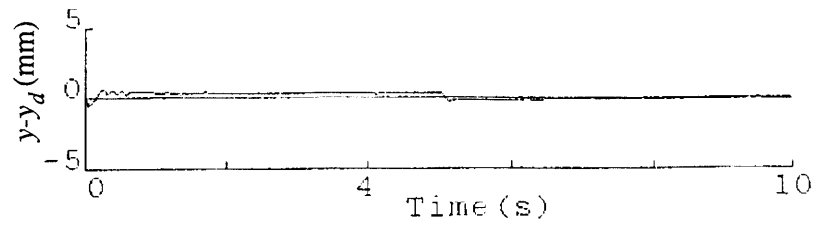


(a) $q_1 = 40^\circ$ $q_2 = 150$ mm

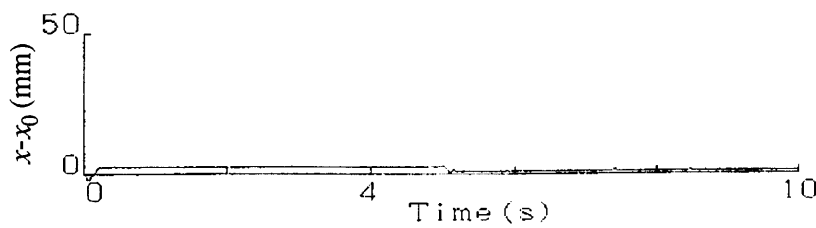
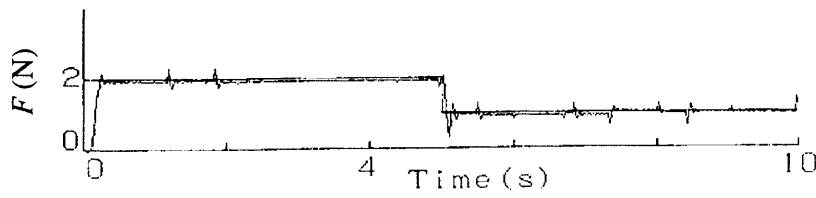
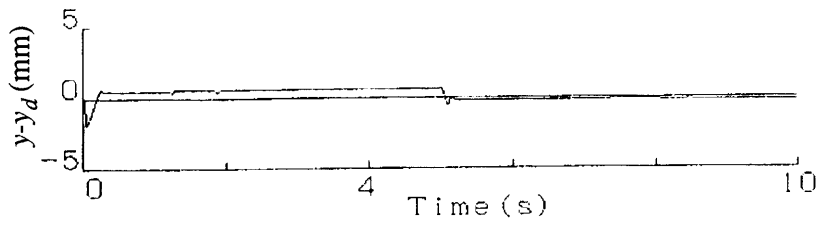


(b) $q_1 = 40^\circ$ $q_2 = 300$ mm

図4. 16 固定ゲイン方式による押し付け動作の実験結果
(マニピュレータの姿勢変化の影響)



(a) $q_1 = 40^\circ$ $q_2 = 150$ mm



(b) $q_1 = 40^\circ$ $q_2 = 300$ mm

図4. 17 可変ゲイン方式による押し付け動作の実験結果
(マニピュレータの姿勢変化の影響)

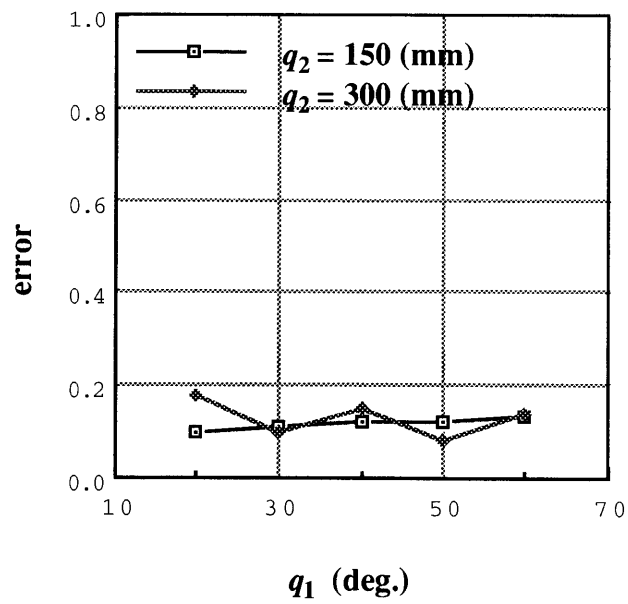
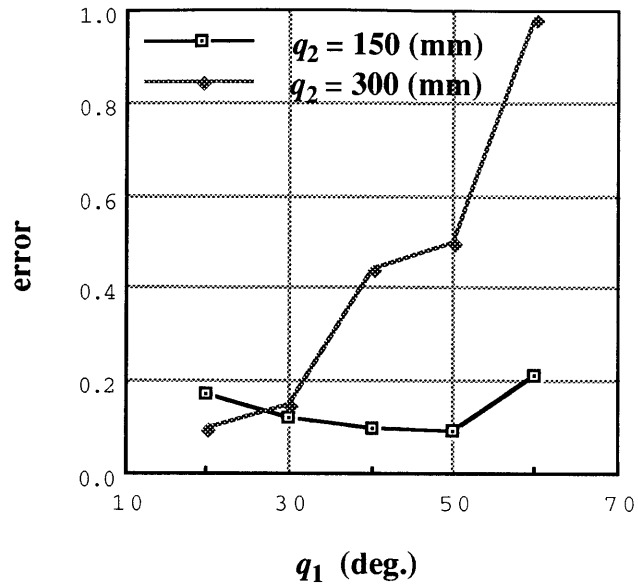


図4. 18 押し付け動作の出力自乗誤差
(マニピュレータの姿勢変化の影響)

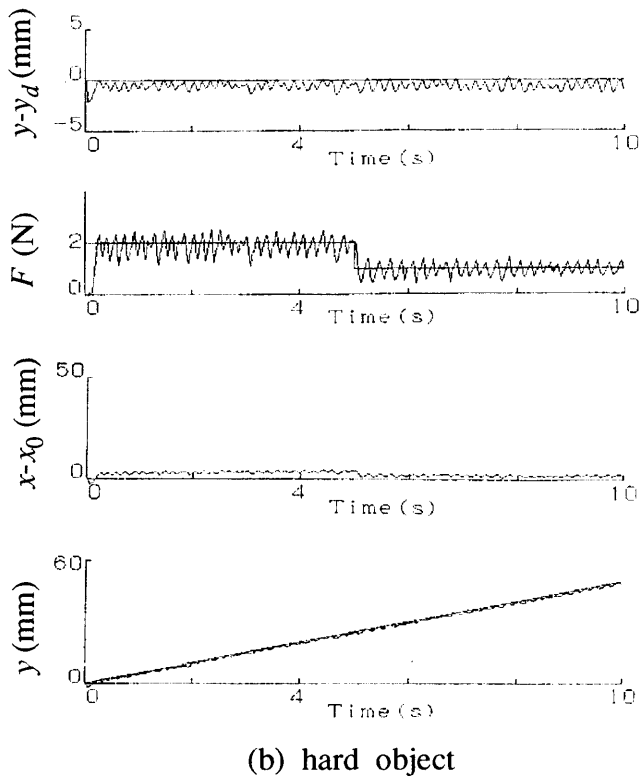
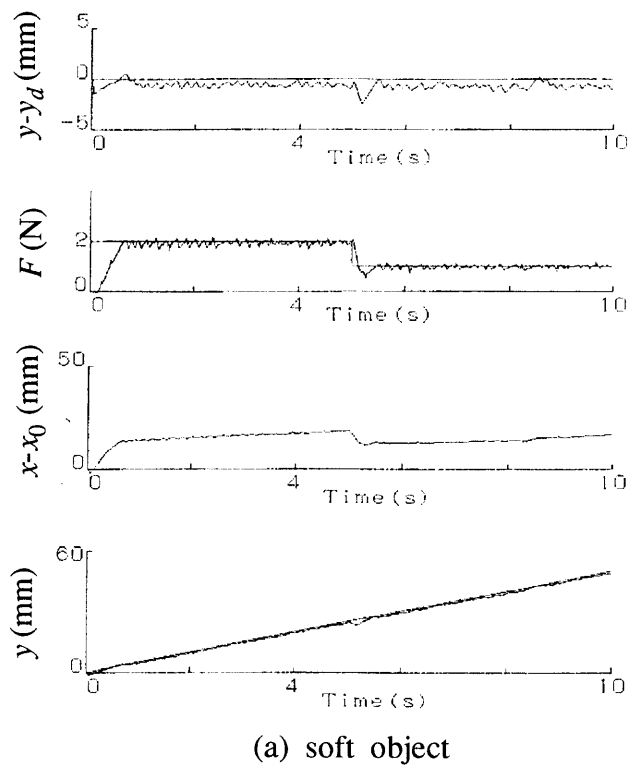
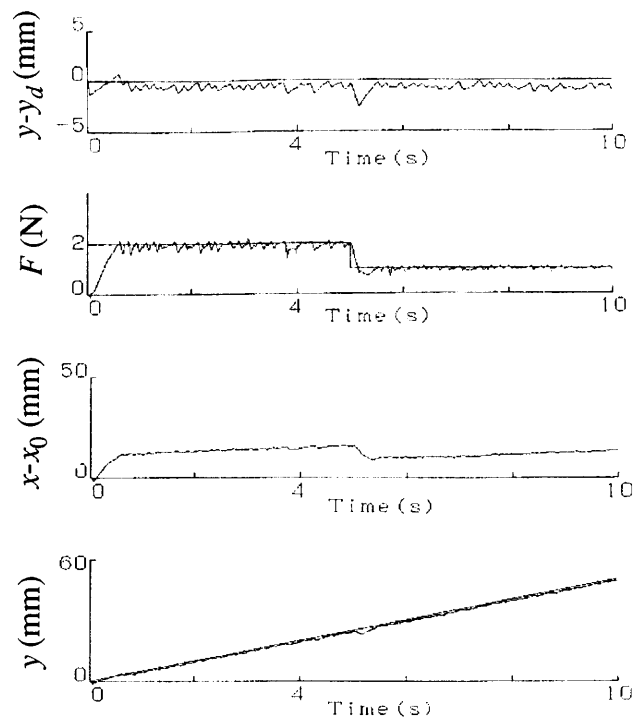
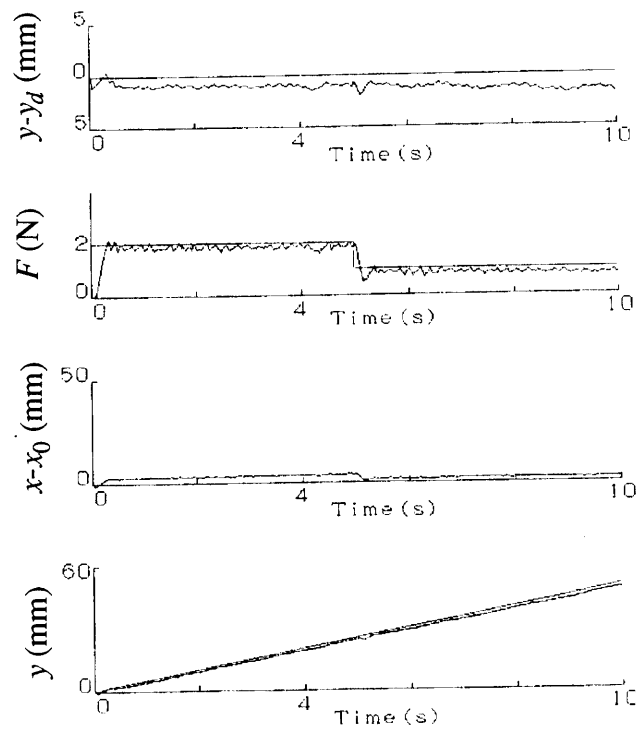


図4. 19 固定ゲイン方式による倣い制御の実験結果



(a) soft object



(b) hard object

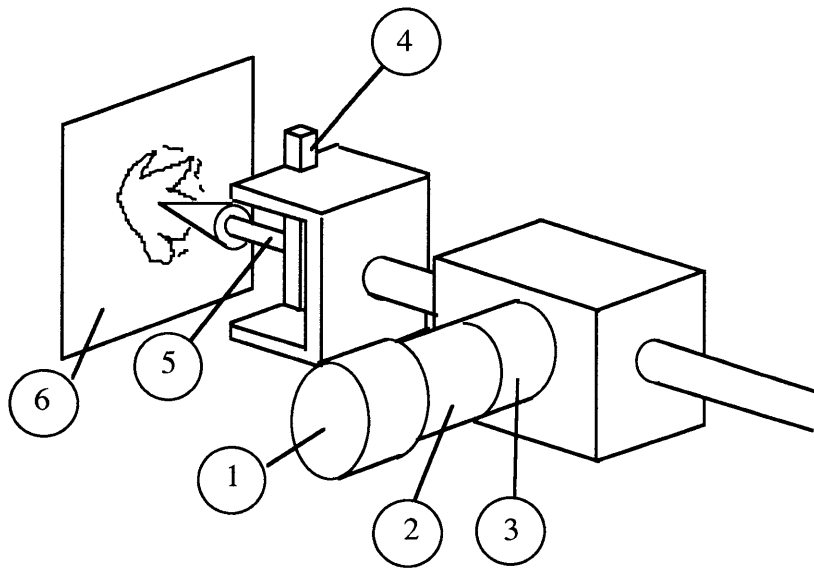
図4. 20 可変ゲイン方式による倣い制御の実験結果

第5章 ニューラルネットワークによる スタビング制御

5.1 スタビング制御

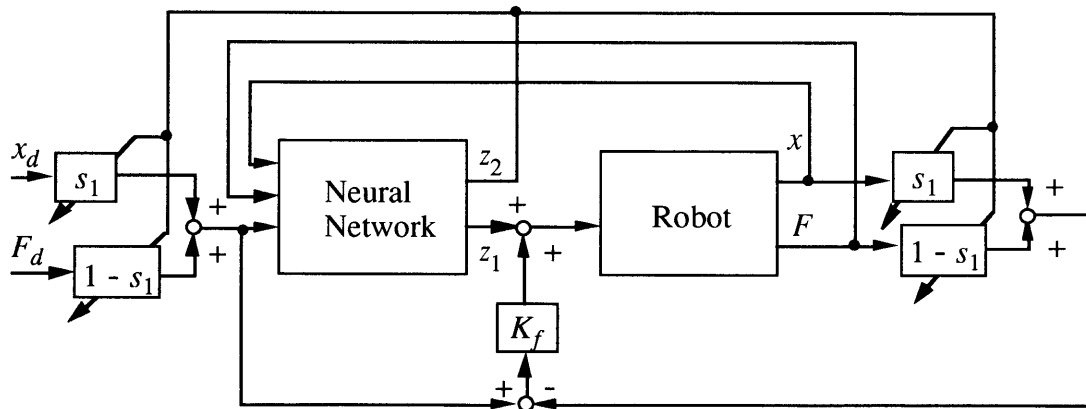
前章ではニューラルネットワークを用いた力制御系について述べたが、さらに複雑な状況における力制御について考察を進める必要がある。マイクロマニピュレータで生物の細胞内の組織を破壊せず細胞膜を破るといったケースや、ドリルによる穴開けといったケースでは対象物の破壊前後において、またその材質によって動特性が異なってくる。このような条件下で破断から反動抑制への一連の動作をスムーズに行うには、位置のみの制御では困難であり、力情報と位置情報を組合せ統合的に処理するようなハイブリッド制御方式が必要になってくる[16][17]。また、対象物が未知の場合には、固定されたシステムでは対応できる範囲が限定される。

本章では図5.1に示す1自由度のスタビングコントロールを取り上げた。適当な材質によりできている薄膜を、針状になったロボットマニピュレータの先端部により破断させ静止するといったコントロールについて考察を進める。このような非線形特性を対象とした研究は従来の制御理論の枠組の中では、あまり取り扱われてこなかった分野である[22]。ニューラルネットワークの学習能力、非線形マッピングを用いて物体により異なる非線形特性を有する対象物の制御を順応的に行うのが本章の目的である。特に、破断時の力情報と位置情報の統合、すなわちセンサーフュージョンによる動作の認識と、ロボットコントローラとしての機能を持つニューラルネットワークモデルを構築しスタビングコントロールに適用する。このコントローラはロボットの厳密なモデルを必要としないこと、教示によりセンサーフュージョンに基づく認識系が自己組織化される点で優れていると考えられる。システムの安定性については、一般論は不可能に近いので、ニューラルサーボコントローラでは、一般にフィージブルな解があればよい。すなわち、本章のニューラルサーボコントローラでは、必ずしも最適性等を求めるのではなく、モデリングが困難な対象に対して、従来のコントローラでは困難なフィージブルな解を求める立場をとることにする。



- | | |
|--------------------|-----------------|
| 1. Encoder | 4. Force Sensor |
| 2. Tacho Generator | 5. Arm |
| 3. Motor | 6. Object |

図 5. 1 スタビリングコントロールの動作



- | | |
|--------------------------|--------------------------|
| s_1 : Hybrid Ratio | K_f : Fixed Gain |
| z_1 : Output of Cell 1 | z_2 : Output of Cell 2 |

図 5. 2 ニューラルサーボコントローラ

5. 2 ニューラルネットワークを用いたスタビリングコントローラ

図5. 2にスタビリングコントロールに用いたニューラルサーボコントロールシステムを示す。ニューラルネットワークモデルの構造は、入力ユニット3個、中間ユニット7個、出力ユニット2個となっている。入力データは、ハイブリッド目標値、力出力値、位置出力値とした。ネットワークからの出力は、一つはモータへのトルク入力となり、もう一つの出力はハイブリッド比 s_1 を表す。 s_1 を出力することにより、ニューラルネットワークで接触と破断の認識を行う。すなわち、対象物に接触している間は $s_1=0.0$ とし、対象物破断後は $s_1=1.0$ を出力するようにする。

5. 3 シミュレーション

5. 3. 1 制御対象の数学モデル

対象物との相互作用を考え、先端質量を考慮に入れると、その数学モデルは図5. 3のように表される。ここで対象物は線形な粘弾性モデルでモデル化されている。対象物が破断した後は、ばね定数と粘性係数を零とする。このモデルに基づき、動特性を導くと以下のようなになる。

$$\text{増幅器： } V(t) = A_m u(t) \quad (5.1)$$

$$\text{DCモータ： } L_a di(t)/dt + R_a + E(t) = V(t) \quad (5.2)$$

$$\text{逆起電力： } E(t) = K_e \dot{\theta}_m(t) \quad (5.3)$$

$$\text{発生トルク： } T(t) = K_t i(t) \quad (5.4)$$

$$\text{モータ側の釣合： } J_m \ddot{\theta}_m(t) + B_m \dot{\theta}_m(t) = T(t) - T_l \quad (5.5)$$

負荷側の釣合：

質量Mについて

$$M\ddot{x}(t) + B_x \dot{x}(t) + C_x \{\dot{x}(t) - \dot{X}(t)\} + K_x \{x(t) - X(t)\} = nT_l / r \quad (5.6)$$

