

報告番号 乙 第 5175 号

ファジィ・ニューラルネットワークを利用した
知的ロボットマニピュレータの研究

木 口 量 夫

①

目次

ファジィ・ニューラルネットワークを利用した 知的ロボットマニピュレータの研究

木口量夫

目次

第1章 緒論	1
1.1 研究の背景	1
1.2 従来の研究	5
1.3 本研究の目的	17
1.4 本研究の構成	20
第2章 ニューラルネットワークを利用したロボットマニピュレータの 未知環境に対する位置/力ハイブリッド制御	23
2.1 緒言	23
2.2 ハイブリッド制御	24
2.3 ニューラルネットワーク制御器	27
2.4 ニューラルネットワークの構成	29
2.5 実験	31
2.5.1 実験装置	31
2.5.2 実験結果	34
2.6 結言	38
第3章 ファジィ・ニューロを利用したロボットマニピュレータの 未知環境に対する位置/力ハイブリッド制御	40
3.1 緒言	40
3.2 位置/力のハイブリッド制御	42
3.3 未知環境での力制御	44
3.3.1 ファジィ・ニューロ力制御器	44
3.3.2 入力値調整ニューラルネットワーク (IANN)	47
3.4 未知環境での位置制御	50
3.4.1 軌道制御用ファジィ・ニューラルネットワーク	50
3.4.2 摩擦補償用ニューロン	53
3.5 ファジィ・ニューラルネットワークの効果的学習	54
3.6 シミュレーション	55
3.6.1 FNN力制御器の有効性	56

3. 6. 2	IANNの有効性	59
3. 6. 3	FNN位置制御器の有効性	65
3. 6. 4	効果的学習法の有効性	67
3. 7	結 言	71
第4章	ファジィ・ニューロを利用したロボットマニピュレータの インテリジェント位置/力制御	72
4. 1	緒 言	72
4. 2	ハイブリッド位置/力制御	74
4. 3	未知拘束面に対する力制御	74
4. 3. 1	ファジィベクトル	76
4. 3. 2	ファジィベクトルを利用した 力制御方向変更アルゴリズム	77
4. 4	インテリジェント力制御	80
4. 4. 1	ファジィ・ニューロ力制御器	81
4. 4. 2	ファジィ環境評価器	83
4. 5	シミュレーション	85
4. 5. 1	ファジィベクトルを利用した力制御方向変更	86
4. 5. 2	特性が未知な環境に対する力制御	92
4. 6	結 言	92
第5章	ファジィ・ニューロを利用した複腕ロボットによる 未知対象物の操り	93
5. 1	緒 言	93
5. 2	対象物の操り	95
5. 3	制御器の構成	96
5. 4	未知対象物の制御	96
5. 4. 1	対象物ファジィ・ニューラルネットワーク制御	98
5. 4. 2	ファジィ・ニューラルネットワーク学習則	101
5. 5	マニピュレータの制御	101
5. 5. 1	ファジィ・ニューラルネットワーク力制御	102
5. 5. 2	標準環境モデルによる補償	103
5. 6	シミュレーション	105
5. 6. 1	未知対象物の操りシミュレーション	105
5. 6. 2	外乱下の対象物の操りシミュレーション	109

5. 7 結 言	112
第6章 ファジィ・ニューロを利用した複腕ロボットによる 摩擦補償を考慮した対象物の滑らせ移動制御	113
6. 1 緒 言	113
6. 2 対象物の軌道制御	114
6. 3 制御器の構成	115
6. 3. 1 摩擦補償用ニューロン	117
6. 3. 2 対象物操り用ファジィ・ニューラルネットワーク	119
6. 3. 3 学習則	120
6. 4 学習係数	122
6. 5 マニピュレータの制御	123
6. 5. 1 ファジィ・ニューラルネットワーク力制御	124
6. 6 シミュレーション	124
6. 7 結 言	128
第7章 バリ取り作業への応用	129
7. 1 緒 言	129
7. 2 ファジィ・ニューラルネットワーク (FNN) による 知識と業の獲得	132
7. 3 ロボットマニピュレータの制御 (下位レベル制御)	132
7. 4 知的作業プランナー	134
7. 5 ファジィ対象物特性評価器	135
7. 6 ファジィ・ニューロ作業プランナー	136
7. 6. 1 ファジィ評価関数	136
7. 7 シミュレーション	137
7. 8 結 言	144
第8章 結 論	145
8. 1 総まとめ	145
8. 2 今後の課題	147
謝 辞	148
参考文献	149

第1章 緒 論

1. 1 研究の背景

ロボットを用いて作業の効率化を図ったり、あるいは人間では作業が難しい環境下での作業を実現させたりすることは今後ますます重要になる。これらを実現させるため、ドライバーやスパナを使ってネジやボルトを締めたり、グラインダーでバリ取りをしたり、あるいは物体を押さえたり移動させたりする基本的な作業動作をロボットで実現させることが期待されている。近年、ロボットは目覚ましい発展を遂げ、すでに種々の分野で活躍しているものも多いが、上記の様な基本的な作業動作を問題なくこなせるものはまだ見あらず、現在も多く研究者により研究が進められている。すでに活躍しているロボットの多くは組立工場などで稼働している産業用ロボットマニピュレータであるが、これらは限定された環境下で限定された動作のみを行うことだけに利用されている。また、これらのロボットは位置制御のみに限定されているため、3次元空間上で対象物を移動させたりすることしかできず、他のロボットや人間と協調した動作を実現させることも難しい。さらに、上記のような基本的作業動作を実現させるには位置と力を同時に制御することが必要であり、現在稼働している位置制御のみ可能なロボットによりこれらの作業動作を実現させるのは難しい。また、力制御の際には力を加える環境の動特性がシステム全体の動特性に影響するため、環境の動特性を十分把握しておく必要がある。しかし、一般的に環境の動特性を前もって把握するのは難しく、力制御の実用化を妨げる一因となっている。

また、位置と力を同時に制御する際の位置制御では、位置のみを制御する際には存在しなかった環境とロボット間の摩擦問題が生じる。この摩擦力は、環境に対し加える力の大きさにより変化するため補償が難しくなる。これもまた位置／力制御の実用化を妨げる一因となっている。

位置／力制御を実際の作業へ応用する場合には、作業を遂行するための目標力と目標位置（目標軌道）を設定する必要がある。しかしながら、普段は、人間の作業者が簡単にこなしている作業、たとえばバリ取り作業は、とくに目標力や目標位置（目標

軌道)が設定されているわけではなく、作業者自身の判断により行われている。したがって、この様な作業をロボットで実現させるためには、人間のように状況に応じて目標力や目標位置(目標軌道)が設定できるような、作業プランニングが必要となる。

複数のロボットマニピュレータにより対象物を操ることも、ロボットに期待されている重要かつ基本的な作業の一つである。そのため、ロボットによる対象物の操り法について、多くの研究 [1] ~ [8] がなされてきた。しかし、これらの研究もまた既知の対象物を扱うものがほとんどであり、重量等の情報が不確かなあるいは未知な対象物に対する制御法などはあまり見られない。

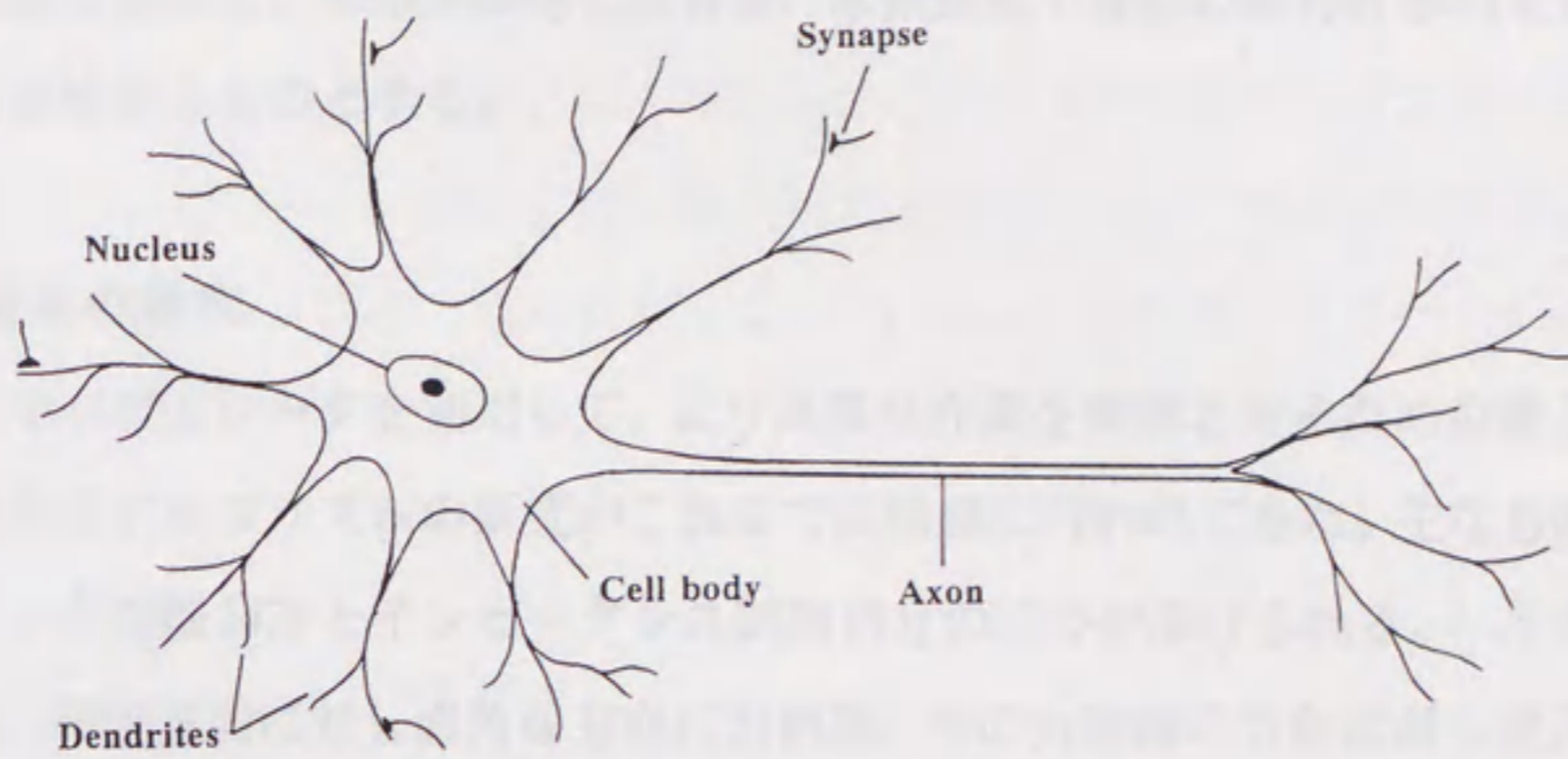
従来、ロボットは産業界への応用、とくに工場等での活用が主として考えられてきたが、最近では宇宙ロボット、海中ロボット、群知能ロボット、自律移動ロボット、医療・福祉ロボット、あるいはアミューズメントロボットなどの行動環境が限定されないロボットや人間と共存するロボットの活躍が期待されている。これらを実現させるには、ロボットは不確かなあるいは未知な環境下で作業を行ったり、不確かなあるいは未知な対象物を自由に操ったりできる能力が要求される。そのためにはロボット自身が学習・適応能力、知識の獲得能力、あるいは人間とのコミュニケーション能力等の高度な能力を獲得しなければならない。しかしながら、現在の段階ではこれらを可能にする確定した方法はいまだ定まっておらず、多くの研究者によって活発に研究が進められている。

一方、人間の言語のような曖昧な情報を扱うことができるようなファジィ理論が近年注目を集め、ロボット制御の分野でも積極的に応用されている。ファジィ理論は1965年に Zadeh [9] により提唱されたもので、1974年に Mamdani [10] によりスチームエンジンの制御に利用されて以来、工学の分野で注目を集めるようになり、現在では地下鉄の運転制御[11]や一般家庭で使われる電化用品にも応用されるようになってきた [12]。ファジィ制御では、ロボットや環境の詳細な数学モデルを作成する必要がなく、IF-THEN形式の曖昧な言語的表現で制御器を設計できる特徴を持っている。このファジィ制御を非線形のロボット制御に用いることにより、設計者あるいは作業者にとって理解し易い非線形制御が実現できる。しかし、ファジィ制御ルールを設計した後には、ファジィ制御器自身には学習/適応能力がないため、制御ルールの調整ができないということが短所となっている。

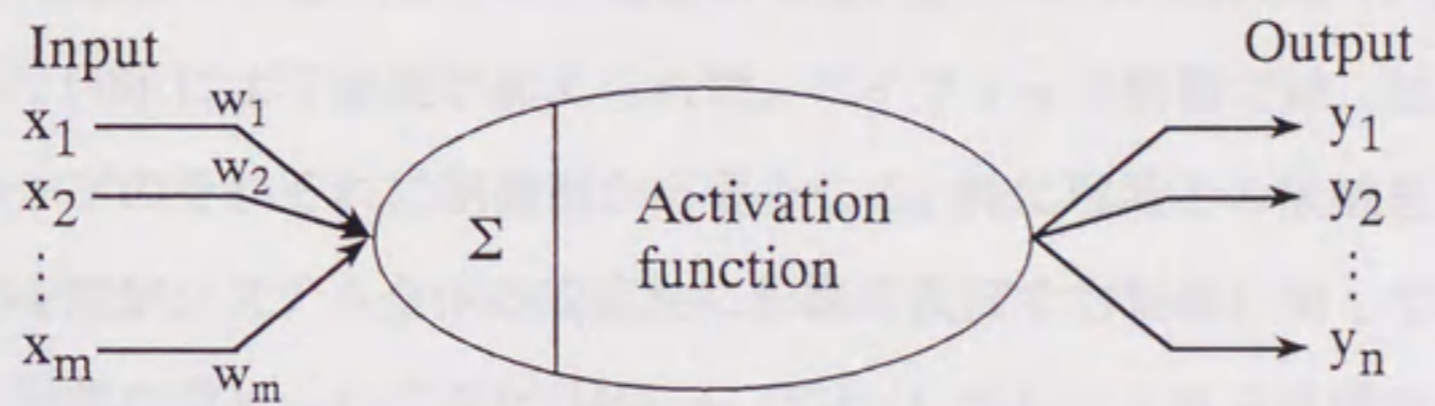
また一方では、生物の学習・適応能力を解明したり工学的に応用するため、ニューラルネットワークの研究が積極的に進められている。工学的なニューラルネットワークは1943年にMcCullochとPitts [13] がニューロンのモデル(図1.1参照)を提案したのを発端にし、1949年にはHebb [14] によりシナプス強化法則が提案され、1961年にはそれらをもとにしてRosenblattによりニューロンを層状に結合した構造のパーセプトロン [15] が提案され、現在よく用いられている多層ニューラルネットワークの原形となった。ニューラルネットワークの研究は、1969年出版されたMinskyとPapertによる著書 [16] で2層のパーセプトロンの限界が明らかにされた後しばらくの間は下火になっていたが、非線形ニューロンから成る多層ニューラルネットワークの学習に有効なバックプロパゲーション法(誤差逆伝播法) [17] ~ [19] が提案されて以来、再び脚光を浴びるようになった。非線形多層ニューラルネットワークでバックプロパゲーション法を用いて学習させることにより、任意の非線形マッピングの構築が可能となる。この特性を利用してロボット制御やロボットや環境の同定を行う研究が盛んに進められている [20] - [41]。これらの詳細については次節で説明する。しかしながら、ニューラルネットワークにより構築される非線形マッピングはブラックボックスであるため、作業にとって学習/適応したニューラルネットワークの各ウエイト値が示す意味が解りにくいという短所がある。

最近になって、これらのファジィ制御とニューラルネットワーク制御のお互いの利点を組み合わせたファジィ・ニューロ制御の研究が注目を集めるようになってきた [12]。このファジィ・ニューロ制御により、曖昧な情報の扱いができ、厳密な数学モデルも作成することなく制御でき、なおかつ制御ルールを自動的に調整したりするユーザーフレンドリーなインテリジェント制御を実現することが期待されている。

本研究は、従来は使用分野が限られていたロボットマニピュレータにファジィ理論やニューラルネットワークおよびそれらをあわせたファジィ・ニューラルネットワークを利用することで、人間の知識を利用したり、不確かなものあるいは未知のものに対し自ら学習・適応したり、あるいは作業を遂行する上で必要な知識を獲得できるようにロボットマニピュレータ自身を知的化するものとする。ロボットマニピュレータの知的化により、より広い分野で、より柔軟に、より効率良く活用されることを目指すものであり、ロボットマニピュレータによる未知環境に対する位置/力制御、未



(a) 生体ニューロン



(b) McCullochとPittsのニューロンモデル

図 1.1 ニューロンモデル

知対象物の操り,あるいは未知対象物に対する高度な作業を実現させることを目標とするものである。本論文において,未知環境とは形状,摩擦係数や材料の動特性が未知である環境を意味し,未知対象物とは質量,摩擦係数や材料の動特性等が未知である対象物を意味するものとする。

1. 2 従来の研究

ロボットマニピュレータを利用して,より高度な作業を実現させるため位置と力を同時に制御するアルゴリズムの研究がこれまでに活発に行われてきた。主なものとしてハイブリッド制御 [42] とインピーダンス制御 [43] の二つが挙げられる。ハイブリッド制御では,環境表面に対し直角な方向に力制御,その力制御の方向に対し直角方向に位置制御することにより,位置と力を同時に直接的に制御するのに対し,インピーダンス制御では速度と加速度も含めた位置と力の関係を制御するというのが主な違いである。位置と力を精度良く制御するには,位置と力を直接制御できるハイブリッド制御が有利であると思われる。

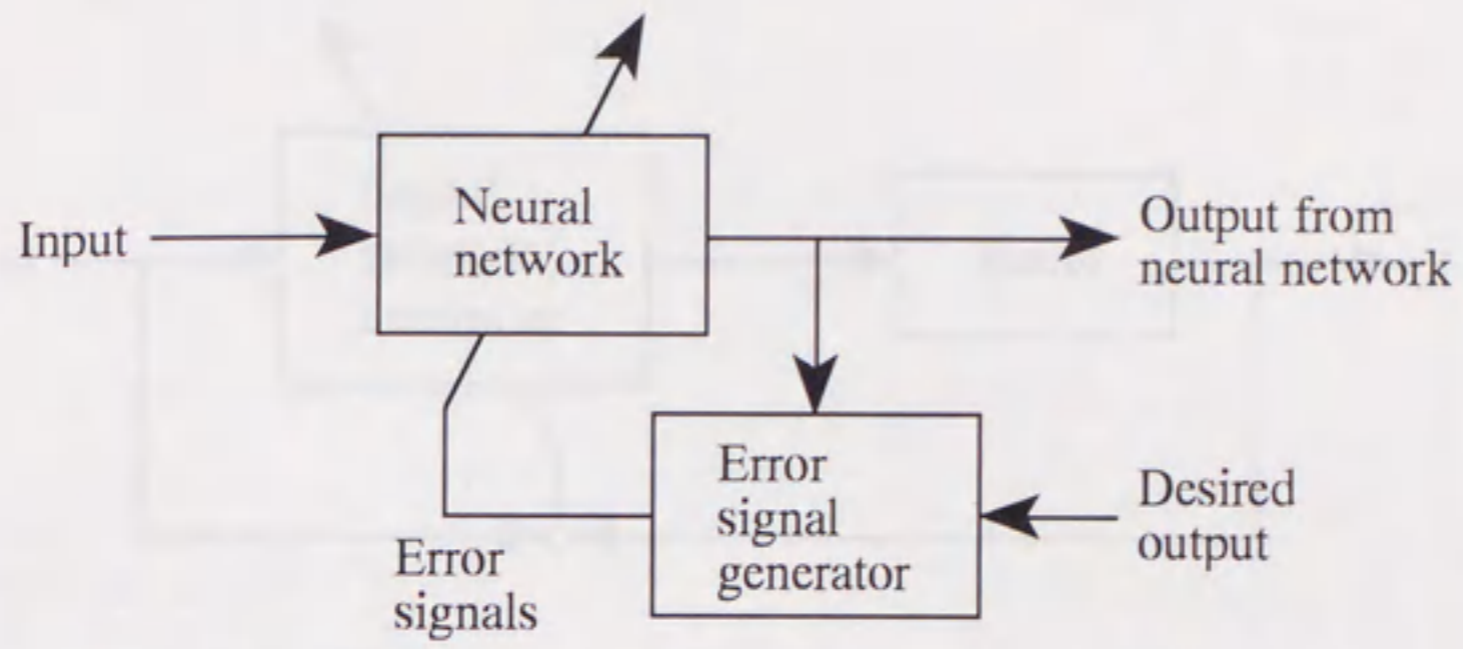
Raibert と Craig により提案されたハイブリッド制御 [42] は,ロボットマニピュレータの動特性を考慮していないなどの不備があったため,のちに吉川ら [44] ~ [46] や Khatib ら [47], [48] により改良が加えられた。ハイブリッド制御では,位置制御ループと力制御ループのそれぞれに制御則が必要となる。特に環境との接触を余儀なくされ,環境の動特性がシステム全体の安定性に影響を及ぼす力制御に関しては,多くの研究者により研究が進められてきた [49] ~ [52]。しかし,これらの研究のほとんどは対象となる環境の特性や形状が既知であることを前提としており,未知環境に対する力制御の研究例は少なかった。これまでの未知環境への力制御の研究では,環境の同定を行ったり [53],環境の動特性をも考慮してインピーダンス制御器のパラメータをニューラルネットワークを用いて変更するものが提案されている [54], [55]。この他にも,制御器を未知環境へ適応させるためにニューラルネットワークの応用が研究されている [20] - [41]。これらの研究例では,学習/適応能力をもつニューラルネットワークを応用することで制御器に柔軟性を持たせている。

ニューラルネットワークの学習則は,主なものとして教師あり学習,教師なし学習及び強化学習が挙げられる。教師あり学習は,ある入力データに対する目標出力値を

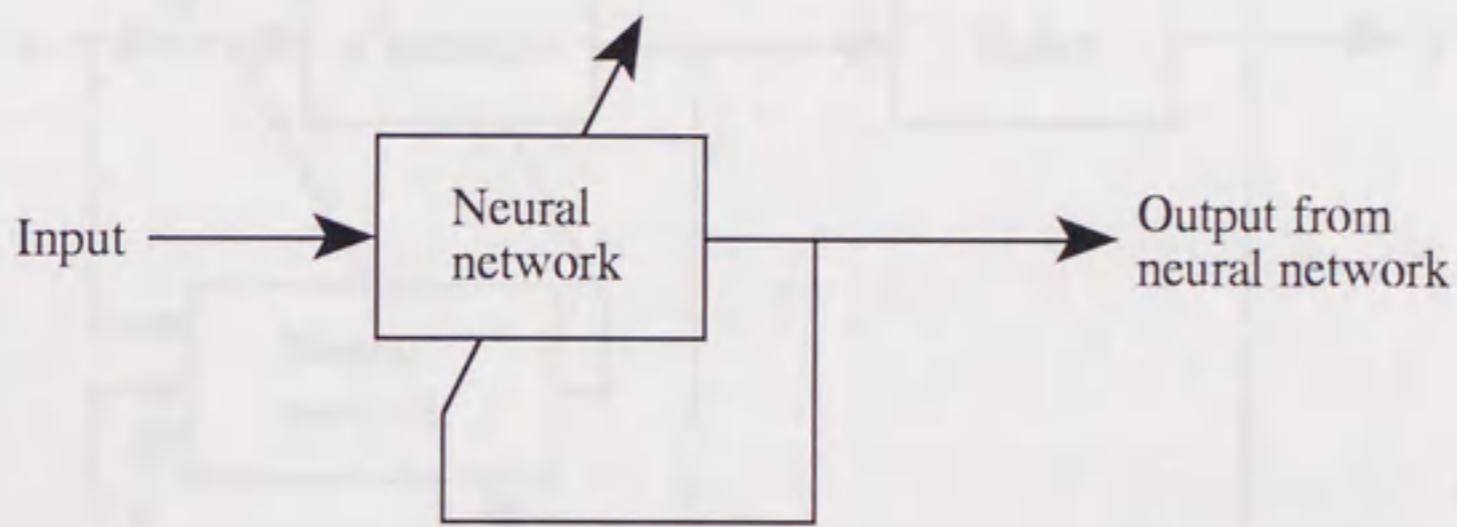
教師として与え、ニューラルネットワークからの出力誤差が小さくなるように各ウエイト値を変更する方法（図 1.2 (a) 参照）であり、それぞれの入力データに対する目標出力値がわかっていることが前提となっている。教師なし学習は、入力データの入力空間におけるパターンにより自動的にクラス分けを行うように各ウエイト値を変更する方法（図 1.2 (b) 参照）で、陽に目標状態を与える必要はない。また強化学習は、ある入力データに対する目標出力値はわからないが、ニューラルネットワークからの出力が良いか悪いかが評価できる場合、ニューラルネットワークからの出力を評価する教示信号に従い各ウエイト値を変更する方法（図 1.2 (c) 参照）である。

ニューラルネットワークをロボットマニピュレータ制御に用いる場合、一般的に学習則として教師あり学習の一つであるバックプロパゲーション法（誤差逆伝播法）が用いられている。バックプロパゲーション法は、局所解に陥ったり、学習が停滞したりする問題があり、最近では、GA（Genetic Algorithm）を用いて局所解に陥ることなくニューラルネットワークの学習を行う方法も提案されている [65]。しかし、制御の分野ではバックプロパゲーション法による学習／適応が主流となっている。バックプロパゲーション法において学習係数は、学習速度や安定性に関わる重要なファクターであり、これまでも適切な学習係数を見つけるために多くの研究がなされてきた。例えば、Jacobs らは、試行錯誤しながら調整するヒューリスティックな学習係数調整法を挙げたうえで、ニューラルネットワークの全てのウエイトがそれぞれ別個の学習係数を持つことを主張しており [66]、Arabshahi らは学習係数をファジィ制御で調整することを提案している [67]。学習係数をファジィ制御で調整する方法は効果的であるが、学習係数の有効範囲をあらかじめ把握しておく必要があると思われる。したがって、適切な学習係数の範囲が予測できない場合は、Jacobs が提案しているような、学習係数を線形的に増加させたり指数的に減少させたりする Delta-Bar-Delta 学習係数調整法 [66] が効果的であると思われる。

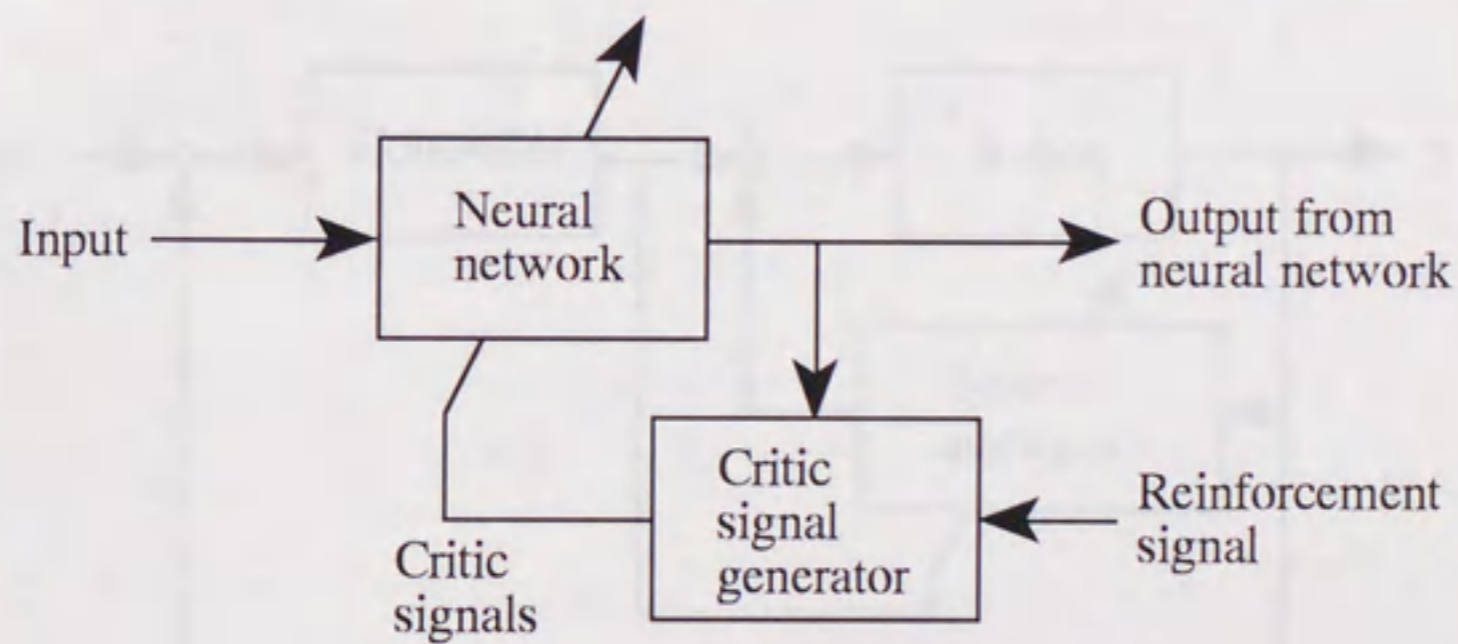
これまでに提案されているニューラルネットワーク制御器は、直接型（図 1.3 (a)）、間接型（図 1.3 (b)）、逆ダイナミクス補償型（図 1.3 (c)）の 3 種類に分類できる。間接的にニューラルネットワークをロボットマニピュレータ制御に利用した例では、あらかじめ学習させておいたニューラルネットワークにより PID 力制御器のゲインを環境の硬さに応じて調整する方法 [25] や、図 1.4 に示すような位置／力ハイブリッド制



(a) 教師あり学習

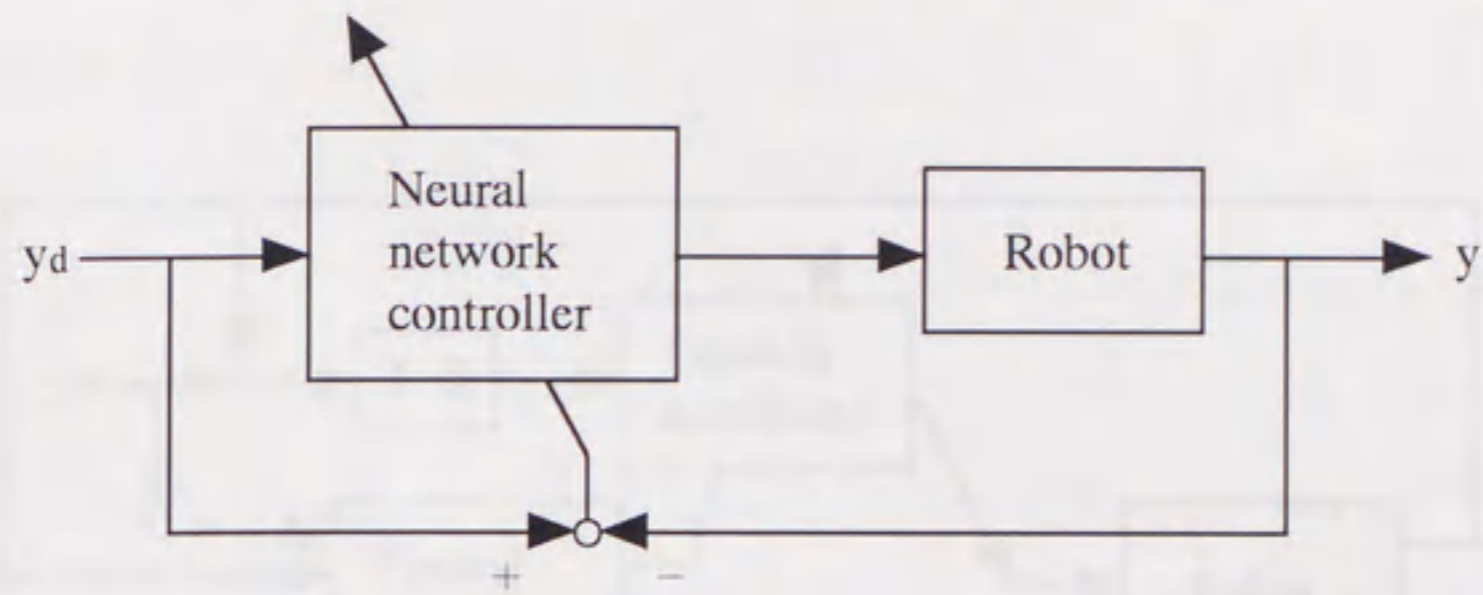


(b) 教師なし学習

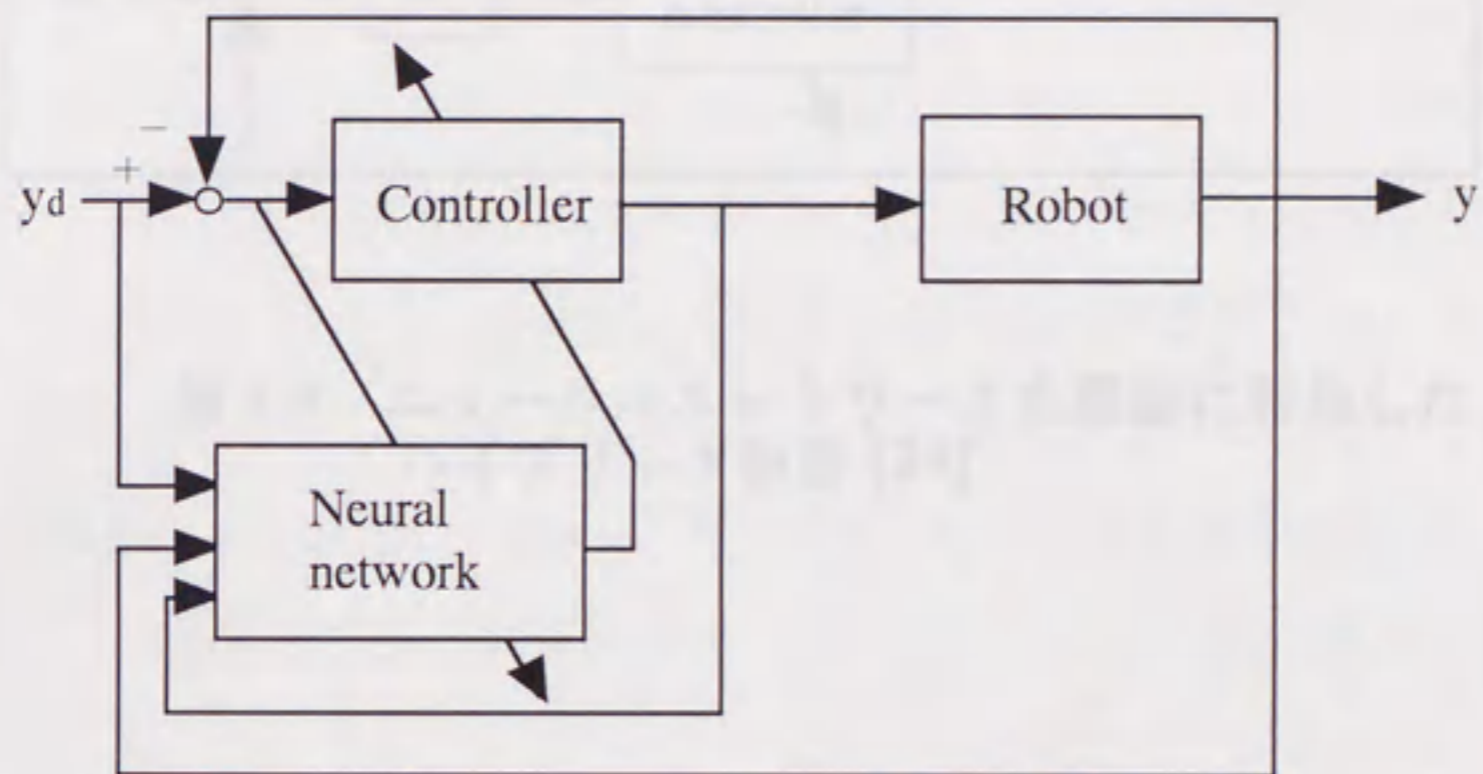


(c) 強化学習

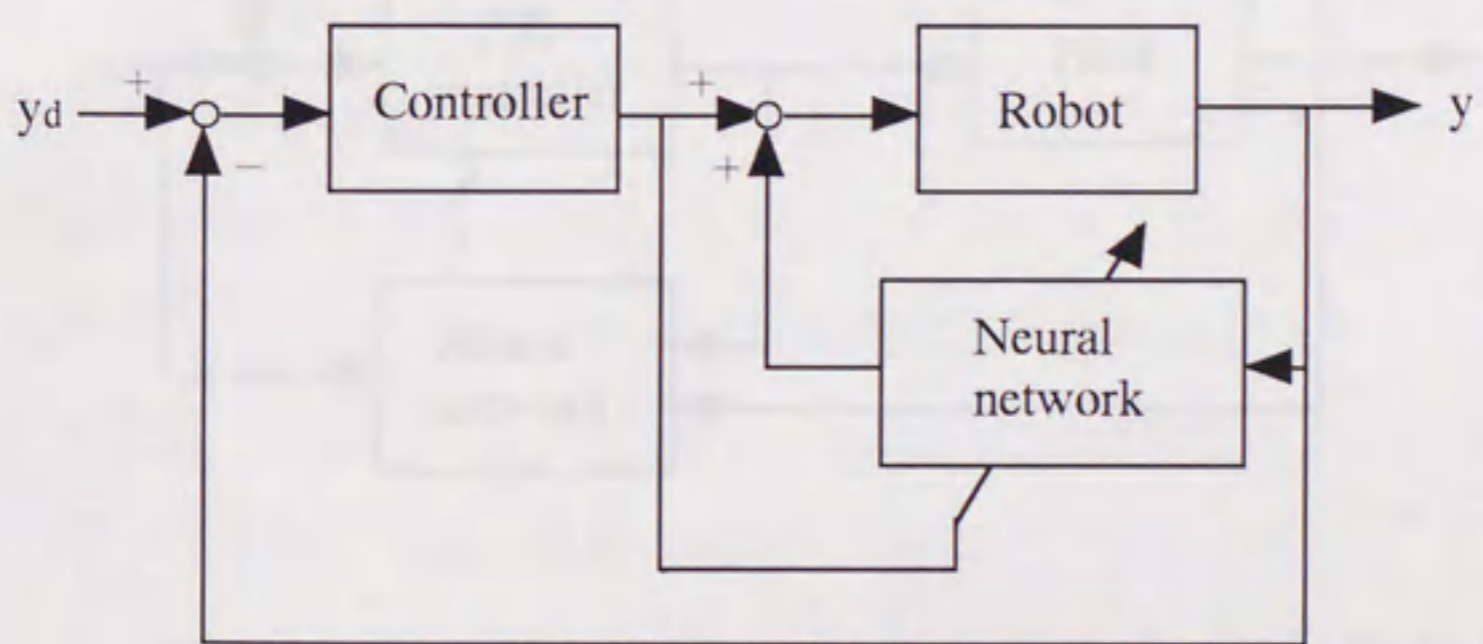
図 1.2 ニューラルネットワークの学習則



(a) 直接型ニューラルネットワーク制御器



(b) 間接型ニューラルネットワーク制御器



(c) 逆ダイナミクス補償型ニューラルネットワーク制御器

図 1.3 ニューラルネットワーク制御器の種類

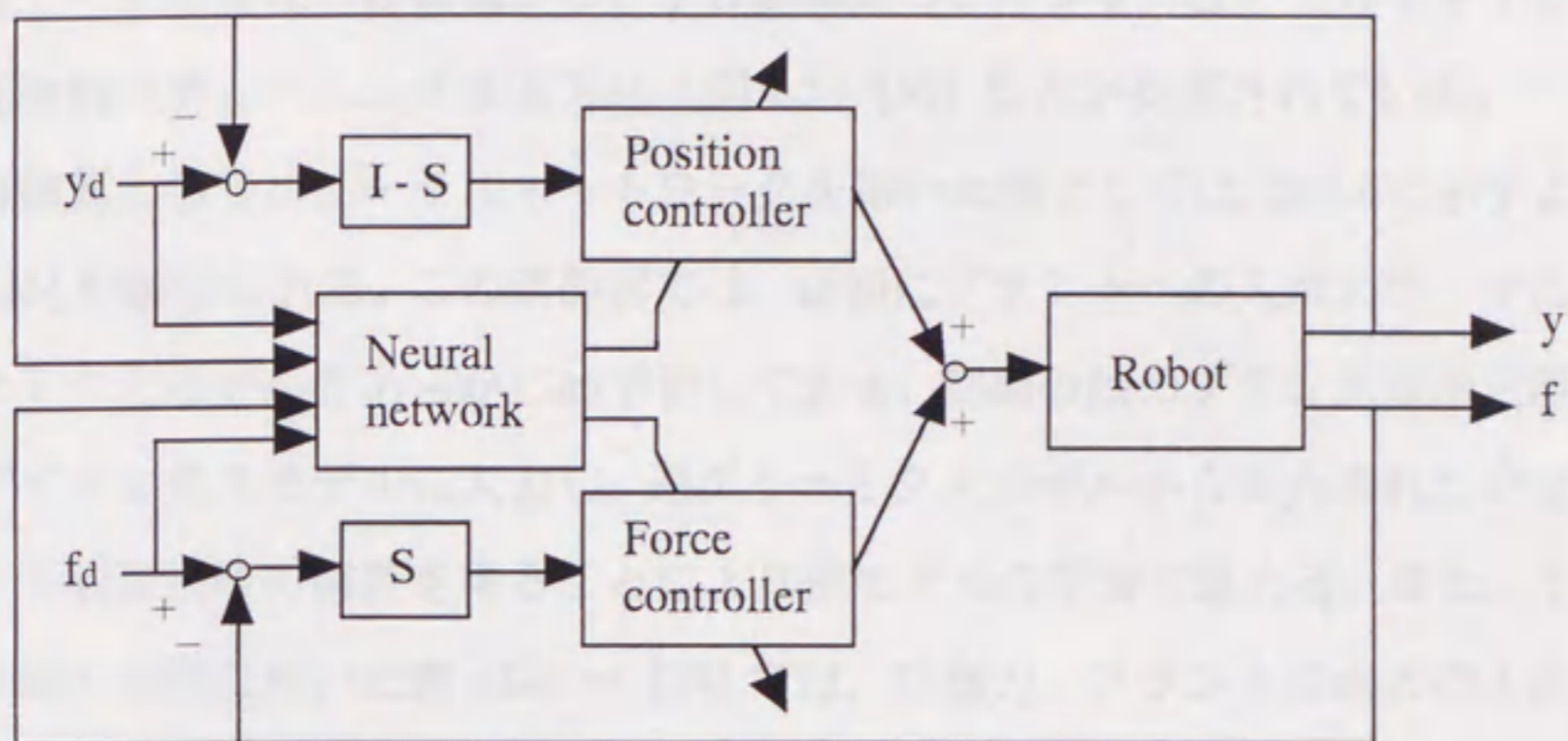


図 1.4 ニューラルネットワークを間接に利用したハイブリッド制御 [26]

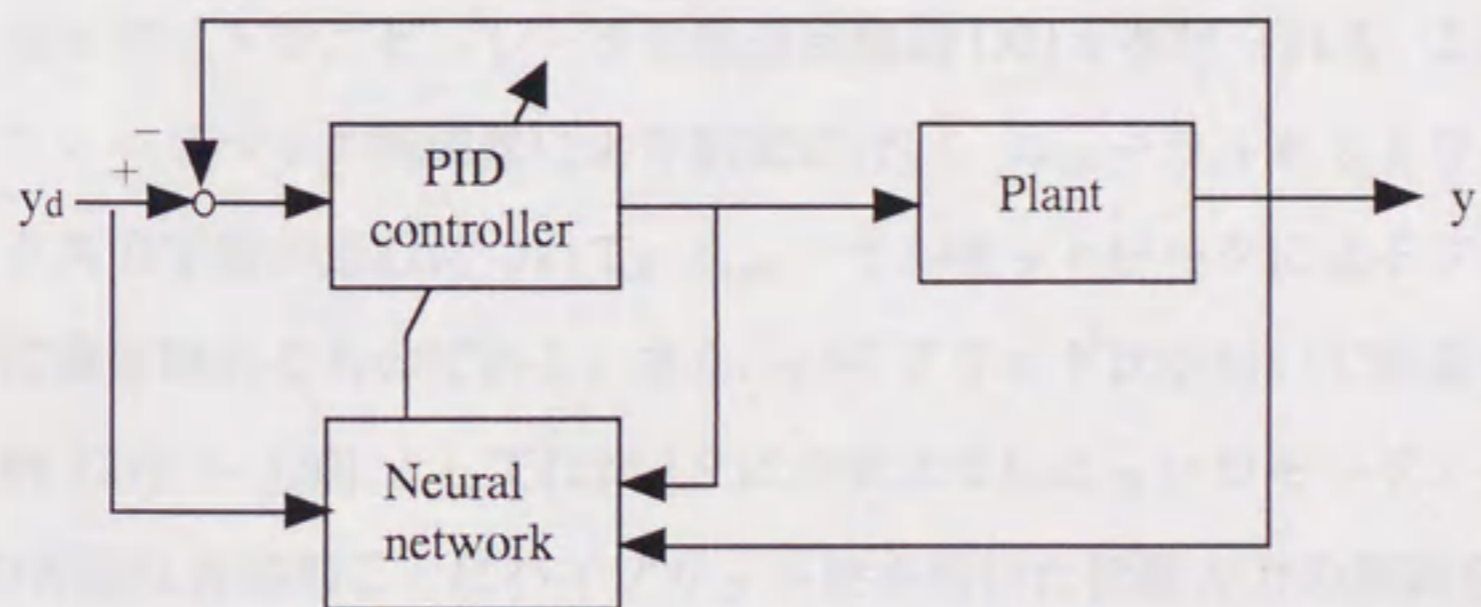


図 1.5 セルフチューニングニューロPID制御 [36]

御器（P I D位置制御則とP I力制御則）のゲインを環境の硬さに応じて調整することに利用する方法 [26] がある。また、一般的な制御器として、プラントに関する過去の入出力データと現在の目標値からP I D制御器のゲインをニューラルネットワークにより逐次的にチューニングする方法（図1.5） [36] などが提案されている。

直接制御器としてニューラルネットワークを用いた例としては、図1.6で示すような制御器 [35] が挙げられる。この制御器では、最初にプラントへの入出力データにより順ダイナミクスモデルをあらかじめ学習しておき、制御の際のプラントの出力誤差をその順ダイナミクスモデルに入力し、逆ダイナミクスモデルから出力されたフィードフォワード制御指令の誤差を得ることにより逆モデルの学習を進める。また、1自由度の力制御の研究に用いた例 [22] ~ [24] では、目標力、プラントの過去の入出力および現在のプラントからの出力を入力情報とし、プラントへの入力コマンドを出力とするニューラルネットワークを用い、プラントからの出力誤差をなくすようにニューラルネットワークの各ウェイト値を変更することでプラントの逆ダイナミクスを獲得している（図1.7）。この他にもロボットマニピュレータの位置制御に用いた例 [39] などがあるが、このタイプの制御器を位置/力ハイブリッド制御に用いた例は見あたらない。

また、逆ダイナミクスの補償にニューラルネットワークを用いた例としては、図1.8で示すようなロボットマニピュレータの軌道制御器 [32] が挙げられる。この制御器では、最初はフィードバック制御器により制御を行い、ニューラルネットワークによる逆ダイナミクスの学習が進むにつれて、ニューラルネットワークによるフィードフォワード制御に置き換わるものである。また、ハイブリッド比を用いて位置と力の制御に応用した例 [27] ~ [30] としては図1.9に示すようなニューロモーフィック制御器がある。この方法は各関節ごとにハイブリッド比を用いた位置と力の制御を行うため、ロボットの関節数が多くなると構造が複雑になる。

これまでにニューラルネットワークをロボットマニピュレータ制御に応用した研究例は軌道制御等に用いたもの [32] - [41] が多く、力制御に用いた例 [22] ~ [31] は少なかった。ロボットマニピュレータの力制御に用いた研究例でも、予備学習を必要とするものが多く、予備学習をすることができない未知対象物に対し多自由度ロボットマニピュレータで基準座標系における位置と力を直接的に制御する際に、ニューラ

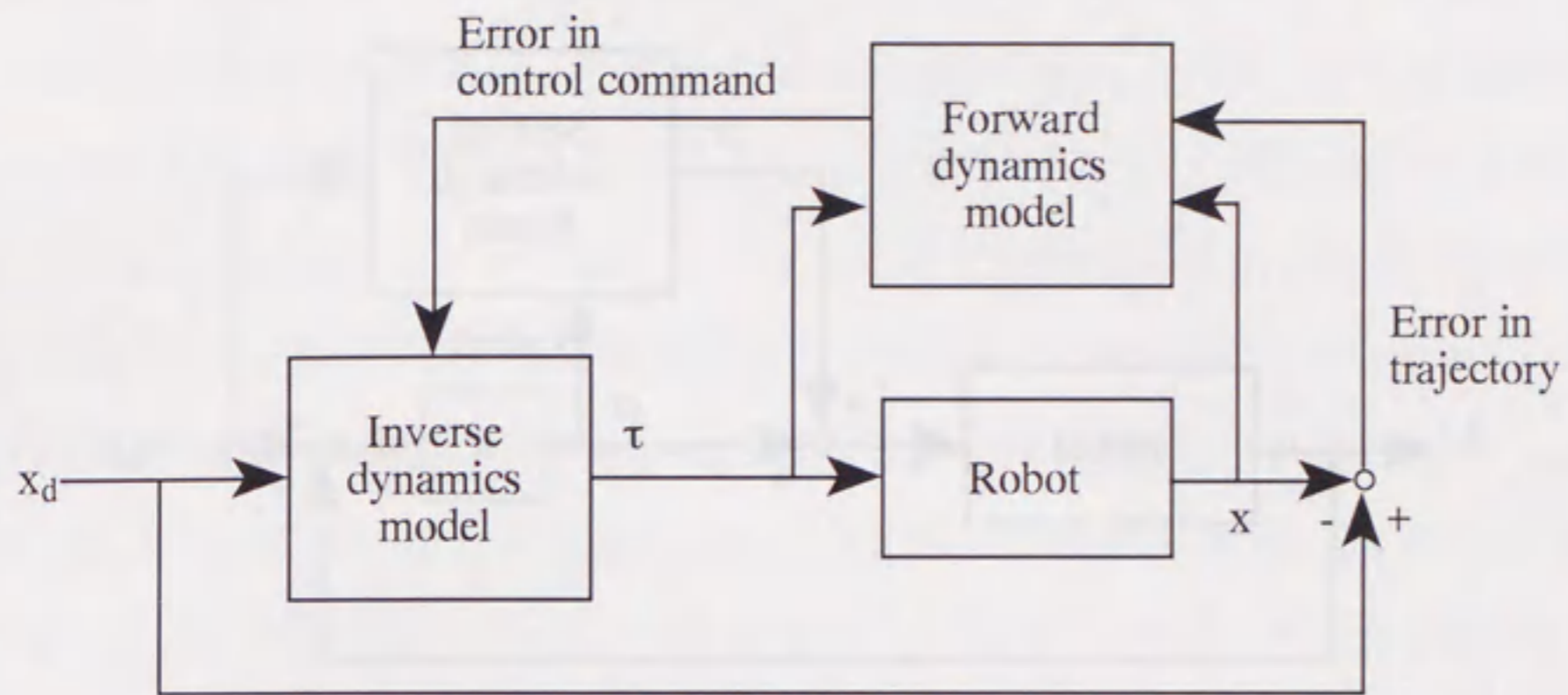


図 1.6 順逆ダイナミクスモデル制御 [35]

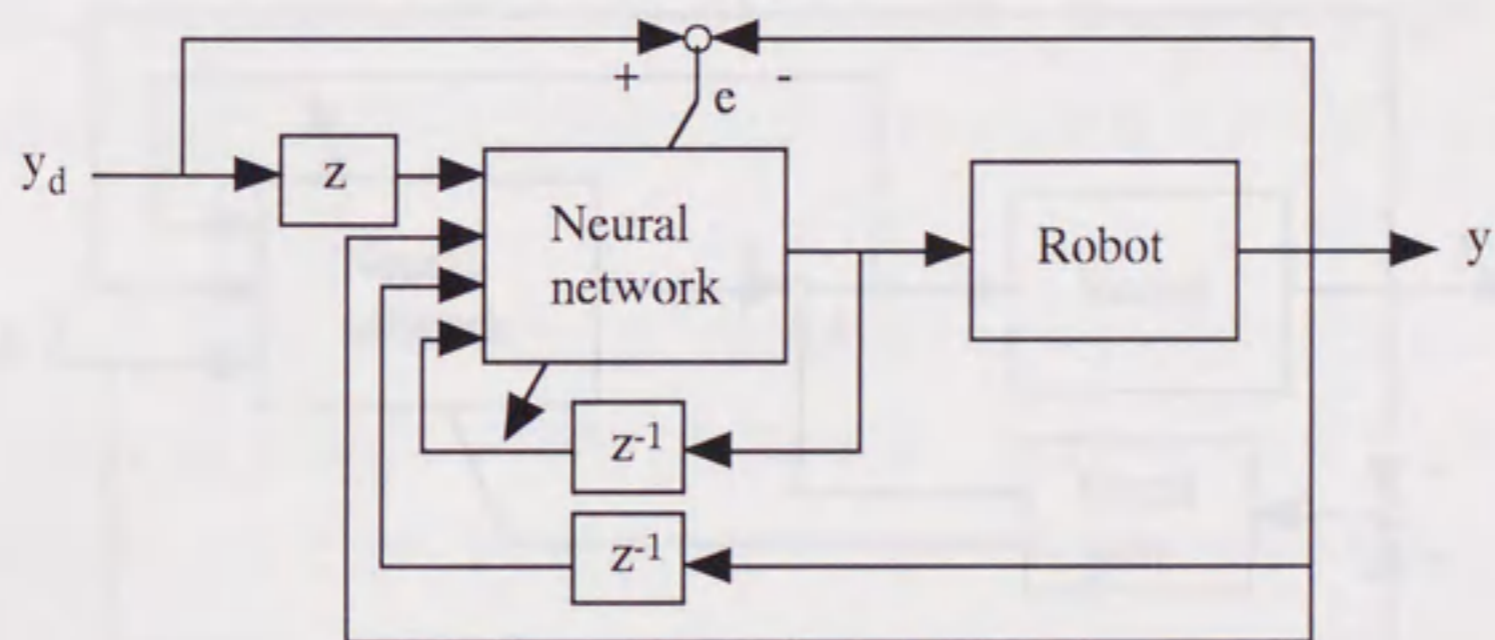


図 1.7 適応型ニューラルネットワーク力制御器 [22]~[24]

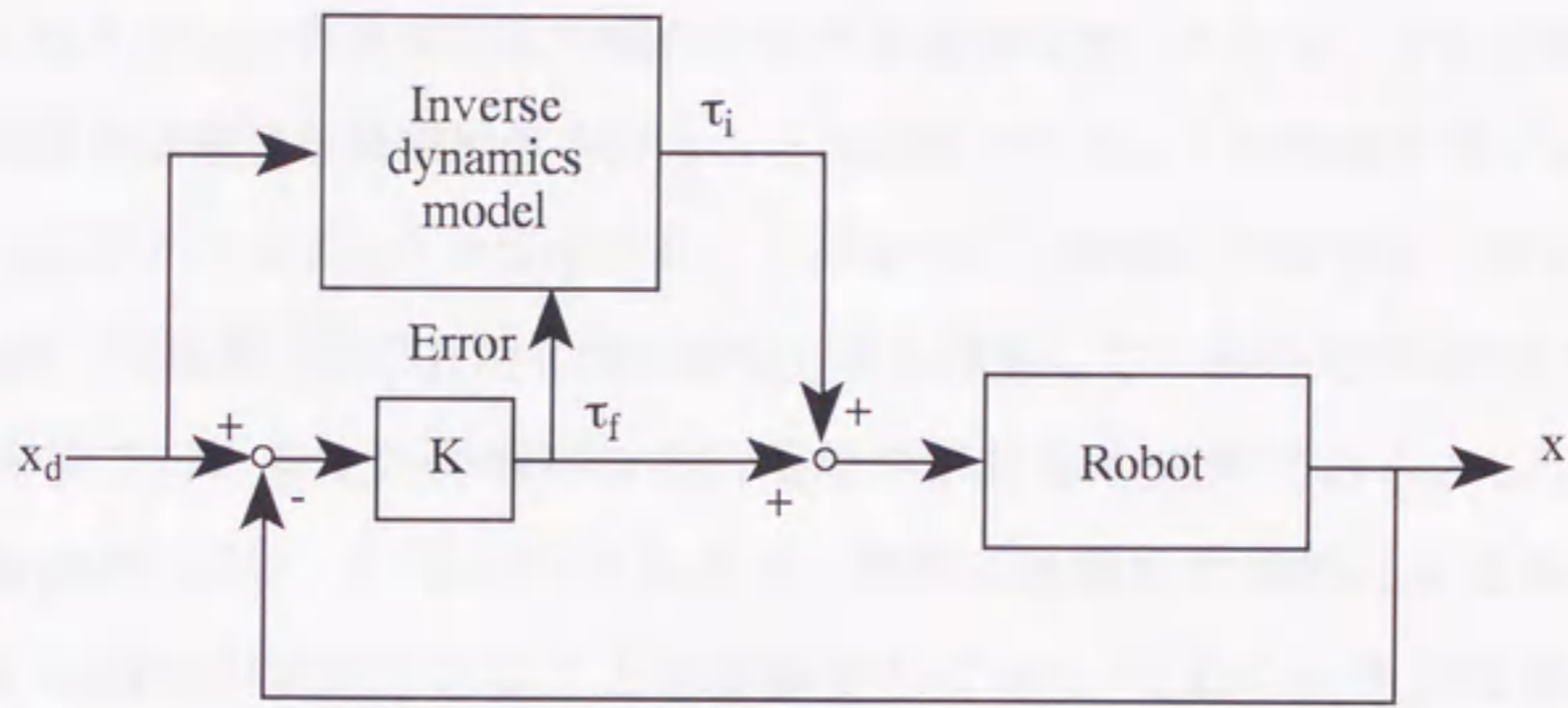


図 1.8 フィードバック誤差学習制御 [32]

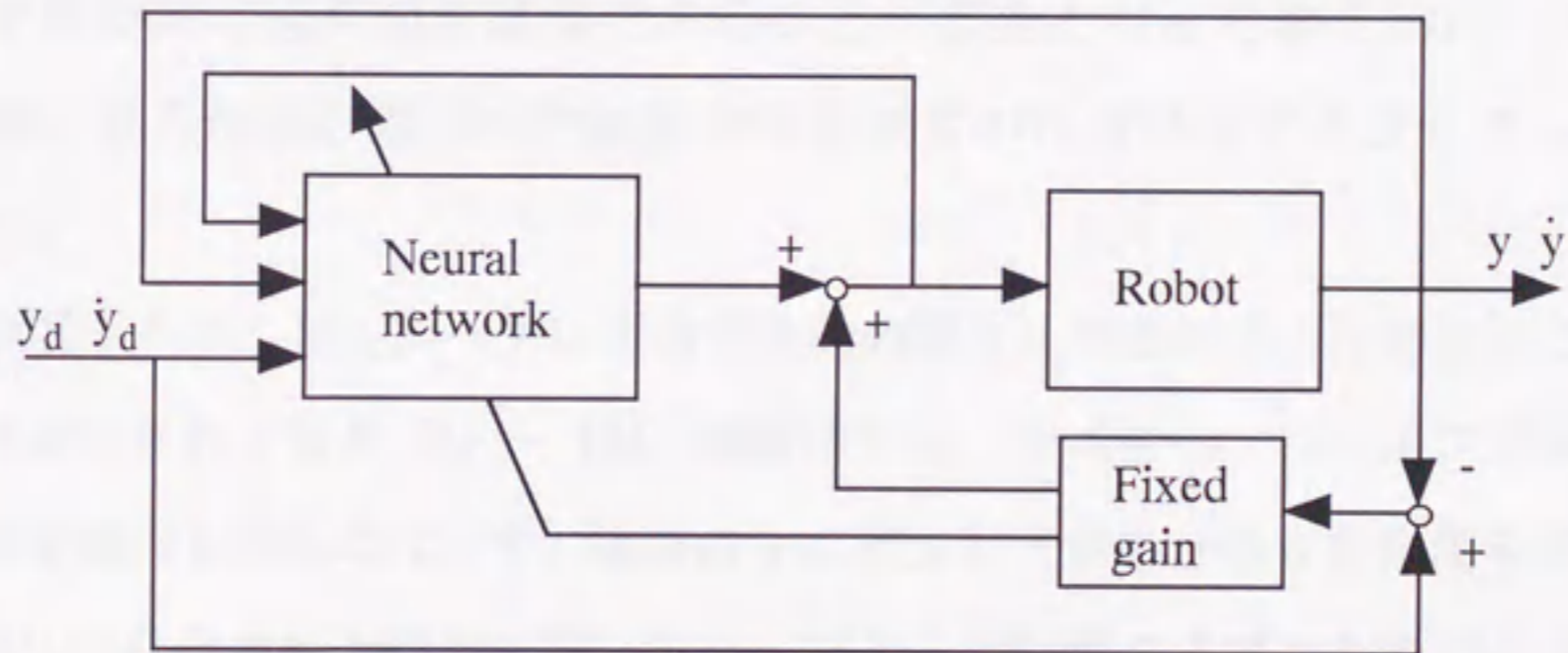


図 1.9 ニューロモーフィック制御 [27]~[30]

ルネットワークを直接型制御器として応用した例は見あたらない。

一方、位置／力のハイブリッド制御の位置制御ループにおいては、環境に力を加えながら位置を制御するため、環境とロボット間の摩擦問題が生じる。この摩擦力は、環境に対し加える力の大きさにより変化するため補償が難しくなる。とくに摩擦係数の未知な環境では適切な摩擦補償を行うことは困難である。この問題を避けるため、ロボットのエンド・エフェクタ先端にローラーを付けて実験した研究例 [56], [57] はあるが、位置／力制御の際の有効な摩擦補償方法を提案している研究例は見あたらない。

未知環境では、環境の動特性の他に環境の形状等も未知である。しかし、これまでの研究例 [42] ~ [57] のほとんどは、環境の動特性や環境による拘束面の傾きあるいは曲率が既知であることを前提としており、それらが未知な環境に対するハイブリッド位置／力制御の研究例は少ない。吉川ら [58] は力センサーから得られた力情報と未知拘束面上をロボットのエンドエフェクタがなぞる際にロボットが移動した位置情報により摩擦力による影響を除去した力制御方向を算出し、さらに力制御方向ベクトルと仮想拘束面の法線ベクトルから次の位置制御方向を算出する方法を提案している。また他の研究例 [59], [60] でも力センサーからの力情報により未知拘束面の法線方向を特定しているものが多いが、これらの研究では初期の拘束面が既知であることが前提となっており、力センサーからの力情報を得るための最初の力制御方向を探すのに有効なアルゴリズムは含まれていない。

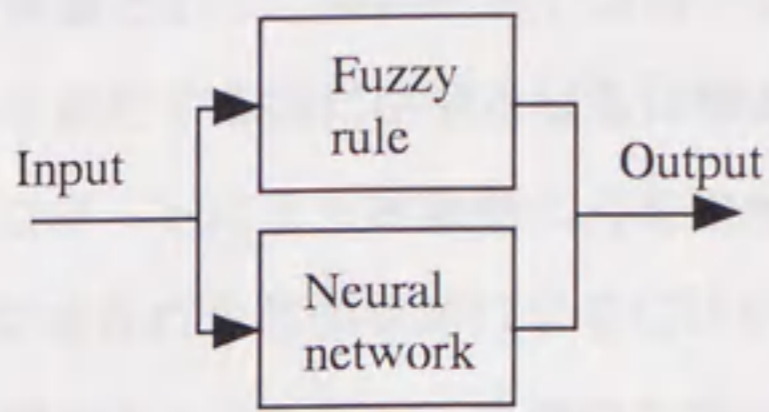
複数ロボットマニピュレータによる対象物の操り方や把持力の決定法についても多くの研究がなされてきた [1] ~ [8]。制御法では、マスター・スレイブマニピュレータの概念を利用したもの [5] や、複数のマニピュレータのうち6自由度を位置制御に用い、残りの自由度を力制御に用いるハイブリッド制御タイプのもの [6], [8] あるいは対象物を直接制御するもの [2] ~ [4] などがある。さらに、Ramadoraiら [7] はタスクレベルの制御法について議論している。マスター・スレイブマニピュレータを利用したタイプのもの [5] は、マスターマニピュレータが対象物の位置制御を行いスレイブマニピュレータが対象物への力制御を行うことにより対象物を操る方法であり、ハイブリッド制御タイプのもの [6] は、マニピュレータを位置制御用と力制御用に分けるのではなく、複数のマニピュレータのうち6自由度を位置制御に用い、残りの自

由度を力制御に用いることにより対象物を操る方法である。また対象物を直接制御するタイプのもの [2] ~ [4] は、目標通り対象物を操るのに必要な力を運動方程式より求め、その力をマニピュレータにより発生させることにより対象物を操る方法である。これらの方法は既知の対象物を操る際に有効な方法ではあるが、未知対象物の操りについては触れられていない。

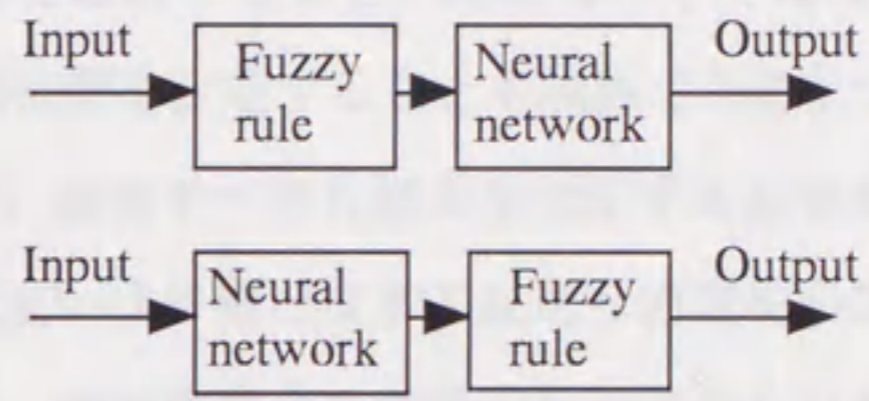
また一方で、厳密な数学モデルを必要とせず、あいまいな言語的表現で制御ルールを作成できるファジィ制御 [10], [61], [62] のロボットへの適用も積極的に行われている [63]。しかし、ファジィ制御器自身では学習/適応能力を持っておらず、一度設計したファジィ制御ルールを調整することが困難であるため [62]、最近では学習/適応能力を持つニューラルネットワークとの融合も盛んに行われている [64]。ファジィ推論とニューラルネットワークの融合方法は一つに限られておらず、種々の融合方法が提案されている [12] [64]。これらの融合方法の種類 [64] を図 1.10 に示す。ファジィ/ニューロはファジィ推論とニューラルネットワークを並列に置くもので、ファジィ-ニューロとニューロ-ファジィはそれらを直列に配置する方法である。ニューロ的ファジィは、ファジィルールをモデルとしニューラルネットワーク的に学習をすることでファジィ制御の調整を行う方法である。ファジィ入出力ニューロとは、ファジィルールの前件部を入力データとし、後件部を出力データとして学習するニューラルネットワークである。また、ファジィ的ニューロとは、ファジィルールを表現するニューラルネットワークである。一方、ニューロ化ファジィは複数のニューラルネットワークによりファジィルールを構築する方法であり、ファジィ化ニューロとはニューラルネットワークのウェイトなどにファジィ集合を用いる方法である。これらの方法はファジィ推論とニューラルネットワークそれぞれの特徴を活かすものとなっている。

これまでに多くの研究者によりファジィ・ニューラルネットワークの研究が行われている [64] が、ロボット制御への応用例 [97], [100], [108], [109] では、移動ロボットの制御 [97] やナビゲーション [100] あるいはロボットビジョン [108] などに用いられた例は多いが、ロボットマニピュレータの位置/力制御に用いられた例は見あたらない。

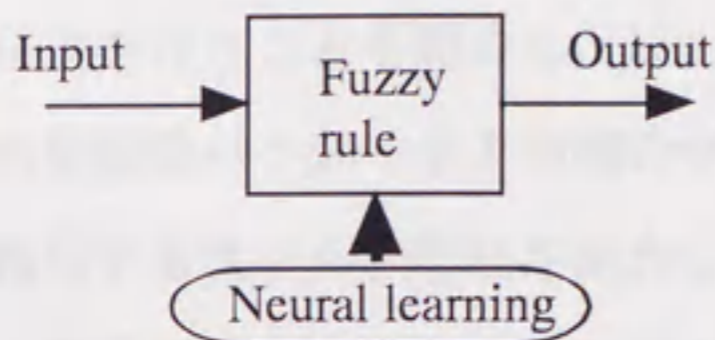
位置/力制御を行うロボットマニピュレータにより高度な作業を実現させるには、



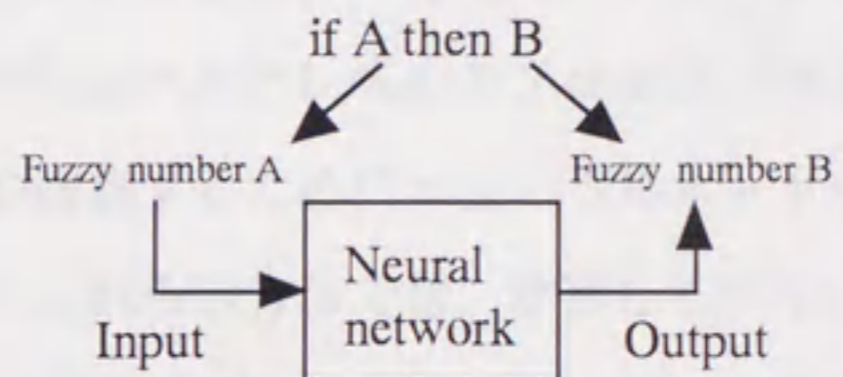
(a) ファジィ/ニューロ



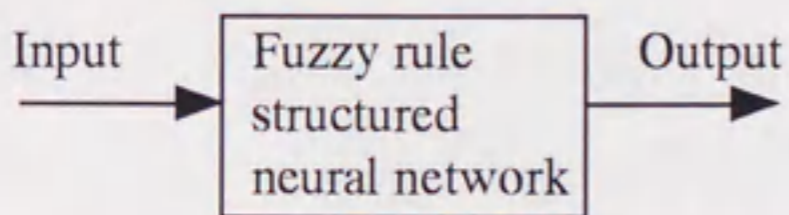
(b) ファジィ-ニューロ
ニューロ-ファジィ



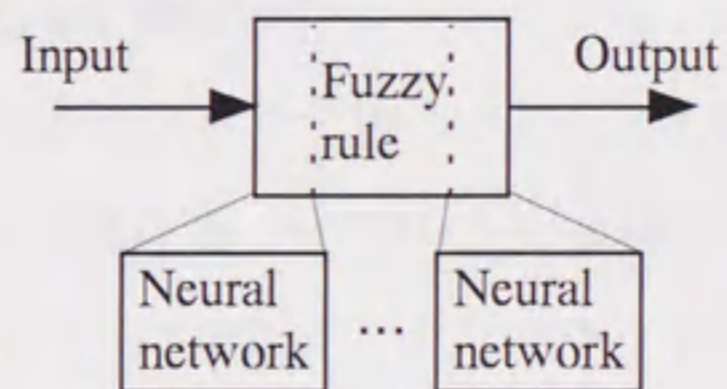
(c) ニューロ的ファジィ



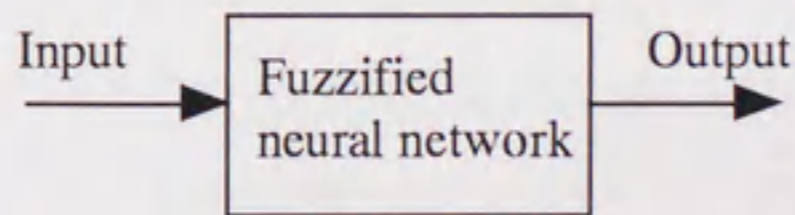
(d) ファジィ入出力ニューロ



(e) ファジィ的ニューロ



(f) ニューロ化ファジィ



(g) ファジィ化ニューロ

図 1.10 ファジィ推論とニューラルネットワークの融合

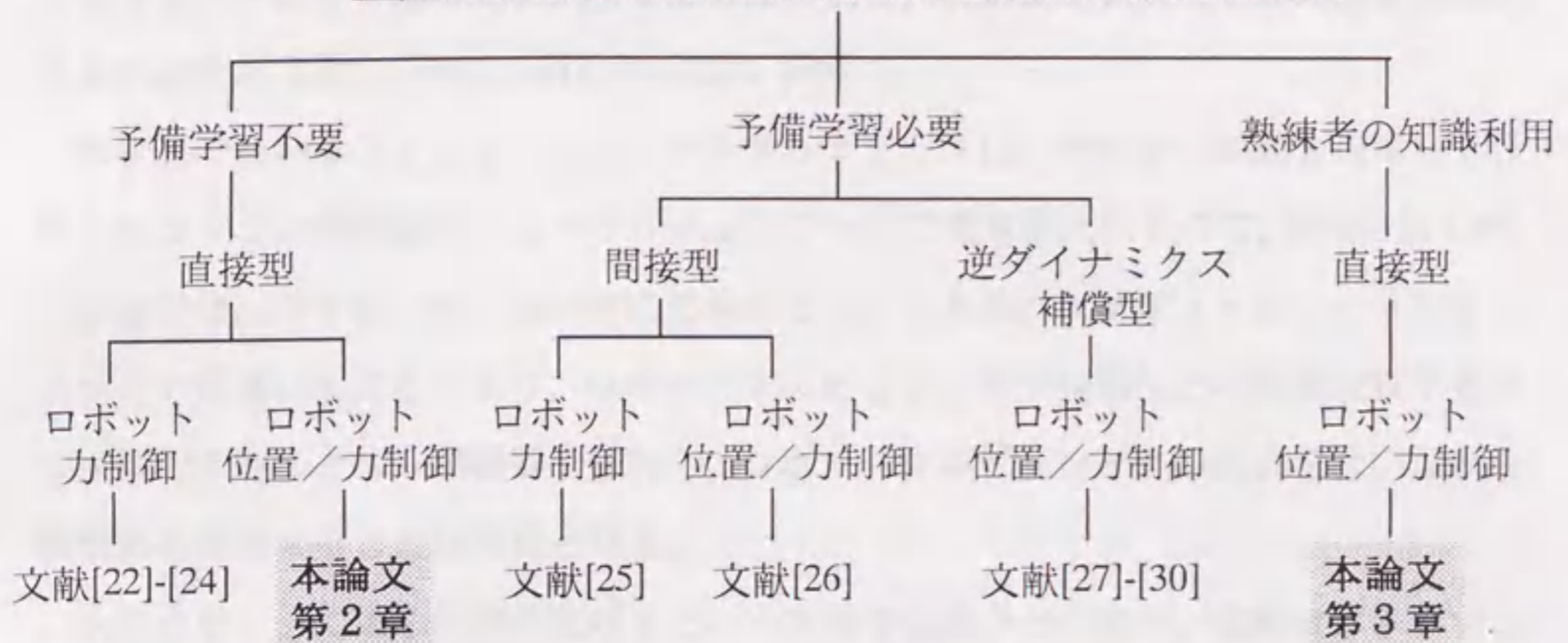
その作業を遂行するのに必要な目標位置と目標力を設定しなければならない。しかし実際問題として、精度の良い位置／力制御器を設計することが困難だけでなく、作業を遂行するのに必要となる目標力や目標位置を設定することも困難である。つまり、ロボットにより対象物に力を加えるには、実現すべき目標力を設定する必要がある。要求された作業を遂行するには目標軌道或いは目標となる工具送り速度を設定する必要がある。しかし、実際の作業などでは、熟練作業者の経験的な知識から目標力や目標工具送り速度などが決められている。そのため、熟練者からロボットへの作業の教示が必要となる。これまでに Yang と Asada [68] は、センサースペースから制御コマンドとの間のマッピングを用いて、熟練者からロボットへ言語的アプローチでスキル伝達を行うことを紹介し、Liu と Asada [69] はタスクレベルのプロセスパラメータと制御戦略パラメータとの間のマッピングを構築するためにニューラルネットワークを利用するスキル伝達法を紹介しているが、これらの方法では、学習していないような予想外の状況を扱うのは困難である。また、Xu と Yang [70] はヒューマンスキルの伝達とモデル化に隠れマルコフモデルを利用している。しかし、これらの方法は、人間が作業をした結果生じるセンサーデータを得るための予備実験と、モデル作成あるいはマッピング作成のための予備学習を必要としていた。

