学位論文

見まねによる運動学習に関する研究

1998年11月

宮本 弘之

論文要旨

近年，ロボット工学は機構学，動力学，制御理論などの導入と発展とともに格段の進歩をとげてきている．最近では，多自由度をもつマニピュレータや，高い運動性をもつものなどが出現してきた．それらのロボットに高精度で高速な運動を行わせることは可能になってきているものの，変動する外部環境との相互作用をともなう熟練作業を行わせることは，いまだに解決の困難な課題として残されている．本論文は，このような課題の解決と，脳の仕組みの解明のための道具となることを目指した，見まねによる運動学習ロボットに関する研究をまとめたものである．

本研究では，脳の運動制御の計算理論のひとつである最適化原理に基づく運動パターンの認識の枠組を用いて，見まねによる運動学習を試みている．そのひとつの応用例として，運動パターンの抽象的な表現である経由点を用いたタスクレベル学習を行なっている．経由点は一種の情報圧縮とみなすこともできる．本論文で述べられている見まねによる学習は一般的には次のように要約できる．(1) ヒトのデモンストレーションが教師情報，(2) 制御対象のダイナミクスと最適化原理に基づいて運動パターンを認知する，(3) 神経回路モデルを用いて運動パターンを再構成できる経由点を抽出する，(4) 固定した軌道計画と制御のスキームを用いて経由点を制御変数として扱う，(5) タスクが実現されるように経由点の位置と時間を修正する．

本論文の前半では，けん玉の運動学習ロボットに関して述べている．けん玉はダイナミックな運動で，しかも比較的簡単なタスク表現が可能となる運動である．見まねによる学習の戦略に基づき，経由点の抽出と学習を完全に自動的に行ない，実機 (SARCOS Dextrous slave arm) を用いて，けん玉のフィードフォーワード制御に成功している．さらに抽出された経由点がタスク表現として適切であるかどうかも計算機シミュレーションにより確かめている．

本論文の後半では，本研究で扱う見まねによる学習の枠組が汎用性をもち得るかどうかを確かめるために，連続する複数の動作が含まれる階層的な運動系列の学習課題や，非線形性の強い制御対象を扱う学習課題の場合など，種々の運動における学習可能性を調べている．その結果，連続運動への拡張 (テニスサーブ) と，動作や環境の変動への適応 (振り子の振り上げ) に関して，見まねによる学習の枠組が有用であることを示した．この研究により運動の制御に他者の運動パターンを観測することが非常に重要であることを示していると同時に，他者の運動パターンの認知に運動制御の神経回路が積極的に用いられていることを示した．

目次

第 1 章

序章

1.1 はじめに

近年，ロボット工学は機構学，動力学，制御理論などの導入と発展とともに格段の進歩を遂げてきている．最近では，多自由度をもつマニピュレータや，高い運動性をもつものなどが出現してきた．それらのロボットに高精度で高速な運動を行わせることは可能になってきているものの，変動する外部環境との相互作用をともなう熟練作業を行わせることは，いまだに解決の困難な課題として残されている．

我々が日常生活の中で何気なく行っている動作は，脳の複雑な情報処理機構や運動制御機構の高度な連係のもとで遂行されている．例えば，テーブルの上の紙コップに手をのばして掴み，中の水をこぼさないように運んで，トレーの穴に差し込む，といったことは人間にとってはなんでもない簡単な作業である．ところが同じことを現在のロボットにやらせようとすれば大変な準備が必要である．上に挙げた例では，安定把握のために操作対象物の形状認識や，指先と対象物との接触面の決定，指先のプリシェイピング，等がまず行われなければならない．さらに，紙コップを握りつぶしてしまわないように，触覚あるいは力センサを用いて力のフィードバックのはいった制御が必要であるし，なめらかな移動のための軌道計画や，ロボット自体だけでなく対象物との相関までも含めたダイナミクスの考慮も必要である．

工場などで使われている産業用ロボットに目的の作業を遂行させるとき，直接教示やティーチングプレイバック等の教示方式が広く用いられている．工場内などの種々の条件の整った作業環境のもとで，限定された繰り返し作業をロボットに行わせるときには，それらの教示方式は有効であろう．しかし，外部環境の変動や不確定性に対処することに関しては，それらの教示方式だけではまだ不十分と考えられる．Teaching by showing (learning by watching) は，それらの困難の解決に有効であると期待される [33, 45, 46, 44, 47, 43].

本章では，まず現在試みられているロボット教示のためのいくつかの研究を概説し，つぎに，双方向性理論，見まねによる運動学習の枠組を簡単に解説する．

1.2 最近のロボット教示の試み

産業用ロボットに広く使われているティーチングプレイバックでは，PTP(Point To Point) や CP(Continuous Path) などの軌道の指定の方法を用いて，マニピュレータの先端の三次元位置の軌道，あるいは各関節の角度の軌道を決めている．そのようにして得られた目標関節角度の軌道をロボットの各関節に与え，位置や速度のフィードバック制御を行うことで高精度で高速な動作を可能としている．教示の方式としては，直接教示，ティーチングボックス，マスタスレーブ，ジョイスティック，ポインタ等があり，単純な繰り返し作業には極めて有効であるが，これらの方式では教示の手間が比較的大きい [69]．最近の組み立てロボットには多品種小量生産に対応するためにフレキシビリティが要求されてきているが [70]，その実現のためには，より簡単で手間のかからない教示方法が望まれる．

1.2.1 AI 的な手法に基づくアプローチ

タスクレベルのロボット言語を用いた教示では，タスクレベルのコマンドから実際のロボットに与えるべき動作コマンドへの変換が非常に難しい．最近，人間の作業動作を例示して，それをもとに作業遂行のためのロボットプログラムを自動生成する試みが数多く行われている．ロボットに与えるコマンド列を自動的に生成するためにマスターアームで記録した教示者の動作データをシンボリックに記述し，人間の動作をパラメータのセット (state, movement, rotation 等) に変換する試み [80] や，教示者の例示を撮影したカメラ画像を用いてマウスなどのポインタにより直観的なプログラミングを目指したもの [76] などが報告されている．

最近，人間が実演して見せた組み立て作業を視覚認識することにより動作列を抽出し，それをもとにロボットに与えるプログラムを自動生成するいくつかの興味深い研究が行われている [27, 45]．これらの研究では，人間の組み立て作業をカメラにより連続的に観察し，画像の時間差分から対象物の状態変化を検出してセグメンテーションを行い，作業に必要な要素的な動作をシンボリックな記述に変換する．記録された位置や姿勢などの軌道をそのまま再生するシステムでは，対象物の初期状態の違いや，人間とマニピュレータの機構の違いなどに自動的に対処することは非常に難しい．観察された動作を pick, place, transfer などのシンボリックな記述に変換することの利点は，それらの差異を吸収して，目的の作業を正しく遂行できることである [43]．これらの研究では操作対象物として積木ブロックや棒などの比較的取り扱いやすい形状のものが使われているが，より複雑な形状の対象物体を操作するためには，把握点や手先の形状認識が必要となる [33]．

1.2.2 動力学に基づくアプローチ

ロボット教示のための自動プログラミングへのアプローチはAI 的な手法に基づくものが多いが，対象物を操作するとき，質量や接触による反力などが大きく影響するような作業を取り扱う場合には，ロボットの動力学に基づくアプローチも重要となる．ペンで字を書くことや，組み立てなどの多くの手作業では，手先が対象物の表面に拘束されて運動する．ティーチングプレイバック方式では，位置制御のみを考慮した位置データの教示であるため，それらのような外部環境との相互作用をともなう作業では，力やコンプライアンスの調整が極めて困難である．人間は特定の作業を習得するとき，その作業を繰り返し行い，位置や速度，さらには力かげんの軌道までを微妙に調整することによって，より巧みな動作を行うことができるようになる．この習得の過程で，人間は複雑な作業のスキルを無意識のうちに獲得するので，教示者がロボットのコマンド列や制御パラメータとして陽に記述するのは現在のところ非常に困難である．直接教示やオフラインプログラミングなどの従来の方法では，熟練した人間の手作業をロボットで再現するのは不可能ではないかもしれないが，大変な手間とコストが要求されると考えられる．

位置と力のハイブリッド制御などの制御原理がタスク理解のために重要な一面をもっている[2]．位置制御と力制御を組み合わせた制御系を構成し，作業者の動作の変位と力の両方を計測して目標値として与え，柔軟なプレイバック制御を達成したり，インピーダンス同定法による動的力制御の教示方式により位置のデータと力のデータの間に関数関係をあてはめたインピーダンス制御則を用いて，熟練作業者の示したグラインダーがけ等の高度な技能をロボットに行わせることも可能となった[3]．

最近，仮想現実空間を遠隔操作に応用しょうとする興味深い研究も行われている．マスタースレーブ制御方式の遠隔操作では，より高精度な操作を可能とするためには撃力などのきめ細かい情報が操作者にフィードバックされることが望ましい．マスターアームにアクチュエータを組み込めば，操作対象物とマニピュレータの手先の間の静的な反力を操作者にフィードバックすることは比較的容易であるが，撃力を再現することは困難である．インピーダンス制御されるマニピュレータの先端に接触面を表現するための小さい効果器をとりつけ，マスターアームで実時間に測定される人間の腕の動きに効果器の位置と向きを追従させることによって仮想接触空間を作る研究が報告されている[79]．

1.2.3 生物に学んだアプローチ

脳の運動学習の計算論的研究はここ20年間で大きく前進した([20, 29, 30, 40])．これら，脳のメカニズムを探る一連の研究から，最近興味深い知見が得られている．人間の腕の動作を

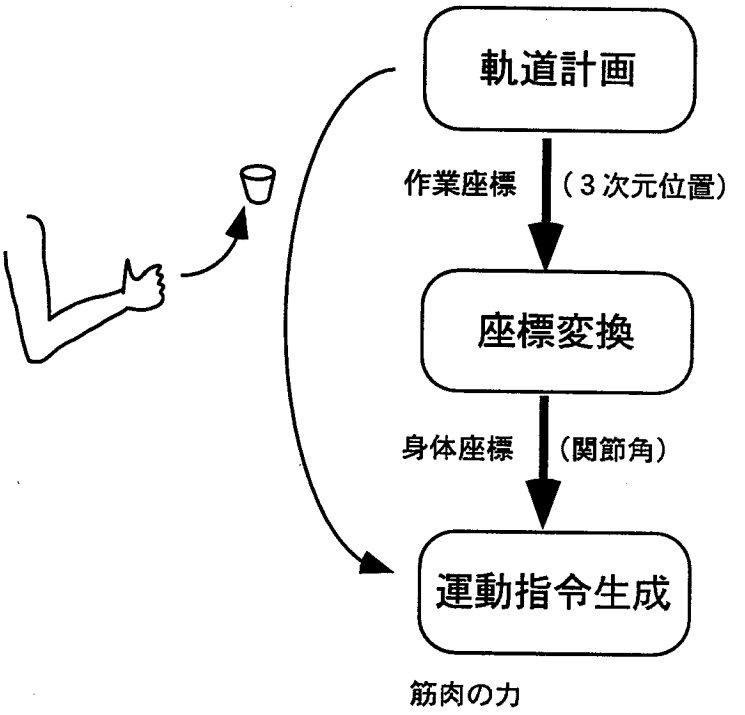


図 1.1: 腕の到達運動における三つの基本的計算問題.

サルに見せたときや, サル自身の手を動かしたとき，特定の動作に反応する神経細胞が上側頭溝多種感覚領野に発見されている [72]．また, サルが実際に動作をしなくても実験者の動作を見るだけでサルの前運動野のニューロンの発火が活発になることが報告されている [10]．さらに, ヒトが物体の把持運動を観察したり, それを行うことを想像しただけで，実際に手を動かさなくても脳の運動に関連する部位が活動することが観察されている [9]．これらの神経生理学のデータは，運動を学習する際，運動の観察やイメージトレーニングが重要であることを示している．

これまで概観してきたように，ロボット工学の研究では，単純な軌道追従の問題から，より高次なタスクレベルで問題を扱う研究が精力的に進められ，着実な成果を挙げている．それでも，ダイナミクスを含めた外部環境との相対的な変化に自動的に対処するシステムを構築するにはまだ十分とはいえない．軌道追従などの問題には，すでに外部環境との相互作用も考慮に入れた学習制御が提案されているが [68]，目的の作業を成功させることに重点を置いたタスクレベルでの適応制御，あるいは学習制御への取り組みはまだ始まったばかりである [63]．

1.3 運動制御の計算理論

ヒトの腕の目標到達運動の制御問題は，図 1.1 に示すように軌道計画，外部座標から身体座標への座標変換，運動指令生成の三つの基本的計算問題に分けることができる [50]．これらの基本的計算問題は図 1.2 に示すように一意に解が決まらないという不良設定性を持つ．まず軌

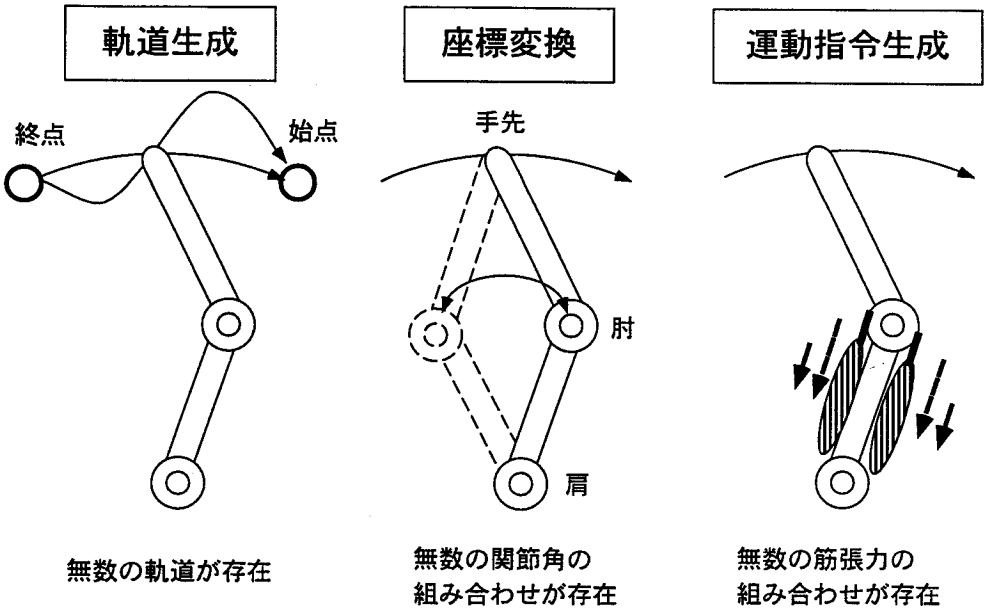


図 1.2: 三つの基本的計算問題の不良設定性.