質量Mpについて

$$M_p \ddot{X}(t) + C^* \dot{X}(t) + K^* X(t) - C_x \{\dot{x}(t) - \dot{X}(t)\} - K_x \{x(t) - X(t)\} = 0 \quad (5.7)$$

モータも回転角とアームの移動距離の関係：

$$\theta_m(t) = nx(t)/r \quad (5.8)$$

ここで、

- A_m : 増幅器のゲイン B_m : モータの摩擦係数
 K_t : モータのトルク定数 K_e : モータの逆起電力定数
 J_m : モータの慣性モーメント
 M : 負荷質量 M_p : 先端質量
 B_l : 負荷の摩擦係数 R_a : モータの電機子抵抗
 K^* : 対象物のばね定数 C^* : 対象物の粘性係数
 K_x : 力センサとマニピュレータのばね定数
 C_x : 力センサとマニピュレータの粘性係数
 L_a : モータのインダクタンス n : 減速比 r : ピニオン半径
 T_l : 負荷によるトルク $T(t)$: 発生トルク
 $u(t)$: 入力電圧 $V(t)$: モータ端子電圧
 $E(t)$: 逆起電力 $i(t)$: モータ電流
 $\theta_m(t)$: モータ回転角 $x(t)$: アームの位置
 $X(t)$: 力センサのたわみ

モータのインダクタンス L_a 、およびモータの摩擦抵抗 B_m を無視して、各式を整理すると運動方程式は次のようになる。

$$\ddot{x}(t) = -a_1 x(t) - a_2 \dot{x}(t) - a_3 w_x(t) - a_4 \dot{w}_x(t) + b_1 u(t) \quad (5.9)$$

$$\ddot{w}_x(t) = -a_5 x(t) - a_6 \dot{x}(t) - a_7 w_x(t) - a_8 \dot{w}_x(t) \quad (5.10)$$

$$\begin{aligned}
 a_0 &= n^2 J_m + r^2 M, & a_1 &= r^2 K_x / a_0, & a_2 &= \{r^2 R_a (B_l + C_x) + r^2 K_e K_t\} / (R_a a_0), \\
 a_3 &= -r^2 K_x / a_0, & a_4 &= -r^2 C_x / a_0, & a_5 &= -K_x / M_p, & a_6 &= -C_x / M_p, \\
 a_7 &= (K_x + K^*) / M_p, & a_8 &= (C_x + C^*) / M_p, & b_1 &= A_m n r K_t / (R_a a_0)
 \end{aligned}$$

式(5.9)(5.10)は

$$\mathbf{X} = [x(t), \dot{x}(t), X(t), \dot{X}(t)]^T$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -a_1 & -a_2 & -a_3 & -a_4 \\ 0 & 0 & 0 & 1 \\ -a_5 & -a_6 & -a_7 & -a_8 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ b_1 \\ 0 \\ 0 \end{bmatrix}$$

とおくと、次のようになる。

$$\dot{X}(t) = AX(t) + Bu(t) \quad (5.11)$$

出力方程式は、位置と力の比を表す係数 s_1 を用いて次式で表される。

$$y(t) = s_1 x(t) + (1 - s_1)F(t), \quad (0 \leq s_1 \leq 1) \quad (5.12)$$

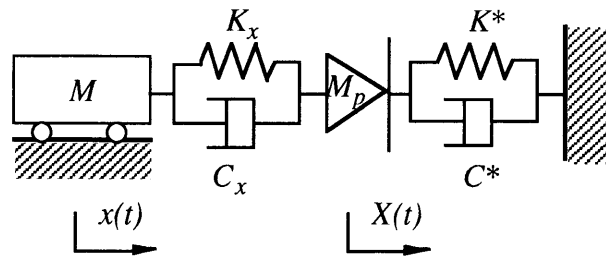
力 $F(t)$ は、

$$F(t) = C_x \{\dot{x}(t) - \dot{X}(t)\} - K_x \{x(t) - X(t)\} \quad (5.13)$$

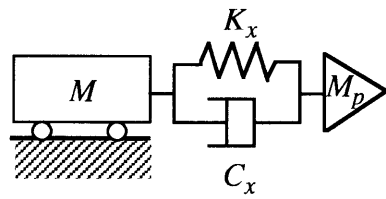
5. 3. 2 シミュレーション結果

図5. 4、図5. 5にスタビングコントロールのシミュレーション結果を示す。対象物のばね定数が $K^*=50$ N/mと $K^*=10$ N/mの二とおりの場合についてシミュレーションを行った。図5. 4(a)(b)(c)はニューラルネットワークを使わなかった例で、図5. 5(a)(b)はニューラルネットワークを使った例である。力出力が出始めたところで対象物に接触し、急激に落ち込んだところで対象物が破断したことを示している。出力例のCell 1、Cell 2はニューラルネットワークの出力を示しCell 2の出力はハイブリッド比 s_1 を表している。それぞれの場合について同じ条件でシミュレーションを行った。ニューラルネットワークを使わない場合は、あらかじめ破断条件として力出力の最大変化量 Δe を設定し、それによって破断の認識を行う。しかし対象物が未知の場合に破断の認識を行えない場合もある。図5. 4(c)の例では力出力が零にまで落ち込んでいないことから対象物が破断していないことがわかる。これに対して、ニューラルネットワーク自身で破断の認識ができるようになっている。

破断までの力の評価については $K^*=50$ N/mの時に固定フィードバックゲインのみによる制御のほうが良好な結果が得られたが、破断の認識と結び付いた反動抑制についてはニューラルネットワークを用いたほうが良好な結果が得られた。

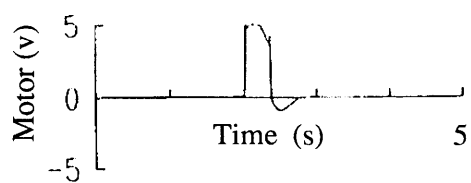
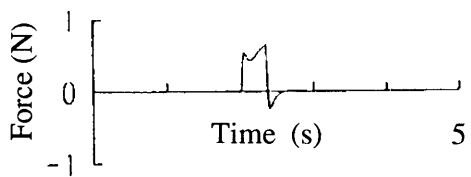
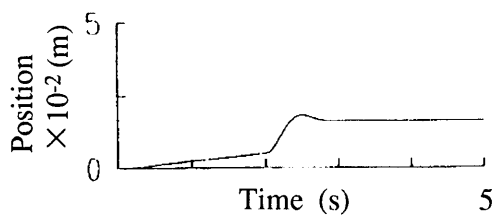


Before Break

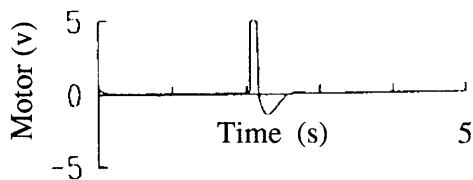
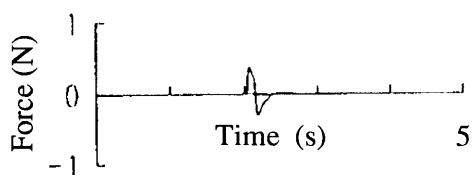
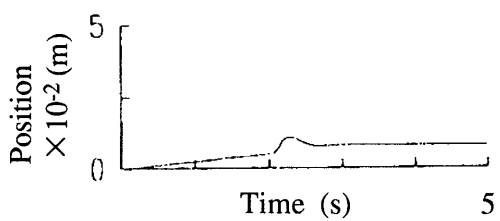


After Break

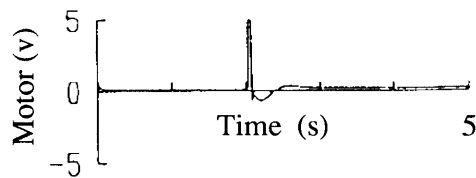
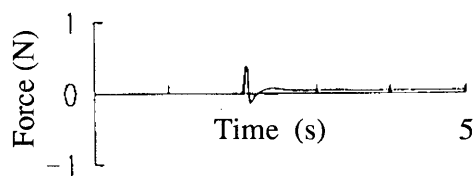
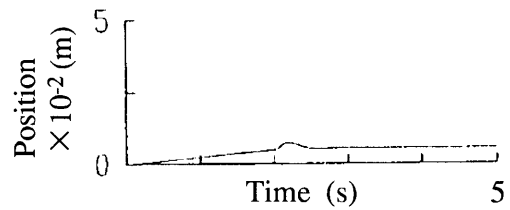
図 5. 3 マニピュレータの数学モデル



(a) $K^* = 50$ (N/m)

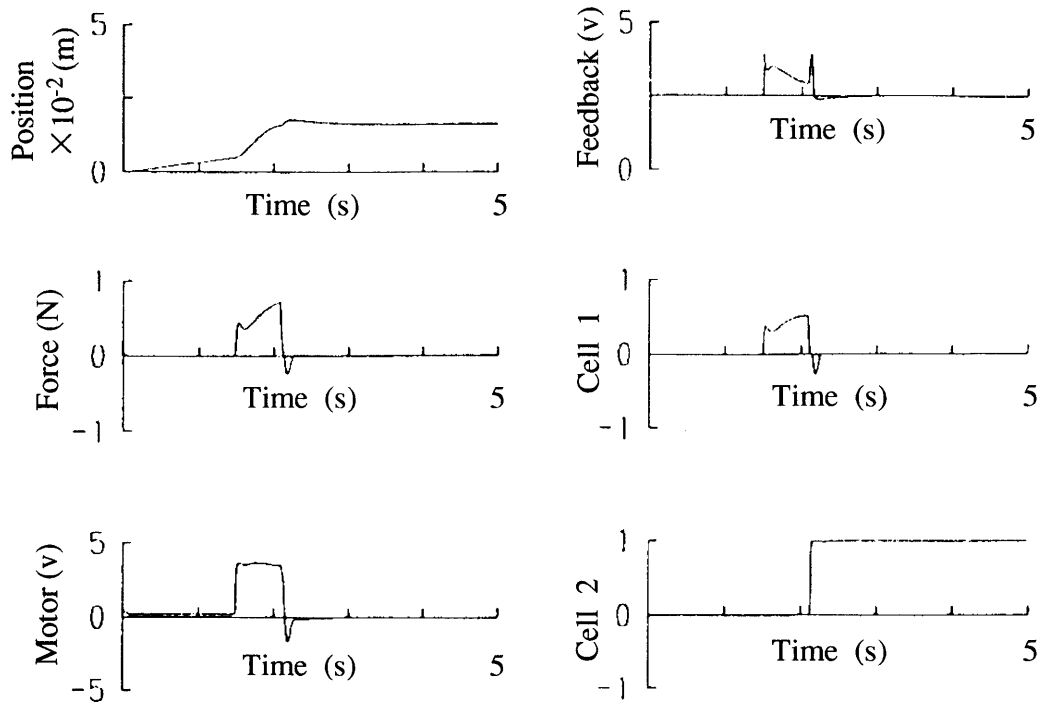


(b) $K^* = 10$ (N/m)

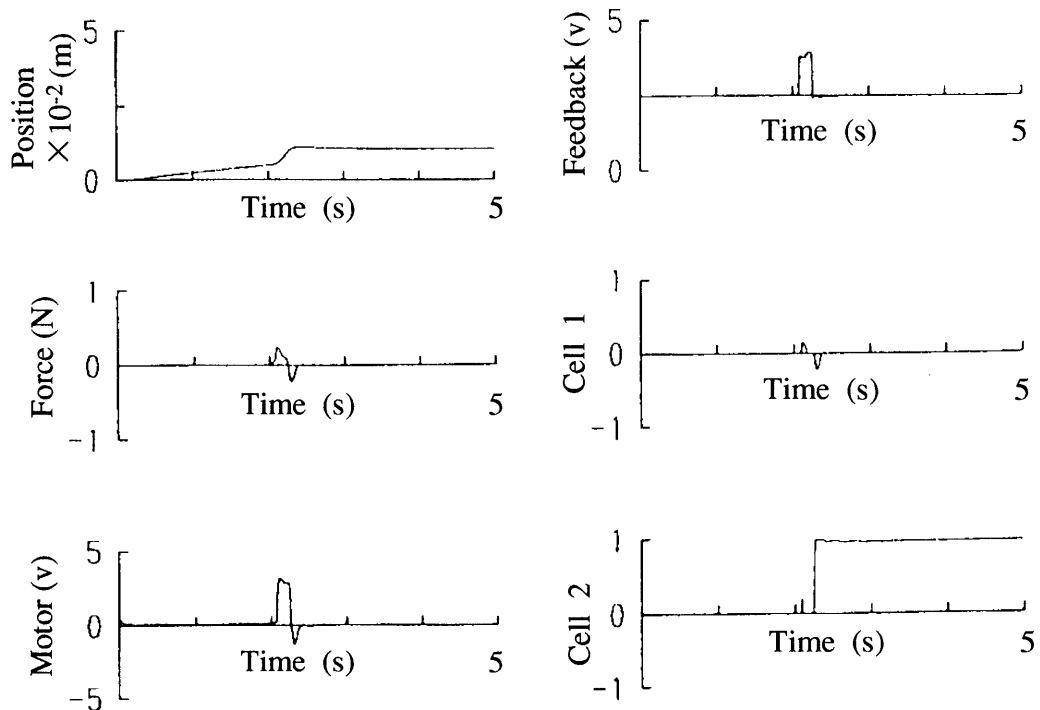


(c) $K^* = 10$ (N/m)

図5. 4 固定ゲインによるシミュレーション結果



(a) $K^* = 50$ (N/m)



(b) $K^* = 10$ (N/m)

図5. 5 ニューラルネットワークを用いたシミュレーション結果

5. 4 実験方法と結果

5. 4. 1 実験装置と実験方法

図5. 6に実験装置を示す。アクチュエータにはDCモータを使い、ラックとピニオン形式で対象物を押し付ける構造となっている。先端には力センサ（ひずみゲージ）を取り付け、位置出力はエンコーダで測定できる。最大移動距離は約150 mmとなっている。未知対象物としては、塩化ビニールの薄膜と、風船を用いた。

スタート位置から対象物までは適当な距離をおく。対象物に接触するまでは一定速度で接近し、力出力が e_0 を越えたところで接触を認識し、スタビリングコントロールに移る。

実験では、シミュレーションに用いたニューラルネットワークをそのまま転用した。

5. 4. 2 実験結果

図5. 7、図5. 8に実験結果を示す。図5. 7(a)(b)はニューラルネットワークを使わなかった例で、図5. 8(a)(b)はニューラルネットワークを使った例である。シミュレーション結果と同じく Cell 2 は、ハイブリッド比 s_1 を表している。破線は停止目標位置（破断認識位置）を示す。図5. 7(b)は固定フィードバックゲインのみによる制御結果だが、シミュレーションからも予測されたように、対象物が破断しないうちにロボットアームが停止している（力出力が零になっていない）。ニューラルネットワークを用いた制御の場合は、対象物が変わっての安定な制御が行え、破断の認識もうまくできている。

5. 5 まとめ

本章では、ニューラルネットワークの学習能力、非線形マッピングを利用した、ロボットマニピュレータのニューラルサーボコントローラを提案し、このコントローラの教示によりロボットに必要な位置と力の制御能力をもたせることができた。

このニューラルサーボコントローラを用いた制御では異なる対象物に対して比較的安定制御ができる点と、位置情報と力情報の融合により、破断の認識ができるという点で有効であることをシミュレーションと実験により示した。

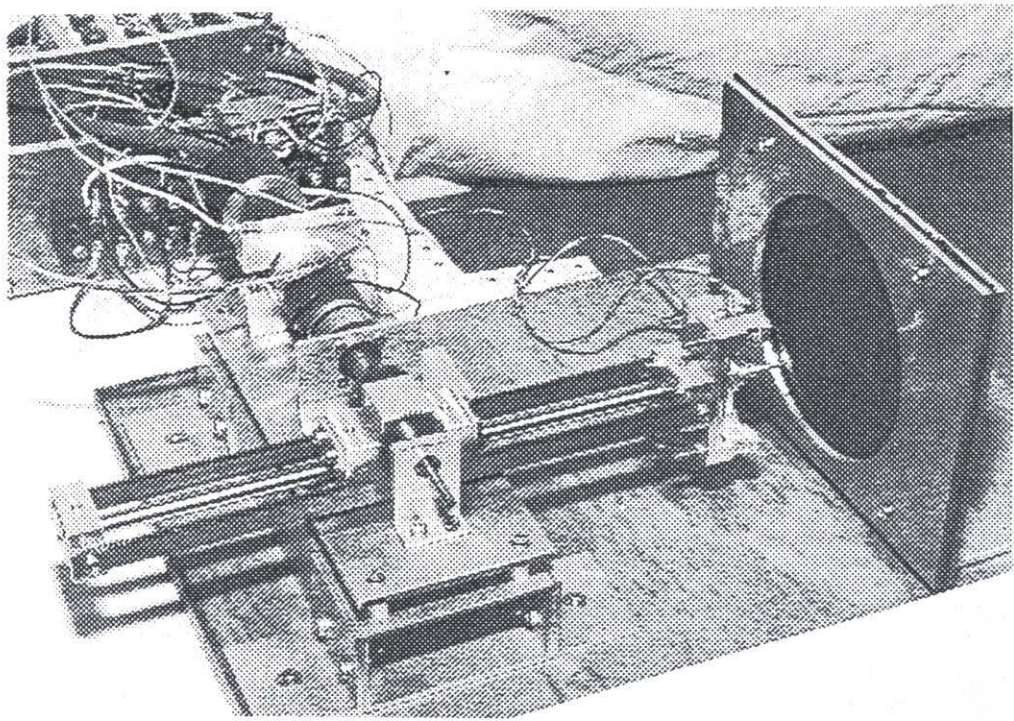
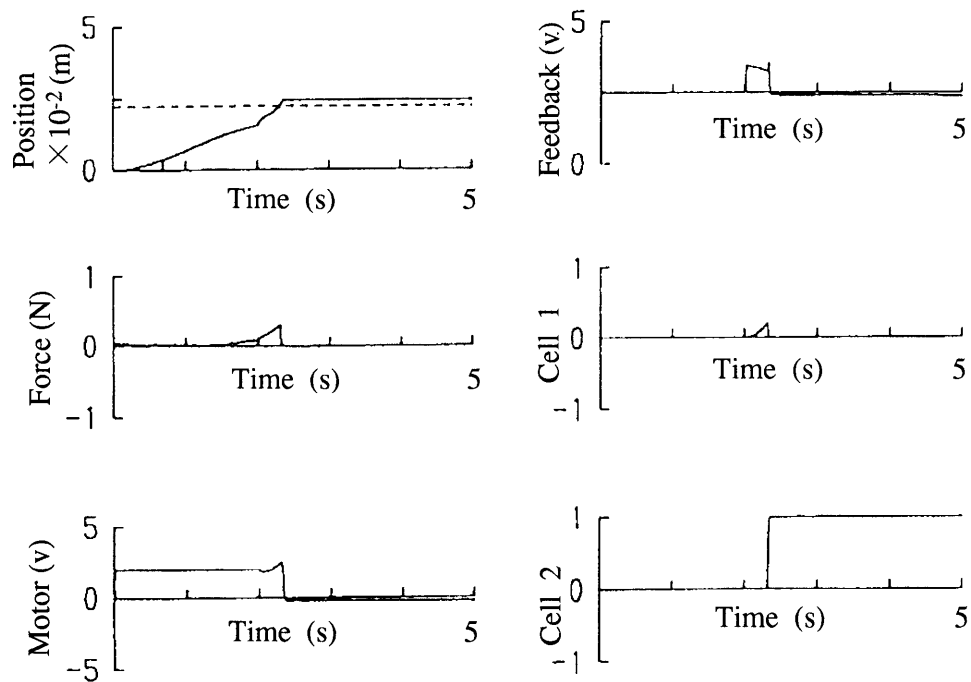
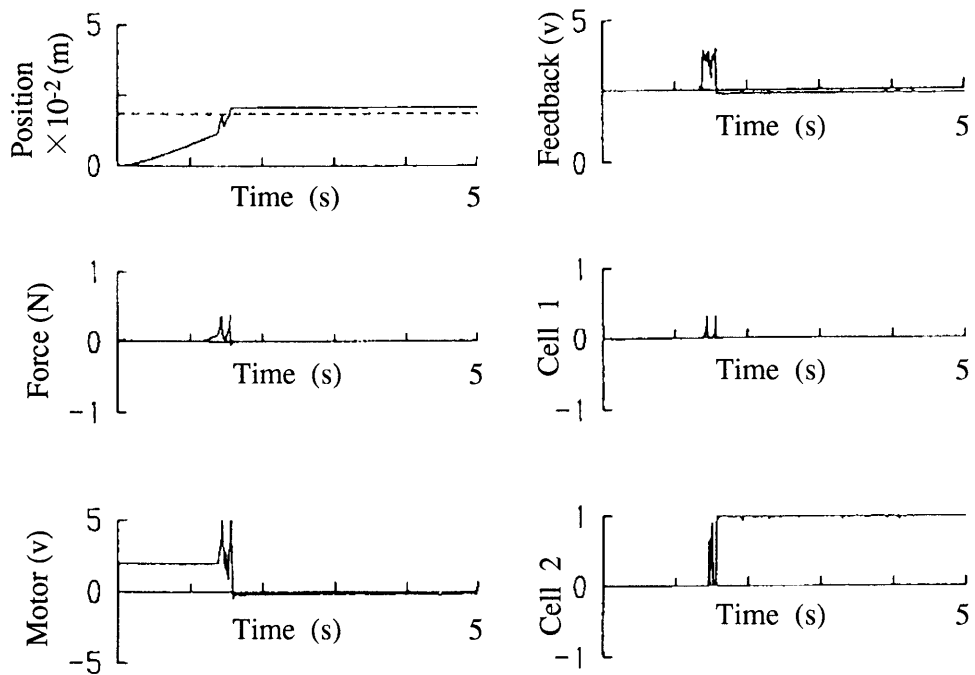