1. 3 本研究の目的

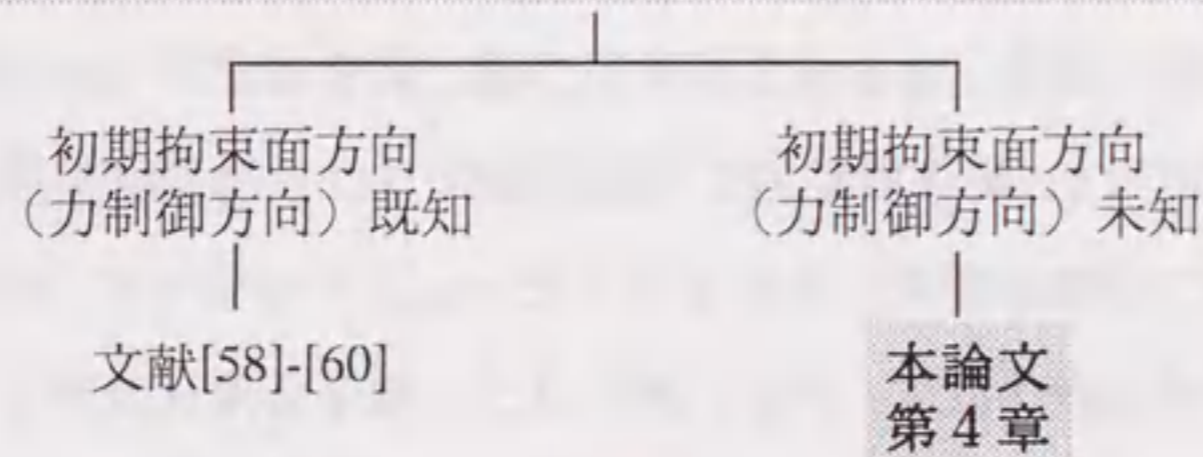
1. 1節でも述べた通り、従来のロボットマニピュレータによる位置/力制御の研究のほとんどはロボットマニピュレータ及び環境（あるいは対象物）の特性が既知であること、すなわち限られた環境下で利用されることを前提としていた。本研究では、ロボットマニピュレータにより一層高度な作業を実現させるため、動特性や形状などが未知な環境に対し位置/力制御を施したり、動特性や重量などが未知な対象物を操ったり、高度な作業を遂行する上で必要な情報を獲得する方法を探究することを目的とする。この目的を実現させるため、人間の言語などの曖昧な表現や不確かなデータ等を扱うことができるファジィ推論、学習/適応能力があるニューラルネットワーク、およびそれらを融合させたファジィ・ニューラルネットワークを応用すると同時に、学習/適応能力を高めるために有効な学習の評価関数 [71] や学習係数決定法 [77] も提案する。これまでの研究に対する本研究の位置付けを図 1.11 に示す。

未知環境でのロボットマニピュレータの使用を可能にするため、材料特性や形状などの情報が未知な環境に対する位置/力制御を実現させる必要がある。しかし前節でも説明したとおり、これまでの研究では、未知環境に対しロボットマニピュレータの位置/力ハイブリッド制御を実現させる有効な制御法はあまり提案されていない（図 1.11 参照）。これまでに提案されているようなニューラルネットワークを間接的に利用して位置/力ハイブリッド制御器のゲイン調整を行う間接型ニューラルネットワーク制御 [26] や、ニューラルネットワークにより逆ダイナミクスの補償を行う逆ダイナミクス補償型制御 [27] - [30] では、あらかじめニューラルネットワークを学習させる予備学習が必要であった。また、ロボットマニピュレータの関節座標系での制御 [28] - [30] では、ロボットの関節数が多くなると構造が複雑になる。予備学習を必要としない適応型のニューラルネットワーク制御器は、これまでに 1 自由度の力制御で提案されているだけ [22] - [24] で、多自由度ロボットマニピュレータによる位置/力ハイブリッド制御に適用された例は見あたらない。また一方で、ロボットマニピュレータによる位置/力ハイブリッド制御の位置制御では、ロボットマニピュレータと環境間の摩擦補償の問題が生じるが、これまでの研究でこの問題を解決している例は見あたらない。そこで本研究では、ニューラルネットワークやファジィ・ニューラルネットワー

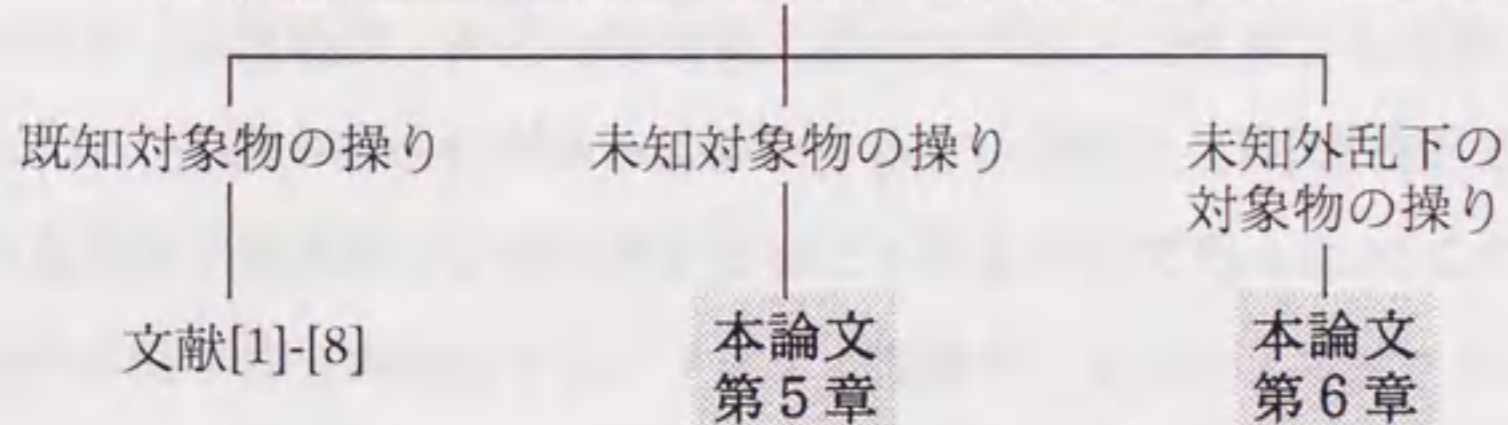
ニューラルネットワーク等を利用した動特性が未知な環境に対する位置／力ハイブリッド制御（或いは力制御）の研究



形状が未知な環境に対する位置／力制御の研究



ロボットマニピュレータによる対象物の操りの研究



作業に必要な目標位置／力生成法の研究

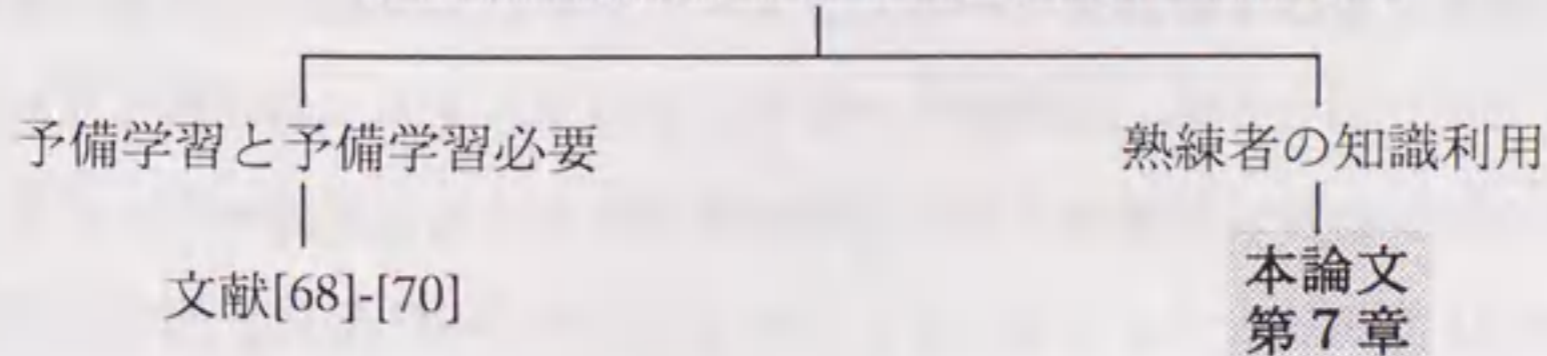


図 1.11 本研究の位置付け

クなどを用いて、材料特性や形状などの情報が未知な環境に対し多自由度ロボットマニピュレータにより基準座標系での高精度の位置/力制御を実現させる方法を提案するものとする [20], [21], [71] ~ [73], [86]。

本研究で用いるファジィ・ニューラルネットワークは、熟練者の知識を利用して設計したファジィ制御器をニューラルネットワークで置き換えたもので、前節の図1.8での分類では、ファジィ的ニューロに該当するものである。ファジィ・ニューラルネットワークを用いることにより、ロボットマニピュレータや環境などの厳密な数学モデルを解析することなく制御器を設計し、ロボットマニピュレータの制御を行いながら制御器を調整することが可能となる。

次にロボットによる作業を広げるという目的を達成させるため、複数ロボットマニピュレータにより動特性や質量などが未知な対象物を操ったり、未知外乱下で対象物を操ったりすることが要求される。しかし、前節で挙げたようなこれまでに提案されている制御法は既知の対象物を扱う際には有効であるが、学習/適応能力を持っていないため、未知対象物を操ったり未知外乱下で対象物を操ったりすることは難しい。そこで本研究では、ファジィ・ニューラルネットワーク等を用いて複数ロボットマニピュレータにより未知対象物を操ったり [74], [75], 未知の外乱下で対象物を操ったり [76], [77] する制御法を提案するものとする。

また、高度な作業を実現させるため、作業を適切に遂行するために必要となる目標力や目標位置（目標軌道、あるいは目標工具送り速度）を生成する必要がある。そのため、熟練者の知識や業をロボットに教示する方法 [68] ~ [70] が提案されてきたが、これらの方法は予備実験や予備学習が必要とされるものであった。そこで本研究では、予備実験や予備学習を実施せずに、ファジィ推論やファジィ・ニューラルネットワークを利用することで、作業を適切に遂行するために必要となる目標力や目標位置（目標軌道、或いは目標工具送り速度）を生成する方法を提案するものとする [78]。

制御器の安定性の解析はロボットマニピュレータを制御する上で重要な問題であるが、ファジィ制御器、ニューラルネットワーク制御器、あるいはファジィ・ニューラルネットワーク制御器のような非線形制御器の安定性解析は容易ではなく、現在も多くの研究者の研究課題 [104] ~ [107] となっている。これらの研究 [104] ~ [106] は主にリアプノフの安定論の立場に基づいているものであるが、一般的にはリアプノフ

関数を見つけること自体が困難である。本研究では新しい制御法を提案しロボットマニピュレータにより高度な作業を実現させることに重点を置き、安定性の数学的な検討については対象外とした。

1. 4 本研究の構成

本研究では、まず未知環境に適応できる学習・適応能力をロボットマニピュレータに与えるため、第2章において、未学習のニューラルネットワークを直接的に多自由度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へ応用し、動特性の未知な環境に対し制御器を適応させる方法を提案する [21]。ここでは、未知環境に対応するために小さいゲインを使用した従来のハイブリッド制御器の出力をニューラルネットワークへの入力情報として利用した。したがって、ここで提案する適応型ニューラルネットワーク制御器は、ニューラルネットワークと従来のハイブリッド制御器を組み合わせたものである。提案した制御器の有効性を実験により証明した。

第3章では、第2章で紹介する未知対象物に対する位置と力のハイブリッド制御では抑制できなかったオーバーシュート性をなくすため、制御器設計者（制御の熟練者）の知識を利用したファジィ・ニューラルネットワークを応用した位置と力のハイブリッド制御器を提案する [71], [72]。この制御器は、第2章で提案するような従来のハイブリッド制御器とニューラルネットワークを合わせた制御器ではなく、ファジィ・ニューラルネットワーク自体が制御器として用いられる。またここでは、力制御 [79] ~ [82] において未知環境の特性がファジィ制御ルール設計時に想定したものと異なっていた場合に、入力値調整用ニューラルネットワーク (IANN = Input Adjusting Neural Networks) を用いてファジィ・ニューラルネットワークを瞬時に調整する方法を提案すると共に、位置/力制御の際の位置制御において未知環境とロボットマニピュレータ間の摩擦を行うため、摩擦補償用ニューロンをファジィ・ニューラルネットワーク制御器に加え、状況により学習を切り替えながら制御する方法を提案する [83], [84]。通常、ロボットマニピュレータのファジィ・ニューラルネットワーク制御器の学習則としてバックプロパゲーション法が利用され、その評価関数としては目標値との誤差を2乗した関数が用いられる。しかし、ロボットマニピュレータの値から制御などでは目標力がステップ状に変化する場合も多く、周期的に目標力が変

化した場合、このような評価関数で学習を継続するとオーバーシュートが生じる可能性がある。そこで第3章では、ファジィ制御された評価関数を用いて効率的に学習を行う方法も提案する [71], [85]。ここで提案した制御器の有効性はコンピュータによるシミュレーションにより確認した。

第4章では、形状等が未知な環境に対し位置と力のハイブリッド制御を実現させるため、力制御方向を探索するアルゴリズムを提案する [73], [86]。ここでは大きさと方向が曖昧なファジィベクトルの概念を提案し、ノイズを含む力センサーからの情報をファジィベクトルにより有効に扱う方法を説明する。また、力制御において未知環境の特性がファジィ制御ルール設計時に想定したものと異なっていた場合に、第3章で説明する IANN より汎用的で、かつ力制御方向の調整具合を考慮しながらファジィ・ニューラルネットワークの調整を行うファジィ環境評価器 [73], [87], [88] を提案する。

次に、ロボットマニピュレータにより重量や特性の未知な対象物を操ったり、未知な外乱が作用している対象物を操ったりすることを実現させるため、未知対象物を操るためのメインのファジィ・ニューラルネットワーク制御器と、メイン制御器からの出力を目標力としてロボットマニピュレータを制御するサブのファジィ・ニューラルネットワーク制御器から構成される制御システム [74] ~ [76] を第5章で提案する。また、ロボットマニピュレータを制御するサブ制御器の内部に標準環境モデルを置き、未知対象物の特性がファジィ制御ルールを設計した際に見積もったものより硬かった場合にファジィ・ニューラルネットワーク制御器の調整を行う簡易的な方法を提案する [74], [76]。未知対象物や未知外乱下の対象物の操りに対する提案する制御法の有効性は、コンピュータによるシミュレーションにより確認した。

複腕ロボットマニピュレータで何らかの環境に接している対象物をその環境に接したまま動かす場合(例えば対象物を床上で引きずる場合)、対象物にクーロン摩擦が作用する。本研究では、この様に対象物を環境に接したまま動かす制御を“滑らせ移動制御”と呼ぶことにする。第5章で、ファジィ・ニューラルネットワークを用いることにより制御器がクーロン摩擦等の外乱の補償を学習していくため、徐々に対象物を目標軌道通り制御ができるようになることを示す [76] が、制御器が学習するまで少し時間がかかり、その間の対象物の動きも滑らかなものではない。そこで第6章におい

て、第3章で提案した位置制御器を応用し、力が与えられているにも関わらず対象物が動かない場合だけ学習する摩擦補償用ニューロンと、対象物が運動しているときだけ学習する対象物操り用ファジィ・ニューラルネットワークを組み合わせた対象物操り制御器を提案する [77]。また、第6章では、ファジィ・ニューラルネットワークをバックプロパゲーション法により学習させる際に適切な学習係数を設定するため、学習係数を線形的に増加させたり指数的に減少させたりする Delta-Bar-Delta 学習法をもとに、学習状況に応じ学習係数が小さい場合の増加割合を変化させる新しい Delta-Bar-Delta 学習法を提案する [77]。

ロボットマニピュレータによる高度な作業を実現させるため、第7章では、予備実験や予備学習を必要とせずに、作業に必要な目標力と目標工具送り速度を獲得する知的作業プランナー（タスクレベル制御器）を提案し [78]、例としてバリ取り作業に適用する。提案する知的作業プランナーはファジィ・ニューロ作業プランナーとファジィ対象物特性評価器から構成されている。効果的な評価関数を用いて作業中にオンラインで調整を行うファジィ・ニューロ作業プランナーは目標工具送り速度を設定するために用いられる。ファジィ対象物特性評価器は、作業の対象となる未知対象物の材料特性を評価し、作業に必要な目標力を設定するとともに、下位レベルの制御器である適応型ファジィ・ニューラルネットワーク力制御器を瞬時に調整するための入力値調整係数を生成する。また、ここでのファジィ・ニューロ作業プランナーの学習には、学習すべき場面でのみ学習を行うファジィ評価関数を用いる。

最後に第8章において本研究をまとめるものとする。

第2章 ニューラルネットワークを利用したロボットマニピュレータの未知環境に対する位置／力ハイブリッド制御

2.1 緒言

当初人間の脳とその神経システムの研究により始められたニューラルネットワークは、数多くのシナプスとニューロンから成り立っている。ちなみに、人間の脳では約 10^{11} のニューロンと 10^{15} のシナプスを持っている。ニューラルネットワークには、いろいろな種類のアーキテクチャと学習則が提案されており、工学的によく用いられる学習則ではバックプロパゲーション法がよく使われている。McCullochとPittsが1943年に最初のニューロンモデル[13]を提案して以来、ニューラルネットワークは多くの研究者により研究されてきた。ロボットの分野では、順／逆キネマティクスあるいはダイナミクスの実現、ロボットのコントロール、および同定などに利用されている。

ニューラルネットワークの制御への応用は、1960年代にWidrowにより行われ[89]、後に川人らによってロボット制御へ適用された[32]、[33]。その後もニューラルネットワークのロボット制御への応用は盛んに行われてきた[22]～[39]、[54]、[55]。しかし、そのほとんどが位置制御や軌道制御への応用であり[32]～[39]、ニューラルネットワークの持つ未知環境への適応能力を力制御に応用した例は少なかった[22]～[30]、[54]、[55]。力制御への応用では、制御器のゲイン調整など間接的にニューラルネットワークを利用することが多く[25]、[26]、[36]、[54]、[55]、直接制御器として用いた例[22]～[24]は、1自由度の力制御だけで、多自由度ロボットマニピュレータの位置／力制御への応用はされていなかった。また、これまでの力制御への応用の多くは、あらかじめニューラルネットワークを学習させておく必要があったり[25]～[30]、繰返し学習する必要があった[55]。そこで本章では、未学習のニューラルネットワークを直接的に多自由度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

へ応用し、動特性の未知な環境に対し制御器を適応させる方法を提案する [21]。

ハイブリッド制御は Raibert と Craig により提案されたもので、位置/力制御で最も重要な方法の一つである [42]。しかし、通常のハイブリッド制御、つまり従来の制御系で未知環境に力を加える場合、動特性の未知な環境に対して力制御のゲインを大きくするとシステムが不安定になる可能性があるため、大きなゲインを使うことができないが、力制御のゲインが小さいと目標力に達することが難しい。したがって、従来の制御系では環境の動特性に応じてゲインを設定してやる必要があり、動特性の未知な環境に対しては適切なゲインの設定ができなかった。この問題を解決するため、ハイブリッド制御器にニューラルネットワークを直接的に利用して補償を行うことにより、動特性が未知な環境に対して多自由度のロボットマニピュレータで位置/力を制御する方法を本章で提案する。この方法を用いることにより、力制御のゲインが小さくても、また、動特性の異なる未知環境に対しても、ニューラルネットワークの補償により目標力に達することができ、目標力の変動に対しても追従できることを実験により確認した。ここでは、まずハイブリッド制御器を構築し、この制御器からのアウトプットをニューラルネットワーク制御器のインプットの一部として使用した。

本章ではまず 2.2 節でハイブリッド制御について説明し、2.3 節と 2.4 節でニューラルネットワーク制御器の説明をした後、2.5 節で本ニューラルネットワーク制御器を平面 3 自由度のダイレクトドライブロボットマニピュレータを用いて実験を行った結果を示す。

2.2 ハイブリッド制御

図 2.1 のようなロボットマニピュレータで力を x 方向に加えるには、エンドエフェクタの y 方向の位置と環境に対する角度も制御しなければならない。このように位置と力を同時に精度良く制御する場合、ハイブリッド制御が適当であると思われる。

平面ロボットの運動方程式を考えると次式のようなになる。

$$M(q)\ddot{q} + h(q, \dot{q}) + F_c \operatorname{sgn}(\dot{q}) = \tau - J^T f \quad \dots\dots\dots (2.1)$$

ただし、 $M(q)$ は慣性行列、 $h(q, \dot{q})$ は遠心力・コリオリ力等の非線形項、 F_c は関節でのクーロン摩擦、 τ は駆動トルクベクトル、 J はヤコビ行列、そして f は環境からロボットのエンドエフェクタ先端に作用している力であり、 q は関節位置ベクトルである。

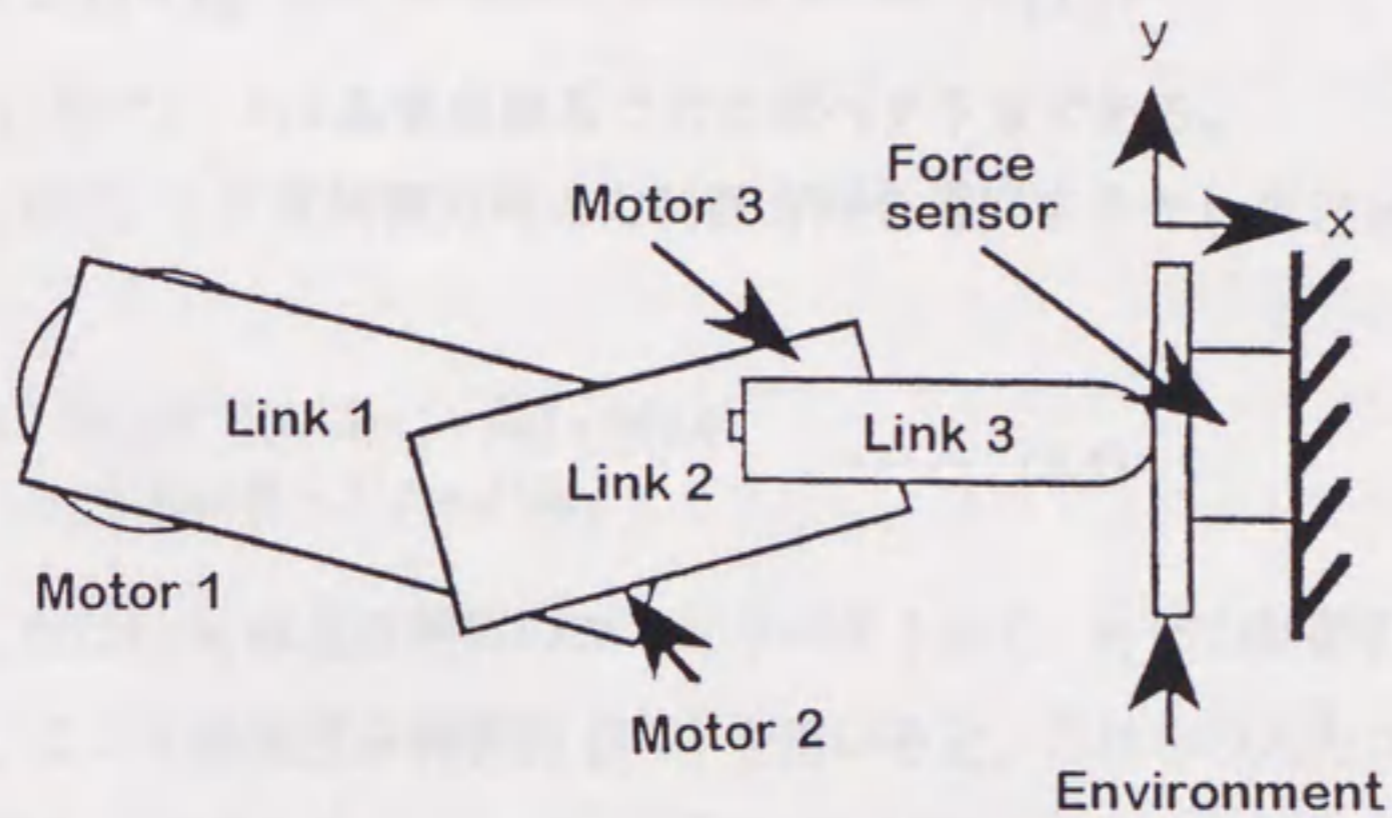


図 2.1 平面 3 自由度ロボットマニピュレータ

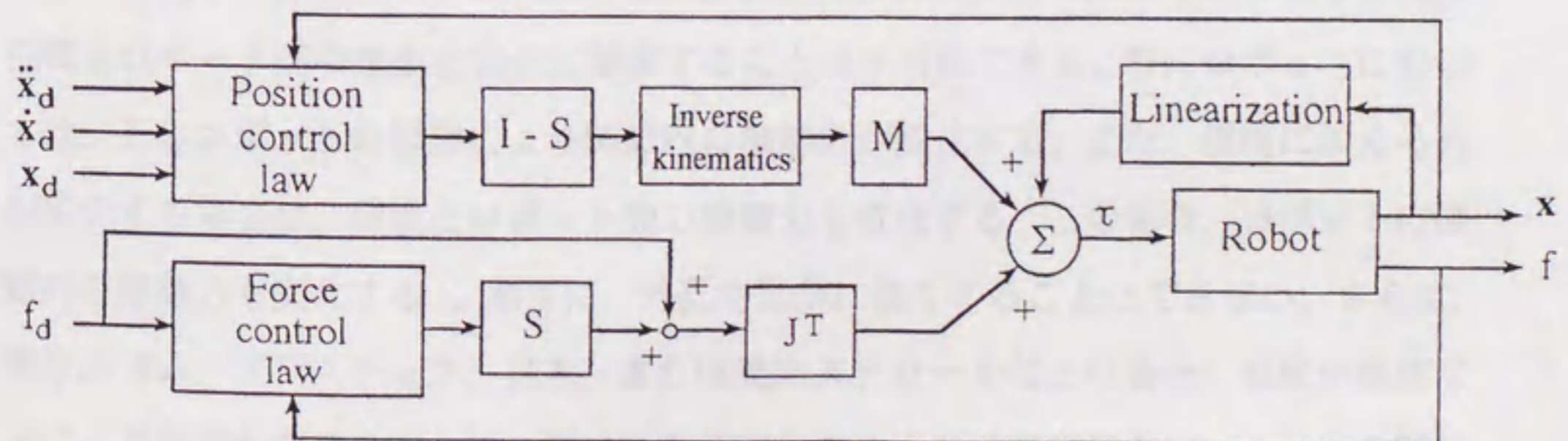


図 2.2 ハイブリッド制御のブロック線図

また、ここではロボットの速度は低速であると仮定し、粘性摩擦抵抗は無視した。

式 (2.1) より、必要なモータトルクは次式のように表される。

$$\tau = M(q)\ddot{q} + h(q, \dot{q}) + F_c \operatorname{sgn}(\dot{q}) + J^T f \quad \dots\dots\dots (2.2)$$

基準座標系でのエンドエフェクタの加速度と関節位置との関係は、

$$\ddot{x} = J\ddot{q} + \dot{J}\dot{q} \quad \dots\dots\dots (2.3)$$

で表される。ただし、 x は基準座標系での位置ベクトルである。

式 (2.2), (2.3) と位置制御方向と力制御方向を選択するセレクションマトリクス S を使うと、

$$\begin{aligned} \tau = & M(q)J^{-1}[(I-S)u_p - \dot{J}\dot{q}] + h(q, \dot{q}) \\ & + F_c \operatorname{sgn}(\dot{q}) + J^T f + J^T S u_f \quad \dots\dots\dots (2.4) \end{aligned}$$

が得られる。ただし u_p は位置制御のコマンドベクトルで、 u_f は力制御のコマンドベクトルである。ここで加速度分解制御 [103] を用いると、これらの入力コマンドは次式のようになる。

$$u_p = \ddot{x}_d + K_{pv}(\dot{x}_d - \dot{x}) + K_{pp}(x_d - x) \quad \dots\dots\dots (2.5)$$

$$u_f = K_{fp}(f_d - f) + K_{fi} \int (f_d - f) dt \quad \dots\dots\dots (2.6)$$

ハイブリッド制御系のブロック線図は図2.2に示される。本制御器により、環境に加える力やエンドエフェクタの角度と y 方向の位置が制御される。

このハイブリッド制御器により目標の位置と力を得ることは次の理由により困難である。第1に、ロボットマニピュレータと環境の完璧な数学モデルを作ることは非常に難しい。第2に、環境の動特性がシステムの安定性に影響するため、未知環境での力制御に高ゲインを使うことはできない。第3に、ロボットの関節内の摩擦や、未知環境とロボット間の摩擦を完全に補償することは不可能である。特にロボットにおいては、そのロボットの形態により関節内の摩擦力が変化する。また、環境に加える力が変化する場合は、環境とロボット間の摩擦力も変化する（この場合、ロボットの関節内の摩擦力も変化する）。第4に、外乱を完全に無くすることはできない。さらに、環境がゴム、プラスチック、材木、または発泡スチロールなどの場合、温度や湿度などにより特性が変化するため、ゲインを固定しておくのは適切でない。これらの問題を解決するためには何らかの適応スキームが制御器に要求される。

2.3 ニューラルネットワーク制御器

目標の位置と力を得るためには、未知環境に適応し、ロボットの非線形性を補償するニューラルネットワーク制御器を用いることが解決策の一つである。ニューラルネットワーク制御器にはいくつかのタイプがあるが、図2.3に示す様な直接型のニューラルネットワーク制御器がニューラルネットワークの最も効果的なロボット制御器への適用だと思われる。しかし、ハイブリッド制御器のような既存型の制御器は比較的容易に設計可能なので、そのような制御器を最初に作成し、この制御器からのアウトプットをニューラルネットワーク制御器のインプットとして利用する方法を提案する(図2.4)。ここでハイブリッド制御器からのアウトプットは、そのままロボットへのインプットとしても利用し、ニューラルネットワークはハイブリッド制御器を補うために利用されるものとする。この制御器においては、ニューラルネットワークが摩擦、モデリングエラー、外乱の補償等を行うので、ハイブリッド制御部での摩擦の補償は不要となる。したがって、式(2.4)は次式のようなハイブリッド制御部からの出力式に書き換えられる。

$$\tau_h = M(q)J^{-1}[(I-S)u_p - \dot{J}\dot{q}] + h(q, \dot{q}) + J^T f + J^T S u_f \quad \dots\dots\dots (2.7)$$

ここでは、ニューラルネットワークの出力 τ_h と区別するためにハイブリッド制御部からの出力を τ_h とした。制御器からのコマンドベクトルは次式のようになる。

$$u_p = \ddot{x}_d + K_{pv}(\dot{x}_d - \dot{x}) + K_{pp}(x_d - x) \quad \dots\dots\dots (2.8)$$

$$u_f = K_{fp}(f_d - f) \quad \dots\dots\dots (2.9)$$

式(2.9)で見られるように、ニューラルネットワークがハイブリッド制御部のエラーの補償を行うため、エラーの積分はもはや必要なくなる。さらに、式(2.9)でのゲインは、ニューラルネットワークが未知環境に適応するため環境が変化しても変える必要がない。このニューラルネットワーク制御器の詳細は、図2.5に示されている。この図より、本制御器は図2.2のハイブリッド制御器と図2.3のニューラルネットワーク制御器を合わせたものであるということがわかる。

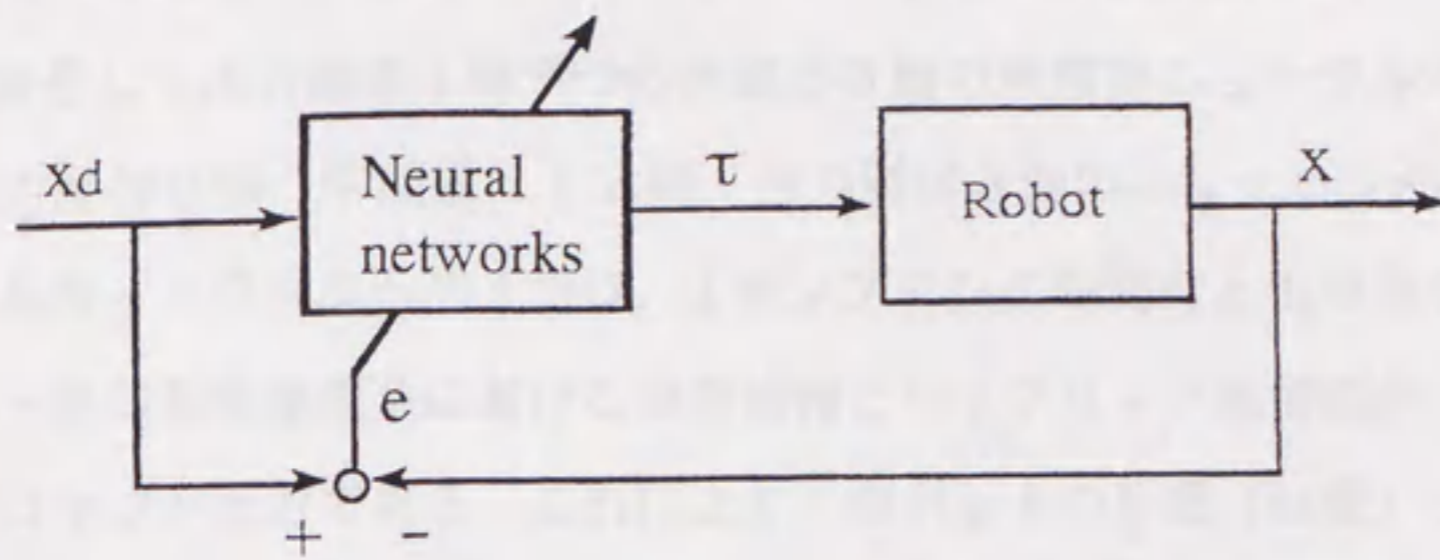


図 2.3 直接型ニューラルネットワーク制御器

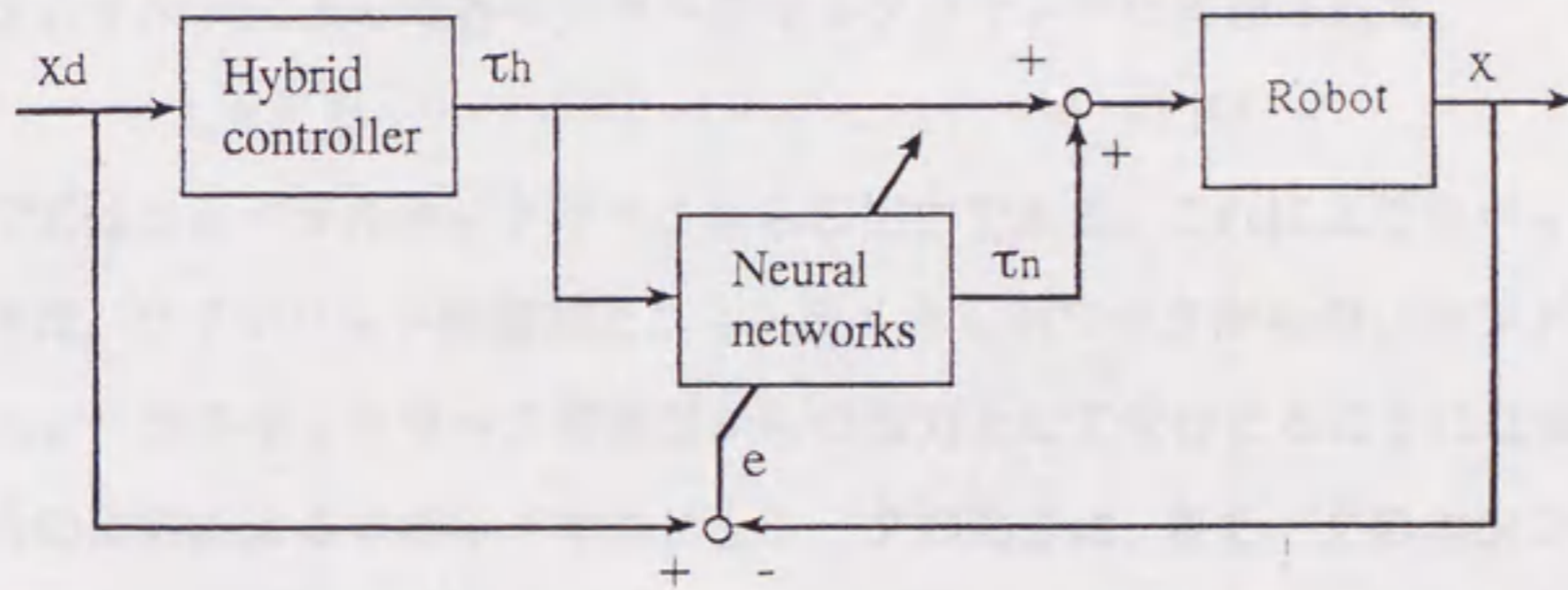


図 2.4 提案するニューラルネットワーク制御器

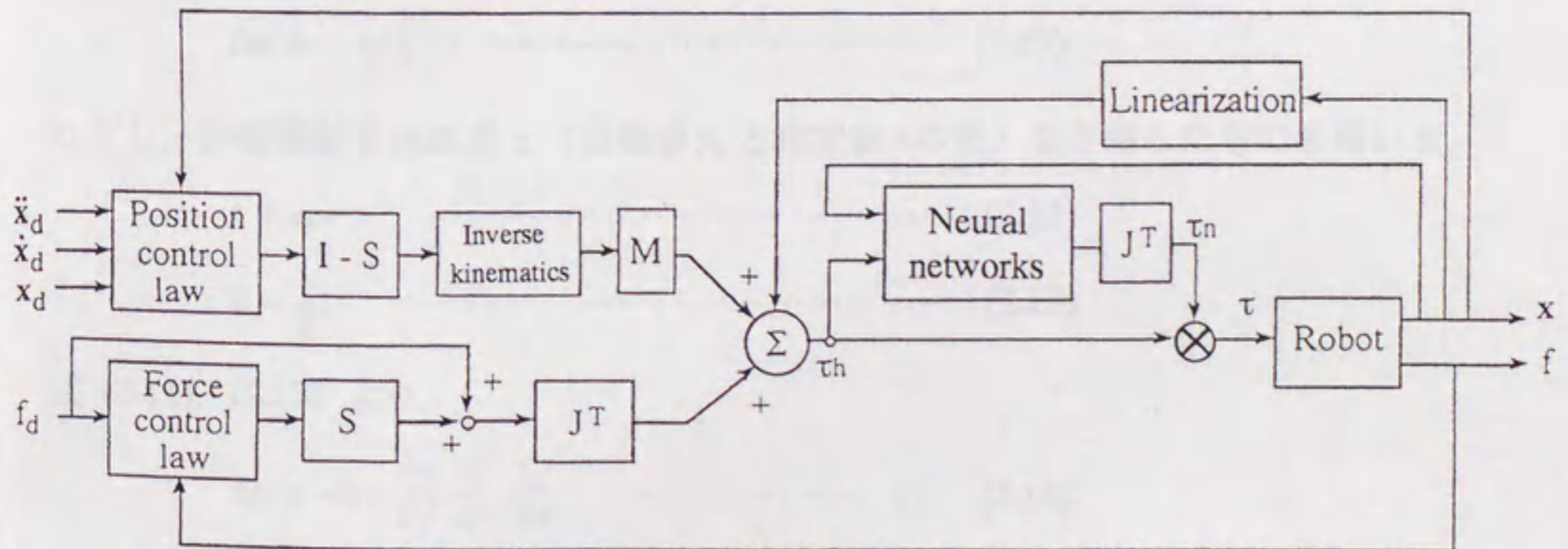


図 2.5 ニューラルネットワーク制御のブロック線図

2. 4 ニューラルネットワークの構成

ニューラルネットワークのアーキテクチャは図2.6に示されている。これは、入力層、中間層そして出力層各1層ずつの非線形3層の階層型ニューラルネットワークである。入力層は9個、中間層は12個、出力層は3個のニューロンから成っている。ニューラルネットワークへの入力は、1サンプリング時間前と現時点でのロボットマニピュレータの基準座標系における位置情報とハイブリッド制御部からの各モータへのトルクコマンド出力である。これにより、ロボットの形態（位置）と速度、目標位置/力を得るためにハイブリッド制御部で計算された目標加速度情報がインプットされる。ニューラルネットワークからの出力は、基準座標系でのロボットマニピュレータへの力コマンドである。これらの力コマンドは、関節トルクと基準座標系の力との関係を表す次式によって各モータへのトルクコマンドに変換される。

$$\tau_n = J^T f_x \quad \dots\dots\dots (2.10)$$

ここで f_x はニューラルネットワークからの出力である。これによりロボットマニピュレータは、ハイブリッド制御部とニューラルネットワークからのコマンドトルクの合力をニューラルネットワーク制御器からの出力として受けとることになる。この制御器からの出力によるロボットマニピュレータの動きは、各モータのエンコーダと力センサーにより計測される。ニューラルネットワークの各ウエイトは、位置と力のエラーの値（目標値と測定値の差）によりサンプリング毎（実験では1ms毎）にバックプロパゲーション法で調整される。

各ウエイトの変化量は、学習係数を $\eta (>0)$ とすると、

$$\Delta w = -\eta \left(\frac{\partial E}{\partial w} \right) \quad \dots\dots\dots (2.11)$$

ただし、評価関数 E は誤差 ε （目標値 y_d と測定値 y の差）を2乗したものをを用いる。

$$\varepsilon = y_d - y \quad \dots\dots\dots (2.12)$$

$$E = \frac{1}{2} \varepsilon^2 \quad \dots\dots\dots (2.13)$$

式(2.11), (2.13) より、

$$\Delta w = -\eta \varepsilon \frac{\partial y}{\partial f} \frac{df}{du} \frac{\partial u}{\partial w} \quad \dots\dots\dots (2.14)$$

となる。ここで、 f はニューラルネットワークで用いたアクティベーションファンクション（シグモイド関数）であり、次式で表される。

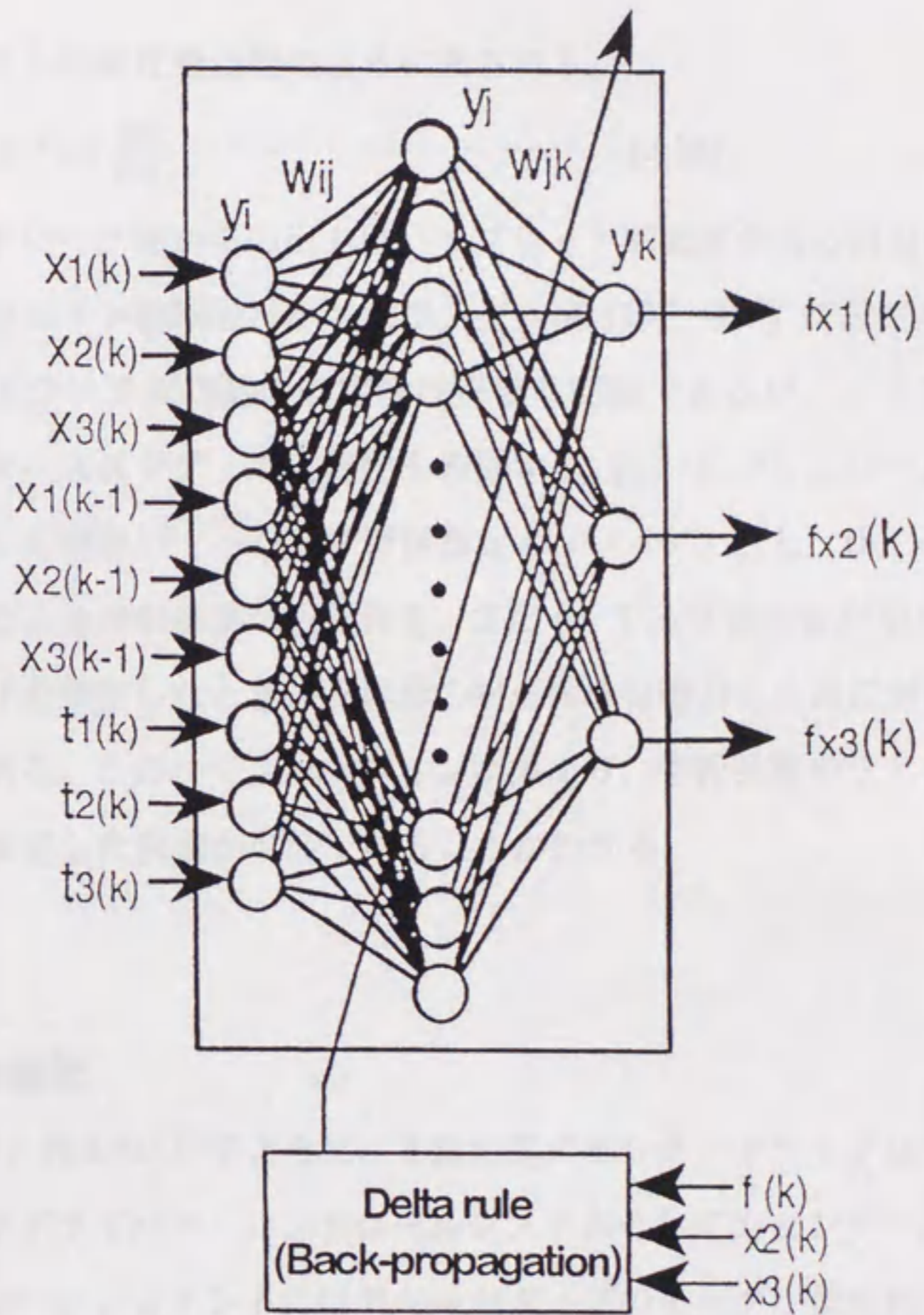


図 2.6 ニューラルネットワークの構成

$$f(u) = \frac{2}{1+e^{-u}} - 1 \quad \dots\dots\dots (2.15)$$

ただし、 u は層への入力である。

式 (2.14) で $\frac{\partial y}{\partial f}$ は不明であるが、調整パラメータである η により加味される [22]。

したがって、ウエイトの変化量は次のように表される。

$$\Delta w = -\eta \varepsilon f'(u) \frac{\partial u}{\partial w} \quad \dots\dots\dots (2.16)$$

ニューラルネットワーク部からの出力はハイブリッド制御部からの出力を補償するものであるため、ウエイト初期値は小さなランダム値 ($10^{-3} \sim 10^{-4}$) に設定した。

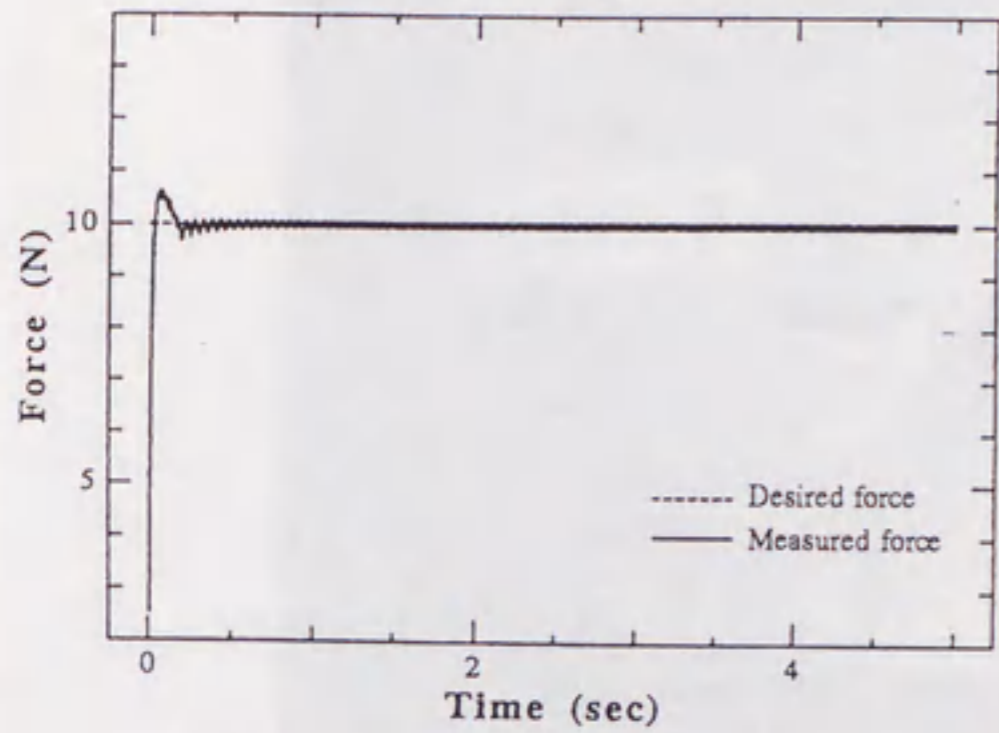
ニューラルネットワーク制御器の安定性は重要な問題であるが、ここではシミュレーションによるケーススタディで安定性を確認する。図2.7にシミュレーション結果を示す。これは (a) を標準パターン (学習係数 $\alpha = 1.0 \times 10^{-4}$) とし、(b), (c) では学習係数を増減させたときの制御器の収束性を、また (d) では学習係数が (a) と同じで、ウエイト初期値だけを変更したときの制御器の収束性を目標力 10 N に対する追従性で確認したものである。このシミュレーション結果より、学習係数やウエイト初期値を多少変更しても安定した制御が可能であることがわかる。

2. 5 実験

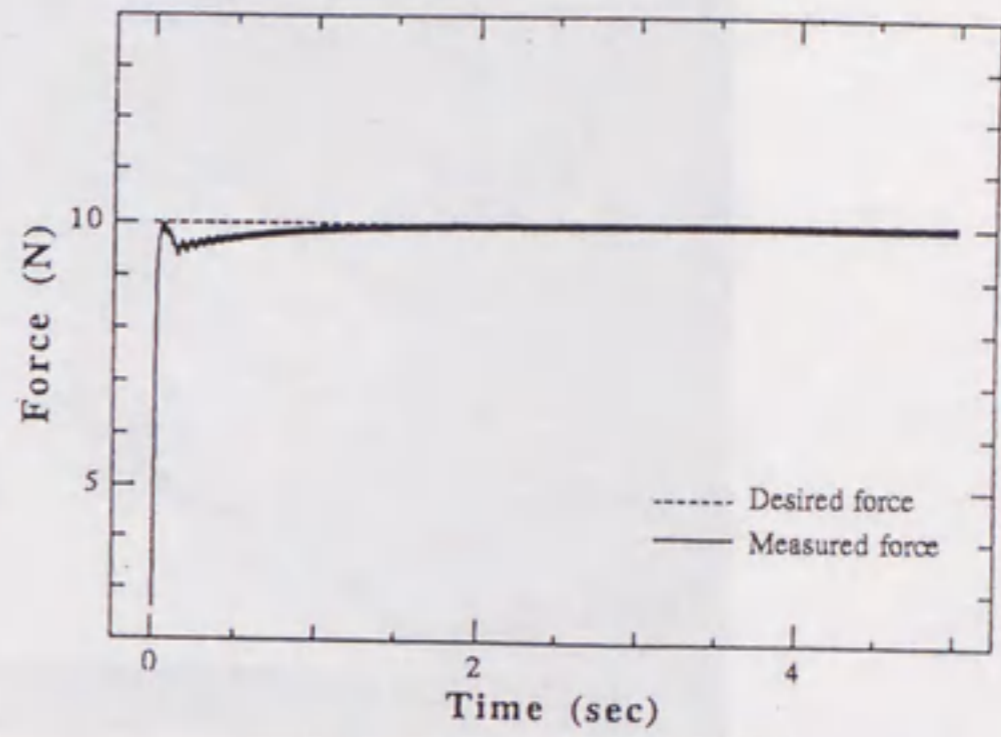
2. 5. 1 実験装置

実験装置は図2.8と図2.9に示すように、3自由度ダイレクトドライブロボットマニピュレータ、モータドライバー、コントロールシステムそして力センサーから構成される。各マニピュレータジョイントには異なったタイプのモータが使われている。コントロールシステムとしてdSPACEを用いた。dSPACEはDSPを基にして、AD/D Aコンバータとインクリメンタルエンコーダを含んだリアルタイムコントロールシステムである。制御器からのトルクコマンドは、コントロールプログラムをストアしたdSPACEから各モータドライバーへ送られ、モータドライバーはモータにパワーを送る。各モータのレゾルバーと力センサからフィードバック信号がdSPACEへ送られることにより、その時点での位置と力がわかる。図2.9での表記は次の通りである。

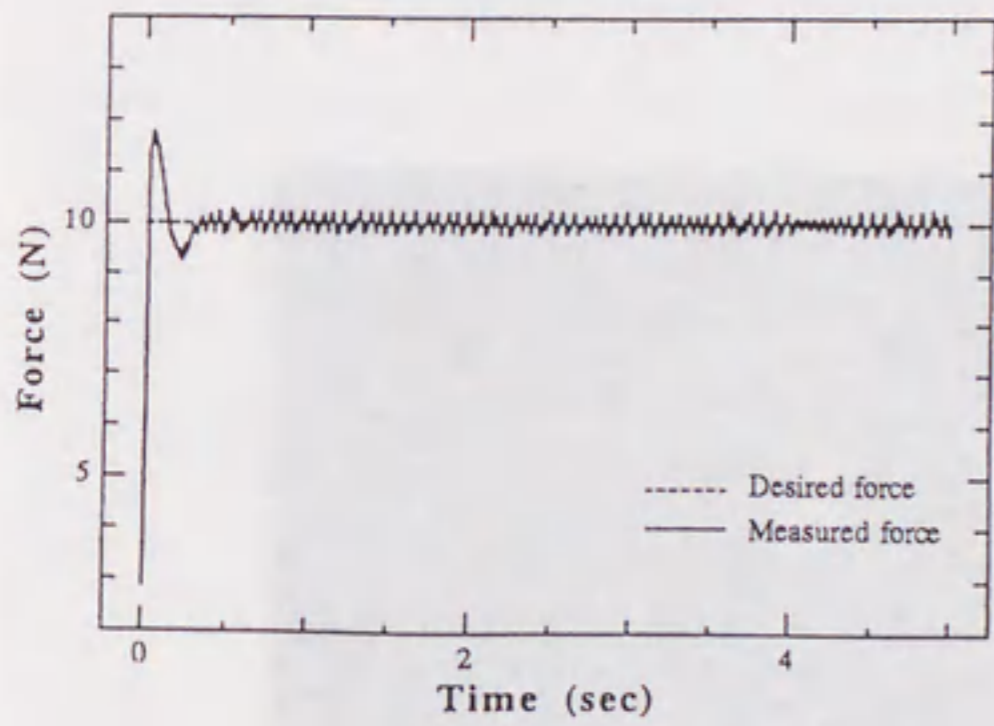
Motor 1 - NSK Megatorque Motor System 0810



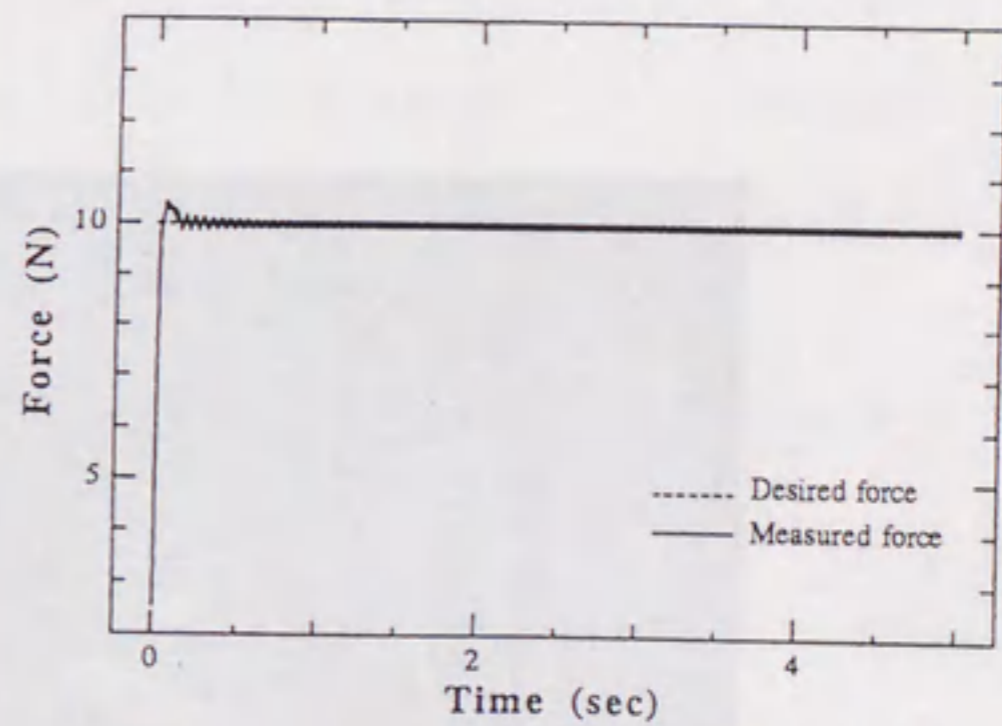
(a) $\alpha = 1.0 \times 10^{-4}$



(b) $\alpha = 0.2 \times 10^{-4}$

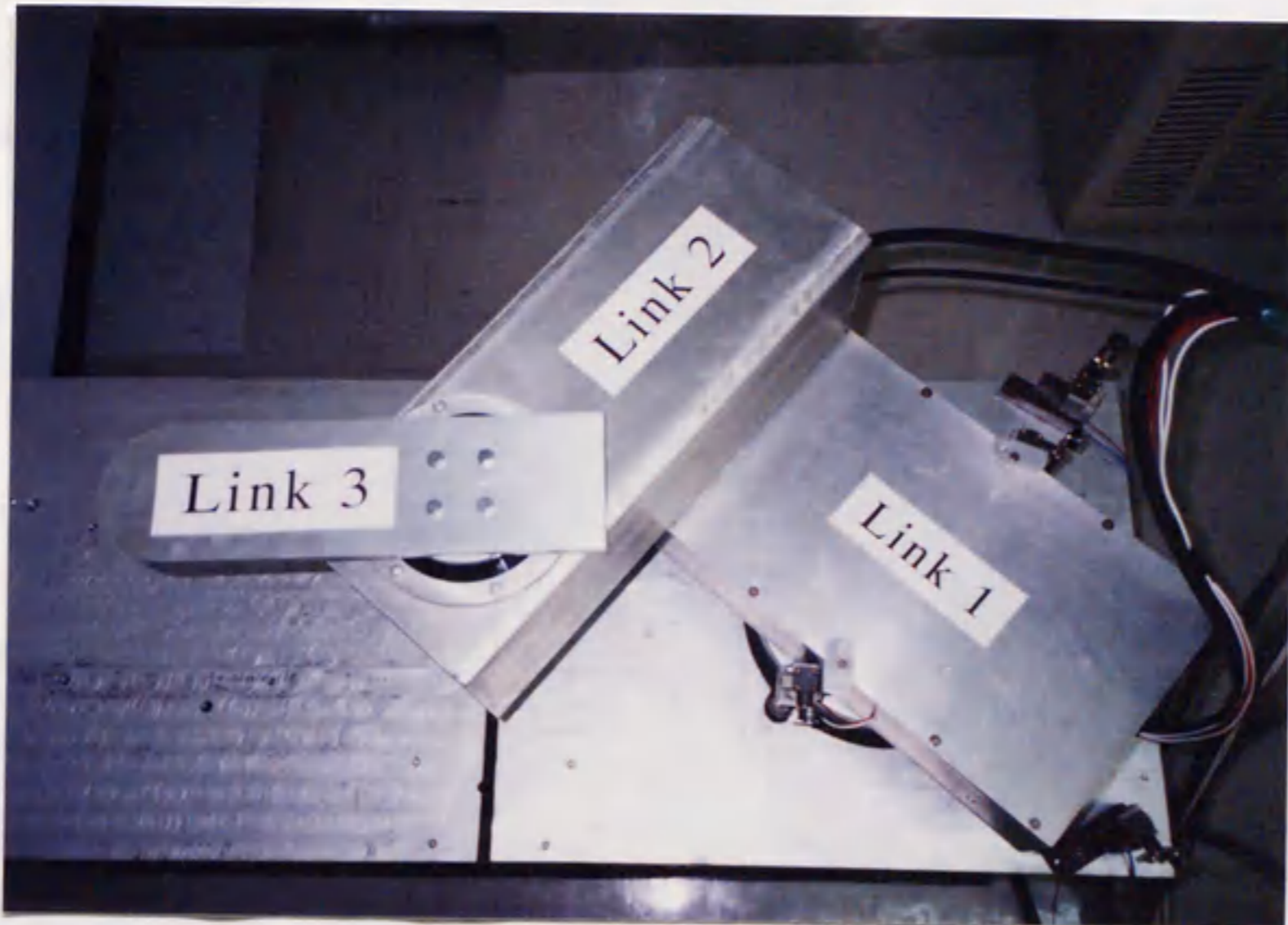


(c) $\alpha = 3.0 \times 10^{-4}$

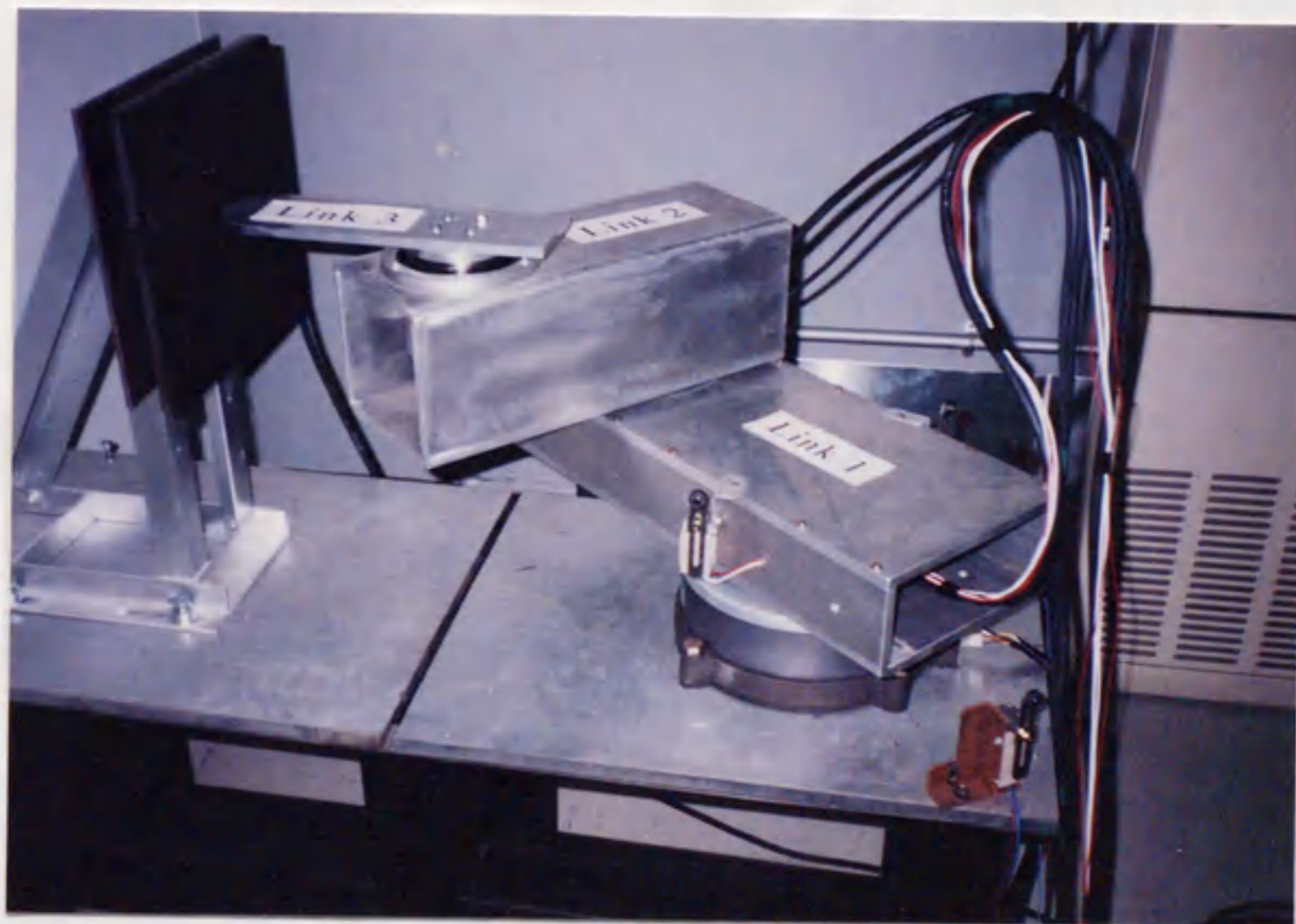


(d) $\alpha = 1.0 \times 10^{-4}$ でウエイト初期値を変更

図 2.7 シミュレーション結果



(a) 平面3自由度ダイレクトドライブロボットマニピュレータ



(b) 実験装置全体

図 2.8 実験装置の写真

(max. 88.2 Nm)

Motor 2 - NSK Megatorque Motor System 0408

(max. 9.8 Nm)

Motor 3 - Parker Hannifin Co. System 7

Type 7521.250 (max. 2 Nm)

Force Sensor - Barry Wright Co.

6-axis force sensor FS6-120A

2. 5. 2 実験結果

本実験において制御器は、環境に加える力、エンドエフェクターの角度と y 方向の位置（図 2.1 参照）を制御する。この実験の目的は、提案したニューラルネットワーク制御器を使うことにより、未知環境に対して目標通りの力を正確に加えることができることを確かめることである。未知環境として、厚さ 25.4mm、幅 300mm、高さ 40mm でばね定数 6.8 (kN/m) の発泡スチロールと、厚さ 25.4mm、幅 100mm、高さ 100mm でばね定数 50K (kN/m) のゴムを使用した。2. 3 節で説明したように、この制御器ではハイブリッド制御部の力フィードバックのゲインを環境の動特性に合わせて変える必要はないので小さな値にした。

実験に用いた式 (2.8) , (2.9) のフィードバックゲイン行列は次の通りである。

$$K_{pp} = \text{diag} (200, 200, 200)$$

$$K_{pv} = \text{diag} (80, 80, 80)$$

$$K_{fp} = \text{diag} (0.25, 0.25, 0.25)$$

図 2.10 は発泡スチロールを環境に用いたときのロボットマニピュレータの力応答の実験を開始してから数十秒後の結果である。目標力はステップシグナルを組み合わせたもので、2.5 秒毎に 10 N と 20 N の間で変化するものとした。この実験結果より、ロボットマニピュレータが目標力に追従しているのがわかる。

この実験でニューラルネットワークの補償をしなかった場合、つまり従来のハイブリッド制御の場合は、ゲインが小さすぎて目標力に追従できない。これを確認するためにニューラルネットワークによる補償無しで同じ実験を行った結果を図 2.11 に示す。この実験結果より、ゲインが小さすぎて目標力まで達していないことが確認できる。

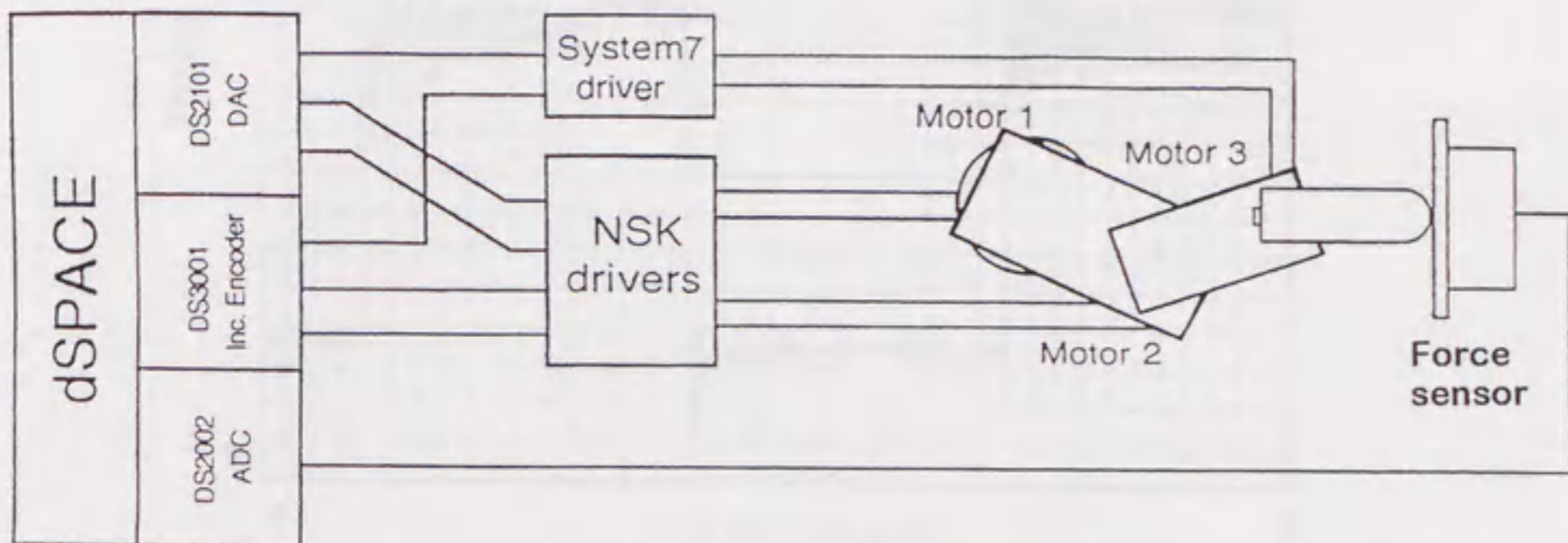


図 2.9 実験装置

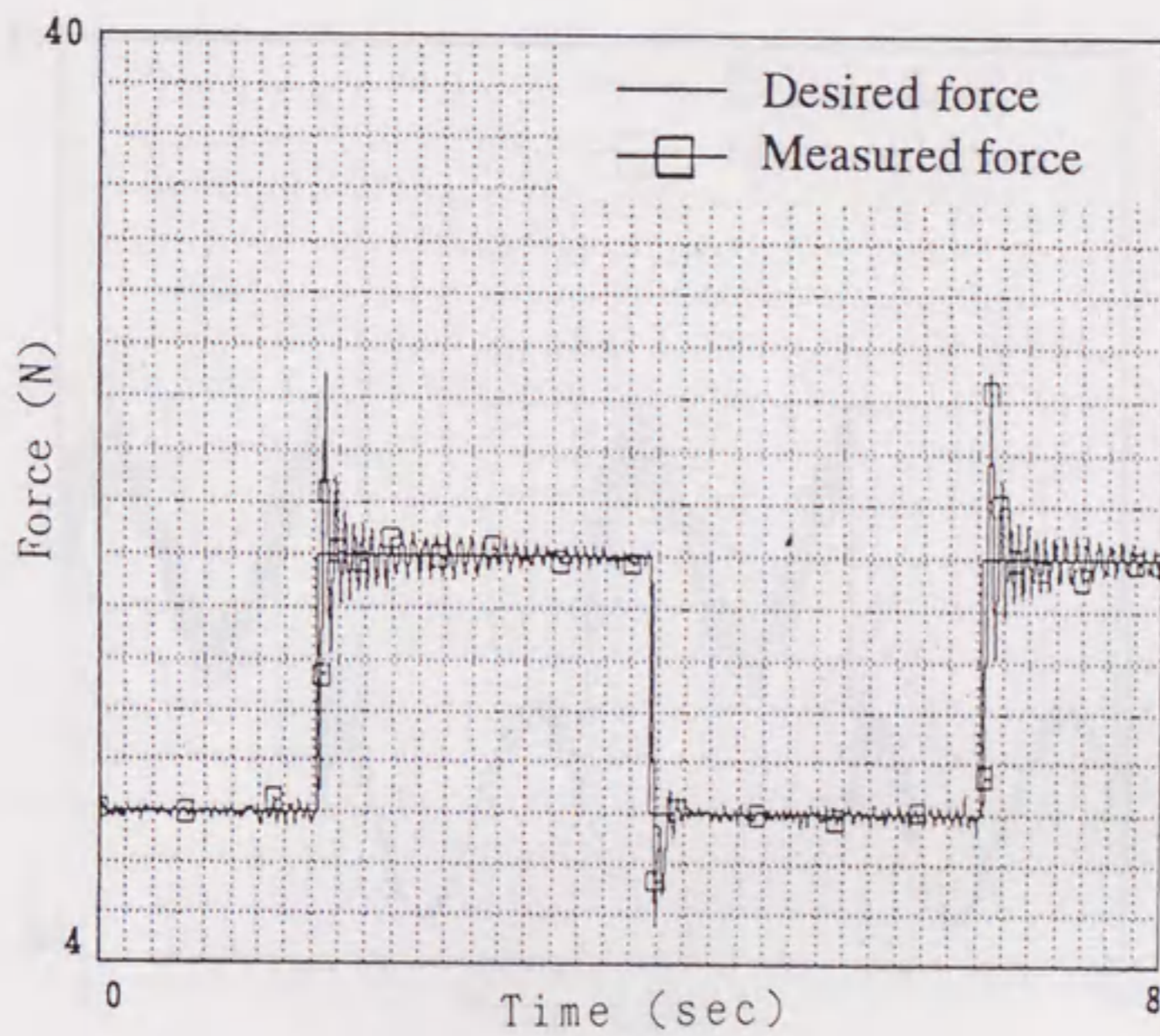


図 2.10 力応答実験結果 1

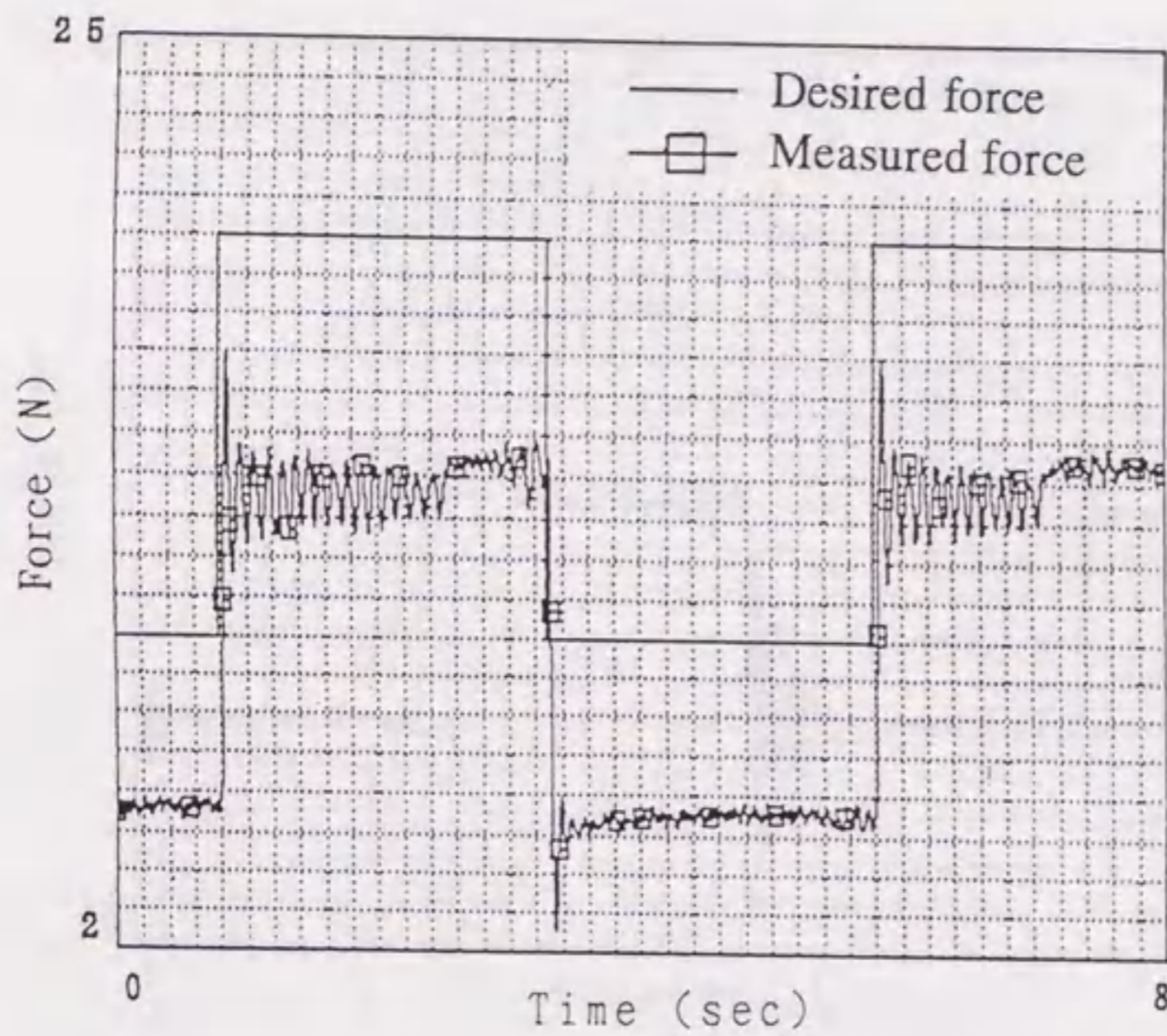


図 2.11 ニューラルネットワーク無しのカ応答実験結果

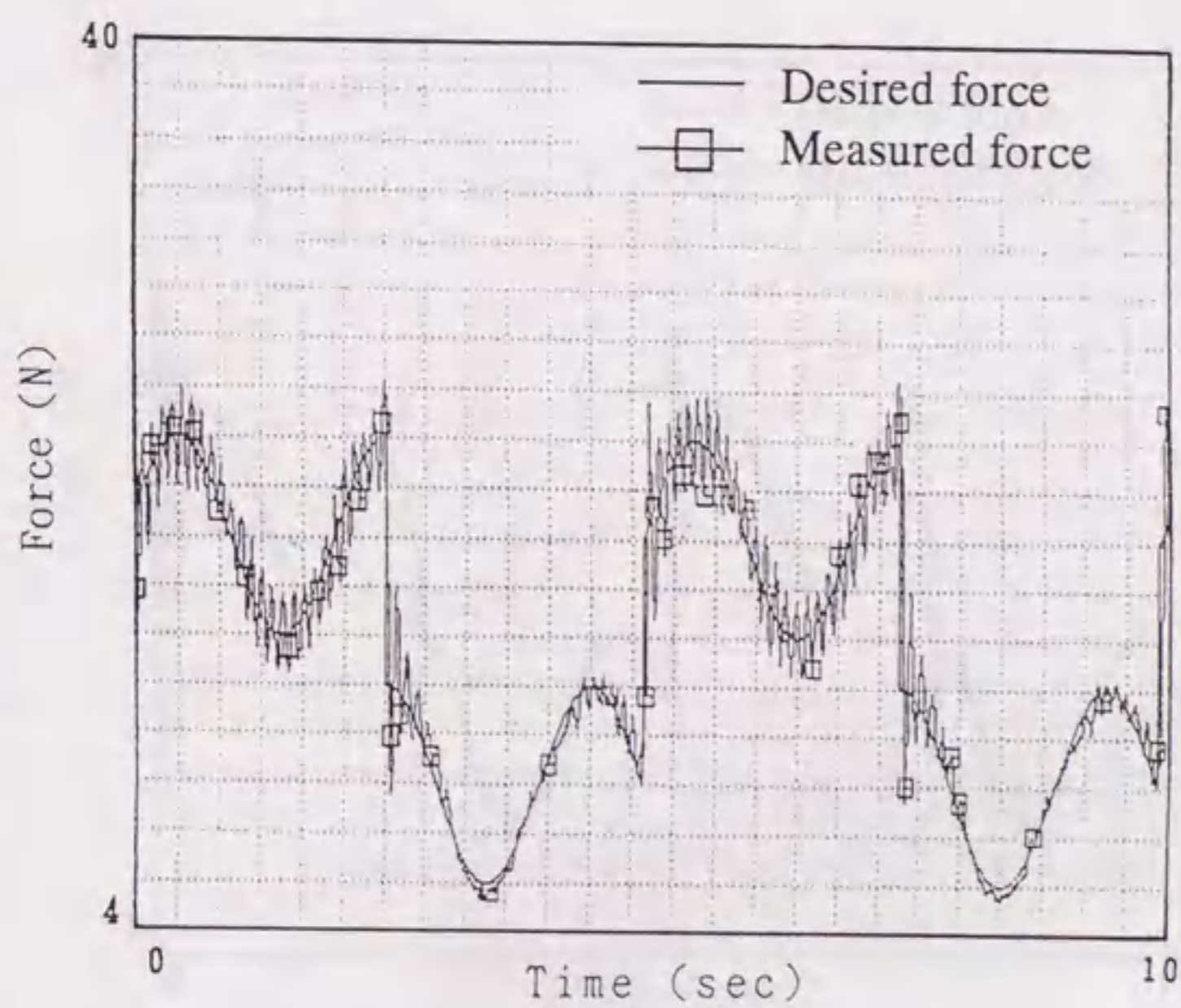


図 2.12 カ応答実験結果 2

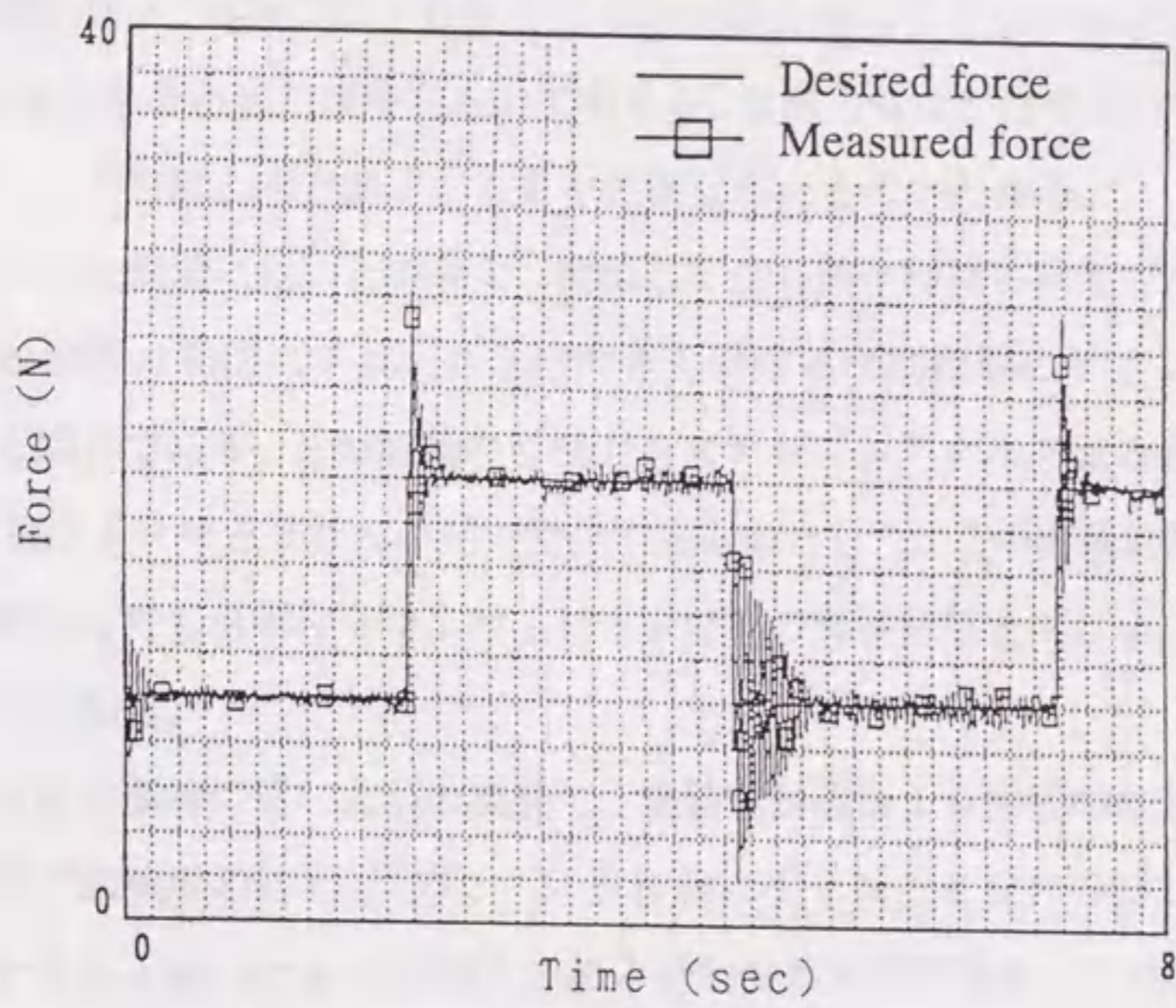


図 2.13 力応答実験結果 3

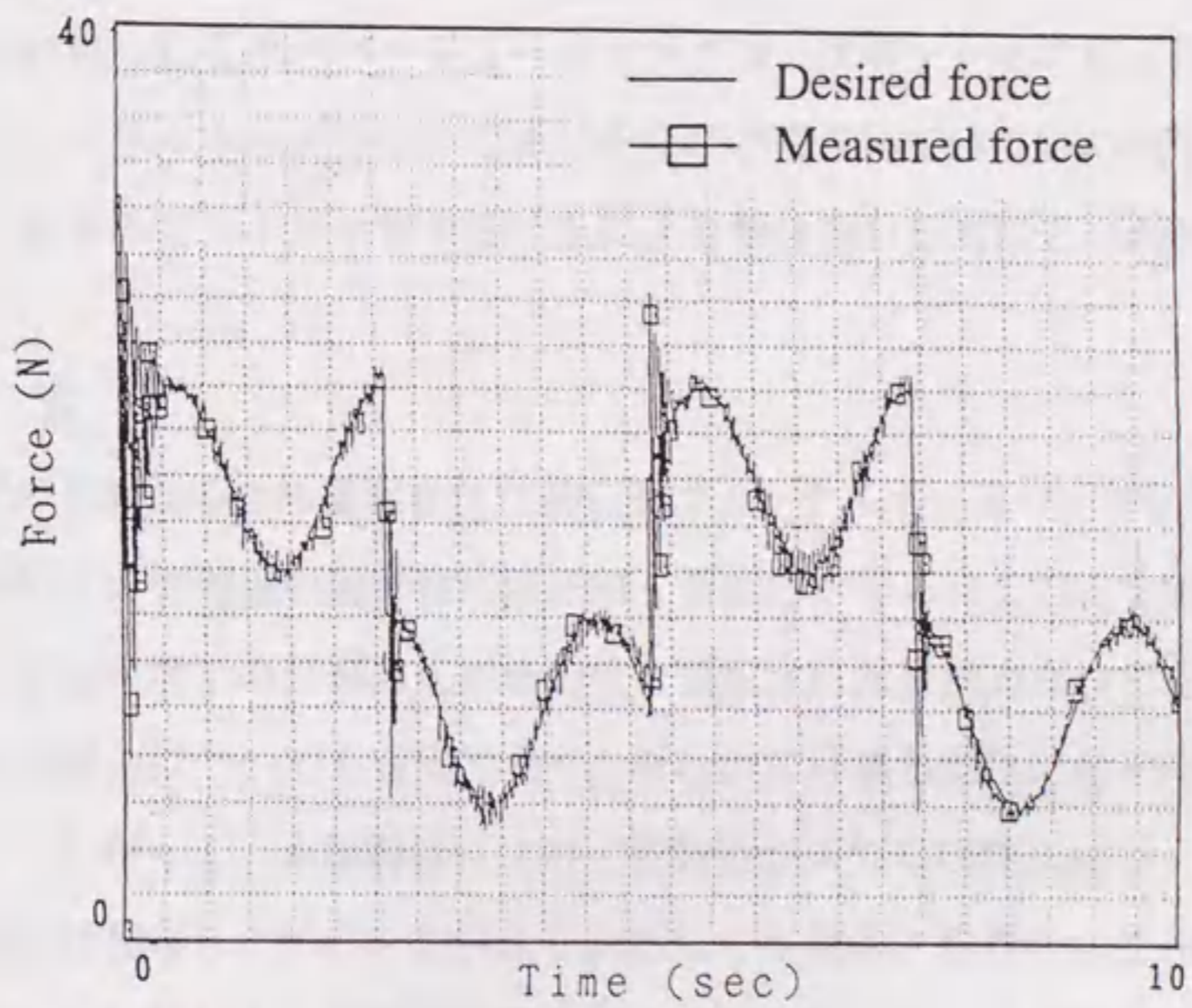


図 2.14 力応答実験結果 4

図2.12は発泡スチロールを環境に用いた別の実験で、開始してから数十秒後の結果である。目標力は2.5秒毎に10Nと20Nの間で変化するステップシグナルと $4\sin(\pi t)$ の曲線とを合わせた複雑なものであるが、目標力変化が滑らかな場合は、オーバーシュティングもほとんど無く、うまく追従しているのがわかる。