道計画では，外部座標 (視覚の三次元座標) で手先の目標軌道が計算されなければならない．このとき，始点から終点に至るまでの軌道は無数に存在するが，それらのうちひとつを決定しなければならない．つぎに座標変換では，三次元座標である外部座標で計画された目標軌道から各関節の関節角や筋肉の長さといった内部座標に変換される必要がある．このとき，なんらかの拘束条件を与えなければ，腕全体の姿勢は一意に決まらない．なぜなら，人間の腕は 7 自由度を持ち冗長性があるので，手先の位置と姿勢 (6 自由度) を決めても，肘の位置をどのようにでもできるからである．最後に，関節角などの内部座標の目標軌道から筋肉の力などの運動指令への変換が行なわれる．人間の四肢は主働筋と拮抗筋の両方の働きで動く．従って，目標の関節角を実現するためにこれらの筋肉が出すべき力は，どのような組合せにでもできる．これらの基本的計算問題の最終目標は，運動の到達目標の空間的特徴を適切な筋肉の活動パターンに変換することである．このとき，脳は上にあげたような一意に解が決まらないという不良設定性を実時間で解決しなければならない．

過去 10 年以上にわたって，運動制御の計算論的研究はこれらの三つの基本的計算問題に関して大きく前進した [40]．例えば運動指令生成に関しては以下のことが分かってきた．生物のフィードバックループは時間遅れが大きくゲインが小さい．したがって，高速で協調的な腕の運動は，前向き制御によって実行されなければならない．前向き制御のための内部モデルの理論的な研究が行なわれてきたが [37, 56, 34]，近年，心理物理実験や生理学実験により実証されつつある．最近，運動中のダイナミックなスティフネスはそれほど大きくないことが分かってきた [7, 6, 22]．したがって，逆ダイナミクスモデルなどの内部モデルが必要となる．単一のプルキン

エ細胞の活動の分析から，小脳に逆ダイナミクスモデルが存在することが示唆されている [41, 77].

1.4 軌道計画の計算モデル

この節では軌道計画に関して概説する．図 1.3 に小池ら [42] によって計測されたヒトの 2 点間運動のデータを示す．このデータは肩の高さの水平面内に制限された肩と肘の 2 自由度の場合の腕の運動軌道の例である．被験者には例えば T2 から T6 まで自然に手先を動かすように指示してある．図 1.3 で示されるように，ある点からほかの点までの腕の多関節運動の軌道はほぼまっすぐな手先の軌跡とベル型の速度プロファイルで特徴づけられる [64]．これらの不変な特徴を説明する為にこれまでにキネマティックな最適化原理 (躍度最小軌道仮説) とダイナミックな最適化原理 (トルク変化最小軌道仮説) が提案されてきた [17, 18, 15, 16, 81, 82].

1.4.1 躍度最小モデル

ヒトの腕の運動軌道の不変的な特徴を予測し再現することのできるモデルとして Flash & Hogan(1985) は躍度最小モデルを提案した．肩の高さの水平面内に限定された手先の位置を直交座標系で (x, y) と表す．このとき，位置の躍度 (3 回微分) の 2 乗和の運動時間全体にわたる積分を評価関数

$$C_{\mathrm{J}} = \frac{1}{2}\int_0^{t_f} \left\{ \left(\frac{d^3x}{dt^3}\right)^2 + \left(\frac{d^3y}{dt^3}\right)^2 \right\} \tag{1.1}$$

と定める．躍度最小モデルはこの評価関数を最小にするような軌道が計画されているとするモデルであり，2 点間や経由点を通る運動軌道を良く予測，再現することができる．式 (1.1) は解析的に解くことができて，(x, y) は時間 t に関する 5 次方程式によって表されることが示されている [17, 18, 15, 16].

1.4.2 トルク変化最小モデル

躍度最小モデルでは視覚座標のキネマティクスのみで評価関数が与えられるので，運動時間と始点，終点，経由点が与えられれば運動軌道は時間の 5 次多項式として一意に決定される．しかし，実際に我々が手の運動を行なうときには，手に道具を持っていたり，手に外力が加わっていたりする場合の方が多い．このような外界との相互作用が加わる場合には躍度最小モデルでは正確な予測や再現は不可能である．

宇野らは筋骨格系のダイナミクスをも考慮した評価関数を持つトルク変化最小モデルを提案した．トルク変化最小モデルでは各関節のトルクの時間変化の 2 乗和の運動時間全体にわたる積分

$$C_\tau = \frac{1}{2}\int_0^{t_f} \left(\frac{d\tau_i}{dt}\right)^2 \tag{1.2}$$

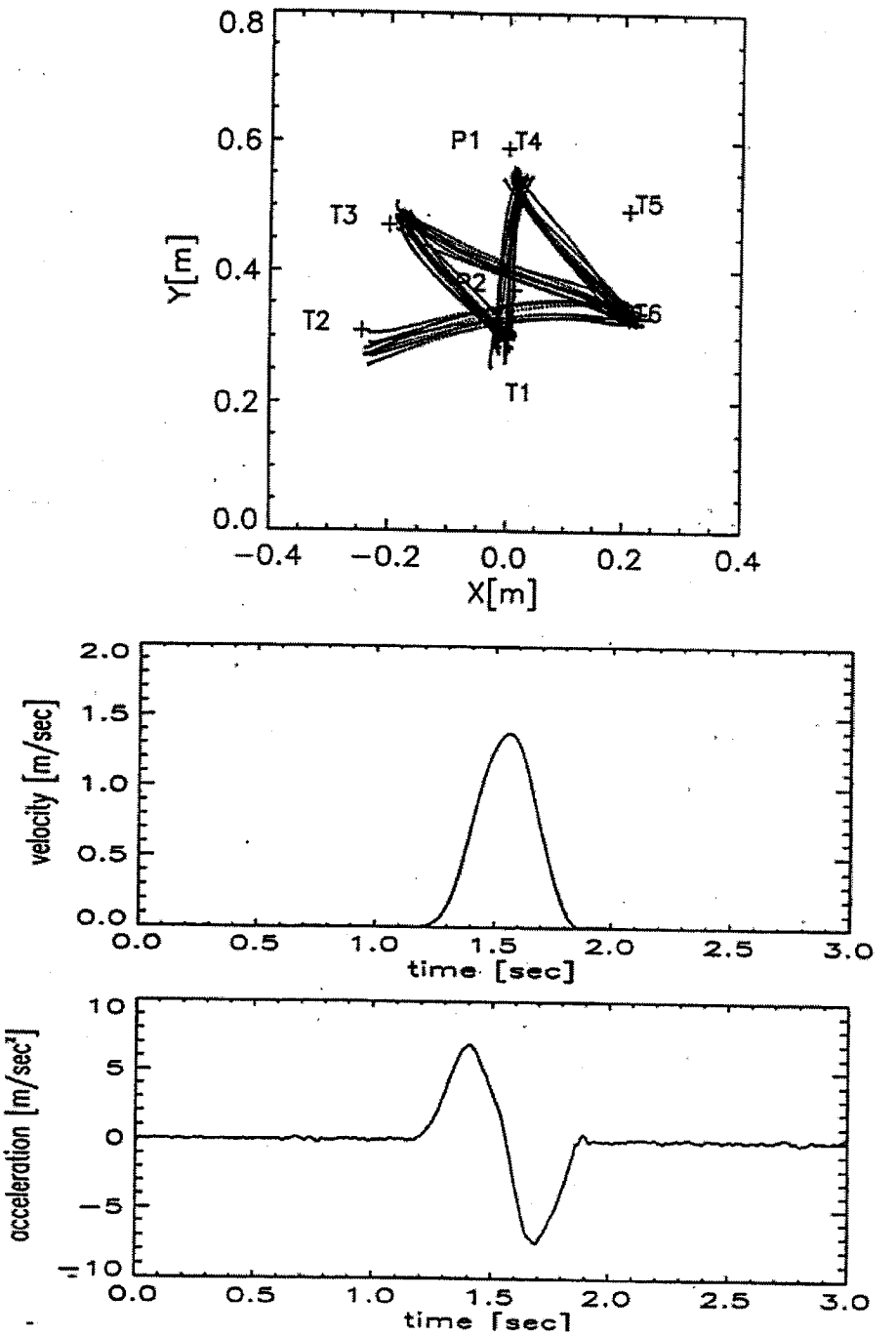


図 1.3: ヒトの 2 点間運動における手先の軌道 (a)，接線方向の速度 (b)，接線方向の加速度 (c)．小池ら (1994) より引用．

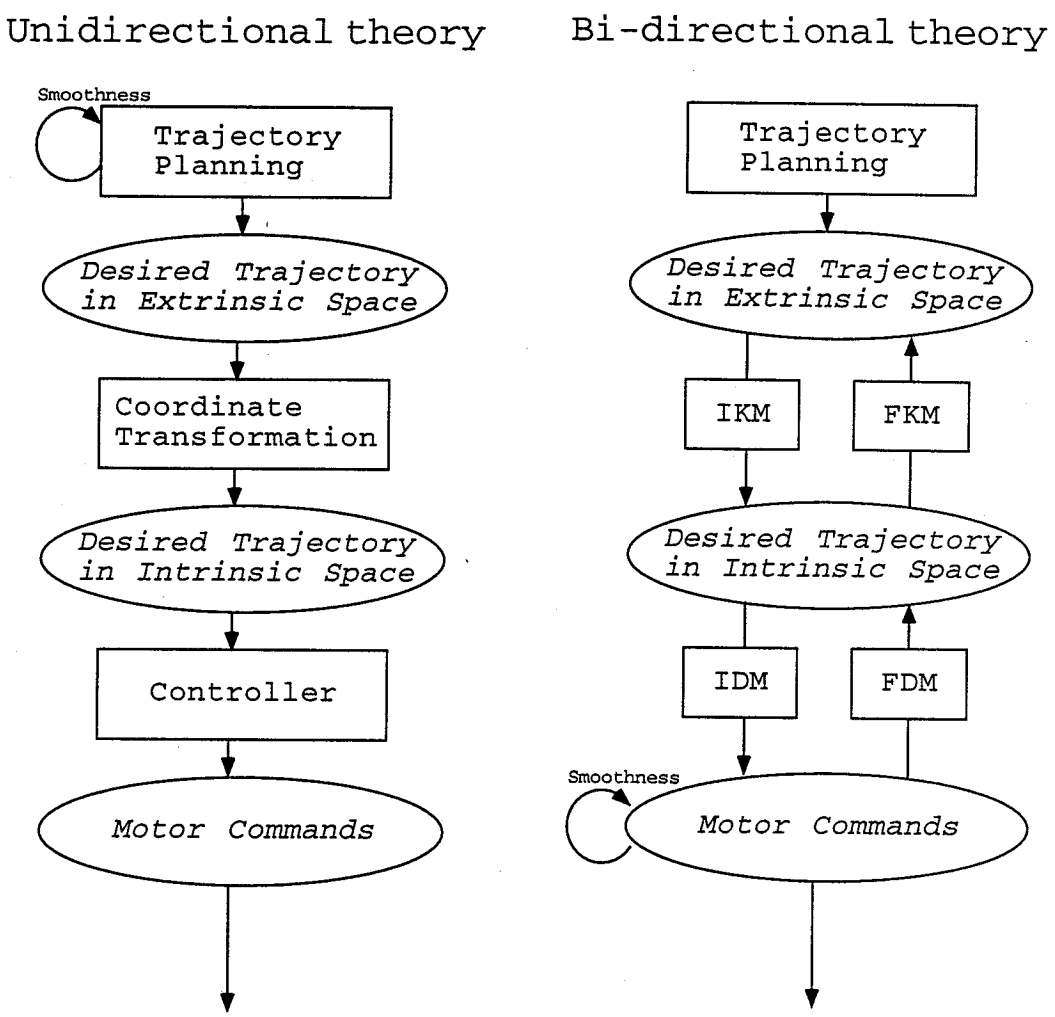


図 1.4: 視覚誘導性到達運動の一方向性理論と双方向性理論の比較．左は一方向理論，右は双方向理論のブロックダイヤグラムを示す．FKM と IKM は順と逆のキネマティクスモデル，FDM と IDM は順と逆のダイナミクスモデルをそれぞれ示す．

が最小になるように運動軌道が計画されるとするモデルである．

手先を真横から正面に動かす場合の運動では，躍度最小モデルよりトルク変化最小モデルのほうが，よく実験結果を説明でき，外界と運動器官の順モデルと逆モデルの両方を必要とするダイナミックな最適化原理が，より正確なモデルであると考えられる [82]．

1.4.3 一方向性理論と双方向性理論

運動制御に関する多くのモデルは大きく分けて二つの理論に分類される．単方向理論と双方向理論である (図 1.4)．どちらの理論も脳内の情報表現とそれに対応する三つの問題が階層的に配置されていることが仮定されている．

一方向理論においては情報の流れは下向きのみである．それに対して，双方向理論では下向きと上向きの両方の流れが許される．下向きの情報の流れは逆キネマティクスモデルと，制御

表 1.1: 視覚運動制御の一方向性理論と双方向性理論.