图 5. 6 实验装置



(a) 風船



(b) 塩化ビニール膜

図5. 8 ニューラルネットワークを用いた実験結果

第6章 追加学習型ニューラルネットワーク による適応力制御

本章では、新しい知識を追加したり間違っ
て記憶した知識を修正する追加学習を考
慮した、制御システムについて述べる。

6.1 シグモイド関数型ニューラルネットワークの問題点

一般的に、ニューラルサーボコントローラ (Neural Servo Controller: NSC) には、シグモイド関数を用いた多層型ニューラルネットワーク (Neural Network with Sigmoid function : NNS) を使用しており、学習にはバックプロパゲーションアルゴリズムを用いているため、新しい知識を追加したり間違っ
て記憶した知識を修正する追加学習ができない。特に、環境 (対象物) まで含めた系として扱わなければならない力制御やハイブリッド制御では、動特性の大きく異なる未学習の対象物が新たに制御対象として加わった場合、今まで学習した全ての対象物を含めて再度学習し直す必要があるため非常に効率が悪い。NNSでは、新たなパターンを追加学習するためには、今まで学習したパターンをすべて用意しなければならないし、学習も最初からやり直さなければならない、追加のための学習に多くの手間と時間を必要とする。

一方、福田、塩谷らはパターン認識において、追加学習可能なニューラルネットワーク (Neural Network Based on Distance between Patterns : NDP) を提案している[23]。NDPは Radial Basis Functionに類似した応答関数を使用し、部分的にニューロン間の結合荷重を修正したり、新たにニューロンを追加することにより、追加学習を可能にしている。追加学習は時間の短縮という点で大きな利点となる。また、NDPは未学習であることを判断できる利点があり、これが新たな知識の獲得を可能としている。それまで学習したパターンと全く異なった未学習パターンに対して、NNSは未学習であることを判断できず誤認識をする可能性がある。一方 NDPは未学習の判断が可能であり、追加学習を行う判断を簡単なものとする。

そこで本章では、追加学習が可能で、環境の特性変化に対して適応的にコント

ローラを自己生成し、適応範囲をより大きく広げることが可能な制御システムについて提案する。システム構成はNDPとNSCを組み合わせた階層構造をとる。制御系の過渡期における出力時系列データは一種のパターンとみなすことができ、動特性の大きく異なる環境に対しては、そのパターンも大きく異なるためNDPで環境特性変化を認識できる。認識後、変化に応じたNSCを選択し、制御を切り替えることにより、環境特性変化にロバストな制御を行うことができる。また、未学習の新たな環境特性に対しては、適応的にコントローラを自己生成することにより、更に適応範囲を広げることが可能となる。

NNを用いて、パターン認識と制御系を階層的に組み合わせた適応制御については、いろいろ研究がなされている[24][25]が、それらは追加学習ができなかったり、未知プラントに適用できない問題がある。また、環境の変化に対して適応的にコントローラを自己生成するといった考えはない。本章で提案するシステムは、追加学習が可能であり、未知プラントにも適応可能である。パターン認識後、それが未学習であると判断されるとシステムは、新たにニューロン、NSCの追加を試み、コントローラを自己生成する。また、新たな環境の特性は未知であるため、中間層のユニット数などNSCの構成をも含めた生成を実施する。一つのNSCの汎化能力だけではその適応範囲が限られているため、このように新たな環境に対してNSCを追加していくことで適応範囲をより大きく広げることが可能となる。

本章では、1自由度マニピュレータを対象とし、未知対象物に対する力制御の適応性の拡大を目的として制御システムを構成し、シミュレーションを行い、提案したシステムの可能性を検討する

6. 2 追加学習システムの構成

6. 2. 1 制御対象とシステムの構成

本章では、3章にも述べた未知対象物に対する力制御を取り扱う。対象物との相互作用を伴う力制御システムでは、システム全体の閉ループ内に対象物の動特性が含まれるため、対象物の特性まで含めて制御系を設計する必要がある。従って、対象物の特性が未知の場合には制御系の設計が困難となる。3章において、NNを使

用した制御システムを提案し、その有効性を示したが、前述のように追加学習ができない。また、一つのNNではその汎化能力に限界があり、学習したものと動特性が大きく異なる対象物に対しては、応答が不安定となり制御性能が悪化する。本章では、追加学習が可能であり、動特性が大きく異なる対象物に対して適応的にコントローラを自己生成することが可能なシステムについて述べる。

提案するシステムの構成を図6. 1に示す。システムは、パターン認識を行うNDPと非線形制御を行うNSCおよび追加学習判断機構を階層的に組み合わせた構成となっている。本システムの動作は、2段階に分けられる。まず制御の始めにおいて、対象物を一定力（アクチュエータ駆動力）で押し付け、図6. 2に示すようにマニピュレータの位置 x と力センサ出力 f を N サンプリング数とする。そのパターンベクトル U からNDPにて対象物の特性認識を行う。

$$U^T = [x(0), x(1), \dots, x(k), \dots, x(N), \\ f(0), f(1), \dots, f(k), \dots, f(N)] \quad (6.1)$$

認識後、それに応じたNSCを選択し、制御を切り替える。図6. 1に示すように、NDPの認識後どのNSCを使うかは、データベースに保持されている。また、そのデータベースにはNSCの適応範囲の指標として出力自乗誤差の許容値 E_a も保持されている。この E_a は切り替え後のNSCの妥当性を確認するためのもので、切り替え後 r サンプリング数の累積自乗誤差である。なお、切り替え直後の応答はステップ応答を想定している。従って切り替え直後の目標値が多数ある場合は、目標値ごとに E_a を保持しておく。

もし、対象物が未学習であると判断されたり、認識できたにもかかわらず制御がうまく行かなかった（出力自乗誤差が許容値 E_a を越えた）場合、システムは追加学習を行う。

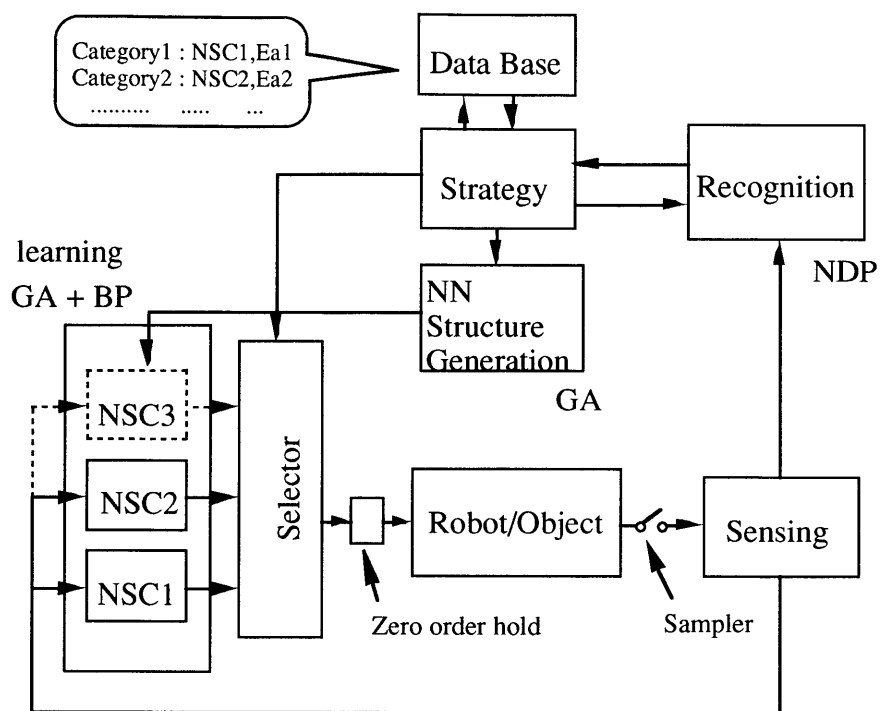


図 6. 1 制御システムの構成

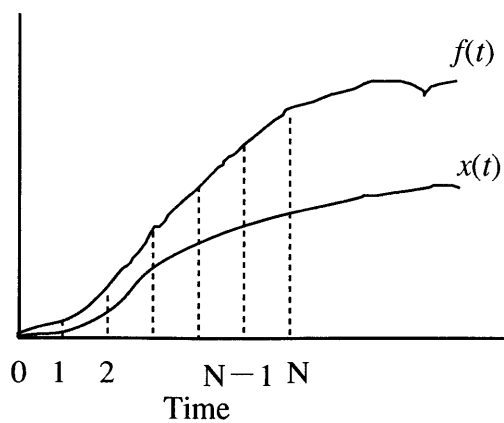


図 6. 2 ステップ応答波形

6. 2. 2 追加学習型ニューラルネットワークの構造と学習則

追加学習型ニューラルネットワーク (NDP) は図 6. 3 に示すように 2 層構造になっている。入力層には、時系列入力パターンの圧縮、観測雑音の除去を目的として、入出力関係が式 (6. 2) で表される時間的加重 (短期記憶) を行うニューロンを用いた。このニューロンにより、時系列入力パターンの伸縮もある程度吸収することができる。

$$y_i(k) = \sum_{j=1}^{\gamma} v(j)u(k-j) \quad (6.2)$$

$$v(j) = \frac{\gamma - j + 1}{\gamma} \quad (6.3)$$

ここで、

$y(k)$: 出力 $u(k)$: 入力 ($x(k)$ 及び $f(k)$) $v(j)$: 加重係数

NDP の入力層ユニットへは、各サンプリング時刻ごとに $x(k)$ 、 $f(k)$ を入力していく。入力された $x(k)$ 、 $f(k)$ はそれぞれ図 6. 3 の白丸で示されるニューロンにより時間的加重が行われる。時間的加重ニューロンは、次の連なる時間的加重ニューロンに信号を伝搬するのに 1 時刻必要とする。

出力層には、入出力関係が式 (6. 4) で表されるニューロンを用いる。この応答関数を使用することによりパターン空間を円形状に分離し、部分的にニューロン間の結合荷重を修正したり、新たにニューロンを追加することにより、追加学習を可能にしている。

$$Y_j = \frac{1}{1 + \alpha_j \sqrt{\sum_i (y_i - W_{ji})^2}} \quad (6.4)$$

ここで、

Y_j : 出力層ニューロンの出力値

y_i : 入力値 (時間的加重ニューロンからの出力値)

W_{ji} : 入力層と出力層の結合係数

α_j : 汎化係数

また、学習則は以下のようなになる。

$$\alpha_j(n+1) = \alpha_j(n) - \beta_1 \frac{\partial E_j}{\partial \alpha_j} \quad (6.5)$$

$$W_{ji}(n+1) = W_{ji}(n) - \beta_2 \frac{\partial E_j}{\partial W_{ji}} \quad (6.6)$$

$$E_j = \left(T_j - \frac{1}{1 + \alpha_j \sqrt{\sum_i (y_i - W_{ji})^2}} \right)^2 \quad (6.7)$$

$$\begin{aligned} \frac{\partial E_j}{\partial \alpha_j} &= 2 \left(T_j - \frac{1}{1 + \alpha_j \sqrt{\sum_i (y_i - W_{ji})^2}} \right) \\ &\quad \times \frac{\sqrt{\sum_i (y_i - W_{ji})^2}}{\left[1 + \alpha_j \sqrt{\sum_i (y_i - W_{ji})^2} \right]^2} \end{aligned} \quad (6.8)$$

$$\begin{aligned} \frac{\partial E_j}{\partial W_{ji}} &= 2 \left(T_j - \frac{1}{1 + \alpha_j \sqrt{\sum_i (y_i - W_{ji})^2}} \right) \\ &\quad \times \frac{-\alpha_j (y_i - W_{ji})}{\left[1 + \alpha_j \sqrt{\sum_i (y_i - W_{ji})^2} \right]^2 \sqrt{\sum_i (y_i - W_{ji})^2}} \end{aligned} \quad (6.9)$$

ここで

E_j : 誤差

β_1 : 汎化係数 α_j の学習係数

β_2 : 結合係数 W_{ji} の学習係数

n : 学習回数

T_j : 抑制方向の場合 0

興奮方向の場合 1

図6. 4はNDPを使ってパターンを分類した時の概念図である。各 Category はNDPにより図のように円形に分類される。円の中心はNDPの結合係数 W_{ji} で決定され、円の大きさは、汎化係数 α_j で決まる。

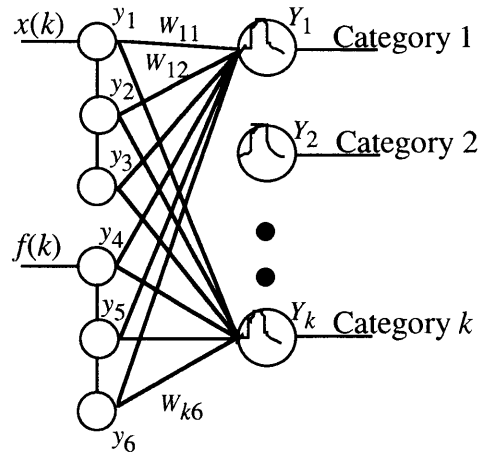


図6. 3 NDPの構造

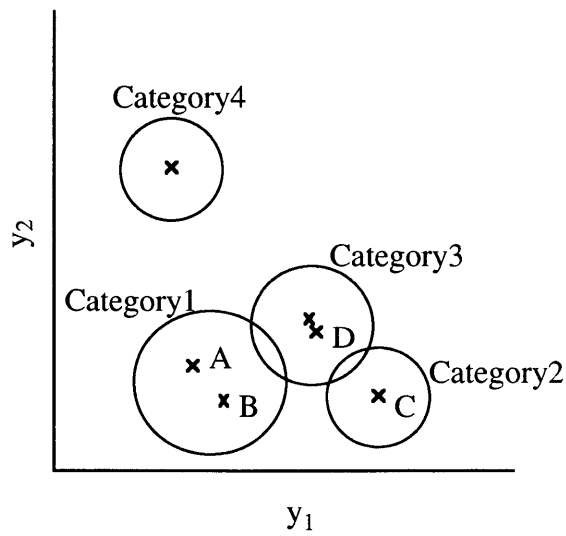


図6. 4 NDPによる分類の概念図

6. 2. 3 追加学習アルゴリズム

NDPの出力ニューロンが、しきい値 R 以上を出力した状態を「発火」と定義する。

まず、NDPでパターン認識をする。その時、最大発火値を示すカテゴリーを Category1 (NSC1, Ea1)、2番目に大きな発火値を示すカテゴリーを Category2 (NSC2, Ea2) とする。

1) Category1に制御を切り替える。システムは、切り替え後、式(6.10)に示すように r サンプリング数の出力自乗誤差 $E1$ を測定する。 $E1 < Ea1$ であったら、学習は行わない。

2) もし $E1 > Ea1$ であったら、Category2に制御を切り替え、同様に出力自乗誤差 $E2$ を測定する。 $E2 < Ea2$ であったらCategory1を示したニューロンを抑制(汎化空間を縮小)する方向に、Category2を示したニューロンを興奮(汎化空間を拡大)する方向に、学習させる。NSCは学習を行わない。

3) もし $E2 > Ea2$ であったら、新たなカテゴリー(新たなNSCとNDPに新たなニューロン)を追加し、NDP、NSCの学習を行う。

4) 発火ニューロンが一つしかなく、 $E1 > Ea1$ の時、あるいは発火ニューロンが一つもなかった時、新たなカテゴリーの追加と学習を行う。

また、もし発火ニューロンが三つ以上あった場合、2)と同様に順次制御を切り替え、学習を行う。ただし、本システムの最初の段階は Category 1 (NSC1, Ea1)のみ学習する。そして新たな対象物が制御対象として加わった場合、NSC1で制御を行う。その結果もし $E1 < Ea1$ であったらNDPの拡大を行い、もし $E1 > Ea$ であったら、カテゴリーの追加を行う。

6. 2. 4. ニューラルサーボコントローラ

NSCは図6. 5に示す多層フィードフォワード型ネットワークを用いる。入力には、マニピュレータの位置 $x(k)$ 、 $x(k-1)$ 、力センサの出力値 $f(k)$ 、 $f(k-1)$ 、アクチュエータ駆動力 $f_a(k)$ 、 $f_a(k-1)$ 及び目標値 $f_d(k+1)$ を加え、出力はアクチュエータ駆動力 $f_a(k+1)$ とする。また、図6. 5の黒丸は各ユニットのしきい値を設定するための常に1を出力しているユニットを表す。