図2.13はゴムを環境に用いた実験で、開始してから数十秒後の結果である。この時の環境のばね定数は発泡スチロールとは大きく異なるが制御器のゲインは発泡スチロールの時と同じである。この実験の目標力もステップシグナルを組み合わせたもので、2.5秒毎に10Nと20Nの間で変化するものとした。この実験結果より、環境の動特性が異なってもロボットマニピュレータは前の実験結果と同じように目標力に追従するのがわかる。

図2.14はゴムを環境に用いた別の実験で、実験開始直後10秒間の結果である。目標力は図2.12での実験の時と同じで、2.5秒毎に10Nと20Nの間で変化するステップシグナルと $4\sin(\pi t)$ の曲線とを組み合わせたものである。この実験により、制御開始後約0.5秒でニューラルネットワーク制御器が未知環境に適応していることがわかる。当然のことながら、未知環境への適応に要する時間はニューラルネットワークの学習レートに依存する。

各実験結果に見られるオーバーシュティングは、ロボットマニピュレータのダイナミクスがニューラルネットワークでまだ十分に学習されていないため生じるものであると考えられるので、十分学習することにより解決されるものと思われる。

2.6 結 言

未知環境での力制御のため提案された適応型ニューラルネットワーク制御器により、基準座標において未知環境の動特性に関係無く目標力が得られることを実験により確認した。この未知環境での目標力追従能力は、固定された目標力だけでなく、時間により変化する目標力についても有効であり、滑らかな目標力変化に対しては問題なく追従できる。したがって、本制御器により、動特性が未知な環境に対し、従来のハイブリッド制御とは異なり、ゲインを小さく設定しても望みどおりの力を加えることが可能となる。しかしながら、制御器が適応するまでの間に生じる、あるいはロボットマニピュレータのダイナミクスが十分に学習されていないために生じるオーバー

第3章 ファジィ・ニューロを利用した ロボットマニピュレータの未知 環境に対する位置／力ハイブ リッド制御

3.1 緒言

現在我々の身近なところで活躍しているロボットとして、工場等で稼働している産業用ロボットが挙げられる。しかし、一般に普及している産業用ロボットは単に位置だけを制御するものであり、物の位置決め等に利用されているものがほとんどである。近年、ロボットマニピュレータに対し、従来の産業用ロボットよりも高度な作業が求められている。ロボットマニピュレータにより高度な作業、例えば研磨作業、拭き取り作業等を実現させるためには、位置と力の両方を同時に制御する必要がある。また、今後のロボットは予め設定した環境内で稼働するだけでなく、未知環境下でも作業できるような柔軟性も期待されている。

位置と力を同時に制御する方法として、主にハイブリッド制御 [42] とインピーダンス制御 [43] の二つに大別できる。ハイブリッド制御では、環境表面に対し直角な方向に力制御、その力制御の方向に対し直角方向に位置制御することにより、位置と力を同時に直接的に制御するのに対し、インピーダンス制御では速度と加速度も含めた位置と力の関係を制御するというのが主な違いである。力制御においては、環境の動特性がシステムの安定性に影響を及ぼすため、従来の制御系では環境の動特性に応じてゲインを設定してやる必要があり、動特性の未知な環境に対しては対応が難しい。そのため、力制御の研究自体は既に多くの研究例が発表されているが、力を加える環境の特性が未知なものについての研究例は少なかった。また、環境に力を加えながら位置を制御する場合、環境とロボット間の摩擦力は、環境に対し加える力の大きさにより変化するため、環境とロボット間の摩擦の補償が大きな問題となる。とくに摩擦係数の未知な環境では適切な摩擦補償を行うことは困難である。この問題を避けるた

めにロボットのエンド・エフェクタ先端にローラーを付けて実験した研究例 [56], [57] はあるが, 環境に力を加えながら位置を制御する場合に有効な摩擦補償方法を提案している研究例は見あたらない。そこで本章では, 上記の問題に対応し, 未知環境に対し位置と力のハイブリッド制御を行うため, ニューラルネットワーク及びファジィ・ニューラルネットワークを応用する方法を提案する [71], [72]。

近年, 曖昧な情報の扱いができ, 厳密な数学モデルを使うことなく制御できるファジィ制御 [10], [91], [92] や経験から学習できる能力を持つニューラルネットワークは, ロボット制御の分野で積極的に取り入れられてきた [20] ~ [39], [87], [88]。ニューラルネットワークを利用して未知環境に対し適応できるハイブリッド制御法を提案したを前章で示したが, 前章で示したような適応型のニューラルネットワーク制御器でさえも瞬時に適応できるわけではなく, 適応するまでの間オーバーシュート等危険性があった。この問題を解決するため本章ではまず, あらかじめファジィ制御の方法を用いてオーバーシュートしないような制御器を設計し, それをニューラルネットワークでモデル化し, バックプロパゲーション法で学習させる, ファジィ・ニューラルネットワーク力制御 [74], [82] を紹介する。この制御器では, 環境の動特性が設計時に予測したものよりも柔らかいか少し硬めの場合は特に問題なく力制御できるので, あらかじめ硬い環境を想定しておけばほとんどの場合問題ない。しかし, もし環境が想定したものよりかなり硬かった場合は, やはり大きなオーバーシュートが発生する等の問題が生じるので, 環境の動特性が設計時に用いたモデルよりも硬かった場合は, 制御器への入力値をオフラインで学習したニューラルネットワーク (IANN) により調整することでメンバーシップ関数の定義を瞬時に変更する方法を提案する。

また, 環境に力を加えながらの位置制御では, 摩擦補償用ニューロンと軌道制御用ファジィ・ニューラルネットワークを用い, 状況により学習を切り替えることで未知環境に対する位置制御の際の摩擦補償を実現した [84]。

通常, ロボットマニピュレータのニューラルネットワーク制御器或いはファジィ・ニューロ制御器の学習則としてバックプロパゲーション法が利用され, その評価関数としては目標値との誤差を2乗した関数が用いられる。しかし, ロボットマニピュレータの値から制御などでは目標力がステップ状に変化する場合も多く, 周期的に目

標力が変化した場合、このような評価関数で学習を継続するとオーバーシュートが生じる可能性がある。そこで本章では、ファジィ制御された評価関数を用いて効率的に学習を行う方法も提案する [71], [85]。

本制御法の有効性の確認は、図3.1に示すような平面3自由度ロボットマニピュレータモデルを用い、コンピュータによるシミュレーションで行った。

3. 2 位置/力のハイブリッド制御

図3.1に示すようなロボットマニピュレータで、環境に対し加える力とエンドエフェクタの位置及びその角度を同時に制御するため、作業空間を力制御方向と位置制御方向に分け、位置と力を同時に直接的に制御するハイブリッド制御を適用するものとする。

2. 2節で既に説明したように、平面ロボットマニピュレータを制御するためのトルクコマンドは次式のように示される。

$$\begin{aligned} \tau = M(q)J^{-1}[(I-S)u_p - \dot{J}\dot{q}] + h(q, \dot{q}) \\ + F_c \operatorname{sgn}(\dot{q}) + J^T f + J^T S u_f \end{aligned} \quad \dots\dots\dots (3.1)$$

ただし u_p は位置制御のコマンドベクトルで、 u_f は力制御のコマンドベクトルである。本稿では、 u_f は力制御器からの出力、 u_p は位置制御器からの出力であるが、位置制御のうちエンドエフェクタの角度制御には特に適応能力は必要とされないので、従来の加速度分解制御を用いることにした。このエンドエフェクタ角度制御の入力コマンド u_{pe} は次式で表される。

$$u_{pe} = \ddot{x}_d + K_v(\dot{x}_d - \dot{x}) + K_p(x_d - x) \quad \dots\dots\dots (3.2)$$

本章で提案するファジィ・ニューロを利用したハイブリッド制御系のブロック線図を図3.2に示す。この制御器を使うことにより、環境に加える力とエンドエフェクタの角度と位置が同時に制御される。ただし、エンドエフェクタの角度制御には適応能力は必要ないので加速度分解制御 [103] を用いた。

また、2. 2節で説明した理由により、ハイブリッド制御器に従来の制御則を利用することは好ましくない。そこで本章では、ハイブリッド制御内の位置制御と力制御の両方に適応能力を持たせるため、ニューラルネットワークとファジィ・ニューラルネットワークを応用する方法を次の3. 3節と3. 4節で説明する。

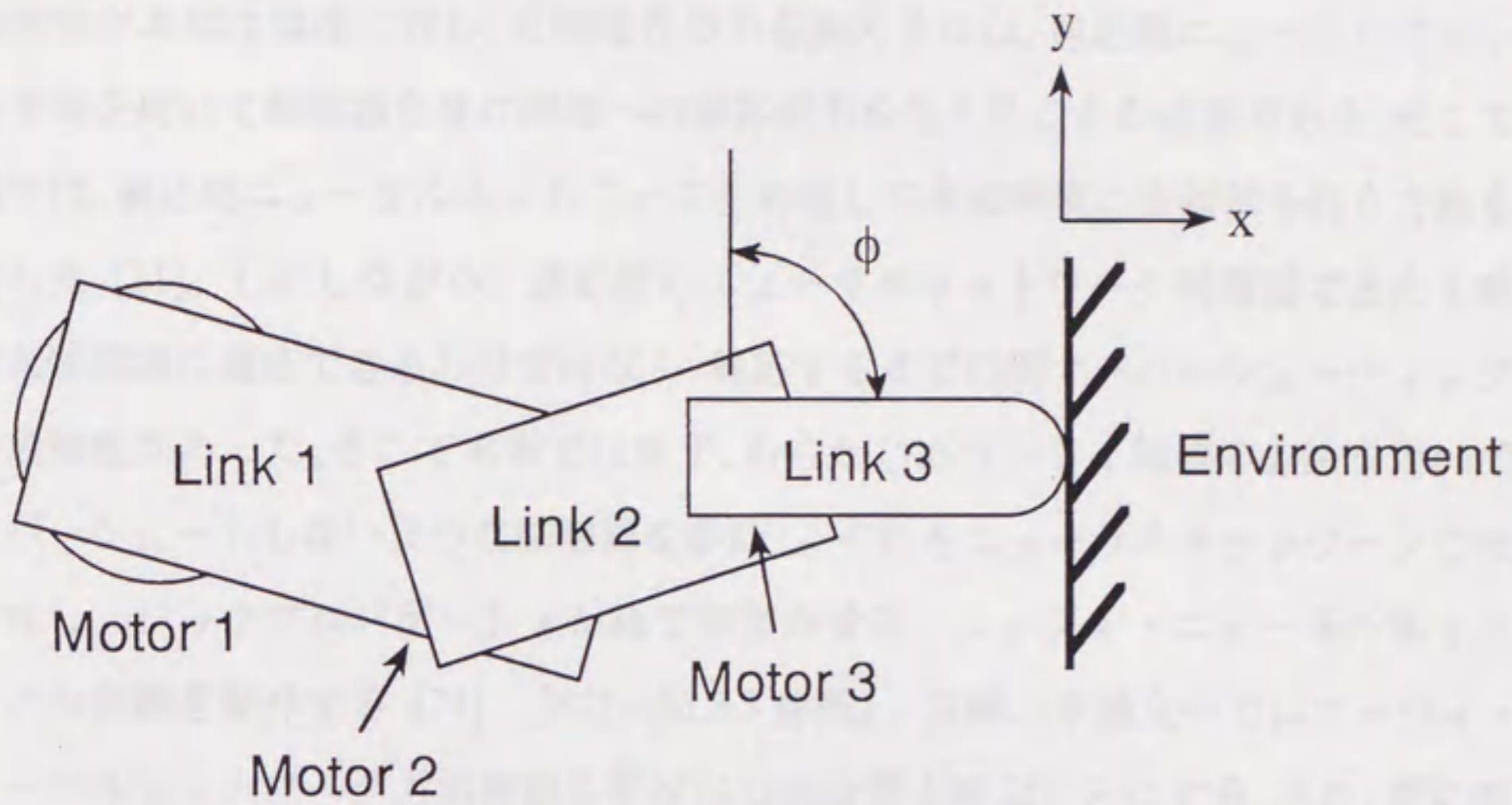


図 3.1 平面 3 自由度ロボットマニピュレータ

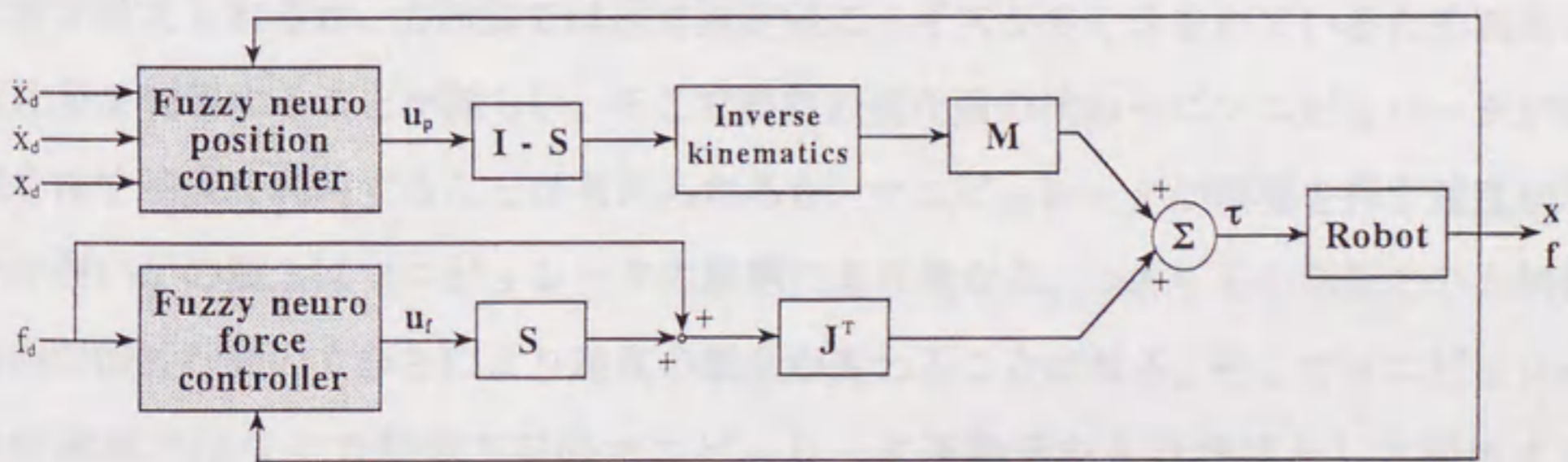


図 3.2 ファジィ・ニューロ位置/力ハイブリッド制御のブロック線図

3. 3 未知環境での力制御

動特性が未知な環境に対し、目標通りの力を加えるには、適応型ニューラルネットワーク等を用いて制御器自身に環境への適応能力を与えることが必要である。そこで前章では、適応型ニューラルネットワークを利用して未知環境に力制御を行う方法を検討した [21]。しかしながら、適応型のニューラルネットワーク制御器でさえも瞬時に未知環境に適応できるわけではなく、適応するまでの間オーバーシュート等の危険性があった。そこで本章ではまず、あらかじめファジィ制御の方法を用いてオーバーシュートしないような制御器を設計し、それをニューラルネットワークでモデル化し、バックプロパゲーション法で学習させる、ファジィ・ニューラルネットワーク力制御を紹介する [74], [82] (図 3.3 参照)。以降、本論文中ではファジィ・ニューラルネットワーク力制御器を FNN 力制御器と呼ぶことにする。また、更に新しい手法として、環境の動特性に応じ FNN 力制御器への入力値を調整するニューラルネットワーク (IANN) をオフラインで学習させ、FNN 力制御器に組み込むことを提案する。本章で提案する FNN 力制御器の詳細を図 3.3 に示す。ここで Σ のニューロンは入力の和を出力し、 Π のニューロンは入力の積を出力するものである。

3. 3. 1 ファジィ・ニューロ力制御器

通常ロボットマニピュレータのファジィ制御では、入力情報として誤差量とその変化量が考えられるが、力制御では通常測定値にノイズが多く含まれているため誤差の変化量を利用することが難しい。そこで誤差の変化量の代わりにマニピュレータが環境を押す速度を利用することが考えられるが、マニピュレータが環境を押す速度が早いか遅いかの概念はマニピュレータの形態により異なる。つまりその形態での力制御方向の慣性行列の大きさにより速度の概念が変わることになる。そこでマニピュレータの速度ではなく力制御方向のマニピュレータ運動量を入力情報として利用する [79] ~ [82]。

$$M_o = M_s(q)v \dots\dots\dots (3.3)$$

ここで、 M_o はマニピュレータ運動量、 v はマニピュレータの速度、 $M_s(q)$ はマニピュレータの関節角度が q のときの基準座標系での慣性行列であり、次式のように表される。

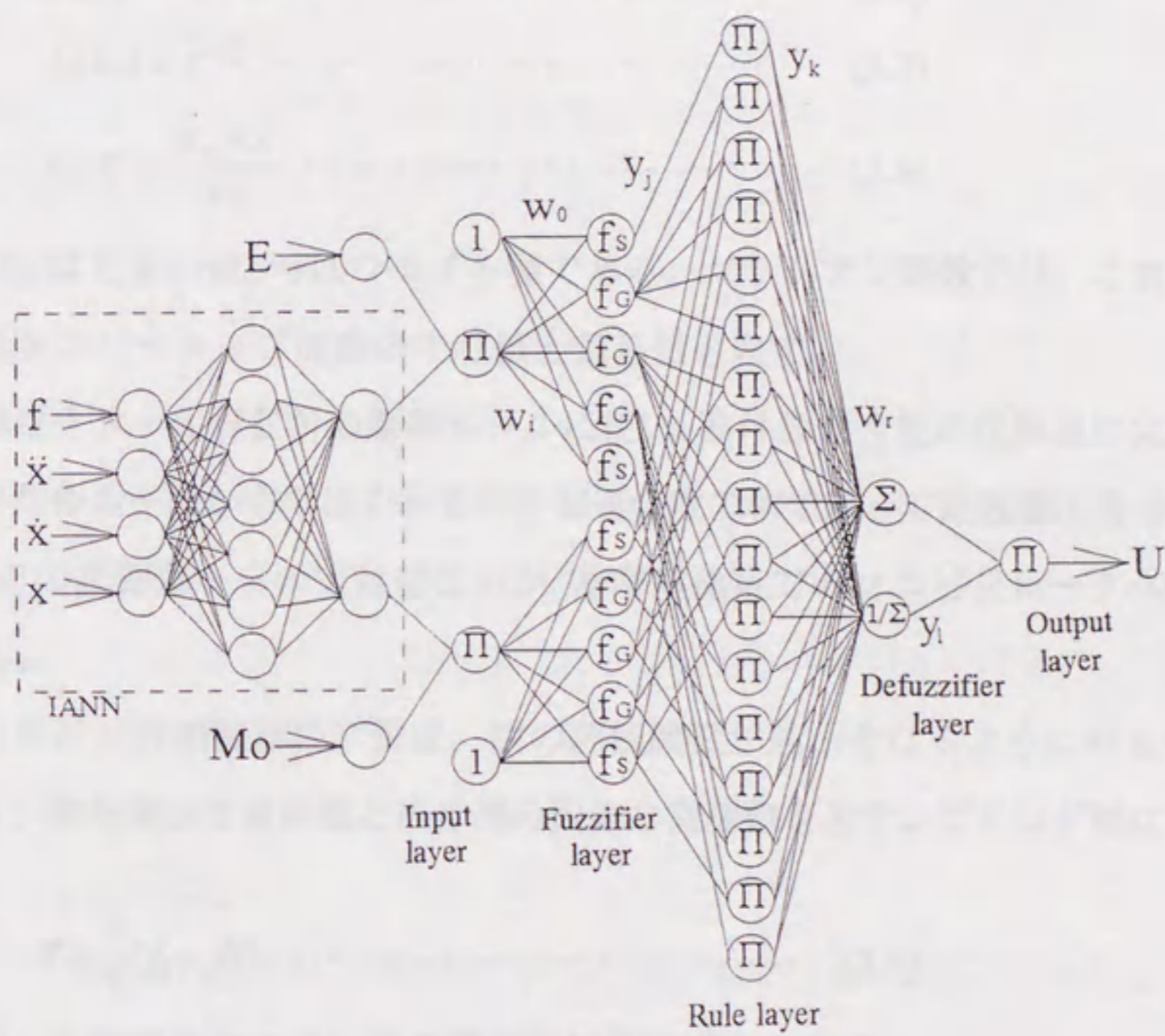


図 3.3 ファジィ・ニューラルネットワーク力制御器

$$M_x(q) = J^{-T}(q)M(q)J^{-1}(q) \dots\dots\dots (3.4)$$

前件部のファジィ変数は5種類 (PB, PS, ZO, NS, NB) とし, 入力変数は目標位置との誤差量 E と力制御方向のマニピュレータ運動量 M_0 の2種類とする。ファジィ変数は, 式 (3.5) で表されるシグモイド関数と式 (3.7) で表されるガウシアン関数の両方を用いることにより単層のニューラルネットワークで前件部を構成するものとする。

$$f_S(u_S) = \frac{1}{1 + e^{-u_S}} \dots\dots\dots (3.5)$$

$$u_S(x) = w_0 + w_r x \dots\dots\dots (3.6)$$

$$f_G(u_G) = e^{-u_G^2} \dots\dots\dots (3.7)$$

$$u_G(x) = \frac{w_0 + x}{w_i} \dots\dots\dots (3.8)$$

ここで, w_0 はしきい値, w_i はウエイト値である。ガウシアン関数では, これらの値はそれぞれメンバーシップ関数の中心値と偏差値を表す。

後件部は I F - T H E N の制御ルールに従い, 前件部適合度の代数積に定数値をかけ合わせた各ルールからの出力和を前件部適合度の和で割って定数値出力を得るものとする。この制御器からの定数値出力が, 基準座標系でのマニピュレータへの力指令 U である。

通常, FNN力制御器の学習は, 次の評価関数が最小となるようにバックプロパゲーション法を用いて前件部と後件部の両方の微調整を各サンプリング毎に行うものとする。

$$Y = \frac{1}{2}(f_d - f)^2 \dots\dots\dots (3.9)$$

この場合, 各部でのウエイト値の変化量は次の通りである。

$$\text{後件部: } \Delta w_r = \eta(f_d - f)y_j y_k \dots\dots\dots (3.10)$$

$$\text{前件部: } \Delta w_i = \eta \sum \frac{\partial e_j}{\partial y_{ij}} f'_{(w)} u' \dots\dots\dots (3.11)$$

ここで η は学習係数, f_d は目標力, f は実際に測定された力, y_{ij} は前件部出力, e_j は前件部出力誤差, $f'_{(w)}$ と u' は式 (3.5), (3.7) で表されるアクティベーション関数とその入力の微分である。

この学習により, 設計したファジィ制御則が完全でなくても, ロボットマニピュ

レータの関節の摩擦が完全に補償されていなくても、また外乱が作用したり環境の動特性が変化しても、目標通りの力を未知環境に加えることが可能になる。

3. 3. 2 入力値調整ニューラルネットワーク (IANN)

本FNN力制御器により、ほとんどの場合オーバーシュートすることもなく目標通りの力を制御できる。しかし、もし対象物の動特性が想定していたものよりもかなり硬かった場合は大きなオーバーシューティングが発生する等の問題が生じる。しかし、もし未知環境に力を与え始めた早い段階で、環境の変位量とその変位により発生した力との関係から環境が想定していたものより硬いか柔らかいかを判断し、それに応じて制御器の前件部を調整すればそれらの問題を解決できる。そこで本章では、FNN力制御器への入力情報である誤差とマニピュレータ運動量の二つ両方を、環境の硬さに応じ調整するニューラルネットワークIANNを提案する。

IANNでFNN力制御器の入力値を瞬時に調整することにより、設計時に定義されたメンバーシップ関数を環境の硬さに応じ、横に広げたり狭めたりしたような形状に変更するのと同じ効果が得られる(図3.4参照)。つまり制御器の前件部の定義の変更が瞬時にできる。例えば、もし環境が想定していたものよりも硬かった場合は、その硬さ加減により制御器への入力情報である目標力との誤差量を小さく、力制御方向のマニピュレータの運動量を大きく変換してやることにより、実際にはまだ誤差が大きい時点で誤差が小さいときのようにきめ細かく制御され、環境を押す速度の定義も変更できるようになる。

IANNへの入力情報は、環境の変位、環境の変位速度、環境の変位加速度、そして環境からの反力の四つとした。IANNの隠れ層と出力層のアクティベーション関数は、シグモイド関数を使うことにした。IANNからの出力は、FNN制御器の入力情報である誤差量 E に対する調整係数 K_E と力制御方向のマニピュレータ運動量 M_0 に対する調整係数 K_{M_0} の二つとした。IANNからの出力である二つの調整係数は正の数とし、出力層にシグモイド関数を用いることにより予め設定した上限と0の間で変化するものとした。環境の動特性が設計時と同じだったら二つの調整係数は両方とも1を出力するように学習させる。IANNの詳細を図3.5に示す。

IANNの学習は、オフラインでバックプロパゲーション法により予め学習してお

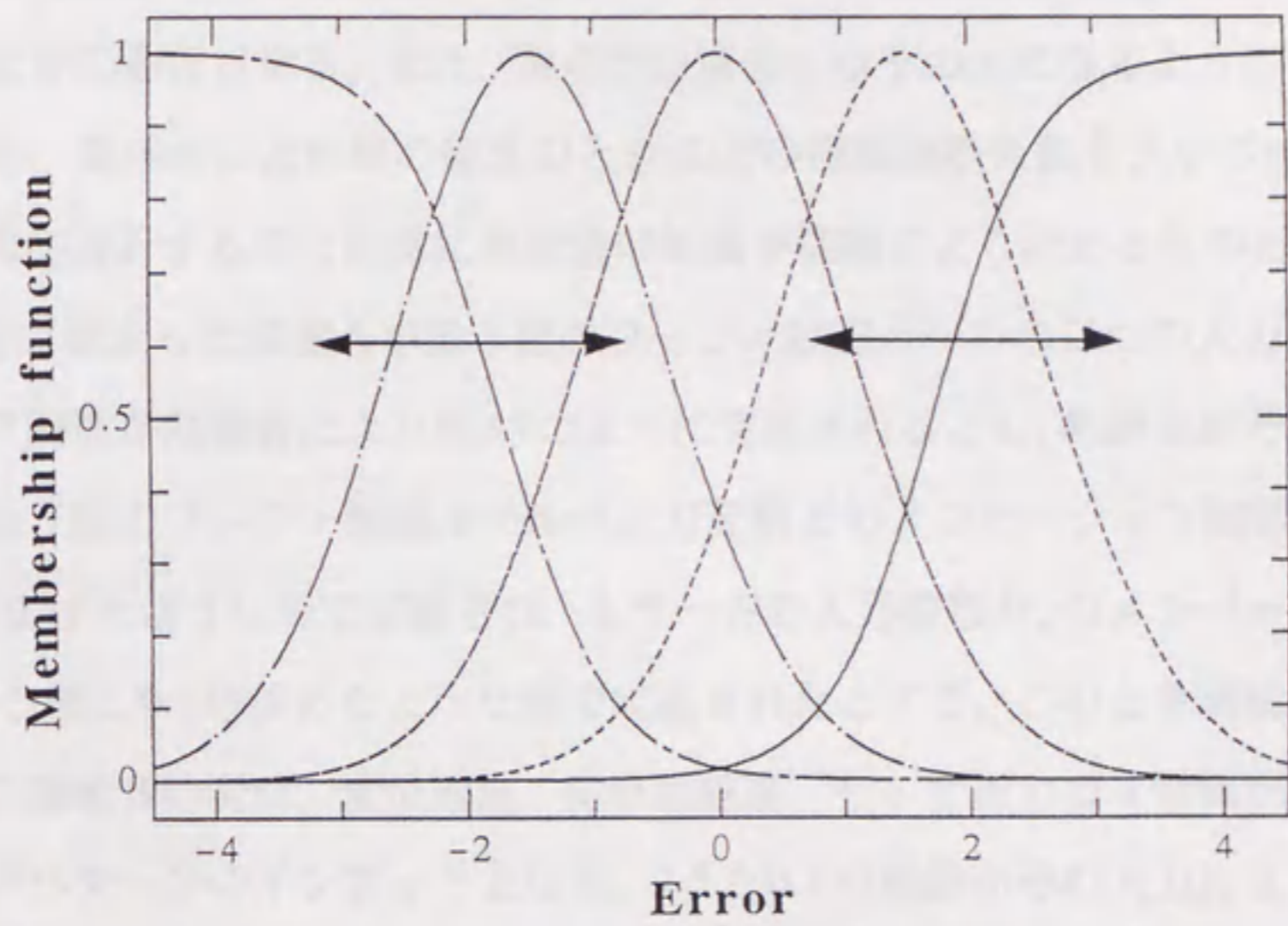


図 3.4 メンバシップ関数の変化

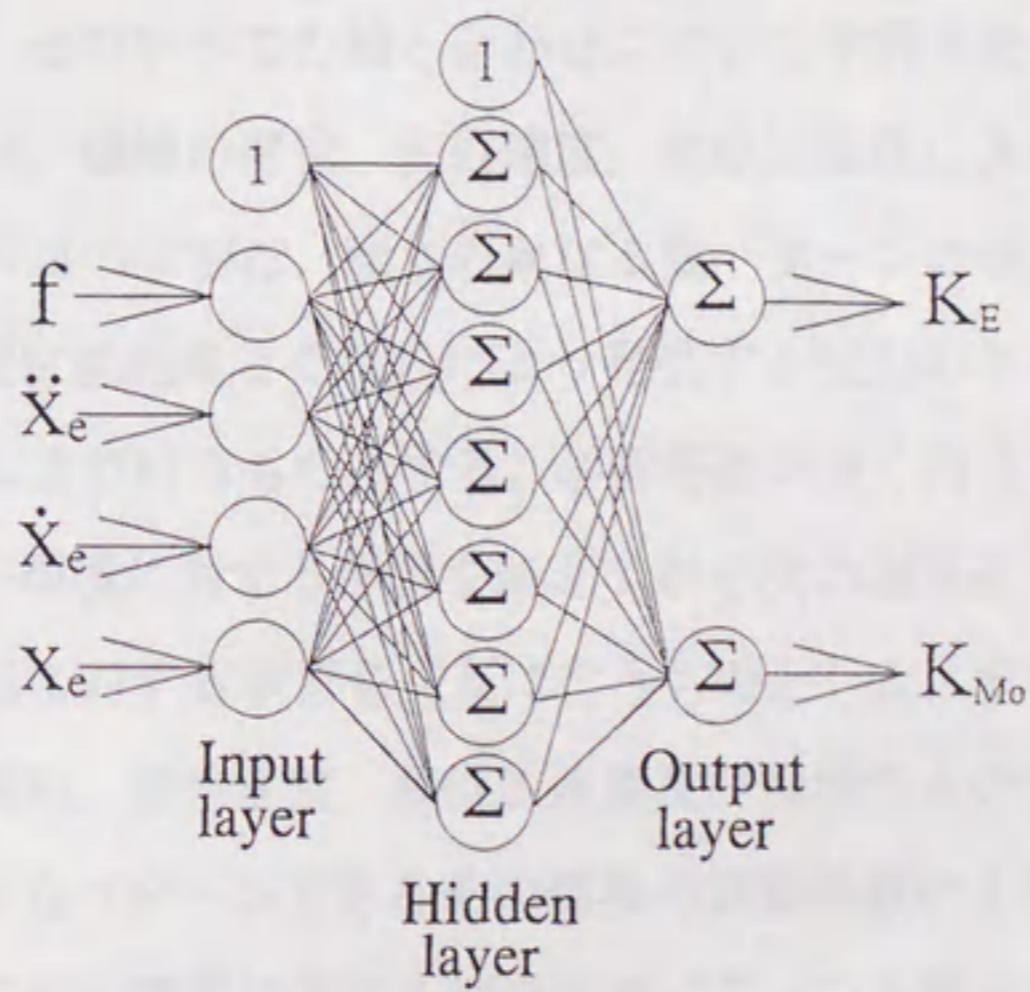


図 3.5 入力調整ニューラルネットワーク (IANN)

くものとした。教師信号として、設計時に想定した環境のよりも硬いものと柔らかいものを数パターン用意し、設計時に想定していたものより硬ければ硬いほど誤差量調整係数 K_E を1より小さく、力制御方向のマニピュレータ運動量調整係数 K_{M_0} を1より大きくなるように学習させる。また、柔らかい場合にはその逆になるように学習させる。このとき、具体的にどれ位の硬さのときにどの程度調整係数を大小させるかは、ファジィ制御を設計するのと同様に熟練者の知識や経験により決めるものとする。例えば、設計時に想定した環境Aを扱う際のファジィ制御ルールの中の二つの入力変数のメンバーシップ関数が熟練者により図3.4のように定義されるとし、熟練者がそれより硬い環境Bを扱う際のファジィ制御ルールの入力変数 E のメンバーシップ関数が、図3.4を横に1.5倍広げたような形で定義され、もう一方の入力変数 M_0 のメンバーシップ関数は、図3.4を横に0.7倍狭めたような形で定義されたとする。このとき環境Bに力を加えたときの環境Bの変位、変位速度、変位加速度、そして反力の4情報が環境Bについての学習パターンのインプットとなり、1.5と0.7の逆数がその入力により発生すべき出力 K_E 、 K_{M_0} となる。ここで、環境Bについての学習パターンの目標出力は固定されるが、入力については無限の組み合わせがある。そこで、入力パターンの組み合わせとして環境Bの変位、変位速度、変位加速度各々の実際に変化する範囲を数十～数百に分割し、そのすべての組み合わせについて学習させるものとする。このとき環境からの反力は、環境の変位、変位速度、変位加速度により決まる。すなわちオフラインでのIANNの学習は、硬さの異なる数パターンの環境に対し、その環境の変位、変位速度、変位加速度とそれらにより発生する反力の各々を前述の組み合わせで変化させることにより行うものとする。学習を効率良く行うため、学習パターンの提示方法は、一つの環境に対する学習を終えてから次の環境について学習を行うのではなく、全ての環境に対する学習を交互に行うものとした。すなわち、一つの環境に対してその環境の変位、変位速度、変位加速度及び環境からの反力の1入力パターンと、それに対する出力パターンであるその環境の調整係数の1学習パターンを提示して学習した後、次にその環境に対する別の学習パターンを提示するのではなく、別の環境に対して1学習パターンを提示し学習させるものとする。

一方、IANNの汎化、すなわち未学習の学習に対しても適切な値を出力できるかどうかは重要な問題である。本IANNのような3層ニューラルネットワークにより

非線形マップが構築できることは既に証明されている [93]。しかし、汎化能力や過学習 [94] の問題については幾つかの方法 [95] が発表されているが、未だ確立した評価法はない。一般的には学習パターンが多ければ多いほど汎化能力が向上されるので、実用の際には実際に扱うであろう全ての材料に対して学習しておくことが望まれる。

3. 4 未知環境上での位置制御

環境に力を加えながら同時に位置を制御しようとする場合、いかに環境とロボット間のクーロン摩擦の補償を行うかが問題となる。クーロン摩擦の大きさは次式で示すように動摩擦係数と環境表面に垂直に作用する力との積であり、環境に加える力によっても変化するので補償するが難しい。

$$F_{cr} = \mu_k f \dots\dots\dots (3.12)$$

ここで、 F_{cr} はクーロン摩擦、 μ_k は動摩擦係数、 f は環境表面に垂直に作用する力である。

この章では、摩擦補償用ニューロンを持つファジィ・ニューラルネットワーク位置制御器を利用して、前章で説明したFNN力制御器により環境に対し目標通りの力を加えながら環境表面上でロボットマニピュレータの位置を制御する際に効果的な摩擦補償方法を説明する。以降、ファジィ・ニューラルネットワーク位置制御器をFNN位置制御器と呼ぶことにする。FNN位置制御器は、ロボットマニピュレータの軌道を制御する軌道制御用ファジィ・ニューラルネットワーク部と、環境とロボット間の摩擦補償を行う摩擦補償用ニューロン部の二つで構成されている。本章で提案するFNN位置制御器の詳細を図3.6に示す。この位置制御器の二つの部は、同時に学習することはなく、状況により学習の切り換えを行うものとした。即ち、位置制御のための力が加えられているにも関わらず、ロボットマニピュレータが環境表面上で動かせない場合に摩擦補償用ニューロン部が学習し、ロボットマニピュレータが運動している場合は軌道制御用ファジィ・ニューラルネットワーク部が学習するものとする。この様子を図3.7に示す。

3. 4. 1 軌道制御用ファジィ・ニューラルネットワーク

軌道制御用ファジィ・ニューラルネットワーク部では、入力情報として目標位置との誤差量とその変化量を用い、目標軌道通りに運動するために必要なマニピュレータへの基準座標系での力指令 U を出力する。入力情報として目標位置との誤差量とその

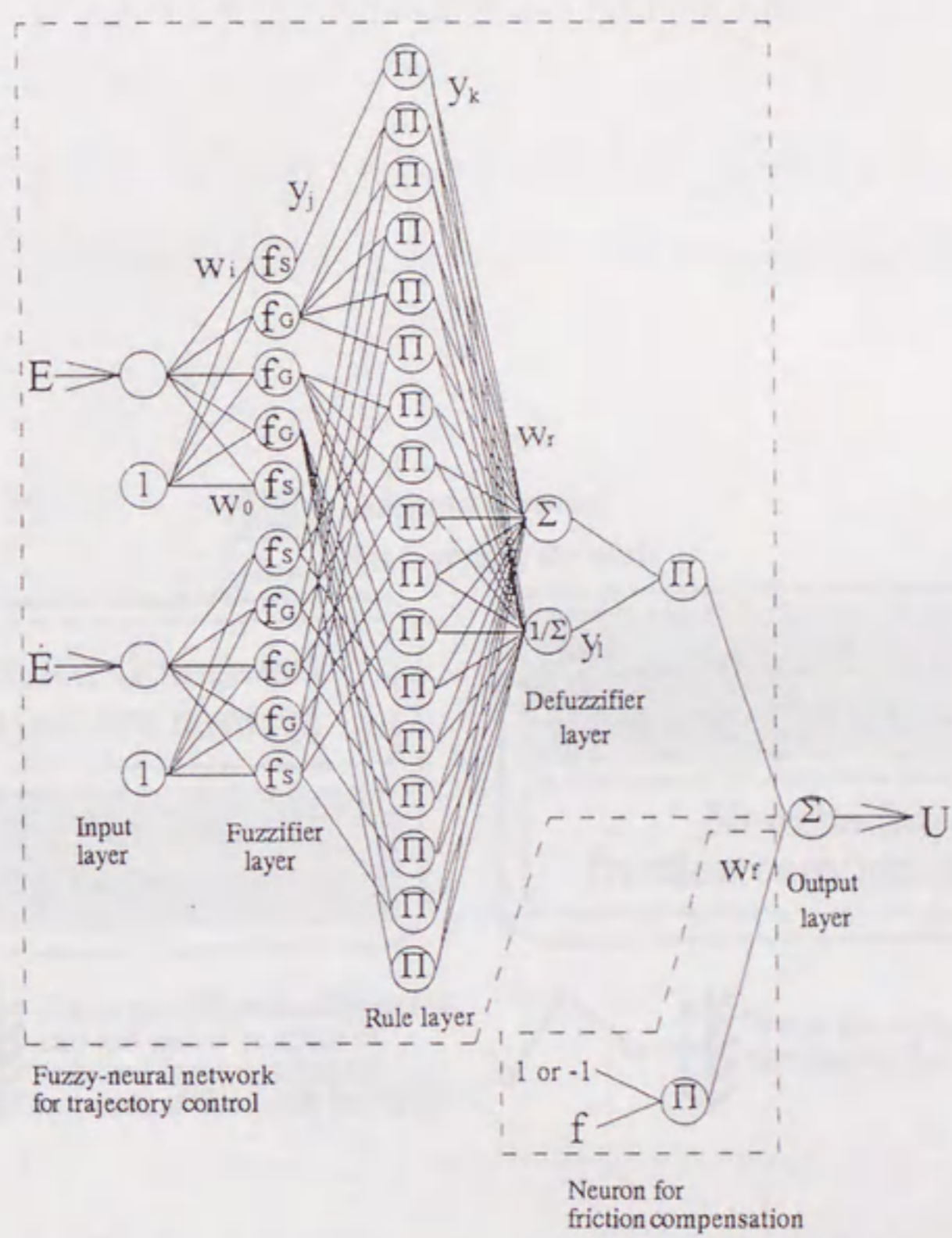


図 3.6 ファジィ・ニューラルネットワーク位置制御器

stability of the system. The proposed method is based on the idea of switching learning. In this method, the fuzzy neural network is used for trajectory control when the robot manipulator is moving on the environment. When the robot manipulator cannot move in spite of applied force command because of Coulomb friction, the neuron for friction compensation is used for trajectory control. This method is called switching learning.

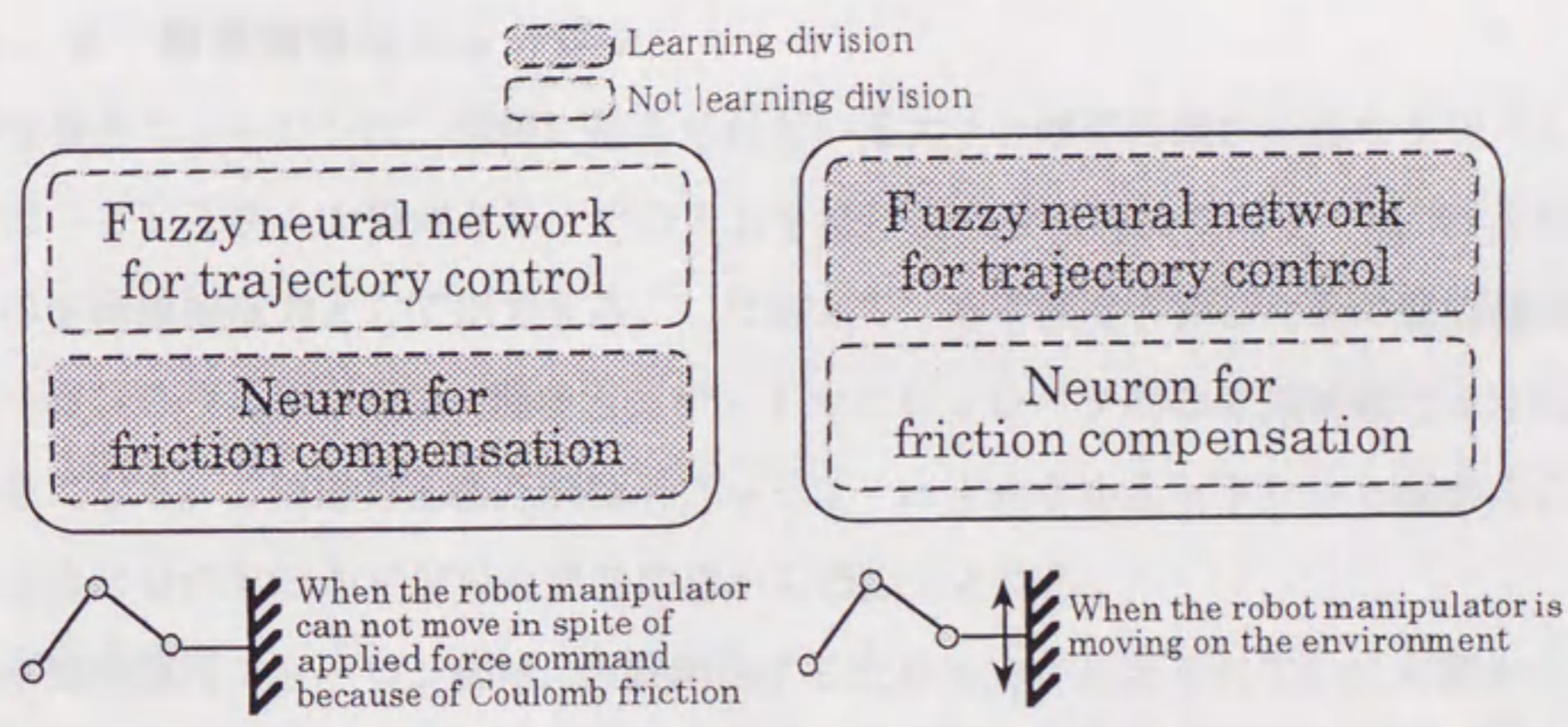


図 3.7 切り替え学習法

変化量を用いることと、学習のための評価関数が次式で表される位置誤差の2乗関数であること以外は、構成及び学習方法共に3.3.1節で説明した通常のFNN力制御器の場合と同じであるのでここでの説明は省略する。

$$Y = \frac{1}{2}(x_d - x)^2 \dots\dots\dots (3.13)$$

ここで、 x_d は目標位置、 x は実際に測定された位置である。ただし、軌道制御用ファジィ・ニューラルネットワーク部は、ロボットマニピュレータのエンドエフェクタ先端が運動しているときのみ学習を行い、エンドエフェクタが動いていないときは学習しない（学習係数が0となる）。

この軌道制御用ファジィ・ニューラルネットワーク部の学習により、たとえ外乱が多少作用しても或いは摩擦補償が完全でなくても、目標通りの位置へ移動させることが可能となる。

3.4.2 摩擦補償用ニューロン

摩擦補償用ニューロンは、環境に加えられている力 f と摩擦補償が必要な方向（1あるいは-1）とを入力情報とし、その入力を掛け合わせた値にウエイト値 Wf を掛けたものを摩擦補償力として出力する。したがって、ある程度学習した後の摩擦補償用ニューロンのウエイト値は、環境とロボットマニピュレータ間の摩擦係数とほぼ同じ大きさになる。この出力と軌道制御用ファジィ・ニューラルネットワーク部からの出力とを合わせたものがFNN位置制御器からの出力となる。

この摩擦補償用ニューロン部は、位置制御するための力が与えられていにも関わらずロボットマニピュレータのエンドエフェクタが動かない場合のみバックプロパゲーション法により学習するものとし、ロボットマニピュレータが運動しているときは軌道制御用ファジィ・ニューラルネットワーク部が学習するものとする。したがって、ロボットマニピュレータが運動し始めた後、学習が軌道制御用ファジィ・ニューラルネットワークに切り替わったら、摩擦補償用ニューロンはそれまでに学習した摩擦係数に従った摩擦補償力を出力し続け、ロボットマニピュレータの運動方向が反転したら摩擦補償用ニューロンへの入力の正負も反転するので、運動の方向に関わらず摩擦補償を行うことができる。

摩擦補償用ニューロン部の学習のための評価関数は、軌道制御用ファジィ・ニュー

ラルネットワーク部と同様で、式 (3.13) で示したような位置誤差の 2 乗関数である。摩擦補償が必要な方向が 1 の場合のウェイト値の変化量は、

$$\Delta w_f = \eta(x_d - x)f \quad \dots\dots\dots (3.14)$$

であり、摩擦補償が必要な方向が -1 の場合のウェイト値の変化量は、

$$\Delta w_f = -\eta(x_d - x)f \quad \dots\dots\dots (3.15)$$

となる。

摩擦補償用ニューロン部が学習しているときは、軌道制御用ファジィ・ニューラルネットワーク部は学習しないがファジィ制御器として設計された通りの出力はするので、ここで学習した摩擦係数は実際の摩擦係数よりも小さな値になる。しかし、摩擦補償の過不足は軌道制御用ファジィ・ニューラルネットワーク部の学習により補償される。

3. 5 ファジィ・ニューラルネットワークの効果的学習

3. 3 節と 3. 4 節でも示した通り、バックプロパゲーション法を用いて学習する際の評価関数として、通常、式 (3.9) や式 (3.13) のような目標値との 2 乗誤差が用いられ、これを最小にするための学習が行われる。しかし、力制御などでは目標力がステップ状に変化する場合も多く、そのような目標力が周期的にステップ変化する場合は、上述の評価関数を小さくするように長時間学習を進めると、より早く目標力まで到達しようとして目標力に達するまでの間の速度が大きくなり過ぎ、その結果オーバーシュートが発生してしまう可能性がある。これを抑えるための方法として評価関数に変化速度を加えることが考えられる。しかし、目標力がステップ状に変化する場合は、目標速度を別に設定してやらない限り目標力変化速度は目標力が変化した瞬間を除けばゼロとなる。この目標力変化速度に対して誤差をなくすということはロボットマニピュレータの動きを止めてしまうことになってしまので、目標力の獲得ができなくなる。そこで本章では、誤差がゼロに近づいたときにだけ評価関数の速度項の影響が大きくなるようにファジィ制御する方法を提案する [71], [85]。このときの評価関数は次式のようになる。

$$Y(k) = \frac{1}{2} \{ [f_d(k) - f(k)]^2 + K_f [(f_d(k) - f(k)) - (f_d(k-1) - f(k-1))]^2 \} \quad \dots\dots\dots (3.16)$$

ここで、 K_f はファジィ制御される目標変化速度の調整係数で、目標力との誤差が大きいときはゼロとなり、誤差が小さくなるに従って大きくなるように制御される。これにより、誤差が大きいときは速度に関係なく学習が進められ、誤差が少なくなるにつれ目標変化速度が学習に考慮されるようになる。したがって、この K_f を制御することにより学習が効率良く調整される。

3. 6 シミュレーション

提案したハイブリッド制御器を評価するため、図 3.1 に示したモデルを使いコンピュータシミュレーションを行った。位置/力制御の際のロボットマニピュレータのエンドエフェクタの角度は、環境に対し常に直角となるように加速度分解制御を用いて制御した。シミュレーションでのサンプリング時間はすべて 1 ms とし、各サンプリングごとに位置/力制御器が学習するものとする。シミュレーションでは、量子化誤差と演算時間遅れは無視できる程度、センサーおよびモータの動作遅れはないものと仮定し、ノイズの影響は無視した。また、シミュレーションでの環境は次式のようにモデル化した。

$$f = M_e \ddot{x}_e + B_e \dot{x}_e + K_e x_e \dots\dots\dots (3.17)$$

ここで、 x_e は環境表面の変位である。

IANNのオフライン学習には、制御ルール設計時に用いた標準環境モデルの他、式(3.17)に示した各係数値を標準環境モデルの1/10倍に変えた柔らかい環境モデル1種類、各係数値を5倍に変えた固い環境モデルと10倍に変えた固い環境モデルの2種類、合計4種類について、環境の変位、変位速度、変位加速度を各々20、100、25パターンで変化させ、その各パターンと式(3.17)により計算したそのパターンでの反力を入力パターンとして用いた。また、各環境に対しIANNで出力する調整係数 K_E と K_{M0} は、標準環境のときは各々1とし、各係数1/10倍の環境モデルのときは1.2と0.7、各係数5倍の環境モデルのときは0.9と1.05、各係数10倍モデルのときは0.7と1.1を各々の目標出力パターンとしてオフライン学習を行った。

3. 6. 1 FNN力制御器の有効性

まず最初に、FNN制御器をロボットマニピュレータの力制御に用いる場合、入力情報としてマニピュレータが環境を押す速度（誤差の変化量に対応）ではなく、マニピュレータの力制御方向運動量を用いることの優位性を確かめるため、IANNを除いたFNN力制御による力制御シミュレーションを行った[80]。ここでの目標力は20Nと10Nの間で2秒毎に変化するステップ信号とし、サンプリング時間は1msとした。

このシミュレーションで使用したリンクの長さは次の通りである。

リンク1 : 0.2207 m

リンク2 : 0.1603 m

リンク3 : 0.1508 m

まずマニピュレータの形態が、基準座標系でのエンドエフェクタ先端の位置が設計時の標準位置 ($x=0.4\text{m}$) のときでシミュレーションを行った。図3.8が入力情報として誤差値とマニピュレータの力制御方向の運動量を利用したときの結果であり、図3.9が入力情報を誤差値とマニピュレータが環境を押す速度を利用したときの結果である。どちらの結果も、マニピュレータの形態が設計時の標準位置であるため、同じような結果を示している。また、立ち上がり時には、設計時のファジィ制御ルールが完全では無いため2%程度の誤差があるが、時間と共にニューラルネットワークによる学習により誤差が徐々に減少するのがわかる。

次にマニピュレータの形態を基準座標系でのエンドエフェクタ先端が0.05m手前 ($x=0.35\text{m}$) の位置になるように変更し、シミュレーションを行った。入力情報として誤差値とマニピュレータの力制御方向の運動量を利用したときの結果を図3.10に示す。図3.8と図3.10の結果を比較すると、マニピュレータの形態が多少変化しても同じような結果が得られることが確認できる。図3.11に入力情報として誤差値とマニピュレータが環境を押す速度を利用したときの結果を示す。この結果は図3.9での結果とは異なり、最初の立ち上がり時に少し振動しているのがわかる。これはマニピュレータのエンドエフェクタ先端を設計時の標準位置から手前に0.05m移動させると力制御方向の慣性力が約3/4になるが、最初の立ち上がり時の押し速度が早いときに制御器が速

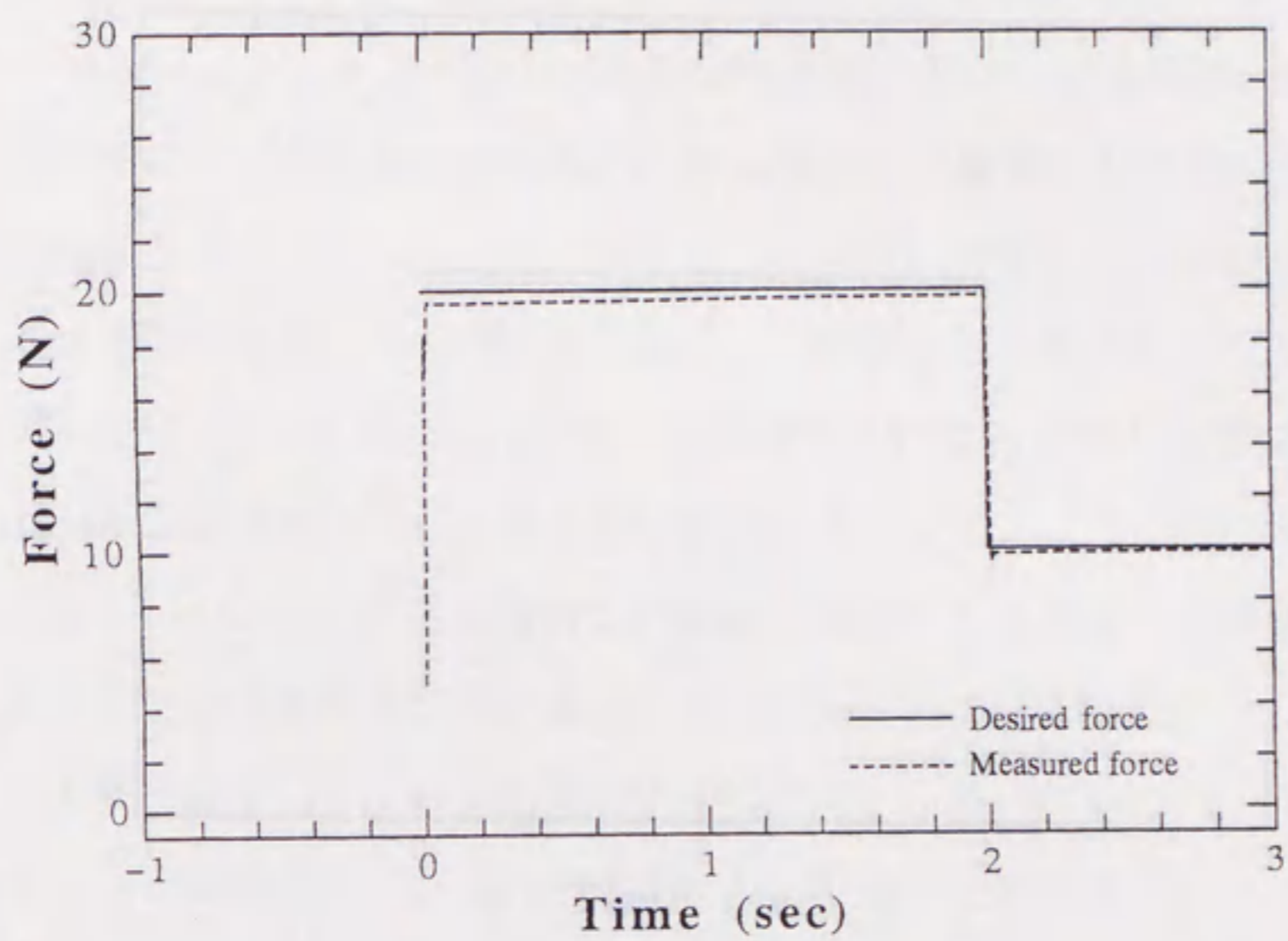


図 3.8 誤差量と運動量を利用したシミュレーション結果（設計時の標準位置）

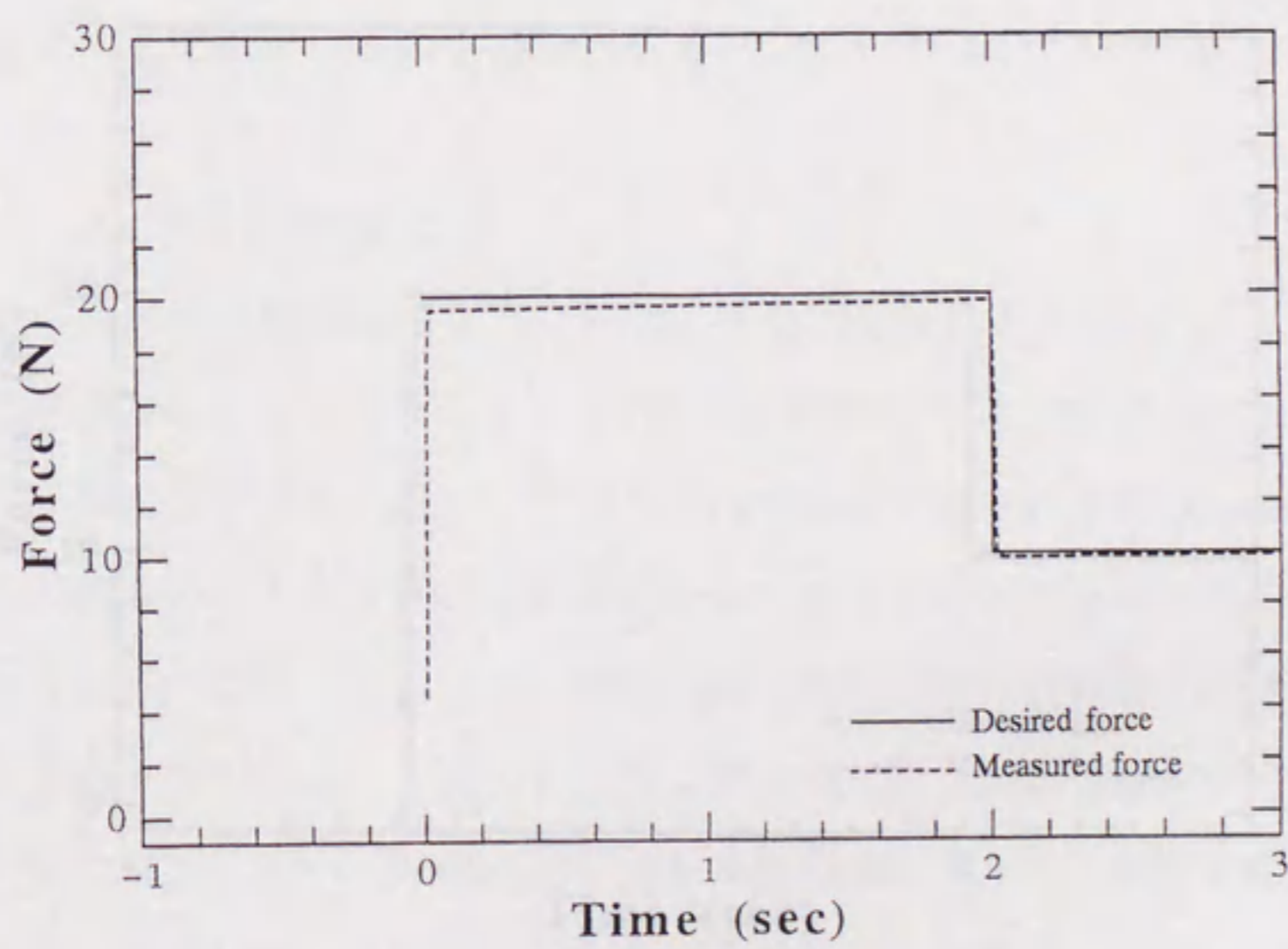


図 3.9 誤差量と速度を利用したシミュレーション結果（設計時の標準位置）

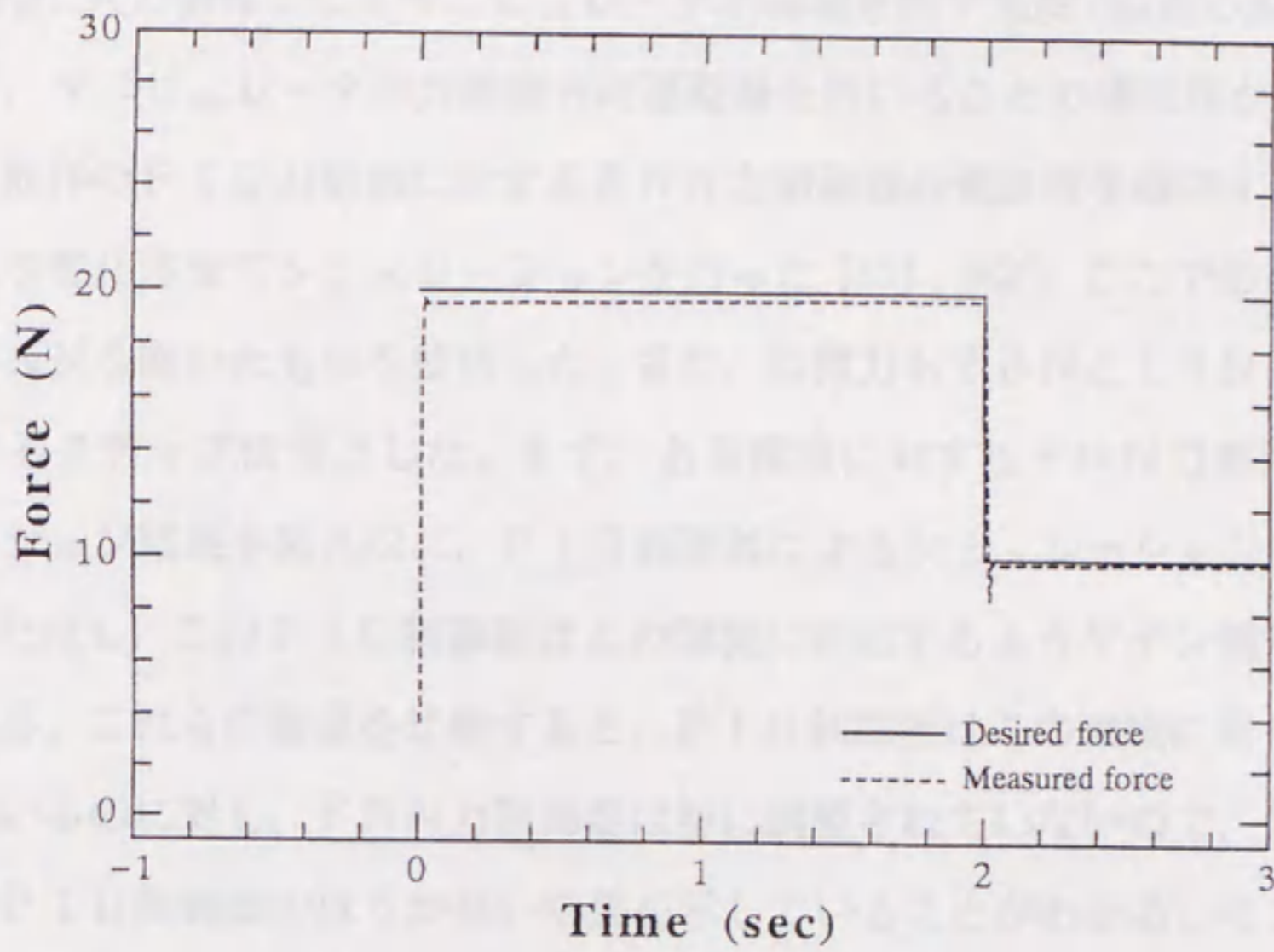


図 3.10 誤差量と運動量を利用したシミュレーション結果 (50mm 手前位置)

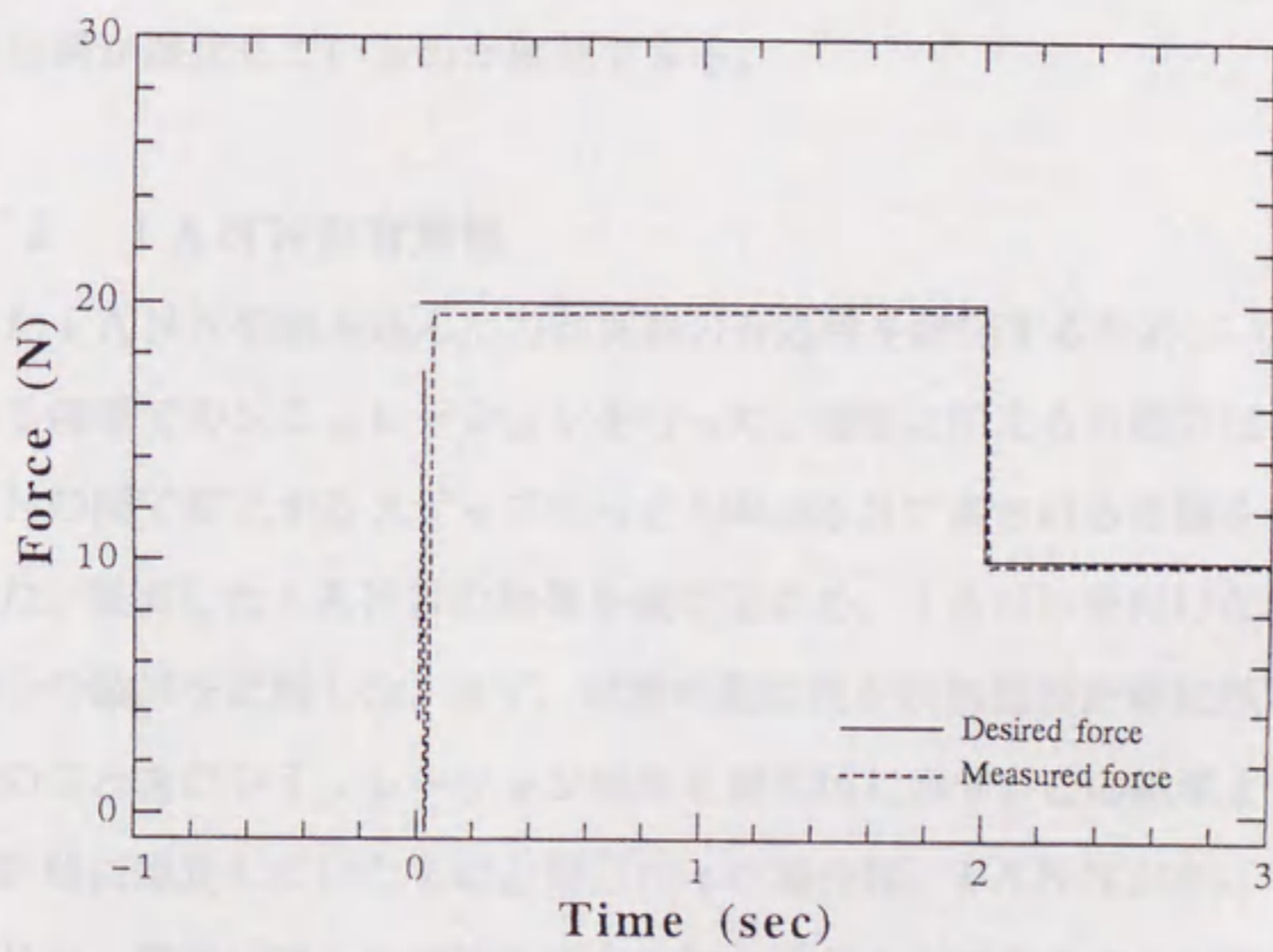


図 3.11 誤差量と速度を利用したシミュレーション結果 (50mm 手前位置)

度を弱めるために、標準位置のときと同じ力で押し返そうとして発生する現象であると思われる。これらの結果より、FNN制御器をロボットマニピュレータの力制御に用いる場合、入力情報としてマニピュレータが環境を押す速度（誤差の変化量に対応）ではなく、マニピュレータの力制御方向運動量を用いることの優位性が確認できる。

次に、既存のPID力制御に対するFNN力制御器の優位性を確認するため、環境の動特性を変化させてシミュレーションを行った [81], [82]。ここでのFNN力制御器もIANNを除いたものを使用した。また、目標力も20Nと10Nの間で2秒毎に変化するステップ信号とした。まず、ある環境に対するFNN力制御器によるシミュレーション結果を図3.12に、PID制御器によるシミュレーション結果を図3.13に示す。ただし、このPID制御器はこの環境に対応するようゲイン調整がなされたものである。これらの結果を比較すると、PID制御器はこの環境に対しゲイン調整がされているのに対し、FNN力制御器は特に調整されていないので、最初の立ち上がりではPID制御器のほうが良い結果を示していることがわかる。そこで、動特性の未知な環境に対する有効性を見るため、式(3.17)の各係数の値を2/5の値に変化させてPID制御器とFNN制御器の結果を比較した。このときのFNN力制御器の結果を図3.14に、PID制御器の結果を図3.15に示す。これらの結果を比較すると、FNN力制御器は環境の動特性が変化しても同様の結果が得られるのに対し、PID制御器では結果が悪化しているのが確認できる。