理論	一方向	双方向
三つの問題をどのように解くか	順序的	同時
軌道が計画される空間	外的空間 (視覚に基づく作業座標)	内的空間 (身体座標) と外的空間
最適化原理	幾何学的 (躍度最小モデル)	ダイナミック (トルク変化最小モデル)
制御	仮想軌道制御仮説	逆ダイナミクスモデル
運動器官と環境の内部モデル	必要なし	順モデルと逆モデル
運動学習	—	内部モデルの学習

対象と環境の逆ダイナミクスモデルによって伝えられる．上向きの情報の流れは順キネマティクスモデルと順ダイナミクスモデルによって伝えられる．

一方向性理論では，情報の流れは高次のレベルから低次のレベルへの下向きの流れのみである．結果として，高次のレベルでの計算問題は低次のレベルの計算問題を参照することなしに解かれる．例えば軌道計画は座標変換や運動指令生成に関する知識なしに解かれなければならない．したがって，表 1.1 に示したように三つの問題は順序的に 1 段 1 段，段階を踏んで解かれる．つまり最初に軌道生成の問題が解かれて，外部空間（多くの場合，視覚座標系）での目標軌道が得られる．次に座標変換が行われて，関節角や筋肉の長さなどの身体座標での目標軌道が得られる．最後に目標軌道を実現するために必要な運動指令が生成される．

それに対して，双方向性理論では下向きの情報の流れだけでなく，上向きの情報の流れも許され，実際にそれが三つの計算問題を十分短い時間で解くために必須のものとなる．高次のレベルでの計算問題は低次のレベルで生じる事象を考慮に入れて解くことができる．例えば軌道計画は運動指令の滑らかさを考慮に入れて行うことができる．したがって，三つの基本的計算問題は直列的，順序的というよりも，同時に解かれる必要がある．

一方向性理論と双方向性理論の最も大きな違いは，軌道が計画される空間に関するものである．

軌道が外的なキネマティックな空間で計画されるのか，あるいは内的なダイナミックな空間

で計画されるのかに関して，現在でも激しい論争がある．一方向性理論では，軌道は外部空間だけで計画される．したがって，低次のレベルで起こるキネマティック，あるいはダイナミックな要因は無視される．それに対して，双方向性理論では，内的空間と外的空間の両者を考慮に入れて軌道が計画される．手先が到達すべき目標位置などの運動の目標は，外的空間で与えられるが，唯一の軌道を決定するための滑らかさの拘束条件は内部空間で与えられる．つまり，軌道計画のために内的空間と外的空間の両者が同時に使われる．

軌道計画が行われる空間の問題と密接に関連して，軌道計画に用いられる最適化原理も大きく異なる．一方向性理論において，軌道計画の過程では低次のレベルを考慮に入れないので，ダイナミックな要因は無視される．したがって，キネマティックな最適化原理のみに基づいて軌道計画が行われる．この代表的なモデルは躍度最小モデルである．それに対して双方向性理論ではダイナミックな最適化原理が用いられる．この代表的なモデルはトルク変化最小モデルである．

運動指令の生成においては，一方向性理論では幾つかの可能性が考えられる．最も純粋な一方向性理論の立場に立ち，長い時間スケールの学習においても階層構造の低次のレベルから高次のレベルへは情報は伝えられないとすれば，仮想軌道制御仮説が，可能な制御原理である．

1.4.4 FIRM(Forward-Inverse Relaxation Neural Network Model) と順序運動

三つの基本的計算問題には解決の難しい不良設定性が含まれるが，図 1.5 に示すダイナミックな最適化原理に基づく双方向神経回路モデルを用いて解決できる [36]．和田らのモデル (FIRM:Forward-Inverse Relaxation Model)[83, 85, 84, 86] を用いると，ヒトの腕の運動軌道データから順逆緩和計算により運動の特徴を少数の経由点という表現で抽出でき，抽出された経由点からもとのヒトの運動を精度よく再現する軌道が再構成できる．音声知覚の運動理論では，調音器官の運動制御のための神経回路が音声認識に重要な役割を担っているという仮説が立てられている [48, 49, 35]．我々のアルゴリズムはこの心理学的なモデルのひとつの計算論的実現であるといえる．

1.5 見まねによる運動学習

タスクレベル学習を行おうとするときには，内部の制御変数と外界の状態変数とのインタラクションが重要となる．タスクの表現 (制御変数と状態変数) が適切に記述されれば，強化学習や遺伝アルゴリズムなどの様々な学習スキームを用いることができる．ここで最も基本的で解決の困難な問題は，タスクに関する適切な表現を選択することであると考えられる (例えば [1, 4, 75])．

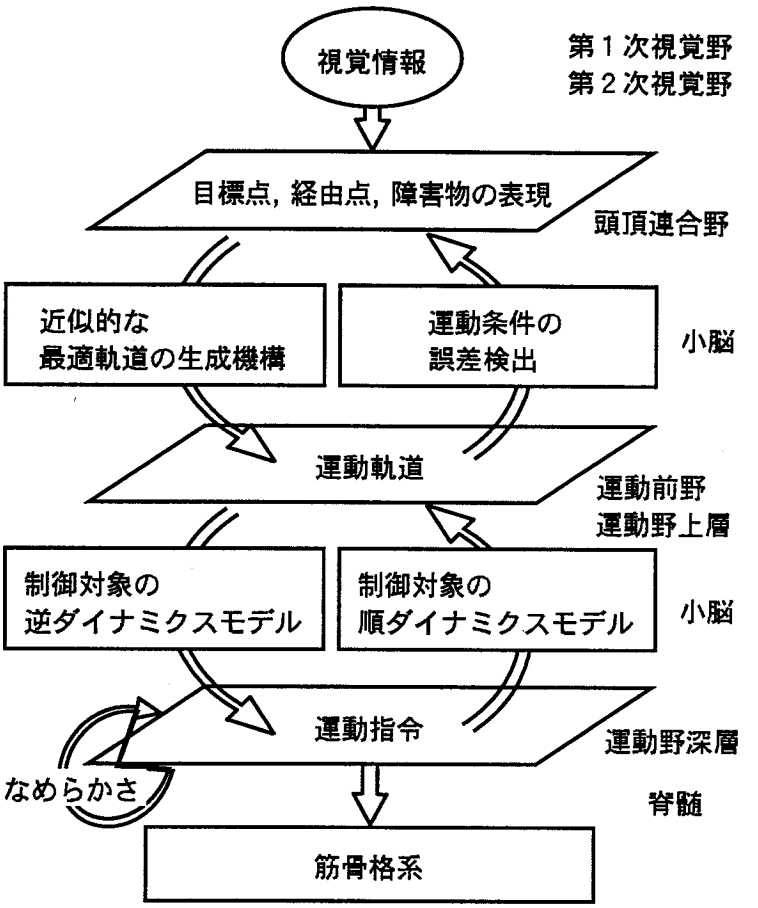


図 1.5: 双方向神経回路モデル.

表 1.2: 見まねによる学習の戦略.

タスク学習の戦略	運動プリミティブの表現	学習者の知能	教師と学習者の違い
「運動意図の理解」	運動意図	完全	全く異なっても良い
「創発計算の獲得」	タスクダイナミクス	良	かなり違っても良い
「抽象的表現を用いたタスクレベル学習」	経由点	貧弱	定量的な違い
「サルまね」	位置と力の制御	殆どない	全く同じ

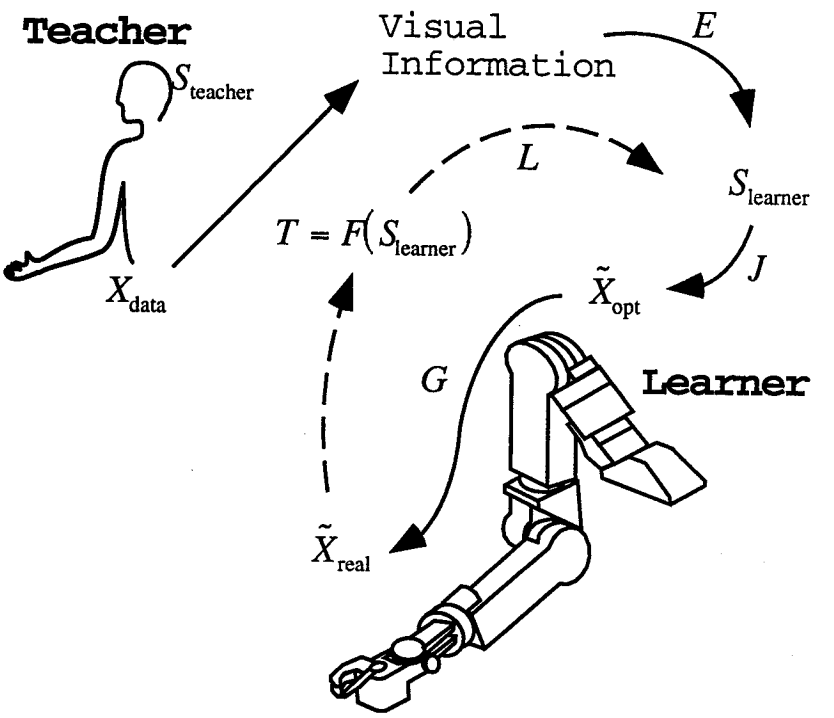


図 1.6: 見まねによる運動学習の流れ.

見まねによる学習は，教師のタスクの遂行に基づいて学習者がタスクを解くことを学ぶことであり，タスクの表現の選択という原理的問題が直接的に現れる問題のひとつである．見まねによるタスク学習の戦略の例を表 1.2 に示す．学習者の知能が完全であれば，教師の運動パターンの観察を通じて運動意図を理解し，学習者自身の身体や外界の物理的な理解と推論に基づいてタスクを遂行できる．AI 的な手法によるロボット教示の自動化などは，この戦略をめざしたアプローチといえるだろう．また，教師の運動の位置と力の軌道をそのまま再現するという戦略もある．しかしこの方法では，教師と学習者の身体構造と外部環境が全く同じで，かつ極めて高い精度の測定と制御が要求される等，現実的でない．

図 1.6 に見まねによる運動学習の流れを示す．大まかに言えば，学習者 (Learner) は教師 (Teacher) の運動軌道を認識して運動計画の内部表現を推定する．これを用いて学習者自身の運動学習制御のための内部表現とする．この内部表現を操作することによって学習者自身に合わせる．

1.5.1 経由点と最適軌道に基づく運動制御

教師の運動軌道 X_{data} から経由点抽出アルゴリズム E によって抽出された直交座標での経由点の集合

$$S_{\text{learner}} = \{\mathbf{X}^1_{\text{via}}, \cdots, \mathbf{X}^N_{\text{via}}\}$$

が得られたとする．X^i_{via} は 6 次元ベクトルである．S_{learner} は教師の運動計画の内部表現 S_{teacher} を推測したものである．

$$S_{\mathrm{learner}} = E(X_{\mathrm{data}}) \tag{1.3}$$

つぎに，教師から得られた経由点の集合から経由点決定のアルゴリズム H により学習者の経由点の集合

$$S_{\mathrm{learner}} = \{\tilde{X}^1_{\mathrm{via}}, \cdots, \tilde{X}^N_{\mathrm{via}}\}$$

を得るとする．

$$S_{\mathrm{learner}} = H(S_{\mathrm{teacher}}) \tag{1.4}$$

この経由点から最適化軌道計算 J によって最適軌道 $\tilde{X}_{\mathrm{opt}}$ を得る．

$$\tilde{X}_{\mathrm{opt}} = J(S_{\mathrm{learner}}) \tag{1.5}$$

手先座標から身体座標への座標変換と，運動指令生成のアルゴリズムをまとめて G と表す．G により実現された軌道を

$$\tilde{X}_{\mathrm{real}} = G(\tilde{X}_{\mathrm{opt}}) \tag{1.6}$$

とする．以上の式を続けると，

$$\begin{aligned} \tilde{X}_{\mathrm{real}} &= G(J(S_{\mathrm{learner}})) \\ &= G(J(E(X_{\mathrm{data}}))) \\ &\equiv \chi(X_{\mathrm{data}}) \end{aligned} \tag{1.7}$$

となる．このようにして合成された汎関数 χ は以下の条件が満たされていれば恒等写像からそれほどかけ離れたものとはならない．

1. 最適化原理がヒトのそれとひどく異ならない．

2. 経由点の表現が適切である．

3. G は恒等写像からそれほどかけ離れていない．

最終的に実現された軌道 $\tilde{X}_{\mathrm{real}}$ は滑らかで，教師の運動軌道 X_{data} に近い．しかし，教師と学習者の動力学特性や環境の違い等により，タスクが実現できるとは限らない．

$\tilde{X}_{\mathrm{real}}$ は少ない制御変数 S_{learner} でパラメトライズされている．タスクの成功失敗を直接評価する表現を T とすると，T は $\tilde{X}_{\mathrm{real}}$ から一意に決定できるから，

$$T = F(S_{\mathrm{learner}})$$

と書ける．図 1.6 の点線で示した学習スキーム L により S_{lear} を修正して，自分自身の行動の結果

$$T = F(S_{\text{learner}})$$

がタスク目標に近付くようにすることができる．L としては，フィードバック誤差学習，順逆モデリング，直接逆モデリング，遺伝アルゴリズム，強化学習，最急降下法など，様々な学習スキームを適用することができる．