環境の特性は未知であるため、精度良く制御を行うには、中間層のユニット数などNSCの構成が重要となる。また、3層のNNで任意の関数を正確に近似することが可能であるが、実際には3層による近似は、多数の中間層ユニットを必要とする。4層以上を使ったほうがNNの規模（結合の数）を小さくできる場合がある[26]。従って、本システムではNSCの構造決定を含めて学習を行う。NSCの学習および最適構造探索にはGenetic Algorithm（以下GA）を使用した。GAは最適化手法の一つであり、NNやファジイシステムの学習[27][28]においてその有効性が示されており、多点探索を行うことができ、勾配情報を必要としない点に特徴がある。

まず、NNの構造に対しては図6. 6 (a)に示すように、層の数を入力層から順に遺伝子座に対応させ、各層のユニットの数を遺伝子としてコーディングする。図6. 6 (b)、(c)に示すように遺伝子が0は、その層がないものとする。次に、各染色体が示す構造に従ってNNを展開する。例えば図6. 6 (c)は、図6. 5に示すネットワークに展開される。展開したNNのユニット間の結合係数（荷重）を遺伝子として図6. 7に示すように順に並べコーディングする。ここで、結合係数評価の適応度を、

$$J_c^{-1} = E = \sum_{k=1}^p \{f_d(k) - f(k)\}^2 \quad (6.10)$$

とし、構造評価の適応度は、結合係数を少なくするのを目的として

$$J_s^{-1} = E + \beta N \quad (6.11)$$

とする。ただし、

E : 出力自誤差、 N : NNの結合総数、 β : 定数

f : カセンサ出力、 f_d : カ目標値、

ρ : サンプル数

二重にGAを使用する。まず、構造を示す個体間で一世代GAの操作（交差、突然変異）を行う。次に、その世代の個体を一つずつ展開し、更にGAにて学習（結合係数評価）を行う。この学習結果を基に、構造の評価を行い、構造を示す個体を淘汰させる。構造評価の適応度の式（6. 11）におけるEは、学習で最終的に得られた集団の中で最も適応度の高い個体のものを用いる。なお、試行回数を減らすため、同一構造のものは評価を繰り返して行わない。また、図6. 8に示す構造のものは意味がないので評価を行わず淘汰させる[26]。図6. 8は、ある中間層のユニット数とその前後の中間層のユニット数より少ない場合で、ユニットが少ない層での情報圧縮が行われてしまい、関数の近似においては、意味をなさない。(b)は中間層が一層もない。(c)は出力層につながる最終の中間層のユニット数が一つしかなく、このユニットの出力値は単純に結合係数倍されるだけであるから中間層の意味をなさない。

以下に、GAの具体的操作について説明する。ただし、NNの構造に関しては入力層と出力層のユニット数は固定であるので、図6. 6に示すようにGAの操作を加えるのは中間層のみとする。結合係数の学習においても、同様の操作を行った。

1) 交差

集団の中から確率 N_c で二つの個体（親）を選び出し、ある一定幅 W_c の遺伝子を交換し、一つの新しい個体（子）を作る。

2) 突然変異

確率 $1 - N_c$ で個体（親）を選び出し、すべての遺伝子に乱数 Δw を加え、新しい個体（子）を作る。

3) 淘汰

適応度の高いものから順に一定数の個体を選び、新しい世代の集団とする。

結合係数の学習においてGAのみの学習では、解近傍においてランダム探索に近くなることから、うまく収束しない[29]。従って、本システムでは、まずGAで学

習を有限回繰返し、最終的に得られた集団の中で最も適応度の高い個体を最適とみなし、バックプロパゲーション（BP）を用いて収束させた。BPは評価関数に式（6.10）を使い、式（6.12）に従って学習を行う[30]。

$$W(i+1) = W(i) + \eta \frac{\partial E}{\partial W(i)} \quad (6.12)$$

ここで、

i : 試行回数、 W : 結合係数、 η : 定数

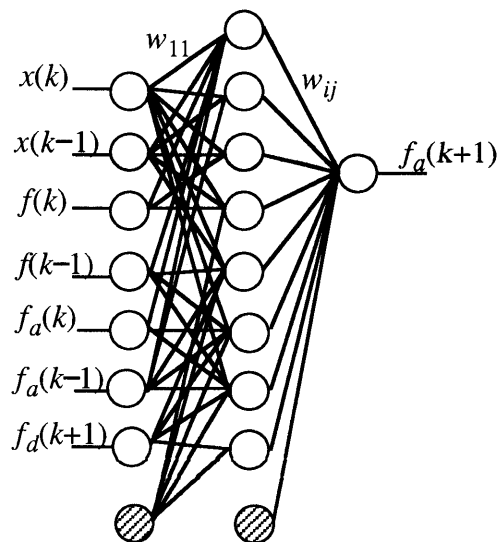


図6.5 NSCの構造

6. 3 シミュレーション

6. 3. 1 制御対象の数学モデル

シミュレーションは、図6. 9に示す1自由度マニピュレータを対象とし、未知対象物に対する力制御について行った。アクチュエータのクーロン摩擦および力センサ部分の動特性を考慮し、対象物と力センサは粘弾性でモデル化した。したがって、対象物の動特性変化は、剛性と粘性変化で表現した。このモデルに基づき、運動方程式を導くと以下のようなになる。

$$\ddot{x} = -a_1x - a_2\dot{x} - a_3w_x - a_4\dot{w}_x - a_5\text{sgn}(\dot{x}) + b_1f_a \quad (6.13)$$

$$\ddot{w}_x = -a_6x - a_7\dot{x} - a_8w_x - a_9\dot{w}_x \quad (6.14)$$

$$f = C_x(\dot{x} - \dot{w}_x) + K_x(x - w_x) \quad (6.15)$$

$$a_1 = K_x/M, \quad a_2 = (B_a + C_x)/M, \quad a_3 = -K_x/M$$

$$a_4 = -C_x/M, \quad a_5 = B_Q/M, \quad a_6 = -K_x/M_p, \quad a_7 = -C_x/M_p$$

$$a_8 = (K^* + K_x)/M_p, \quad a_9 = (C^* + C_x)/M_p, \quad b_1 = 1/M$$

$$\text{sgn}(\dot{x}) = \begin{cases} 1 & \dot{x} > 0 \\ 0 & \dot{x} = 0 \\ -1 & \dot{x} < 0 \end{cases} \quad (6.16)$$

ここで、

B_a : アクチュエータの等価粘性摩擦係数、

B_Q : アクチュエータのクーロン摩擦係数

シミュレーションに用いたマニピュレータのパラメータを表6. 1に、学習に用いたパラメータを表6. 2に示す。

6. 3. 2 NNSとNDPの追加学習の比較

図6. 2に示す応答パターンの追加学習について、NNSとNDPの比較を行った。学習する応答パターンは、対象物のばね定数 $K^*=1\text{kN/m}$ (Category 1) と

$K^*=30\text{kN/m}$ (Category 2) の 2 種類とし、サンプリング時間は 20ms 、サンプリング数 N は 20 とした。追加学習の手順は、最初にCategory 1について学習し、その後Category 2の応答パターンだけを提示し学習する。

NNSは3層モデルで、各層のユニット数は入力層から 40 、 10 、 2 個を用いた。入力にパターンベクトル U を与え、これを二つのカテゴリーに分類する。図6. 10に、NNSのシミュレーション結果を示す。図は対象物のばね定数が $K^*=0.1\sim 50\text{kN/m}$ の応答パターンに対するNNSの出力値を示している。発火のしきい値は 0.5 に設定した。図6. 10(a)は、Category 1だけを学習をしているが、未学習の対象物に対しても同じような出力値を示してしまっている。これでは未学習の判断は不可能である。図6. 10(b)は、Category 2を追加学習した結果を示している。結果はCategory 1は忘却され、 K^* のすべてにおいてCategory 2を示している。つまり追加学習が不可能であることを示している。

図6. 11にNDPの追加学習結果を示す。NDPの発火しきい値は $R=0.4$ とした。図に示すようにNDPはCategory 1を忘却することなくCategory 2を追加学習でき、未学習の判断が可能であることが分かる。

6. 3. 3 力制御のシミュレーション結果

本システムのサンプリング時間は 20ms とし、パターン認識のサンプリング数 N は 20 とした。また、パターン認識のために最初にアクチュエータ加える力は $0.5N$ とした。図6. 12に、本システムの対象物ばね定数が $K^*=1\text{kN/m}$ に対する応答結果を示す。押し付け開始から 0.4s 後に制御をNSC1に切り替えている。このNSC1は $K^*=1\text{kN/m}$ の対象物を学習しているので、良い応答を示している。しかし、新たな対象物 ($K^*=30\text{kN/m}$) が加わるとNSC1の汎化能力には限界があり、図6. 13に示すように、発振現象を起こしてしまう。システムは、この限界を出力自乗誤差で検出し、新たなカテゴリー (NSC2) を追加する。学習点における出力自乗誤差の 1.1 倍を許容誤差 E_a とした。図6. 14にNSC2による応答を示す。学習後は、良い応答を示している。図6. 15は、対象物の粘性係数が変化した場合のNSC2の応答を示す。図6. 16はNSC3を追加し、学習後の結果を示す。このように、一つのNSCではその汎化能力に限界があるので、対

象物に応じて切り替えることにより適応範囲を拡大できる。また、新たな動特性の異なる対象物が制御対象として加わっても、コントローラ（NSC）を自己生成することにより適応範囲を拡大できる。

NSCの学習及び構造探索の結果について、世代ごとの適応度の1例を図6.17と図6.18に示す。NSCの構造は、20世代の繰返しによって今回の制御対象に対しては最終的に、入力層から順にユニット数は7、6、4、1（結合数81）となった。同じような出力誤差を3層構造で実現すると7、10、1（結合数91）となった。多層構造にすることによって結合数を少なくできる。しかし、適応度の式（6.11）における β の決定に問題がある。 β が大きいと出力誤差が大きくなり、小さいとNNの規模が大きくなり汎化能力が狭まる。今回は、試行錯誤で決定したが、 β の決定はさらに検討を必要とする。

6.4 まとめ

本章では、追加学習が可能で、環境の特性変化に対して適応的にコントローラを自己生成し、適応範囲をより大きく広げることが可能な制御システムについて提案し、その可能性を示した。

今回、NSCの入力次数は固定としたが、これはプラントモデルの次数が既知でなければならない。完全に未知な対象の場合、プラントの次数、すなわちNSCの入力次数の決定も含めてNSCの構造を決定しなければならない。試行回数の低減、本システムの実システムへの適用も含めて、今後更に検討していかなければならない。

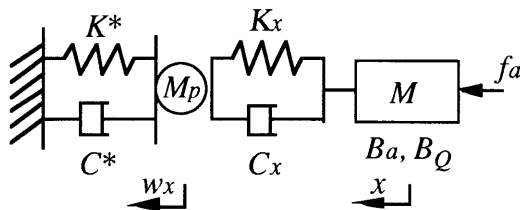


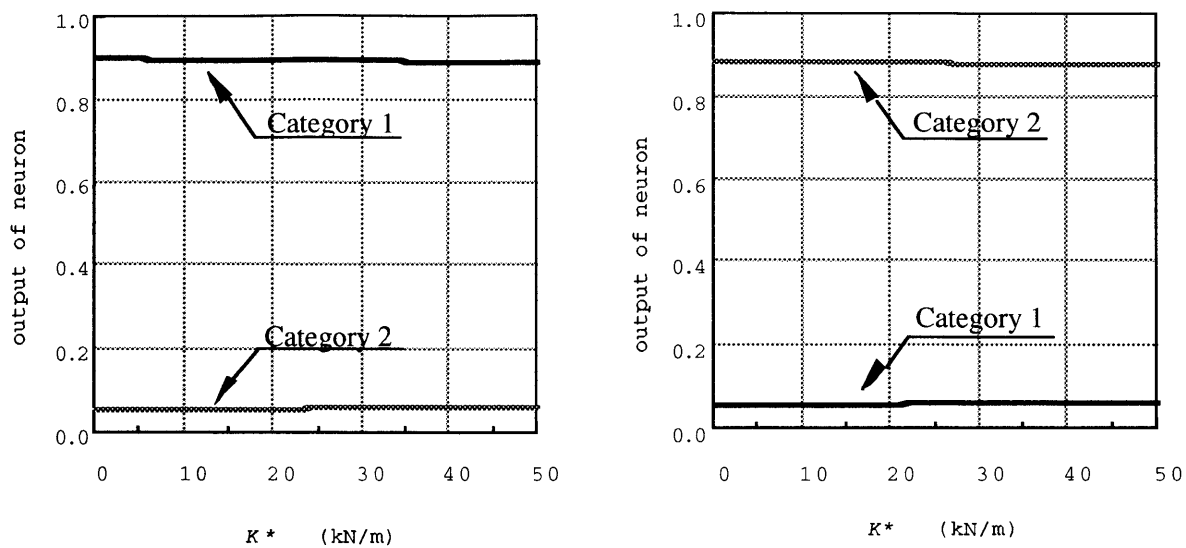
図6. 9 マニピュレータの数学モデル

表6. 1 マニピュレータのパラメータ

	Symbol	Value
Actuator	M	2 [kg]
	B_a	0.2 [Ns/m]
	B_Q	0.2[N]
End-effector	M_p	0.5 [kg]
Force sensor	K_x	2×10^4 [N/m]
	C_x	0.2 [Ns/m]
Object	K^*	$1 \times 10^3, 30 \times 10^3$ [N/m]
	C^*	0.5 [Ns/m], 10 [Ns/m]

表6. 2 学習のパラメータ

	Symbol	Value
NDP	N	20
	γ	3
	α	3
	A	0.01
	B	0.01
	R	0.4
GA (structure)	Nc	0.6
	Wc	5
	β	0.01
	Population	10
	Generation	20
	Maximum number of hidden layer Maximum number of unit in a hidden layer	3 10
GA (connection)	Nc	0.6
	Wc	number of allele / 2
	Population	40
	Generation	20
BP	η	0.001
	iteration	100



(a) 追加学習前

(b) 追加学習後

図6.10 NNSによる二つのカテゴリーの分類

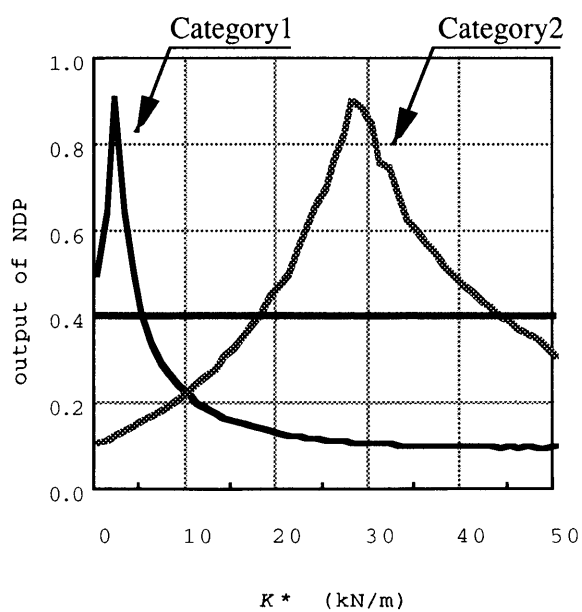


図6.11 NDPによる二つのカテゴリーの分類

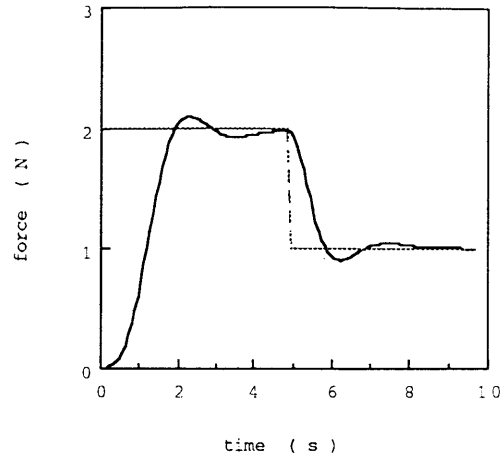


図 6. 1 2 NSC 1 によるステップ応答
 $(K^*=1 \text{ kN/m}, C^*=0.5 \text{ Ns/m})$

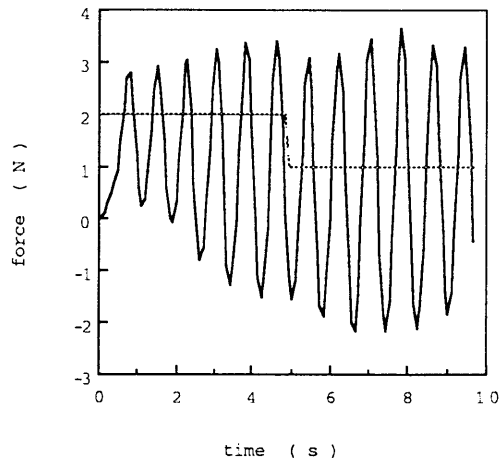


図 6. 1 3 NSC 1 によるステップ応答
 $(K^*=30 \text{ kN/m}, C^*=0.5 \text{ Ns/m})$

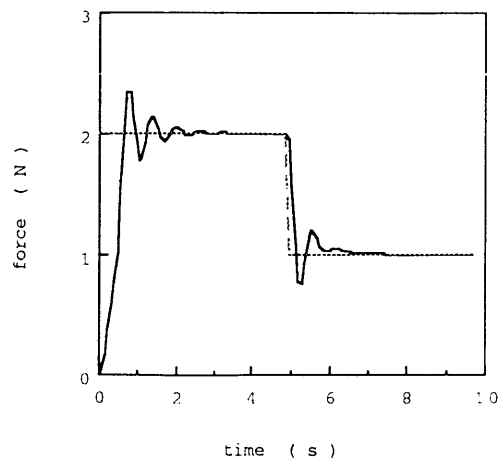


図 6. 1 4 NSC 2 によるステップ応答
 $(K^*=30 \text{ kN/m}, C^*=0.5 \text{ Ns/m})$

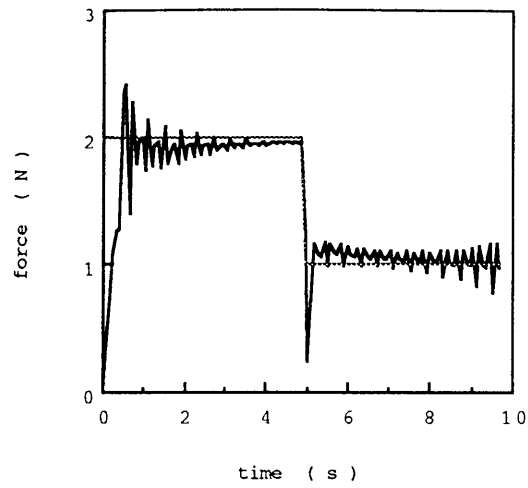


図 6. 15 NSC 2 によるステップ応答
($K^*=30$ kN/m, $C^*=10$ Ns/m)