3.6.2 IANNの有効性

提案したIANNを組み込んだ力制御器の有効性を評価するため、いくつかの動特性の異なる環境でのシミュレーションを行った。環境に加える目標力は2秒毎に20Nと10Nの間で変化するステップ信号と $4 \sin(\pi t)$ Nで表される曲線を組み合わせたものとした。提案したIANNの効果を確認するため、IANNを付けた場合と付けない場合の結果を比較した。まず、環境の動特性が制御器設計時に想定したモデルと同じものときのシミュレーション結果を図3.16に示す。この結果より、環境の動特性が設計時に想定していたものと同じだった場合は、IANNが有っても無くても結果は同じで、環境に対し目標通りの力を加えることができることが確認できる。

次に式(3.17)に示した環境モデルの各係数を制御器設計時に用いた値の1/10倍に

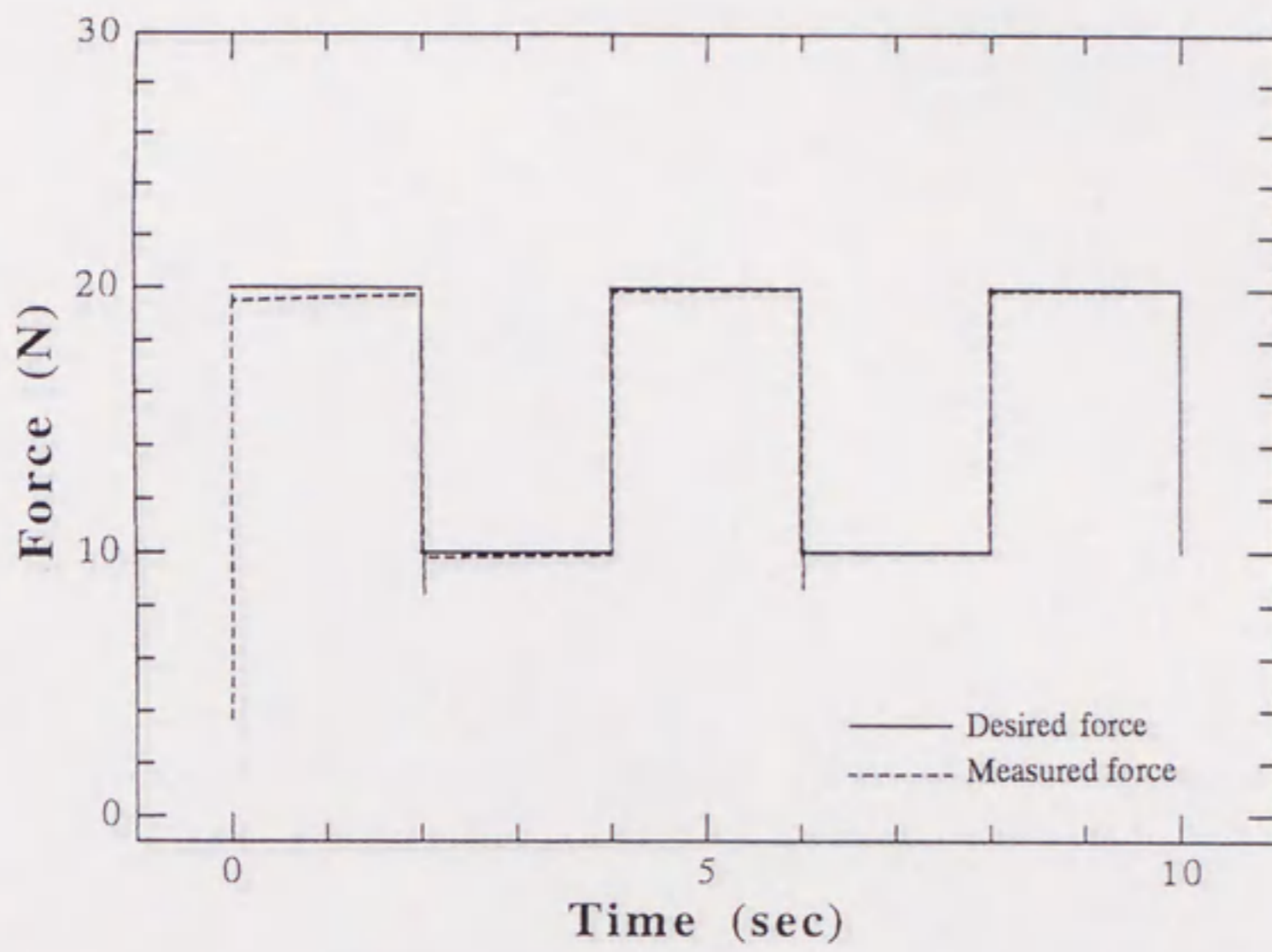


図 3.12 FNN力制御器によるシミュレーション結果

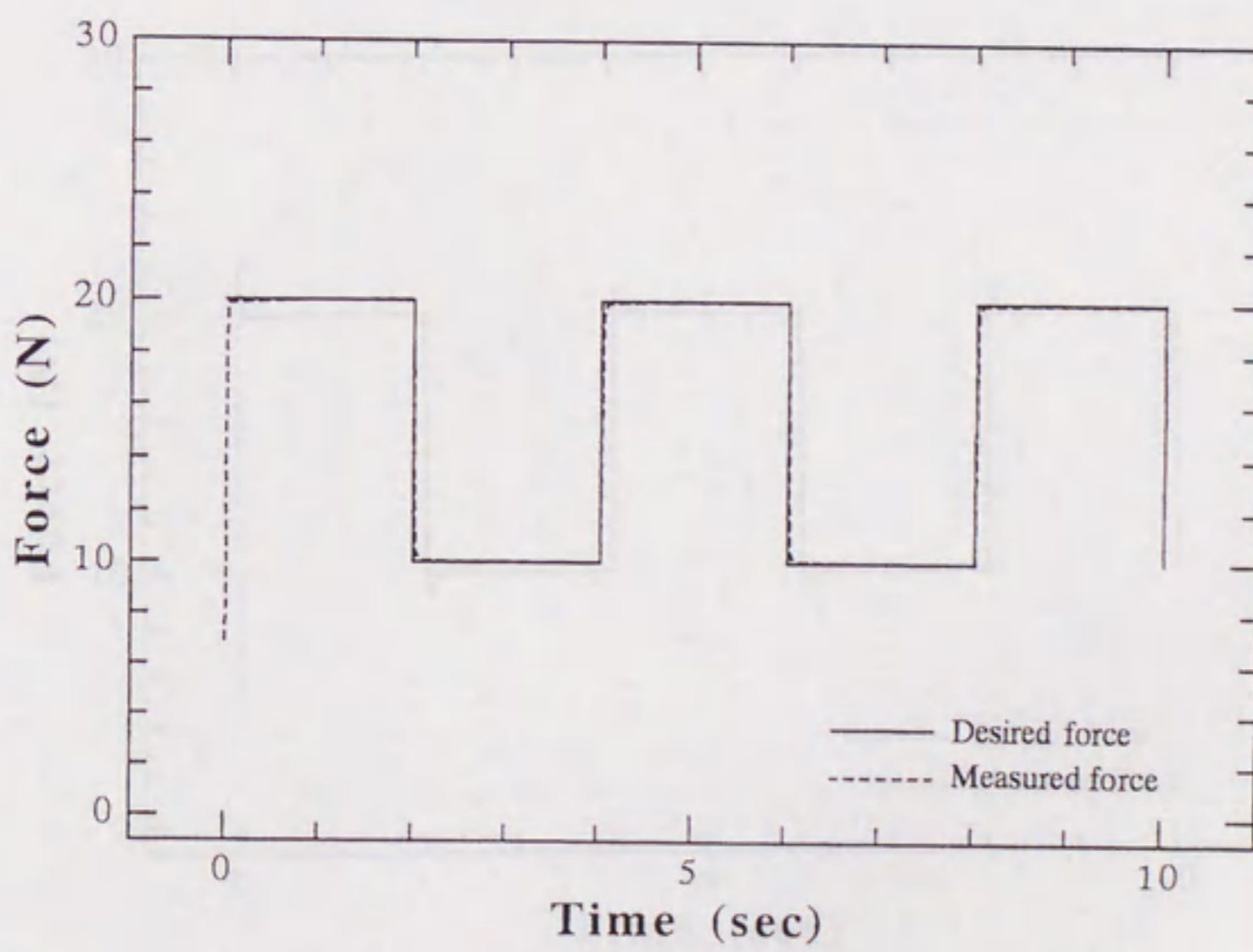


図 3.13 PID制御器によるシミュレーション結果 (ゲイン調整済み)

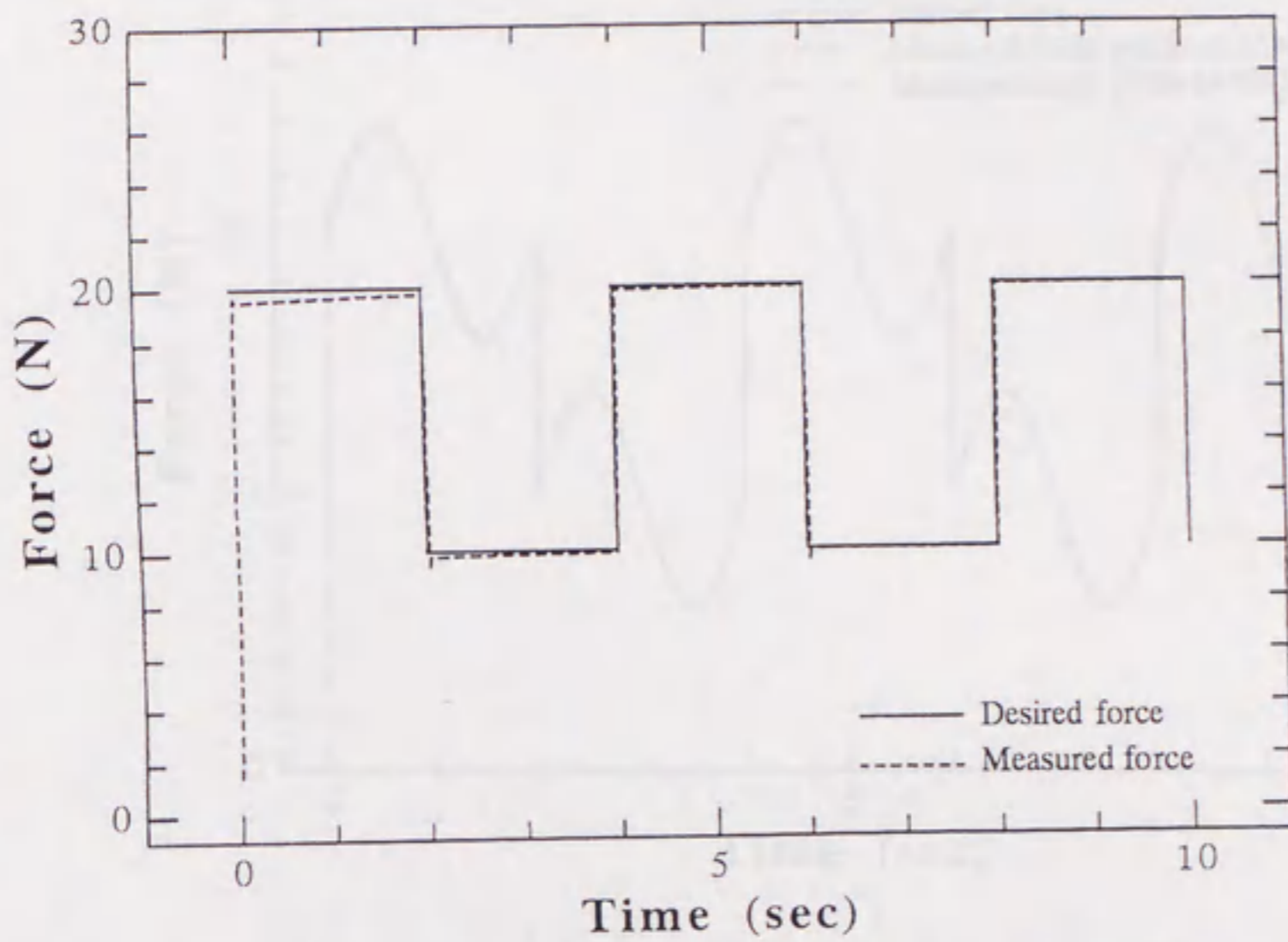


図 3.14 FNN力制御器によるシミュレーション結果（環境動特性係数 2/5）

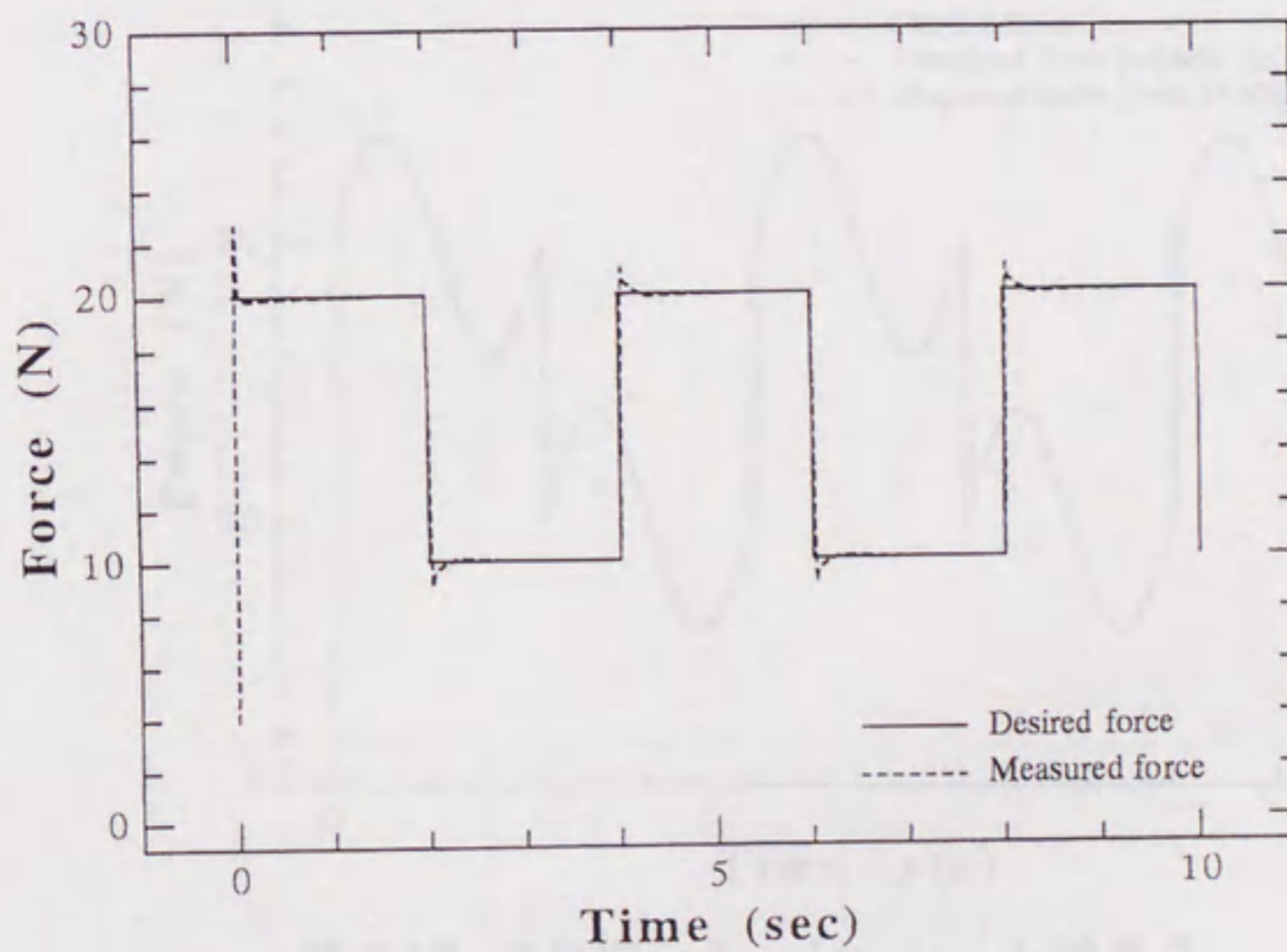


図 3.15 PID制御器によるシミュレーション結果（環境動特性係数 2/5）

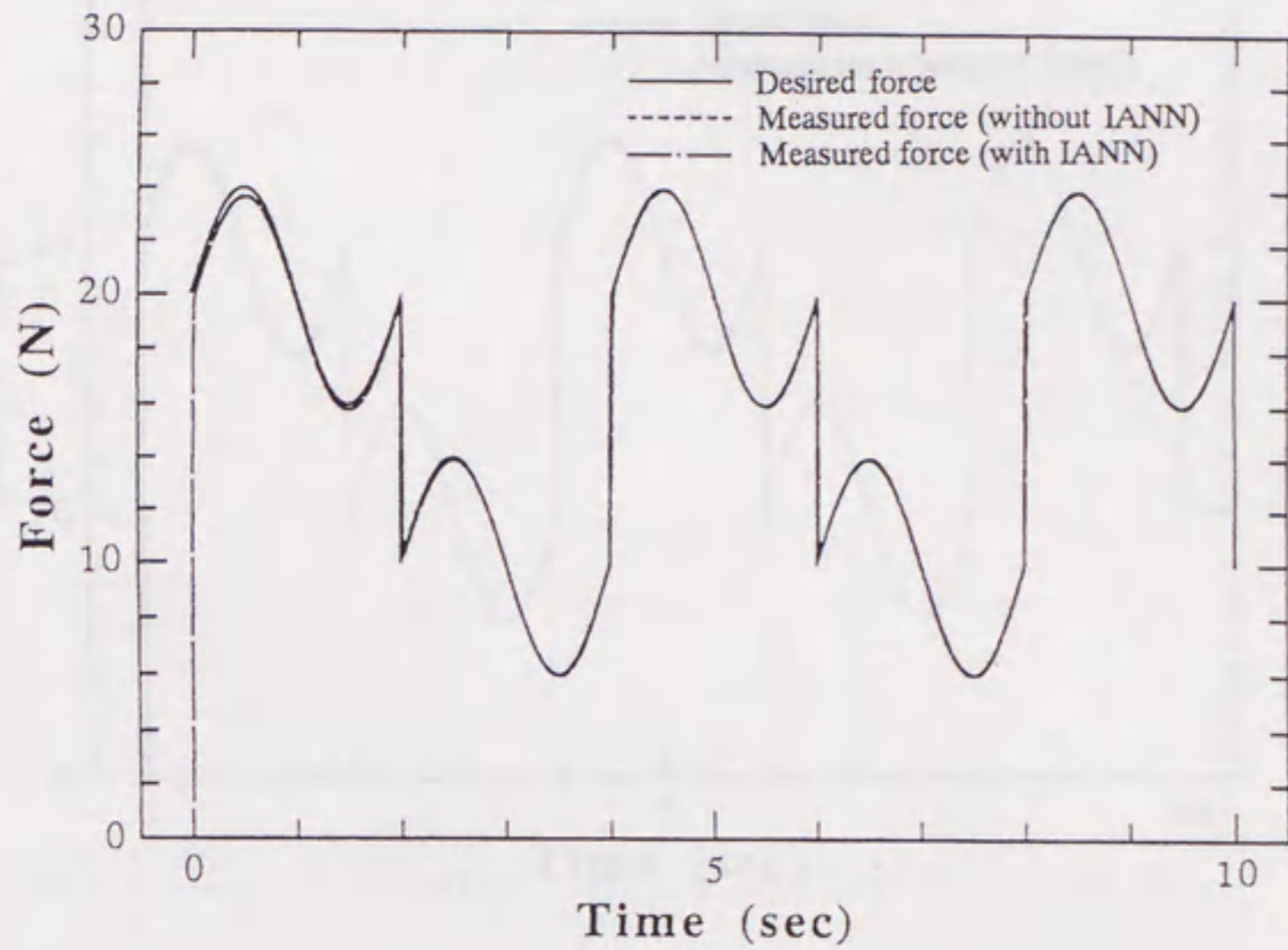


図 3.16 力制御シミュレーション結果 1
 (環境の動特性が制御器設計時に想定したものと同一場合)

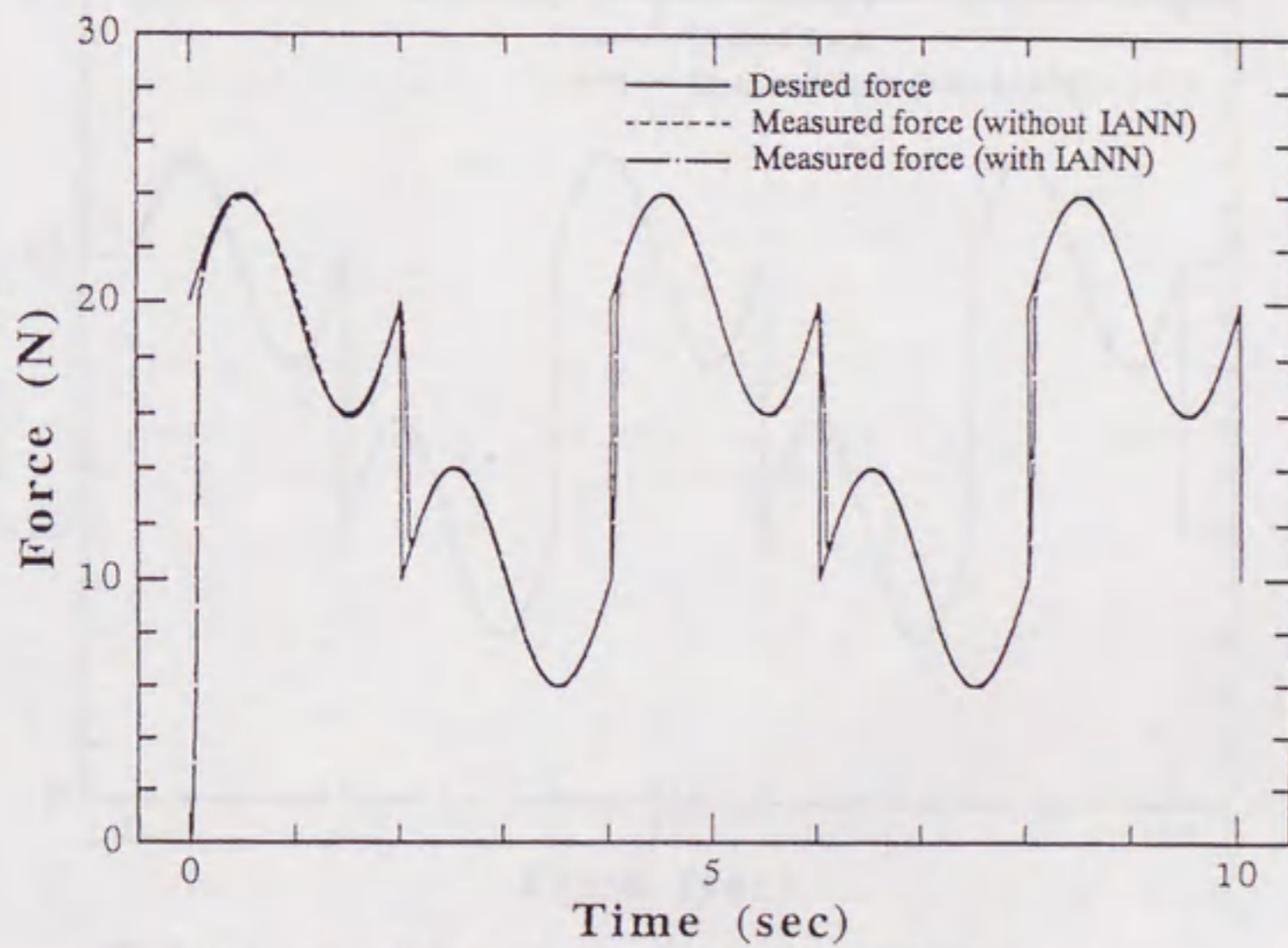


図 3.17 力制御シミュレーション結果 2
 (環境が制御器設計時に想定したものより柔らかい場合)

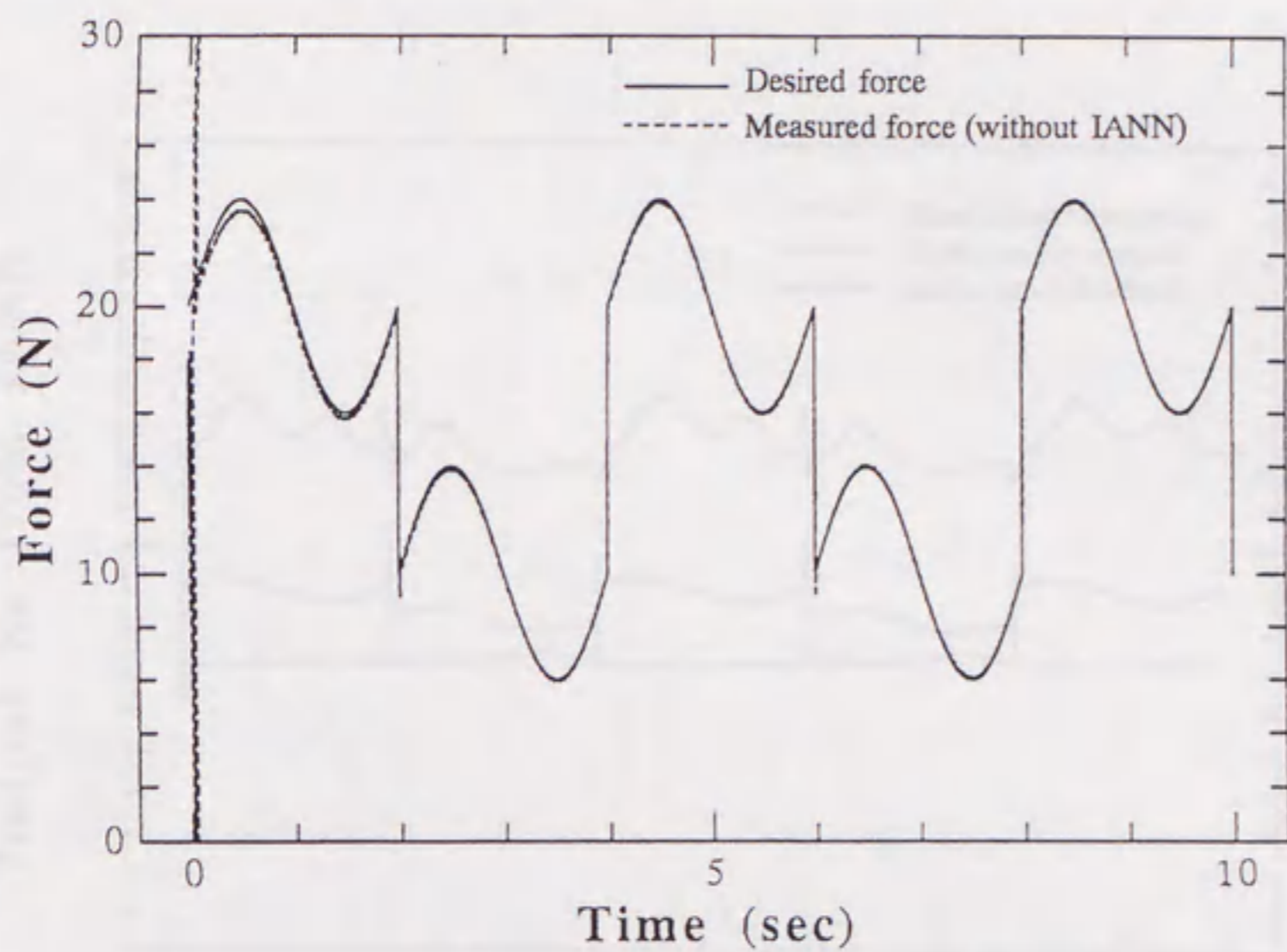


図 3.18 力制御シミュレーション結果 3
 (環境が制御器設計時に想定したものより硬い場合 - IANN無し)

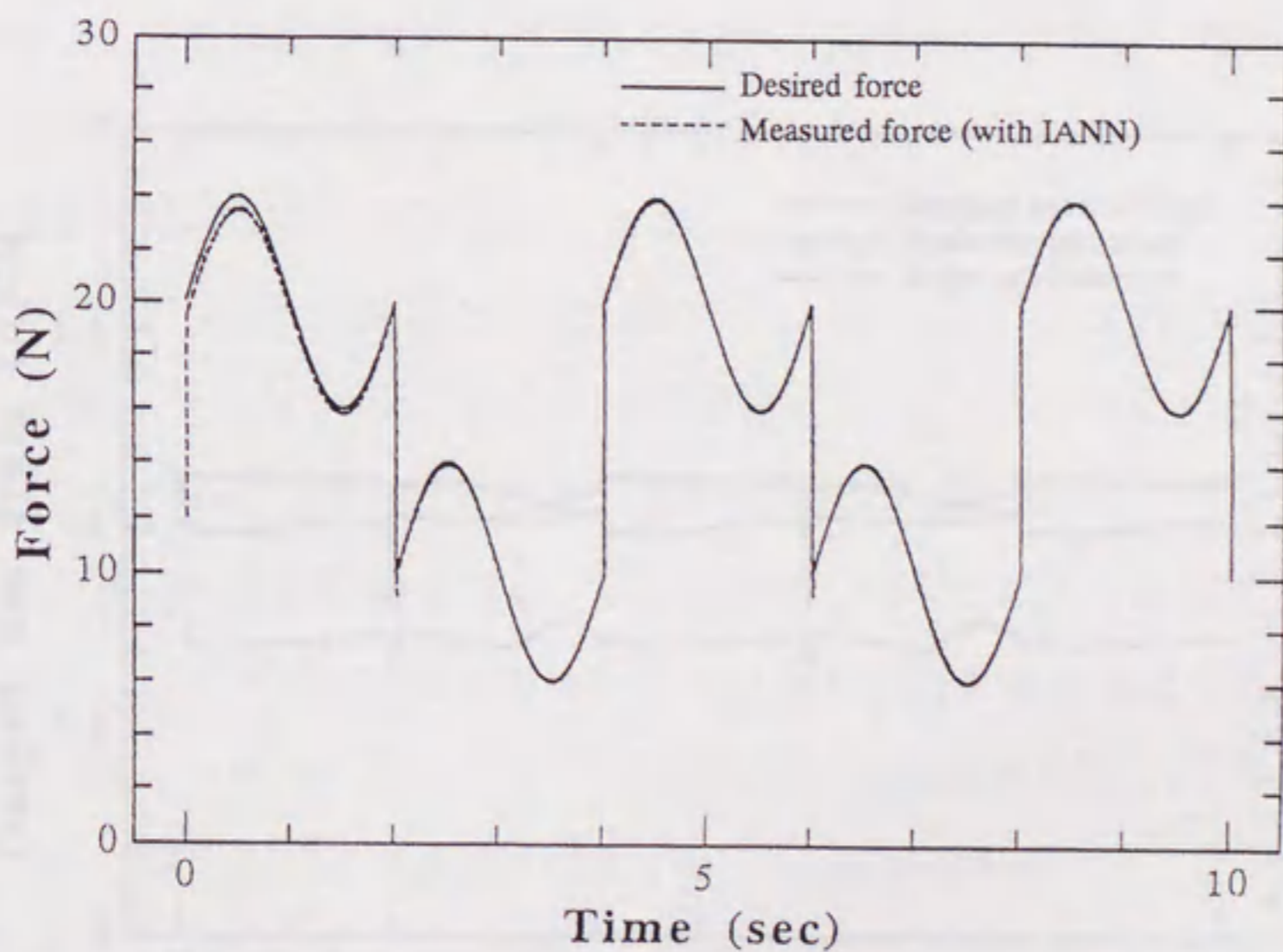


図 3.19 力制御シミュレーション結果 4
 (環境が制御器設計時に想定したものより硬い場合 - IANN有り)

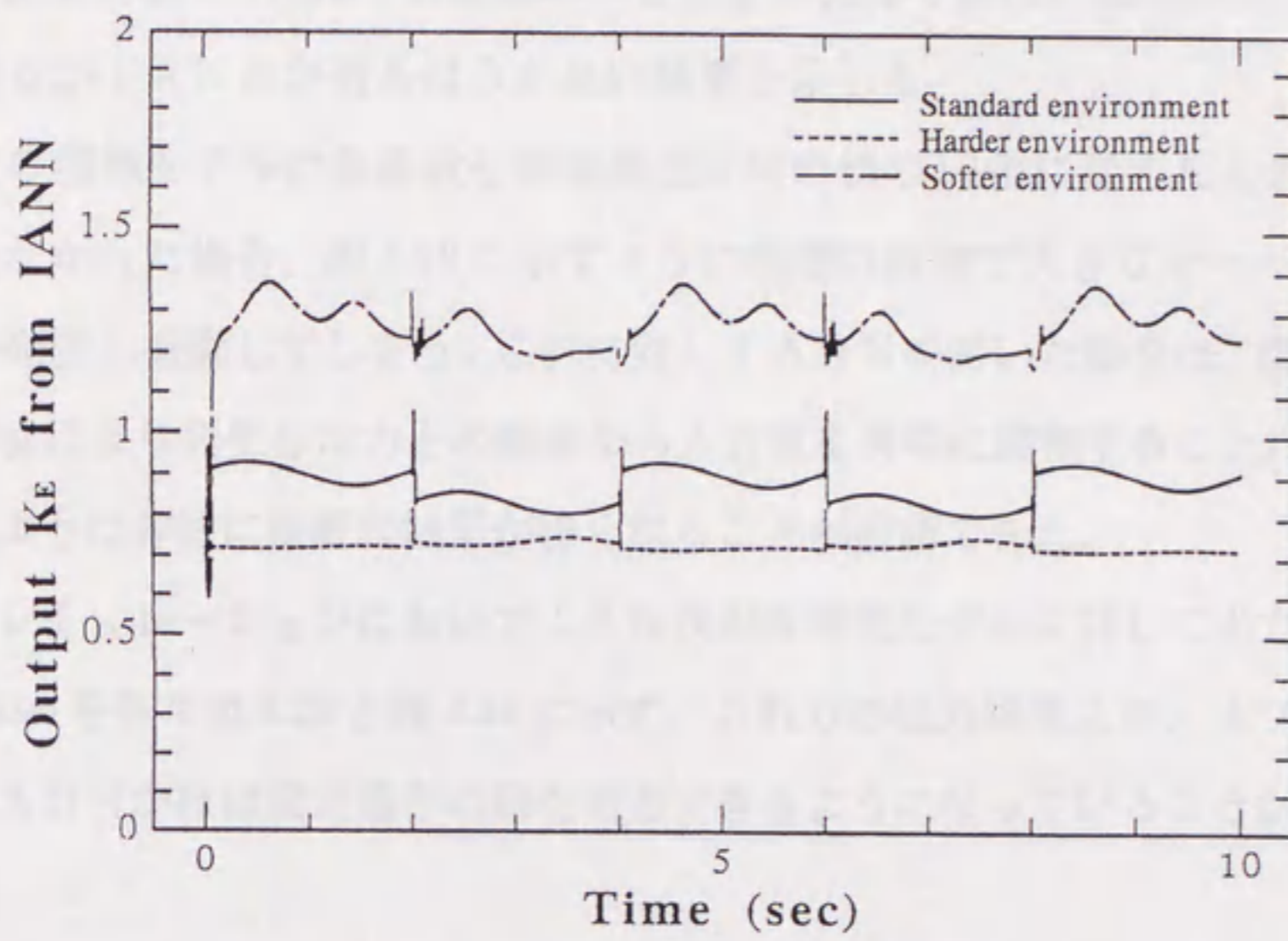


図 3.20 IANNからの出力： K_E

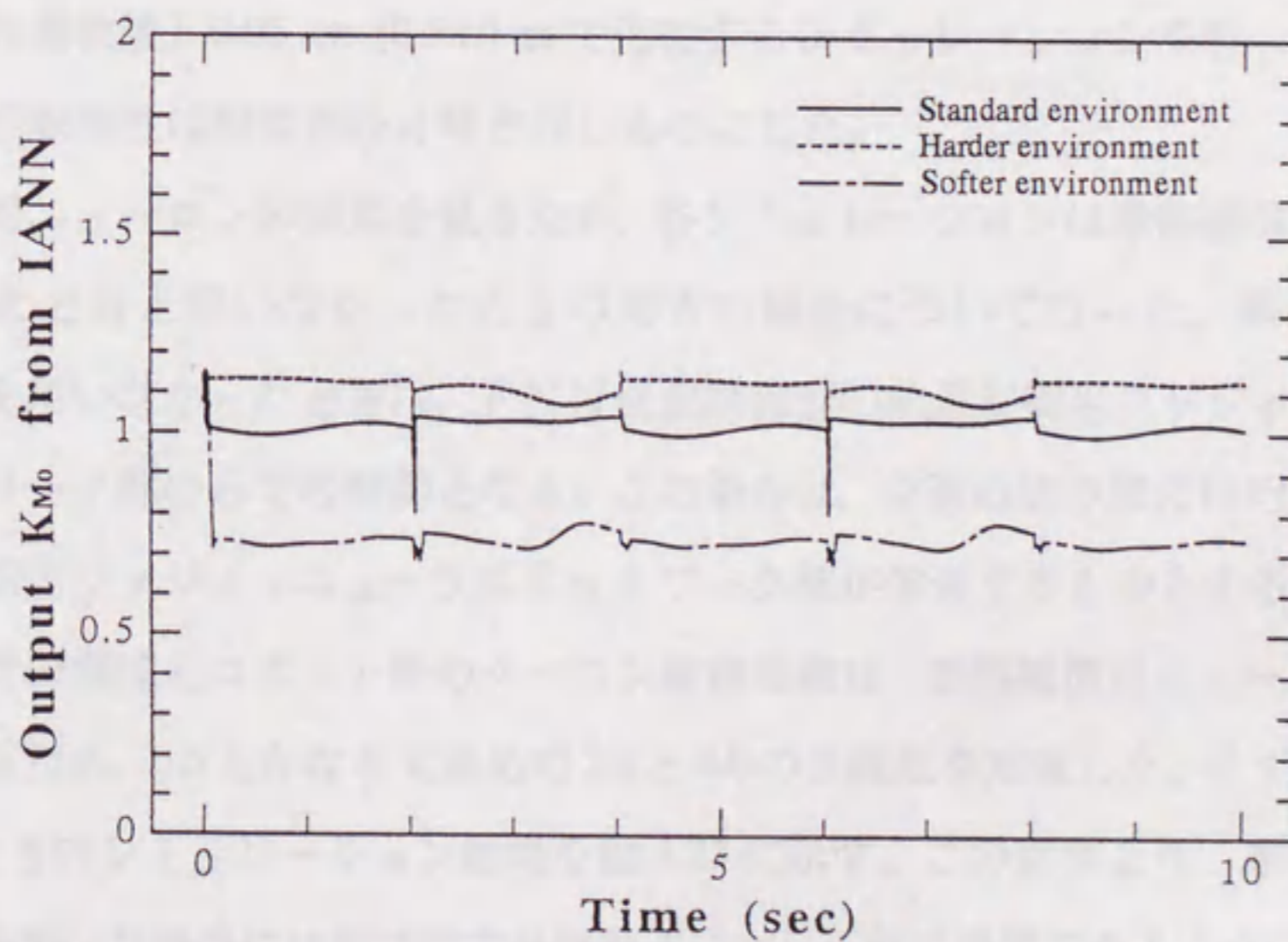


図 3.21 IANNからの出力： K_{M_0}

変えた柔らかい環境に対してのシミュレーションを行った。このときの結果を図3.17に示す。この結果より、柔らかい環境に対してもIANNが有っても無くても同じような良好な結果が得られるが、制御器の学習がまだ充分ではない最初の約2秒間は、若干ではあるがIANNが有るほうが良い結果を示した。

式(3.19)の環境モデルの各係数を制御器設計時の値の10倍に変えたときは、IANNを用いなかった場合、図3.18に示すように初期の段階で大きなオーバーシュアが発生し振動してしまう。これに対しIANNを用いた場合は、環境の変位量とその変位により発生した力との関係から入力値を瞬時に調整することにより、図3.19に示すように非常に良好な結果が得られることが確認できた。

これらのシミュレーションにおいてIANNが各環境モデルに対して出力した調整係数 K_E 、 K_{M0} を各々図3.20と図3.21に示す。これらの出力結果より、オフライン学習によりIANNがほぼ設定通りの値を出力できるようになっていることが確認できる。

3.6.3 FNN位置制御器の有効性

提案した位置制御器の有効性を評価するため、未知環境に対し図3.22に示すような2秒毎に20Nと10Nの間でステップ状に変化する力を加えながら、環境表面上を目標位置(目標軌道) $0.05 \sin(0.5\pi t)$ mで追従するシミュレーションを行った。このときの環境の動特性は制御器設計時と同じものにした。

摩擦補償用ニューロンの効果を見るため、各シミュレーションは摩擦補償用ニューロンを用いたときと用いなかったときの両方の場合について行った。摩擦補償用ニューロンを用いなかったときは、FNN位置制御器の軌道制御用ファジィ・ニューラルネットワーク部のみでの制御となる。この場合は、学習の切り替えは行わず最初から軌道制御用ファジィ・ニューラルネットワーク部が学習するものとする。シミュレーションでの環境とロボット間のクーロン摩擦係数は、摩擦補償用ニューロンの効果を評価するため、0.5とかなり大きめの2.0と4.0の3種類を用意した。まず、摩擦係数が0.5のときのシミュレーション結果を図3.23に示す。この結果より、摩擦補償用ニューロンを用いた場合には短時間で目標軌道に対し正確に追従できるようになるが、用いなかった場合は3周期目(1周期は4秒)でようやく目標軌道通りの運動になっ

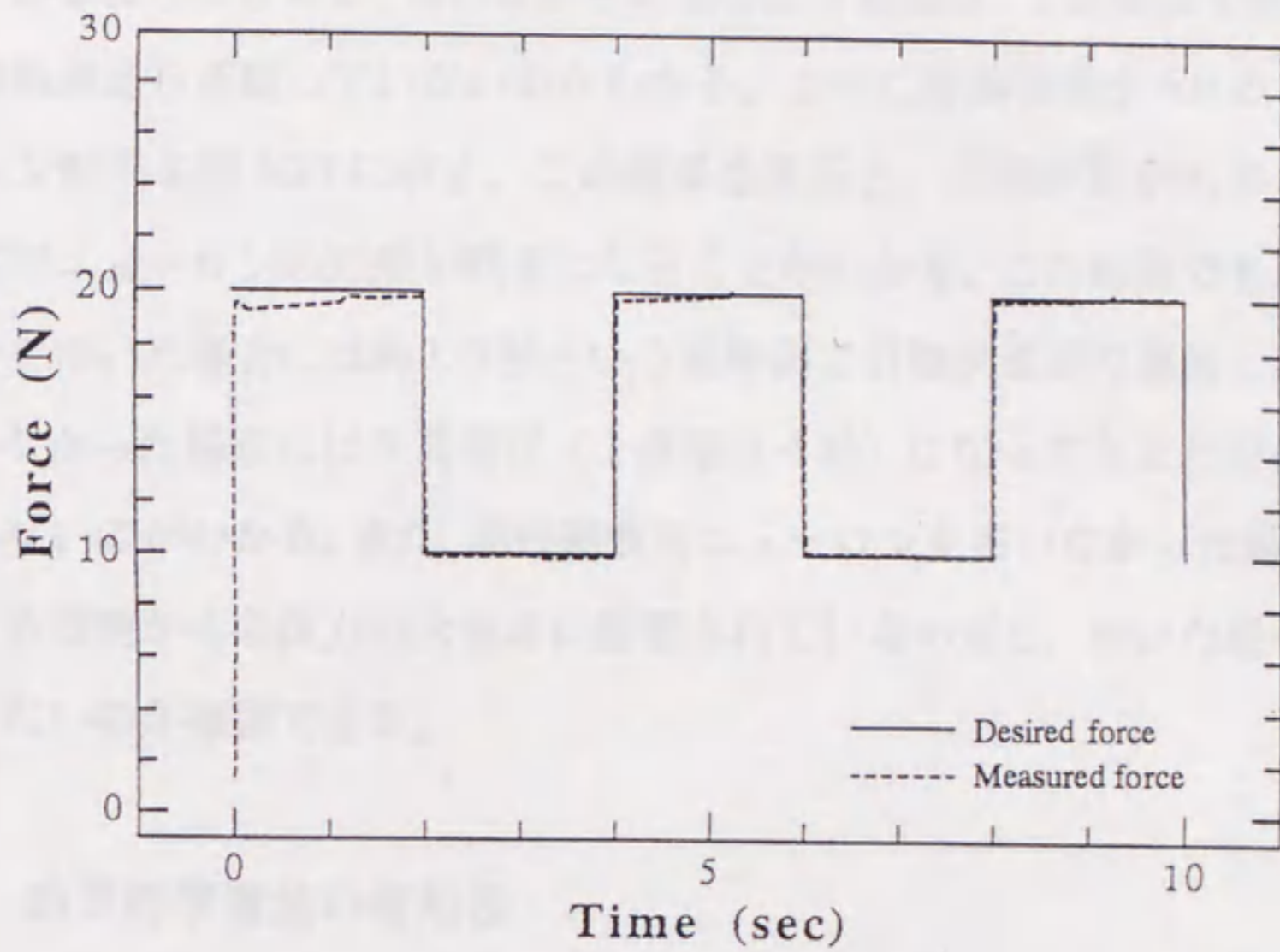


図 3.22 ハイブリッド制御時の環境からの反力

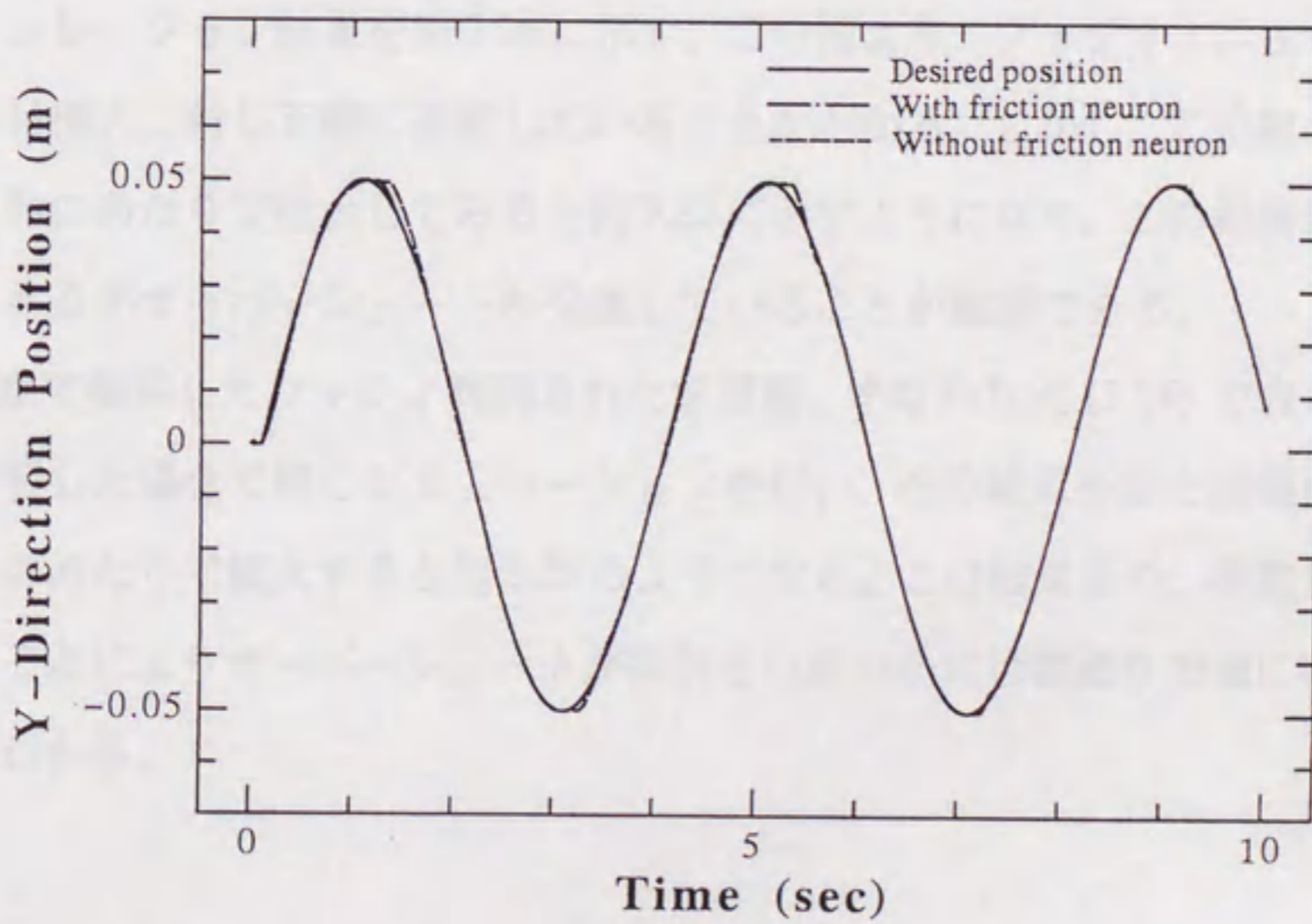


図 3.23 位置制御シミュレーション結果 1 (摩擦係数 0.5 の場合)

ていることがわかる。次に摩擦係数が2.0のときのシミュレーション結果を図3.24に示す。この結果より、摩擦補償用ニューロンを用いた場合は短時間で目標軌道に対し正確に追従できるようになるが、用いなかった場合は3周期目(1周期は4秒)になってもまだ目標軌道通り運動していないのがわかる。さらに摩擦係数が4.0のときのシミュレーション結果を図3.25に示す。この結果を見ると、摩擦係数が大きくなると益々摩擦補償用ニューロンの効果が顕著になることがわかる。この結果でも摩擦補償用ニューロンを用いた場合には約0.3秒という短時間で目標軌道通り運動し始めるのに対し、用いなかった場合には3周期目(1周期は4秒)になってもまだ目標軌道通り運動していないのがわかる。また、摩擦補償用ニューロンを用いなかった場合は、2秒毎に変化する環境からの反力の大きさに影響されているのに対し、用いた場合には影響を受けていないのが確認できる。

3. 6. 4 効果的学習法の有効性

最後に、3. 5章で提案した効果的学習法の有効性を確認するため、目標力を2秒毎に20Nと10Nの間でステップ状に変化する力に設定し、これまでより長時間シミュレーションを行った。また、環境は制御器設計時に想定したモデルと同じものとした。まず従来の学習則、すなわち式(3.9)で表される評価関数を用いて長時間制御した時のシミュレーション結果を図3.26に示す。この図より、ファジィ・ニューロ力制御器により目標力に対し正確に追従していることがわかる。しかし、この結果を目標力の前後2%のあたりで拡大してみると図3.27に示すようになり、この段階においてわずかではあるがオーバーシュートが発生していることが確認できる。

次に、本章で提案したファジィ制御された学習則、すなわち式(3.16)で表される評価関数で学習した場合で同じシミュレーションを行い、その結果を前と同様に目標力の前後2%のあたりで拡大すると図3.28のようになる。この結果より、提案した学習則を用いることによりオーバーシュートが抑制され緩やかに目標通りの値に収束していることがわかる。

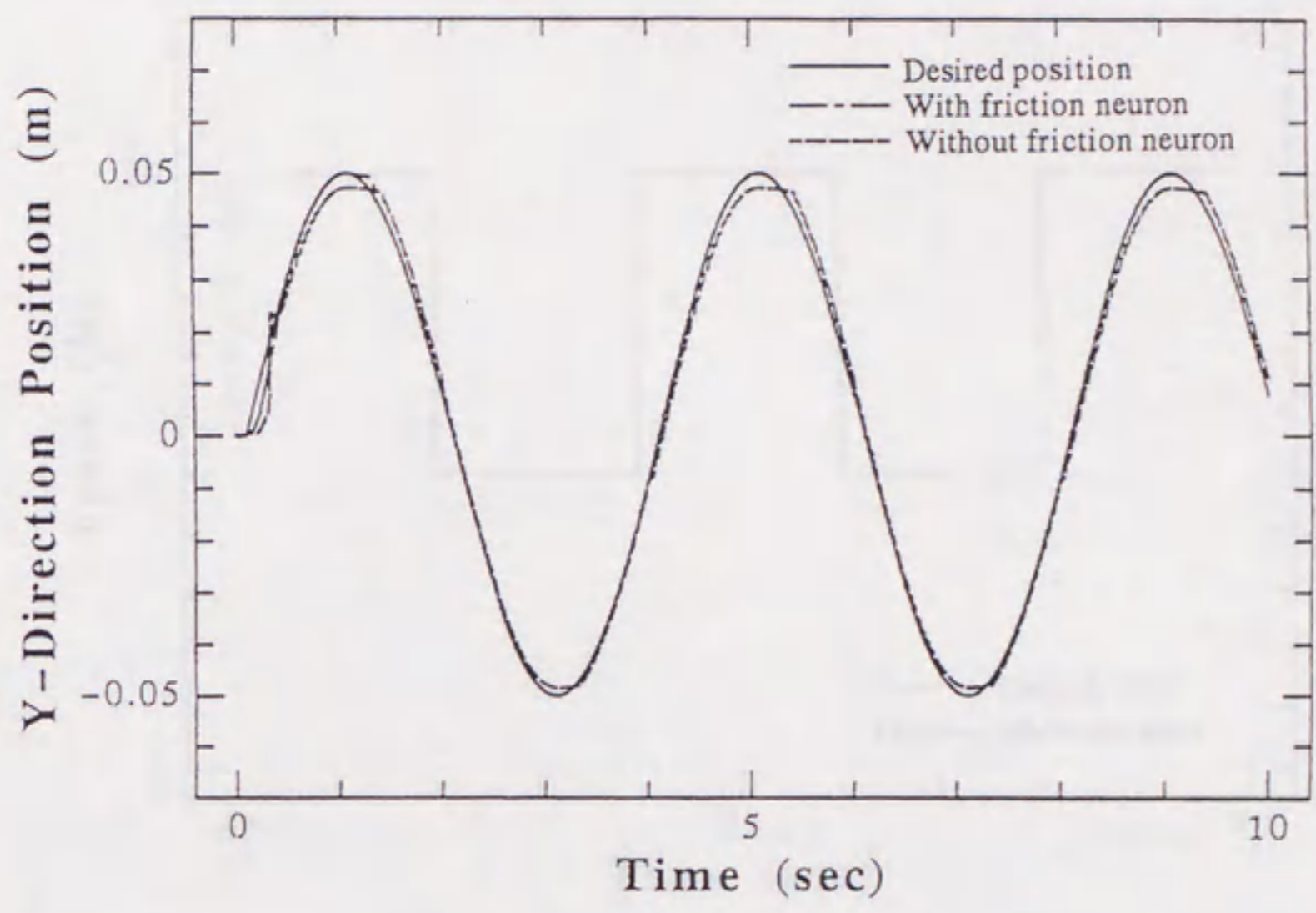


図 3.24 位置制御シミュレーション結果 2 (摩擦係数 2.0 の場合)

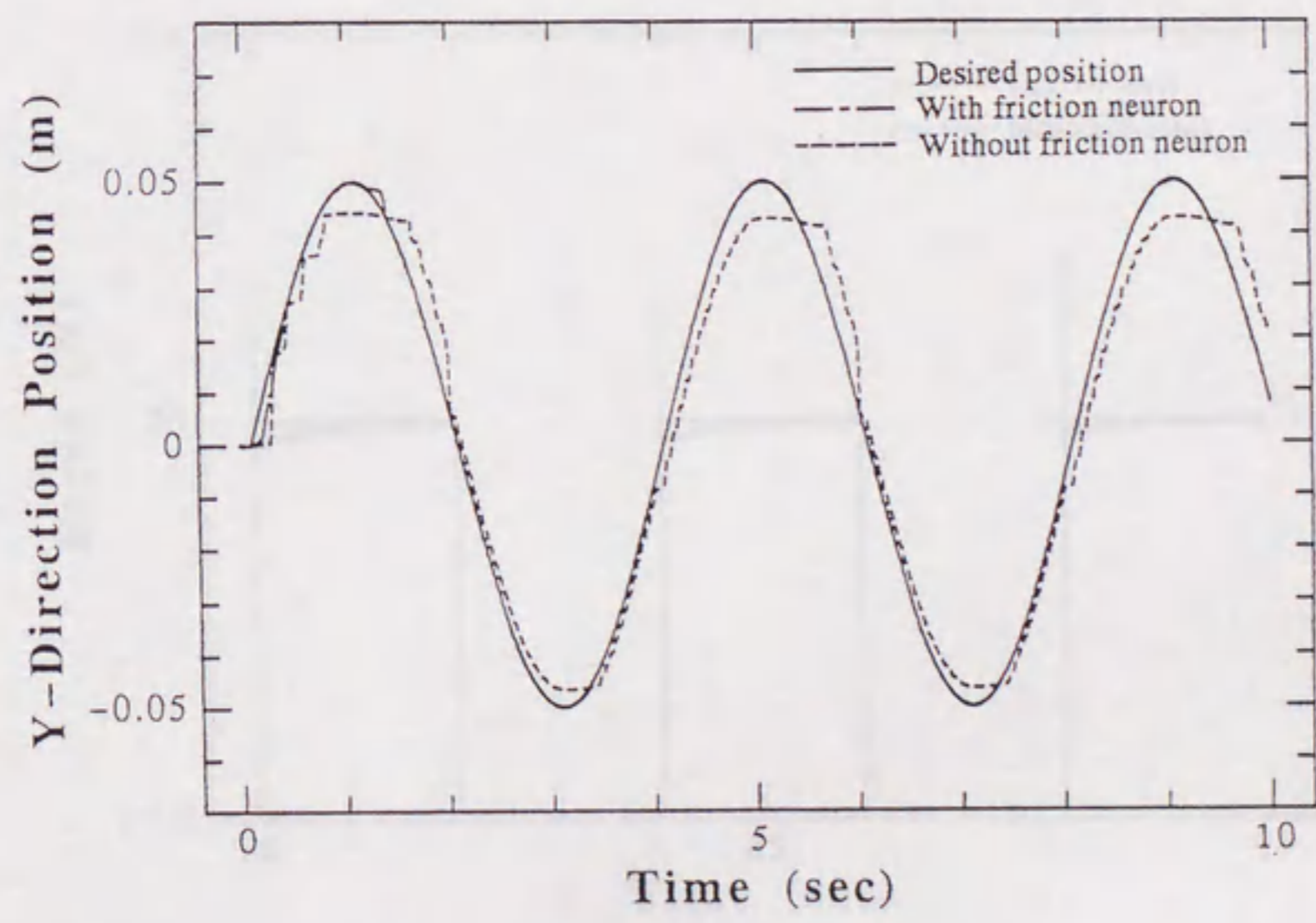


図 3.25 位置制御シミュレーション結果 3 (摩擦係数 4.0 の場合)

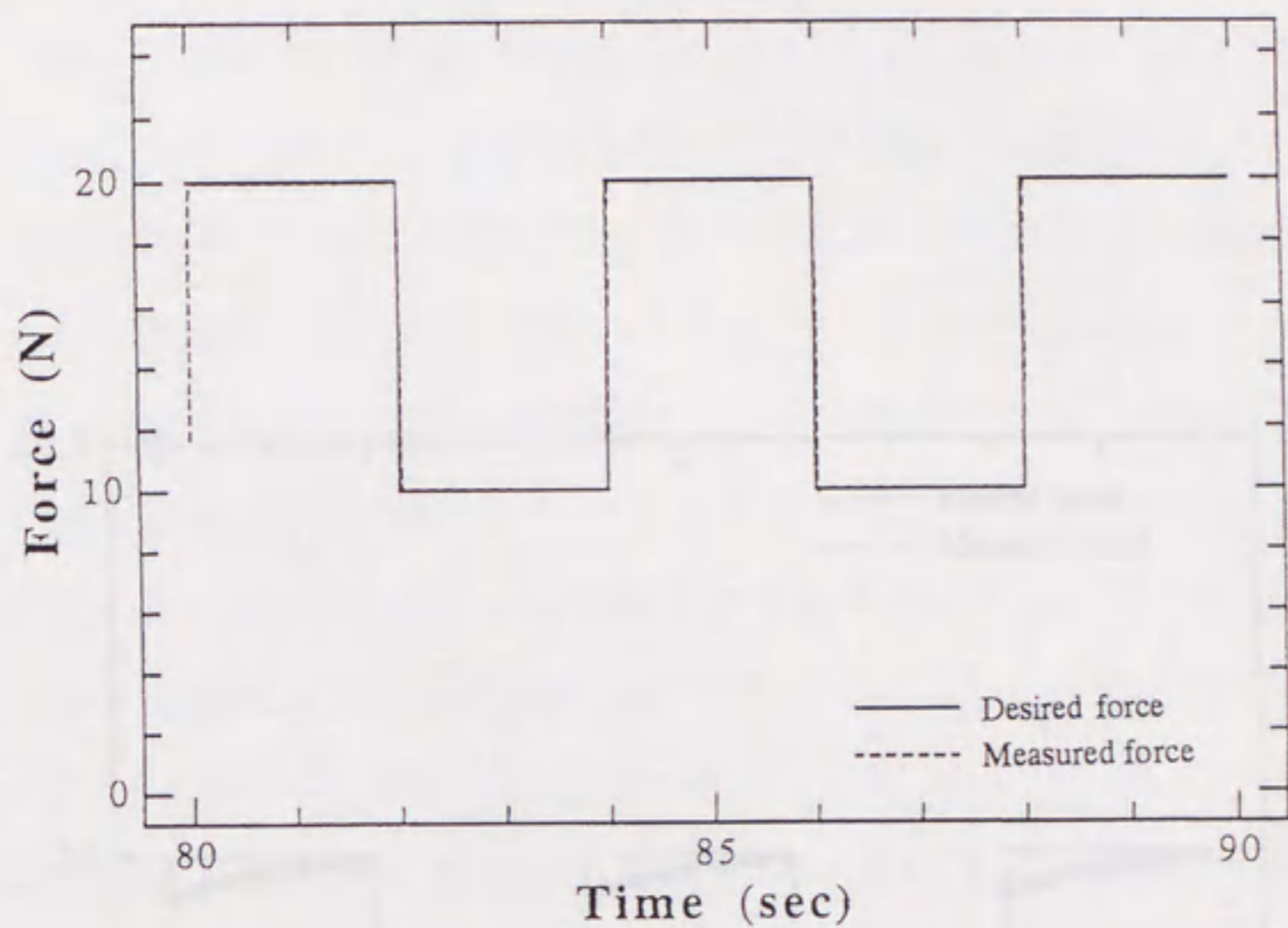


図 3.26 長時間学習（適応）後の力制御シミュレーション結果

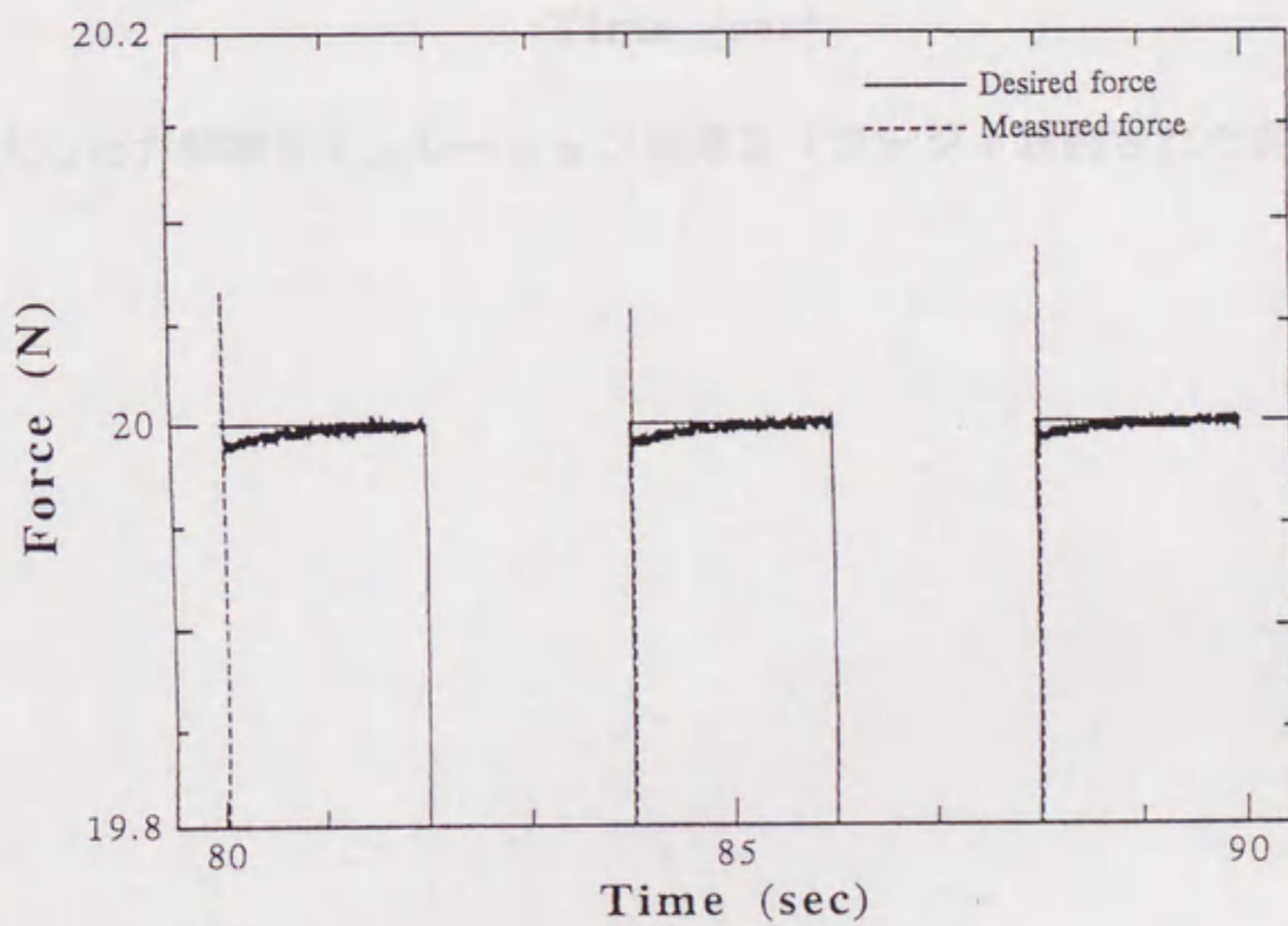


図 3.27 拡大した力制御シミュレーション結果 1（既存の評価関数）

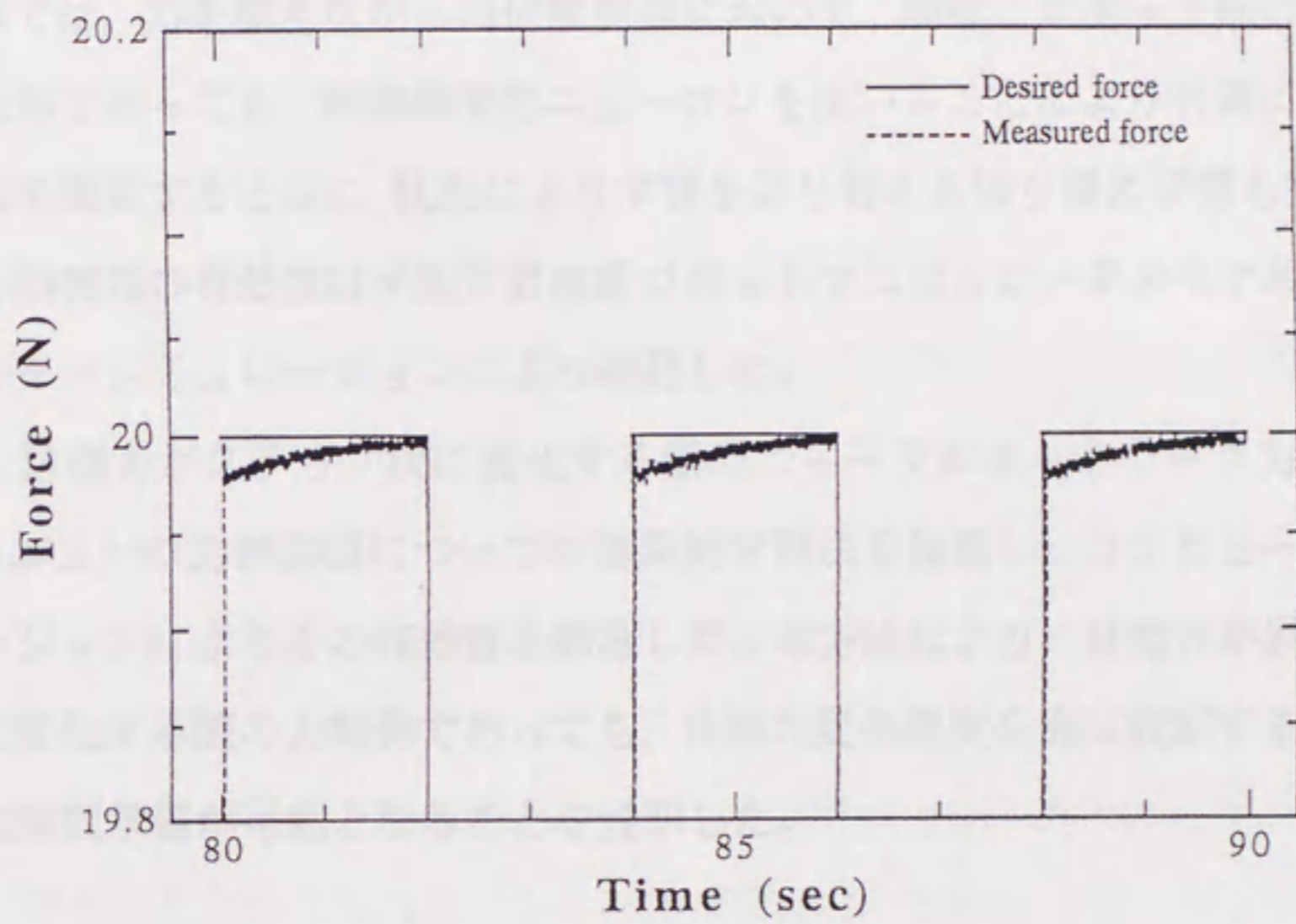


図 3.28 拡大した力制御シミュレーション結果2 (ファジィ制御された評価関数)

3.7 結 言

本章では、ニューラルネットワーク及びファジィ・ニューラルネットワークを利用して未知環境に対し位置と力を同時に制御するハイブリッド制御法を提案した。本制御器を用いることにより、動特性の未知な環境に対し目標通り力制御できることを示した。ただし、今回本研究でIANNの学習に用いた教師データは架空の値なので、今後教師データとして実際の材料の動特性データを使うことが望ましい。

また本章では、力を加えながらの位置制御において、環境とロボット間の摩擦力の補償量が未知であっても、摩擦補償用ニューロンを用いることにより有効に摩擦補償を行う方法を提案すると共に、状況により学習を切り替える切り替え学習も提案した。本位置/力制御器の有効性は平面3自由度ロボットマニピュレータのモデルを使用しコンピュータシミュレーションにより確認した。

さらに、目標力がステップ状に変化する際のニューラルネットワーク力制御器やファジィ・ニューロ力制御器についての効果的学習法を提案し、コンピュータによるシミュレーションによりその有効性を確認した。本方法により、目標力が周期的にステップ状に変化する際の力制御であっても、目標力変化速度を別に設定することなく効率的な長時間学習が可能となることを説明した。

第4章 ファジィ・ニューロを利用したロボットマニピュレータのインテリジェント位置／力制御

4.1 緒言

ロボットマニピュレータにより高度な作業を実現させるため、第2章と第3章でも述べたように、環境に対する位置と力を同時に制御する位置／力制御の研究 [21], [26] ~ [30], [42] ~ [48], [54] ~ [60], [71], [72] が盛んに行われている。位置および力を直接精度良く制御するにはハイブリッド位置／力制御 [42], [46] が有効である。しかし、これらの研究例のほとんどは環境の動特性や環境による拘束面の傾きあるいは曲率が既知であることを前提としており、それらが未知な環境に対するハイブリッド位置／力制御の研究例は少ない。吉川ら [58] は力センサーから得られた力情報と未知拘束面上をロボットのエンドエフェクタがなぞる際にロボットが移動した位置情報により摩擦力による影響を除去した力制御方向を算出し、さらに力制御方向ベクトルと仮想拘束面の法線ベクトルから次の位置制御方向を算出する方法を提案している。また他の研究例 [59], [60] でも力センサーからの力情報により未知拘束面の法線方向を特定しているものが多いが、これらの研究では初期の拘束面が既知であることが前提となっており、力センサーからの力情報を得るための最初の力制御方向を探すのに有効なアルゴリズムは含まれていない。環境からの反力情報を得るには、まず環境に対し力を加えなければならず、その時点でロボットのエンドエフェクタが接している未知拘束面の方向を知ることが必要となるが、これまでの研究例では初期の拘束面の方向を得る手段については明確に述べられていない。例えば、ロボットが拘束面に与えている力が小さい場合は、力センサーから得られた拘束面の法線方向はノイズの影響による誤差が大きいため、各サンプリング時間毎に力制御方向を変更する

際に誤った方向に変更する恐れが大きい。そこで本章では、未知拘束面に対しその法線方向の逆方向（力制御方向）をノイズを含む力センサーからの情報を基に力制御方向を修正しながら力制御を行うために有効な方法を提案すると共に、特性が未知な環境に対して有効なファジィ・ニューロ力制御 [71], [72] をより一層効率良くするためにファジィ環境評価器を用いる方法を提案する。

本章では、まずノイズを含む力センサーからの情報により未知拘束面の法線ベクトルを効率良く得るために、ファジィベクトルを提案し、ファジィ法線ベクトルの逆方向に力制御を行う方法について説明する [73], [86]。

特性が未知な環境に対する位置／力制御などでは、ファジィ・ニューロ制御は非常に有効な制御法の一つとなっている。ファジィ・ニューロ制御器は、環境の特性がファジィ力制御ルール設計時に予測したものよりも柔らかいか少し硬めの場合は特に問題なく力制御できるので、予め硬い環境を想定しておけばほとんどの場合問題ない。しかし、もし環境が想定したものよりかなり硬かった場合は、やはり大きなオーバーシュートが発生する等の問題が生じる危険性があるので、ファジィ力制御ルールを変更する必要がある [71], [72]。我々人間が環境に対し力を加える際の制御ルールを考えてみると、環境の特性や目標力が変更された場合、制御ルールをそれぞれの場合に合うように調整している場合も多いと思われる。そこで第3章では、環境の特性が硬かった場合に別の制御ルールを用いるのではなく、ファジィ・ニューロ力制御器への入力値を入力値調整ニューラルネットワーク (IANN) により調整することでメンバーシップ関数の定義を瞬時に変更する方法を紹介した。IANNは使用される環境の種類が限定されているとき（一般的にはほとんどの場合がこれに該当する）は非常に有効な方法であるが、もしも環境が全く未知である場合、ニューラルネットワークの汎化能力の面から見ると問題が発生する恐れがある。本章では、全く未知な環境に対しても対応できるよう、ファジィ環境評価器を提案し、これを用いてファジィ・ニューロ力制御器への入力値を調整することでメンバーシップ関数の定義を瞬時に、しかも力制御方向の調整具合を考慮しながら変更する方法を紹介する [73], [86]。

4. 2 ハイブリッド位置/力制御

ハイブリッド制御 [42], [46] は作業空間を力制御方向と位置制御方向に分け、位置と力を同時に直接的に制御する方法である。この制御法を平面ロボットマニピュレータで基準座標系のX軸に直角に設置された環境に対し加える力とエンドエフェクタの位置を同時に制御するような場合に適用する方法は既に2. 2節と3. 2節で説明しているが、ここでは任意の形状の環境を扱うため、ロボットマニピュレータを制御するためのトルクコマンドは次式のように示す。

$$\begin{aligned} \tau = & M(q)J^{-1}\{E^{-1}[(I-S)u_p - \dot{E}J\dot{q}] - \dot{J}\dot{q}\} + h(q, \dot{q}) \\ & + F_c \operatorname{sgn}(\dot{q}) + J^T E^T f + J^T E^T S u_f \end{aligned} \quad \dots\dots\dots (4.1)$$

ただし u_p は位置制御のコマンドベクトル、 u_f は力制御のコマンドベクトル、 E は基準座標系から作業座標系への変換行列である。本研究では環境による拘束面が未知であると仮定しているため、作業座標系への変換行列 E をオンラインでサンプリング時間毎に変更しながら制御するものとする。

ハイブリッド制御系のブロック線図を図4.1に示す。色がついている箇所が本章に関連する部分である。この制御器により、環境に加える力とエンドエフェクタの位置が同時に制御されるが、本章では主に力制御について説明するものとする。

4. 3 未知拘束面に対する力制御

未知な環境による拘束面に対し力を加える場合、まず拘束面の法線ベクトルを探ることが必要となる。最初にある方向に仮の力制御方向を仮定し、その仮の力制御方向に力を加えながら力センサーからの情報を基に拘束面を探すものとする。ただし、仮の力制御方向は未知環境による拘束面と平行ではなく、ある程度の角度で交わっていることを前提とする。環境に対しある程度の力を加えると、力センサーで検出された未知環境の反力方向により拘束面の法線方向が算出できるので、各サンプリング時間毎に算出された拘束面の法線方向の逆方向に力制御方向を変更すれば未知拘束面に対し力制御が可能となる。ただし、ここで得られる未知環境からの反力には摩擦力による影響も含まれているため、算出された法線方向は図4.2に示すように摩擦コーン内の方向を示していることになる。吉川ら [58] の研究のように未知拘束面上をロボットマニピュレータのエンドエフェクタが運動している場合は、摩擦力の影響を除外し、未