第 2 章

みまねによるけん玉学習ロボット

2.1 はじめに

本研究では第 1 章で述べた脳の運動制御の計算理論のひとつである “最適化原理に基づく運動パターン認識” の枠組を用いて，見まねによる運動学習を試みている．ここではそのひとつの応用例として，抽象的表現として経由点を用いたタスクレベル学習を行なう．その第一段階として，まず本章ではタスクとしてけん玉をとりあげた．けん玉は習得が比較的容易なダイナミックな運動で，しかも簡単なタスク表現が可能である．けん玉を行うにはかなり高速な動作が要求され，各リンク間に複雑な干渉のある腕型マニピュレータでは高い精度で軌道追従を行うことは困難である．けん玉をロボットに行わせる試みは，視覚フィードバックを用いた研究 [52] や，外乱オブザーバを用いた研究 [24] がある．これらの研究は人間の動作を参考にして高い成功率でけん玉を成功させているが，研究者が陽に軌道計画を行っている．

本研究では，和田らの FIRM(Forward-Inverse Relaxation Model) を用いてヒトの運動軌道から経由点を抽出し，これらの経由点を制御変数とみなし修正することにより作業目標の達成を目指す．学習スキームとして広義ニュートン法を用いる．本章では，まず 2.3 節で実機を用いた実験を行い本方式の有効性を検証し，つぎに 2.4 節で，抽出された経由点がタスク表現として適切なものであるかを数値実験により検討する．

2.2 ケン玉ロボットの特徴と概要

経由点は一種の情報圧縮の表現のひとつと考えることもでき，軌道生成の最適化原理に基づいて，与えられたデータから抽出できる．抽出された経由点から軌道を再構成する場合も最適化問題として解くことができ，運動指令の変化が最小となる最適軌道が求まる．ここで用いる最適化原理が人間のそれに近いもので経由点の表現が適切であれば，もとの人間の運動軌道を再現することができる．しかしロボットなどの制御対象に経由点から再構成された軌道をそのまま与えても，似た運動はできても目的の作業がうまく行えるとは限らない．腕の長さや重さ

といった物理的な変数が人間とロボットでは異なるからである．そこで，経由点のうち作業の成功に重要な経由点を制御変数とみなし，適切に修正することによって再構成された軌道を制御する．ここで経由点をどのように修正すればよいかが問題となるが，目的の作業が成功するように，ケン玉の場合ではボールの飛び方を観察し，例えばもっと右に飛ぶようにするにはどの経由点をどのように修正すればよいかをあらかじめ調べておく．ボールの飛ぶ高さや方向に関していろいろ調べた結果を使ってボールがうまく飛ぶように経由点を少しずつ修正する．最初はうまくできなくても，これを何度か繰り返していくうちにだんだん上達していくのである．これはタスクレベル学習という考え方で，制御系の精度はそれほど高くなくてもうまく目標が達成できる．

ここで制御対象を図 2.1 に示す SARCOS Dextrous Slave Arm(米国 SARCOS 社製) とする．この SARCOS アームはもともとマスター・スレーブ方式の遠隔操作の研究のためにユタ大学で開発された腕型ロボットで，人間がマスター・アームを操作することによってスレーブ・アームを制御する．マスター・アームには各関節に角度検出器だけでなくアクチュエータも取り付けられていて，ロボットと操作対象物との間の力の相互作用も操作者にフィードバックされる仕組みになっている．軌道制御は Vx-Works とよぶマルチタスク OS のもとで，VME バス上のマルチ MPU で行われ，専用のバスを介して図 2.2 に示す各軸毎のサーボコントローラから制御指令が SARCOS アームに送られる．本研究ではスレーブアームのみを実験に用いたので，本論文では以降スレーブ・アームを単に SARCOS アームと呼ぶこととする．

一般的に用いられている産業用の腕型電動マニピュレータは 6 自由度しか持たないが，SARCOS アームは図 2.4 に示すように人間の腕と同じ 7 自由度を持ち，人間の腕に非常に似た動作を行わせることができる．SARCOS アームは図 2.3 に示す油圧ポンプからの油圧により油圧アクチュエータで各関節が駆動される．油圧アクチュエータは産業用マニピュレータに一般的に用いられている電動モータに比べて，小型軽量で大トルクを発生させることができるが，油洩れが発生し保守性が悪いことが欠点である．しかし SARCOS アームは電動の産業用マニピュレータでは不可能な高速の運動が可能である．上にあげた理由から，SARCOS アームを用いることにより，本論文で扱うようなけん玉やテニスサーブといった比較的高速でダイナミックな運動を SARCOS アームに行なわせる実験が可能となった．

2.3 SARCOS アームによるけん玉運動学習実験

2.3.1 運動軌道計測

実験に用いたシステムの概略を図 2.5 に示す．ケン玉ロボット開発のごく初期段階ではマスター・アームで人間のケン玉運動を測定しようとしていたのであるが，重力保障等の工夫だけ

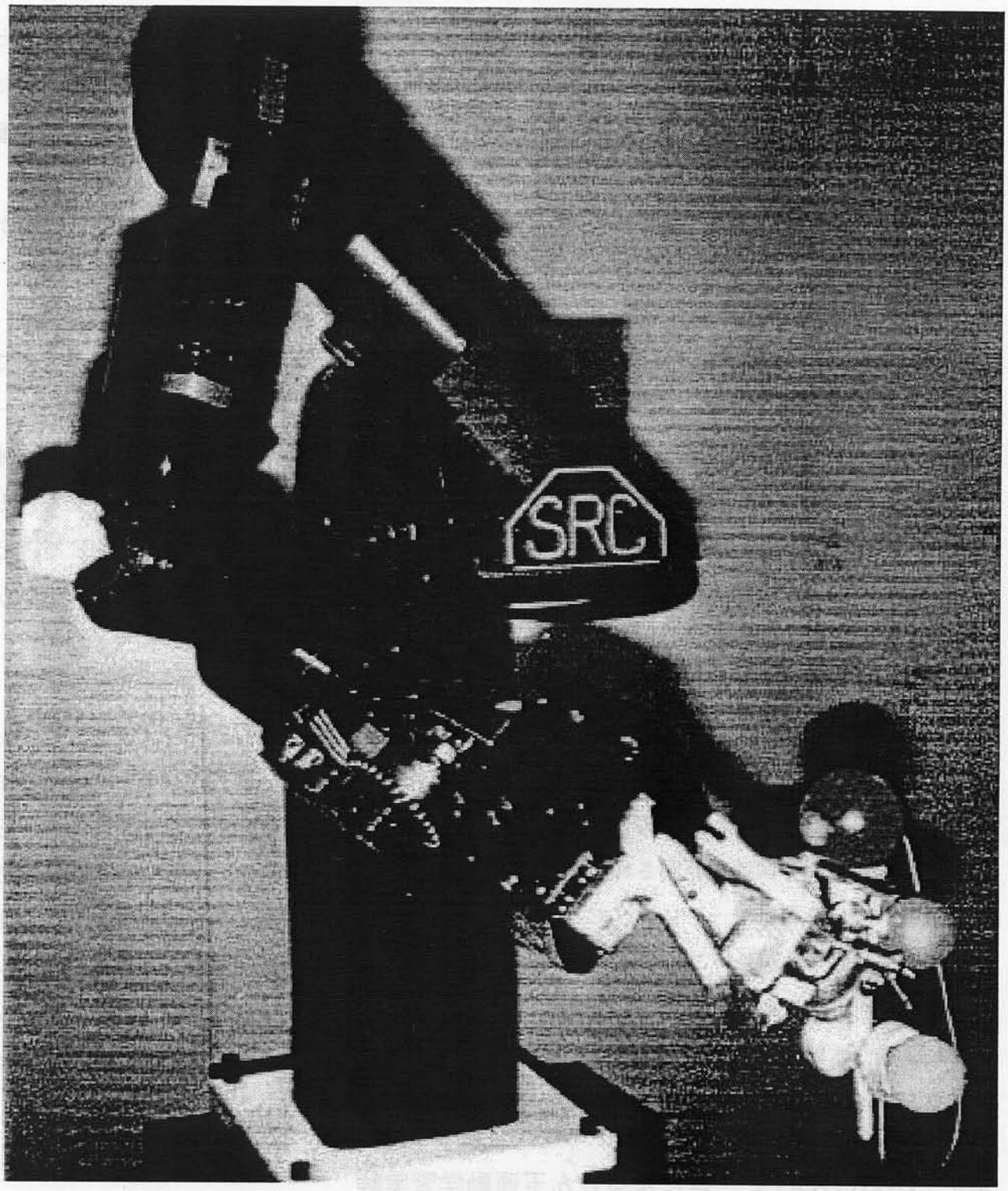


図 2.1: SARCOS Dextrous Slave Arm(米国 SARCOS 社製). けん玉のけんは固定した指先に小さな万力で堅く取り付けられている. ボールは受け皿 (中) のうえに置かれている.

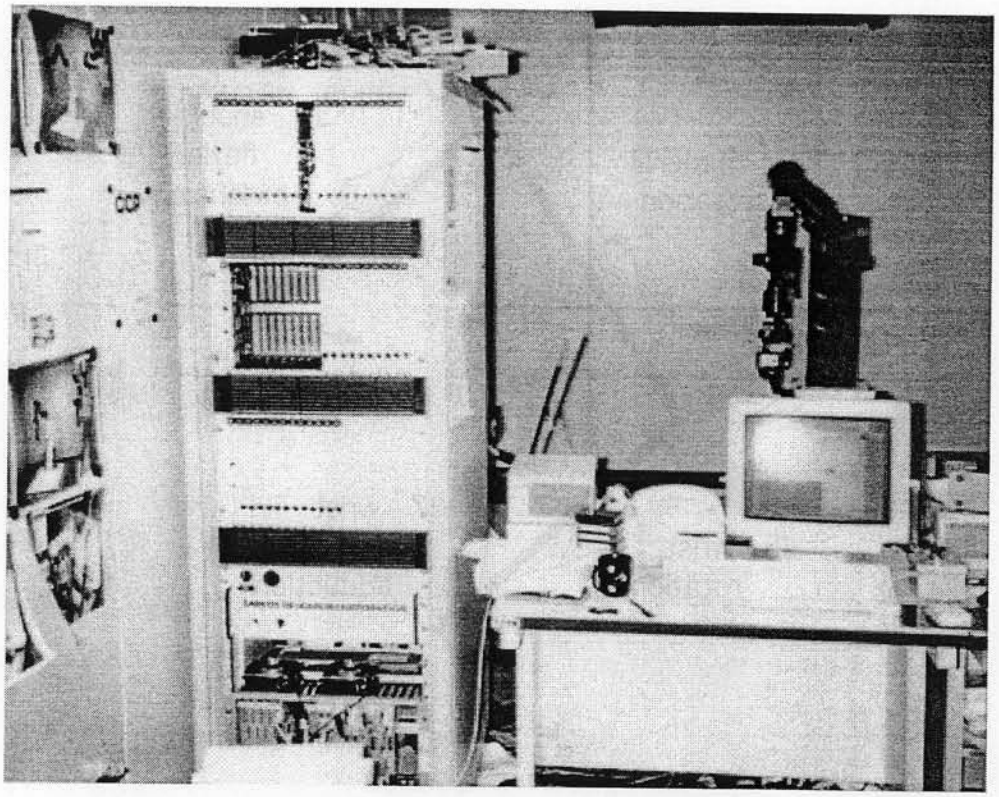


図 2.2: SARCOS アームのコントローラ部 (米国 SARCOS 社製).

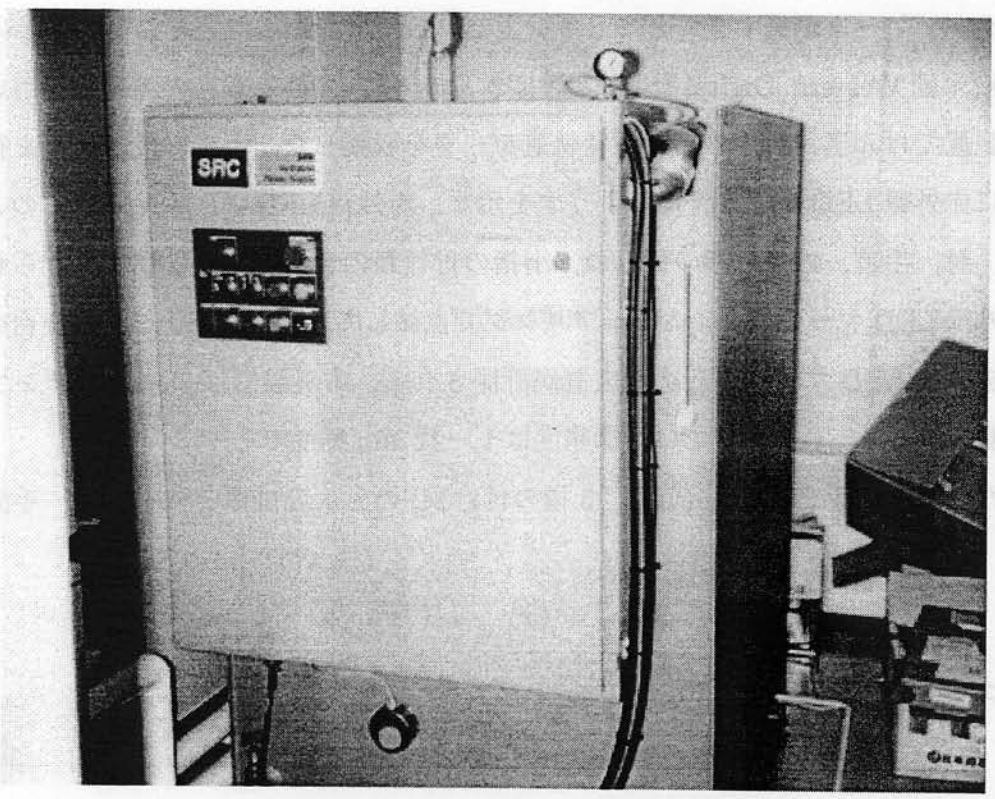


図 2.3: SARCOS アームの油圧ポンプ部 (米国 SARCOS 社製).