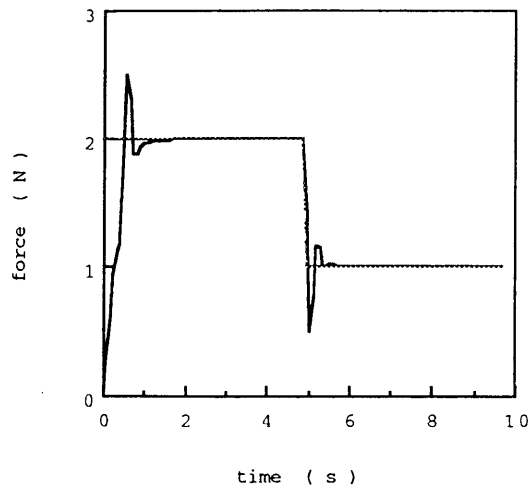


図 6. 16 NSC 3 によるステップ応答
($K^*=30$ kN/m, $C^*=10$ Ns/m)

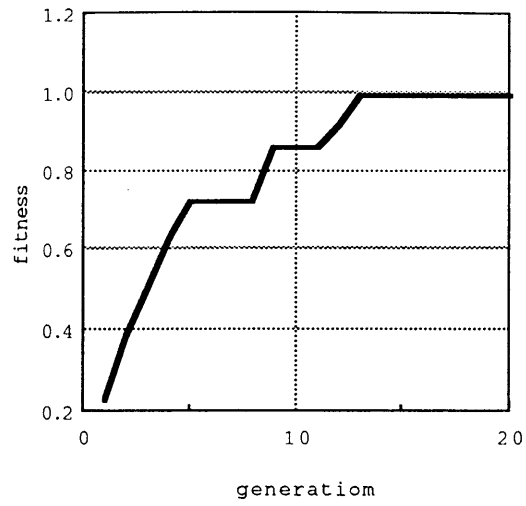


図6. 17 世代ごとの適応度
(NNの結合係数)

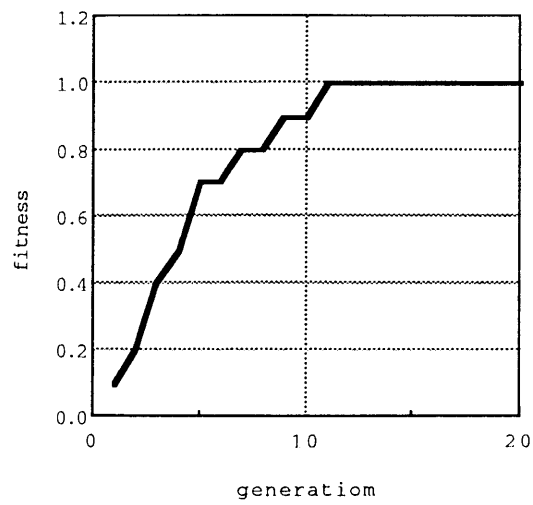


図6. 18 世代ごとの適応度
(NNの構造)

第7章 強化学習によるマニピュレータの最適接近速度の学習

本章では、対象物の動特性と据え付け位置に不確定要素が存在する環境下での力制御に、強化学習が有効であることを述べる。

7.1 組立作業ロボットへの強化学習の適用

近年、明確な教師がない環境での有効な学習アルゴリズムとして、強化学習 (Reinforcement learning) の研究がすすめられている。強化学習とは、環境から「報酬」あるいは「罰」という形で評価を得ることにより、より良い行動を探索、発見していく学習方法である。自律的なシステムの実現のための一手法として注目されている[34][35]。

組立作業をロボット・マニピュレータで行おうとすると部品の寸法誤差、位置決め誤差、マニピュレータの剛性、制御誤差、さらには環境の未知特性など、さまざまな不確定要素が存在するため、どのような行動をマニピュレータに与えれば、組立を確実に達成できるか明確でない。力情報を用いたり、力制御を行うことにより不確定性を適当に補償する手法[36]が提案されているが、環境をモデル化する必要がある、部品の形状が異なったり複雑になると、明確な行動を得ることが容易でなくなる。このようなプロセスに強化学習が有効と考えられる。作業の試行を繰り返し、経験を積み重ねていくうちに、スキルを獲得し、確実に作業を遂行できるようになるロボットが作れる可能性がある。ロボット・マニピュレータに、人間のように作業の試行を繰り返すことにより自律的に行動を獲得していく能力を持たせことができれば実に有効であると考ええる。

強化学習による組立作業のスキル獲得をめざし、本章では、組立作業の基礎となる力制御、特に接触動作をとりあげ、強化学習の可能性について述べる。環境(対象物)の特性が未知で、位置決め誤差が存在する接触動作において、強化学習による最適接近速度の獲得の可能性を検討した。接触動作における接近速度の最適化は北垣ら[37]の研究があるが、システムのパラメータが既知である必要があるし、対

象物の位置決め誤差がある場合についての検討はなされていない。

ロボットに強化学習を応用した研究は多数あるが、そのほとんどが移動ロボットに関するもので、マニピュレータの力制御や組立作業に関しては非常に少ない。Yangら[38]は、ダンピング制御のパラメータ学習に強化学習を用いる方法を提案している。Jordan [39]のニューラルネットワークモデルを用いており、報酬（強化信号）を与える環境をニューラルネットワークで同定しながら、その学習情報を基に行動決定のニューラルネットワークの学習を行っていく手法である。しかし、環境を同定するニューラルネットワークの学習に勾配法を使用しているため、強化信号は連続量である必要があるし、シーケンスの最後に強化信号が与えられるといったプロセスには適用できない。また、行動に対して強化信号が遅延を伴うことや各時刻で得られる強化信号を統合して最適化するという考慮がされていない。一方、Q-Learning [35]は動的計画法を基礎とした強化学習の手法であり、こういった強化信号の遅延の問題、強化信号を統合した最適化の考慮がなされている。

そこで、本章ではマニピュレータの接触動作の最適接近速度の獲得にQ-Learningを用いる方法を提案する。そして、対象物の特性が未知で位置決め誤差が存在する場合の1自由マニピュレータの接触動作を対象としてシミュレーションを行い、その有効性を示す。

7. 2 強化学習

7. 2. 1 強化学習の枠組み

強化学習とは、環境から「報酬」あるいは「罰」という形で評価を得ることにより、より良い行動を探索、発見していく学習方法である。その動作は同一手順の繰返しからなる。以下に、1サイクルの手順を示す。

(1) 環境の状態 X を観測し、行動 u を決定する。

$$u(t) = f(X(t)) \quad (7.1)$$

(2) 行動 u により、環境の状態が遷移する。

$$X(t+1) = g(X(t), u(t)) \quad (7.2)$$

(3) 状態の応じて、強化信号（報酬） r を環境から得る。

$$r(t+1) = h(X(t+1)) \quad (7.3)$$

(4) 強化信号 r を最大化する行動決定ルール f を探索する。

なお、状態遷移関数 g 及び強化信号関数 h についての知識は事前にはわからず。状態 X 、強化信号 r のみが観測可能である。強化学習は行動決定のルールを上記のような環境のもとに獲得していくことを目指している。

7. 2. 2 Q-Learning

Q-Learningでは各状態行動対 (X,u) に割り当てられた効用関数値 $Q(X,u)$ をもとに行動を決定し、次の状態での効用によりQ値を評価、更新していく学習アルゴリズムである。Q値はその時点から将来に渡って期待される効用の重みつき総和である。Q-Learningでは、Q値をユートイリティネットワークで学習により近似し、その値を最大化する行動を決定していく。学習は、次式のように行われる。

$$Q(X(t),u(t)) = Q(X(t),u(t)) + \beta[r + \gamma \max_{u \in U} Q(X(t+1),u(t+1)) - Q(X(t),u(t))] \quad (7.4)$$

ここで

r : 強化信号、 γ : 割引率 ($0 < \gamma < 1$)、

X : 環境の状態、 u : 行動、 β : 学習係数

準最適な行動から最適な行動を探索するための行動選択は、ある確率分布に従ってランダムに選択される。図 7.1にQ-Learningの構造を示す。

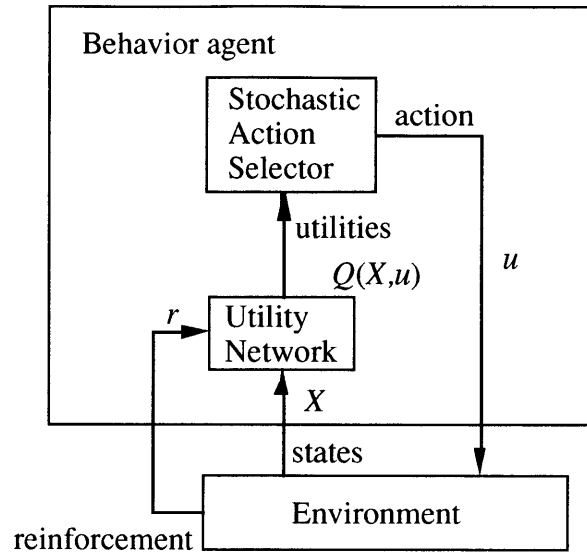


図 7. 1 Q-Learning

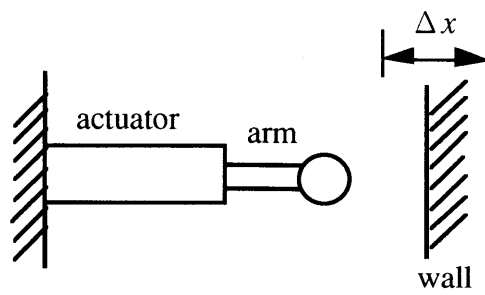


図 7. 2 1 自由度マニピュレータの接触動作

7. 3 制御対象と制御システムの構成

7. 3. 1 制御対象

本章では、図 7.2 に示すような 1 自由度のマニピュレータの接触動作を対象とする。非接触状態から接触状態へ移行し接触力を一定に保つ制御を行う。マニピュレータを使用した組立作業においては、部品とエンドエフェクタが非接触の状態から接触状態へ移行するプロセスは頻繁に行われ、状態遷移をスムーズに行う必要がある。接触動作は非線形な現象を伴い、従来からの線形制御理論では制御が困難である。また、環境（対象物）の特性が未知であったり、ハンドに把持した部品の寸法誤差、部品の位置決め誤差といった接触までの距離が不確定である場合が多い。そこで、本章では、対象物の特性が未知で、対象物の据え付け位置誤差 Δx がある場合の接触動作を想定し、強化学習により最適接近速度を求める手法を提案する。

7. 3. 2 制御システムの構成

図 7.3 と図 7.4 に、提案するシステムの構成を示す。制御系はダンピング制御を用いその目標接近速度 V_d を調整して接触動作を行う。アドミタンス A と制御ゲイン K_d は固定とする。

行動 u を目標接近速度 V_d として強化学習（Q-Learning）により探索することは可能である。しかし、Q-Learning では、学習空間の探索はランダム探索となることから、衝突を伴う力制御においては非常に危険である。従って、探索空間をある範囲に限定する必要がある。また、探索空間を限定することにより学習効率も良くなる。そこで、本方式では、図 7.5 に示すように、あらかじめ先験的知識をもとに接近速度 V_n を決めておき、強化学習により V_n を補正し、最適接近速度を求める手法をとる。強化学習により得る行動 u はその補正值 ΔV_d とする。従って目標接近速度 V_d は、

$$V_d = V_n + \Delta V_d \quad (7.5)$$

となる。

ユーティリティネットワークは小脳神経回路モデル（CMAC : Cerebellor Model Arithmetic Computer）[40]を用いた。図 7.6 に示すように、ユーティリティネット

ワークに入力する状態 X は、マニピュレータの位置 x 、速度 v 、力センサ出力 f とした。ポリシーは、このユーティリティネットワークの出力から最善な動作を決定する。本方式では次式のように、ユーティリティの最も高い行動 ΔV_d を選ぶものとする。

$$\Delta V_d(X) = \arg \max_{\Delta V_d \in V} Q(X, \Delta V_d) \quad (7.6)$$

また、学習空間の探索は、乱数により確率的に動作を選択することで行う。

学習アルゴリズムを以下に示す：

```

 $n = \text{Random} (\{z | 0 \leq z \leq 1\})$ 
if  $n < \delta$ 
  then  $\Delta V_d(t) = \arg \max Q(X(t), \Delta V_d(t))$ 
  else  $\Delta V_d(t) = \text{Random} (V)$ 
 $\langle X(t+1), r \rangle = \text{Simulate} (X(t), \Delta V_d(t))$ 
 $Q(X(t), \Delta V_d(t)) = Q(X(t), \Delta V_d(t))$ 
  +  $\beta [r + \gamma \max Q(X(t+1), \Delta V_d(t+1))$ 
  -  $Q(X(t), \Delta V_d(t))]$ 

```

接触動作開始から、時刻 N までの間を 1 回の試行とし、その試行の最後で強化信号を与えた。強化信号 r は、1 回の試行の間の目標接触力 f_d と力センサ出力 f の偏差の自乗和とした。

$$r = - \sum_{i=1}^N (f_d(t) - f(t))^2 \quad (7.7)$$

従って、式 (7.7) に示す接触力偏差の自乗和を最小にする接近速度が最適接近速度となる。

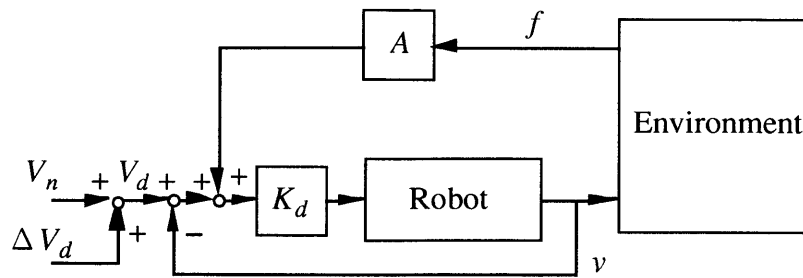


図 7. 3 ダンピング制御系の構成

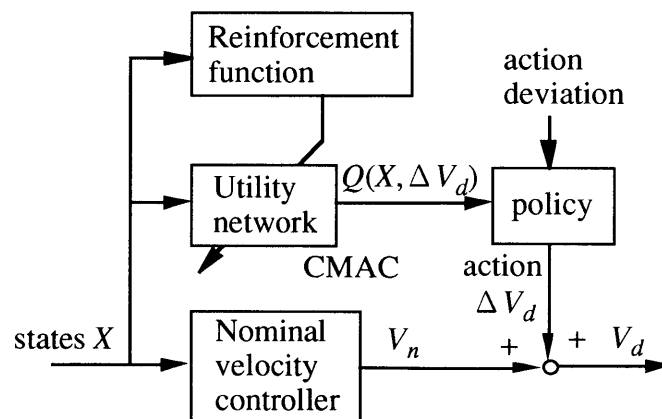


図 7. 4 行動主体の構造

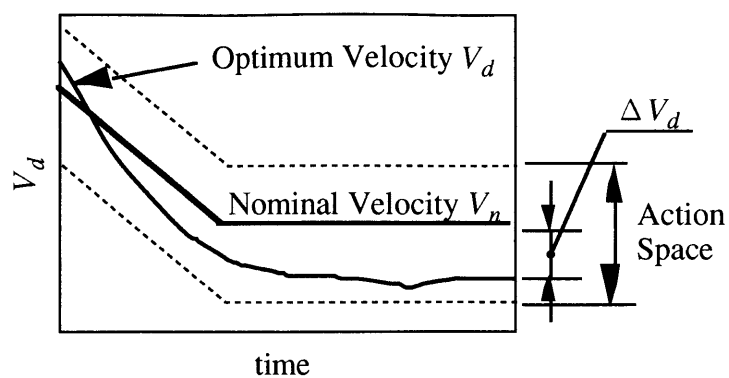


図 7. 5 学習空間

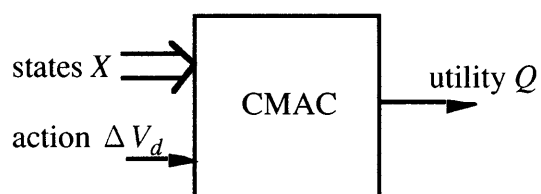


図 7. 6 ユーティリティネットワークの構造

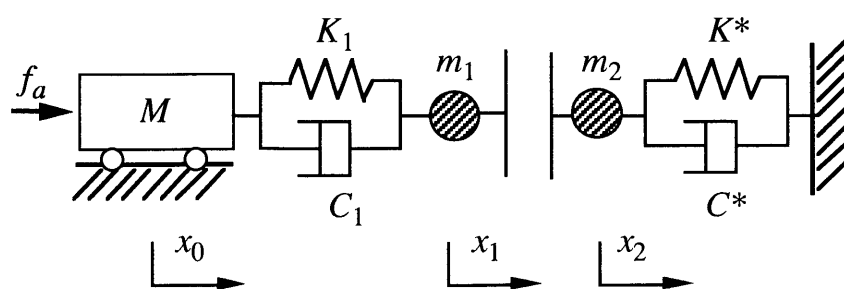


図 7. 7 制御対象の数学モデル

7. 4 シミュレーション方法

7. 4. 1 制御対象の数学モデル

図7.7に示すように、力センサおよび対象物を線形なばねとダンパと質量でモデル化した。従って運動方程式は次のようになる。

マニピュレータ：

$$\begin{aligned} M\ddot{x}_0(t) = & -K_1\{x_0(t) - x_1(t)\} \\ & -C_1\{\dot{x}_0(t) - \dot{x}_1(t)\} - f_a \end{aligned} \quad (7.8)$$

先端質量：

$$\begin{aligned} m_1\ddot{x}_1(t) = & -K_1\{x_1(t) - x_0(t)\} \\ & -C_1\{\dot{x}_1(t) - \dot{x}_0(t)\} - f_w \end{aligned} \quad (7.9)$$

対象物：

$$m_2\ddot{x}_2(t) = -K^*x_2(t) - C^*\dot{x}_2(t) + f_w \quad (7.10)$$

ここで、

M ：マニピュレータの質量、

m_1 ：先端質量、 m_2 ：対象物の質量、

x_0 ：マニピュレータの位置、

x_1 ：マニピュレータ先端の位置、

x_2 ：対象物の位置、 f_a ：制御入力、

K^* ：対象物のばね定数、 C^* ：対象物の粘性係数、

K_1 ：力センサのばね定数、

C_1 ：力センサの粘性係数、

f_w ：衝突力

力センサの出力 f は次式となる。

$$f(t) = K_1\{x_0(t) - x_1(t)\} + C_1\{\dot{x}_0(t) - \dot{x}_1(t)\} \quad (7.11)$$