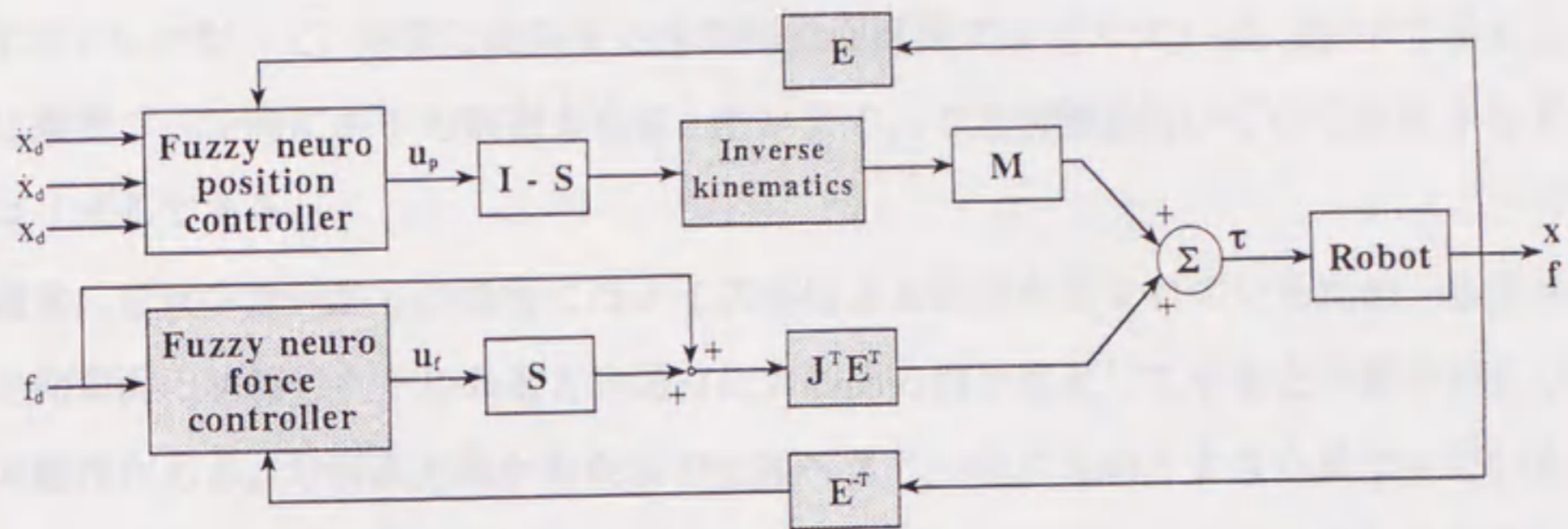


図 4.1 ファジィ・ニューロ位置/力ハイブリッド制御のブロック線図

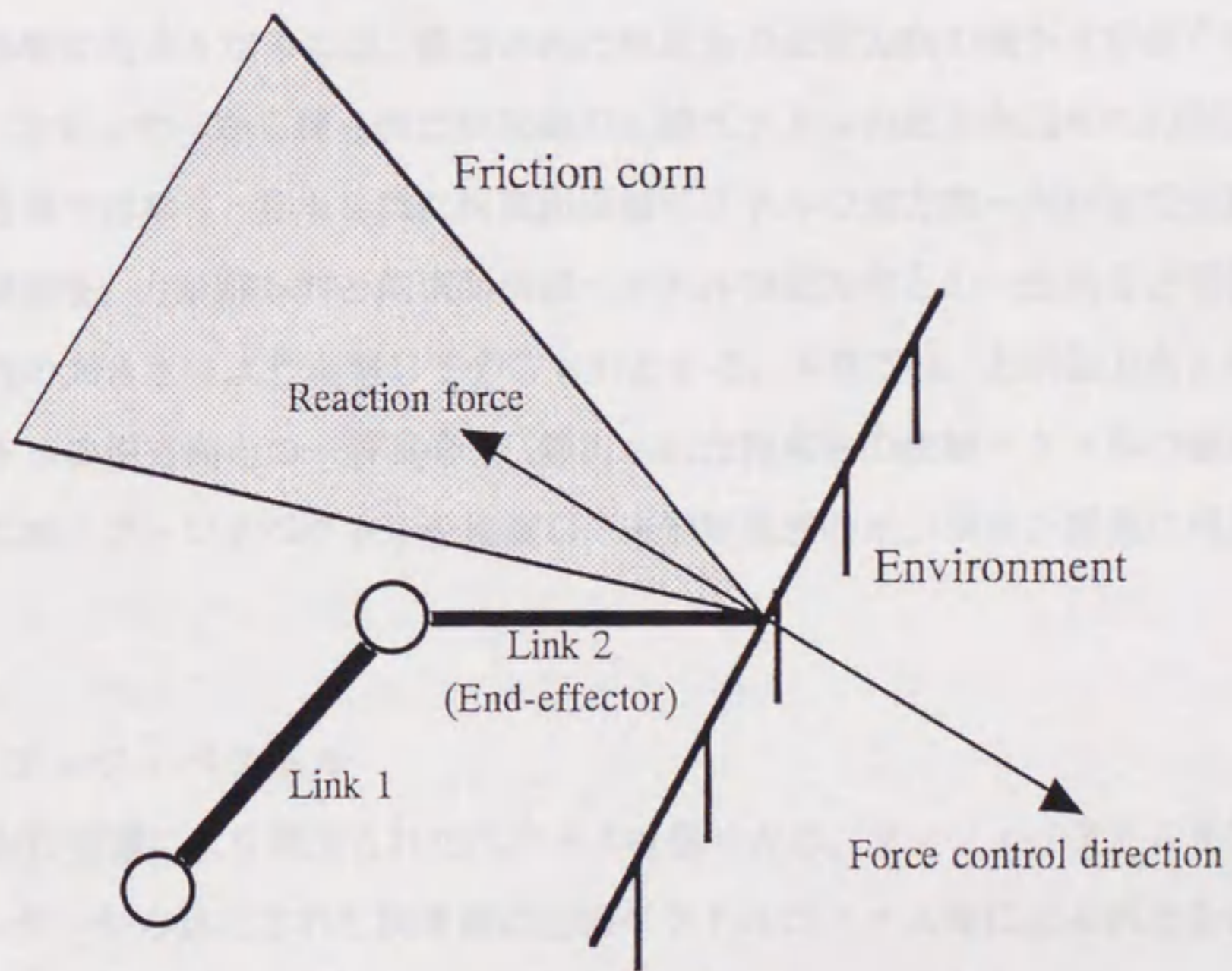


図 4.2 摩擦コーン

知環境面に垂直な方向を得ることが可能である。本研究では、ロボットマニピュレータにより未知環境の同じ位置に力を加えるものとし、拘束面上の移動は考えないものとする。したがって、本章で提案する未知拘束面探索アルゴリズムは、図4.2で示すような摩擦コーン内にある力制御方向を、オンラインで力制御を行いながら探索するアルゴリズムである。

通常、力センサーからの情報にはノイズ等による誤差が含まれているため、算出された拘束面の法線ベクトルの逆方向通りに力制御方向を変更してやると不都合が生じる可能性がある。力制御方向が拘束面の法線ベクトルの逆方向と大きく異なっている場合は、力センサーから得られた情報により算出された拘束面の法線ベクトルの逆方向通りに力制御方向を変更してやればよい。しかし、力制御方向が既に拘束面の法線ベクトルの逆方向とほぼ一致している場合は、ノイズの影響により各サンプリング時間毎に力制御方向が大幅に変動するのを避けるため変更の度合を小さく抑えてやる必要がある。また、環境に加えている力が小さい場合は、力センサーからの情報に含まれるノイズの影響が大きくなるため、算出された拘束面の法線方向の確かさが低くなる。そのため、力センサーから得られた拘束面の法線ベクトルの逆方向通りに力制御方向を変更するのではなく、算出された拘束面法線ベクトルの逆方向へ向かって変更する角度の変更量を、力制御方向と拘束面法線ベクトルの逆方向との一致具合と環境に加えている力の大きさにより調整してやるものとする。本章では、力制御方向と拘束面法線ベクトルの逆方向との一致具合や、算出された拘束面の法線ベクトルの確かさを表現するため、ファジィベクトルを提案し、未知拘束面のオンライン探索に利用する。

4. 3. 1 ファジィベクトル

ノイズを含んだ情報により算出されたベクトルを扱うため、ファジィベクトルを提案する。力センサーから算出された拘束面の法線ベクトルはノイズ等による誤差を含んでいるため、曖昧な大きさと方向を持つベクトルとして扱うと都合がよい。ファジィベクトルを用いることにより、大きさが小さいベクトルほど曖昧なベクトルとして扱うことが可能となり、力制御方向がファジィ法線ベクトルの逆方向とほぼ同じかどうかという表現が可能となる。平面2自由度ロボットマニピュレータの第1リンク

と第2リンク（エンドエフェクタ）の間に取り付けた力センサーにより、力制御方向の力成分とその方向に直交する方向の力成分を測定するものとする。これら2方向の力成分それぞれにノイズ等の影響による誤差が含まれているので、これらの情報により得られる拘束面の法線ベクトルには大きさと角度の両方に誤差が含まれることになる。このベクトルを示したのが図4.3である。図4.3では、力制御方向の力成分とその方向に直交する方向の力成分それぞれに誤差を含んでいるため、ベクトルの先端が矢印でなく楕円で示されている。この楕円で表わした区間は、ベクトルの先端がおよそその辺りにあることを表わしている。図4.3で示しているように、これらの誤差を含んだ力成分から合成されたベクトルは、ベクトルの先端が四角形の内側のどこかにあり、方向が黒色で表わしてある部分の内側のどこかにあるファジィベクトルで表わされる。図4.3の (a) と (b) を比較すると、誤差の程度が同じであれば力成分が大きいほどファジィベクトルの方向のあいまいさが少なくなることがわかる。

ファジィベクトルの大きさはファジィ数 [61] として扱うものとする。ファジィ数で表現された大きさを持つベクトルをファジィ単位ベクトルに変換すると、大きさが小さいベクトルほど曖昧なファジィ単位ベクトルに変換される。例えば、図4.4の (a) に示すようにおよそ6というファジィ数と (b) に示すおよそ3というファジィ数をそれぞれの大きさのクリスプ数で除するとおよそ3の数から算出されたおよそ1のメンバーシップ関数のほうが横に広がった形、すなわち、より曖昧な定義となる。

力制御方向がファジィ法線ベクトルの逆方向とほぼ同じ場合には、ノイズ等による悪影響を少なくするため、その時点での力制御方向から算出された拘束面法線ベクトルの逆方向へ向かって変更する角度の変更量を小さくするようにしてやる必要がある。そのため、方向の一致具合が大きいほど方向変更量を小さくし、方向が一致していないときには力センサーから得られた情報により算出された拘束面の法線ベクトルの逆方向通りに力制御方向を変更するよう、ファジィ制御の概念を利用して調整してやるものとする。

4. 3. 2 ファジィベクトルを利用した力制御方向変更アルゴリズム

まず最初にファジィ法線ベクトル方向の角度のファジィ度を示すメンバーシップ関数を、図4.4で示したように力センサーから算出された法線ベクトルの大きさにより定

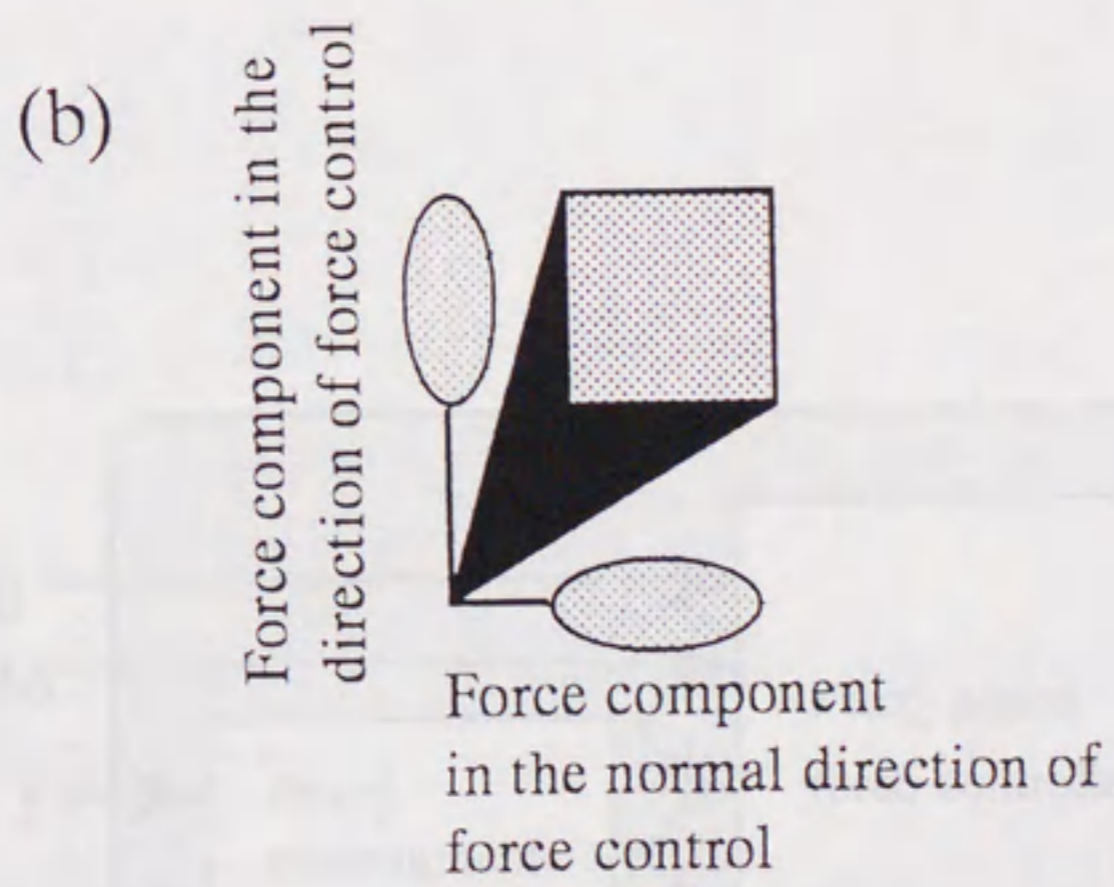
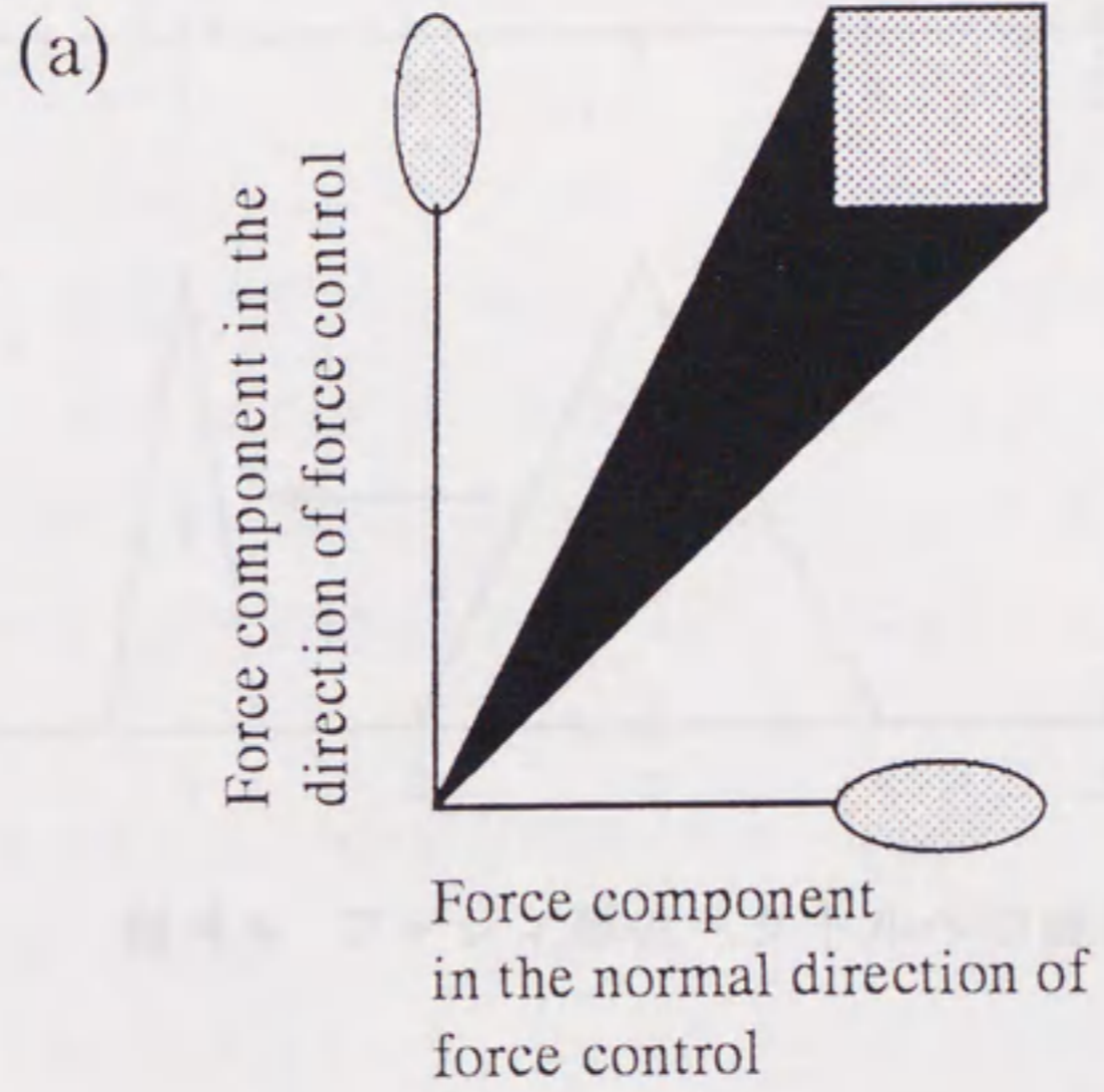


図 4.3 ファジィベクトル

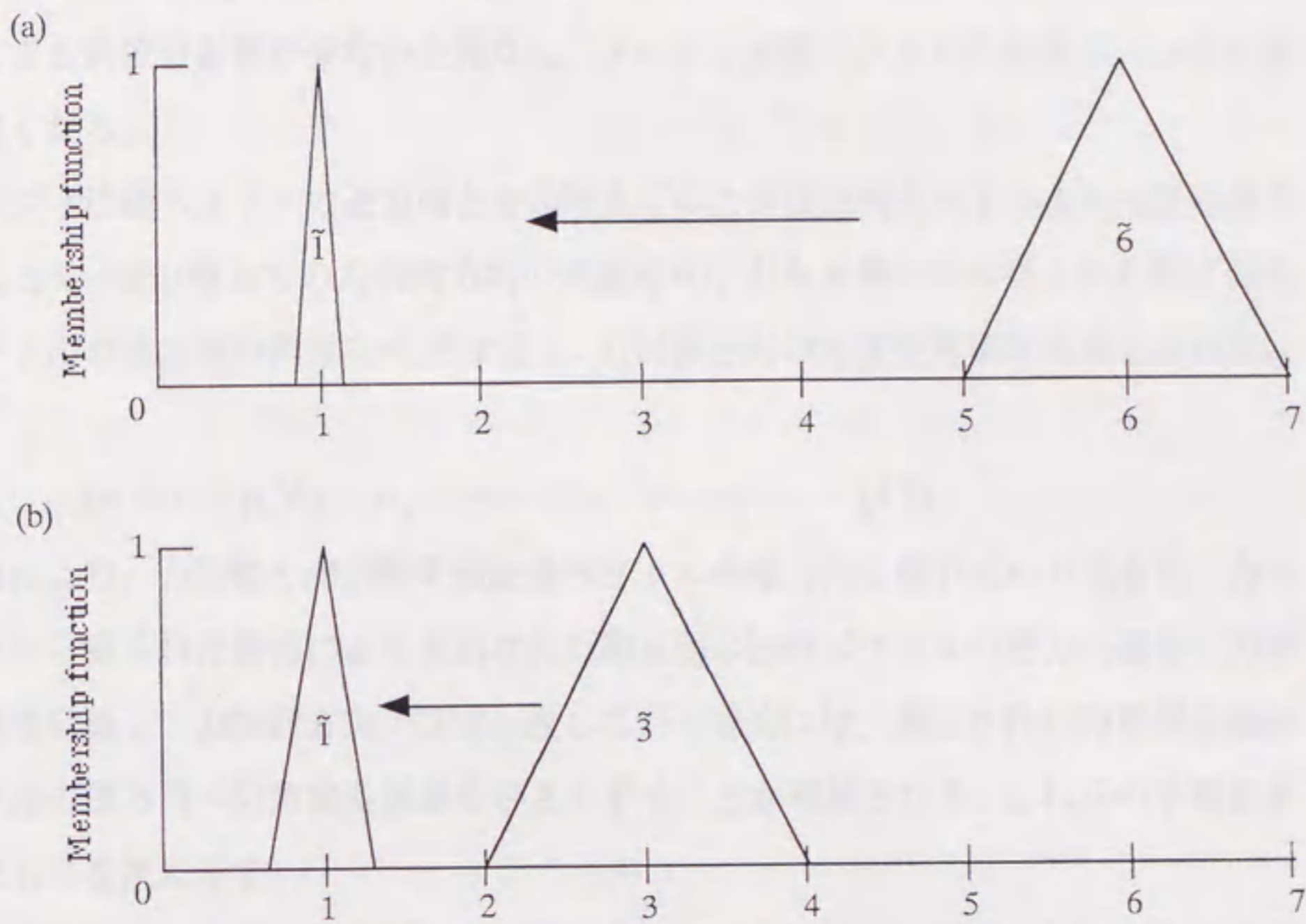


図 4.4 ファジィ単位ベクトルへの変換

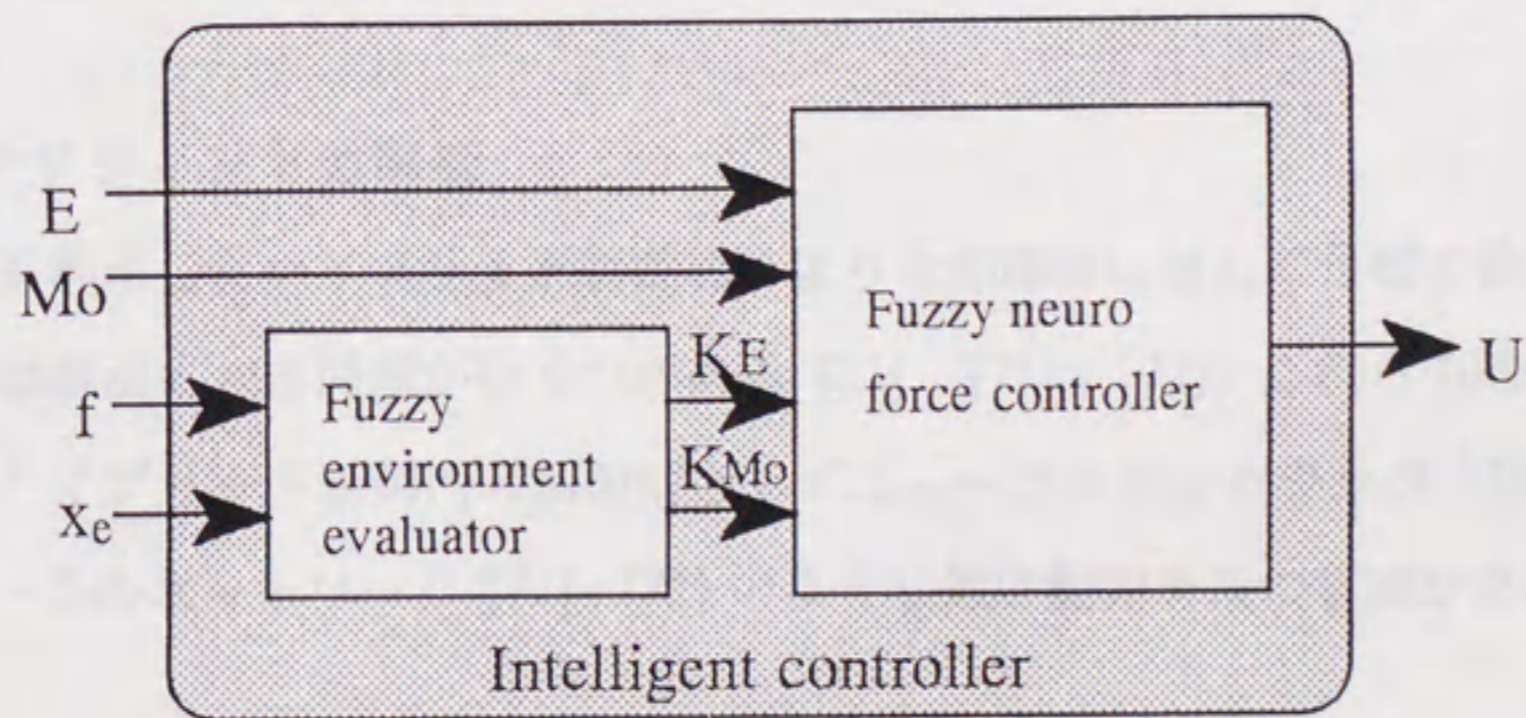


図 4.5 インテリジェント力制御器のブロック線図

義する。したがって、算出された法線ベクトルの大きさが大きければ大きいほどノイズ等による誤差の影響が少ないと見なし、ファジィ法線ベクトルの角度のファジィ度は少なくなる。

ファジィ法線ベクトルの逆方向とその時点での力制御方向ベクトルとの角度の適合度を μ_n とし、その時点での力制御方向の角度を α_f 、力センサーから得られた拘束面法線ベクトルの逆方向の角度を α_n とすると、力制御方向の角度変更量は次式で表わされる。

$$\Delta\alpha_f = (1 - \mu_n)(\alpha_n - \alpha_f) \dots\dots\dots (4.2)$$

これにより、力制御方向が拘束面法線ベクトルの逆方向と離れている場合は、力センサーから得られた情報により算出された拘束面の法線ベクトルの逆方向通りに力制御方向を変更し、力制御方向がほぼ一致している場合には、算出された拘束面の法線ベクトルの逆方向への角度変更量を小さくすることが可能となる。これらの手順をまとめたものを次に示す。

- Step 1: 力センサーから得られた拘束面法線ベクトルをファジィ単位ベクトルに変換する (図 4.4 参照)。
- Step 2: ファジィ単位ベクトルのファジィ度に比例し、ファジィ法線ベクトル方向 (角度) のファジィ度を定義。
- Step 3: 定義されたファジィ法線ベクトルの逆方向とその時点における力制御方向の角度の適合度 μ_n を求める。
- Step 4: 力制御方向を式 (4.2) で算出された力制御方向変更量の分だけ変更する。

4. 4 インテリジェント力制御

従来の制御則を利用したハイブリッド制御器により未知環境に対して目標の位置と力を得るためには解決すべき問題がいくつかある [21], [71], [72]。これらの問題を解決するには、ハイブリッド制御内の制御則としてニューラルネットワーク [21] やファジィ・ニューラルネットワーク [71], [72] のような適応能力を持つ制御が必要となる。

厳密な数学モデルを用いず熟練者の制御知識を利用して設計されたファジィ制御 [10], [61] を、学習・適応能力のあるニューラルネットワークに変換したファジィ・

ニューロ制御器の応用は、未知環境に対する位置／力制御を実現させる有効な方法の一つである。しかし、未知環境の特性がファジィ力制御ルール設計時に想定したものと大きく異なっていた場合、ファジィ・ニューロ制御器が環境に適応するために必要な時間が長くなり、その間にオーバーシュートが発生したりするような不都合が生じる可能性があった。そこで前章では、オフライン学習させたニューラルネットワークを用い、未知環境の動特性に応じてファジィ・ニューロ力制御器への入力値を調整することにより、瞬時に制御器を環境に適応させる方法を提案した [71], [72]。この方法は実際に使用する環境の材料の種類が予め限られているときには非常に有効ではあるが、学習に時間がかかるうえ、学習された材料と異なる環境に対する汎用性が問題となる場合が考えられる。そこで本章では全く未知な環境に対しても有効で、事前の学習も必要としないファジィ環境評価器を提案し、ファジィ・ニューロ制御器への入力情報の調整を行うものとする。ファジィ環境評価器とファジィ・ニューロ制御器の2つから成る本インテリジェント力制御器のブロック線図を図4.5に示し、構成を図4.6に示す。ここで Σ のニューロンは入力の和を出力し、 Π のニューロンは入力の積を出力するものである。

4. 4. 1 ファジィ・ニューロ力制御器

ファジィ・ニューロ力制御器 [71], [72] については既に第3章で説明されているので、ここでは概略だけ説明する。

ファジィ・ニューロ力制御器への入力変数は目標力との誤差量 E と力制御方向のマニピュレータ運動量 M_0 の2種類とし、前件部のファジィ変数は5種類 (PB, PS, ZO, NS, NB) とする。ファジィ変数は、式 (4.3) で表されるシグモイド関数と式 (4.5) で表されるガウシアン関数の両方を用いることにより単層のニューラルネットワークで前件部を構成するものとする。

$$f_S(u_S) = \frac{1}{1 + e^{-u_S}} \dots\dots\dots (4.3)$$

$$u_S(x) = w_o + w_i x \dots\dots\dots (4.4)$$

$$f_G(u_G) = e^{-u_G^2} \dots\dots\dots (4.5)$$

$$u_G(x) = \frac{w_o + x}{w_i} \dots\dots\dots (4.6)$$

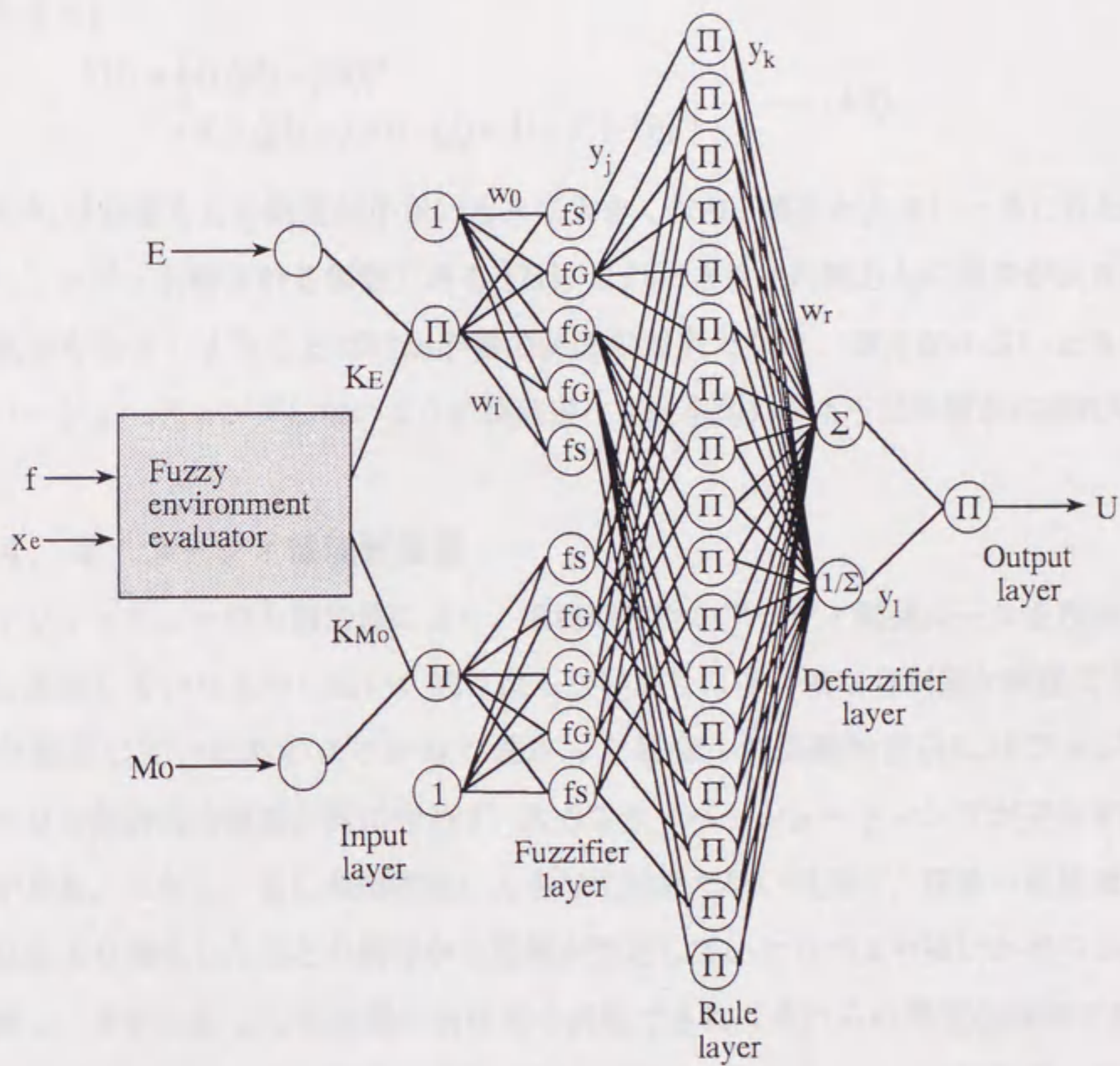


図 4.6 インテリジェント力制御器の構成

ここで、 w_0 はしきい値、 w_i はウエイト値である。

後件部はファジィ制御ルールに従い、前件部適合度の代数積に定数値をかけ合わせた各ルールからの出力和を前件部適合度の和で割って定数値出力を得るものとする。この制御器からの定数値出力が基準座標系でのマニピュレータへの力指令 U である。

ファジィ・ニューロ力制御器の学習は、次の評価関数が最小となるようにバックプロパゲーション法を用いて前件部と後件部の両方の微調整を各サンプリング毎に行うものとする。

$$Y(k) = \frac{1}{2} \{ [f_d(k) - f(k)]^2 + K_z [(f_d(k) - f(k)) - (f_d(k-1) - f(k-1))]^2 \} \dots\dots\dots (4.7)$$

ここで K_z は目標力との誤差が小さいときに大きくなり、誤差が大きいうちに0となるようにファジィ制御される係数である [71]。これにより、目標力との誤差が大きいうちは誤差を小さくすることだけが学習の評価の対象となり、誤差が小さいときにはオーバーシュートしないように誤差変化速度を弱めるような学習が可能になる。

4. 4. 2 ファジィ環境評価器

ファジィ・ニューロ力制御器により、環境の特性がファジィ制御ルールを作成したときに想定していたものに近いか或は柔らかい場合には有効な力制御が実現できる。環境が想定していたものよりかなり硬かった場合、制御開始直後にはファジィ・ニューロ力制御器の適応が間に合わず、大きなオーバーシュートが発生する可能性がある。しかし、もし未知環境に力を与え始めた早い段階で、環境の変位量とその変位により発生した力との関係から環境が想定していたものより硬いか柔らかいかを判断し、それに応じて制御器の前件部を調整できればそれらの問題を解決できる。そこで本章では、ファジィ環境評価器を用いてファジィ・ニューロ力制御器への入力情報である誤差 E とマニピュレータ運動量 M_0 に、それぞれ環境の硬さに応じた調整係数 K_E 、 K_{M_0} を乗じてやることにより制御器の前件部を瞬時に調整する方法を説明する。

本ファジィ環境評価器で瞬時に未知環境の硬さを評価し、ファジィ・ニューロ力制御器の入力値に環境の硬さに応じた調整係数を乗じてやることにより、設計時に定義されたメンバーシップ関数を環境の硬さに応じ、横に広げたり狭めたりしたような形

4.7.1 調整係数の影響

調整係数は、メンバーシップ関数の幅を調整する。調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

調整係数が1未満の場合、メンバーシップ関数はより広くなる。調整係数が1より大きい場合、メンバーシップ関数はより狭くなる。調整係数が1の場合、メンバーシップ関数は元のままになる。

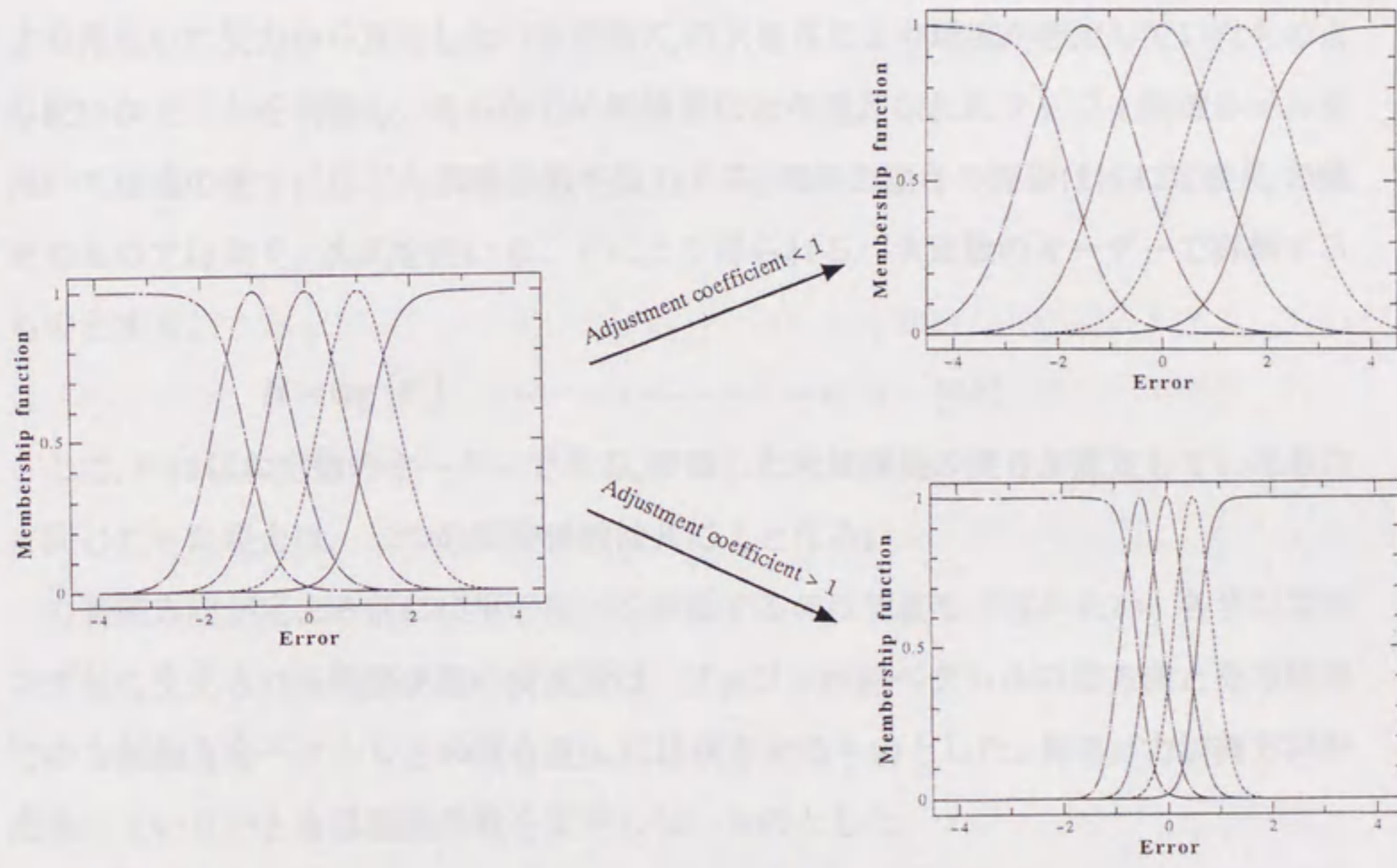


図 4.7 メンバーシップ関数の調整

状に変更するのと同じ効果が得られる(図4.7参照)。つまり瞬時に制御器の前件部の定義を変更することが可能となる。例えば、もし環境が想定していたものよりも硬かった場合は、その硬さ加減により制御器への入力情報である目標力との誤差量を小さく、力制御方向のマニピュレータの運動量を大きく変換してやることにより、実際にはまだ誤差が大きい時点で誤差が小さいときのようにきめ細かく制御され、環境を押し速さの定義も変更できるようになる。

ファジィ環境評価器への入力は、環境の力制御方向の変位量とその変位により発生した反力とし、出力はファジィ・ニューロ力制御器への入力値EとMoに対する調整係数 K_E と K_{M_0} である。ファジィ環境評価器は環境の力制御方向の変位量とその変位により発生した反力から算出したバネ定数 K_s の大きさにより環境が想定していたものより硬いかどうかを判断し、あらかじめ熟練者により設定されたファジィ制御ルールを用いて環境の硬さに応じた調整係数を出力する。環境の硬さの評価はばね定数 K_s の値そのものではなく、次式を用いることにより得られるバネ定数のオーダーで評価するものとする。

$$H = \log(K_s) \dots\dots\dots (4.8)$$

ここで、 H はばね定数のオーダーである。評価した未知環境の硬さが想定していたものと同じだった場合は、二つの調整係数は共に1となる。

力制御方向が定まる前に環境の硬さを評価するのは望ましくないため、各サンプリング毎に変更される調整係数の変更量は、ファジィ法線ベクトルの逆方向とその時点での力制御方向ベクトルとの適合度 μ_s に比例させるものとした。即ち、力制御方向が定まっていないときは調整係数を変更しないものとした。

4.5 シミュレーション

提案したインテリジェント制御器の有効性を確かめるため、コンピュータによるシミュレーションを行った。シミュレーションには平面2自由度ロボットマニピュレータをモデルとして用い、サンプリング時間は1msとした。また、量子化誤差と演算時間遅れは無視できる程度、センサーおよびモータの動作遅れはないものと仮定した。環境に加える目標力は、2秒毎に20Nと10Nの間で変化するステップ信号とし、初期の力制御方向は基準座標系のx軸方向とした。シミュレーションで用いる環境は

次式のようにモデル化した。

$$f = M_p \ddot{x}_e + B_p \dot{x}_e + K_p x_e \dots\dots\dots (4.9)$$

ここで、 x_e は環境表面の変位である。ロボットマニピュレータのエンドエフェクタと環境との摩擦係数は0.05に設定した。

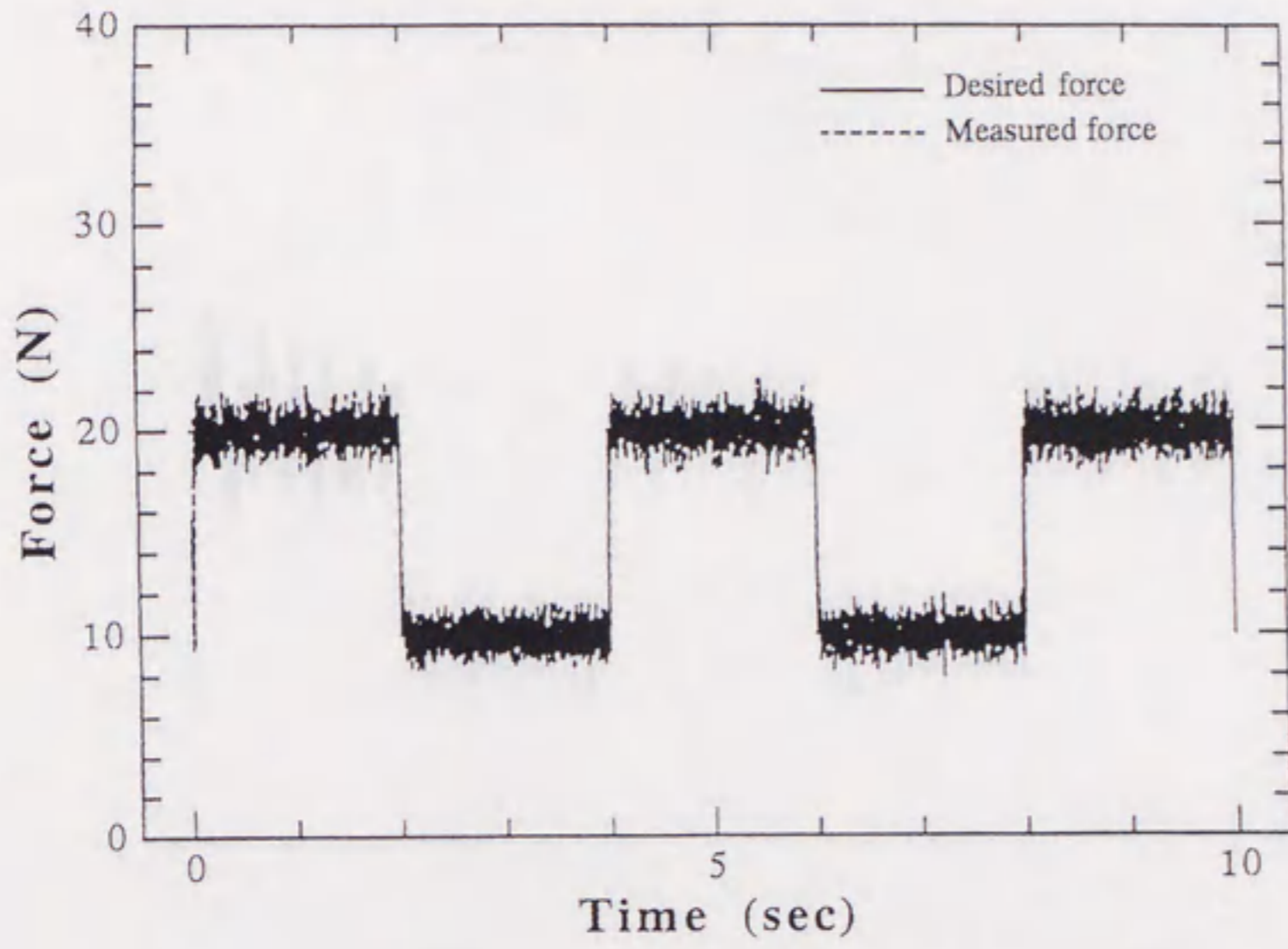
また、ノイズによる誤差の影響を考慮するため、各サンプリング時間ごとに力センサーにより得られた力制御方向の力成分とその方向に直交する方向の力成分の両方に、 ± 1 N以内でランダムに変化する誤差を含ませた。

4. 5. 1 ファジィベクトルを利用した力制御方向変更

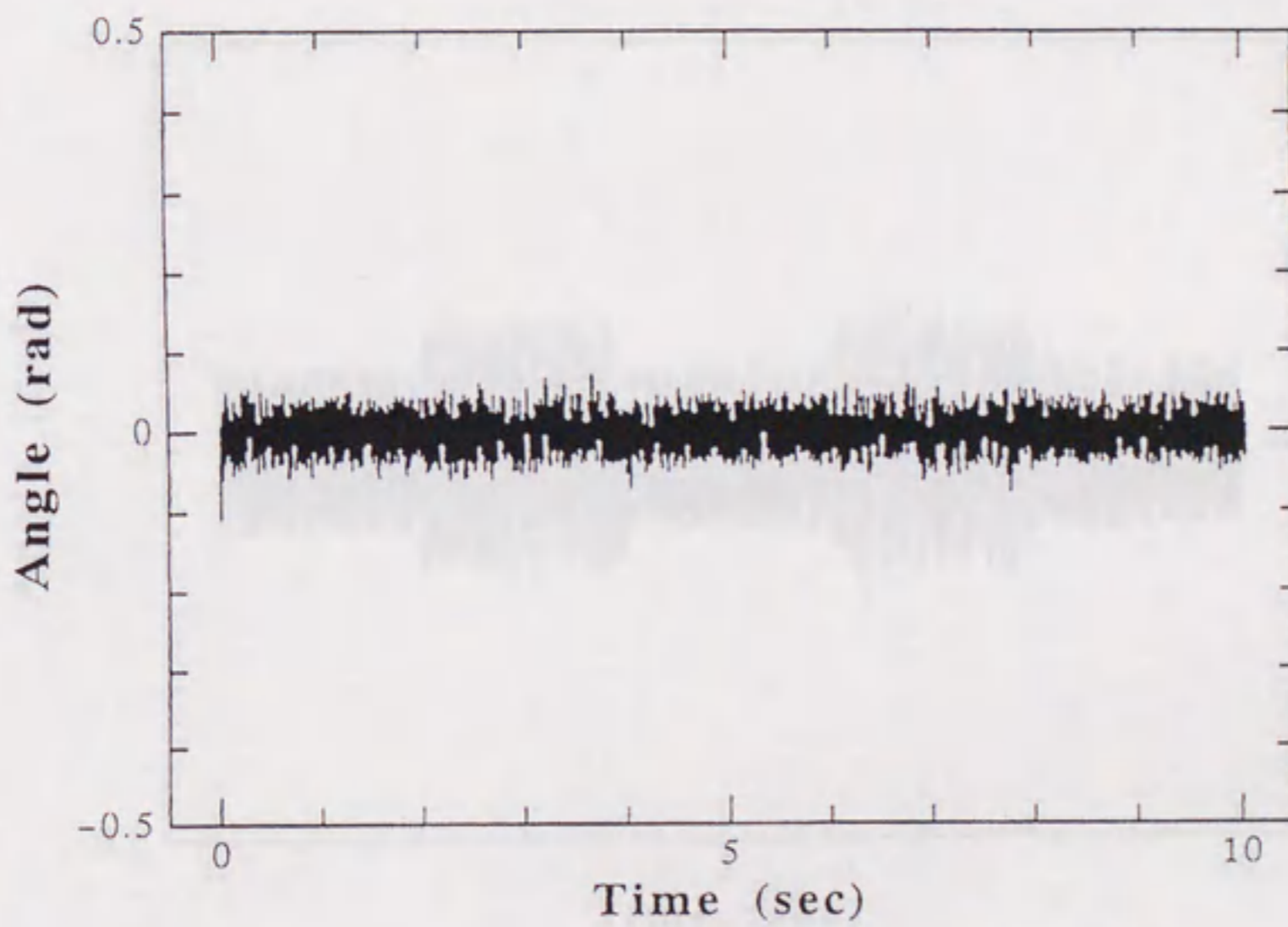
未知環境による拘束面の法線方向の逆方向と力制御方向を力制御しながら一致させるアルゴリズムの有効性を確認するため、拘束面の角度を基準座標系のY軸から0度と30度に設定しシミュレーションを行った。環境の硬さはファジィ制御ルール設計時に想定したものと同じに設定した。

まず拘束面の角度が0度するとき、すなわち最初から力制御方向が拘束面法線方向の逆方向と一致しているときに、ファジィベクトルを利用して力制御方向を調整したときの結果を図4.8に、ファジィベクトルを利用せず拘束面法線方向の逆方向と力制御方向との差をそのまま力制御方向の変更量としたときの結果を図4.9に示す。これらの図中の(a)は目標力に対する追従性を示したもので、(b)は基準座標系に対する力制御方向を示している。拘束面の角度が0度の場合には既に方向が一致しているため、ファジィベクトルを利用している図4.8(b)は方向の変更量が小さく、しかも環境に加えられた力の大きさにほとんど影響を受けていないことがわかる。ファジィベクトルを利用していない図4.9(b)は既に方向が一致しているにも関わらず、ノイズの影響をそのまま方向の変更反映させているため各サンプリング毎にかなり変動し、さらに環境に加えられた力の大きさに強く影響を受けているのがわかる。また、目標力に対する追従性もファジィベクトルを利用したほうが良い結果が得られていることがわかる。

次に、拘束面の角度が30度ときにファジィベクトルを利用して力制御方向を調整したときの結果を図4.10に、ファジィベクトルを利用せず拘束面法線方向の逆方向と力制御方向との差をそのまま力制御方向の変更量としたときの結果を図4.11に示す。これらの結果より、ファジィベクトルを利用した場合は、制御開始直後に力制御方向

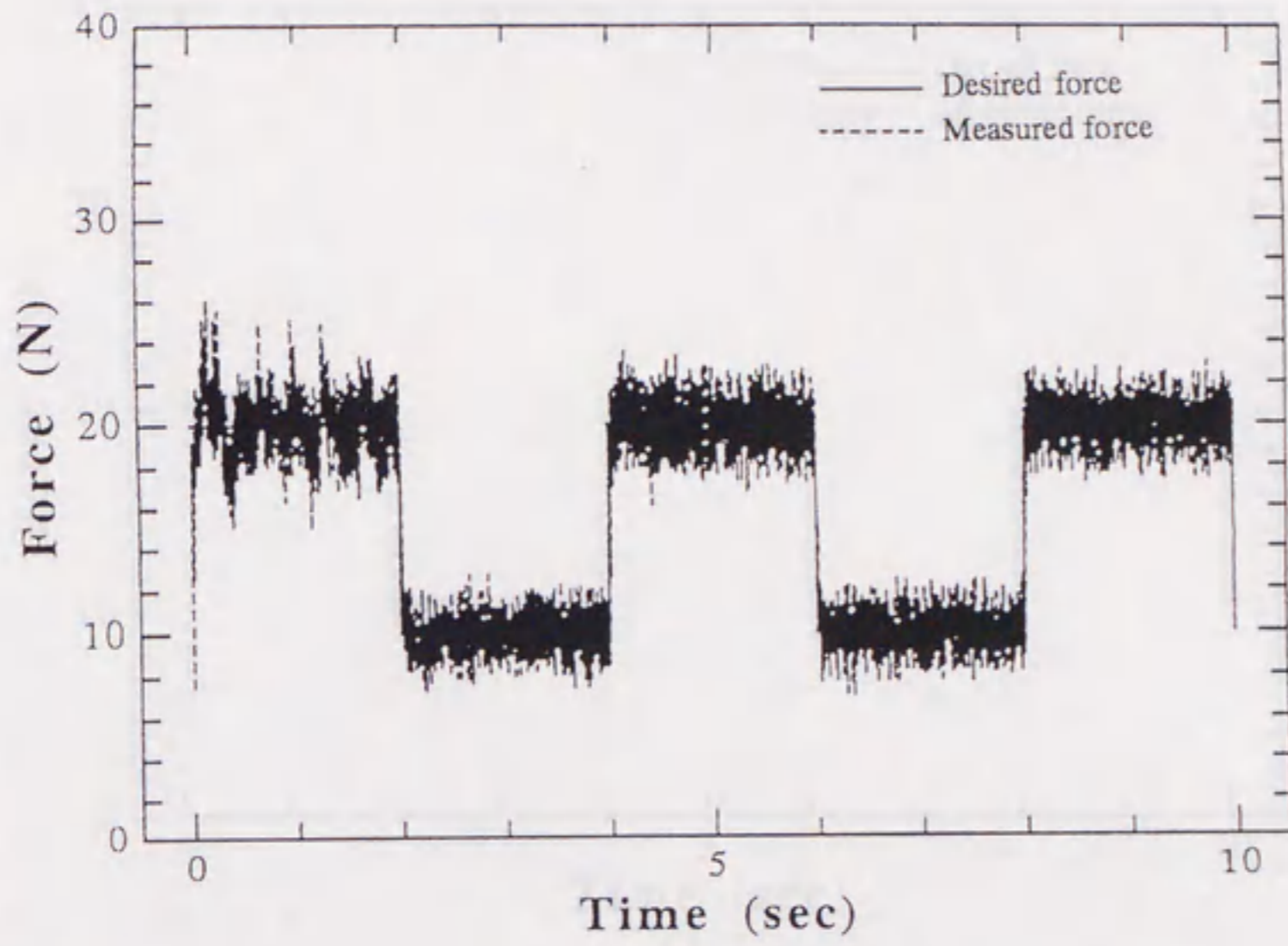


(a) 力制御結果

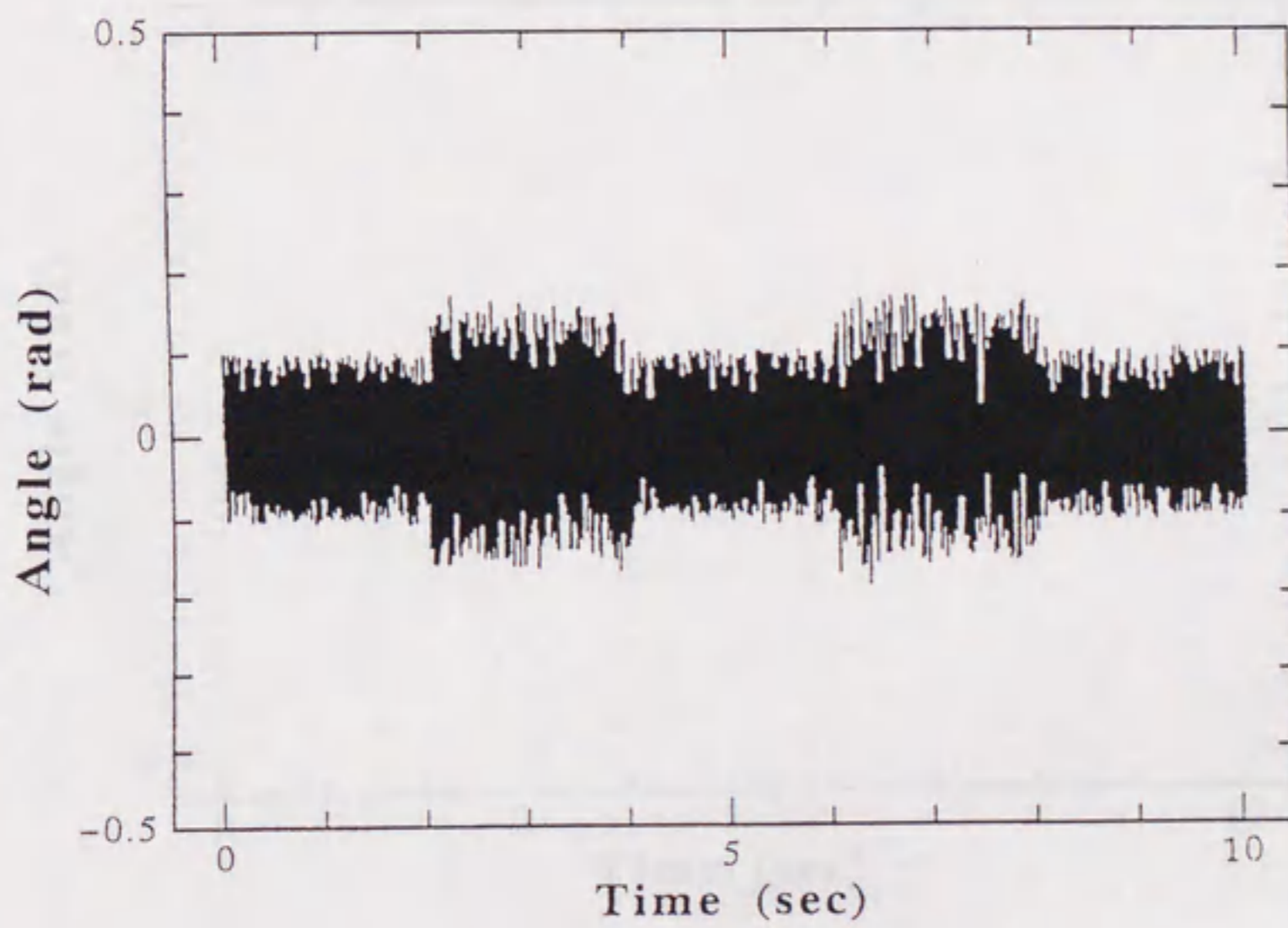


(b) 力制御方向

図 4.8 ファジィベクトルを用いたときのシミュレーション結果 (0度)

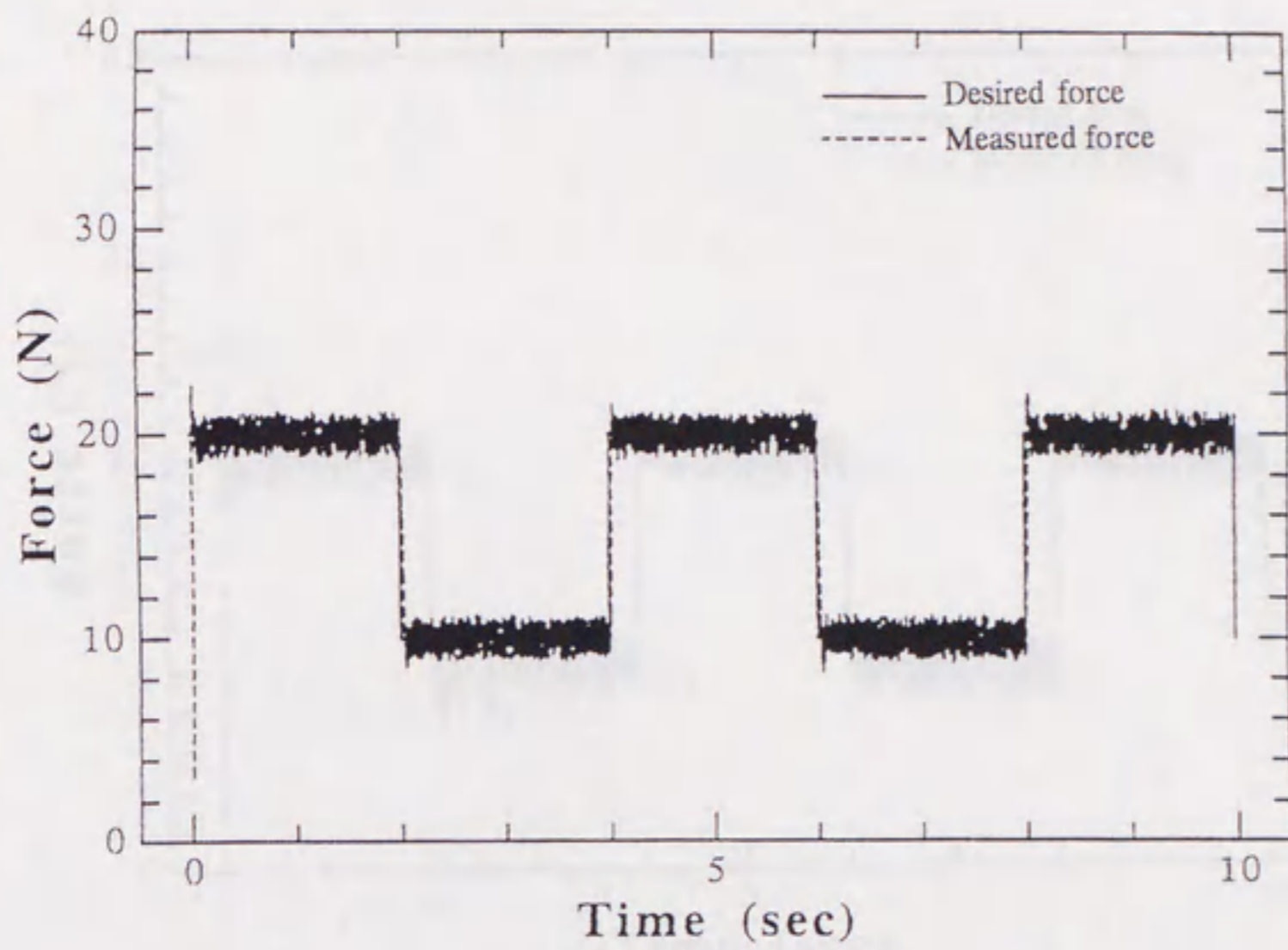


(a) 力制御結果

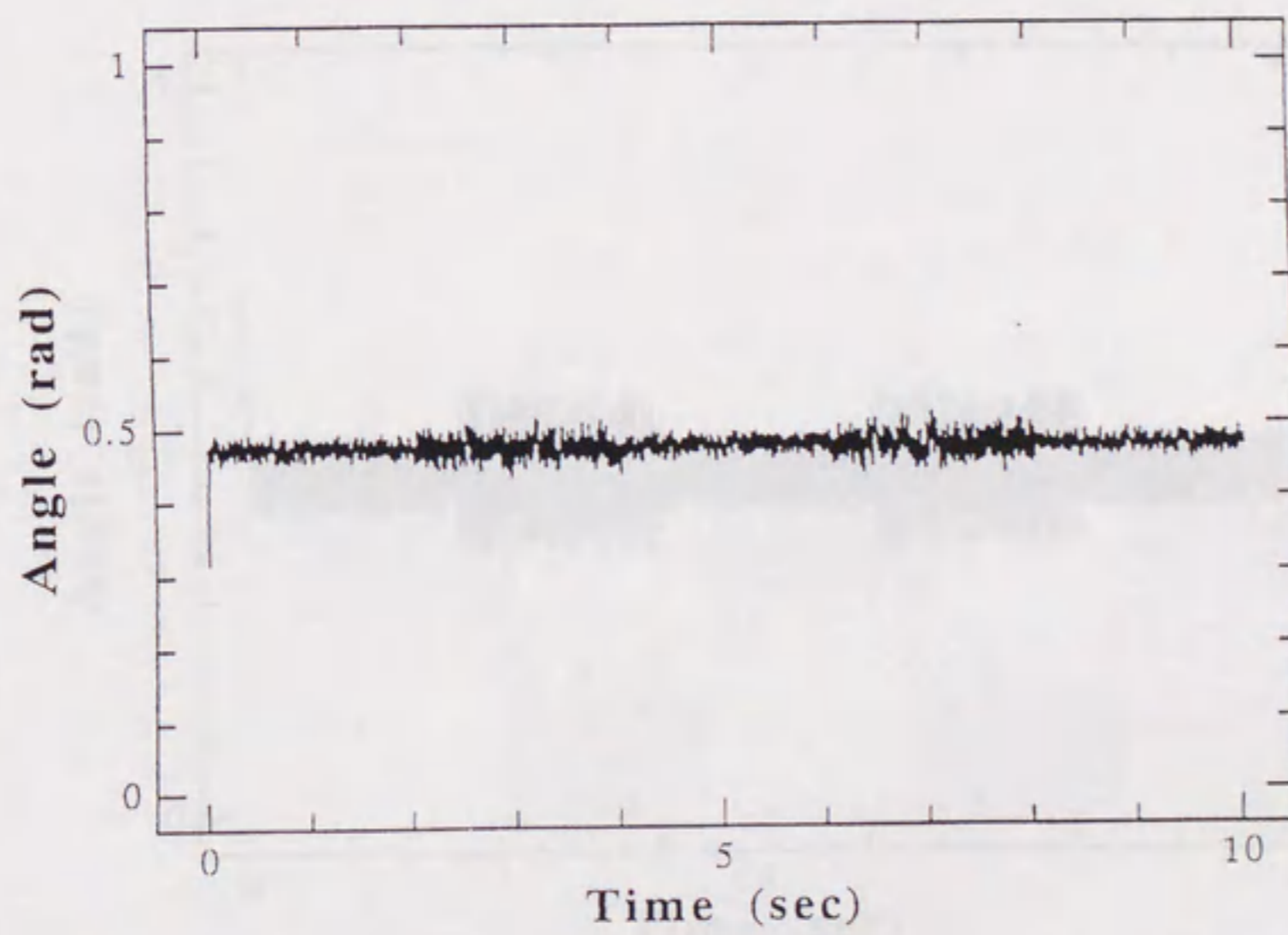


(b) 力制御方向

図 4.9 ファジィベクトルを用いなかったときのシミュレーション結果 (0度)

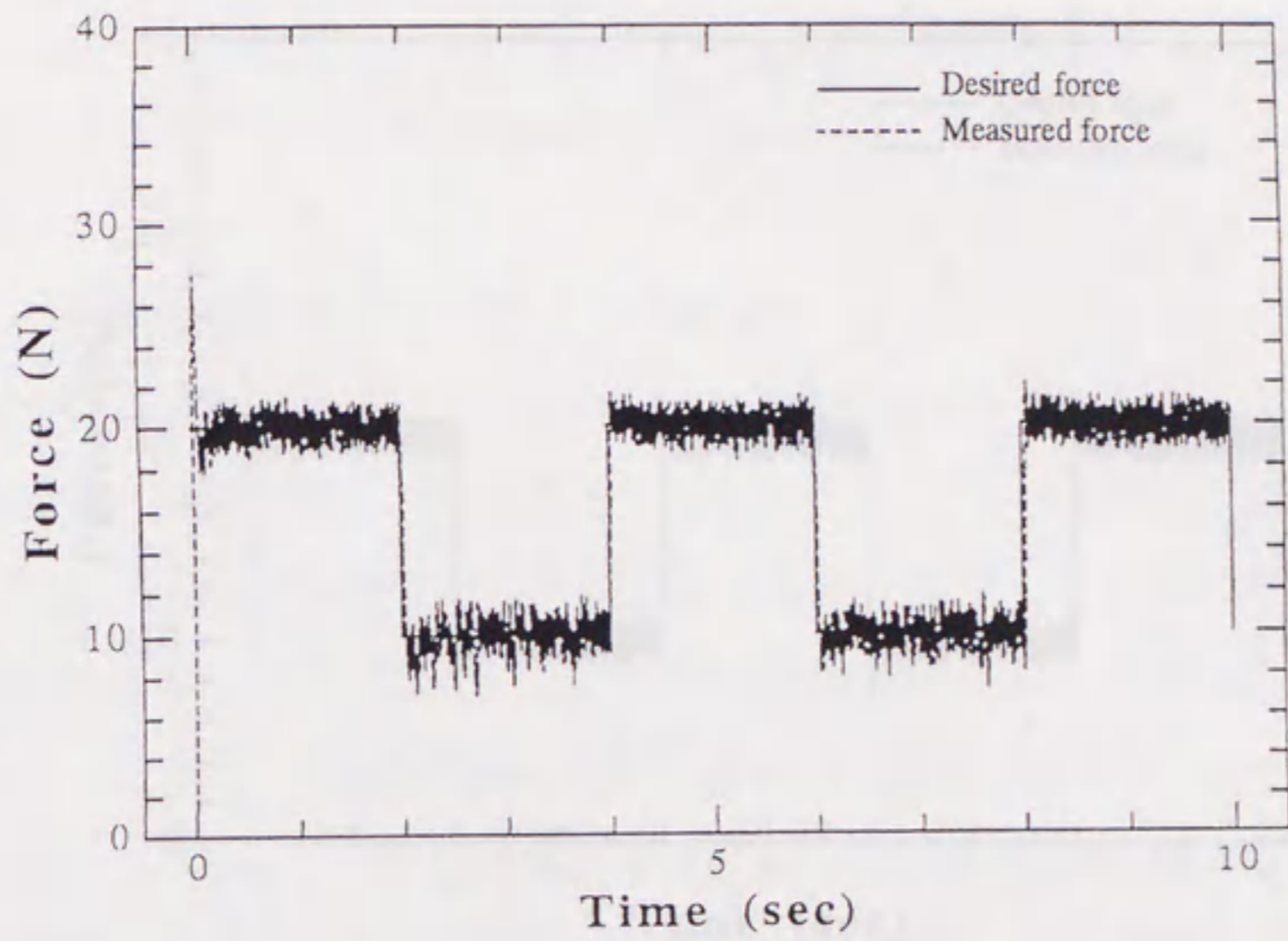


(a) 力制御結果

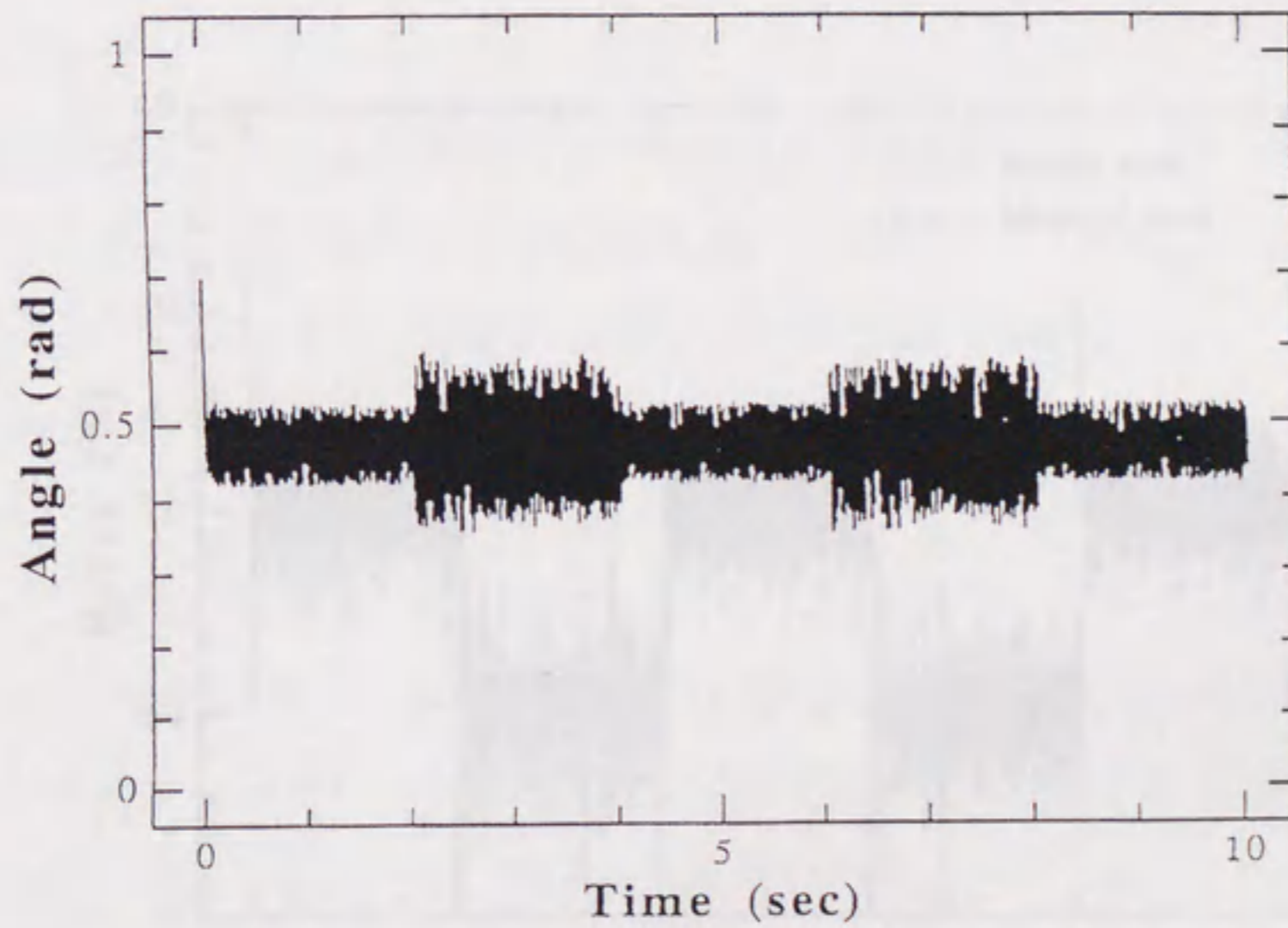


(b) 力制御方向

図 4.10 ファジィベクトルを用いたときのシミュレーション結果 (30 度)



(a) 力制御結果



(b) 力制御方向

図 4.11 ファジィベクトルを用いなかったときのシミュレーション結果 (30度)