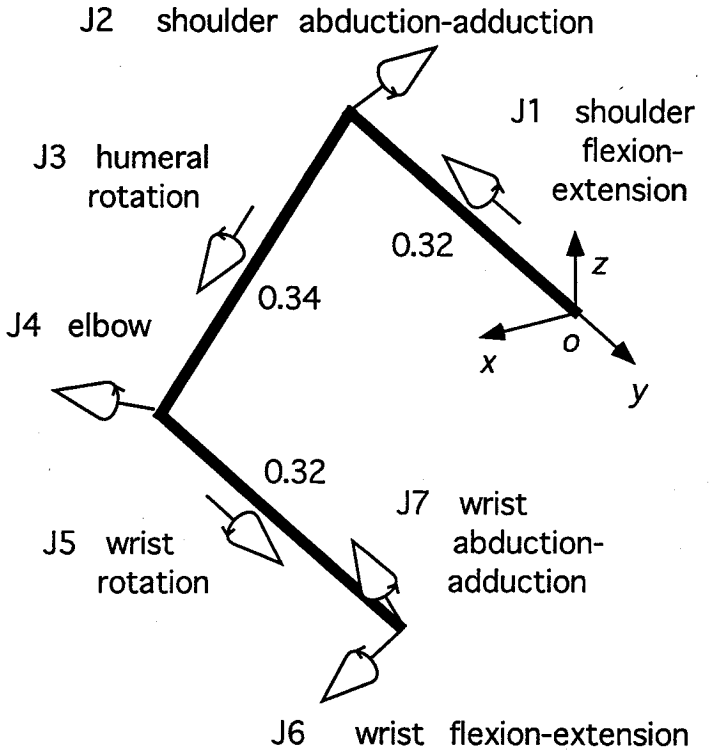


図 2.4: SARCOS アームのキネマティック・ストラクチャー．各リンクの横の数字はリンク長 (単位はメートル) を示す．矢印は回転軸と回転の方向を表す．

ではマスター・アームが重すぎてとてもケン玉どころではなかった．そこで図 2.6 に示す OPTOTRAK(米国 Western Digital 社製) と呼ばれる赤外線 LED と 3 台のカメラを用いた三次元位置計測装置でけん玉遂行中のヒト運動軌道データを収集した．図 2.7 にけん玉を行なうときの被験者に赤外線 LED を取り付けたようすを示す．赤外線 LED マーカは図 2.7 のように，被験者の肩，肘，手首，人さし指の付けね，小指の付けねの合計 5 個所に取り付けられた．それぞれの赤外線 LED マーカはタイムシェアリングで点滅しており，図 2.6 に示す 3 台のカメラによって各赤外線 LED マーカの三次元位置が計測される．本実験では 250Hz のサンプリング周波数で計測した．本実験の設定では計測精度は 1 ～数 mm 程度である．

収集した肩，肘，手首，手先 (第 2，5 指の付け根) の三次元位置データから，手先の位置と向き

$$X_{\mathrm{data}} = \{x^1_{\mathrm{data}}, \cdots, x^6_{\mathrm{data}}\}, \tag{2.1}$$

および 7 つの関節角の軌道

$$\Theta_{\mathrm{data}} = \{\theta^1_{\mathrm{data}}, \cdots, \theta^7_{\mathrm{data}}\} \tag{2.2}$$

を計算によって求めた．手先の位置は第 5 指の付け根の三次元位置データをそのまま用いた．手の高さ (z 軸方向) が最も高くなる時間がデータ全体 (2 秒間) の中心になるようにデータをそ

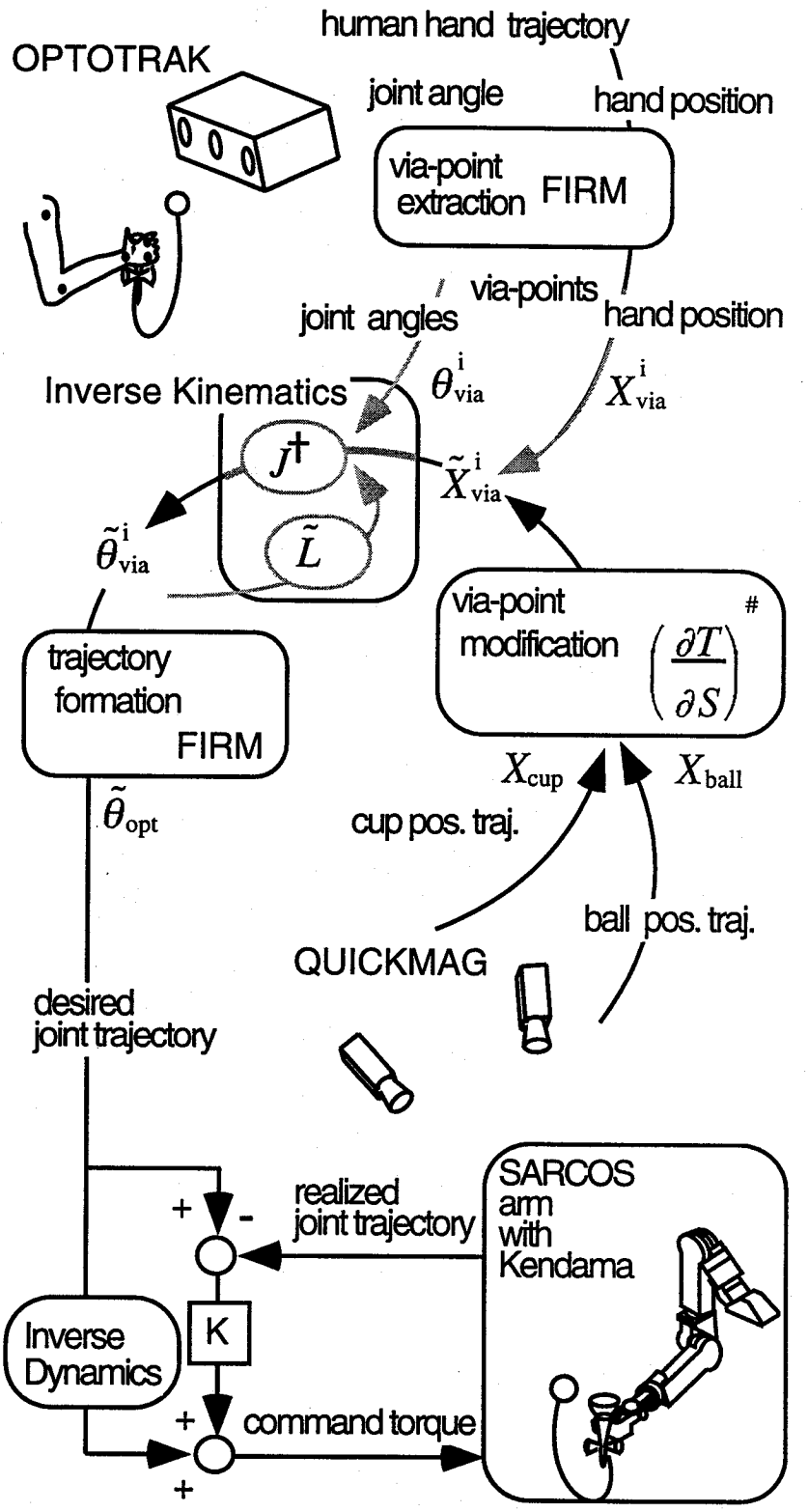


図 2.5: けん玉学習システムの模式図.

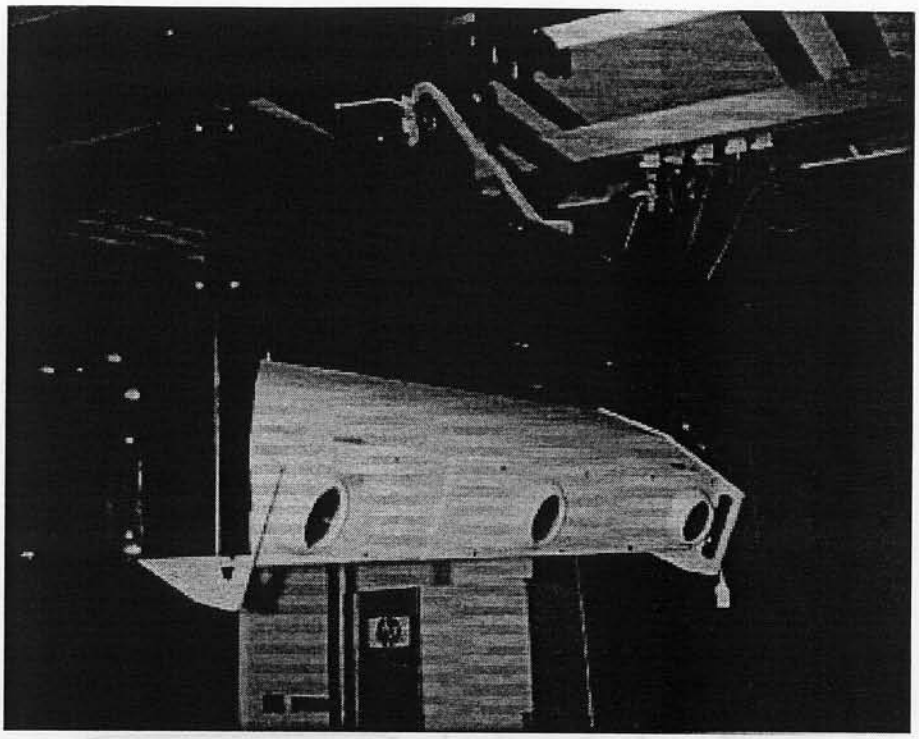


図 2.6: OPTOTRAK のカメラ部 (米国 Western Digital 社製).

図 2.7: 被験者と LED マーカー．赤外線 LED マーカは被験者の肩，肘，手首，人さし指の付けね，小指の付けねの合計 5 個所に取り付けられている．

ろえた.

2.3.2 経由点の抽出と最適軌道の再構成

次に FIRM を用いて経由点 X_{via} を抽出した．経由点抽出モジュールは

$$\int_0^T \sum_{i=1}^{6} \{x^i(t) - x^i_{\mathrm{data}}(t)\}^2 dt \to \min \tag{2.3}$$

を満たすように経由点が抽出される．x^i は後述する最適軌道生成モジュールにより得られる最適軌道である．Θ_{data} から X_{via} と同時刻のデータ点を取り出すことにより，関節角の経由点 Θ_{via} を得た．

最適軌道生成モジュールは簡単のため元々の非線形ダイナミクスではなく，近似として質点系の線形ダイナミクスを拘束条件としたトルク変化最小基準を用いた．この場合，躍度最小モデルと等価になり，

$$\int_0^T \sum_{i=1}^{6} \left(\frac{d\ddot{x}^i(t)}{dt} \right)^2 dt \to \min \tag{2.4}$$

を満たす解析解が与えられる．

次に FIRM を用いてそれぞれの経由点 $\mathbf{X}_{\mathrm{via}}$, $\boldsymbol{\theta}_{\mathrm{via}}$ を抽出した．最適軌道は簡単のため躍度最小軌道を用いた．躍度最小軌道は

$$\int_t \left(\frac{\mathrm{d}\ddot{x}}{\mathrm{d}t} \right)^2 \mathrm{d}t \tag{2.5}$$

が最小となる軌道である．これは解析的に解けて時間に関する 5 次方程式となる．

図 2.8 に経由点抽出の模式図を示す．丸は経由点を示す．実線は被験者の運動軌道，点線は経由点から再構成された最適軌道をそれぞれ示す．ϕ, θ, ψ 成分は YZY オイラー角 [8] である．まず図 2.8左 に示すように 1 番の始点と 2 番の終点を結ぶ最適軌道を生成する．つぎに図 2.8中 の 3 番の丸で示される最初の経由点を抽出し，そこを通る最適軌道を生成する．経由点として，手本の運動軌道と最適軌道の x, y, z 成分の自乗和の誤差が最大の点を選ぶ．同様に図 2.8右 に示すようにさらに経由点を追加する．もとの手本の軌道に最適軌道が十分近くなるまで，この経由点抽出と最適軌道の生成を繰り返す[1]．本実験では簡単のため，経由点の個数を始点と終点を含めて 9 個に固定した．

2.3.3 手先座標から関節角座標への変換

SARCOS アームへの指令値は図 2.2 に示すコントローラを通じて関節角軌道の目標値として与えられる．SARCOS アームの関節角の目標値を人間の関節角度に合わせたのでは，腕の長さ等のキネマティクスが異なるため目的の作業は達成できない．目的の作業は直交座標で観測さ

[1] この手順はスプライン補間における節点の抽出に類似している [26, 74]