<衝突のモデル化>

衝突力は次式を用いて表現する[41]。

$$f_w = \begin{cases} (1 + p\dot{u})Hu^{3/2} & (u \geq 0 \text{ and } \dot{u} \geq -1/p) \\ 0 & (\text{otherwise}) \end{cases} \quad (7.12)$$

$$u = x_1 - x_2, \quad \dot{u} = \dot{x}_1 - \dot{x}_2 \quad (7.13)$$

ここで、

u : 対象物とマニピュレータ先端の相対位置

\dot{u} : 対象物とマニピュレータ先端の相対速度

p : エネルギー損失係数

H : Hertz の値

衝突による力は正（お互いに反発する力を及ぼし合う状態）のみとする。従って、

$$u \geq 0 \quad \text{かつ} \quad \dot{u} \geq -1/p \quad (7.14)$$

のときを接触状態、その他を非接触状態と考える。

7. 4. 2 シミュレーション方法

マニピュレータ先端と対象物との距離を 1 (mm) とし、その位置から接触動作を開始する。対象物の据え付け位置誤差は $\Delta x = \pm 0.3$ (mm) 存在するものとした。コントローラでは、常にスタート位置から 1 (mm) 先に対象物があるものとして観測される。また、力の目標値は $f_d = 5$ (N) とした。

制御のサンプリング周期 T_c は 2 (ms) とし、強化学習のサンプリング周期 T_r は T_c の 3 周期ごとの 6 (ms) とした。目標速度を制御のサンプリング周期毎に変化させても応答に時間がかかるためあまり意味をなさないので、強化学習のサンプリング周期を大きくとった。また、1 試行時間は 0.5 (s) とした。シミュレーションに用いたマニピュレータのパラメータを表7.1 に示す。強化学習のパラメータは、割引率を $\gamma = 0.5$ 、学習係数を $\beta = 0.05$ とした。

ユーティリティネットワーク (CMAC) において、マニピュレータの位置は $-2.0 \leq x \leq +2.0$ (mm)、速度は $-10 \leq v \leq +10$ (mm/s)、力は $0 \leq f \leq +10$ (N)

をそれぞれ 20 に量子化し、範囲外はスクワッシングを行った。接近速度の補償値は $-1.6 \leq \Delta V_d \leq +1.6$ (mm/s) を 16 に量子化した。荷重の活性化帯域幅を 3 とし、出力は 6 個の荷重の和で求めた。

7. 5 シミュレーション結果

7. 5. 1 対象物の動特性が未知の場合

まず、対象物の動特性が未知で、対象物の据え付け位置誤差がない場合についてのシミュレーション結果を示す。

図 7.8 に、強化学習を適用する前の状態を示す。図はマニピュレータの位置 x_0 、力センサ出力 f 、衝突力 f_w 、目標速度 V_d を示している。学習前に図 7.8 (d) に示す目標接近速度を与えた。図 7.8 (a) のマニピュレータの位置は、対象物の表面を基準 ($x_0 = 0$ mm) として示してある。接触前に力センサ出力が出ているが、これは、センサの前に取り付けてあるマニピュレータ先端質量の加速による影響である。

図 7.8 では、対象物の特性が未知のため接触力の偏差が生じている。また、接近速度を早めに減速したため接触力の立上りが悪くなっている。図 7.9 に、強化学習を適用後の結果を示す。図 7.9 (e) に示す ΔV_d は強化学習にて得た V_d の補償値を示している。図に示すように接触力の立上りが早まり偏差がなくなっている。強化学習により接近速度の補償、及び対象物の未知特性の補償が可能なのことがわかる。

7. 5. 2 対象物の特性が未知で据え付け位置誤差がある場合

図 7.10 と図 7.11 に強化学習を適用する前の応答を示している。図 7.10 は $\Delta x = +0.3$ (mm) の場合で、実際の距離が測定距離より遠いため早めに速度を落としてしまい、接触力の立上りが悪くなり、力の偏差が大きくなっている。図 7.11 は $\Delta x = -0.3$ (mm) の場合で、実際の距離が測定距離より近いため、速度を落とすのが遅くなり、力のオーバーシュートが大きくなっている。図 7.12 と図 7.13 に、強化学習を適用後の結果を示す。図に示すように、強化学習により対象物の据え付け位置誤差の補償が可能なのことがわかる。なお、学習において Δx は ± 0.3 (mm) の間を

ランダムに与えた。

7. 5. 3 学習の収束性

図 7.14 に、学習の収束性を示す。図は、試行 50 回ごとの接触力偏差の自乗和 (式 (7.7)) を示している。試行を積み重ねることにより誤差が減少することがわかる。しかし、少なくとも 3000 回の試行を必要とし実用的とはいえない。これは、強化信号がシーケンスの最後で与えられるため、強化信号が時間を逆方向に伝播するのに時間がかかるためだと考えられる。

与える強化信号の情報量を多くする目的で、式 (7.15) のように、各サンプリング時刻ごとの接触力の自乗誤差を強化信号とし、毎サンプリングごとに与えた結果を図 7.15 に示す。

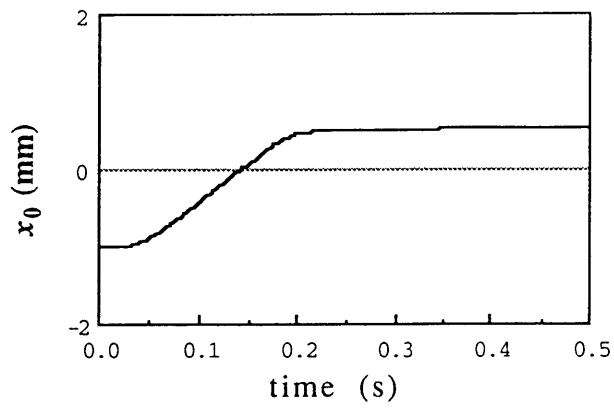
$$r = -(f_d(t) - f(t))^2 \quad (7.15)$$

図は式 (7.4) の割引率 γ が $\gamma = 0$ の場合と $\gamma = 0.4$ の場合について示してある。 $\gamma = 0$ の場合は学習が収束しなかったが、 $\gamma = 0.4$ ではシーケンスの最後で強化信号を与える場合よりずっと早く収束していることがわかる。 $\gamma = 0$ の場合、行動に対して強化信号が遅延を伴うため、各状態での直接の強化値のみでは学習ができないことを示している。Q 値は、各状態での直接の強化値と後続状態の減衰 (割引) 強化値の和であり、各時刻で得られる強化信号を統合して最適化を方策している。従って、強化信号が行動に対して遅延を伴う場合でも学習が可能であることを示している。

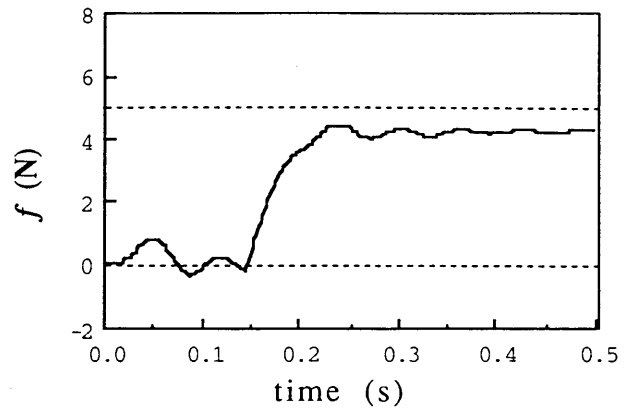
7. 6 まとめ

本章では、対象物の特性と据え付け位置に不確定性が存在する環境下での力制御に、強化学習が有効であることを示した。

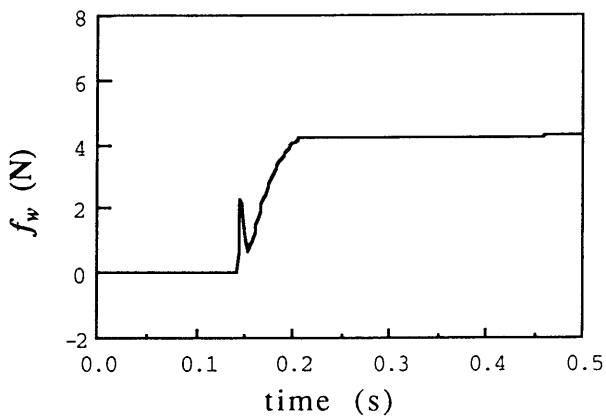
強化学習は教師あり学習法に比べ弱い学習法であり、外界から得られる情報量が少ないため、学習に膨大な時間を要する。探索領域を限定し、強化信号の情報量を多くすることにより、膨大な学習時間のある程度に抑えることが可能であることを示した。しかし、実用的な試行回数からみて、まだ検討の余地がある。



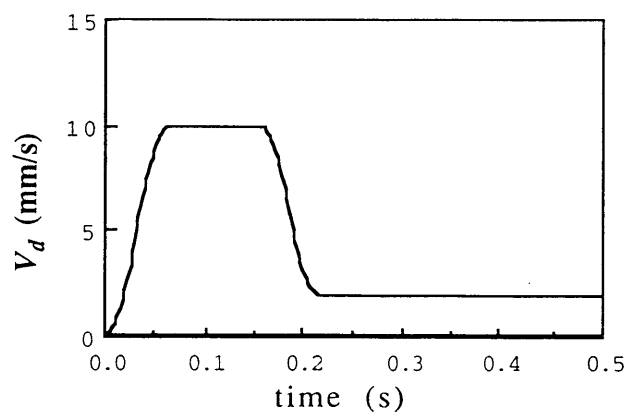
(a) Position of a manipulator



(b) Force at a tip

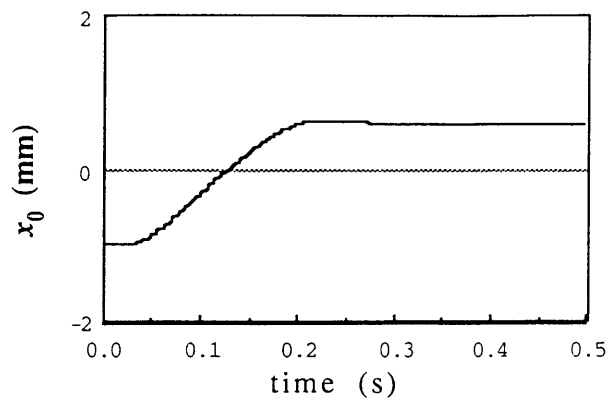


(c) Impact force

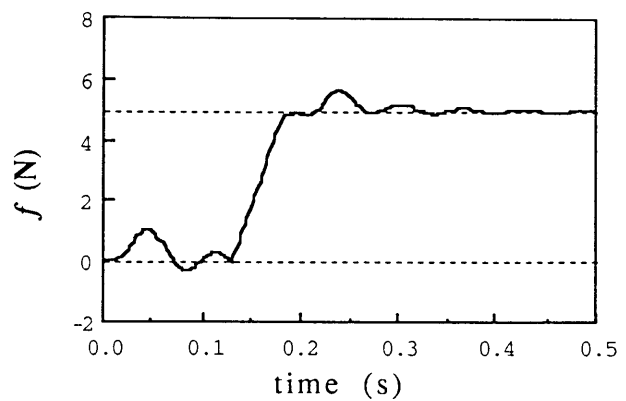


(d) Desired velocity

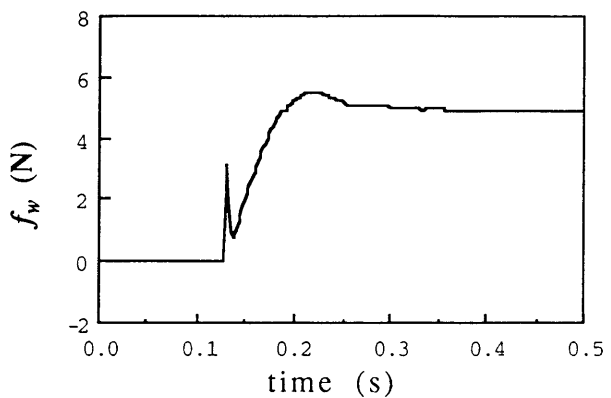
図7. 8 未知対象物に対する応答結果（強化学習なし）



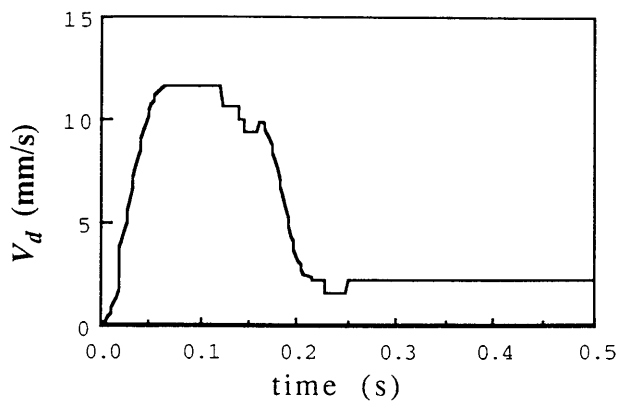
(a) Position of a manipulator



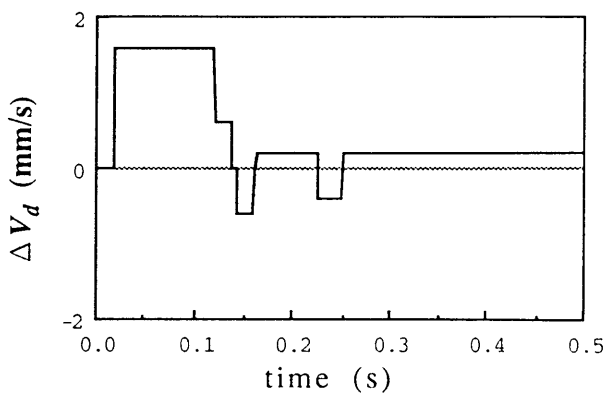
(b) Force at a tip



(c) Impact force

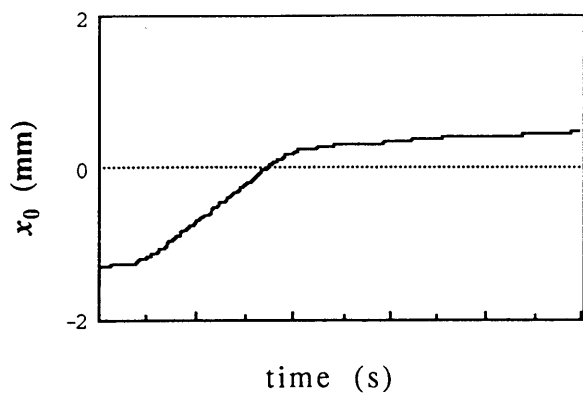


(d) Desired velocity

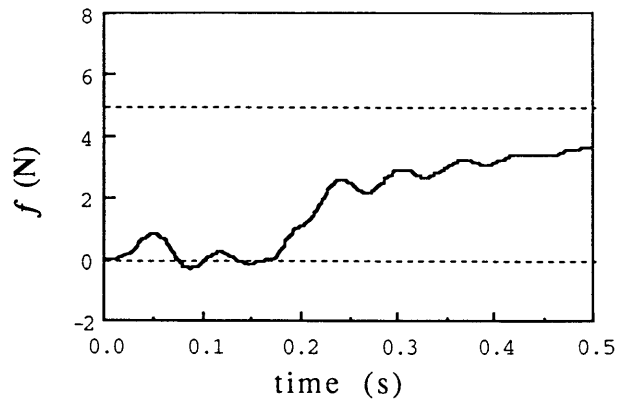


(e) Compensated velocity

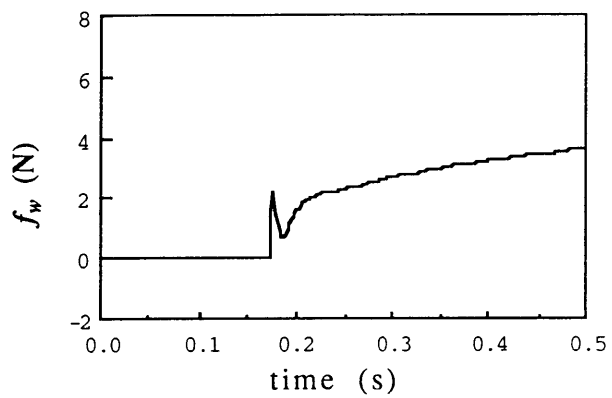
図7. 9 未知対象物に対する応答結果（強化学習あり）



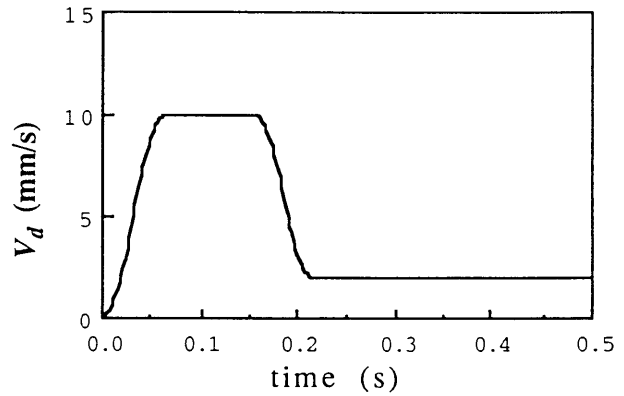
(a) Position of a manipulator



(b) Force at a tip

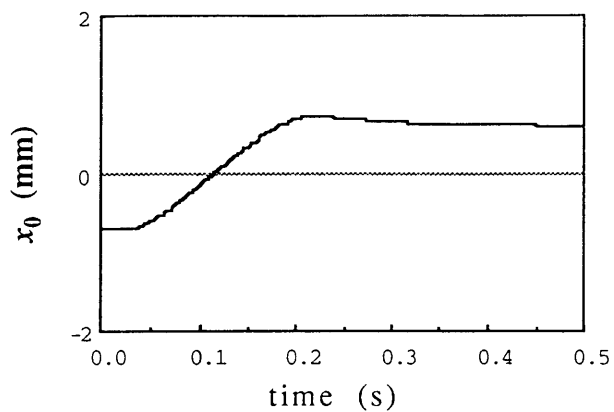


(c) Impact force

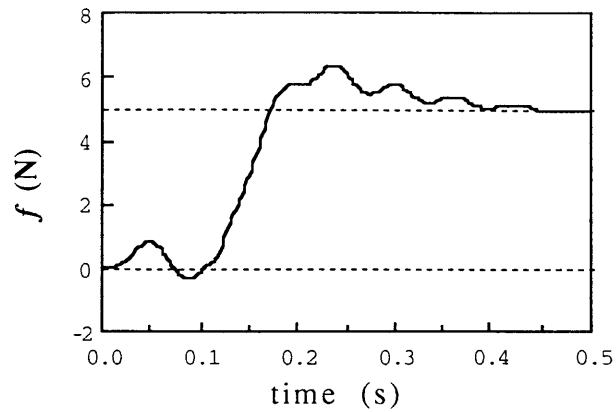


(d) Desired velocity

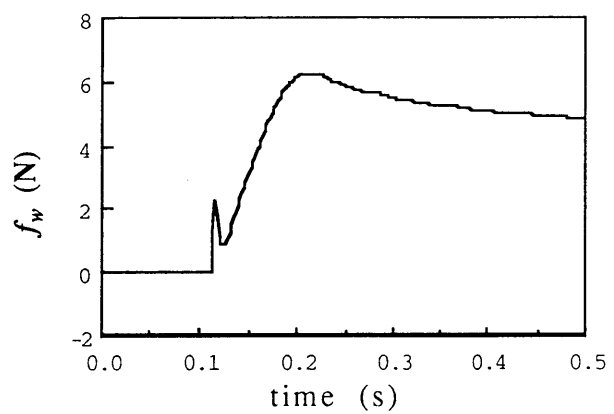
図7. 10 位置誤差 $\Delta x = +0.3$ (mm) の応答結果 (強化学習なし)