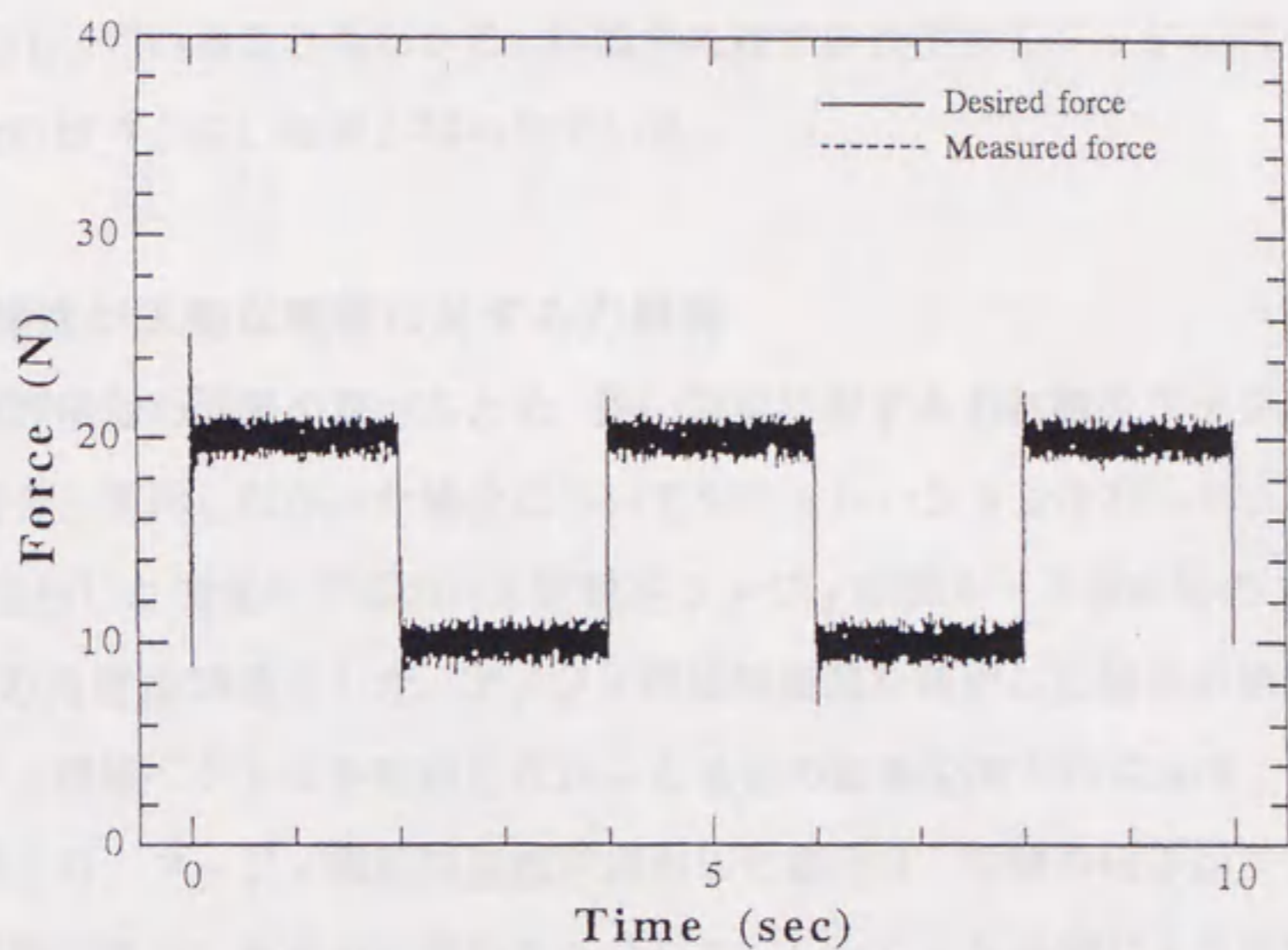


図 4.12 ファジィ環境評価器を用いたときの力制御シミュレーション結果

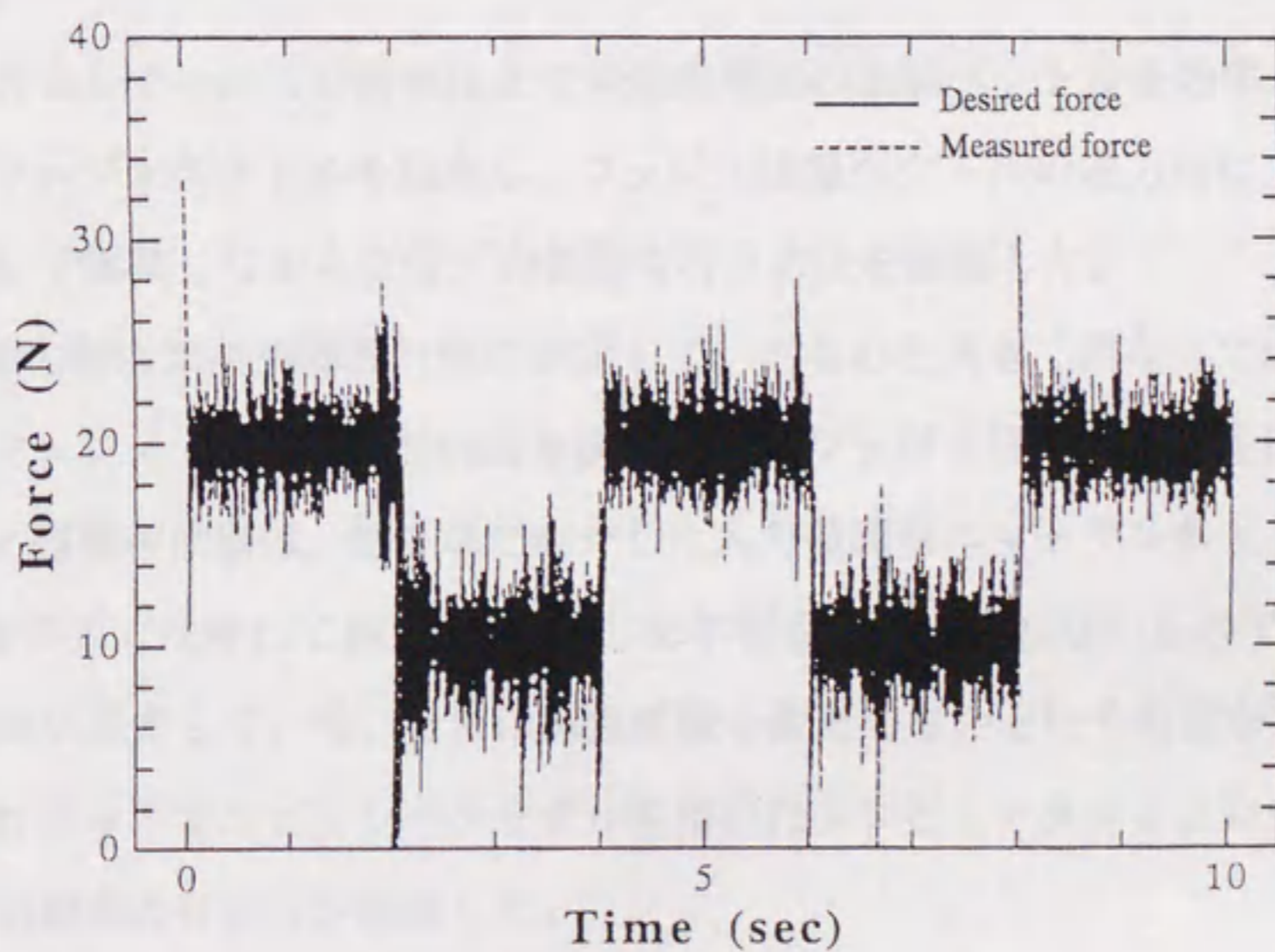


図 4.13 ファジィ環境評価器を用いなかったときの力制御シミュレーション結果

が定まり、その後もノイズによる影響や環境に加えられた力の大きさの変化による影響もあまり受けていないことがわかる。またファジィベクトルを利用していない場合には、力センサーから得られた拘束面法線ベクトルの大きさが小さい制御開始直後に、誤った方向に変化していることもわかる。目標力に対する追従性もファジィベクトルを利用した場合のほうが良い結果が得られている。

4. 5. 2 特性が未知な環境に対する力制御

ファジィ環境評価器の効果を調べるため、硬い環境に対する力制御をファジィ評価器を利用した場合と利用しなかった場合についてシミュレーションを行った。ここでは、式(4.9)で表わした環境モデルのバネ定数をファジィ制御ルール設計時の10倍に変更し、拘束面の角度は30度とした。ファジィ環境評価器を利用した場合の結果を図4.12に、ファジィ環境ベクトルを利用しなかった場合の結果を図4.13に示す。

これらの結果より、ファジィ環境評価器を利用した場合は、環境の硬さがファジィ制御ルール設計時に想定したものと同じときのシミュレーションと同じような結果が得られるが、ファジィ環境評価器を利用しなかった場合は、かなり悪化していることがわかる。

4. 6 結 言

ノイズを含む力センサーからの情報により未知拘束面の法線ベクトルを効率良く算出するために、ファジィベクトルを提案し、ファジィ法線ベクトルの逆方向に力制御方向をオンラインで調整しながら位置/力制御を行う方法を提案した。

また、未知環境の特性が制御器設計時に想定していたものと大きく異なっている場合でも、瞬時にファジィ・ニューロ制御器を調整できるファジィ環境評価器を提案した。このファジィ環境評価器は、第3章で紹介した入力値調整ニューラルネットワークと比べ、汎用性の面では優れており、あらかじめ学習を行う必要もないものである。また、力制御方向が定まっていないときは調整係数を変更しないという特徴がある。

平面2自由度ロボットマニピュレータモデルを用いたコンピュータシミュレーションで、提案した制御法の有効性を確認した。

第5章 ファジィ・ニューロを利用した複腕ロボットによる未知対象物の操り

5.1 緒言

ロボットマニピュレータの高度な作業内容の一つとして複腕での対象物の操りが挙げられる。これまでに複腕ロボットマニピュレータによる対象物の操りについて多くの研究がなされており、いろいろな制御法や操り力の決定法が提案されてきた [1] ~ [6]。制御法では、マスター・スレイブマニピュレータの概念を利用したもの [5] や、複数のマニピュレータのうち6自由度を位置制御に用い、残りの自由度を力制御に用いるハイブリッド制御タイプのもの [6]、あるいは対象物を直接制御するもの [2] ~ [4] などがある。このうち各マニピュレータを力発生器と見なして対象物を直接制御する方法が、対象物を操るのに最も自然な方法と思われる。しかし、これまでの研究は対象物の重量などの特性が既知であることを前提としており、重量などの特性が曖昧であったり未知であるときの制御法、あるいは対象物に大きな外乱が作用するときの制御法は提案されていなかった。

一方、曖昧な情報の扱いができ、厳密な数学モデルを使うことなく制御できるファジィ制御 [10], [61], [62] や経験から学習できる能力を持つニューラルネットワークは、本論文でも既に紹介している通りロボット制御の分野で積極的に取り入れられており [20] ~ [39]、最近では多くの研究者により、これら両方の長所を合わせたファジィ・ニューラルネットワーク (FNN) 制御の研究も積極的に行われている [79], [96], [97]。例えば、第3章でも説明したように、ロボットマニピュレータで動特性の未知な対象物に目標通りの力を与えるためには環境適応能力のあるニューラルネットワークを利用することなどが有効であるが、適応型のニューラルネットワーク制御器でも瞬時に適応できるわけではなく、適応するまでの間オーバーシュート等の

危険性がある。そこで、FNN制御では、あらかじめファジィ制御の方法を用いてオーバーシュートしないような制御器を設計し、それをニューラルネットワークでモデル化し、バックプロパゲーション法で学習させることにより制御器を調整するようなことが実現できる [79]。しかしながら、これまでこの様なFNNを利用して質量などの特性が未知な対象物を制御したり、大きな外乱が作用する対象物を制御しようとした例はなかった。

本章では、FNNを利用して、複腕ロボット(ここでは2台のロボットマニピュレータとした)により重量などの特性が未知である対象物 [74] 或いは大きな外乱が作用する対象物 [76] を操る方法を提案する。本制御器は、対象物の目標値からの誤差量とその変化量を入力情報とし、未知対象物を制御するための力指令を出力するメインのFNN制御器と、マニピュレータの運動量と目標力との誤差量を入力情報とし、動特性の未知な環境に対しメイン制御器からの力指令通りの力を各マニピュレータから出力させるための複数のサブのFNN制御器の複数の制御器から構成されている。本制御器により、未知対象物を目標通り操るためのメイン制御器から指示された目標力は複腕ロボット(複数のマニピュレータ)に割り振られ、各サブ制御器(各マニピュレータの制御器)により割り振られた通りの力を発生させることが可能となる。ここでは問題を簡単にするために、ロボットマニピュレータの台数を必要最低数の2台とした。

サブのファジィ・ニューラル制御器において、環境の動特性が設計時に予測したものよりも柔らかいか少し硬めの場合は特に問題なく力制御できる。したがって、あらかじめ硬い環境を想定しておけばほとんどの場合問題ない。しかし、もし環境が想定したものよりかなり硬かった場合は、大きなオーバーシュティングが発生する等の問題が生じる。そのため、第3章ではFNNの入力値調整ニューラルネットワーク(IANN)を提案し、第4章ではより柔軟性の高いファジィ環境評価器を提案したが、本章では設計時の予測よりも硬い環境に対して大きなオーバーシュティングを押しえるために、制御器内に標準環境モデルを置き、必要に応じ補償を行う機能を付けるような簡略化された方法を提案する [74], [76]。

対象物を確実に把持するためには、把持力すなわち内力の決定方法が重要な課題となるが [1], [98], 本研究では対象物が常に十分な内力で把持されているものと仮定し、内力の決定方法は本研究の対象外とする。

本制御の有効性はコンピュータシミュレーションにより確認した。

5. 2 対象物の操り

まず対象物をどのように操るかを考える。対象物の操り方法はこれまでにいろいろ提案されてきた。例えば、マスター・スレイブマニピュレータを利用したタイプのもの [5] は、マスターマニピュレータが対象物の位置制御を行いスレイブマニピュレータが対象物への力制御を行うことにより対象物を操る方法であり、ハイブリッド制御タイプのもの [6] は、マニピュレータを位置制御用と力制御用に分けるのではなく、複数のマニピュレータのうち6自由度を位置制御に用い、残りの自由度を力制御に用いることにより対象物を操る方法である。また対象物を直接制御するタイプのもの [2] ~ [4] は、目標通り対象物を操るのに必要な力を運動方程式より求め、その力をマニピュレータにより発生させることにより対象物を操る方法である。本研究では、最も自然な方法と思われる対象物を直接制御するタイプのものを採用した。

対象物の運動方程式は次式で表される。ただし、ここではマニピュレータは対象物と点接触し、任意の力を加えられるものとする。

$$m\ddot{x} = f_{ext} + \sum_{i=1}^n f_i \dots\dots\dots (5.1)$$

$$I\dot{\omega} + \omega \times I\omega = \tau_{ext} + \sum_{i=1}^n \tau_i \dots\dots\dots (5.2)$$

$$\tau_i = r_i \times f_i \dots\dots\dots (5.3)$$

ここで、

m : 対象物の重量

x : 対象物重心の基準座標系での位置ベクトル

f_{ext} : 外部から対象物重心に働く力

f_i : マニピュレータ i で対象物重心に加える力

I : 対象物の重心回りの慣性行列

ω : 対象物の回転角速度ベクトル

τ_{ext} : 外部から対象物重心に働くトルク

τ_i : マニピュレータ i で対象物重心に加えるトルク

r_i : 対象物重心から見たマニピュレータ i との接触点の位置

とする。ただし、本研究では、図5.1に示すような平面2自由度ロボットマニピュレータをモデルとして考えたので、重力の影響は無視した。

式(5.1)～(5.3)より、 f_{ext} 、 r_{ext} 及び r_i が既知の場合、 f_i を制御することで対象物の位置と姿勢を任意に制御できることがわかる。ここで、各マニピュレータからの発生力とそれらの合力との関係は次式で表される。

$$\left| \sum_{i=1}^n f_i \right| \leq |f_1| + \dots + |f_n| \dots\dots\dots (5.4)$$

各マニピュレータからの発生力が対象物の把持力と逆方向を向かないとすると、上式での右辺と左辺との差が内力、すなわち対象物の把持力となる。

5. 3 制御器の構成

本制御器は未知対象物を制御するための力指令を出力するメインのファジィ・ニューラル制御器と、各マニピュレータを力発生器と考え、動特性の未知な環境に対しメイン制御器からの力指令通りの力を各腕から出力させるための二つのサブのファジィ・ニューラル制御器の複数の制御器から構成されている。このシステムのブロック線図を図5.2に示す。

対象物の制御器では入力情報を基に各マニピュレータで発生すべき目標力を出力し、その目標力通りの力を各マニピュレータで発生するよう各マニピュレータ力制御器でフィードバック制御する。各マニピュレータから発生された力により対象物が運動し、その情報が対象物の制御器へフィードバックされる。

5. 4 未知対象物の制御

本研究では各マニピュレータを力発生器と考え、任意の力を対象物に加えられるものとする。問題を簡単にするため、対象物を平行移動させること、つまり式(5.1)で表される運動の制御のみを考えるものとする。

対象物操りのための目標力(マニピュレータで発生させる合力から内力のための力を除いたもの)は、各マニピュレータで平等に分担するものとする。ただし、各マニピュレータの中で目標操り力を加えることができないものがある場合、つまり対象物操りのための目標力がそのマニピュレータにより対象物を引くかたちになる場合のも

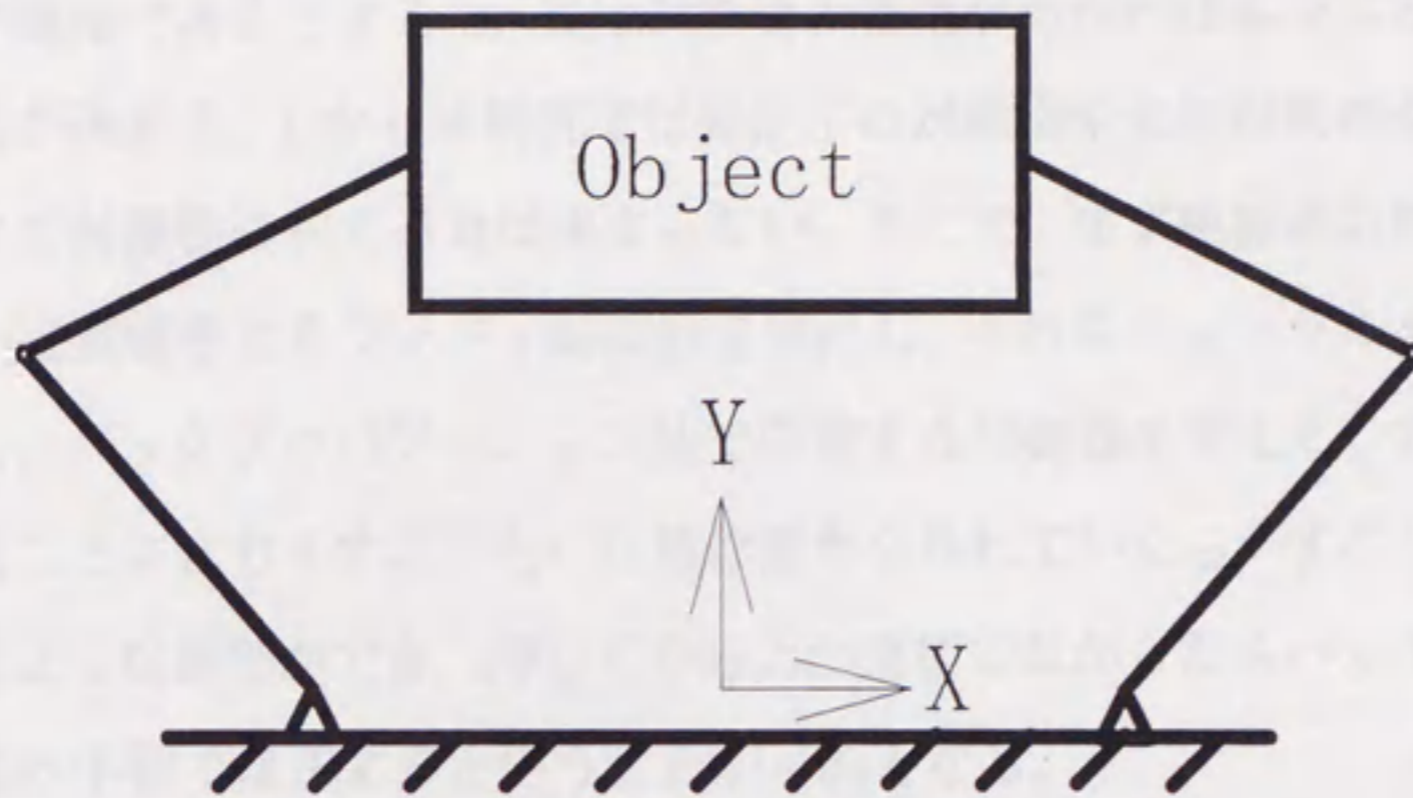


図 5.1 平面 2 自由度ロボットマニピュレータにより把持された対象物

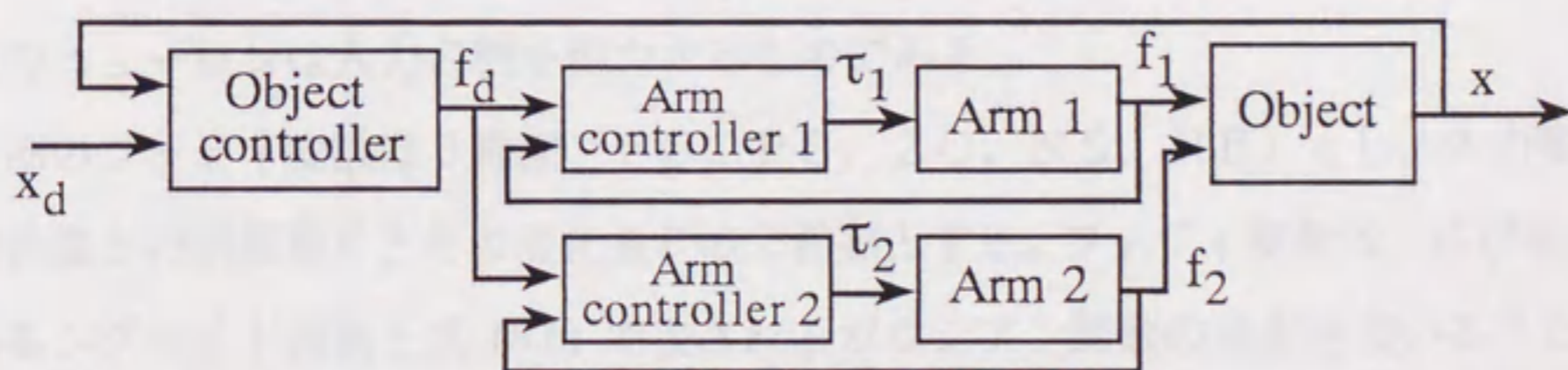


図 5.2 システムのブロック線図

のを除き、残りのマニピュレータで平等に分担するものとする。また、対象物とマニピュレータとの接触は点接触とし、適当な内力を加えている限り滑ったり離れたりしないものとする。

式 (5.1) より、目標通りの加速度を発生するには、外部から働く力と対象物の把持に必要な力が既知であるとする、後は対象物の重量がわかれば各マニピュレータで発生させる力が決まる。しかし本研究では外乱下の対象物や未知対象物を扱うためマニピュレータで対象物に加える力は決まらない。そこで、まず熟練者の制御知識を用いて後件部が定数値をとるファジィ制御器を設計し、それをニューラルネットワークでモデル化し、バックプロパゲーション法で学習する制御器を考える。FNNを利用して制御することにより「すこし右に目標位置から外れていたなら、すこし左に押してやる」というような制御ができ、押してやる力が適切でなかったらバックプロパゲーション法による学習で修正するということが可能となる。

5. 4. 1 対象物ファジィ・ニューラルネットワーク制御

まずファジィ理論を用いた制御を考える。制御器への入力情報は、対象物重心位置の目標位置からの誤差量とその変化量とする。

ファジィ理論による制御器の設計は、メンバーシップ関数と制御ルールを決定することによる。メンバーシップ関数と制御ルールを決定した後は、それらを図5.3のようなニューラルネットワークでモデル化する。ここで Σ のニューロンは入力の和を出力し、 Π のニューロンは入力の積を出力するものである。

前件部のファジィ変数は5種類 (PB, PS, ZO, NS, NB) とし、入力変数は目標位置との誤差量 E とその変化量 \dot{E} の2種類とする。ファジィ変数は、式 (5.4) で表されるシグモイド関数と式 (5.6) で表されるガウシアン関数の両方を用いることにより単層のニューラルネットワークで前件部を構成するものとする。

$$f_S(u_S) = \frac{1}{1 + e^{-u_S}} \dots\dots\dots (5.5)$$

$$u_S(x) = w_o + w_i x \dots\dots\dots (5.6)$$

$$f_G(u_G) = e^{-u_G^2} \dots\dots\dots (5.7)$$

$$u_G(x) = \frac{w_o + x}{w_i} \dots\dots\dots (5.8)$$

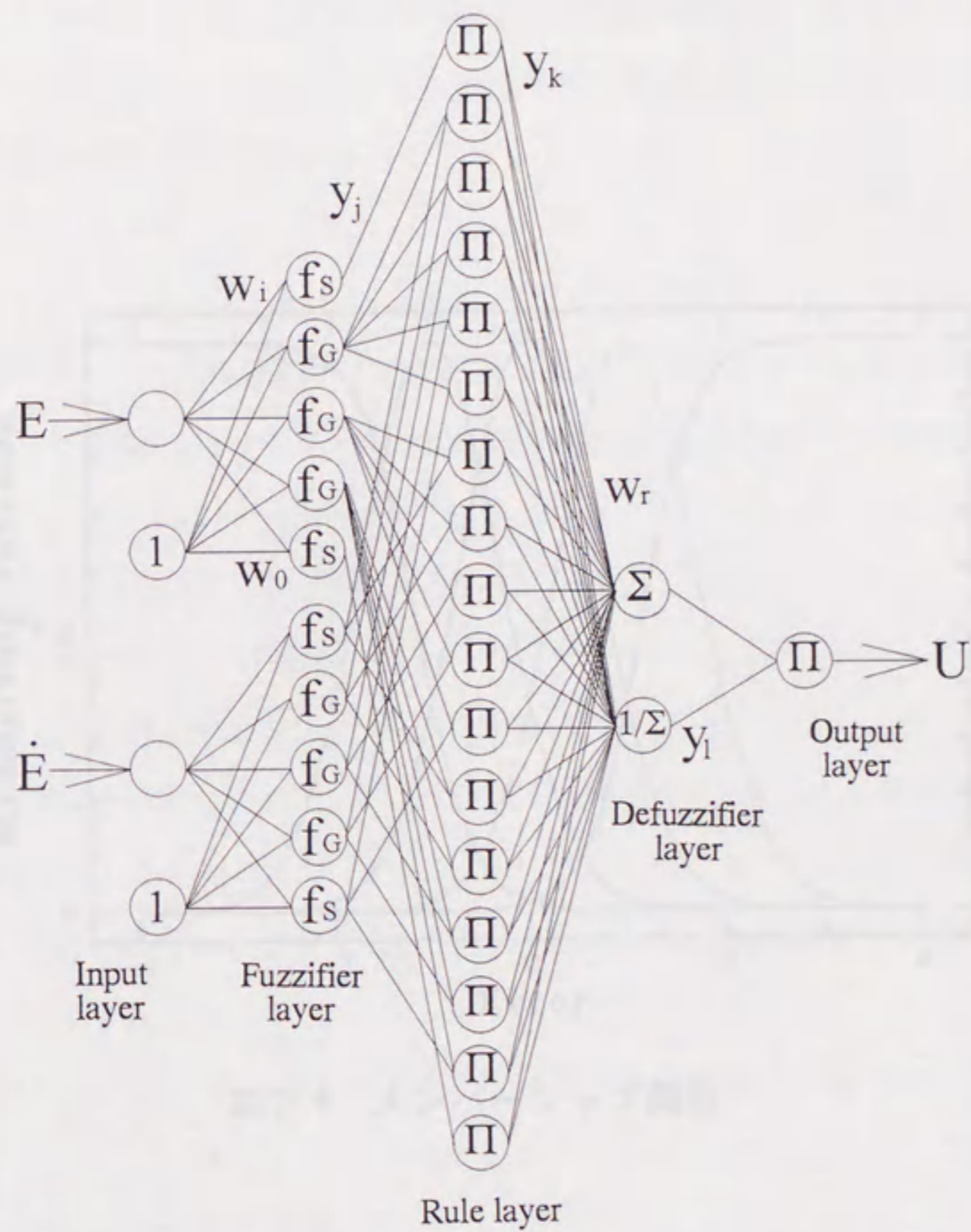


図 5.3 ファジィ・ニューラルネットワークの構成

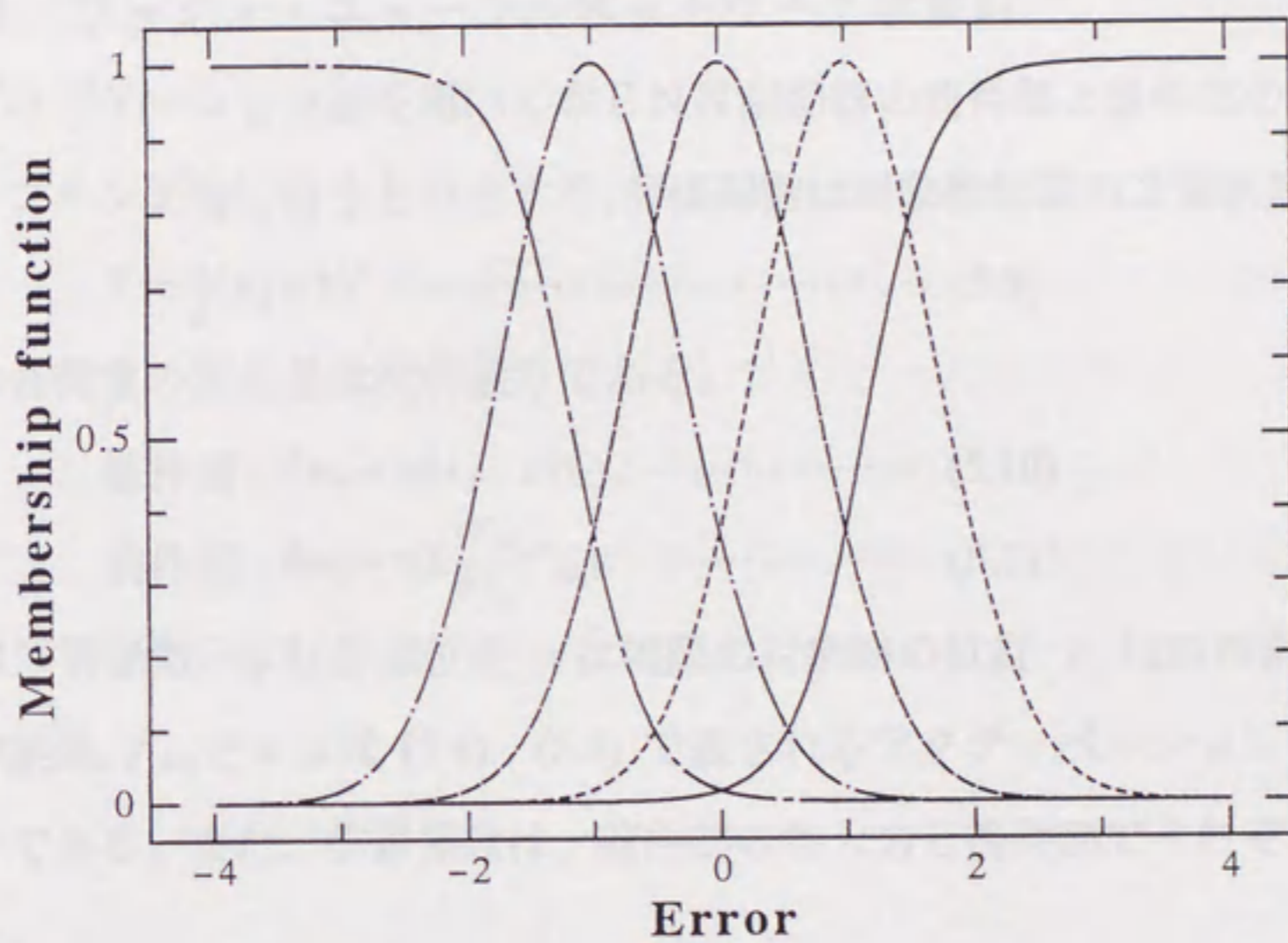


図 5.4 メンバーシップ関数

ここで、 w_0 はしきい値、 w_i は結合荷重である。ガウシアン関数では、これらの値はそれぞれメンバーシップ関数の中心値と偏差値を表す。これらの関数により図5.4に示すようなメンバーシップ関数が表現できる。

後件部は I F - T H E N の制御ルールに従い、前件部適合度の代数積に定数値をかけ合わせた各ルールからの出力和を前件部適合度の和で割って定数値出力を得るものとする。この F N N からの出力が未知対象物制御器からの出力、つまり各マニピュレータから発生させる目標力 U である。

5. 4. 2 ファジィ・ニューラルネットワーク学習則

バックプロパゲーション法を用いて本 F N N 制御器の前件部と後件部の両方の微調整を各サンプリング毎に行うものとする。評価関数は対象物位置の 2 乗誤差値とする。

$$Y = \frac{1}{2}(x_d - x)^2 \dots\dots\dots (5.9)$$

各部での結合荷重の変化量は次の通りである。

$$\text{後件部} : \Delta w_r = \eta(x_d - x)y_l y_k \dots\dots\dots (5.10)$$

$$\text{前件部} : \Delta w_i = \eta \sum \frac{\partial e_{ij}}{\partial y_{ij}} f'_{(w)} u' \dots\dots\dots (5.11)$$

ここで η は学習係数、 x_d は目標位置、 x は実際の対象物の位置、 y_{ij} は前件部出力、 e_{ij} は前件部出力誤差、 $f'_{(w)}$ と u' は式 (5.4)、(5.6) で表されるアクティベーション関数とその入力の微分である。また、学習係数は、前件部の各入力と後件部でそれぞれ別々の値に設定した。

この学習則により、メンバーシップ関数の定義や制御器からの出力が調整される。

5. 5 マニピュレータの制御

未知対象物制御器（メイン制御器）で出力された目標力通りの力を各マニピュレータにより発生できなければ対象物は予定通り運動しない。したがって、目標力通りの力を各マニピュレータで発生できるということが重要となる。動特性が未知な対象物に目標通りの力を加えるには適応型ニューラルネットワーク制御のような環境に対する適応力のある制御が必要である [21]。しかし適応型のニューラルネットワークでも瞬時に適応できるわけではなく、環境に適応までの間、目標力に対しオーバーシュートしたりするなど予定外の力が対象物に加わる恐れがある。そこで本章ではファジィ制

御の方法を用いて適当な力制御器を設計し、それをニューラルネットワークでモデル化し、バックプロパゲーション法で学習するFNN力制御器 [79] を利用する。

5. 5. 1 ファジィ・ニューラルネットワーク力制御

通常ロボットマニピュレータのファジィ制御では、入力情報として誤差量とその変化量が考えられるが、力制御では通常測定値にノイズが多く含まれているため誤差の変化量を利用することが難しい。そこで誤差の変化量の代わりにマニピュレータが環境を押す速度を利用することが考えられるが、マニピュレータが環境を押す速度が早いか遅いかの概念はマニピュレータの形態により異なる。つまりその形態での力制御方向の慣性行列の大きさにより速度の概念が変わることになる。そこで本研究ではマニピュレータの速度ではなく力制御方向のマニピュレータ運動量を入力情報として利用することを第3章で提案している。運動量よりも運動エネルギーを利用するほうが慣性行列の大きさの差による影響を無くすという面では効果的であるが、速度が2乗で影響するため経験によるファジィ制御器の設計が非常に厳密になるので、ここでは運動量を利用している。対象物制御の際の力制御は通常力制御と違い環境自身が運動する。そこで運動量算出に使う速度は力制御方向のマニピュレータと対象物との相対速度を使用する。

$$M_n = M_s(q)v \dots\dots\dots (5.12)$$

ここで、 M_n はマニピュレータ運動量、 v はマニピュレータと対象物との相対速度、 $M_s(q)$ はマニピュレータの関節角度が q のときの基準座標系での慣性行列であり、マニピュレータの関節座標系での慣性行列を $M(q)$ としマニピュレータの手先位置の関節角度に関するヤコビ行列を J とすると次式のように表される。

$$M_s(q) = J^{-1}(q)M(q)J^{-1}(q) \dots\dots\dots (5.13)$$

制御器からの出力は基準座標系でのマニピュレータへの力指令とする。従って、図5.1のモデルでx方向とy方向の2方向の力制御器が必要である。

マニピュレータ力制御用FNNの学習則は、次に示す評価関数以外は前項の対象物制御用FNNの学習則と同じである。

$$Y = \frac{1}{2}(f_d - f)^2 \dots\dots\dots (5.14)$$

5. 5. 2 標準環境モデルによる補償

本制御器において対象物の動特性を設計時に硬めに想定しておけば、ほとんどの場合オーバーシュートすることもなく目標通りの力を制御できる。しかし、もし対象物の動特性が想定していたものよりかなり硬かった場合は大きなオーバーシュートが発生する等の問題が生じる。そこで、第3章ではFNNの入力値調整ニューラルネットワーク(IANN)を提案し、第4章ではより柔軟性のあるファジィ環境評価器を提案した。本章では、それらを大幅に簡略化するため、制御器内に設計時に想定した環境モデルを置き、環境が想定したものよりも硬かった場合に誤差のメンバーシップ関数を調整する方法を提案する。

環境を図5.5に示すようにモデル化すると、環境からの反力つまりマニピュレータにより加えられた力 F は、環境表面の変位を x_e とすると次式のように表される。

$$F = M_e \ddot{x}_e + D_e \dot{x}_e + K_e x_e \dots\dots\dots (5.15)$$

ファジィ制御則の設計時に標準となる M_e , D_e , K_e の各係数を想定し、制御器内で式(5.15)を各サンプリング毎に計算することにより設計時に想定した環境での環境反力がわかる。したがって、算出された環境反力と実際に測定した反力を比較することにより、想定した環境よりも硬いかどうか判断できる。もし想定した環境よりも硬かった場合は、オーバーシュートなどの問題を抑制するため、環境の硬さに応じ制御器への入力情報である目標値との誤差値に補正係数をかけてやることにより補償を行う。補正係数 H の算出式は次の通りである。

$$H = \frac{1}{(kF_m / F_{est})} \dots\dots\dots (5.16)$$

ここで、 k は補正のための経験値($k \geq 1$)、 F_m は実際に測定された力、 F_{est} は標準環境モデルで算出した力である。この補正係数 H をかけてやることにより、設計時に定義されたメンバーシップ関数を環境の硬さに応じ横に広げてやったような形状に変更すると同じ効果が得られる。この補償によるメンバーシップ関数の変化の様子を図5.6に示した。この効果により実際にはまだ誤差が大きい時点で誤差が小さいときのようにきめ細かく制御されるようになる。

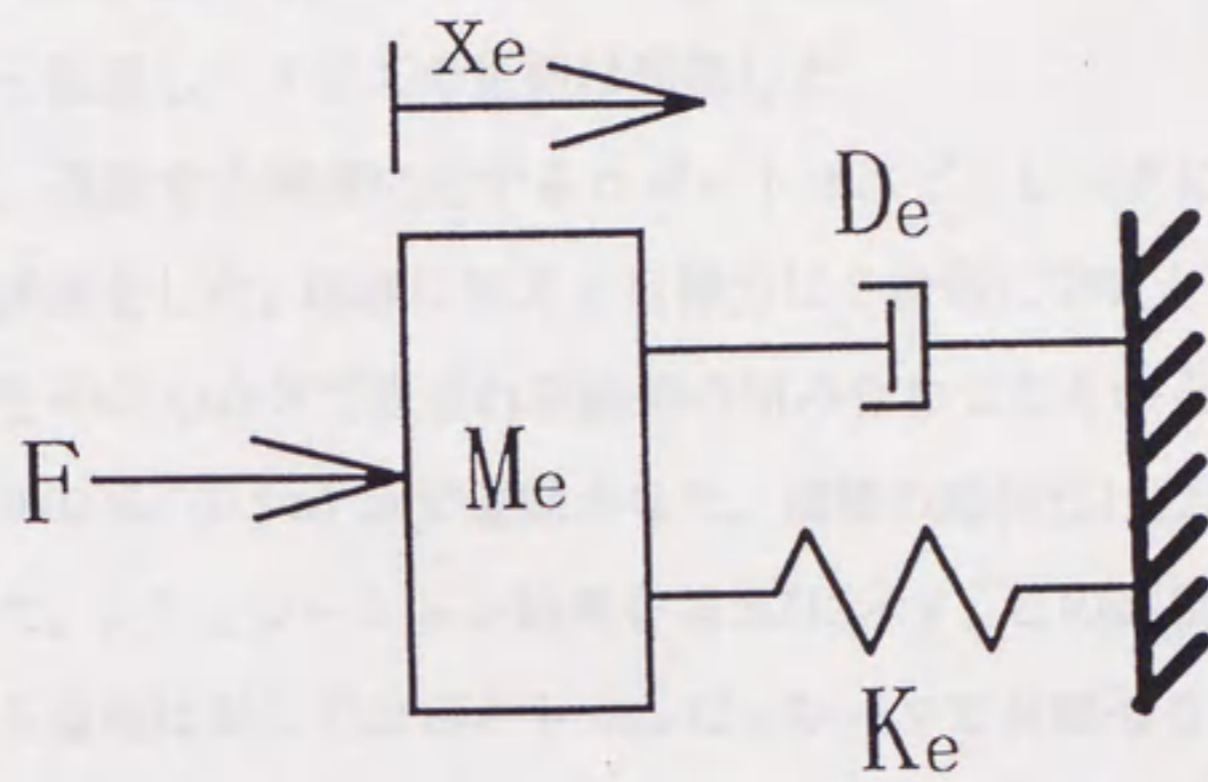


図 5.5 環境モデル

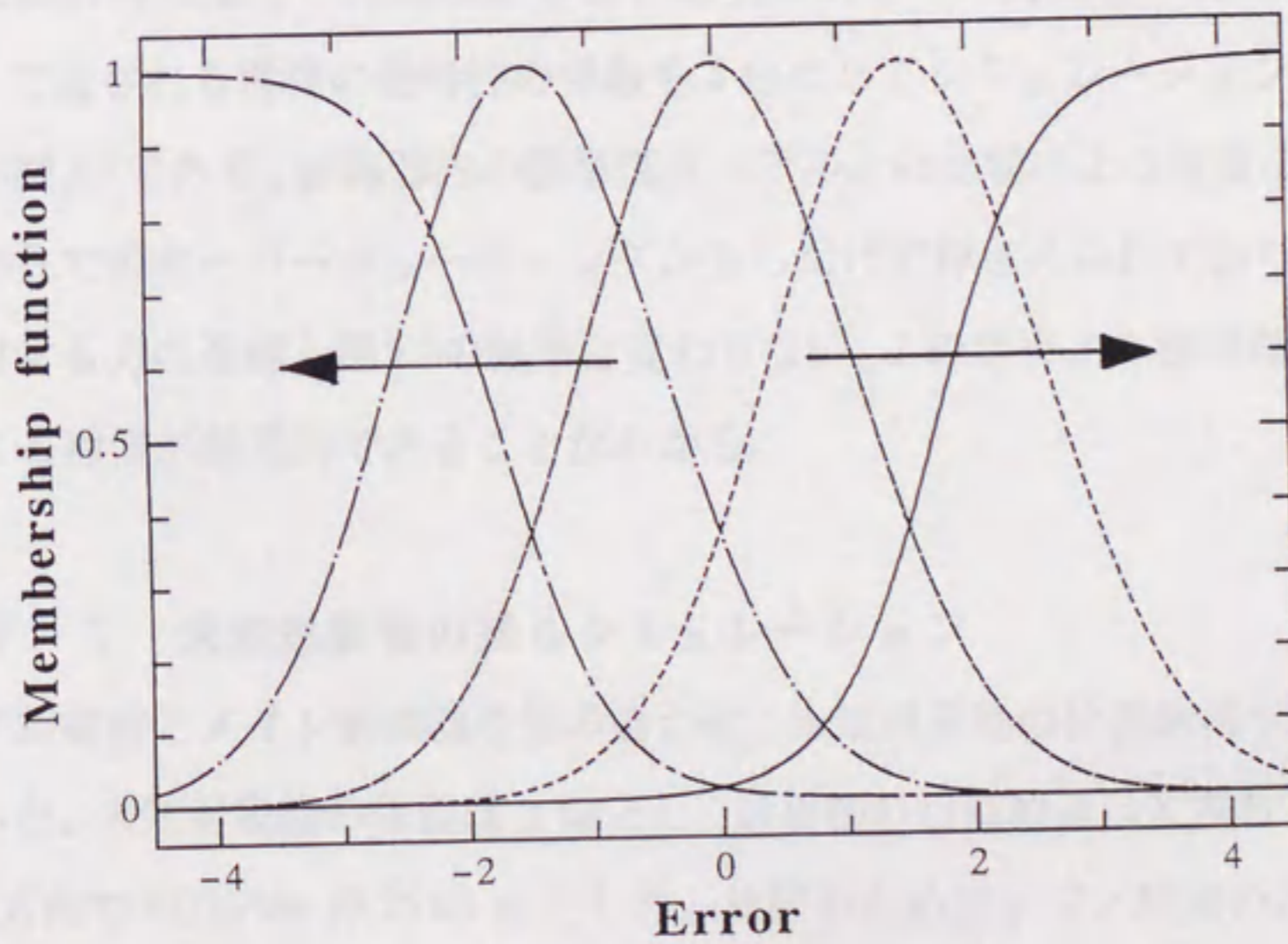


図 5.6 メンバシップ関数の変化

5. 6 シミュレーション

本FNN制御器を評価するため、図5.1に示したモデルを使いコンピューターシミュレーションを行った。シミュレーションでのサンプリング時間はすべて1msとした。また、量子化誤差と演算時間遅れは無視できる程度、センサーおよびモータの動作遅れはないものと仮定し、ノイズの影響は無視した。

まず最初に、運動する環境に対するロボットマニピュレータによる力制御を行い、サブ制御器の評価をした。環境に加える目標力は2秒毎に20Nと10Nの間で変化するステップ信号と $4 \sin(\pi t)$ Nで表される曲線の組み合わせたものとした。また、環境は力制御方向に $0.02 \sin(0.5\pi t)$ mで運動させた。環境の動特性は設計時の標準モデルと同じものにした。シミュレーション結果を図5.7に示す。この結果より、本サブ制御器により運動する環境に対してロボットマニピュレータで目標通りの力を加えられることが確認できる。比較のため、本FNN制御器の学習係数を0にして、すなわちファジィ制御のみで同じシミュレーションを行った結果が図5.8である。図5.7の結果と比較すると目標力から少し外れており、ニューラルネットワークによる補償無しでは目標通りの力が完全には加えられないことがわかる。

次に設計時に想定した環境より硬い環境に対しての力制御の結果を見るため、式(5.15)で表される環境の動特性の係数を4倍にしてシミュレーションを行った。その結果が図5.9である。制御器内の標準環境モデルとの比較による補償により、最初の立ち上がりでのオーバーシュートが少しだけで押さえられており、その後の目標力に対する力の追従も図5.7の結果と変わらない。この結果より標準環境モデルとの比較による補償が効果的であることがわかる。

5. 6. 1 未知対象物の操りシミュレーション

サブ制御器とメイン制御器を組み合わせ、未知対象物の位置制御シミュレーションを行った。未知対象物の質量は8kgとし、対象物の目標軌道はX方向で $0.05 \sin(0.25\pi t)$ m、Y方向で $0.025 \sin(0.25\pi t)$ mとした。比較のためファジィ制御のみ(学習係数0)の場合とFNN制御の場合についてシミュレーションを行った。この時のX方向の結果を図5.10に、Y方向の結果を図5.11に示した。これらの結果よりファジィ制御のみ

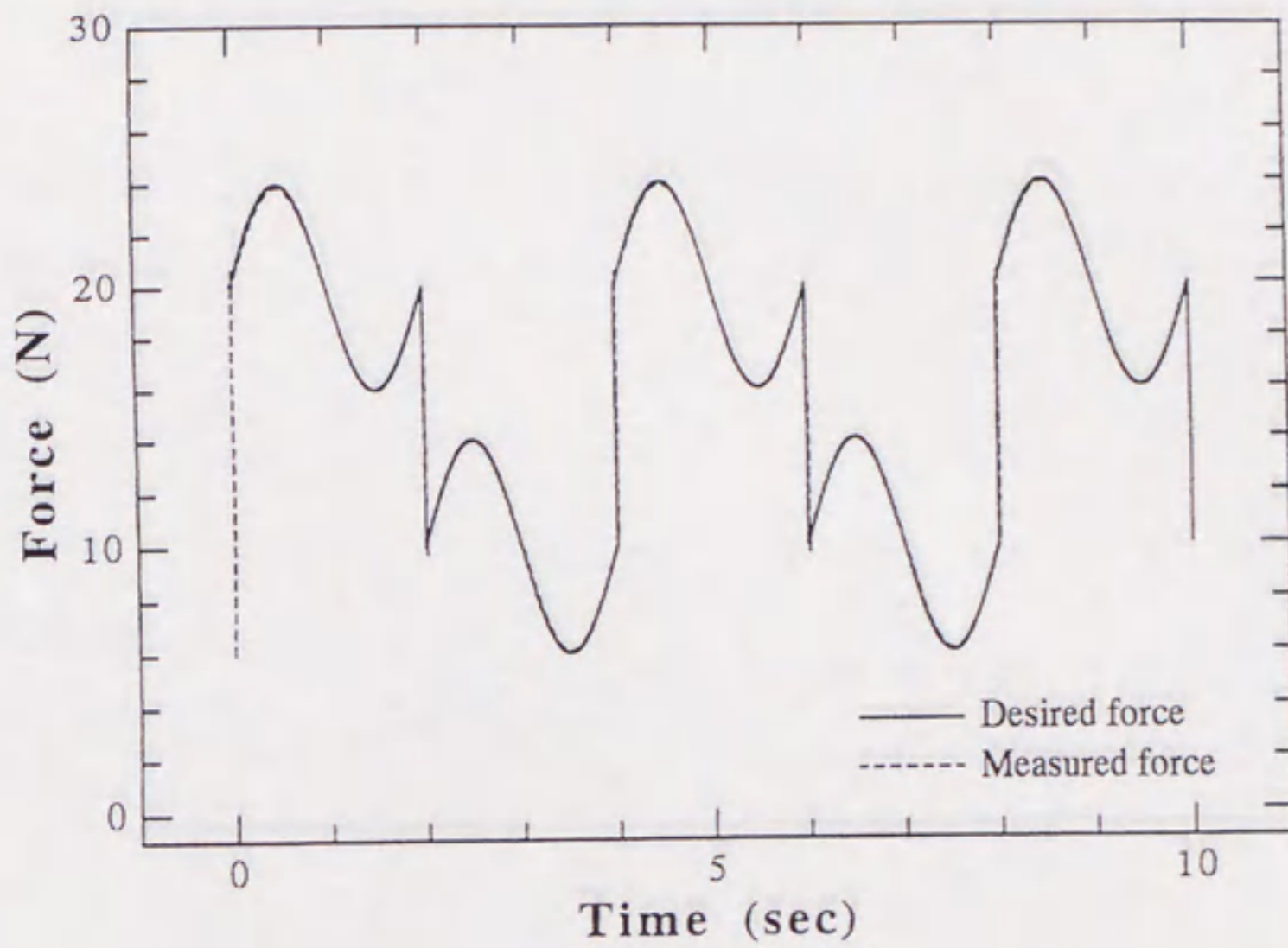


図 5.7 力制御シミュレーション結果 1

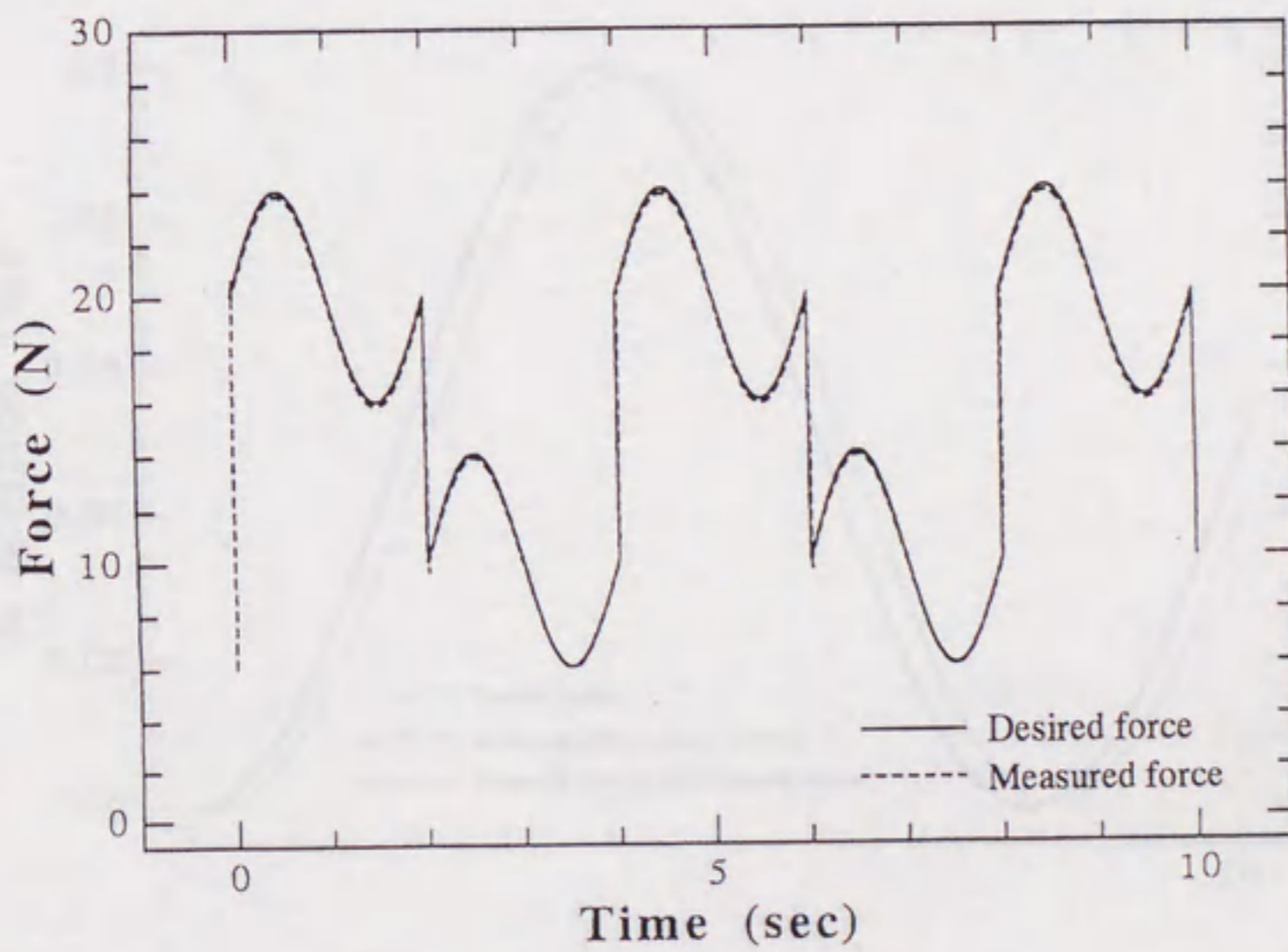


図 5.8 力制御シミュレーション結果 2

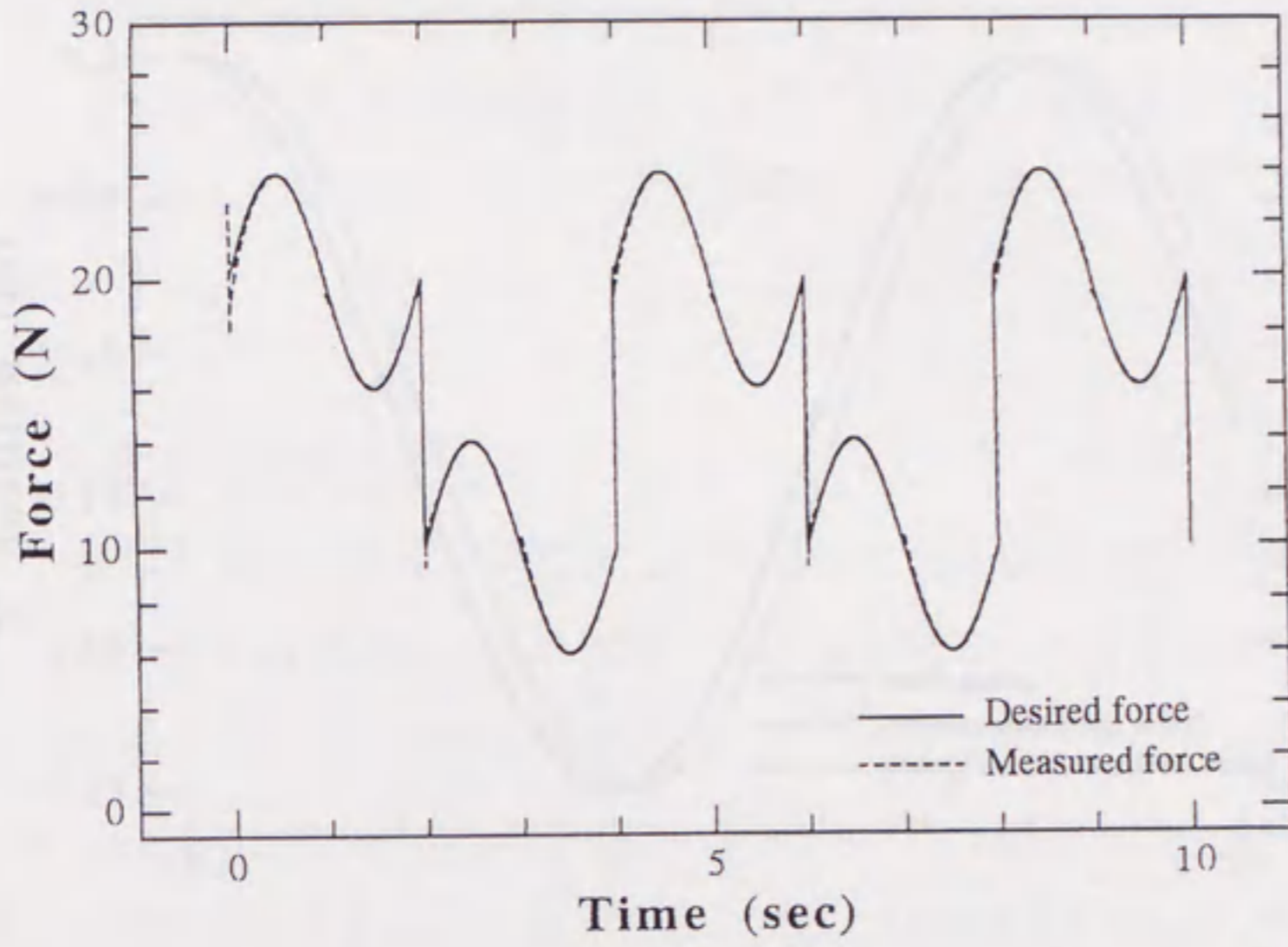


図 5.9 力制御シミュレーション結果 3

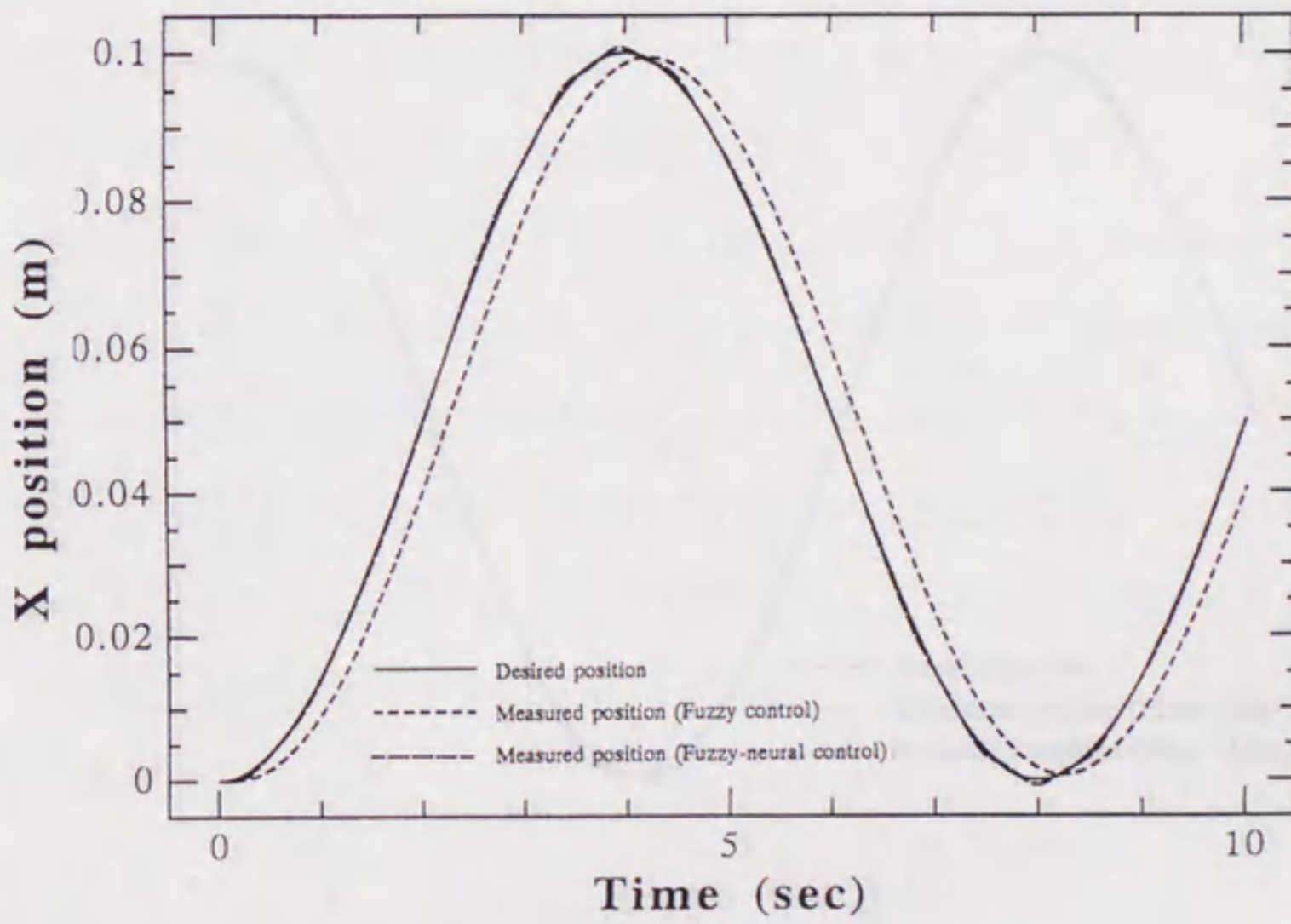


図 5.10 対象物位置制御シミュレーション結果 1

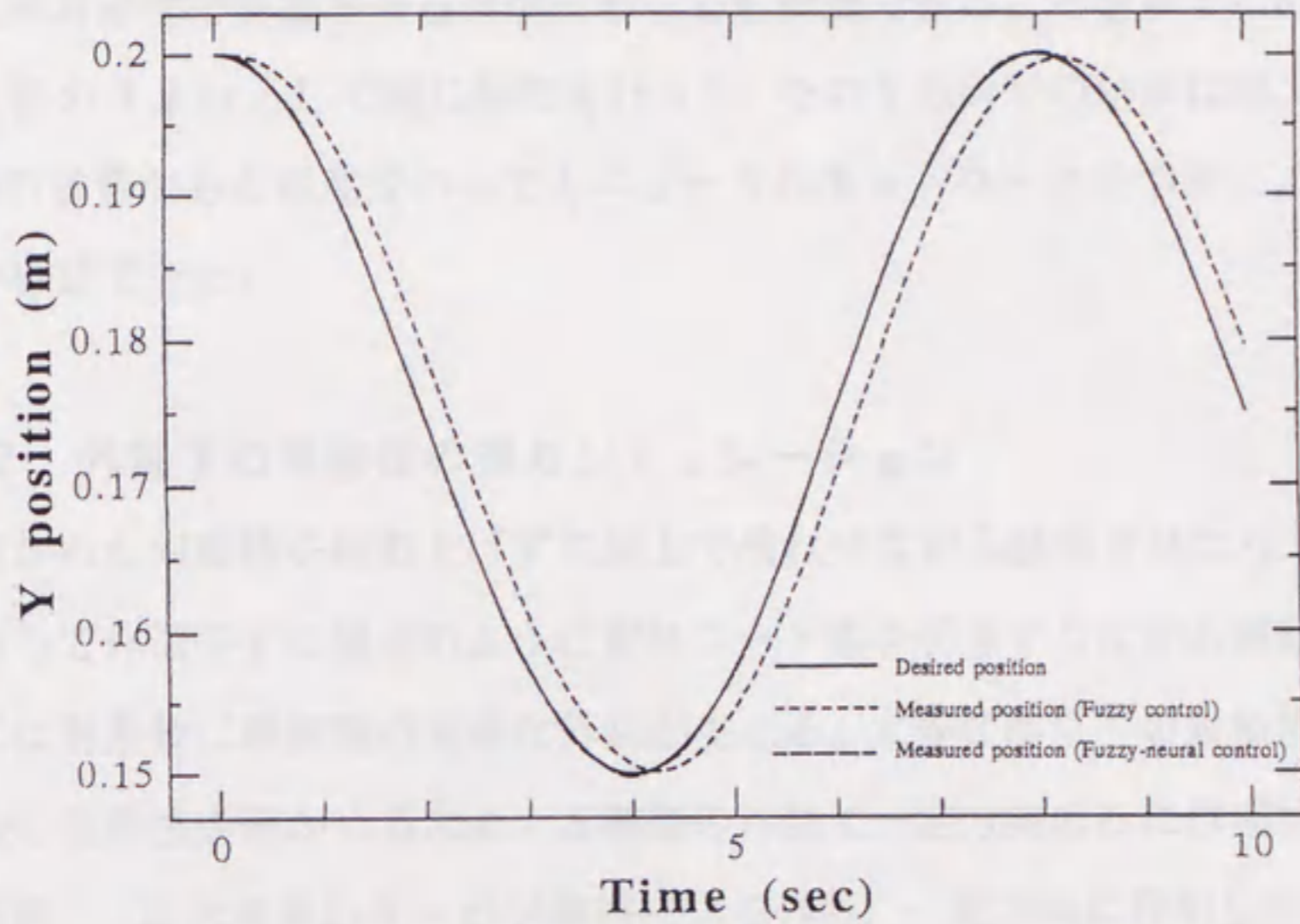


図 5.11 対象物位置制御シミュレーション結果 2

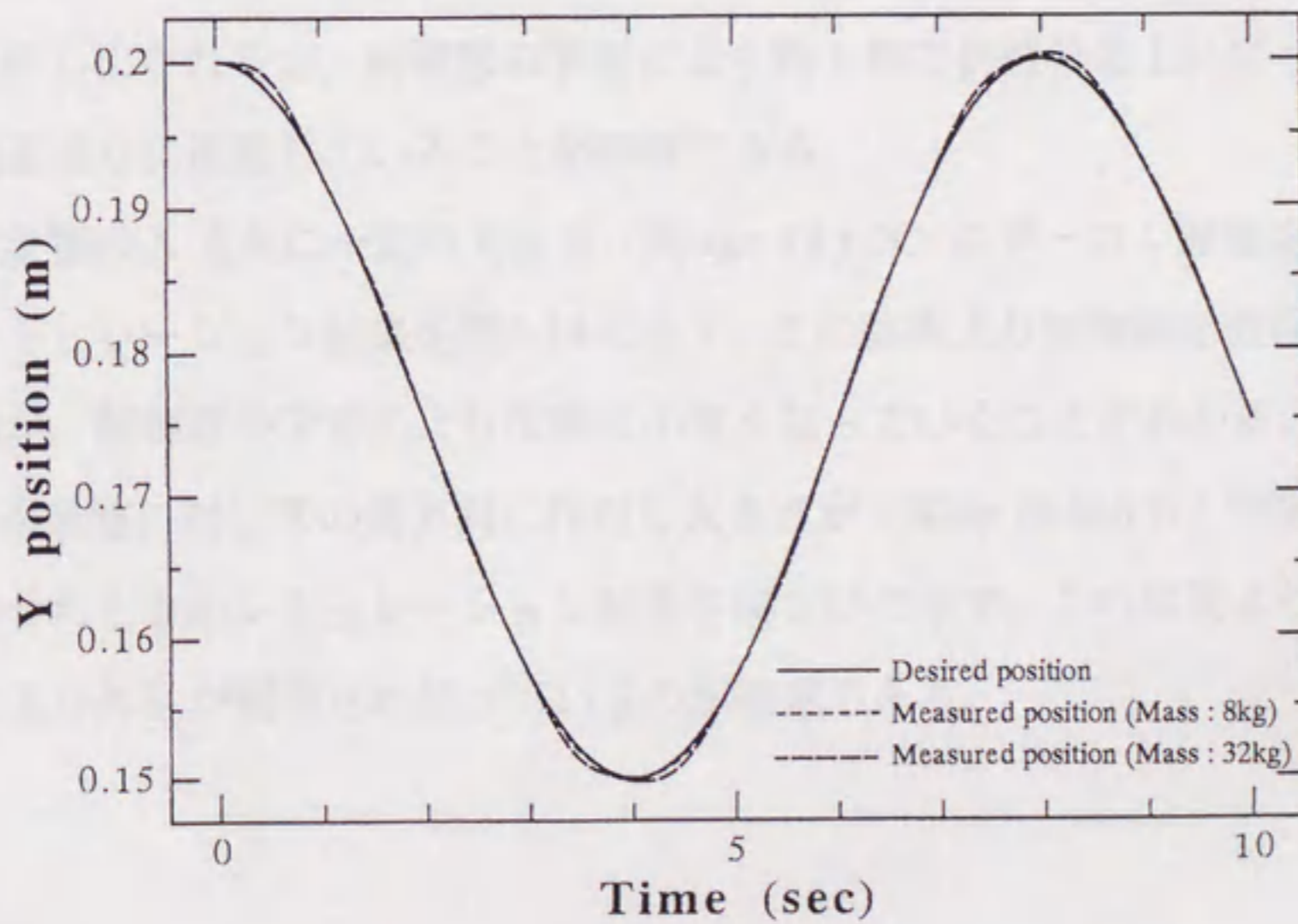


図 5.12 対象物位置制御シミュレーション結果 3

では目標軌道から少し外れるがFNN制御ではほぼ完璧に目標軌道に追従できることがわかる。

また、未知対象物の質量がある程度変わっても制御できることを示すため、対象物の質量を4倍の32 kgとして同じ制御を行った。そのY方向での結果は図5.12で示され、対象物の重量がある程度変わってもニューラルネットワークの学習により補償されることが確認できた。

5. 6. 2 外乱下の対象物の操りシミュレーション

床上に置かれた対象物を持ち上げずに床上で滑れせながら移動させたり、電動工具などを利用して作業をする場合のように電気コード等を引きずりながら移動させたりする場合には対象物に摩擦等の大きな外乱が生じる。大きな外乱下の対象物を操る際の本制御法の有効性を確かめるため、3種類の外乱（一定方向のみに作用する一定の大きさの外乱、一定大きさのクーロン摩擦による外乱、一定方向に作用し大きさがsin曲線で変化する外乱）を作用させ、対象物を操るシミュレーションを行った。目標軌道は5. 7. 1節でのシミュレーションと同じに設定した。

まず最初に、対象物に対しXの負方向に作用する一定の大きさ（20N）の外乱を作用させたときのシミュレーション結果を図5.13に示す。制御開始直後に外乱により目標軌道から押し出されるが、制御器の学習により約1秒で目標軌道上に戻され、その後は目標軌道通りに運動していることが確認できる。

次に、対象物のX方向に一定の大きさ（ $20 \operatorname{sgn}(\dot{x})$ N）のクーロン摩擦を作用させたときのシミュレーション結果を図5.14に示す。この結果より制御開始直後には大きかった誤差が、制御器の学習により次第に小さくなっていくことがわかる。

最後に、対象物に対しXの負方向に作用し大きさが（ $20 \sin(0.5\pi t)$ N）で変化する外乱を作用させたときのシミュレーション結果を図5.15に示す。この結果より、提案した制御器により外乱が補償され続けているのが確認できる。

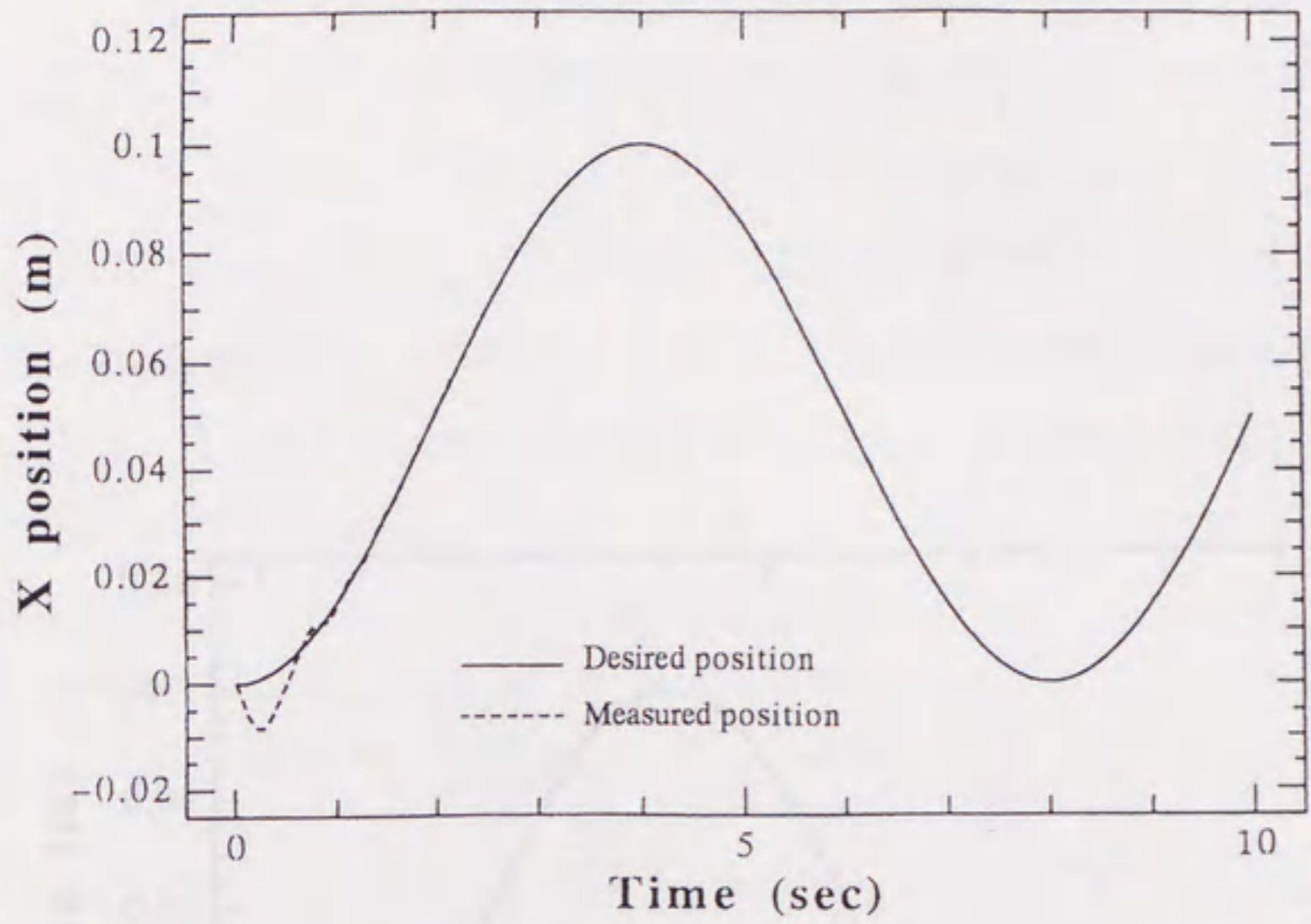


図 5.13 外乱下の対象物位置制御シミュレーション結果 1

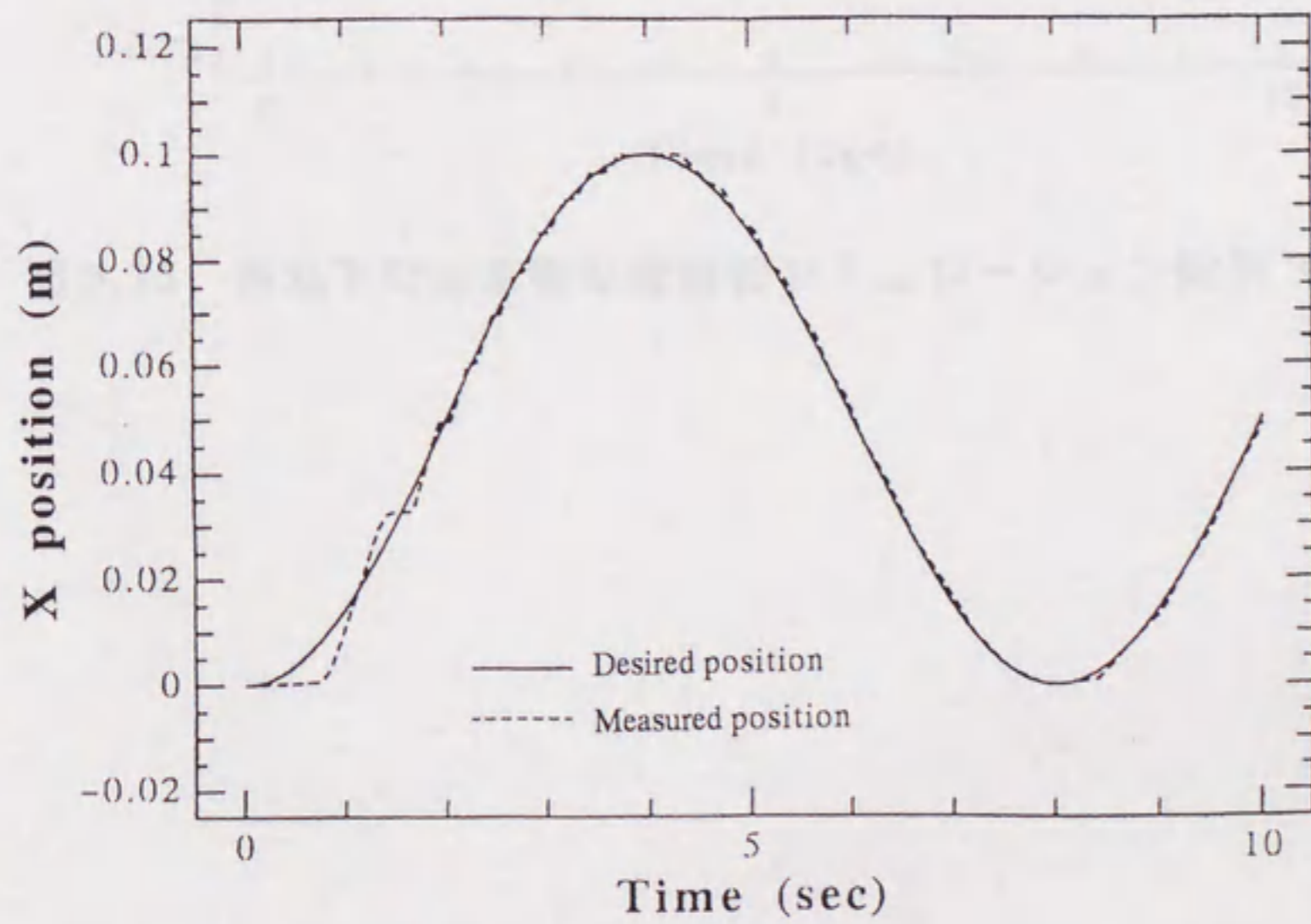


図 5.14 外乱下の対象物位置制御シミュレーション結果 2

5.3 結果

実験では、システムに外乱を加えて、その影響を軽減して目標位置に制御する方法を検討した。結果として、目標位置からの変位が小さくなることを確認した。また、制御系に外乱を加えて、その影響を軽減して目標位置に制御する方法を検討した。結果として、目標位置からの変位が小さくなることを確認した。