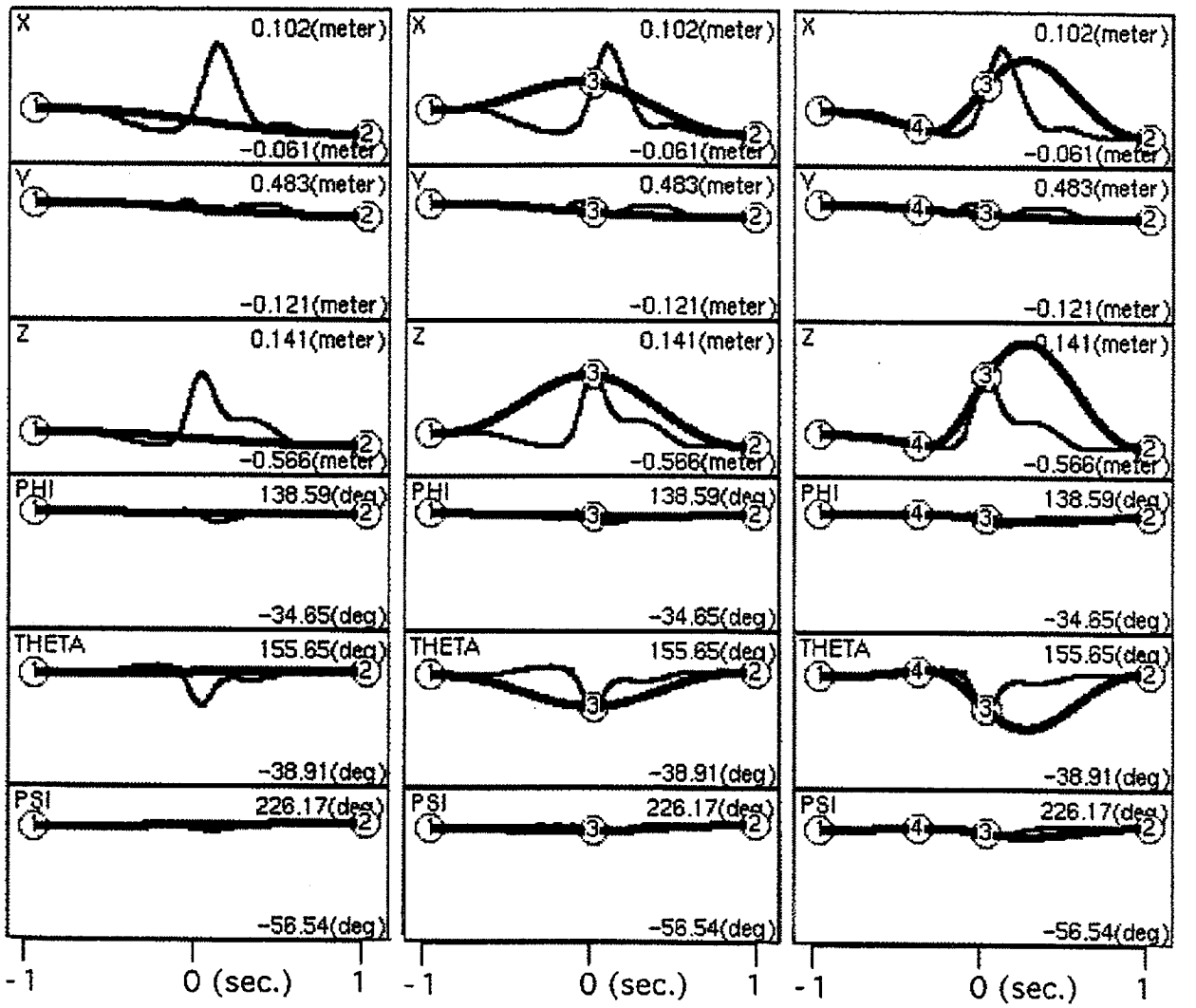


図 2.8: 経由点抽出過程．丸は経由点，実線は被験者の運動軌道，点線は経由点から再構成された最適軌道をそれぞれ示す．上から三次元位置と向きの $x, y, z, \phi, \theta, \psi$ 成分をそれぞれ示す．

れるので，抽出された人間の運動軌道の手先の位置と姿勢の経由点からSARCOSアームの関節角軌道の経由点を求める．このとき，手先の位置と向きが決まっても腕全体の姿勢はどのようにもできるように，冗長性のため直交座標から関節角座標への変換は一意ではないので，SARCOSアームの手先が直交座標の経由点で人間のそれと厳密に一致し，かつ関節角の経由点で人間の関節角と最も近くなるように決める．

ヒト運動軌道から抽出された N 点の経由点を関節角度で

$$\boldsymbol{\theta}^i_{\rm via}(i=1,2,...,N), \tag{2.6}$$

直交座標 (手先位置と姿勢) で

$$\mathbf{X}^i_{\rm via}(i=1,2,...,N) \tag{2.7}$$

と表す．$\boldsymbol{\theta}^i_{\rm via}$ は7次元，$\mathbf{X}^i_{\rm via}$ は6次元ベクトルである．SARCOSアームの順キネマティクス方程式を

$$\tilde{\mathbf{X}} = \tilde{\mathbf{L}}(\tilde{\boldsymbol{\theta}}) \tag{2.8}$$

で表す．$\tilde{\mathbf{X}}$, $\tilde{\boldsymbol{\theta}}$ はそれぞれSARCOSアームの腕の直交座標，関節角座標での状態を表す．このとき，冗長性のため直交座標から関節角座標への変換は一意ではないので以下のようにする．$\tilde{\boldsymbol{\theta}}^i_{\rm via}$ は

$$\tilde{\mathbf{X}}^i_{\rm via} = \tilde{\mathbf{L}}(\tilde{\boldsymbol{\theta}}^i_{\rm via}) = \mathbf{X}^i_{\rm via} \tag{2.9}$$

を満たす $\tilde{\boldsymbol{\theta}}$ のうち

$$\| \boldsymbol{\theta}^i_{\rm via} - \tilde{\boldsymbol{\theta}}^i_{\rm via} \| \tag{2.10}$$

を最小にするものに決定する．言い換えれば，SARCOSアームの関節角度の経由点 $\tilde{\boldsymbol{\theta}}^i_{\rm via}$ をヒトの経由点と直交座標では厳密に一致し，関節角度では最も近くなるように決める．

上で述べた方法を具体的にインプリメントすると以下のようになる．SARCOSアームの順キネマティクス方程式

$$\tilde{\mathbf{X}} = \tilde{\mathbf{L}}(\tilde{\boldsymbol{\theta}}) \tag{2.11}$$

と広義ニュートン法を用いて [28, 39] SARCOSアームの関節角の経由点 $\tilde{\boldsymbol{\theta}}^i_{\rm via}$ を得る．

$${}^{n+1}\tilde{\boldsymbol{\theta}}^i_{\rm via} = {}^{n}\tilde{\boldsymbol{\theta}}^i_{\rm via} + J^{\dagger}({}^{n}\tilde{\boldsymbol{\theta}}^i_{\rm via}) \left\{ \mathbf{X}^i_{\rm via} - \tilde{\mathbf{L}}({}^{n}\tilde{\boldsymbol{\theta}}^i_{\rm via}) \right\} \tag{2.12}$$

ただし，左上の添字 n は広義ニュートン法の繰り返し回数を示す．$J^{\dagger}$ は特異点低感度運動分解行列 [67]

$$J^{\dagger}(\theta) = (\partial \tilde{\mathbf{L}}(\boldsymbol{\theta})/\partial \boldsymbol{\theta})^{\dagger} \tag{2.13}$$

を用いた．この行列は

$$\|(I - JJ^{\dagger}\|^2 + k\|J^{\dagger}\|^2 \tag{2.14}$$

を最小にする．ここで，I は単位行列である．この式の第 1 項は正確な逆変換を求めるもので，第 2 項は関節角があまり変化しないことを求めるものである．k はこれらの二つの要求の重み係数である．$J^\dagger$ の正確な記述は以下のとうりである．

$$J^\dagger = (J^{\mathrm{T}}J + kI)^{-1}J^{\mathrm{T}}. \tag{2.15}$$

ここで，J^{T} は J の転置行列を表す．k は以下の式にしたがって変化する．

$$k = \begin{cases} K_0(1 - w/w_0)^2 & \text{if } w < w_0 \\ 0 & \text{otherwise} \end{cases}, \tag{2.16}$$

$$w = \sqrt{\det(J^{\mathrm{T}}J)}. \tag{2.17}$$

ここで w は特異姿勢からの距離を表す．特異姿勢とは例えば肘がまっすぐになった状態で，SARCOS アームのキネマティクス方程式の逆変換が存在しないような姿勢である．w_0 は特異姿勢の近傍を表す．SARCOS アームが特異姿勢にあるときは k は $k_0 > 0$ となる．特異姿勢の近傍内で w が大きくなるにつれ k は 0 に近付く．w が十分大きいとき，すなわち特異姿勢から充分離れているときは $k = 0$ となり，$J^\dagger$ は Moore-Penrose 一般逆行列 [71, 73] と完全に一致する．

$\tilde{\boldsymbol{\theta}}^i_{\mathrm{via}}$ の初期値は

$${}^0\tilde{\boldsymbol{\theta}}^i_{\mathrm{via}} = \boldsymbol{\theta}^i_{\mathrm{via}} \tag{2.18}$$

とする．$J^\dagger$ はノルム最小の性質をもつ [71, 73] から，$\tilde{\boldsymbol{\theta}}^i_{\mathrm{via}}$ はヒトの経由点と直交座標で厳密に一致し，関節角度では最も近くなるように決まる．

2.3.4 SARCOS アームによるタスクの遂行

抽出されたヒト運動軌道の経由点 $X_{\mathrm{via}}, \theta_{\mathrm{via}}$ から得られた SARCOS アームの関節角の経由点 $\tilde{\theta}^i_{\mathrm{via}}$ から FIRM を用いて最適軌道 $\tilde{\theta}_{\mathrm{opt}}$ を再構成し，SARCOS アームに与える関節角目標軌道とする．腕のダイナミクスが

$$\tau^j = I^j\ddot{\theta}^j \tag{2.19}$$

のように線形近似できるときは最適軌道を生成する方法は二通りある．ここで $\tau^j, I^j, \ddot{\theta}^j$ は j 番目のアクチュエータによって発生するトルク，リンクの慣性モーメント，関節角加速度をそれぞれ表す．ひとつの方法はスプライン関数に基づく方法である．本章で採用したもうひとつの方法では経由点を通る軌道は順番に生成される．式 (2.19) で示されるような線形近似されたダイナミクスに対して最小化原理を用いると FIRM で最適軌道 $\tilde{\theta}_{\mathrm{opt}}$ を計算できる．

図 2.9 に手先位置の軌道と経由点を示す．X, Y, Z はそれぞれ右方向，前方向，上方向が正となるような座標のとり方をしてある．丸で示されているのが経由点で，腕を上げる直前や上が

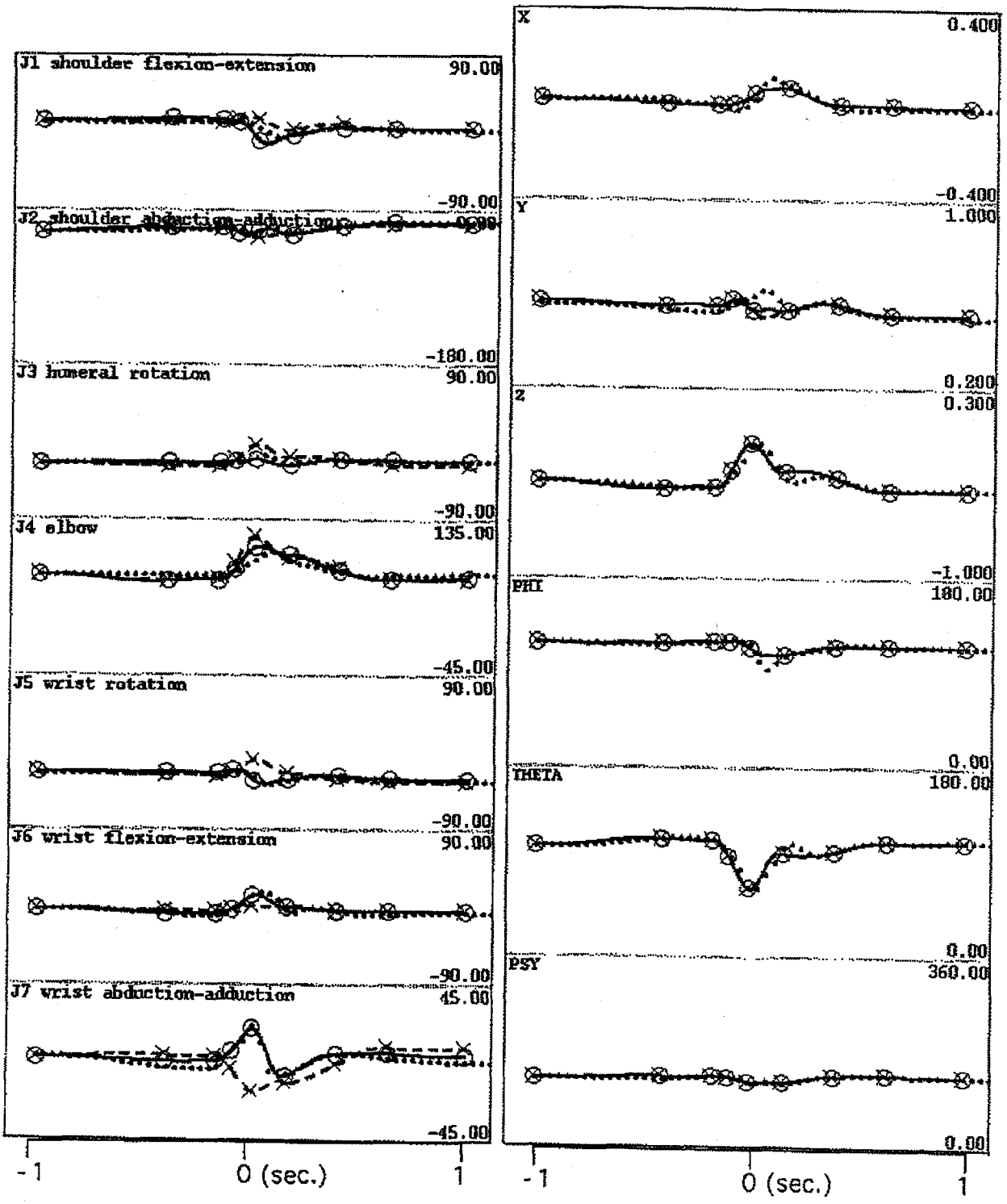


図 2.9: ヒトの腕と SARCOS アームの 左: 関節角の軌道， 右図:SARCOS アームの手先の三次元位置と向きの軌道． X, Y, Z はそれぞれ右方向，前方向，上方向が正．丸は経由点，実線は最適化原理によって再構成された目標軌道，点線は SARCOS アームによって実現された軌道をそれぞれ示す．