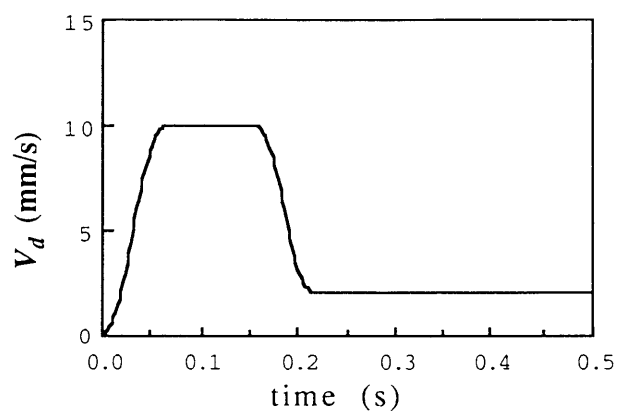
(a) Position of a manipulator



(b) Force at a tip

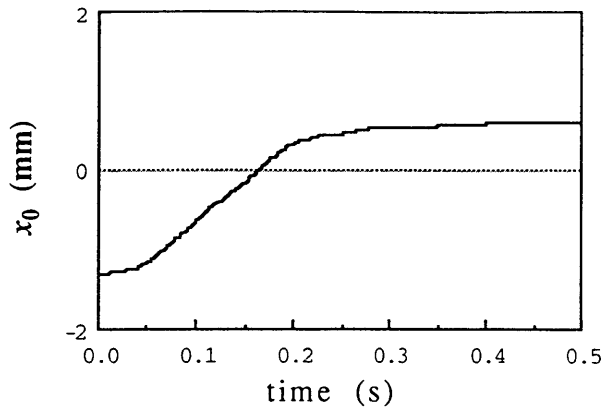


(c) Impact force

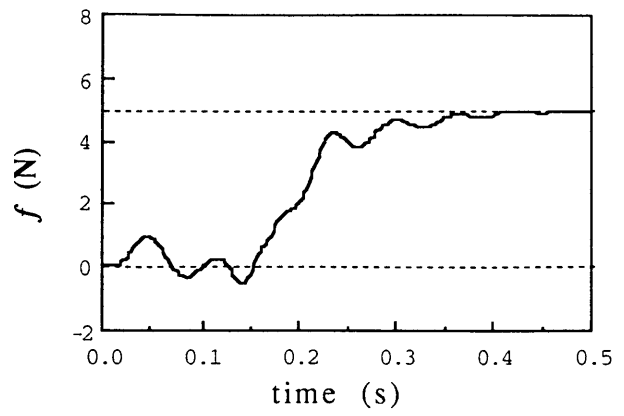


(d) Desired velocity

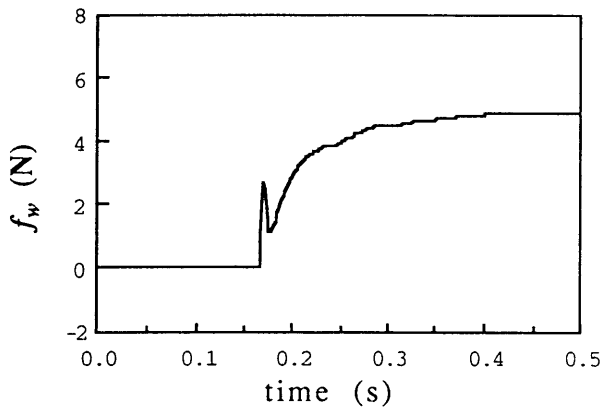
図7. 1 1 位置誤差 $\Delta x = -0.3$ (mm) の応答結果 (強化学習なし)



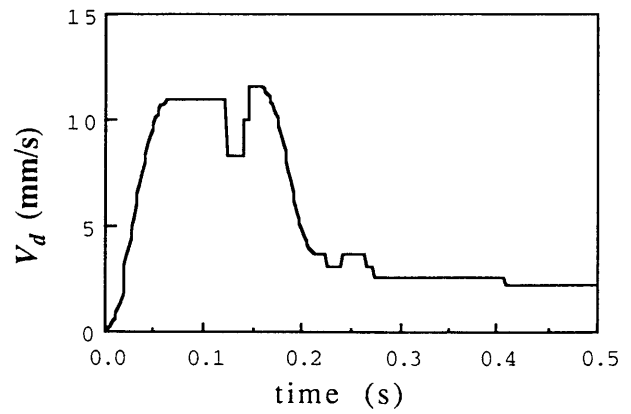
(a) Position of a manipulator



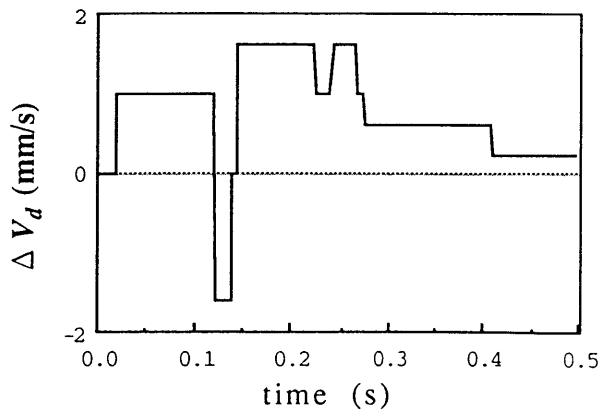
(b) Force at a tip



(c) Impact force

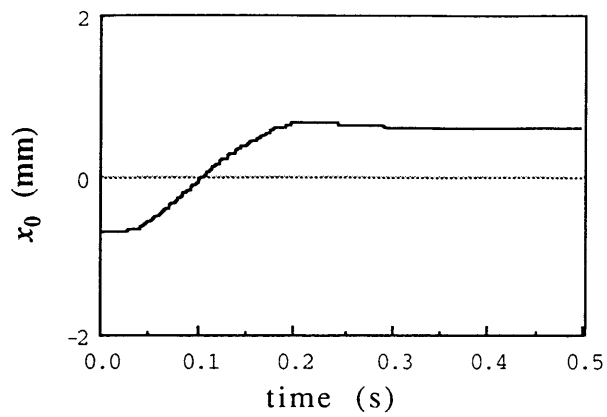


(d) Desired velocity

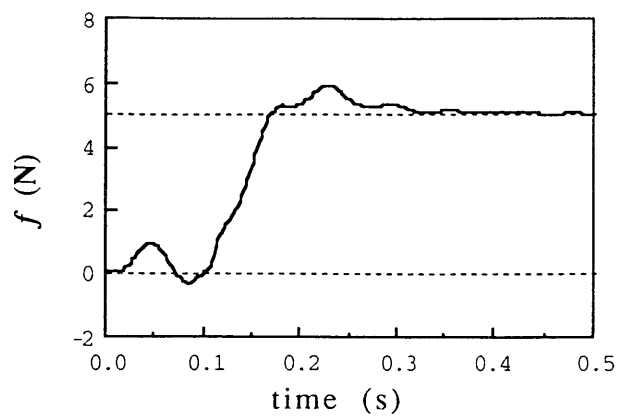


(e) Compensated velocity

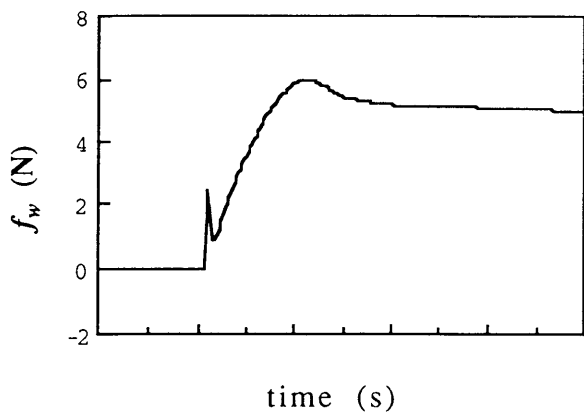
図7. 1 2 位置誤差 $\Delta x = +0.3$ (mm) の応答結果 (強化学習あり)



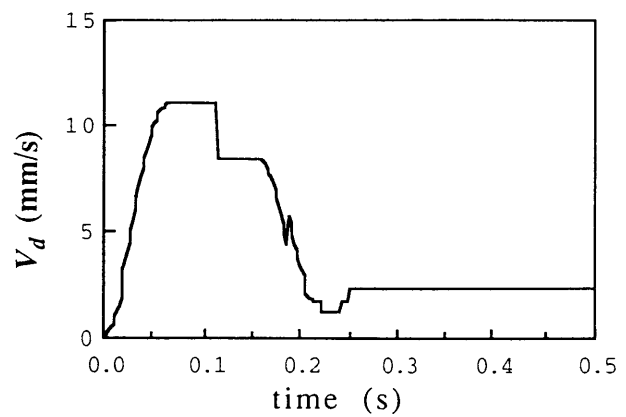
(a) Position of a manipulator



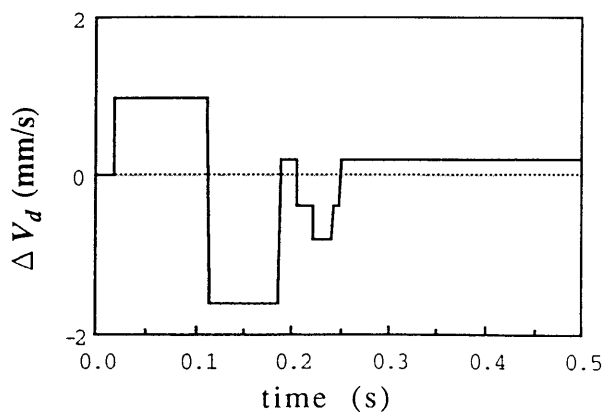
(b) Force at a tip



(c) Impact force



(d) Desired velocity



(e) Compensated velocity

図7. 13 位置誤差 $\Delta x = -0.3$ (mm) の応答結果 (強化学習あり)

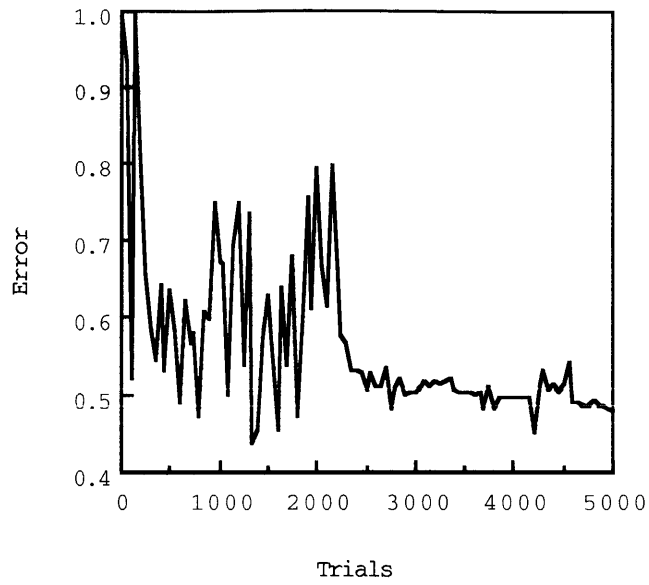


図7. 1 4 学習の収束性

シーケンスの最後で強化信号が与えられる場合

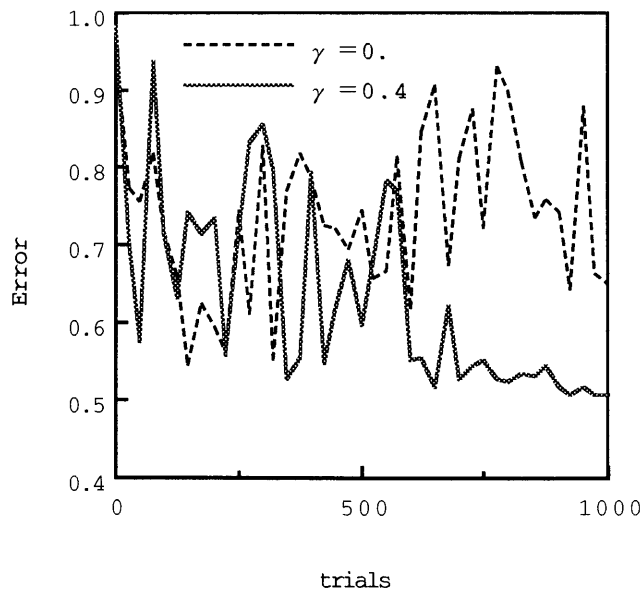


図7. 1 5 学習の収束性

毎サンプリング時刻で強化信号が与えられる場合

表7. 1 マニピュレータのパラメータ

	Symbol	Value
Actuator	M	0.2[kg]
End-effector	m_1	0.05[kg]
Force sensor	K_1 C_1	2.5×10^4 [N/m] 40[Ns/m]
Object	m_2 K^* C^*	0.04[kg] 2×10^4 [N/m] 10[Ns/m]
Impact force	H p	8×10^4 0.6

第8章 結論

8.1 本研究の成果

本研究では、ロボットマニピュレータの位置と力の制御システムにニューラルネットワークを使用し、制御対象物の動特性変化、非線形性の強い制御対象に対して、柔軟に対応することが可能な制御方法について研究し、シミュレーションおよび実験により提案した手法の有効性を示した。

まず第2章では、本研究で使用しているニューラルネットワークモデルの構成および学習について、その概要を述べ、ニューラルネットワークをコントローラとして用いるための構成方法と学習方法についても述べた。また、ニューラルネットワークをコントローラとして用いる場合の利点およびニューラルコントローラの学習における問題点を明らかにした。

第3章では、ニューラルネットワークを用いた、未知対象物に対する力制御方式について述べた。対象物との相互作用を伴う制御システムでは、閉ループ内に対象物の動特性が含まれるため、対象物の動特性が既知でないと制御系の設計ができない。また、対象物だけでなく歯車のバックラッシュ、摩擦といったマニピュレータ機構の非線形性、パラメータ推定の困難さがあり、制御系の設計が容易でない。更に、対象物の特性が大きく変化する場合、固定フィードバックゲインによる制御では適応できない。そこで、ニューラルネットワークの学習機能、非線形マッピングを用いることにより、従来の制御方式と比較して厳密な数学モデルを必要とせず、未知対象物に適応でき、制御対象の大幅な特性変化に対しても適応できることを示した。そして、ニューラルネットワークの構成について検討をし、層数、中間層のユニット数、学習点数の指標を与えた。

第4章では、第3章で述べた適応力制御システムを多自由度マニピュレータに拡張する方法について述べた。マニピュレータによるグライディングや刷毛塗りといった作業では、位置と力のハイブリッド制御が有効である。環境によってエンドエフェクタが拘束されている方向には力を制御し、それ以外の方向には位置および姿勢を制御する制御方式である。この場合も、対象物の動特性が、制御の閉ループ

内に入るため、対象物の動特性が大きく異なる場合や未知対象物に対して、安定な制御が困難となる。また、多自由度マニピュレータは非線形システムであり、従来のハイブリッド制御システムでは、姿勢変化による不安定性が問題とされている。そこでニューラルネットワークを使用した、対象物の動特性の変化やマニピュレータの姿勢の変化に対してロバストな制御システムが構築でき実現可能であることを示した。

第5章では、非線形性の強いスタビング制御にニューラルネットワークが有効であることを述べた。マイクロマニピュレータで生物の細胞内の組織を破壊せずに細胞膜を破るといった制御や、マニピュレータを使ってドリルによる穴あけを行うといったスタビング制御では対象物の破壊前後において、動特性が大きく異なる。破断から反動抑制への一連の動作をスムーズに行う必要があるため、このような非線形特性を有する制御対象には、従来からの制御システムでは適応が困難である。位置情報と力情報を組み合わせ統合的に処理するようなハイブリッド制御システムが必要になる。ニューラルネットワークの学習機能、非線形マッピングを用いて、非線形特性を有する対象物の制御を順応的に行うことが可能であることを示した。また、ニューラルネットワークにより位置情報と力情報を融合して膜の破断の認識が可能であることを示した。

第6章では、新しい知識を追加したり間違っただけの知識を修正する追加学習を考慮した、制御システムについて述べた。一般的に、ニューラルサーボコントローラには、シグモイド関数を用いた多層型ニューラルネットワークを使用しており、学習にはバックプロパゲーションアルゴリズムを用いているため、新しい知識を追加したり間違っただけの知識を修正する追加学習ができない。特に、環境（対象物）まで含めた系として扱わなければならない力制御やハイブリッド制御では、動特性の大きく異なる未学習の対象物が新たに制御対象として加わった場合、今まで学習した全ての対象物を含めて再度学習し直す必要があるため非常に効率が悪い。新たなパターンを追加学習するためには、今まで学習したパターンをすべて用意しなければならないし、学習も最初からやり直さなければならないため、追加のための学習に多くの手間と時間を必要とする。そこで一種のラジアル・ベース関数を使用したニューラルネットワークと一般的なニューラルサーボコントローラを階層的

に組み合わせることにより、追加学習が可能となり、環境の特性変化に対して適応的にコントローラを自己生成し、適応範囲をより大きく広げることが可能な自律型インテリジェント制御システムの構築が可能であることを述べた。

第7章では、マニピュレータの接触動作の最適接近速度の獲得に強化学習（Q-Learning）を用いる方法について述べた。組立作業をロボット・マニピュレータで行おうとすると部品の寸法誤差、位置決め誤差、マニピュレータの剛性、制御誤差、さらには環境の未知特性など、さまざまな不確定要素が存在するため、どのような行動をマニピュレータに与えれば、組立を確実に達成できるか明確でない。このようなプロセスに強化学習が有効と考え、第7章では、組立作業の基礎となる力制御、特に接触動作をとりあげ、強化学習の可能性について述べた。環境（対象物）の特性が未知で、位置決め誤差が存在する接触動作において、強化学習による最適接近速度の獲得の可能性を検討し、有効性を示した。

8. 2 今後の課題

(1) ニューラルネットワークのサーボレベルだけでなくさらに上位のレベルへの展開

ニューラルネットワークをサーボレベルだけでなくプランニングレベルへ適用し、ニューラル・コントロール・システムを階層構造にすることにより、より適応範囲を広げ一般化したコントロールシステムを構成することが可能である[31][32]。

制御システムのプランニングにおいて、例えばグライディング（研削）制御システムのように、数多くの因子によって影響される複雑な加工プロセスの制御では、その加工条件を決定するのは容易ではない。いろいろな条件での加工データを測定し、加工現象をモデル化する時に、ニューラルネットワークが有効となる[33]。ニューラルネットワークは、あらゆる連続な非線形マッピングを近似得る、しかもマッピングの因子の数が多くても近似が可能であり、並列処理機能により演

算時間が短縮できる。また、数少ないデータでも補間機能を利用できる。

(2) 強化学習による組立制御則の学習

本論文の第7章において、対象物の特性と据え付け位置に不確定性が存在する環境下での力制御（接触動作）に、強化学習が有効であることを述べた。今後は、さらに組立作業へ本方式を拡張する必要がある。本方式では、ダンピング制御の目標接近速度のみ強化学習にて得たが、同時にアドミタンスをも学習していけば、組立作業中に生ずる位置決め誤差などの不確定性を減少させるように運動を適宜修正することが可能となる。

強化学習作業の経験を積み重ねていくうちに、スキルを獲得し、確実に作業を遂行できるようになる自律型インテリジェントシステムを構築できる可能性は十分あると考える。

参考文献

- [1] W.S.McCulloch and W.H.Pittsa, A Logical Calculus of the Ideas Immanent in Neural Nets, Bull.Math.Biophysics.5, pp.115-123 (1943)
- [2] J.J.Hopfield, Neural Networks and Physical Systems with Emergent Collective Computational Abilities, Prc. of the National Academy of Sciences, Vol.79, pp.2554-2558 (1982)
- [3] T.Kohonen, Self-organized formation of topologically correct feature maps, Biolog. Cybernetics, Vol.43, pp.59-69(1982)
- [4] 中野、アソシアトロン、晃光堂 (1986)
- [5] F.Rosenblatt, The perceptron: a probablistic model for information storage and oganization in the brain, Psychological Review 65, pp.368-408 (1958)
- [6] Rumelhart,Hinton and Williams, Learning repres-entation by back-propagating errors,Nature,323 pp.533-536,(1986)
- [7] 銅谷、リカレントネットワークの学習アルゴリズム、計測と制御、Vol.40, No.4, pp.296-301 (1991)
- [8] Narendra, K.Parthasaraty, Identification and Control of Dynamical Systems Using Neural Networks, IEEE Trans. on Neural Networks, Vol.1, pp.4-27 (1990)
- [9] 藪田、山田、ニューラルネットワークによる力制御、日本ロボット学会誌、Vol.9, No2, pp.224-231 (1991)
- [10] H.Gomi, M.Kawato, Neural Network Control for a closed-loop system using feedback error learning, Neural Networks, Vol.6, No.7, pp.933-946 (1993)
- [11] 鴫田、光岡、福田、柴田、” 神経回路モデルによるロボットマニピュレータのハイブリッド制御 (2自由度マニピュレータの適応制御)”、日本ロボット学会誌、Vol.9, No.1, pp.45-54 (1991) .
- [12] R.M.Sanner, D.L. Akin, Neuromorpjic Pitch Attitude Reguration of an Undermater Telerobot, IEEE Control System Magazine, 10-3, pp.62-68 (1990).