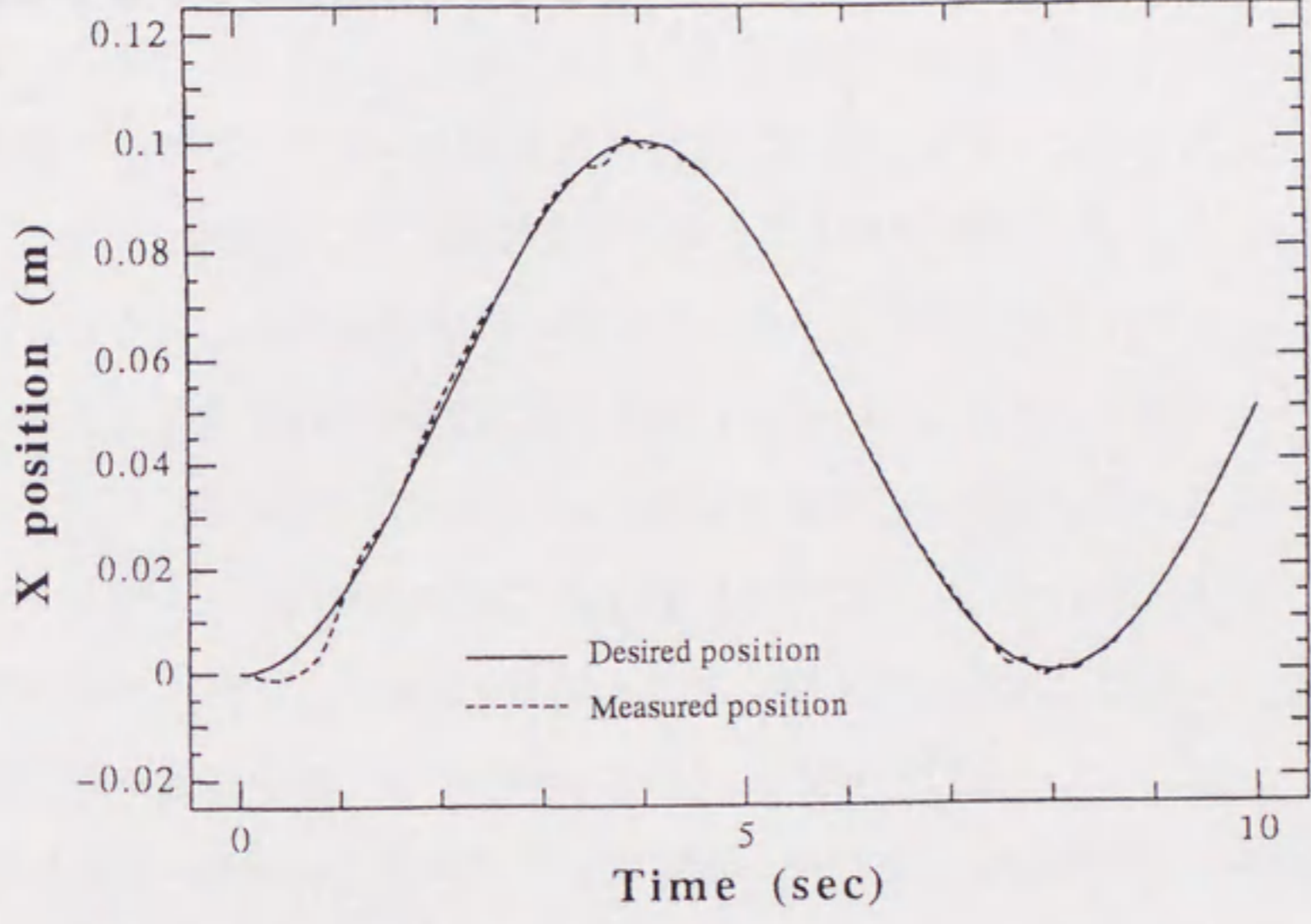


図 5.15 外乱下の対象物位置制御シミュレーション結果 3

5. 7 結 言

本章では、ファジィ・ニューラルネットワーク (FNN) を利用して未知対象物を制御する方法を提案した。重量や動特性が未知な対象物でも、また未知な外乱が作用する対象物でもファジィ・ニューラルネットワーク制御器を用いることにより目標通り制御できることを示した。また、制御器内に標準環境モデルを置き必要に応じ補償を行う簡単な方法を提案した。提案した制御器により質量や動特性等が不明な対象物を制御できることや、様々な外乱が対象物に作用していても制御器の学習により有効な外乱補償ができることを示し、コンピューターシミュレーションによりその有効性を確認した。

第6章 ファジィ・ニューロを利用した複腕ロボットによる摩擦補償を考慮した対象物の滑らせ移動制御

6.1 緒言

前章で述べた通り、ロボットマニピュレータの高度な作業内容の一つとして複腕での対象物の操りが挙げられる。これまでに前章で紹介したファジィ・ニューラルネットワーク(FNN)を利用した未知対象物の操り法の研究 [74], [76] も含め複腕ロボットマニピュレータによる対象物の操りについて多くの研究がなされており、いろいろな制御法や操り力の決定法が提案されてきた [1] ~ [6], [74], [76]。

また、FNNは、曖昧な情報の扱いができ厳密な数学モデルを使うことなく制御できるファジィ制御と、経験から学習できる能力を持つニューラルネットワークの両方の長所を合わせたもので、本論文で紹介してきた研究例を含め、ロボット制御の分野でも積極的に取り入れられている [81], [82], [96], [97]。

複腕ロボットマニピュレータで何らかの環境に接している対象物をその環境に接したまま動かす場合(たとえば対象物を床上で引きずる場合)、対象物にクーロン摩擦が作用する。本研究では、この様に対象物を環境に接したまま動かす制御を“滑らせ移動制御”と呼ぶことにする。第5章では、FNNを用いることにより制御器がクーロン摩擦等の外乱の補償を学習していくため、徐々に対象物を目標軌道通り制御ができるようになることを示した [76] が、制御器が学習するまで少し時間がかかり、その間の対象物の動きも滑らかなものではなかった。そこで本章では、第3章で提案した位置制御器を応用し、力が与えられているにも関わらず対象物が動かない場合だけ学習する摩擦補償用ニューロンと、対象物が運動しているときだけ学習する対象物操り用FNNを組み合わせた対象物操り制御器を提案する [77]。この“状況に応じて学習を切り替える制御器”により、対象物にクーロン摩擦が作用していても摩擦補償用

ニューロンで即座に摩擦を補償し、その後は対象物操り用FNN制御器により対象物の運動を滑らかに制御することが可能となる。

ニューラルネットワークのバックプロパゲーション法において学習係数は、学習速度や安定性に関わる重要なファクターであり、これまでに適切な学習係数を見つけるために多くの研究がなされてきた。たとえば、Jacobsは四つのヒューリスティックな学習係数調整法を挙げたうえで、ニューラルネットワークの全てのウエイトがそれぞれ別個の学習係数を持つことを主張しており [66]、Arabshahiらは学習係数をファジィ制御で調整することを提案している [67]。学習係数をファジィ制御で調整する方法は効果的であるが、学習係数の有効範囲を予め把握しておく必要があると思われる。従って、適切な学習係数の範囲が予測できない場合は、Jacobsが提案しているような、学習係数を線形的に増加させたり指数的に減少させたりするDelta-Bar-Delta学習係数調整法が効果的であると思われる。しかしDelta-Bar-Delta学習係数調整法では学習係数が最適値より小さい場合、学習係数は常に一定の割合で増加するが、本研究では学習状況により増加の割合も変化させたほうがより効果的であると考え、学習状況に応じ学習係数が小さい場合の増加割合を変化させる新しいDelta-Bar-Delta学習係数調整法を提案し、対象物操り用のFNNと摩擦補償用ニューロンの学習に利用する [77]。

これらの制御方法の効果は、コンピュータによるシミュレーションにより確認した。

6. 2 対象物の軌道制御

本章においても問題を簡単にするため、対象物の平面の並進運動のみを扱うものとする。第5章で用いたものと同じ2台の平面2自由度ロボットマニピュレータを用いて対象物の滑らせ移動制御を行うものとする。本章では、第5章図5.1に示すような平面2自由度ロボットマニピュレータで対象物を机上平面で滑らせながら操るようなモデルを考え、対象物は机で支えられるものとして対象物の重力補償は行わなかった。また、マニピュレータは対象物と点接触し、任意の力を加えられるものとする。したがって、対象物を操るのに必要な力を求める運動方程式は前章と同じであるので、ここでは省略する。

対象物操りのための目標力（マニピュレータで発生させる合力から内力のための力を除いたもの）は、各マニピュレータで平等に分担するものとする。ただし、各マニ

ピュレータの中で目標操り力を加えることができないものがある場合、つまり対象物操りのための目標力がそのマニピュレータにより対象物を引くかたちになる場合のものを除き、残りのマニピュレータで平等に分担するものとする。また、対象物とマニピュレータとの接触は点接触とし、適当な把持力を加えている限り滑ったり離れたりしないものとする。本研究では適当な把持力が既知であると仮定し、常に一定の把持力が作用しているものとした。

滑らせ移動制御の際に、対象物に作用する摩擦力が補償されていない場合も、制御開始後暫くすればバックプロパゲーション法による制御器自身の学習により対象物を目標通り制御できるようになる [76]。しかし、5. 6. 2節で見た通り、制御器が学習するまで少し時間がかかり、その間の対象物の動きも滑らかなものではなかった。そこで本章では、第3章でロボットマニピュレータの位置/力制御の際の位置制御に提案した方法を対象物操り制御に応用し、摩擦補償用ニューロンで摩擦力の補償を行い、対象物操り用FNNで対象物の滑らせ移動制御を行う対象物操り制御器を提案する。本制御器により、より早く対象物の運動を滑らかに制御することが可能となる。

6. 3 制御器の構成

ここでは対象物を滑らせ移動制御するために必要な力指令を出力するメインの制御器 (Object controller) と、各マニピュレータを力発生器と考え、動特性の未知な環境に対しメイン制御器からの力指令通りの力を各腕から出力させるための二つのサブの制御器 (Arm controllers) の複数の制御器から構成される制御器を利用する。制御器全体のブロック線図を図6.1に示す。メインとサブの制御器から成る構成自体は第5章の制御器 [74], [76] と同じであるので詳しくは前章を参照されたい。メイン制御器への入力対象物の目標位置とフィードバックされた実際の位置情報であり、メイン制御器からの出力つまり各マニピュレータで発生させる目標力と力センサーからフィードバックされた実際の力情報がサブ制御器への入力である。この各サブ制御器からの出力が各マニピュレータへの入力となる。

本章で提案するメイン制御器は、ロボットマニピュレータにより力が与えられているにも関わらず対象物が動かない場合だけ学習する摩擦補償用ニューロンと、対象物が運動しているときだけ学習する対象物操り用FNNとを組み合わせた制御器であり、

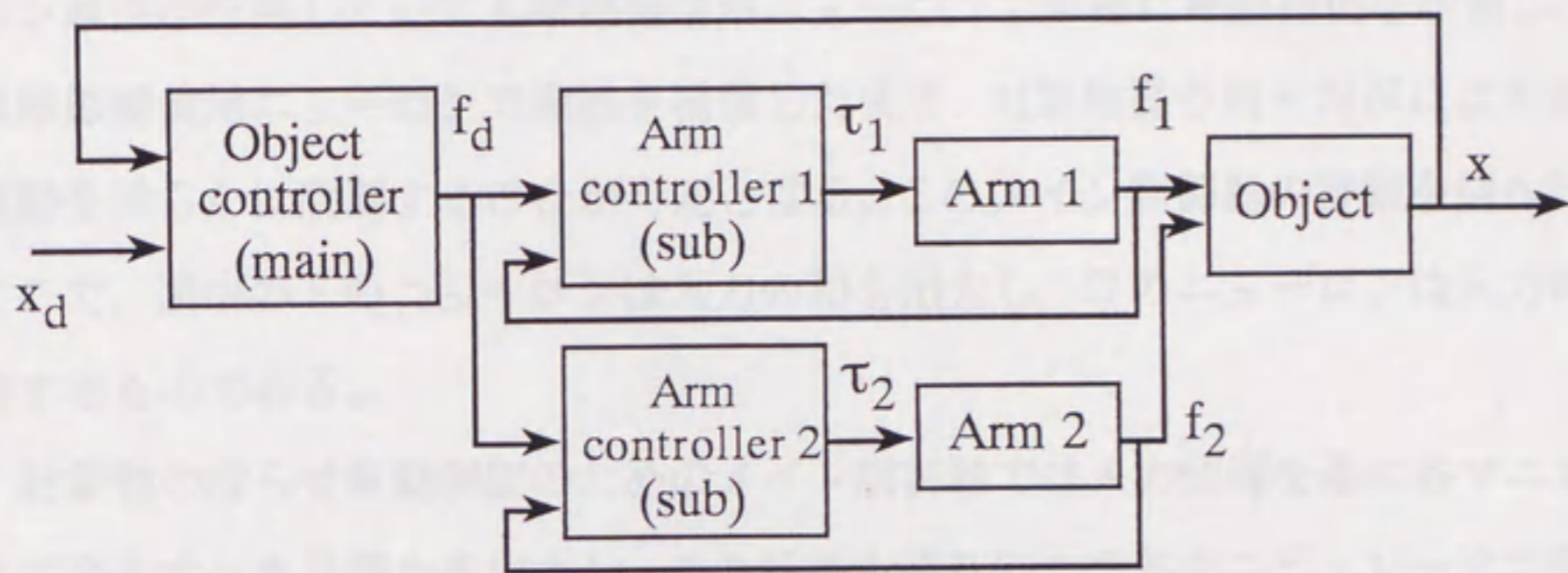


図 6.1 システムのブロック線図

対象物の運動状況により制御器内の学習する部分を切り替えるというところにも特徴がある。この学習の切り替えは対象物の位置情報をもとにして、力が与えられているにも関わらず対象物が動かない場合は対象物操り用FNNの学習係数をゼロに、対象物が運動しているときは摩擦補償用ニューロンの学習係数をゼロにすることにより行われる。メイン制御器からの出力は、対象物操り用FNNからの出力と摩擦補償用ニューロンからの出力を合わせたものである。この組み合わせにより、対象物にクーロン摩擦が作用していても摩擦補償用ニューロンで即座に摩擦補償を学習し、その後は摩擦補償用ニューロンで摩擦を補償したまま、対象物操り用FNNにより対象物の運動を滑らかに制御することが可能となる。このメイン制御器の詳細を図6.2に示す。ここで、図中の Σ のニューロンは入力のを出力し、 Π のニューロンは入力の積を出力するものである。

対象物の滑らせ移動制御のためのメイン制御器では入力情報を基に各マニピュレータで発生すべき目標力を出力し、その目標力通りの力を各マニピュレータで発生するようサブ制御器である各マニピュレータ力制御器でフィードバック制御する。各マニピュレータから目標通り発生された力により対象物が運動し、その情報が対象物の制御器へフィードバックされる。

この章では主にメイン制御器について説明するものとする。サブ制御器については第5章を参照されたい。

6. 3. 1 摩擦補償用ニューロン

この章では対象物の滑らせ移動制御のためのメイン制御器内の摩擦補償用ニューロンについて説明する。第3章で提案した位置/力制御の際の位置制御の摩擦補償用ニューロンは環境に加えられた力と摩擦補償が必要な方向(1あるいは-1)の二つを入力としたのに対し、このニューロンは、摩擦補償が必要な方向(1あるいは-1)のみを入力とし、その入力にウエイト値を掛けたものを摩擦補償力として出力する。したがって、ある程度の学習した後のウエイト値はクーロン摩擦力とほぼ同じ大きさになっている。この出力と対象物操り用FNNからの出力とを合わせたものがメイン制御器からの出力となる。

この摩擦補償用ニューロンは、力が与えられているにも関わらず対象物が動かない

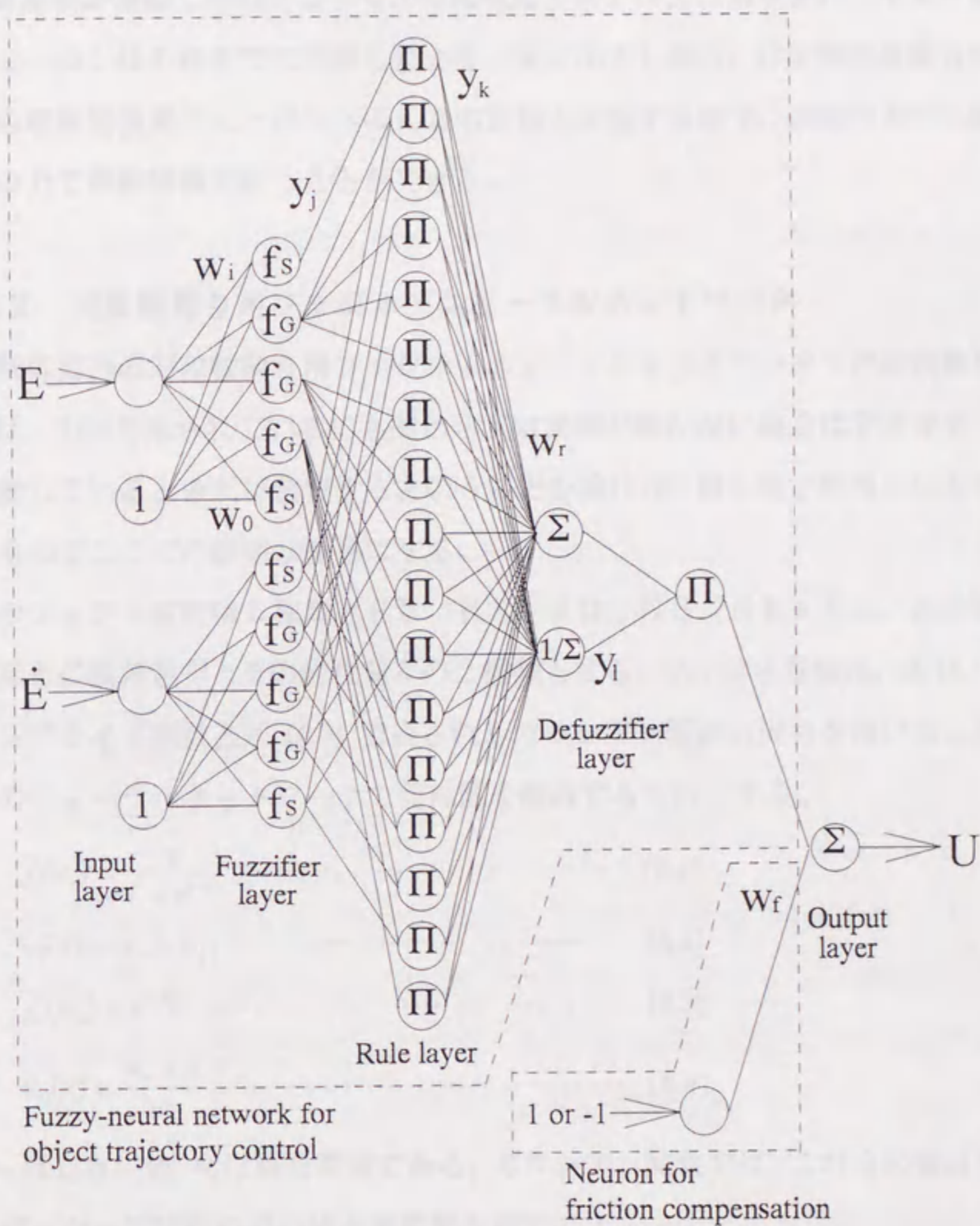


図 6.2 対象物操り用ファジィ・ニューラルネットワークの構成
(メイン制御器)

場合のみ学習するものとし、対象物が運動しているときは学習しないものとする。従って、対象物が運動し始めた後学習が対象物操り用FNNに切り替わっても、摩擦補償用ニューロンはそれまでに学習した力を一定に出力し続け、対象物の運動方向が反転したら摩擦補償用ニューロンへの入力の正負も反転するので、運動の方向に関わらず一定の力で摩擦補償を行うことができる。

6. 3. 2 対象物操り用ファジィ・ニューラルネットワーク

メイン制御器内の対象物操り用ファジィ・ニューラルネットワーク（対象物操り用FNN）は、力が与えられているにも関わらず対象物が動かない場合は学習せず、対象物が運動しているときだけ学習するというものを除けば、第5章で利用したものと同じであるのでここでの説明は簡単にする。

前件部のファジィ変数は5種類（PB, PS, ZO, NS, NB）とし、入力変数は目標位置との誤差量 E とその変化量 \dot{E} の2種類とする。ファジィ変数は、式(6.1)で表されるシグモイド関数と式(6.3)で表されるガウシアン関数の両方を用いることにより単層のニューラルネットワークで前件部を構成するものとする。

$$f_s(u_s) = \frac{1}{1 + e^{-u_s}} \dots\dots\dots (6.1)$$

$$u_s(x) = w_0 + w_i x \dots\dots\dots (6.2)$$

$$f_G(u_G) = e^{-u_G^2} \dots\dots\dots (6.3)$$

$$u_G(x) = \frac{w_0 + x}{w_i} \dots\dots\dots (6.4)$$

ここで、 w_0 はしきい値、 w_i は結合荷重である。ガウシアン関数では、これらの値はそれぞれメンバーシップ関数の中心値と偏差値を表す。

後件部はIF-THENの制御ルールに従い、前件部適合度の代数積に定数値をかけ合わせた各ルールからの出力和を前件部適合度の和で割って定数値出力を得るものとする。前述の通り、この対象物操り用FNNの出力と摩擦補償用ニューロンからの出力を合わせたものがメイン制御器からの出力、つまり各マニピュレータから発生させる目標力 U である。

6.3.3 学習則

バックプロパゲーション法を用いて対象物操り用FNNの前件部と後件部の両方のウェイト値の微調整,あるいは摩擦補償用ニューロンの学習を各サンプリング毎に行うものとする。評価関数は次式で示される対象物位置の2乗誤差値とする。

$$Y = \frac{1}{2}(x_d - x)^2 \dots\dots\dots (6.5)$$

ここで, x_d は目標位置, x は実際の対象物の位置である。ただし, 対象物操り用ファジィ・ニューラルネットワークと摩擦補償用ニューロンは同時に学習することは無く, 対象物が動いている時は対象物操り用FNNが学習し, 力が与えられているにも関わらず動かない時は摩擦補償用ニューロンが学習する(図6.3参照)。すなわち, 対象物の位置情報をもとにして, 力が与えられているにも関わらず対象物が動かない場合は対象物操り用FNNの学習係数をゼロに, 対象物が運動しているときは摩擦補償用ニューロンの学習係数をゼロにすることにより学習の切り替えが行われる。対象物操り用FNNの前件部と後件部の各部でのウェイト値の変化量は次の通りである。

$$\text{後件部} : \Delta w_r = \eta(x_d - x)y_i y_k \dots\dots\dots (6.6)$$

$$\text{前件部} : \Delta w_i = \eta \sum \frac{\partial e_j}{\partial y_i} f'_{(w)} u' \dots\dots\dots (6.7)$$

ここで η は学習係数, x_d は目標位置, x は実際の対象物の位置, y_{ij} は前件部出力, e_j は前件部出力誤差, $f'_{(w)}$ と u' は式(6.1)(6.3)で表されるアクティベーション関数とその入力の微分である。また, 学習係数は, 前件部の各入力と後件部でそれぞれ別々の値に設定した。これにより, メンバシップ関数の定義や対象物操り用FNNからの出力が調整される。

摩擦補償用ニューロンの入力が1のときのウェイト値の変化量は次式で表される。

$$\Delta w_f = \eta(x_d - x) \dots\dots\dots (6.8)$$

入力が-1のときは式(6.8)の変化量に-1を掛けたものがウェイト値の変化量となる。

摩擦補償用ニューロンの学習は対象物が動いていない時だけ行われるので, 評価関数が減少するように学習するというよりは, 対象物が動き出すまでウェイト値を大きな値に変更するものである。

メイン制御器内の対象物操り用FNNと摩擦補償用ニューロンの全てのウェイト値

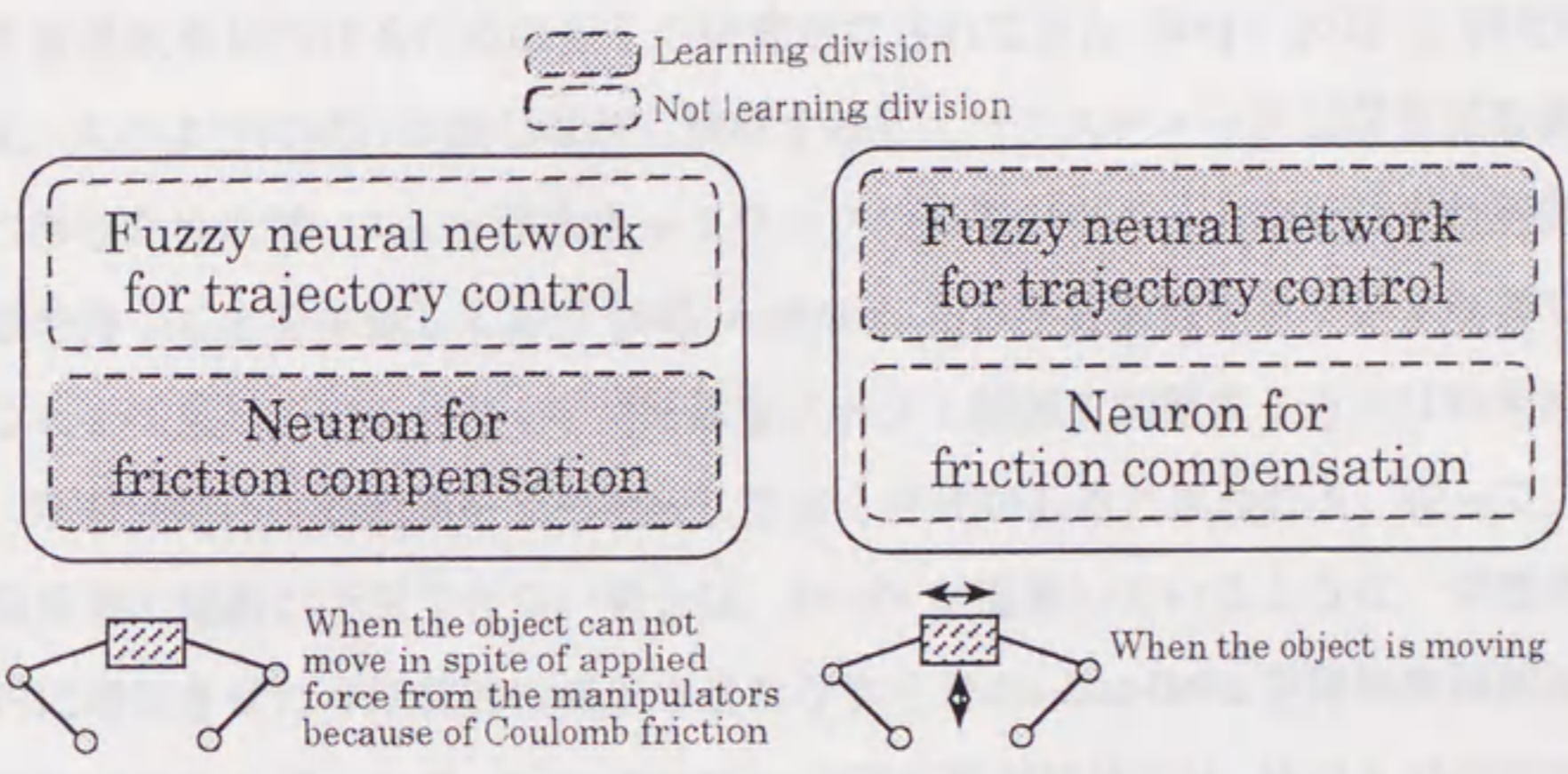


図 6.3 対象物操り用制御器（メイン制御器）の切り替え学習

は、それぞれ別個の学習係数により調整される。

6. 4 学習係数

ニューラルネットワークのバックプロパゲーション法において学習係数は、学習速度や安定性に関わる重要なファクターである。学習係数が固定されている一般的なバックプロパゲーション法では、誤差関数曲面の勾配が緩い場合学習速度が遅くなる。そこで、学習速度を上げるためには学習係数を大きくすれば良いのだが、学習係数が大きすぎると発散してしまう危険性がある。この問題を解決するため、これまでも適切な学習係数を見つけるための多くの研究がなされてきた [66], [67]。例えば、Jacobs は、人のように試行錯誤しながら調整するヒューリスティックな学習係数調整法を四つ挙げたうえで、ニューラルネットワークの全てのウェイトがそれぞれ別個の学習係数を持つことを主張しており [66], Arabshahi らは学習係数をファジィ制御で調整することを提案している [67]。学習係数をファジィ制御で調整する方法は効果的であるが、学習係数の有効範囲を予め把握しておく必要があると思われる。従って、適切な学習係数の範囲が予測できない場合は、Jacobs が提案しているような、学習係数を線形的に増加させたり指数的に減少させたりする Delta-Bar-Delta 学習係数調整法の方が効果的であると思われる。Delta-Bar-Delta 学習係数調整法では、時刻 t での学習係数の調整量 $\Delta\eta(t)$ は次に示すように定義される。

$$\Delta\eta(t) = \begin{cases} \kappa & \text{if } \bar{\delta}(t-1)\delta(t) > 0 \\ -\phi\eta(t) & \text{if } \bar{\delta}(t-1)\delta(t) < 0 \\ 0 & \text{otherwise} \end{cases} \dots\dots (6.9)$$

$$\delta(t) = \frac{\partial Y(t)}{\partial w(t)} \dots\dots\dots (6.10)$$

$$\bar{\delta}(t) = (1 - \theta)\delta(t) + \theta\bar{\delta}(t-1) \dots\dots\dots (6.11)$$

ここで、 κ と ϕ は定数、 $\delta(t)$ は時刻 t における評価関数 Y のウェイト w に対する編微分、 $\bar{\delta}(t)$ は θ をベースとした現在と過去の微分の指数的平均である。しかし、このような学習係数調整法では学習係数が適切な値より小さかった場合、学習係数は常に一定の割合で増加するため、学習係数とその増加割合が適切な学習係数に比べ小さい場合は効率が悪いという問題がある。

本章では、状況により学習を切り替える制御器に加え、学習係数が最適値より小さい場合の学習係数増加割合 κ を学習状況に応じて変化させる新しいDelta-Bar-Delta学習係数調整法を提案し、対象物操り用のFNNと摩擦補償用ニューロンの学習に利用する。新しいDelta-Bar-Delta学習係数調整法では、学習係数が最適値よりも小さいとき、一定の割合で増加させるのではなく、ウェイトの変化分に対し出力誤差の変化分が小さいニューロンの学習係数は大きな割合で、ウェイトの変化分に対し出力誤差の変化分が大きいニューロンの学習係数は小さな割合で増加させる。すなわち、誤差関数曲面の勾配が緩いものには大きな割合で、誤差関数曲面の勾配が急なものには小さな割合で学習係数を増加させるものとする。ただし、学習係数を減少させる場合は既存のDelta-Bar-Delta学習係数調整法と同様に指数的に減少させるものとする。したがって本稿では、新しいDelta-Bar-Delta学習法として、学習状況により式(6.9)での κ の大きさを下記の例のように調整する方法を提案し、メイン制御器の対象物操り用FNNと摩擦補償用ニューロンの両方の学習係数の調整に利用した。

例：

$$\kappa = \begin{cases} 10\kappa & \text{if } 0.2 > \delta(t) \\ \kappa & \text{if } 0.2 \leq \delta(t) \leq 1.0 \\ 0.1\kappa & \text{if } 1.0 < \delta(t) \end{cases}$$

また、 κ の大きさを連続で変化させる方法も考えられる。例えば、上の例で $\delta(t)$ が0.2より小さいときに、

$$\kappa = (20 - 95\delta(t)) \kappa \quad \text{if } 0.2 > \delta(t)$$

ただし、力が与えられているにも関わらず対象物が動いていない場合は対象物操り用FNNの学習係数がゼロ、対象物が運動しているときは摩擦補償用ニューロンの学習係数がゼロに設定される。

6.5 マニピュレータの制御

メイン制御器で出力された目標力通りの力を各サブ制御器によりマニピュレータで発生できなければ対象物は予定通り運動しない。したがって、目標力通りの力(メイン制御器からの出力)を各マニピュレータで発生できるということが重要となる。そこで本章では第5章と同じく、ファジィ制御の方法を用いて適当な力制御器を設計し、

それをニューラルネットワークでモデル化し、バックプロパゲーション法で学習するファジィ・ニューラルネットワーク力制御器 [81] [82] を利用する。ここでのマニピュレータの制御法については前章でも述べているので、ここでは簡単に説明するものとする。

6. 5. 1 ファジィ・ニューラルネットワーク力制御

通常ロボットマニピュレータのファジィ制御では、入力情報として誤差量とその変化量が考えられるが、力制御の場合は通常測定値にノイズが多く含まれているため誤差の変化量を利用することが難しい。そこで誤差の変化量の代わりにマニピュレータが環境を押す速度を利用することが考えられるが、マニピュレータが環境を押す速度が早いか遅いかの概念はマニピュレータの形態により異なる。つまりその形態での力制御方向の慣性行列の大きさにより速度の概念が変わることになる。そこで、ここではマニピュレータの速度ではなく力制御方向のマニピュレータ運動量を入力情報として利用するものとする [81] [82]。対象物軌道制御の際の力制御は通常力制御と違い環境自身が運動するので、運動量算出に使う速度は力制御方向のマニピュレータと対象物との相対速度を使用する。

サブ制御器からの出力は基準座標系でのマニピュレータへの力指令とする。したがって、図 5.1 のモデルで X 方向と Y 方向の 2 方向の力制御器が必要である。

サブ制御器のファジィ・ニューラルネットワークの学習則は、次に示す評価関数以外前項のメイン制御器のニューラルネットワークの学習則と同じである。

$$Y = \frac{1}{2}(f_d - f)^2 \dots\dots\dots (6.12)$$

ただし、サブ制御器の学習係数の調整は行わず、一定の定数を使用した。また、サブ制御器内部には、第 5 章で提案した標準環境モデルを置き、サブ制御器を設計した際に想定した環境よりも硬かった場合には制御器の補償を行うものとした (5. 5. 2 節参照)。

6. 6 シミュレーション

本制御器の有効性を評価するため、第 5 章の図 5.1 に示したモデルを用いコンピューターによるシミュレーションを行った。対象物の質量は 8 kg とし、X 方向と Y 方向へ

同時に目標軌道通りに並進運動するよう制御されるものとする。また、対象物にはX方向Y方向共に10Nのクーロン摩擦が作用するものとする。シミュレーションでのサンプリング時間はすべて1 msとした。また、量子化誤差と演算時間遅れは無視できる程度、センサーおよびモータの動作遅れはないものと仮定し、ノイズの影響は無視した。

第5章において、既にサブ制御器により目標通りの力が得られることを確認しているので、本章ではサブ制御器の目標力に対する追従性のシミュレーション結果は省略するものとする。

まず最初に、メイン制御器において、提案した新しい学習係数調整法の効果を確認するため、摩擦補償用ニューロンによるクーロン摩擦補償無しの場合について、従来のDelta-Bar-Delta学習係数調整法を用いた対象物滑らせ移動制御の結果と本章で提案した新しいDelta-Bar-Delta学習係数調整法を用いた対象物滑らせ移動制御の結果とで比較を行った。マニピュレータの可動範囲を考え、対象物の目標軌道はX方向で $0.05\sin(0.25\pi t)$ m、Y方向で $0.025\sin(0.25\pi t)$ mとした。また、対象物の動特性は設計に想定したモデルと同じものにした。このときのX方向のシミュレーション結果を図6.4に、Y方向のシミュレーション結果を図6.5に示す。これらの結果より、飛躍的な向上は見られないものの、新しい学習係数調整法を用いた方がより滑らかに誤差も少なく目標軌道に近づいていくことが確認できる。

次に、メイン制御器における摩擦補償用ニューロンによるクーロン摩擦補償効果を確認するための対象物滑らせ軌道制御シミュレーションを行った。このとき、対象物操り用ファジィ・ニューラルネットワークと摩擦補償用ニューロンのバックプロパゲーション法による学習の学習係数の調整は、学習係数がゼロに切り替えられている時以外、本章で提案した新しいDelta-Bar-Delta学習法により調整した。対象物の目標軌道は最初のシミュレーションのときと同じくX方向で $0.05\sin(0.25\pi t)$ m、Y方向で $0.025\sin(0.25\pi t)$ mとした。このときのX方向のシミュレーション結果は図6.6に、Y方向のシミュレーション結果は図6.7に示されている。ここでは、制御開始後0.3秒間位摩擦補償用ニューロンが学習し、その後は対象物が運動し始めるため、対象物操り用ファジィ・ニューラルネットワークが学習している。対象物操り用ファジィ・ニューラルネットワークが学習し始めるときには摩擦補償用ニューロンのよる摩擦補償の学

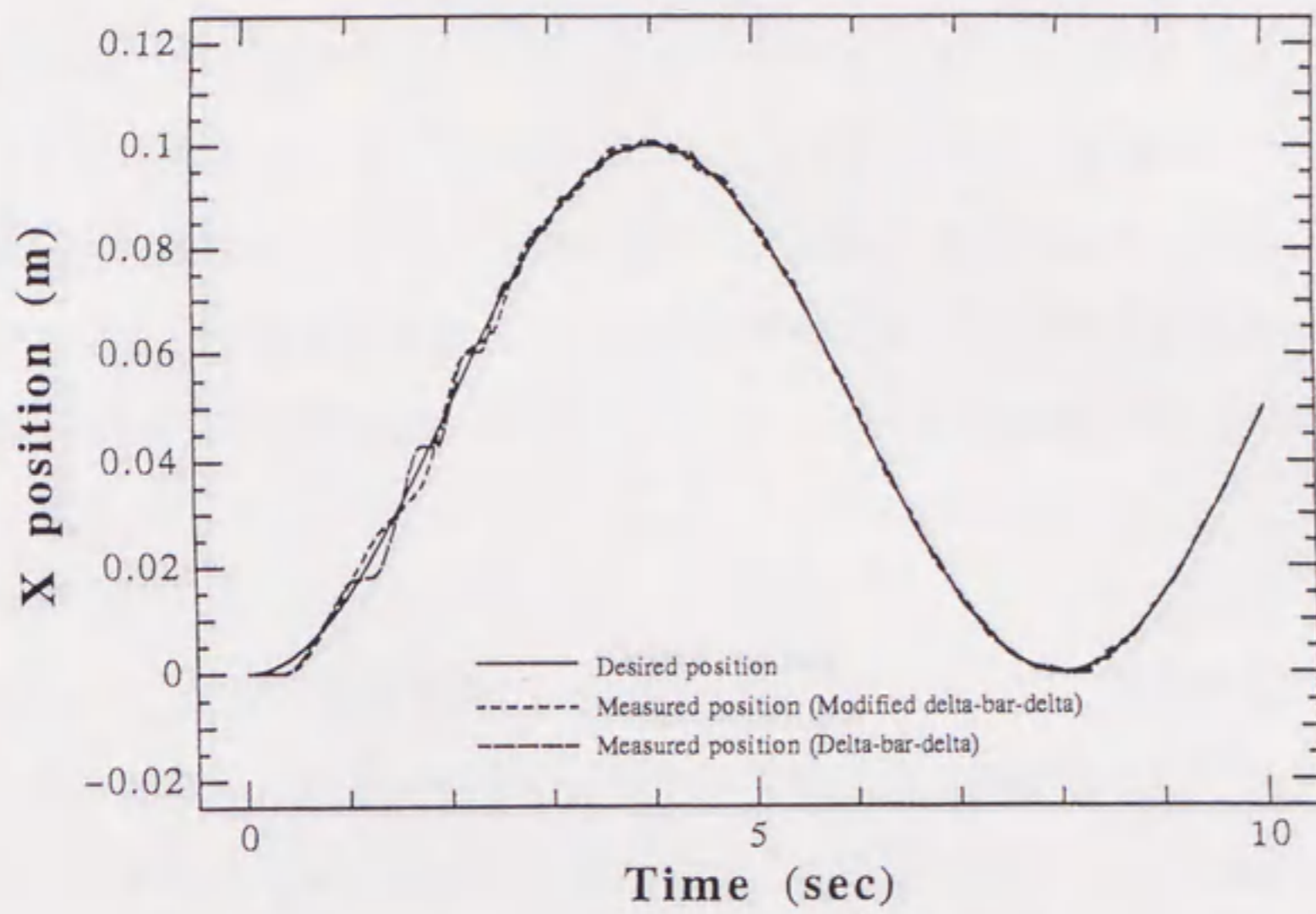


図 6.4 対象物の位置制御シミュレーション結果 (X 方向)
提案した Delta-Bar-Delta 学習係数調整法の評価

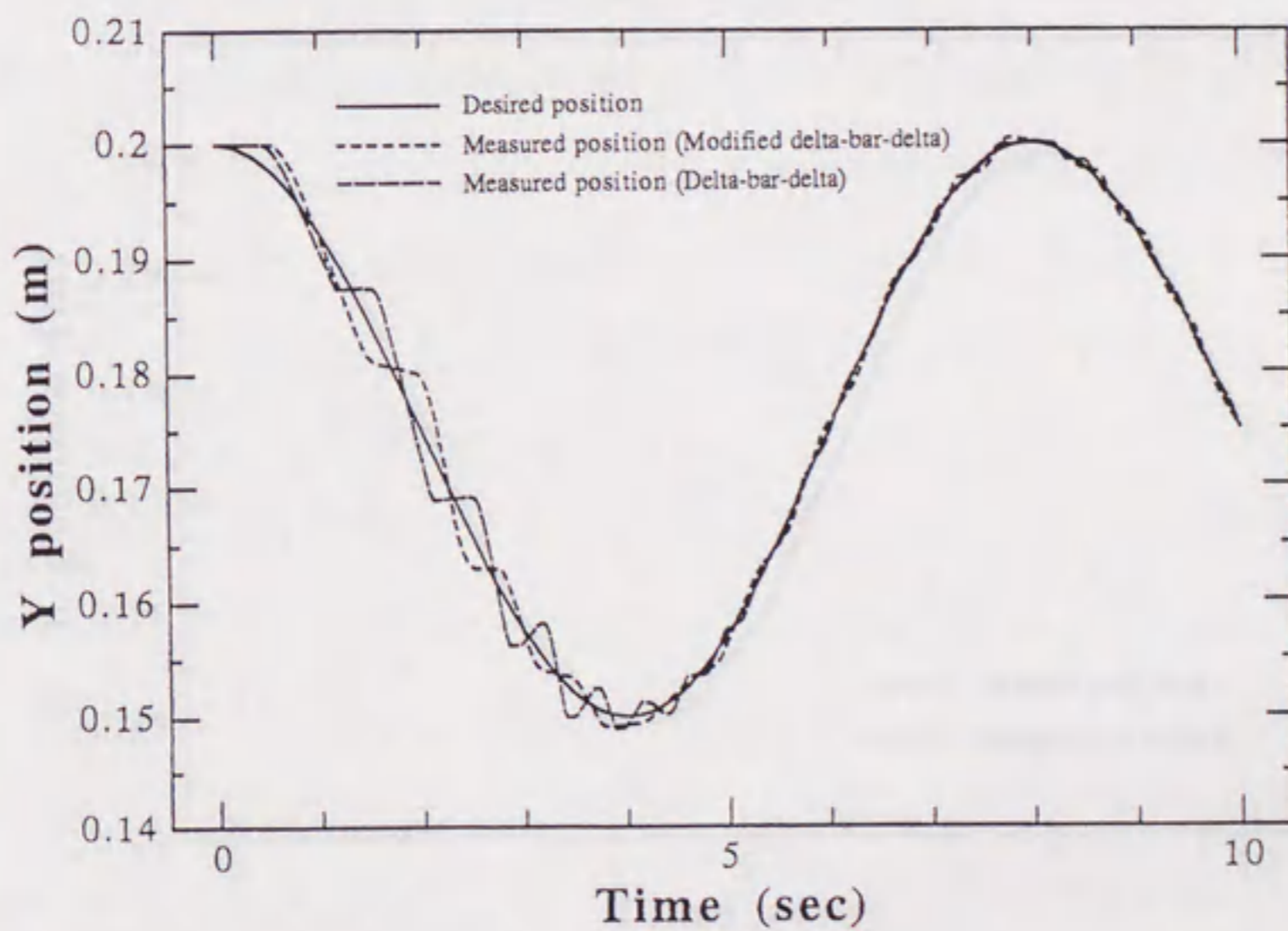


図 6.5 対象物の位置制御シミュレーション結果 (Y 方向)
提案した Delta-Bar-Delta 学習係数調整法の評価

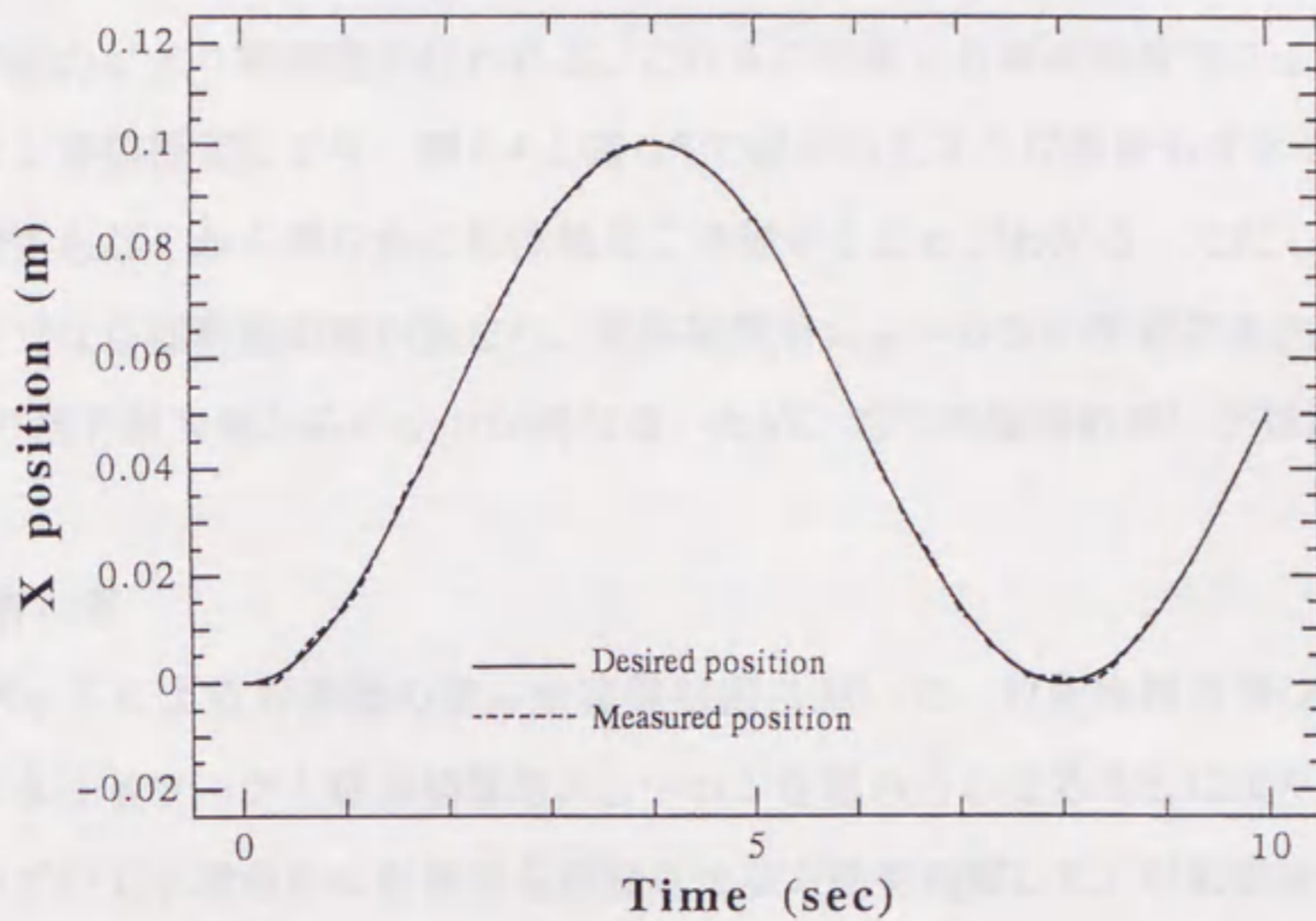


図 6.6 対象物の位置制御シミュレーション結果 (X 方向)
提案した摩擦補償法の評価

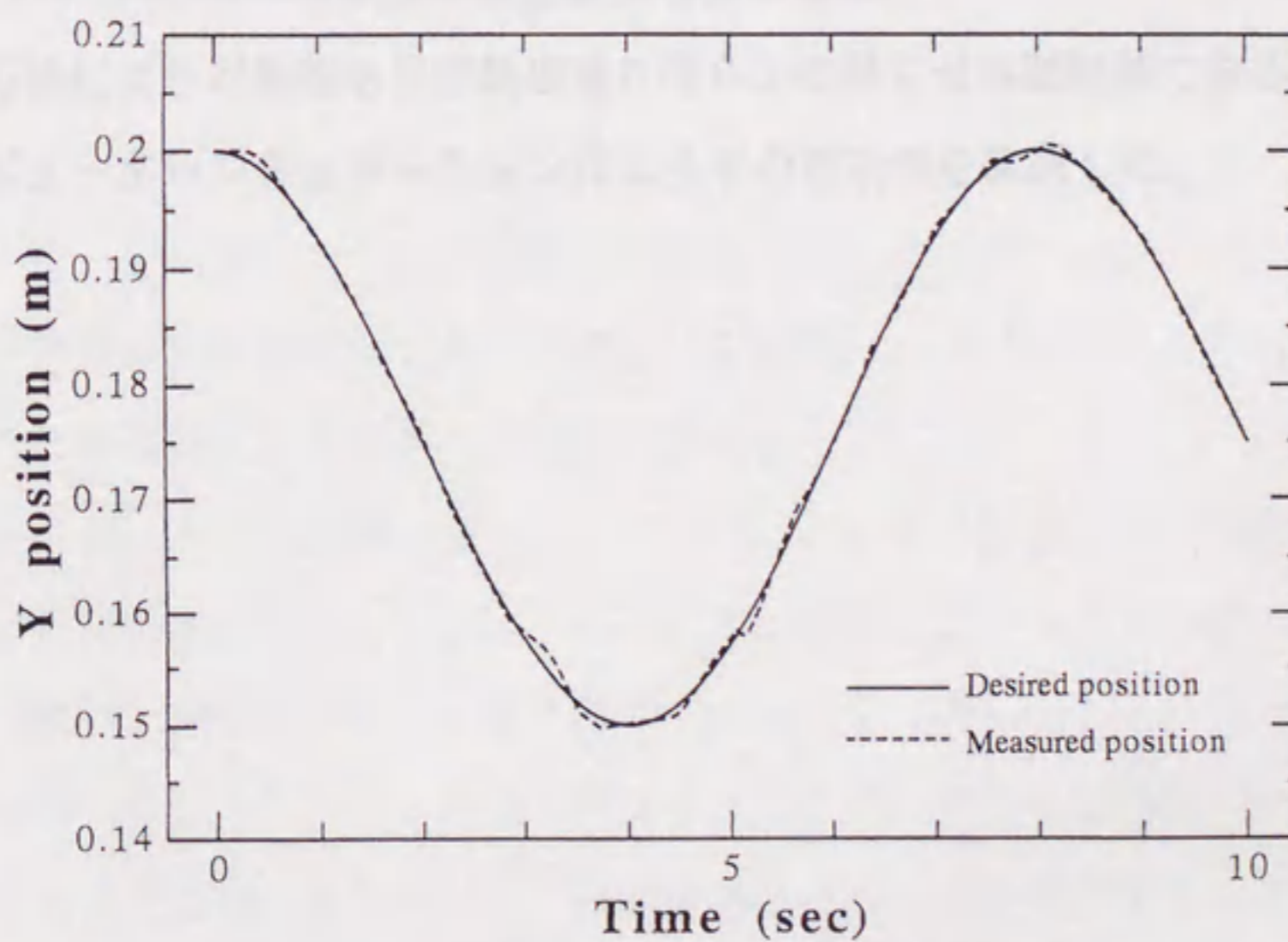


図 6.7 対象物の位置制御シミュレーション結果 (Y 方向)
提案した摩擦補償法の評価

習がほぼ完了しているため、その後の対象物の動きは滑らかになる。摩擦補償用ニューロンによる摩擦補償量が多少多すぎたり少なすぎたりしても、対象物操り用FNNの適応能力により微調整が行われる。これらの結果より摩擦補償用ニューロンによるクーロン摩擦補償により、図6.4と図6.5で見られたような振動もすること無く、制御開始後まもなくから滑らかに目標軌道に追従することがわかる。ただし、X方向とY方向とでは目標移動距離が異なり、摩擦補償用ニューロンの学習の条件が異なる(対象物操り用FNN部からの出力が異なる)ため、若干の傾向の違いが見られる。

6.7 結 言

複腕ロボットによる対象物の滑らせ移動制御において、対象物操り用ファジィ・ニューラルネットワークと摩擦補償用ニューロンを組み合わせることによりクーロン摩擦が働いていても滑らかに対象物を運動させる方法を提案した。対象物操り用ファジィ・ニューラルネットワークは対象物が運動しているときだけ学習し、摩擦補償用ニューロンは力が与えられているにも関わらず対象物が動かない場合だけ学習するという新しい方法を示した。また、ニューラルネットワークの学習係数を適切な値に調整する新しいDelta-Bar-Delta学習係数調整法を提案した。

これらの方法により対象物を目標軌道通り滑らかに滑らせ移動制御できることを説明し、コンピューターシミュレーションによりその有効性を確認した。

第7章 バリ取り作業への応用

7.1 緒言

近年、多くのロボットマニピュレータが工場等で利用されている。ロボットマニピュレータの重要でかつ基本的な作業の一つとして位置/力制御が挙げられるが、力制御を実用化するには幾つかの問題があるため、ほとんどの産業用ロボットは対象物等の位置制御のみに利用されている。しかしながら、最近では組立工場等においても、ロボットマニピュレータにより単なる位置制御だけでなく、より一層高度な作業を実現することが期待されている。グラインディング作業、バリ取り作業、拭き取り作業、あるいは組み立て作業のような高度な作業を実現させるためには、適切な送り速度で適切な力を作用させる必要があるが、実際問題として、優れた位置/力制御器を設計することが困難だけでなく、作業を遂行するのに必要となる目標力や目標送り速度を設定することも困難である。つまり、ロボットにより対象物に力を加えるには、実現すべき目標力を設定する必要があるが、要求された作業を遂行するには目標となる工具送り速度を設定する必要があるが、実際のバリ取り作業などでは、対象物の特性やバリのサイズなどにより熟練者の経験的な知識から目標力や目標工具送り速度などが決められている。したがって、もし熟練者の知識モデルの作成が可能であれば、効果的で柔軟な制御ルールが実現できるが、曖昧な言語的ルールで表わされる熟練者の知識から数学的な制御則を作成するのは困難である。

熟練者からロボットへの知識の教示については、これまでに幾つかの研究がなされている。YangとAsada [68] は、センサースペースから制御コマンドとの間のマッピングを用いて、熟練者からロボットへ言語的アプローチでスキル伝達を行うことを紹介し、LiuとAsada [69] はタスクレベルのプロセスパラメータと制御戦略パラメータとの間のマッピングを構築するためにニューラルネットワークを利用するスキル伝達法を紹介した。しかし、彼等の方法では、学習していないような予想外の状況を扱うのは困難である。また、XuとYang [70] はヒューマンスキルの伝達とモデル化に隠れマルコフモデルを利用している。これらの方法は、人間が作業をした結果生じるセン

サーデータを得るための予備実験と、モデル作成あるいはマッピング作成のための予備学習を必要とするが、実験をしたりモデルを学習したりするのは時間がかかる。

近年、ファジィ理論やニューラルネットワークが制御に応用されている。ファジィ制御 [10], [61], [99] は人間の知識を扱えることが知られている。したがって、プラントとその対象物の精密な数学モデルは制御器の設計には必要とされない。ファジィ制御器を設計するという事は、熟練者の制御スキルを言語的に表現されたルールにより伝達するのと同様である。ファジィ制御ルールは人間の制御ルールを近似したものであるが、体系的にファジィ制御器を設計するのは難しく、一度設計されたファジィ制御ルールとメンバーシップ関数は、例え制御器が完全なものでなくとも修正されない。一方で、ニューラルネットワーク制御器 [22], [31], [92] は経験から学習を進めたり、新しい環境に適応したりできる能力があることが知られている。これらの能力により、ニューラルネットワーク制御器は特に未知環境に対して有効である。しかし、ニューラルネットワーク制御は予備学習が必要であったり、適応するのに多少の時間がかかったりするため、適応型ニューラルネットワーク制御器をロボットマニピュレータの制御に使用した場合、制御器が未知環境に適応する前にロボットマニピュレータにより環境にダメージを与えてしまう恐れがある。さらに、ニューラルネットワークの各ウエイトの持つ意味は、作業者或いは設計者にとって理解し難いものとなっている。これらの問題を解決する方法の一つが、本論文でも提案しているようなファジィ制御とニューラルネットワーク制御とを組み合わせたファジィ・ニューラル制御器である [81]。これまでも述べてきたように、ロボットの知的化を図るため、このようなファジィ・ニューラルネットワークは種々の分野で活用されている [81] ~ [84], [100]。

本章では、予備実験や予備学習を必要とせずに、作業に必要な目標力と目標工具送り速度を獲得する知的作業プランナー（タスクレベル制御器）を提案し、例としてバリ取り作業に適用する [78]。提案する知的作業プランナー（図 7.1 参照）はファジィ・ニューロ作業プランナーとファジィ対象物特性評価器から構成されている。効果的な評価関数を用いて作業中にオンラインで調整を行うファジィ・ニューロ作業プランナーは目標工具送り速度を設定するために用いられる。ファジィ対象物特性評価器は、作業の対象となる未知対象物の材料特性を評価し、作業に必要な目標力を設定すると

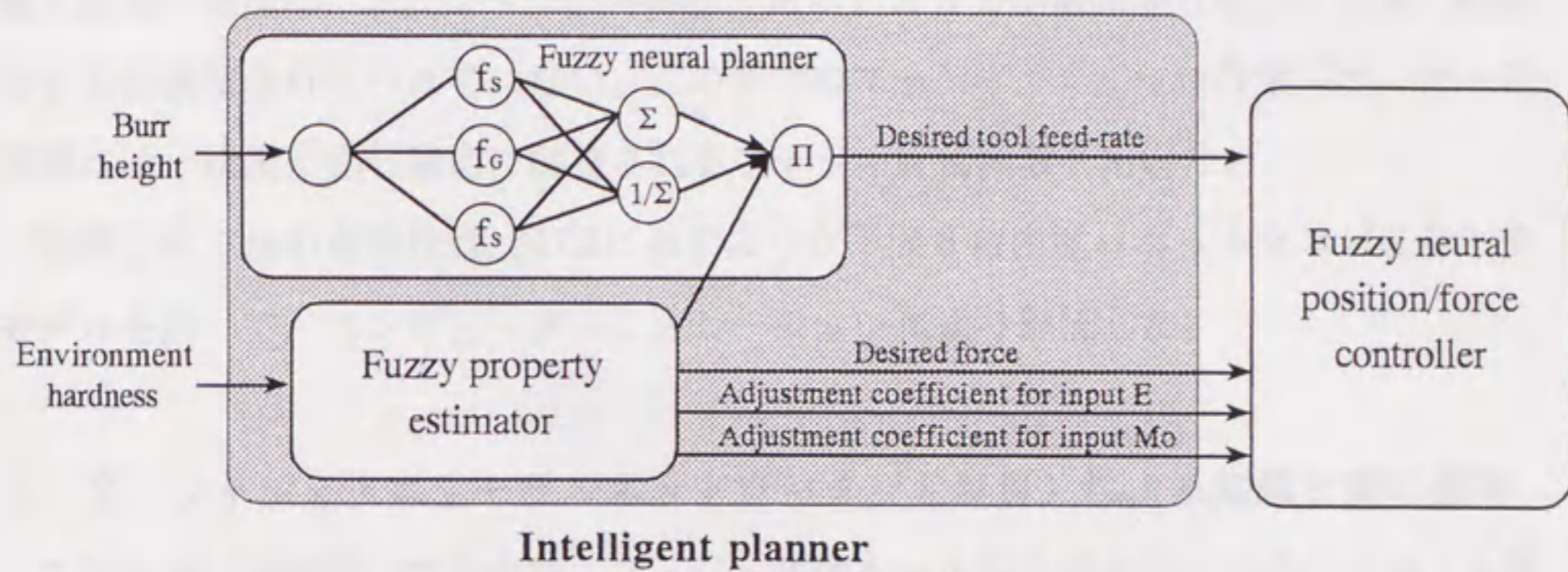


図 7.1 知的作業プランナー

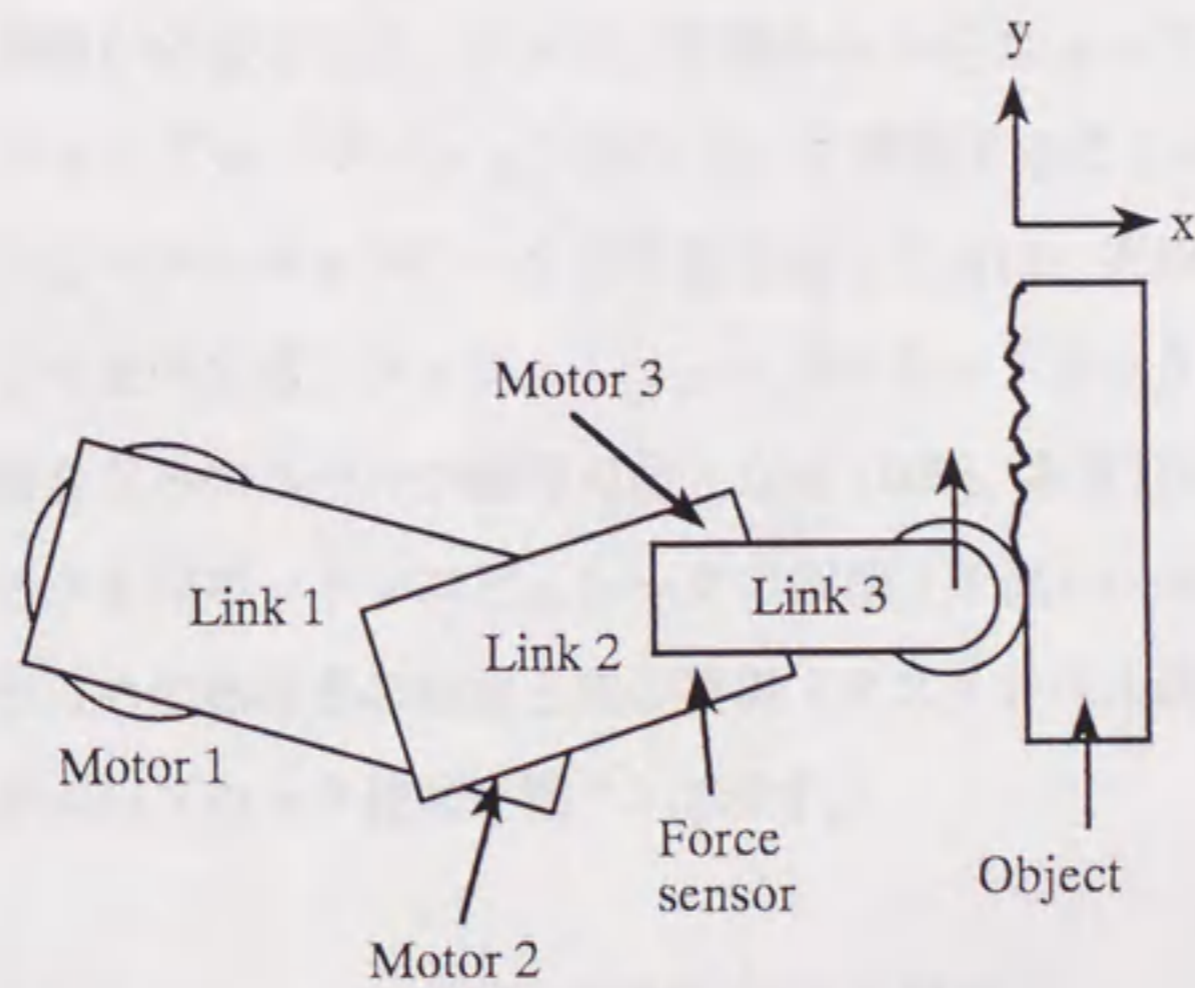


図 7.2 平面 3 自由度ロボットマニピュレータによるバリ取り作業

共に、下位レベルの制御器である適応型FNN力制御器を瞬時に調整するための入力値調整係数を生成する。

効果的な評価関数を設定することは、ファジィ・ニューロ作業プランナーの学習にとって重要な課題である。ファジィ・ニューロ作業プランナーは学習すべき場面でのみ学習を行うべきである。例えば、知的作業プランナーからの出力である目標力と目標工具送り速度は、下位レベルの制御器であるFNN力制御器が目標力に追従し始めるまでは調整されるべきではない。この章でのファジィ・ニューロ作業プランナーの学習には、状況に応じ適切に調整されるファジィ評価関数を用いた。

提案した方法の有効性は、図7.2に示すような平面3自由度ロボットマニピュレータモデルを用いて、コンピュータシミュレーションにより評価した。

7. 2 ファジィ・ニューラルネットワーク (FNN) による知識と業の獲得

これまでも説明してきたが、ファジィ制御ルールを作成するということは、人間の言語的表現のルールを用いて人間の知識と業に近似させることである。しかし、曖昧な人間の言語的表現のルールを用いて、人間の知識と業が完全にファジィ制御ルールに転移させることは難しい。したがって、設計されたファジィ制御ルールをより効果的にするためには調整が必要となる。ファジィ制御ルールをニューラルネットワークに変換した後は、バックプロパゲーション法を用いて調整することが可能となる。この様にファジィ・ニューラルネットワークを学習させることは、近似した人間の知識と業を調整することを意味する。ファジィ・ニューラルネットワークを調整することによって人間の熟練者なみのルールが獲得可能となる [100]。本章では、ファジィ・ニューラルネットワークをロボットマニピュレータの制御（下位レベル制御）だけでなく、バリ取り作業のための熟練者の知識と業の獲得（タスクレベル制御）にも適用した。提案するシステムのブロック線図を図7.3に示す。

7. 3 ロボットマニピュレータの制御（下位レベル制御）

ロボットマニピュレータを用いてバリ取り作業を行うため、3. 2節で説明したハイブリッド制御 [71], [72] を用いることにする。ロボットによりスムーズな動きでバリ取り作業を行うには、バリ取り工具と対象物の間のクーロン摩擦を補償する必要が

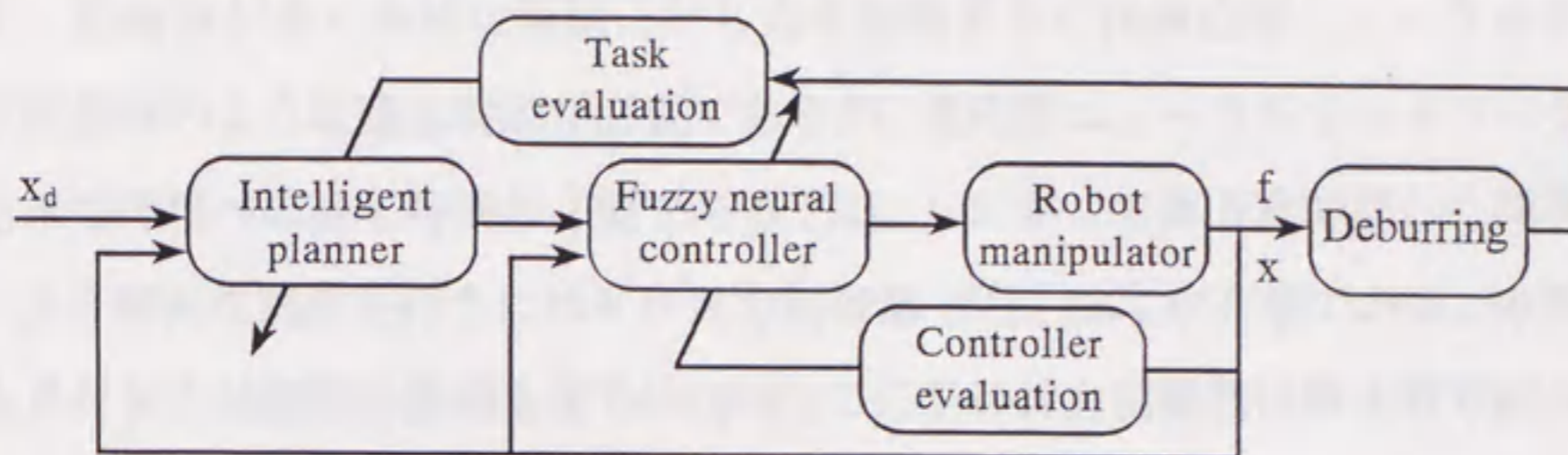


図 7.3 提案するシステムのブロック線図

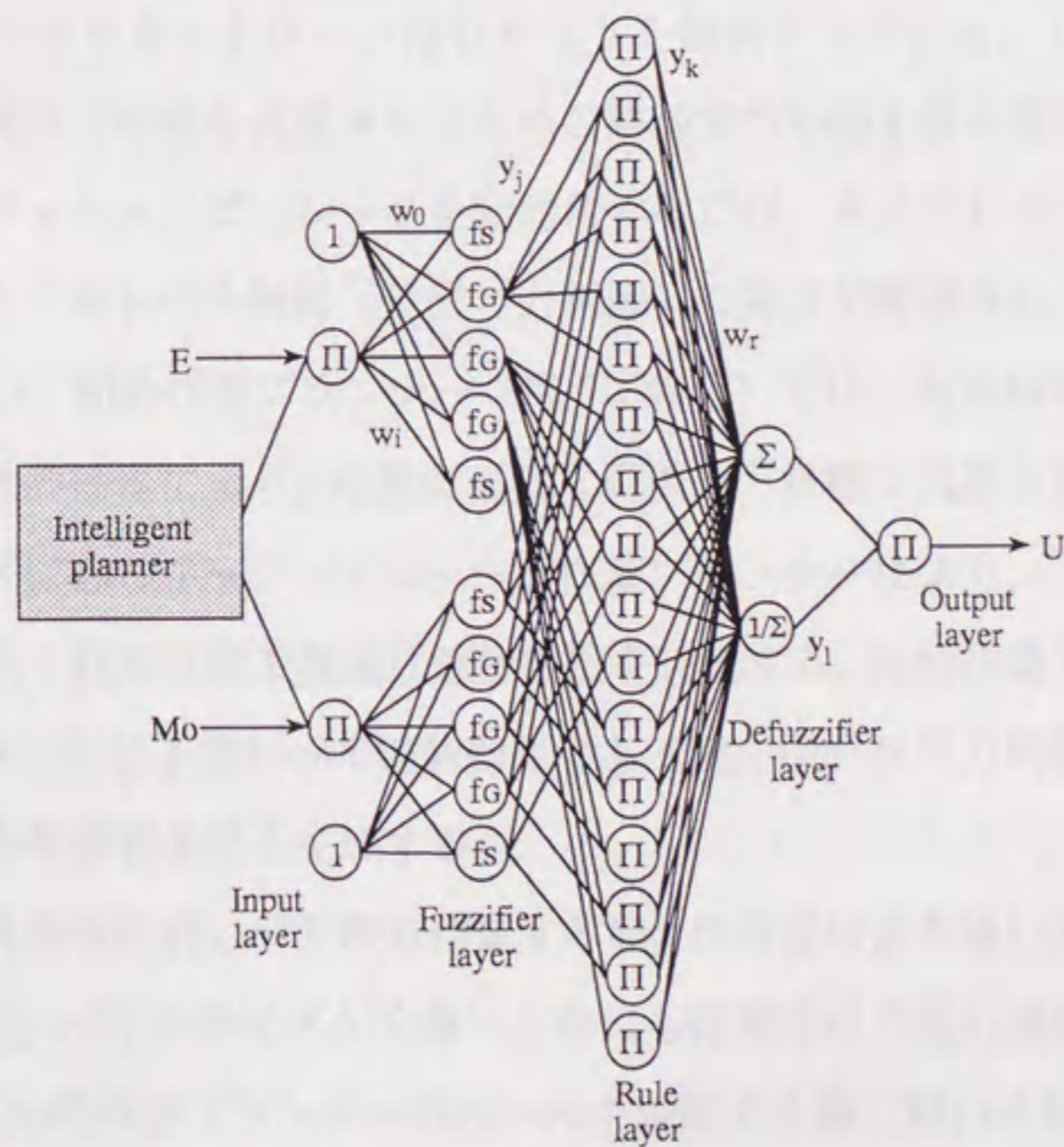


図 7.4 FNN力制御器の詳細

ある。この様に、バリ取り工具を対象物に押し付けながらスムーズな位置制御を行うには、第3章で提案した摩擦補償用ニューロン部と軌道制御用FNN制御部から成り、切り替え学習を行う位置制御器 [84] が適切である。この位置制御器については既に第3章で詳しく説明されているので、ここでは詳細な説明を省略する。

一方、動特性が全く未知な環境に対し力を制御するには適応型ニューラルネットワーク制御器のような適応制御が必要であるが、適応型ニューラルネットワーク制御器でも未知環境への適応時間が必要である [92]。したがって第3章で詳しく説明した通り、より確実な制御を行うにはFNN力制御器 [81], [82] が有効である。本章で利用するFNN力制御器の詳細を図7.4に示す。このFNN力制御器は第4章で紹介したものと基本的に同一であるが、ここではファジィ環境評価器に代り作業に必要な目標値も生成する知的作業プランナーが接続させている。

7. 4 知的作業プランナー

ファジィ・ニューラルネットワークはロボットを制御するのにも、ロボットマニピュレータにより高度な作業を実現させるための熟練者の知識を業を獲得するのにも効果的である。ロボットマニピュレータを知的にするには、タスクレベル制御（知的作業プランナー）と下位レベル制御（位置/力制御）の両方が要求され、両方共知的でなければならない。知的作業プランナー（図7.1参照）では、対象物の硬さに応じファジィ対象物特性評価器により、作業に必要な目標力と目標工具送り速度係数を生成し、バリのサイズに応じファジィ・ニューロ作業プランナーにより、ファジィ対象物特性評価器で生成された目標工具送り速度係数を調整する。知的作業プランナーはまた、対象物の硬さに応じ下位レベルの制御器である適応型FNN力制御器を瞬時に調整するための入力値調整係数を生成する。

材料特性が硬い対象物に対しバリ取り作業する際、作業者はより強い力でゆっくりと工具を送る。また、バリのサイズが大きいときにも作業者は工具の送り速度を下げる。この様な知識が知的作業プランナーのルールを設計する際に用いられる。通常、これらのルールは曖昧な言語的ルールを基に設計されたものなので完全なものではない。そのため、何らかのルール調整能力が必要となる。もしバリ取り作業で削ったバリの量が少な過ぎたり多過ぎたりした場合、知的作業プランナーで生成される目標力或い

は目標工具送り速度を調整する必要がある。本研究では、作業中にファジィ・ニューロ作業プランナーのメンバーシップ関数とファジィルールをサンプリング時間ごとに調整することにより、目標工具送り速度を調整し、適切な量のバリを削る方法を提案する。

7. 5 ファジィ対象物特性評価器

知的作業プランナー内のファジィ対象物特性評価器は未知対象物の硬さを評価し、その対象物のバリ取りを行うために適切な目標力と目標工具送り速度係数を生成する。ロボットにより対象物に対し押し込んだ距離とそれにより発生した反力情報を利用し、次式により対象物のバネ定数が得られる。

$$K_s = f / x_e \quad \dots\dots\dots (7.1)$$

ここで K_s は対象物のばね定数、 f は対象物に加えられた力、 x_e は対象物表面の変位である。通常、硬いものと柔らかいものではばね定数の大きさはオーダーが異なるため、入力値のファジィ変数の設定が難しい。そこで本研究では、次式で得られるばね定数のオーダー値をファジィ対象物特性評価器への入力値として用い、ファジィ変数を定義するものとする。

$$K_f = \log (f / x_e) \quad \dots\dots\dots (7.2)$$

ファジィ対象物特性評価器では目標値を生成するため、2種類のファジィルールが用いられる。一つは、作業に必要な目標力を生成するためのものであり、もう一つは、対象物の硬さに応じ目標工具送り速度係数を生成するためのものである。本評価器のファジィルールは人間の熟練者の知識と経験を基に設計される。

ファジィ対象物特性評価器のもう一つの役割は、適応型FNN力制御器の入力値調整係数を生成することである。未知対象物の硬さがFNN力制御器を設計する際に用いたモデルよりも柔らかい環境か、少し硬い環境の場合は、FNN力制御器によりオーバーシュージングを生じさせることもなく環境に対し目標通りの力を加えることができる。しかし、これまでも述べてきた通り、未知対象物が見積もっていたものよりもかなり硬かった場合、オーバーシュージングが生じたりする危険性がある。そこで、ここでは第4章で提案したのと同じ方法でFNN力制御器の入力値調整係数

を生成するものとする。

7. 6 ファジィ・ニューロ作業プランナー

知的作業プランナー内のファジィ・ニューロ作業プランナーは、バリ取り作業において、バリのサイズと対象物の硬さに応じ目標工具送り速度調整係数を生成するという重要な役割を果たす。ファジィ・ニューロ作業プランナーへの入力情報としてバリのサイズを用いることにした。

まず最初に、ファジィ・ニューロ作業プランナー内部でバリのサイズに応じ目標工具送り速度調整係数を生成する。次に、内部で生成された目標工具送り速度調整係数を、対象物の硬さに応じてファジィ対象物特性評価器で生成した目標工具送り速度調整係数で乗じることで、バリのサイズと対象物の硬さの両方に応じた目標工具送り速度調整係数が生成される。この係数を標準工具送り速度に乗ずることで、バリのサイズと対象物の硬さの両方を考慮した目標工具送り速度が生成される。

ファジィ・ニューロ作業プランナーのルールは、当初曖昧な人間の熟練者の知識から作成されたものなので、バックプロパゲーション法を利用して、さらに効率の良いバリ取り作業が実行できるよう調整する必要がある。カメラ [101] やレーザ変位計 [102] などを利用してバリの高さと削られたバリの高さの両方が測定可能とし、それらの2乗誤差をファジィ・ニューロ作業プランナーの評価関数として利用した。この評価関数を最小にすることで、ロボットによる効率の良いバリ取り作業の実現が期待される。

7. 6. 1 ファジィ評価関数

ファジィ・ニューロ作業プランナーは、下位の制御器であるFNN力制御器とFNN位置制御器により最初に設定された目標値に測定値が到達するまでの間は学習を進めるべきではない(ウエイト値を変更すべきではない)。ファジィ・ニューラル作業プランナーが下位制御の状況に応じて効率良く学習するため、本章ではファジィ評価関数を提案する。実際には、評価関数そのものではなく、評価関数の係数がファジィ制御されることとなる。ファジィ制御される評価関数の入力変数は下位レベル制御器(力制御器と位置制御器)の誤差である。評価関数の係数は、両方の入力変数の値が大

きい（下位レベル制御器の誤差が大きい）場合には0あるいは非常に小さい値に、両方の入力変数がゼロ（下位レベル制御器が目標力／軌道に精度良く追従）に近づくとつれ1になっていくように制御される。このファジィ制御される評価関数を学習に用いることにより、ファジィ・ニューロ作業プランナーは未知対象物のバリ取り作業に効率良く適応できることになる。