り切ったところなどの運動の特徴を良く表すところに経由点が選ばれている．実線は最適化原理によって再構成された目標軌道で，点線はSARCOSアームによって実現された軌道をそれぞれ示す．SARCOSアームはフィードバック制御だけでなく逆ダイナミクスを用いた前向き制御も行われているが，ケン玉運動がかなり速い運動であるためと，精度の高い前向き制御が困難であるため，実現される軌道は目標軌道とかなりずれていることが分かる．

図2.9左の丸は始点と経由点と終点，薄い実線はけん玉遂行中のヒト関節角軌道を，濃い実線は経由点から再構成されたSARCOSアームに与える最適軌道 $\tilde{\theta}_{\rm opt}$ を，点線はSARCOSアームによって実現された関節角軌道 $\tilde{\theta}_{\rm real}$ をそれぞれ示す．図2.9右 の実線はSARCOSアームに与える関節角軌道から順キネマティクス方程式を用いて求めた手先の位置と向きの軌道を示す．順キネマティクス方程式，順ダイナミクス方程式，および逆ダイナミクス方程式は回転行列を用いる方法で解いた[13, 88, 31]．ヒトの手先の位置と向きの軌道はSARCOSアームの軌道とほぼ重なっているので，図では区別出来ない．点線はSARCOSアームの関節角実現軌道から順キネマティクス方程式を用いて求めた手先の位置と向きの軌道を示す．SARCOSアームは前向きとフィードバックの制御が行われるが，図2.9から分かるように，実現される軌道は目標軌道とは完全には一致しない．

2.3.5 タスクの定義

前節で得られた関節角の目標軌道をそのまま用いる，(つまり人間のけん玉運動を単にまねした) だけでは，図2.14(学習前のけん玉の試行) に示すようにSARCOSアームはけん玉に成功しない．SARCOSアームにけん玉を成功させるようにするため，タスクレベル学習の概念を導入した．Aboafらは下位のレベルのモジュールが完全でなくともタスクレベルで学習を進めることができることを示した[1]．ここではまず最初に，より直観的で直接的なタスク変数の定義を示し，つぎにより実際的なタスク変数に定義しなおす．

けん玉の成功のために重要であると考えられるタスク表現 $\mathbf{T}$ を以下に示す．実現タスクと目標タスクを

$$\mathbf{T}^{\rm int} = \begin{pmatrix} x_{\rm h}^{\rm b} \\ y_{\rm h}^{\rm b} \\ z_{\rm h}^{\rm b} \end{pmatrix}, \tag{2.20}$$

$$\mathbf{T}_{\rm desired}^{\rm int} = \begin{pmatrix} x_{\rm h}^{\rm c} \\ y_{\rm h}^{\rm c} \\ z_{\rm h}^{\rm c} \end{pmatrix} \tag{2.21}$$

のように表現する．ここで $(x_{\rm h}^{\rm b}, y_{\rm h}^{\rm b}, z_{\rm h}^{\rm b})^{\rm T}$ と $(x_{\rm h}^{\rm c}, y_{\rm h}^{\rm c}, z_{\rm h}^{\rm c})^{\rm T}$ はボールが受け皿の高さまで落ちて来たときのボールの位置と受け皿の位置をそれぞれ示す．このとき $z_{\rm h}^{\rm b})^{\rm T} = z_{\rm h}^{\rm c})^{\rm T}$ である．図2.3.5

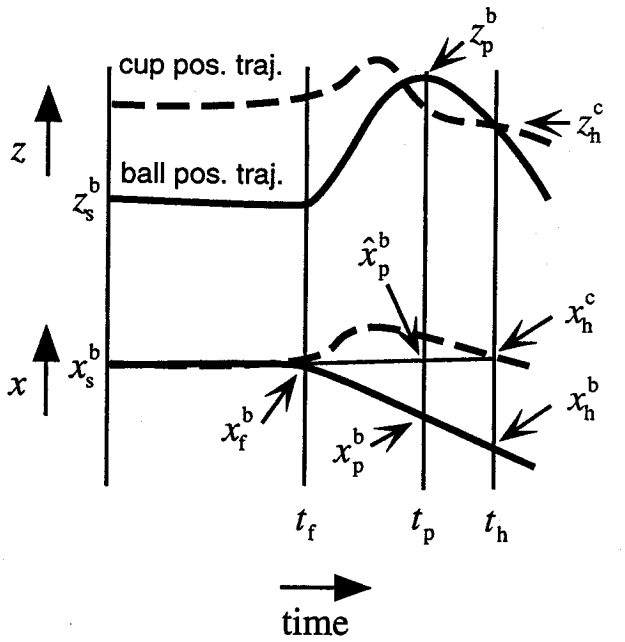


図 2.10: タスク表現の模式図．実線はボールの位置の時間変化，破線は受け皿の位置の時間変化をそれぞれ表す．x, z は三次元位置の左右方向 (右方向正)，上下方向 (上方向正) をそれぞれ示す．y は省略してある．$t_{\mathrm{f}}, t_{\mathrm{p}}, t_{\mathrm{h}}$ はボールが自由落下を始めたとき，ボールが最も高く上がったとき，ボールが受け皿の高さまで落ちて来たときの時間をそれぞれ表す．$x_{\mathrm{s}}^{\mathrm{b}}, z_{\mathrm{s}}^{\mathrm{b}}$ はボールの初期位置，$x_{\mathrm{f}}^{\mathrm{b}}, z_{\mathrm{f}}^{\mathrm{b}}$ はボールが自由落下を始めたときの位置，$x_{\mathrm{p}}^{\mathrm{b}}, z_{\mathrm{p}}^{\mathrm{b}}$ はボールが最も高く上がったときのボールの位置，$x_{\mathrm{h}}^{\mathrm{b}}, z_{\mathrm{h}}^{\mathrm{b}}$ はボールが受け皿の高さまで落ちて来たときのボールの位置，$x_{\mathrm{h}}^{\mathrm{c}}, z_{\mathrm{h}}^{\mathrm{c}}$ はボールが受け皿の高さまで落ちて来たときの受け皿の位置をそれぞれ表す．$\hat{x}_{\mathrm{p}}^{\mathrm{b}}$ はボールが受け皿の上にちょうど落ちるようになるための目標位置を表す．

にここで用いるタスク表現の模式図を示す．しかし，このタスク表現をそのまま用いると以下のような不都合が生じる．

- ボールの飛ぶ高さを表現していない．
- ボールがSARCOSアームやけん玉のけんの影に隠れて，三次元位置計測装置でボールの位置の計測が不可能となる場合がある．

そこで，基本的には上のタスク表現と同等であるが，より実際的なタスク表現を以下で考える．

タスク変数 $\mathbf{T}^{\mathrm{prac}}$ をボールの初期位置と最も高く上がったときの距離とする．

$$\mathbf{T}^{\mathrm{prac}} = \begin{pmatrix} x_{\mathrm{p}}^{\mathrm{b}} - x_{\mathrm{s}}^{\mathrm{b}} \\ y_{\mathrm{p}}^{\mathrm{b}} - y_{\mathrm{s}}^{\mathrm{b}} \\ z_{\mathrm{p}}^{\mathrm{b}} - z_{\mathrm{s}}^{\mathrm{b}} \end{pmatrix}. \tag{2.22}$$

ここで $(x_{\mathrm{s}}^{\mathrm{b}}, y_{\mathrm{s}}^{\mathrm{b}}, z_{\mathrm{s}}^{\mathrm{b}})^{\mathrm{T}}$ はボールの初期位置，$(x_{\mathrm{p}}^{\mathrm{b}}, y_{\mathrm{p}}^{\mathrm{b}}, z_{\mathrm{p}}^{\mathrm{b}})^{\mathrm{T}}$ はボール最も高く上がったときのボールの位置をそれぞれ表す．したがって上記のタスク変数 $\mathbf{T}^{\mathrm{prac}}$ はボールが最も高く上がったときの初期位置からの左右，前後，高さのずれをそれぞれ表す．$\mathbf{T}$ をけん玉成功に近付けるためには，ボールが適切な高さに上がり，受け皿の中にちょうど入るような角度で投げ上げられればよい．したがってタスク目標 $\mathbf{T}_{\mathrm{desired}}$ を以下のようにした．

$$\mathbf{T}_{\mathrm{desired}}^{\mathrm{prac}} = \begin{pmatrix} x_{\mathrm{desired}} \\ y_{\mathrm{desired}} \\ z_{\mathrm{desired}} \end{pmatrix} = \begin{pmatrix} \hat{x}_{\mathrm{p}}^{\mathrm{b}} - x_{\mathrm{s}}^{\mathrm{b}} \\ \hat{y}_{\mathrm{p}}^{\mathrm{b}} - y_{\mathrm{s}}^{\mathrm{b}} \\ \mathrm{Const.} \end{pmatrix}, \tag{2.23}$$

$$\hat{x}_{\mathrm{p}}^{\mathrm{b}} = x_{\mathrm{s}}^{\mathrm{b}} + (x_{\mathrm{h}}^{\mathrm{c}} - x_{\mathrm{f}}^{\mathrm{b}}) r_{\mathrm{t}}, \tag{2.24}$$

$$\hat{y}_{\mathrm{p}}^{\mathrm{b}} = y_{\mathrm{s}}^{\mathrm{b}} + (y_{\mathrm{h}}^{\mathrm{c}} - y_{\mathrm{f}}^{\mathrm{b}}) r_{\mathrm{t}}, \tag{2.25}$$

$$r_{\mathrm{t}} = \frac{t_{\mathrm{p}} - t_{\mathrm{f}}}{t_{\mathrm{h}} - t_{\mathrm{f}}}. \tag{2.26}$$

ここで $x_{\mathrm{h}}^{\mathrm{c}}, y_{\mathrm{h}}^{\mathrm{c}}$ はボールが受け皿と同じ高さまで落ちてきたときの受け皿の位置を表す．$x_{\mathrm{f}}^{\mathrm{c}}, y_{\mathrm{f}}^{\mathrm{c}}$ はボールが引き上げられて，自由落下を始めたときのボールの水平位置を表す．$t_{\mathrm{f}}, t_{\mathrm{p}}, t_{\mathrm{h}}$ はボールが自由落下を始めたとき，ボールが最も高く上がったとき，ボールが受け皿と同じ高さまで落ちたときをそれぞれ表す．ボールが自由落下の状態にあるときは x, y 平面内で直線の軌跡を描くので，線形補間を用いることができる．$\mathbf{T}_{\mathrm{desired}}^{\mathrm{prac}}$ の z 成分，すなわち高さに関しては，ボールが適当な高さに上がるように定数を設定した．本実験では 0.5 ～ 0.8 m に設定した．

2.3.6 経由点修正

本実験では，学習スキームとして広義ニュートン法を用いた．N 個の経由点の集合を

$$\mathbf{S} = \{\tilde{\mathbf{X}}^1_{\text{via}}, \tilde{\mathbf{X}}^2_{\text{via}}, \cdots, \tilde{\mathbf{X}}^N_{\text{via}}\} \tag{2.27}$$

とする．n 回目の学習での経由点の修正は

$$\mathbf{S}_{n+1} = \mathbf{S}_n + (\partial \mathbf{T}_n / \partial \mathbf{S}_n)^{\#} \mathbf{T}_{\text{gain}} (\mathbf{T}^{\text{prac}}_{\text{desired}} - \mathbf{T}^{\text{prac}}_n) \tag{2.28}$$

となる．ここで $\#$ は単純正則化一般逆行列 [71, 73]

$$(\partial \mathbf{T}/\partial \mathbf{S})^{\#} = A^{\#} = (A^T A + kI)^{-1} A^T \tag{2.29}$$

を用いた．これは

$$\|I - AA^{\#}\|^2 + k\|A^{\#}\|^2 \tag{2.30}$$

を最小にする逆変換行列であり，第 1 項は逆変換が正確に行われることを要求し，第 2 項は一度の修正が大きくならないことを要求する．本実験では安全のため $k = 1.0$ とした．$\mathbf{T}_{\text{gain}}$ は，けん玉運動では前後左右の運動に比べて上下方向の運動が大きいので誤差の収束量をそろえるために付加したもので，

$$\mathbf{T}_{\text{gain}} = \begin{pmatrix} 0.2 & 0 & 0 \\ 0 & 1.0 & 0 \\ 0 & 0 & 1.0 \end{pmatrix} \tag{2.31}$$

のように設定した．手先位置の経由点の初期値はヒトのけん玉運動軌道から抽出した経由点

$$\mathbf{S}_0 = \{\mathbf{X}^1_{\text{via}}, \mathbf{X}^2_{\text{via}}, \cdots, \mathbf{X}^N_{\text{via}}\}$$

とした．

$\partial \mathbf{T}_n / \partial \mathbf{S}_n$ を解析的に求めることは困難なので，S_0 のまわりで各経由点を $+\Delta \mathbf{X}^i_{\text{via}}$ および $-\Delta \mathbf{X}^i_{\text{via}}$ ずつ動かした時の $\mathbf{T}_{\text{real}}$ の変化 $\Delta \mathbf{T}_{\text{real}}$ を観測することによって求めた

$$\frac{\Delta \mathbf{T}_{\text{real}}}{\Delta \mathbf{X}^i_{\text{via}}} = \frac{\mathbf{T}^+_{\text{real}} - \mathbf{T}^-_{\text{real}}}{2\Delta \mathbf{X}^i_{\text{via}}} \tag{2.32}$$