- [13] M.I.Jordan, Generic Constrains on Underspecified Target Trajectories, Proc. of IJCNN89, Washignton D.C, (1989)
- [14] 川人、随意運動制御における適応と学習、日本ロボット学会誌、Vol.4, No.2、pp.184-193、(1986)
- [15] D.E.Goldberg, Genetic Algorithms in Serch, Optimization, and Machine Learning, Addison Welsey (1989)
- [16] 福田、北村、谷江、対象物の特性を考慮したロボットの力制御方式、機論、53-496、pp.2577-2583、(1987)
- [17] 福田、北村、谷江、対象物の動特性を考慮したグリッパの適応ハイブリッド力制御、日本ロボット学会誌、Vol.6, No.6, pp. 91 - 100 (1988).
- [18] M.H.Raibert, J.J.Craig, Hybrid Position/force Control of Manipulators, Trans. ASME, Journal of Dynamic Systems, Measurement and Control, Vol.102, pp. 126 - 133 (1981)
- [19] C.H.An, J.M.Hollerbach, Dynamic Stability Issues in Force Control of Manipulators, Proceedings of IEEE International Conference on Robot & Automation, pp. 890 - 896 (1987).
- [20] 吉川、杉江、田中、ロボットアームの位置と力の動的 ハイブリッド制御（制御系設計と実験）、日本ロボット学会誌、Vol.6, No.5, pp 380 - 387 (1988).
- [21] 薮田、ハイブリッド制御系の安定性に関する考察、第6回日本ロボット学会学術講演会予稿集、pp .471 - 474 (1988)
- [22] 川人、宇野、鈴木、随意運動における適応と学習2、日本ロボット学会誌、Vol.6, No.3, pp.222-230 (1988)
- [23] 福田、塩谷、新井、柴田、佐々木、竹内、木下、新規パターンを追加学習するための新しいニューロン・モデル、計測自動制御学会論文集、Vol.29. No.3, pp.356-364, (1993)
- [24] K.S.Narendra, S.Mukhopadhyay, Intelligent Control Using Neural Networks, IEEE Control Systems, Vol. 12, No.2 , pp.11-18, (1992)

- [25] D.J.Cooper, L.Megan, R.F.Hinde, Disturbance Pattern Classification and Neuro-Adaptive Control, IEEE Control Systems, Vol. 12, No.2 , pp.42-48, (1992)
- [26] T.M.Nabhan, A.Y.Zomaya, Toward Generating Neural Network Structures for Function Approximation, Neural Networks, Vol.7, No.1, pp.89-99, (1994)
- [27] Y. Ichikawa, T. Sawa, Neural Network Application for Direct Feedback Controllers, IEEE Trans. Neural Network, Vol.3, No.2, pp.224-231, (1992)
- [28] 下島、福田、新井、RBFファジイ推論とその応用、第11回日本ロボット学会学術講演会予稿集、pp.131-132, (1993)
- [29] 山田、薮田、興野、ジェネティックアルゴリズムを用いたニューラルサーボコントローラの一考察、計測自動制御学会第32回学術講演予稿集、pp.533-534, (1993)
- [30] T.Yabuta, T.Yamada, "Possibility of Neural Networks Controller for Robot Manipulators", Proc. IEEE Int. Conf. on Robotics and Automation, pp.1686-1691,(1990)
- [31] 柴田、福田、塩谷、鴫田、光岡、"神経回路モデルによるロボット・マニピュレータの位置・力制御の研究（第5報、階層的ハイブリッドニューロモーフィック制御システム）", 日本機械学会論文集, 57巻540号 C編, pp.2659-2666 (1991)
- [32] 柴田、福田、小菅、新井、鴫田、光岡、"神経回路モデルによるロボット・マニピュレータの位置・力制御の研究（第6報、ニューロモーフィック制御とシンボリック制御による階層的知的制御システム）", 日本機械学会論文集、58巻549号 C編, pp.1442-1449 (1992)
- [33] 鴫田、光岡、福田、橋本、神経回路モデルによるロボットの位置/力ハイブリッド制御（マニピュレータによるグライディング作業）、第30回計測自動制御学会学術講演会、pp.125-126 (1991)
- [34] R.S.Sutton, Learning to predict by the methods of temporal differences, Machine Learning, Vol.3, pp.9-44 (1988)

- [35] C.J.C.H.Watkins, P.Dayan, Technical Note: Q-Learning, Machine Learning, Vol.8, pp.279-292 (1992)
- [36] 例えば、高瀬、比留川、微細動作の実現、日本ロボット学会誌、Vol.11, No.2, pp.206-211 (1993)
- [37] 北垣、内山、外部環境に対するマニピュレータの最適接近速度、日本ロボット学会誌、Vol.8, No.4, pp.413-420 (1990)
- [38] B.H.Yang, H.Asada, A New Approach of Adaptive Reinforcement Learning, Proc. Int. Joint Conf. on Neural Networks(IJCNN, Nagoya), pp.627-630 (1993)
- [39] M.I. Jordan and D.E. Rumelhart, Forward Models: Supervised Learning with a Distal Teacher, Cognitive Science, 16 (1992)
- [40] James S. Albus, Brain, Behaviour and Robotics, BYTE Books pp.139-179 (1981)
- [41] 正司、稲葉、福田、細貝、衝突を含むマニピュレータの安定制御 (第1報: 1本の1自由度マニピュレータの押し付け制御の安定性、機論、56-527, C, pp.1847-1853 (1990)
- [42] Long-Ji Lin, Self-Improving Reactive Agents Based On Reinforcement Learning, Planning and Teaching, Machine Learning, Vol.8, No.3/4, pp.293-321 (1992)
- [43] H.C.An, J. M. Hollerbach, Dynamic Stability Issues in Force Control of Manipulators, Proc. of IEEE Int. Conf. on Robotics and Automation, pp.890-896 (1987)
- [44] 吉川、梅野、対象物体ダイナミクスを考慮した動的ハイブリッド制御、日本ロボット学会誌 Vo.11, No.8, pp.1229-1235 (1993)
- [45] 辻、知的ロボティクスの幾つかの問題、日本ロボット学会誌 Vo.9, No.1, pp.79-84 (1991)
- [46] I.D. ランダウ、富塚、適応制御の理論と実際、オーム社 (1981)
- [47] K. J. Åström, B. Wittenmark, Adaptive Control, Addison-Wesley (1989)
- [48] 富塚、適応制御の実際と適用限界、計測と制御、Vol.29, No.8, pp.709-715 (1990)

- [49] Albus, A New Approach to Manipulatoe Control : The Cerebellar Model Articulation Control (CMAC), Trans. of ASME,Journal of Dynamics Systems, Measurement and Control, pp220-227 (1975)
- [50] Z. Ahmad, A. Guez, On the Solution to the Inverce Kinematic Problem, IEEE Proc. of Int. Conf. on Robotic and Automation, vol.3, pp.1692-1697 (1990)
- [51] 橋本、久保田、佐藤、原島、視覚情報によるマニピュレータの位置・姿勢制御 (ニューラルネットワークの利用) 、日本ロボット学会誌、Vol.8, No.4, pp.330-396 (1990)
- [52] S. Lui, H. Asada, Transferring Manupulative Skills to Robots : Representation and Aequisition of Tool Manupulative Skills Using a Process Dynamics Model, Trans. of ASME,Journal of Dynamics Systems, Measurement and Control, Vol.114, pp.220-228 (1992)
- [53] N. Hogan, Impedance Control Part1-Part3, Trans. of ASME,Journal of Dynamics Systems, Measurement and Control, Vol.107, pp.1-24 (1985)
- [54] 五味、川人、生体運動制御系における内部モデルを利用した運動司令の生成、第4回インテリジェント・システム・シンポジウム講演論文集、pp.155-160 (1994)
- [55] 柏木、バリ取り、グラインダがけ、日本ロボット学会誌、Vol.9, No.3, pp.155-160 (1991)
- [56] 比留川、多面体間の接触による拘束条件を満たす微小変位を幾何モデルから求める方法、日本ロボット学会学術講演会予稿集、 pp.525-528 (1989)
- [57] J. Barroquard, J-C.Latomb, A Monte-Carlo Algorithm for Path Planning with Many Degrees of Freedom, IEEE Int. Conf. Robotics and Automation, pp.1712-1717(1990)
- [58] E.H.Mamdani, S.Assilian : Am Experiment in Linguistic Synthesis with a Fuzzy Logic Controller, Int. J. Man-Machine Studies, Vol.7, pp/1-3(1974)

- [59] 廣田、ファジイ制御について、日本ロボット学会誌、Vol.9, No.2, pp.232-237 (1991)
- [60] 下島、福田、鴫田、教師無し学習型RBFファジイ・ニューロによる力制御、第4回インテリジェント・システム・シンポジウム講演論文集、pp.249-252 (1994)
- [61] 福田、柴田、インテリジェントシステムのためのニューロ・ファジイ・GAの統合・融合化技術の展望、機論、59巻564号、C編、pp.1442-1449 (1992)
- [62] 末広、高瀬、スキルの基づくマニピュレーションシステム、日本ロボット学会誌、Vol.8, No.5, pp.551-562 (1990)
- [63] 山下、島、石動、ニューラルネットワークによる学習、適応制御。計測と制御、Vol.30, No.4, pp.302-308 (1991)
- [64] 浅田、西川、欧米における次世代ロボットの動向、日本ロボット学会誌、Vol.11, No.1, pp.14-19 (1993)

謝辞

本研究は、研究開始当初は東京理科大学工学部機械工学科において、さらに、名古屋大学工学部機械情報システム工学科において、名古屋大学工学部機械情報システム工学科の福田敏男教授（研究開始当時は、東京理科大学工学部機械工学科助教授）のご指導のもとに行われたものであります。本論文を結ぶにあたり、研究の機会を与えていただき、終始、懇切なるご指導とご鞭撻を賜った福田敏男教授に深心から感謝の意を表します。

また、本論文をまとめるにあたり、ご教示、ご指導を頂いた名古屋大学工学部機械工学科の太田博教授ならびに電子機械工学科の末松良一教授に心より感謝いたします。

さらに、研究当初より公私の両面にわたりご指導と励ましを賜った木更津工業高等専門学校電子制御工学科の光岡豊一教授に厚く感謝の意を表します。

また、研究を進めるにあたり、ご指導頂いた名古屋大学工学部機械情報システム工学科の小菅一弘助教授、大学院工学研究科マイクロシステム工学専攻の新井史人講師、下島康嗣助手、機械技術研究所ロボット工学部バイオロボティクス研究室の柴田崇徳氏に厚く感謝の意を表します。そして、いろいろな面から支援して下さった名古屋大学工学部福田研究室の方々に感謝致します。

本研究を行うに当たってその機会を与えて下さった木更津工業高等専門学校電子制御工学科主任の高上輝雄教授をはじめ諸先生方、技官の方々に深く感謝致します。

最後に研究を続けるにあたり常に心身両面から支えてくれた家族や大勢の友人に深く感謝いたします。

付 録

本研究に関連した発表論文

I 学会誌等

1. 鴫田、光岡、福田、栗原、” 神経回路モデルによるロボットの力制御（1自由度マニピレータの制御）”、日本ロボット学会誌、7巻1号, pp.47-51（1989）
2. 鴫田、光岡、福田、栗原、” 神経回路モデルによるロボットマニピュレータの力制御（1自由度マニピュレータの実験と評価）”、日本ロボット学会誌、8巻3号, pp.292-299（1990）
3. 福田、栗原、鴫田、光岡、” 神経回路網によるロボット・マニピュレータの位置・力制御の研究（第1報、ニューラル・サーボ・コントローラのスタビリング制御への応用）”、日本機械学会論文集、56巻527号 C編, pp.1854-1860（1990）
4. M.Tokita, T.Mituoka, T.Fukuda, T.Kurihara, "Force Control of Robot Manipulator by Neural Network Model, Experiment and Evaluation of One-Degree-of-Freedom Manipulator", Journal of Robotics and Mechatronics, Vol.2, No.4, pp.273-281 (1990)
5. M.Tokita, T.Mituoka, T.Fukuda, T.Kurihara, "Force Control of Robotic Manipulator by Application of a Neural Network", Journal of Advanced Robotics, Vol.5, No.1, pp. 15 - 24 (1991)
6. M.Tokita, T.Mituoka, T.Fukuda, T.Shibata, F.Arai, "Position and Force Hybrid Control of a Robotic Manipulator by Application of a Neural Network", Mathematical Modeling and Scientific Computing, Vol.1, No.3-4, pp.235-246 (1993)

7. T.Shibata, T.Fukuda, S.Shiotani, T.Mituoka, M.Tokita, "Hierarchical Hybrid Neuromorphic Control System for Robotic Manipulators", JSME International Journal Vol.36, No.1, pp.100-109 (1993)

8. 鵜田、福田、” 神経回路モデルによるロボットの力制御（追加学習型ニューラルネットワークを用いた1自由度マニピュレータの適応力制御）”、日本ロボット学会誌（投稿中）

9. 鵜田、福田、光岡、” 神経回路モデルによるロボットマニピュレータの位置・力制御の研究（第7報、強化学習による1自由度マニピュレータの最適接近速度の学習）”、日本機械学会論文集 c 編（投稿中）

II 国際会議

1. T.Shibata, T.Fukuda, K.Kosuge, F.Arai, M.Tokita, T.Mituoka, "Skill Based Control by Using Fuzzy Neural for Hierarchical Intelligent Control", Proc. Int'l Joint Conf. on Neural Networks (IJCNN), Vol.2, pp.81-86 (1992)

III その他

1. 鵜田、光岡、福田、柴田、” 神経回路モデルによるロボットマニピュレータの位置と力のハイブリッド制御（2自由度マニピュレータの適応制御）”、日本ロボット学会誌、9巻1号, pp.45-54 (1991)

2. 福田、柴田、鵜田、光岡、” 神経回路モデルによるロボットマニピュレータの位置・力制御の研究（第2報、Time Delay形N. N. の位置と力のハイブリッド制御への応用）”、日本機械学会論文集、57巻535号 C編, pp.866-873 (1991)

3. 福田、柴田、鵜田、光岡、” 神経回路モデルによるロボットマニピュレータの位置・力制御の研究（第3報、ファジイルールを適応した位置と力のハイブリッド制御への応用）”、日本機械学会論文集、57巻535号 C編, pp.874-881 (1991)

4. 柴田、福田、新井、和田、鵜田、光岡、” 神経回路モデルによるロボットマニピュレータの位置・力制御の研究（第4報、衝突を考慮したセンシングとハイブリッド制御）”、日本機械学会論文集、57巻539号 C編, pp.2305-2312 (1991)

5. T.Fukuda, T.Kurihara, T.Shibata, M.Tokita, T.Mituoka, "Application of Neural Network-Based Servo Controller to Position, Force and Stabbing Control by Robotic Manipulator, JSME International Journal, Vol.34, No.2, pp.303-309(1991)

6. 柴田、福田、塩谷、鵜田、光岡、” 神経回路モデルによるロボットマニピュレータの位置・力制御の研究（第5報、階層的ハイブリッドニューロモーフィック制御システム）”、日本機械学会論文集、57巻540号 C編, pp.2659-2666 (1991)

7. 柴田、福田、小菅、新井、鵜田、光岡、” 神経回路モデルによるロボットマニピュレータの位置・力制御の研究（第6報、ニューロモーフィック制御とシンボリック制御による階層的知的制御システム）”、日本機械学会論文集、58巻549号 C編, pp.1442-1449 (1992)

8. T.Shibata, T.Fukuda, K.Kosuge, F.Arai, M.Tokita, T.Mituoka, "Hybrid Symbolic and Neuromorphic Control for Hierarchical Interigent Control of Robotic Manipulators", Nonlinear Phenomena in Electromagnetic Fields, pp.29-32 (1992)

9. T.Fukuda, T.Shibata, M.Tokita, T.Mituoka, "Neuromorphic Control (Adaptation and Learning)", IEEE Transactions on Industrial Electronics, Vol.39, No.6, pp.497-503 (1992)

10. T.Fukuda, T.Shibata, K.Kosuge, F.Arai, M.Tokita, T.Mituoka, "Sensing and Control for Robotic Motions by Neural Network - Position, Force, and Impact Control of Robotic Manipulator", Mathematical Modeling and Scientific Computing, Vol.1, No.3-4, pp.247-262 (1993)