7. 7 シミュレーション

提案した知的作業制御法を評価するため、平面3自由度ロボットマニピュレータモデル（図7.2参照）を用いてコンピュータシミュレーションを行った。シミュレーションでのバリのサイズは、通常の本バリのサイズは3mm前後であることを考慮し、他の研究例 [69], [102] でも見られるように本研究でも図7.5で示すような3mm程度のサイズに設定し、形状は3種類用意した。また、削るべきバリの高さで削られたバリの高さはモニターできるものと仮定した。サンプリング時間は全てのシミュレーションにおいて1msに設定した。またシミュレーションでは、量子化誤差と演算時間遅れは無視できる程度、センサーおよびモータの動作遅れはないものと仮定し、ノイズの影響は無視した。ロボットマニピュレータのエンド・エフェクタは、加速度分解制御 [103] により、対象物に対し常に直角になるように制御される。対象物に作用させる力と対象物上の位置（軌道）は、本論文で提案しているFNNを利用した知的な位置／力制御器により制御される。

提案する知的作業プランナーを評価するため、バリ取り作業のシミュレーションを柔らかい対象物、標準の硬さ（制御器のファジィルールを設計する際に見積もった硬さ）の対象物、硬い対象物それぞれに対して行った。知的作業プランナー内のファジィ対象物特性評価器は、標準の硬さの対象物に対し20Nの目標力、より硬い対象物にはそれより高い目標力、より柔らかい対象物にはより低い目標力を生成するように設定した。図7.6に知的作業プランナーが3種類の対象物に対し生成した目標力を示す。このシミュレーションでは、標準の硬さ（中間の硬さ）の対象物は制御器のファジィルールを設計する際に見積もったものと同じ硬さに設定し、柔らかい対象物として標準の硬さの対象物のばね定数を1/10倍に変更したもの、硬い対象物として標準の硬さの対象物のばね定数を10倍に変更したものを利用した。図7.6の結果より、知的作業プラ

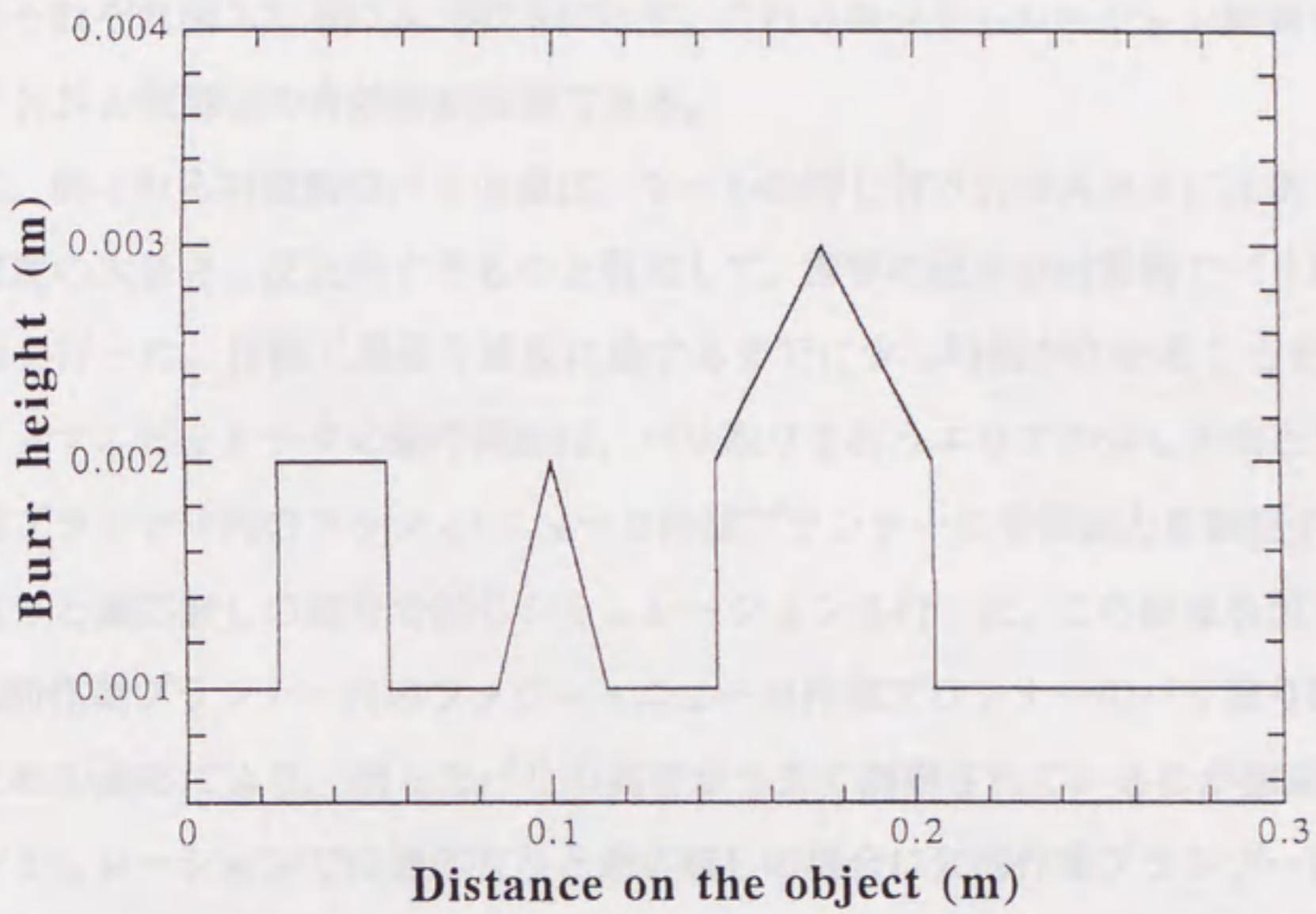


図 7.5 設定したバリのモデル

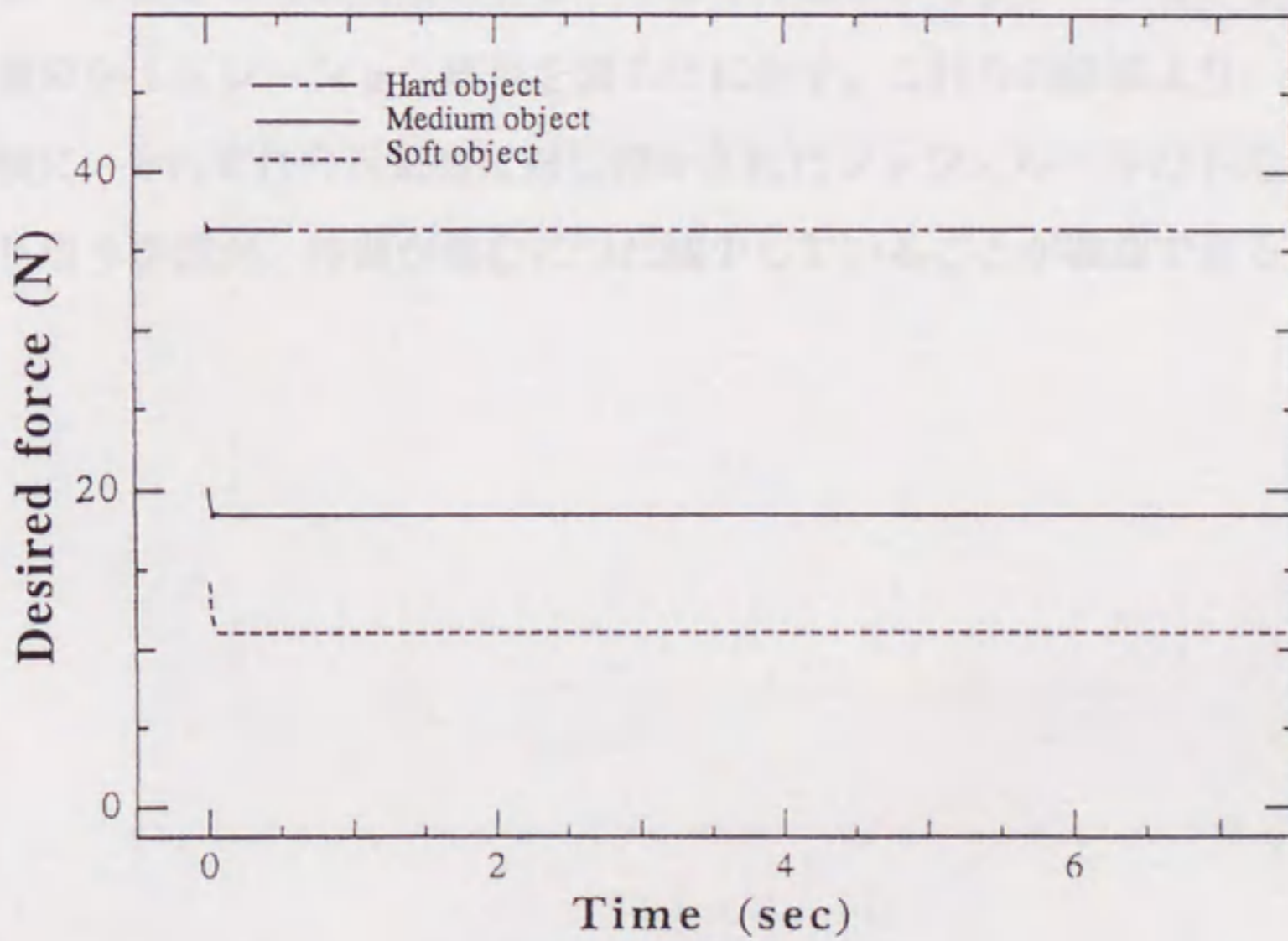


図 7.6 知的作業プランナーにより生成された目標力

ンナーにより各々の材料に対し適切な目標力が生成されていることがわかる。柔らかい対象物、標準の硬さの対象物、硬い対象物のそれぞれに対し加えられた力のレスポンスをそれぞれ図 7.7, 図 7.8, 図 7.9 に示す。これらのシミュレーション結果により適応型 FNN 力制御器の有効性が確認できる。

次に、削られる対象物のバリの量は、ツールの押し付け力の大きさに比例し、工具送り速度の大きさに反比例するものと仮定して、標準の硬さの対象物でバリ取り作業の評価を行った。目標工具送り速度に達するまでに少し時間がかかることを考慮し、ロボットマニピュレータの動作開始は、バリ取りを行うエリアの少し手前とした。知的作業プランナー内のファジィ・ニューロ作業プランナーの学習能力を確認するため、適応有り と 適応無し の場合で同じシミュレーションを行った。この結果を図 7.10 に示す。知的作業プランナー内のファジィ・ニューロ作業プランナーのバリ取り誤差をなくすための適応により、削ったバリの高さがうまく調整されているのが確認できる。このシミュレーションでの適応有り と 適応無し の場合に知的作業プランナーにより生成された目標工具送り速度を図 7.11 に示す。図 7.12 には目標工具送り速度に追従するために FNN 位置制御器により制御された工具送り速度が示されている。この結果により適応型 FNN 位置制御器の有効性が確認できる。

最後に各々の硬さの対象物に対するバリ取り作業のシミュレーションを行った。これらの作業のシミュレーション結果を図 7.13 に示す。これらの結果より、バリ取り作業開始直後に、それぞれの対象物に対し設計されたファジィルールの不完全さのため生じたバリ取り誤差が、時間が進むにつれ減少していることが確認できる。

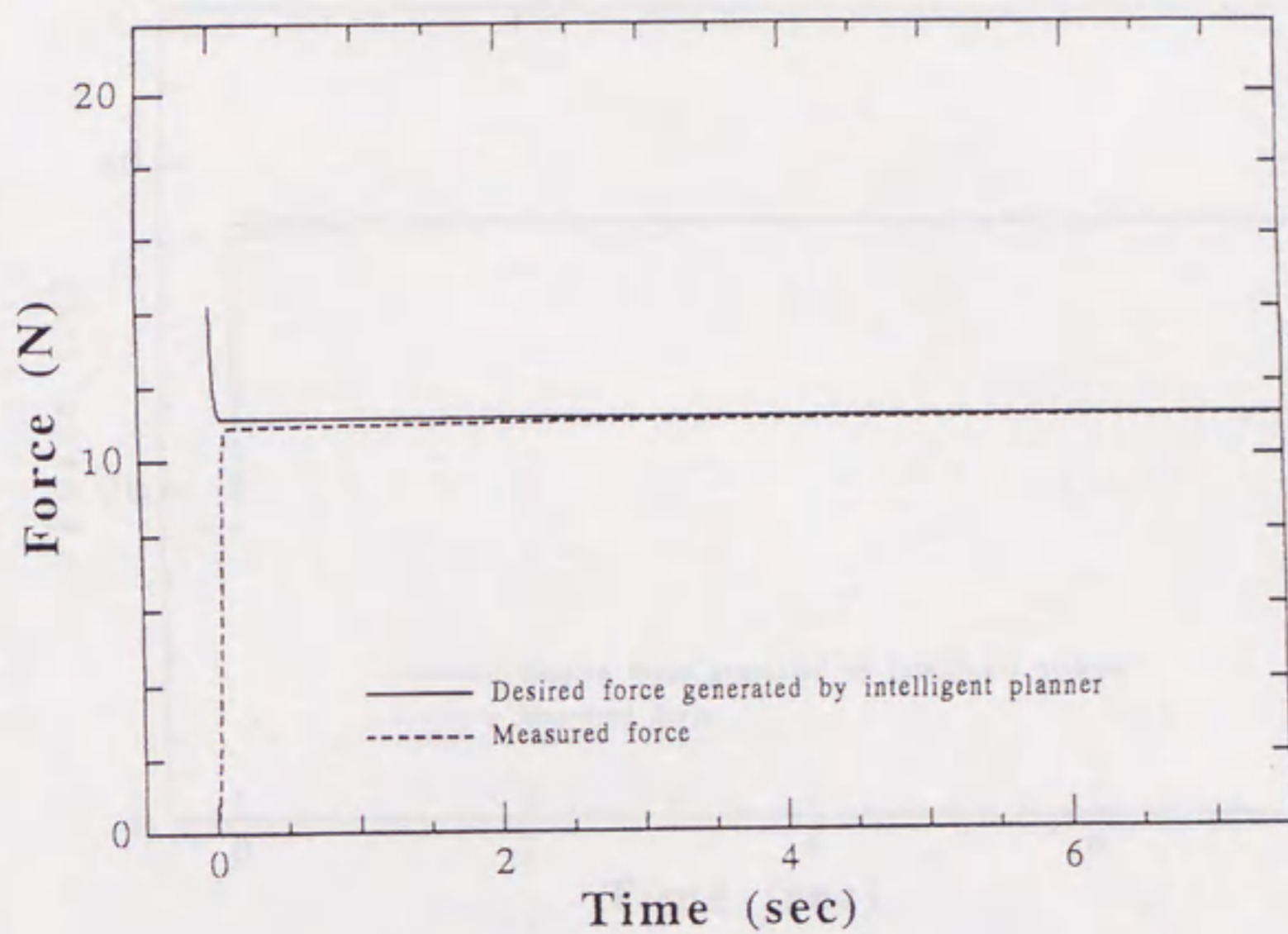


図 7.7 柔らかい対象物に対する力制御結果

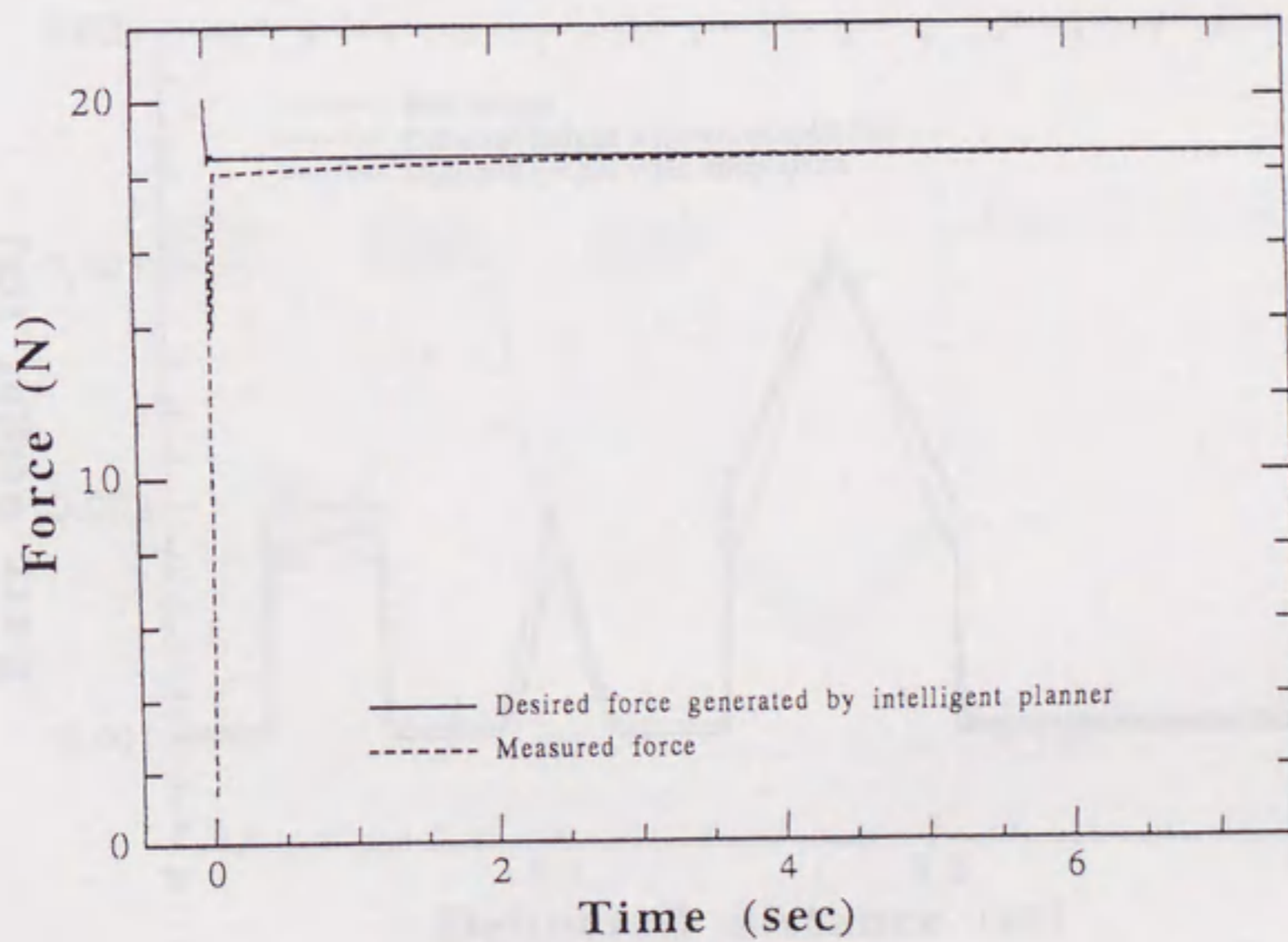


図 7.8 標準の硬さの対象物に対する力制御結果

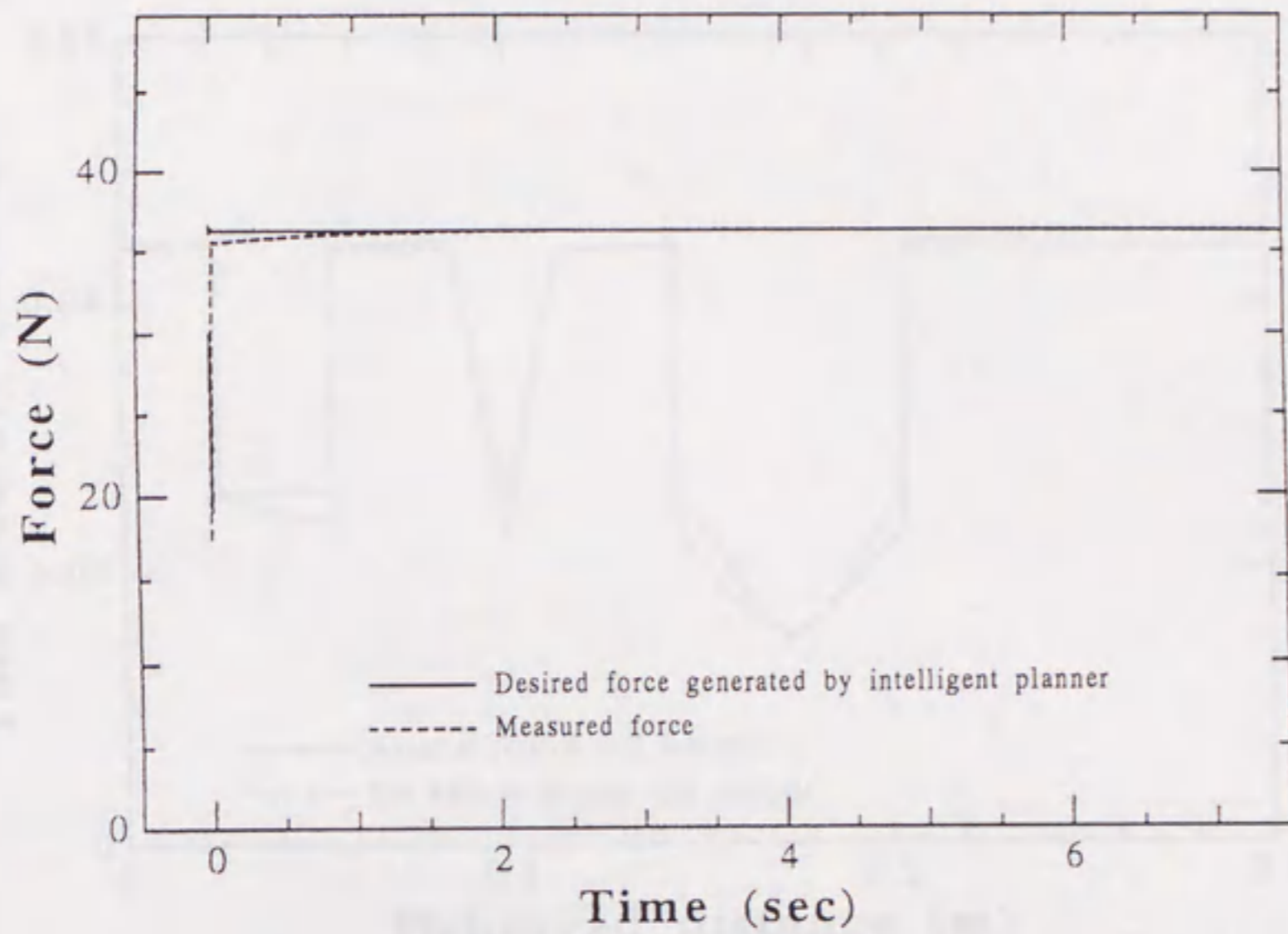


図 7.9 硬い対象物に対する力制御結果

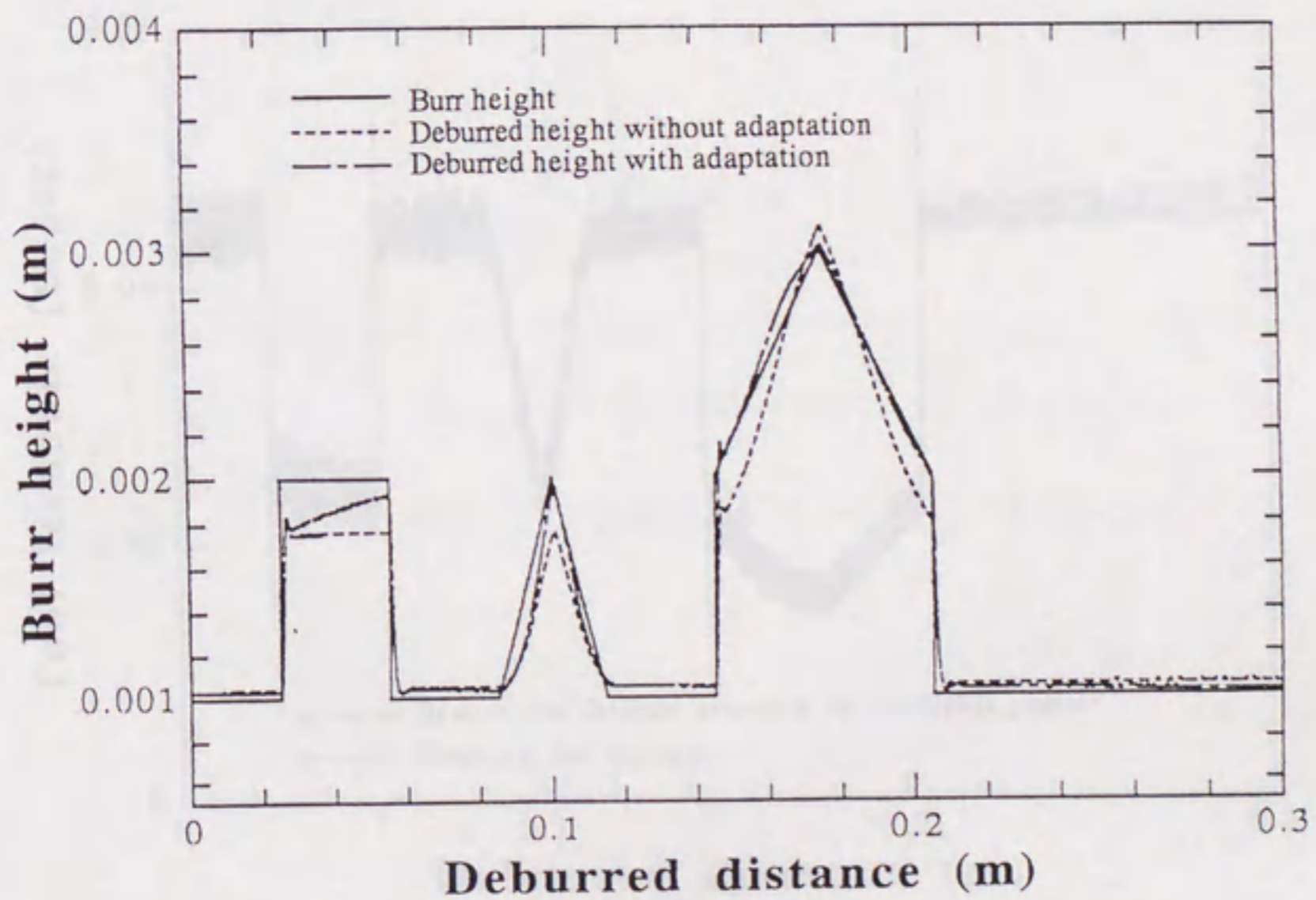


図 7.10 バリ取り作業シミュレーション結果

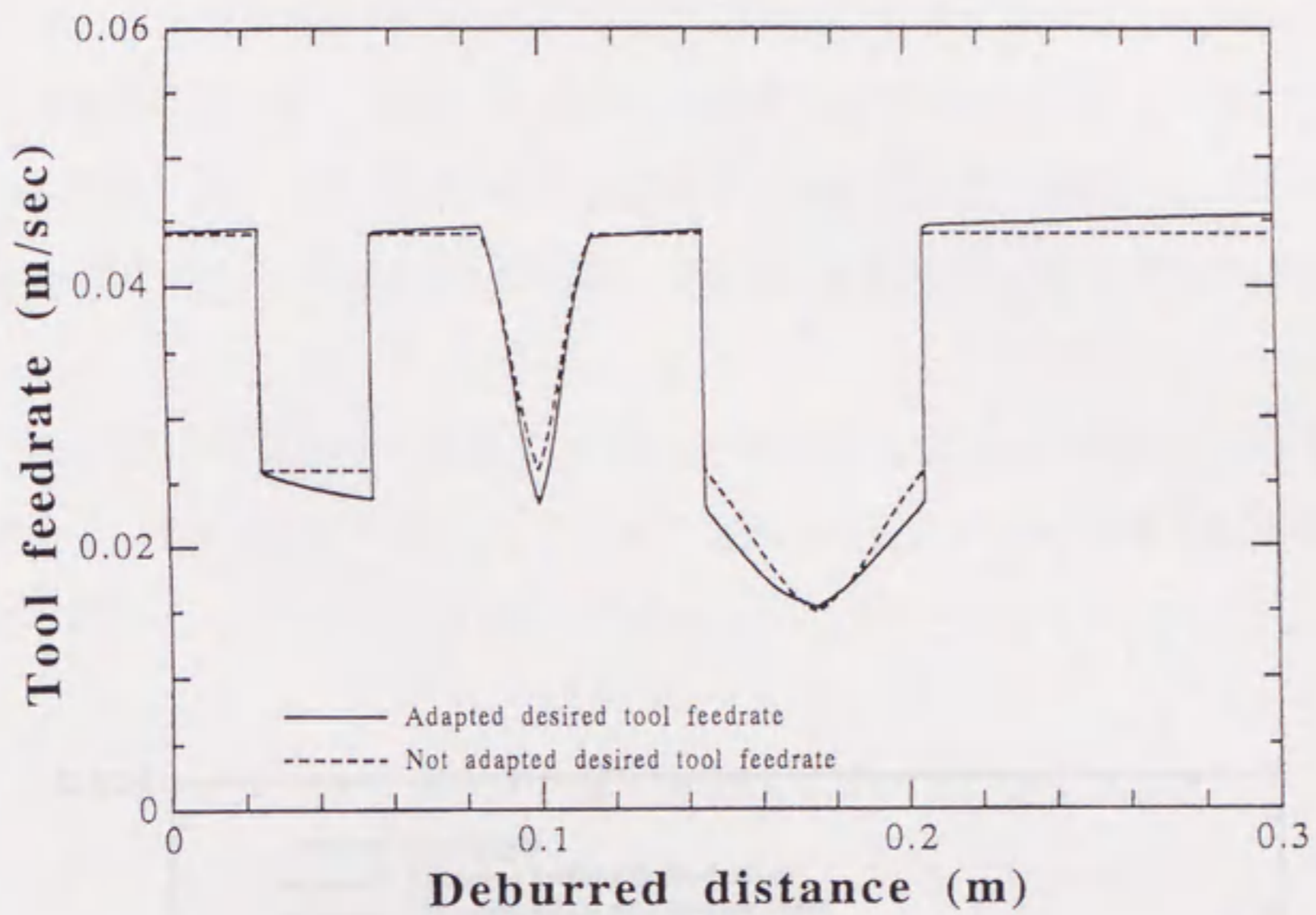


図 7.11 知的作業プランナーにより生成された目標工具送り速度

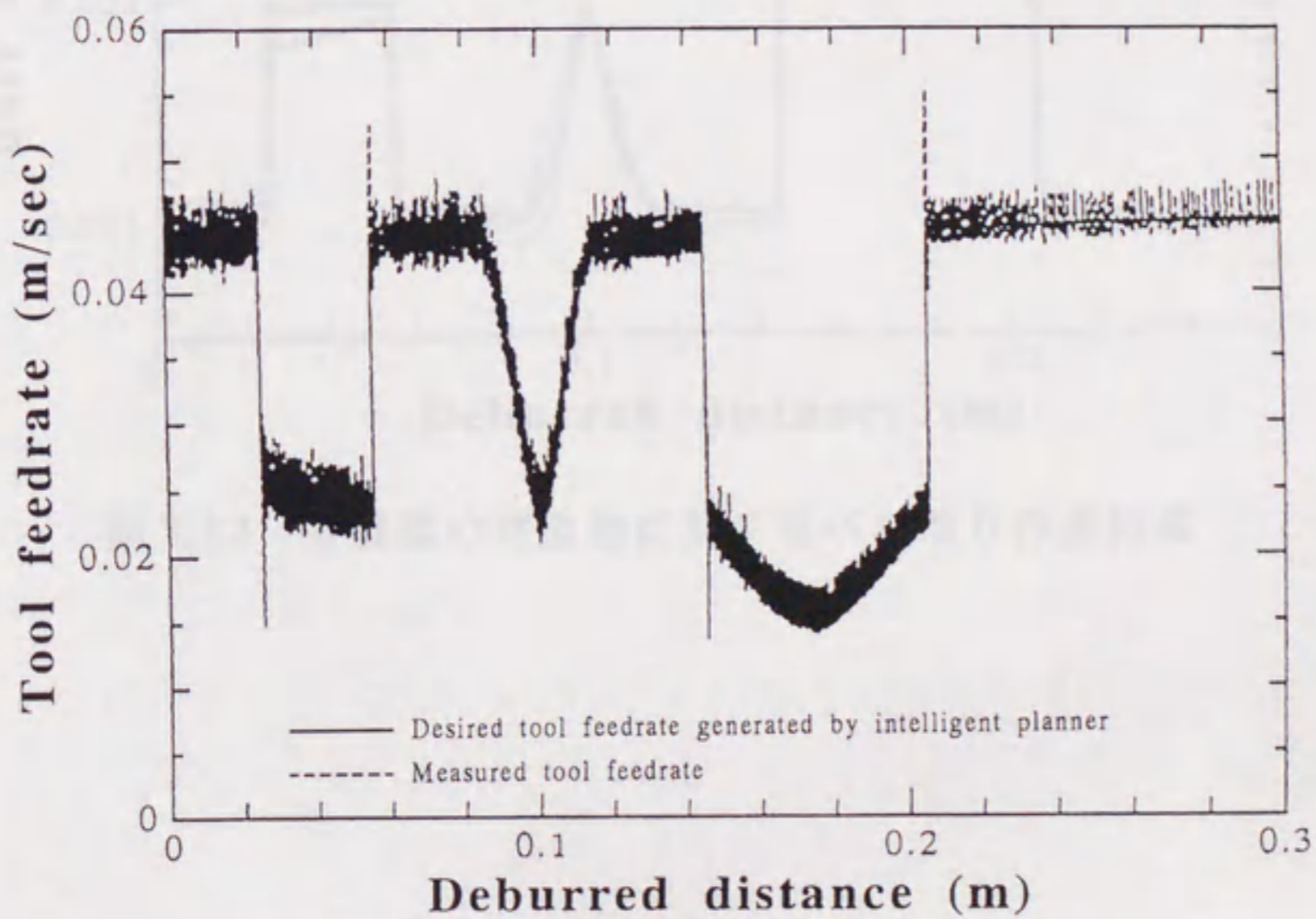


図 7.12 バリ取り作業時の工具送り速度の結果

7.13 図 7.13

図 7.13 は、バリ取り作業の結果を示すグラフである。縦軸はバリ高さ (Burr height) をメートル (m) で表し、横軸はバリ取り距離 (Deburred distance) をメートル (m) で表している。グラフには四つのデータ系列が示されている: 1. Burr height (実線): 元のバリ高さの分布。2. Deburred height with hard object (長短線): 硬い対象物に対するバリ取り後の高さ。3. Deburred height with medium object (短長線): 中程度の対象物に対するバリ取り後の高さ。4. Deburred height with soft object (点線): 柔らかい対象物に対するバリ取り後の高さ。元のバリ高さには、0.001m 程度の低レベルのバリ、0.002m 程度のステップ状のバリ、0.1m 付近の山状のバリ、0.15m 付近の大きな山状のバリ、および 0.2m 付近のステップ状のバリが含まれている。硬い対象物では、元のバリ高さの大部分が除去され、残りのバリは元の高さの約 80% 程度に減少している。中程度の対象物では、元のバリ高さの大部分が除去され、残りのバリは元の高さの約 60% 程度に減少している。柔らかい対象物では、元のバリ高さの大部分が除去され、残りのバリは元の高さの約 40% 程度に減少している。

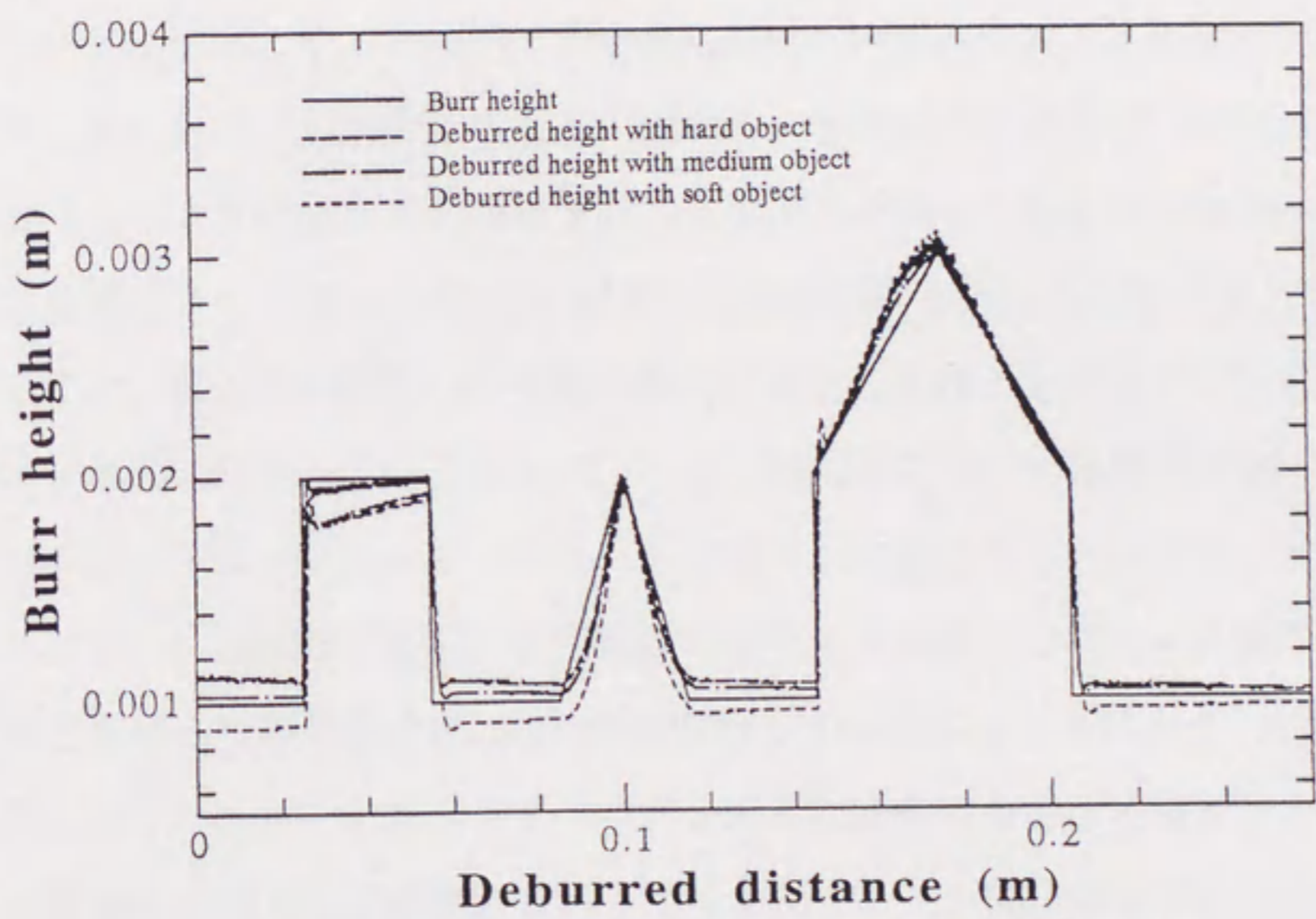


図 7.13 3 種類の対象物に対するバリ取り作業結果

7. 8 結 言

ファジィ・ニューロを利用したアプローチにより、ロボットマニピュレータによる知的な作業を実現させた。提案した方法は、高度な作業への適用例としてバリ取り作業に用いられた。この方法を用いることにより、高度な作業を実現するためのプランニングと制御を行うために人間の熟練者の知識と業を容易に利用することが可能となった。

平面3自由度ロボットマニピュレータモデルを用いて、未知対象物に対するコンピュータシミュレーションを行い、提案した知的制御法の任意の対象物に対する有効性を確認した。

第8章 結 論

8. 1 総まとめ

本研究は、従来使用分野が限られていたロボットマニピュレータが、より広い分野で、より柔軟に、より効率良く活用されることを目指し、言語的な表現や曖昧なデータを扱うことができるファジィ理論や適応/学習能力のあるニューラルネットワーク及びそれらを結合させたファジィ・ニューラルネットワークを用いて、未知環境に対する位置/力制御、未知対象物の操り、或いは未知対象物に対する高度な作業を実現させることを目標に行われた。

まず、ロボットマニピュレータにより未知環境に対する位置/力制御を実現させるため、第2章において、未学習のニューラルネットワークを直接的に多自由度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へ応用し、動特性の未知な環境に対し制御器を適応させる方法を提案し、提案した制御器の有効性を実験により証明した。

第3章では、第2章で紹介した未知対象物に対する位置と力のハイブリッド制御では抑制できなかったオーバーシュートینگを無くすため、制御器設計者の知識を利用したファジィ・ニューラルネットワークを用いた位置と力のハイブリッド制御器を提案した。また、力制御において未知環境の特性がファジィ制御ルール設計時に想定したものと異なっていた場合に、入力値調整用ニューラルネットワーク(IANN)を用いてファジィ・ニューラルネットワークを瞬時に調整する方法を提案し、力制御の際の位置制御において未知環境とロボットマニピュレータ間の摩擦を行うため、摩擦補償用ニューロンをファジィ・ニューラルネットワーク制御器に加え、状況により学習を切り替えながら制御する方法を提案した。さらに、ロボットマニピュレータのファジィ・ニューラルネットワーク制御器の学習則として利用されるバックプロパゲーション法の評価関数として、ファジィ制御された評価関数を用いて効率的に学習を行う方法も提案した。提案した制御器の有効性はコンピュータによるシミュレーションにより確認した。

第4章では、形状等が未知な環境に対し位置と力のハイブリッド制御を実現させる

ため、大きさと方向が曖昧なファジィベクトルを提案し、ノイズを含む力センサーからの情報をファジィベクトルにより有効に扱うことで、力制御方向を効率良く探索するアルゴリズムを提案した。また、力制御において未知環境の特性がファジィ制御ルール設計時に想定したものと異なっていた場合に、第3章で提案したIANNより汎用的で、かつ力制御方向の調整具合を考慮しながらファジィ・ニューラルネットワークの調整を行うファジィ環境評価器を提案した。ノイズを意図的に含ませたシミュレーションにより提案した方法の有効性を確認した。

次に、ロボットマニピュレータにより重量や特性の未知な対象物を操ったり、未知な外乱が作用している対象物を操ったりすることを実現させるため、第5章において、未知対象物を操るためのメインのファジィ・ニューラルネットワーク制御器と、メイン制御器からの出力を目標力としてロボットマニピュレータを制御するサブのファジィ・ニューラルネットワークから構成される制御システムを提案した。また、ロボットマニピュレータを制御するサブ制御器の内部に標準環境モデルを置き、未知対象物の特性がファジィ制御ルールを設計した際に見積もったものより硬かった場合にファジィ・ニューラルネットワーク制御器の調整を行う簡易的な方法を提案した。提案した制御法の有効性は、コンピュータによるシミュレーションにより確認した。

第6章では、複腕ロボットマニピュレータによる対象物の滑らせ移動制御において、第3章で提案した位置制御器を応用し、力が与えられているにも関わらず対象物が動かない場合だけ学習する摩擦補償用ニューロンと、対象物が運動しているときだけ学習する対象物操り用ファジィ・ニューラルネットワークを組み合わせた対象物操り制御器を提案した。また、ファジィ・ニューラルネットワークをバックプロパゲーション法により学習させる際に適切な学習係数を設定するため、学習状況に応じ学習係数が小さい場合の増加割合を変化させる新しいDelta-Bar-Delta学習法を提案した。

最後に第7章では、ロボットマニピュレータによる高度な作業を実現させるため、予備実験や予備学習を必要とせず、作業に必要な目標力と目標工具送り速度を獲得する知的作業プランナー（タスクレベル制御器）を提案し、例としてバリ取り作業に適用した。提案された知的作業プランナーはファジィ・ニューロ作業プランナーとファジィ対象物特性評価器から構成され、作業に必要な目標力と目標工具送り速度を生成すると共に、下位レベルの制御器である適応型ファジィ・ニューラルネットワー

ク力制御器を瞬時に調整するための入力値調整係数を生成する役割を果たした。また、ファジィ・ニューロ作業プランナーの学習には、学習すべき場面でのみ学習を行うファジィ評価関数を用いることを提案した。

これらの研究により、未知環境においてロボットマニピュレータを利用して高度な作業が実現できることが示された。

8. 2 今後の課題

本研究では、未知環境においてロボットマニピュレータにより高度な作業を実現させることを目標とし、ファジィ理論やニューラルネットワークおよびそれらを結合させたファジィ・ニューラルネットワークを用いて新しい制御法を提案したが、これらの制御法の限界や安定性の解析については、まだ数学的に十分な検討がされていないため、今後の課題としたい。

また、位置/力ハイブリッド制御の位置制御時の摩擦補償や対象物の滑らせ移動制御時の摩擦補償では、静止摩擦や動摩擦に加え粘性摩擦をも考慮した有効な摩擦補償を行うことで、より効率の良い制御が実現できる可能性も考えられるので、今後さらに研究を進めたい。

本研究では作業に必要な情報を主に力センサーからの情報に頼っていたが、今後より一層高度な作業を実現させるためには、視覚や超音波等の情報を積極的に利用していくことが望まれる。

謝 辞

最後に臨み、本研究に対し終始懇切丁寧なご指導ご鞭撻を賜りました名古屋大学大学院工学研究科マイクロシステム工学専攻 福田 敏男 教授に厚く感謝の意を表します。

また、本研究に有益なご指導とご助言を賜りました名古屋大学大学院工学研究科機械工学専攻 太田 博 教授，ならびに電子機械工学専攻 末松 良一 教授に深く感謝の意を表します。

本研究は、University of Ottawa 大学院在学中に開始し、その後、新潟工業短期大学に奉職して行ったものであります。University of Ottawa 大学院在学中には D.S.Necslescu 教授，新潟工業短期大学奉職中には 下田 茂 学長をはじめ 反町 和則 教授，渡部 清一 教授，坂本 信 助教授，生産システム工学科の教員諸氏に本研究を遂行するに際してのご指導とご援助を賜りました。ここに厚く感謝の意を表します。

また、本論文をまとめる上で、適切なご助言を賜りました新潟大学工学部 新田 勇 助教授に厚く感謝の意を表します。

最後に、本研究を遂行する上で、また論文の作成にあたり、絶えず励まし元気づけてくれた妻 美香，長女 英香，長男 実朋，次女 真晶 に心から感謝します。

参 考 文 献

- [1] J.Kerr and B.Roth, Analysis of Multifingered Hands, The International Journal of Robotic Research, 4-4, pp.3-17, 1986.
- [2] 中村・永井・吉川, 複数のロボット機構による協調的あやつりの力学, 日本ロボット学会誌, 4-5, pp.489-498, 1986.
- [3] 小菅・吉田・福田・酒井・蟹谷・針木, 複数のロボットの分散協調制御, 第11回日本ロボット学会学術講演会予稿集, pp.705-706, 1993.
- [4] S.A.Schneider and R.H.Cannon,Jr. : "Object Impedance Control for Cooperative Manipulation : Theory and Experimental Results", IEEE Transactions on Robotics and Automation, 8-3, pp.383-394, 1992.
- [5] E.Nakano, S.Ozaki, T.Ishida, I.Kato : "Cooperational Control of the Anthro-morphous Manipulator MELARM", Proceedings of 4th International Symposium on Industrial Robots, pp.251-260, 1974.
- [6] 内山・ドシェ, 両手ロボットの対称型運動学と非マスタスレーブ協調制御, 日本ロボット学会誌, 7-1, pp.19-30, 1989.
- [7] A.K.Ramadorai, T.-J.Tarn, A.K.Bejczy, N.Xi : "Task-Driven Control of Multi-Arm Systems", IEEE Transactions on Control Systems Technology, 2-3, pp.198-206, 1994.
- [8] 内山, 北野, 宮脇 : " 双腕ハンドリングロボットの協調制御実験", 日本機械学会論文集, 62-593, C, pp.202-208, 1995.
- [9] L.A.Zadeh : "Fuzzy Set", Information and Control, 8, pp.338-353, 1965.
- [10] E.H.Mamdani : "Application of fuzzy algorithms for control of simple dynamical plants", Proceedings of IEE, 121, pp.1585-1588, 1974.
- [11] 安信, 宮本, 井原 : " 予見Fuzzy制御方式による列車自動運転", システムと制御, 28-10, pp.605-613, 1984.
- [12] 田中編 : インテリジェント制御システム -ファジィ・ニューロ・GA・カオスによる知的制御-, 共立出版, 1996.

- [13] W.S.McCulloch and W.Pitts : "A Logical Calculus of the Idea Immanent in Nervous Activity", Bull. Math. Biophysics, 5, pp.115-133, 1943.
- [14] D.O.Hebb : *The Organization of Behavior*, Wiley, 1949.
- [15] F.Rosenblatt : *Principles of Neurodynamics*, Spartan, 1961.
- [16] M.Minsky and S.Papert : *Perceptrons*, MIT Press, 1969.
- [17] P.J.Werbos : *Beyond Regression : New Tools for Prediction and Analysis in the Behavioral Sciences*, Master's Thesis, Harvard University, 1974.
- [18] D.B.Parker : "Learning-Logic", Invention Report, S81-64, File 1, Office of Technology Licensing, Stanford University, 1982.
- [19] D.E.Rumelhart, G.E.Hinton, R.J.Williams : Learning Internal Representations by Error Propagation. In *Parallel Distributed Processing*, vol.1, pp.318-362, MIT Press, 1986.
- [20] K.Kiguchi : *Robot Contact Motion Control Using Environment Impedance Identification Application of Neural Network*, Master's Thesis, University of Ottawa, 1993.
- [21] 木口・D.S.Necsulescu・福田 : "ニューラルネットワークを利用した未知環境でのハイブリッド制御", 日本ロボット学会誌, 13-2, pp.131-136, 1995.
- [22] T.Yabuta and T.Yamada : "Possibility of Neural Networks Controller for Robot Manipulators", Proceedings of IEEE International Conference on Robotics and Automation, pp. 1686-1691, 1990.
- [23] 山田, 藪田 : "ダイレクトニューロコントローラに関する一考察," 計測自動制御学会論文集, 27-7, pp784-791, 1991.
- [24] T.Yamada and T.Yabuta : "Neural Network Controller Using Autotuning Method for Nonlinear Functions", IEEE Transactions on Neural Networks, 3- 4, pp.595-601, 1992.
- [25] 鶴田, 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 自由度マニピュレータの制御)", 日本ロボット学会誌, 7-1, pp.47-51, 1989.
- [26] 鶴田, 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, 9-1, pp.45-54, 1991.
- [27] 福田, 栗原, 鶴田, 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)", 日本機械学会論文集, 56-527, C, pp.210-216, 1990.

- [28] 福田, 柴田, 鶴田, 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2報)”, 日本機械学会論文集, 57-535, C, pp.866-873, 1991.
- [29] 福田・柴田・鶴田・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, ファジィルールを適用したニューラルネットワーク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, 57-535, C, pp.874-881, 1991.
- [30] 柴田, 福田, 小菅, 新井, 鶴田, 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)”, 日本機械学会論文集, 58-549, C, pp.108-115, 1992.
- [31] T.Shibata, T.Fukuda, K.Kosuge, F.Arai, M.Tokita, T.Mitsuoka : "Hybrid symbolic and neuromorphic control for hierarchical intelligent control", Proceedings IEEE International Conference on Robotics and Automation, pp.2081-2086, 1992.
- [32] M.Kawato, K.Furukawa and R.Suzuki : "A Hierarchical Neural-Network Model for Control and Learning of Voluntary Movement", Biological Cybernetics, 57, pp.169-185, 1987.
- [33] M.Kawato, Y.Uno, M.Isobe and R.Suzuki : "Hierarchical Neural Network Model for Voluntary Movement With Application to Robotics", IEEE Control Systems Magazine, 8, pp.8-16, 1988.
- [34] M.Kawato, Y.Maeda, Y.Uno, R.Suzuki : "Trajectory Formation of Arm Movement by Cascade Neural Network Model Based on Minimum Torque-Change Criterion", Biological Cybernetics, 62, pp.275-288, 1990.
- [35] M.I.Jordan : "Supervised Learning and Systems with Excess Degree of Freedom", COINS Technical Report, 88-27, pp.1-41, 1988.
- [36] 青山, 大松 : “ニューラルネットワークによるセルフチューニングPID制御系の設計”, 電気学会論文集, 116-11, C, pp.1197-1201, 1996.
- [37] S.Kung and J.Hwang : "Neural Network Architectures for Robotic Applications", IEEE Transactions on Robotics and Automation, 5-5, pp.641-657, 1989.
- [38] 堤 : “ニューロコンピューティングによるロボットの軌道生成”, 日本ロボット学会誌, 9-2, pp.89-95, 1991.
- [39] J.Helferty and S.Biswas : "Neuromorphic Control of Robot Manipulators", Proceedings

- of IEEE International Conference on Robotics and Automation, pp.2436-2441, 1991.
- [40] F.L.Lewis, K.Liu, A.Yesildirek : "Neural Net Robot Controller with Guaranteed Tracking Performance", IEEE Transactions on Neural Networks, 6-3, pp.703-715, 1995.
- [41] A.U.Levin and K.S.Narendra : "Control of Nonlinear Systems Using Neural Networks - Part II : Observability, Identification, and Control, IEEE Transactions on Neural Networks, 7-1, pp.30-42, 1996.
- [42] M.H.Raibert and J.J.Craig : "Hybrid Position/Force Control of Manipulators", Trans. ASME Journal of Dynamic Systems, Measurement and Control, 102, pp.126-133, 1981.
- [43] N.Hogan : "Impedance control : An approach to manipulation : Part I - Theory ; Part II - Implementation ; Part III - Applications", ASME Journal of Dynamic Systems, Measurement, and Control, 107, pp.1-24, 1985.
- [44] 吉川 : " ロボットアームの位置と力の動的ハイブリッド制御" , 日本ロボット学会誌, 3-6, 3-6, pp.531-537, 1985.
- [45] T.Yoshikawa, T.Sugie, N.Tanaka : "Dynamic Hybrid Position/Force Control of Robot Manipulators - Controller design and Experiment", IEEE Journal of Robotics and Automation, 4-6, pp.699-705, 1988.
- [46] T.Yoshikawa : "Dynamic Hybrid Position/Force Control of Robot Manipulators - Description of Hand Constrains and Calculation of Joint Driving Force", IEEE Journal of Robotics and Automation, 3-5, pp.386-392, 1987.
- [47] O.Khatib and J.Burdick : "Motion and Force Control of Robot Manipulators", Proceedings of IEEE International Conference on Robotics and Automation, pp.1381-1386, 1986.
- [48] O.Khatib : "A Unified Approach to Motion and Force Control of Robot Manipulators : The Operational Space Formulation", IEEE Journal Robotics and Automation, 3-1, pp.43-53, 1987.
- [49] S.Eppinger and W.Seering : "On Dynamic Models of Robot Force Control", Proceedings of IEEE International Conference on Robotics and Automaiton, pp.29-34, 1986.
- [50] R.Volpe and P.Khosla : "A Theoretical and Experimental Investigation of Explicit Force Control Strategies for Manipulators", IEEE Transactions on Automatic Control, 38-11, pp.1634-1650, 1993.

- [51] D.E.Whitney : "Historical Perspective and State of the Art of Robot Force Control", The International Journal Robotics Research, 6-1, pp.3-14, 1987.
- [52] S.Payandeh, A.A.Goldenberg : "A Robot Controller : Theory and Experiments", Proceedings of IEEE International Conference on Robotics and Automaiton, pp.36-41, 1991.
- [53] S.Komada, K.Nomura, M.Ishida, K.Ohnishi, T.Hori : "Robust Force Control Based on Estimation of Environment", Proceedings of IEEE International Conference on Robotics and Automaiton, pp.1362-1367, 1992.
- [54] M.Cohen and T.Flash : "Learning Impedance Parameters for Robot Control Using an Assosiative Search Network", IEEE Transactions on Robotics and Automation, 7-3, pp.382-390, 1991.
- [55] 辻, 西田, 伊藤 : "ニューラルネットによるインピーダンスパラメータの繰返し学習", 計測自動制御学会論文集, 27-7, pp.1461-1468, 1991.
- [56] Z.Lu and A.A.Goldenberg : "Robot Impedance Control and Force Regulation : Theory and Experiments," The International Journal of Robotics Research, 14-3, pp.225-254, 1995.
- [57] Z.Peng and N.Adachi : "Position and Force Control of Manipulators without Using Force Sensors", JSME International Journal, Ser.III, 35-2, pp.252-258, 1992.
- [58] T.Yoshikawa and A.Sudou : "Dynamic Hybrid Position/Force Control of Robot Manipulators - On-Line Estimation of Unknown Constraint", IEEE Transactions on Robotics and Automation, 9-2, pp.220-226, 1993.
- [59] D.Bossert, U.Ly, J.Vagners : "Experimental Evaluation of a Hybrid Position and Force Surface Following Algorithm for Unknown Surfaces", Proceedings of IEEE International Conference on Robotics and Automation, pp.2252-2257, 1996.
- [60] F.Hsu and L.Fu : "Adaptive Fuzzy Hybrid Force/Position Control for Robot Manipulators Following Contours of an Uncertain Object", Proceedings of IEEE International Conference on Robotics and Automation, pp.2232-2237, 1996.
- [61] 菅野 : ファジィ制御, 日刊工業新聞社, 1988.
- [62] 下島・福田・新井 : " R B F 型ファジィ推論とその応用", 第11回日本ロボット学会学術講演会予稿集, pp.131-132, 1993.

- [63] S.W.Kim, M.Park : "Fuzzy Compliance Robot Control Using Multi Rule-Base", Proceedings of IEEE International Conference on Robotics and Automation, pp.1343-1348, 1992.
- [64] 林・馬野 : " ファジィ・ニューラルネットワークの現状と展望", 日本ファジィ学会誌, 5-2, pp.178-190, 1993.
- [65] D.Whitley, S.Dominic, R.Das : "Genetic Reinforcement Learning with Multilayer Neural Networks", Proceedings of the Fourth International Conference Genetic Algorithm, pp.562-569, 1991.
- [66] R.A.Jacobs : " Increased Rates of Convergen-ce Through Learning Rate Adaptation", Neural Networks, 1, pp.295-307, 1988.
- [67] P.Arabshahi, J.J.Choi, R.J.Marks II, T.P.Caudell : "Fuzzy Control of Backpropagation", Proceedings of IEEE International Conference on Fuzzy Systems, pp.967-972, 1992.
- [68] B. Yang and H. Asada : "Hybrid linguistic/numeric control of deburring robots based on human skills", Proceedings of IEEE International Conference on Robotics and Automation, pp.1467-1474, 1992.
- [69] S. Liu and H. Asada : "Transfer of human skills to robots : learning from human demonstrations for building an adaptive control system", Proceedings of 1993 American Control Conference, pp.1414-1417, 1993.
- [70] Y. Xu and J. Yang : "Towards human-robot coordination : skill modeling and transferring via hidden markov model", Proceedings IEEE International Conference on Robotics and Automation, pp.1906-1911, 1995.
- [71] K.Kiguchi and T.Fukuda : "Fuzzy Neural Hybrid Position/Force Control for Robot Manipulators", Journal of Applied Mathematics and Computer Science, 6-3, pp.101-121, 1996.
- [72] 木口, 福田 : "ファジィ・ニューラルネットワークを利用した未知環境に対するハイブリッド制御", 電気学会論文集, 116-7, C, pp.785-793, 1996.
- [73] 木口, 福田 : "ファジィ・ニューロを利用したロボットマニピュレータのインテリジェント位置/力制御", 日本機械学会論文集, C編 (投稿中)
- [74] 木口, 福田 : "ファジィ・ニューラルネットワークを利用した複腕ロボットによる

- 未知対象物の制御”, 日本機械学会論文集, 61-589, C, pp.3643-3649, 1995.
- [75] 木口, 福田: “ファジィ・ニューラルネットワークを利用した対象物の操り”, 日本機械学会ロボティクス・メカトロニクス講演会'95講演論文集 (B), pp.897-900, 1995.
- [76] K.Kiguchi and T.Fukuda: "Object Manipulation Under Disturbance Using Fuzzy-Neural Network", Mathematics and Computers in Simulation (Trans. of IMACS), 41-3&4, pp.235-244, 1996.
- [77] 木口, 福田: “ファジィ・ニューロを利用した複腕ロボットによる摩擦補償を考慮した対象物の滑らせ移動制御”, 日本機械学会論文集, 62-595, C, pp.997-1004, 1996.
- [78] K.Kiguchi and T.Fukuda: "Intelligent Position/Force Controller for Industrial Robot Manipulators Application of Fuzzy Neural Networks", IEEE Transaction on Industrial Electronics. (投稿中)
- [79] 木口, 福田: “ロボットマニピュレータのファジィ・ニューロ力制御”, 第12回日本ロボット学会学術講演会予稿集, pp.929-930, 1994.
- [80] 木口: “ファジィ・ニューラルネットワークを利用したロボットマニピュレータ力制御”, 新潟工業短期大学研究紀要, 14, pp.3-8, 1995.
- [81] K.Kiguchi and T.Fukuda: "Fuzzy Neural Controller for Robot Manipulator Force Control", Proceedings of the Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and the Second International Fuzzy Engineering Symposium (FUZZ-IEEE/IFES'95), vol.2, pp.869-874, 1995.
- [82] K.Kiguchi and T.Fukuda: "Robot Manipulator Contact Force Control Application of Fuzzy-Neural Network", Proceedings of IEEE International Conference on Robotics and Automation, vol.1, pp.875-880, 1995.
- [83] 木口, 福田: “ファジィ・ニューロを利用した位置/力制御での摩擦補償”, 第13回日本ロボット学会学術講演会予稿集, pp.1071-1072, 1995.
- [84] K.Kiguchi and T.Fukuda: "Fuzzy Neural Friction Compensation Method of Robot Manipulation During Position/Force Control", Proceedings of IEEE International Conference on Robotics and Automation, vol.1, pp.372-377, 1996.
- [85] 木口, 福田: “ファジィ・ニューロ制御器の効果的学習”, 日本機械学会北陸信越

- 支部第33期総会・講演会講演論文集, pp.301-302, 1996.
- [86] K.Kiguchi and T.Fukuda : "Robot Manipulator Force Control for an Unknown Environment Using Fuzzy Vector Method", Proceedings of IEEE International Conference on Robotics and Automation, 1997. (投稿中)
- [87] 木口, 福田 : "ファジィ・ニューロを利用したロボットマニピュレータのインテリジェント力制御", 日本機械学会ロボティクス・メカトロニクス講演会'96講演論文集 (A) , pp.5-8, 1996.
- [88] K.Kiguchi and T.Fukuda : "Hybrid Control of Robot Manipulator for an Unknown Environment Using Fuzzy Neural Networks", Proceedings of IMACS/IEEE CESA'96 International Multiconference, pp.777-782, 1996.
- [89] B.Widrow : "Generalization and information storage in network of adaline 'neurons' ", Self-Organizing Systems - 1962, Spartan Books, pp.435-462, 1962.
- [90] M.R.Popovic, D.M.Gorinevsky, A.A.Goldenberg : "Fuzzy Logic Controller for Accurate Positioning of Direct-Drive Mechanism Using Force Pulses", Proceedings of IEEE International Conference on Robotics and Automation, vol.1, pp.1166-1171, 1995.
- [91] T.Fukuda and T.Shibata : "Neural Network Application for Robotic Motion Control", Journal of Robotics and Mechatronics, 2-3, pp.21-25, 1991.
- [92] K.Kiguchi and D.S.Necsulescu : "Control of Multi-DOF Robots Using Neural Networks", Proceedings of the Knowledge-Based Systems & Robotics Workshop, pp.747-754, 1993.
- [93] K.Funahashi : "On the Approximate Realization of Continuous Mappings by Neural Networks", Neural Networks, 2, pp.183-192, 1989.
- [94] 小川・山崎 : " 過学習の理論", 電子情報通信学会論文誌 D-II, vol.J76-D-II, 6, pp.1280-1288, 1993.
- [95] G.Deco, W.Finnoff, H.G.Zimmermann : "Unsupervised Mutual Information Criterion for Elimination of Overtraining in Supervised Multilayer Networks", Neural Computation, 7-1, pp.86-107, 1995.
- [96] 堀川・古橋・大熊・内川 : " ニューラルネットワークによる学習型ファジィ制御器", 計測自動制御学会論文集, 27-2, pp.208-215, 1991.

- [97] 湯・渡辺・中村・古賀：“ファジィ・ガウシアン・ニューラルネットワーク制御器とその移動ロボット車の制御への応用”，日本機械学会論文集，59-564，C，pp.2290-2297，1993.
- [98] 永井・吉川：“多指ハンドの適応制御”，日本ロボット学会誌，11-1，pp.111-120，1993.
- [99] W.Pedrycz, *Fuzzy Set Engineering*, CRC Press, 1995.
- [100] I.Hiraga, T.Furuhashi, Y.Uchikawa, S.Nakayama：“An acquisition of operator's rules for collision avoidance using fuzzy neural networks”，IEEE Transactions on Fuzzy Systems，3-3，pp.280-287，1995.
- [101] D.E.Whitney, A.C.Edsall, A.B.Todtenkopf, T.R.Kurfess, A.R.Tate：“Development and control of an automated robotic weld bead grinding system”，Transactions ASME Journal of Dynamic Systems, Measurement, and Control，112，pp.166-176，1990.
- [102] 下倉，浅田，劉：“バリ取り作業におけるヒューマンスキルの教示方法に関する検討”，日本ロボット学会誌，14-1，pp.144-149，1996.
- [103] J.Y.S.Luh, M.W.Walker, R.P.C.Paul：“Resolved-Acceleration Control of Mechanical Manipulator”，IEEE Transactions on Automatic Control，25-3，pp.468-474，1980.
- [104] K.Tanaka：“Stability and Stabilizability of Fuzzy-Neural-Linear Control Systems”，IEEE Transactions on Fuzzy Systems，3-4，pp.438-447，1995.
- [105] K.Tanaka：“An Approach to Stability Criteria of Neural-Network Control Systems”，IEEE Transactions on Neural Networks，7-3，pp.629-642，1996.
- [106] S.Kuntanapreeda and R.R.Fullmer：“A Training Rule Which Guarantees Finite-Region Stability for a Class of Closed-Loop Neural-Network Control Systems”，IEEE Transactions on Neural Networks，7-3，pp.745-751，1996.
- [107] S.G.Cao, N.W.Rees, and G.Feng：“Stability Analysis of Fuzzy Control Systems”，IEEE Transactions on Systems, Man, and Cybernetics - Part B : Cybernetics，26-1，pp.201-204，1996.
- [108] B.Chen and L.L.Hoberock：“Machine Vision Recognition of Fuzzy Objects Using a New Fuzzy Neural Network”，Proceedings of IEEE International Conference on Robotics and Automation，vol.2，pp.1596-1601，1996.

[109] C.H.Tsai, J.S.Liu, and W.S.Lin : "A Neuro-Fuzzy Logic Controller for Trajectory Tracking of Uncertain Robots", Proceedings of IEEE International Conference on Robotics and Automation, vol.2, pp.1929-1934, 1996.

論文題目	発表学会誌名	巻号
1. 論文題目		
2. 論文題目		
3. 論文題目		
4. 論文題目		
5. 論文題目		
6. 論文題目		
7. 論文題目		
8. 論文題目		
9. 論文題目		
10. 論文題目		
11. 論文題目		
12. 論文題目		
13. 論文題目		
14. 論文題目		
15. 論文題目		
16. 論文題目		
17. 論文題目		
18. 論文題目		
19. 論文題目		
20. 論文題目		
21. 論文題目		
22. 論文題目		
23. 論文題目		
24. 論文題目		
25. 論文題目		
26. 論文題目		
27. 論文題目		
28. 論文題目		
29. 論文題目		
30. 論文題目		
31. 論文題目		
32. 論文題目		
33. 論文題目		
34. 論文題目		
35. 論文題目		
36. 論文題目		
37. 論文題目		
38. 論文題目		
39. 論文題目		
40. 論文題目		
41. 論文題目		
42. 論文題目		
43. 論文題目		
44. 論文題目		
45. 論文題目		
46. 論文題目		
47. 論文題目		
48. 論文題目		
49. 論文題目		
50. 論文題目		
51. 論文題目		
52. 論文題目		
53. 論文題目		
54. 論文題目		
55. 論文題目		
56. 論文題目		
57. 論文題目		
58. 論文題目		
59. 論文題目		
60. 論文題目		
61. 論文題目		
62. 論文題目		
63. 論文題目		
64. 論文題目		
65. 論文題目		
66. 論文題目		
67. 論文題目		
68. 論文題目		
69. 論文題目		
70. 論文題目		
71. 論文題目		
72. 論文題目		
73. 論文題目		
74. 論文題目		
75. 論文題目		
76. 論文題目		
77. 論文題目		
78. 論文題目		
79. 論文題目		
80. 論文題目		
81. 論文題目		
82. 論文題目		
83. 論文題目		
84. 論文題目		
85. 論文題目		
86. 論文題目		
87. 論文題目		
88. 論文題目		
89. 論文題目		
90. 論文題目		
91. 論文題目		
92. 論文題目		
93. 論文題目		
94. 論文題目		
95. 論文題目		
96. 論文題目		
97. 論文題目		
98. 論文題目		
99. 論文題目		
100. 論文題目		

本研究に関する論文等

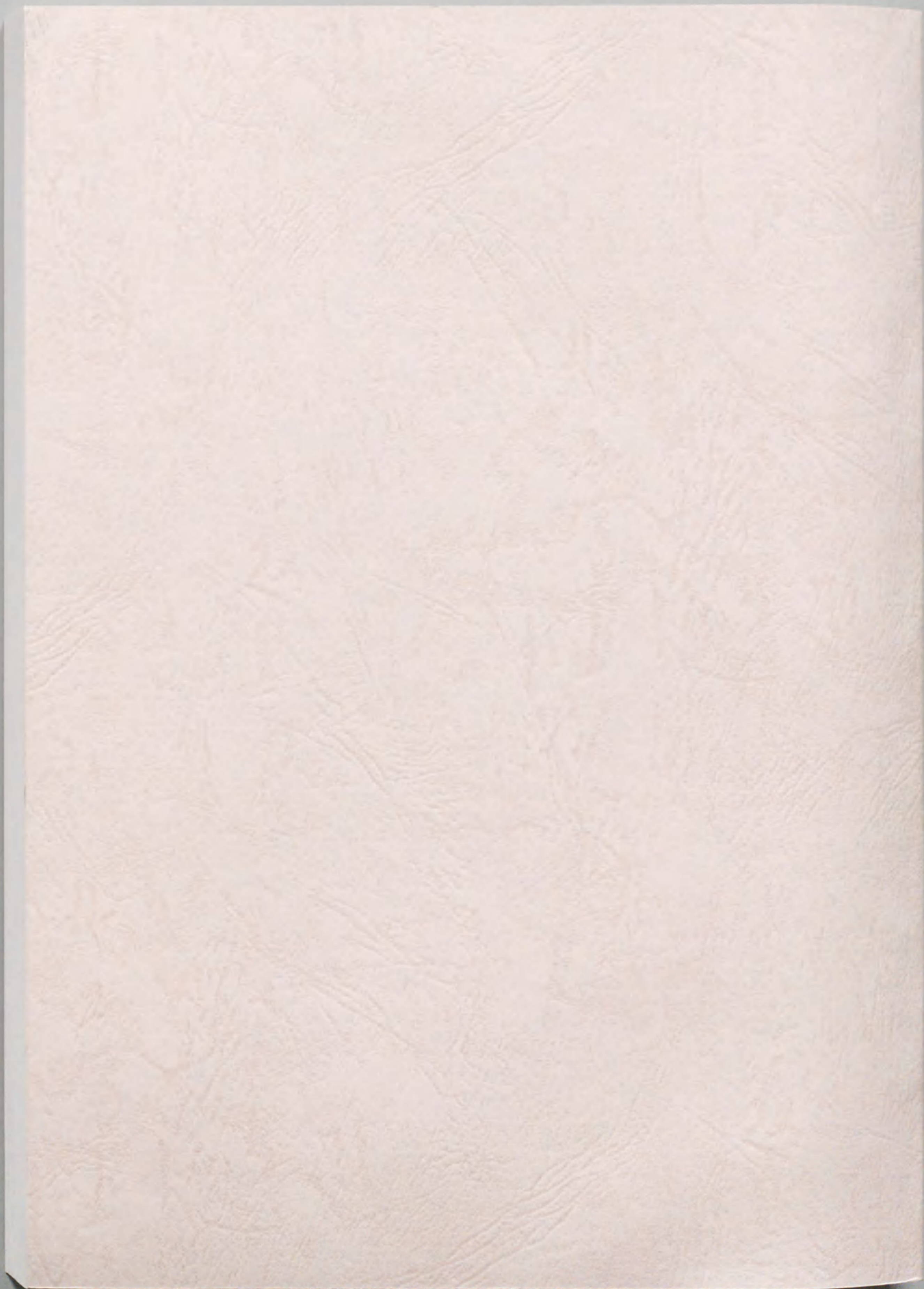
論文題目	公表の方法及び時期	著者
I 学会誌等		
1) ニューラルネットワークを利用した未知環境でのハイブリッド制御	日本ロボット学会誌, 13巻2号, pp.131-136 (1995)	木口量夫 D.S.Necsulescu 福田敏男
2) ファジィ・ニューラルネットワークを利用した複腕ロボットによる未知対象物の制御	日本機械学会論文集C編, 61巻589号, pp.3643-3649 (1995)	木口量夫 福田敏男
3) ファジィ・ニューロを利用した複腕ロボットによる摩擦補償を考慮した対象物の滑らせ移動制御	日本機械学会論文集C編, 62巻595号, pp.997-1004 (1996)	木口量夫 福田敏男
4) Object Manipulation Under Disturbance Using Fuzzy-Neural Network	Mathematics and Computers in Simulation (Trans. of IMACS), vol.41, pp.235-244 (1996)	Kazuo KIGUCHI Toshio FUKUDA
5) ファジィ・ニューラルネットワークを利用した未知環境に対するハイブリッド制御	電気学会論文集C編, 116巻7号, pp.785-793 (1996)	木口量夫 福田敏男
6) Fuzzy Neural Hybrid Position/Force Control for Robot Manipulators	Journal of Applied Mathematics and Computer Science, vol.6, no.3, pp.101-121, (1996)	Kazuo KIGUCHI Toshio FUKUDA
7) Intelligent Position/Force Controller for Industrial Robot Manipulators Application of Fuzzy Neural Networks	IEEE Transaction on Industrial Electronics (論文投稿中)	Kazuo KIGUCHI Toshio FUKUDA
8) ファジィ・ニューロを利用したロボットマニピュレータのインテリジェント位置/力制御	日本機械学会論文集C編 (論文投稿中)	木口量夫 福田敏男

本研究に関する論文等

論文題目	公表の方法及び時期	著者
II 国際会議等		
1) Control of Multi-DOF Robots Using Neural Networks	Proceedings of the Knowledge-Based Systems & Robotics Workshop, pp.747-754 (1993)	Kazuo KIGUCHI Dan S.NECSULESCU
2) Fuzzy Neural Controller for Robot Manipulator Force Control	Proceedings of the Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and the Second International Fuzzy Engineering Symposium (FUZZ-IEEE/IFES'95), vol.2, pp.869-874 (1995)	Kazuo KIGUCHI Toshio FUKUDA
3) Robot Manipulator Contact Force Control Application of Fuzzy-Neural Network	Proceedings of IEEE International Conference on Robotics and Automation, vol.1, pp.875-880 (1995)	Kazuo KIGUCHI Toshio FUKUDA
4) Fuzzy Neural Friction Compensation Method of Robot Manipulation During Position/Force Control	Proceedings of IEEE International Conference on Robotics and Automation, vol.1, pp.372-377 (1996)	Kazuo KIGUCHI Toshio FUKUDA
5) Hybrid Control of Robot Manipulator for an Unknown Environment Using Fuzzy Neural Networks	Proceedings of IMACS/IEEE CESA'96 International Multi-conference, pp.777-782 (1996)	Kazuo KIGUCHI Toshio FUKUDA
6) Robot Manipulator Force Control for an Unknown Environment Using Fuzzy Vector Method	Proceedings of IEEE International Conference on Robotics and Automation (論文投稿中)	Kazuo KIGUCHI Toshio FUKUDA

本研究に関する論文等

論文題目	公表の方法及び時期	著者
III その他		
1) ロボットマニピュレータのファジィ・ニューロ力制御	第12回日本ロボット学会学術講演会予稿集, pp.929-930 (1994)	木口量夫 福田敏男
2) ファジィ・ニューラルネットワークを利用したロボットマニピュレータ力制御	新潟工業短期大学研究紀要, 第14号, pp.3-8 (1995)	木口量夫
3) 未知物体に対するロボットマニピュレータの力制御	日本機械学会北陸信越支部第32期総会・講演会講演論文集, pp.213-214 (1995)	木口量夫 福田敏男
4) ファジィ・ニューラルネットワークを利用した対象物の操り	日本機械学会ロボティクス・メカトロニクス講演会'95講演論文集(B), pp.897-900, (1995)	木口量夫 福田敏男
5) ファジィ・ニューロを利用した位置/力制御での摩擦補償	第13回日本ロボット学会学術講演会予稿集, pp.1071-1072, (1995)	木口量夫 福田敏男
6) ファジィ・ニューロ制御器の効果的学習	日本機械学会北陸信越支部第33期総会・講演会講演論文集, pp.301-302 (1996)	木口量夫 福田敏男
7) ファジィ・ニューロを利用したロボットマニピュレータのインテリジェント力制御	日本機械学会ロボティクス・メカトロニクス講演会'96講演論文集(A), pp.5-8 (1996)	木口量夫 福田敏男
8) 目標力の違いを考慮したロボットマニピュレータのインテリジェント力制御	第14回日本ロボット学会学術講演会予稿集, pp.37-38 (1996)	木口量夫 福田敏男



Inches 1 2 3 4 5 6 7 8
cm 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

Kodak Color Control Patches

© Kodak, 2007 TM: Kodak

Blue Cyan Green Yellow Red Magenta White 3/Color Black



Kodak Gray Scale



© Kodak, 2007 TM: Kodak

A 1 2 3 4 5 6 M 8 9 10 11 12 13 14 15 B 17 18 19