で代用した．これはタスクを経由点で数値微分したものとなり，各学習毎に得られた軌道 $\tilde{X}_{\text{via}}$ のまわりで毎回測定することが望ましい．しかし，各学習毎に毎回測定するには非常に多数の試行が必要となり，現実的でない．したがって，ここでは最初の動きが最適な動きに充分近いものと仮定して，すべての学習に上で求めたものを用いた．各経由点のうち，数値微分を求めたのは三次元位置のみで，手先の向きに関しては修正を行わないものとした．第 6, 7 番目の経由点は落ちてくるボールを受け止める付近であり，これらの経由点を修正すると学習の収束が非常に遅くなったので第 6, 7 番目の経由点に関しても修正を行わないこととした．

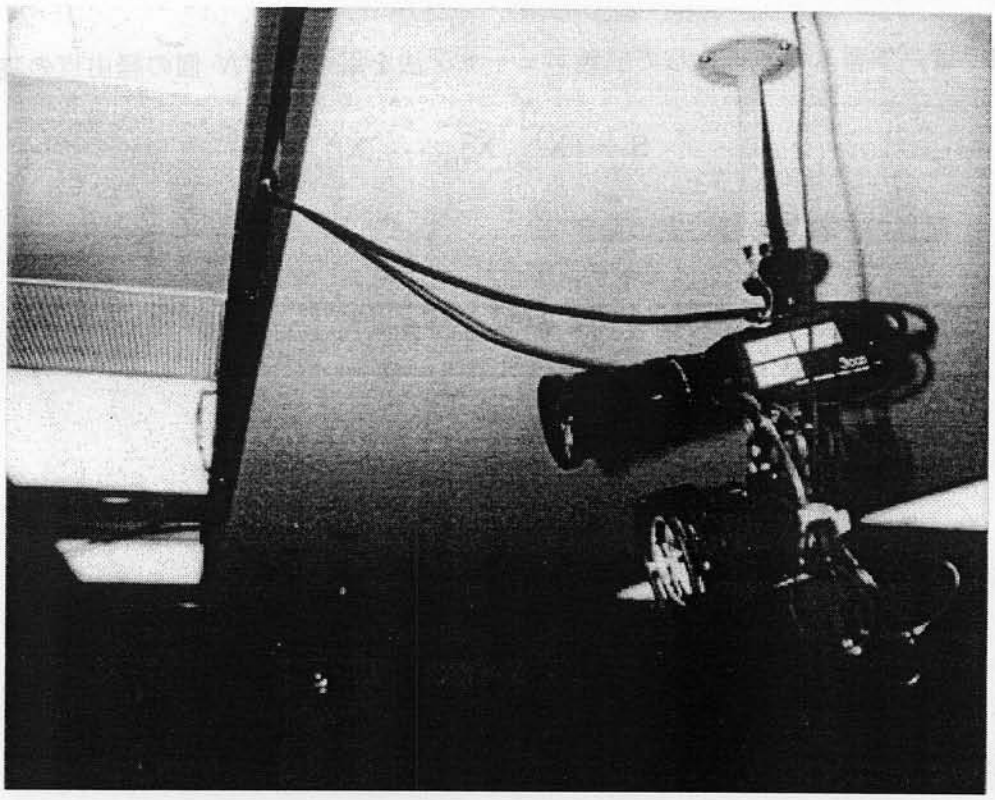


図 2.11: QUICKMAG のカメラ部 (応用計測株式会社製).

図 2.12: QUICKMAG のコントローラ部 (応用計測株式会社製).

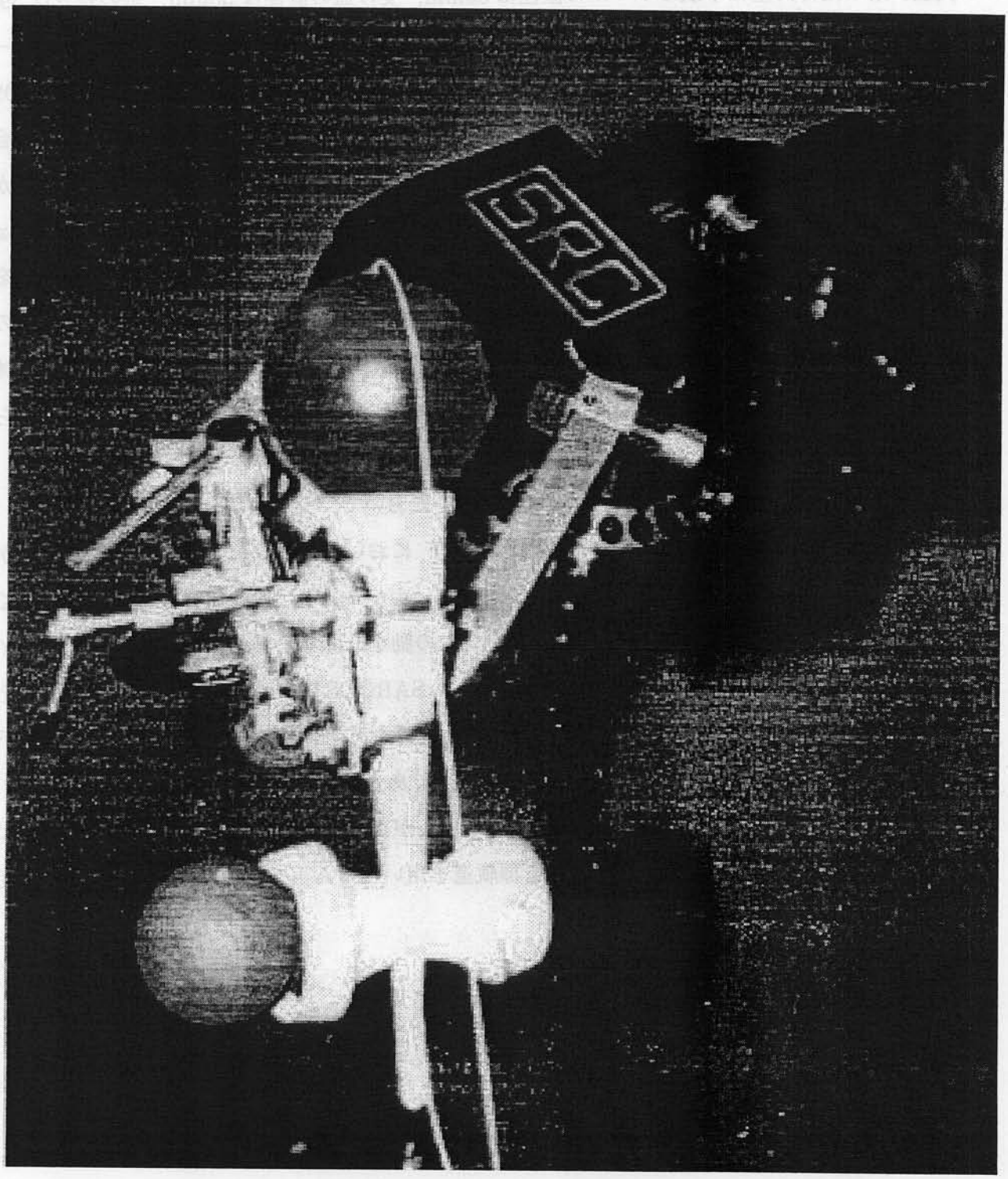


図 2.13: SARCOS アームの手先に固定したけん玉．けんは動作終了時の姿勢で受け皿がほぼ水平になるように小さい万力で SARCOS アームの手先に固定した．受け皿の三次元位置を QUICKMAG で測定するために緑色のマーカーがけんに付けられている．ボールの直径は 58mm，受け皿の直径は 33mm である．

2.3.7 SARCOS アームによるけん玉学習

実験に用いたけん玉は木製でボールの直径は 58mm，受け皿の直径は 33mm，糸の長さは 395mm である．けんは動作終了時の姿勢で受け皿がほぼ水平になるように小さい万力で SARCOS アームの手先に固定した．図 2.13 に SARCOS アームの手先に固定したけん玉を示す．SARCOS アームで遂行中のけん玉の受け皿の位置とボールの位置の軌道は QUICKMAG(応用計測株式会社製) と呼ばれる三次元位置計測装置によって計測した．これは図 2.11 に示す二つのカメラから得られた画像から，特定の色の面積重心の三次元位置を求める装置で，サンプリングは 60Hz，精度は数ミリ程度である．本実験では赤色のボールを用い，受け皿の近くに緑色のマーカーを張り付けた．計測されたボールと受け皿の三次元位置はパラレルポートから SARCOS アームのコントローラに送られ，500Hz でサンプリングされた後，遮断周波数 5Hz のバターワース低域通過フィルタを通して軌道データとして得られる．得られた受け皿とボールの位置の軌道から手先位置の経由点 $\tilde{X}^i_{\mathrm{via}}$ を式 (2.40) に従って修正する．

図 2.14 に学習前のけん玉の試行を示す．図 2.14上 において，実線はボールの三次元位置を，点線は受け皿の三次元位置をそれぞれ表す．図 2.14下 において，左は前面から，右は左側面からみた SARCOS アームの動きとボールの動きを示す．ボールと受け皿の位置の軌道は QUICKMAG によって計測されたもので，SARCOS アームの動きは関節角実現軌道から順キネマティクス方程式を用いて腕の姿勢に変換したものである．SARCOS アームの手先の軌道はヒトのそれと近いものとなっているが，前向き制御に用いた逆ダイナミクスとフィードバック制御の不完全性やけんの握り方の違い等によってボールがうまくあがらず，けん玉は成功しない．

図 2.15 に 7 回学習後のけん玉の試行を示す．ボールが適切に上がって，けん玉が成功するようになっている．ここで学習を止め，同じ目標軌道を用いてけん玉を試みると 10 回に数回程度の成功率が得られた．

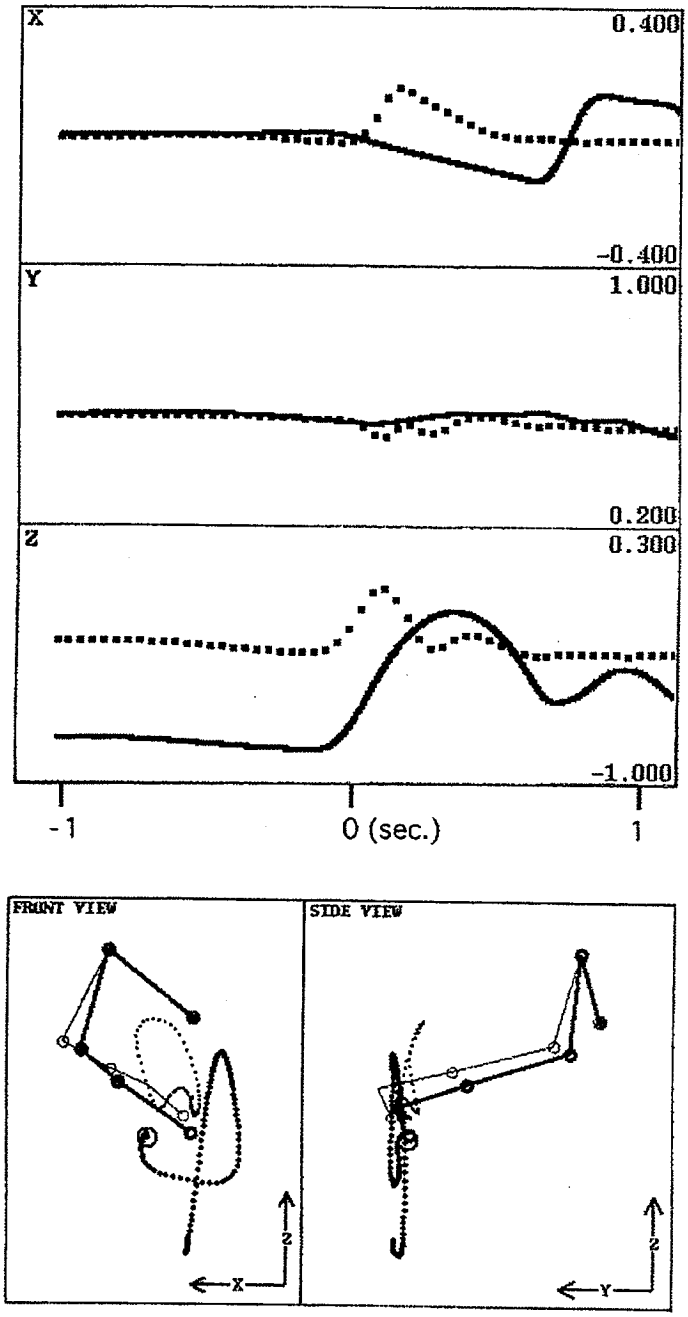


図 2.14: SARCOS アームによる学習前のけん玉の試行．上：実線はボールの三次元位置の時間変化，破線は受け皿の三次元位置の時間変化をそれぞれ示す．x は右方向正，y は前方向正，z は上方向正を表す．ボールと受け皿の三次元位置は QUICKMAG によって計測された．縦軸の単位はメートル．下：SARCOS アームの姿勢の三次元表示．細い線図は初期姿勢，太い線図は最終姿勢をそれぞれ表す．大きい丸はボールを表す．小さい丸は SARCOS アームの各関節を表す．SARCOS アームの姿勢は順キネマティクス方程式を用いて SARCOS アームの実現された関節角軌道から計算した．ボールの三次元位置は QUICKMAG によって計測された．

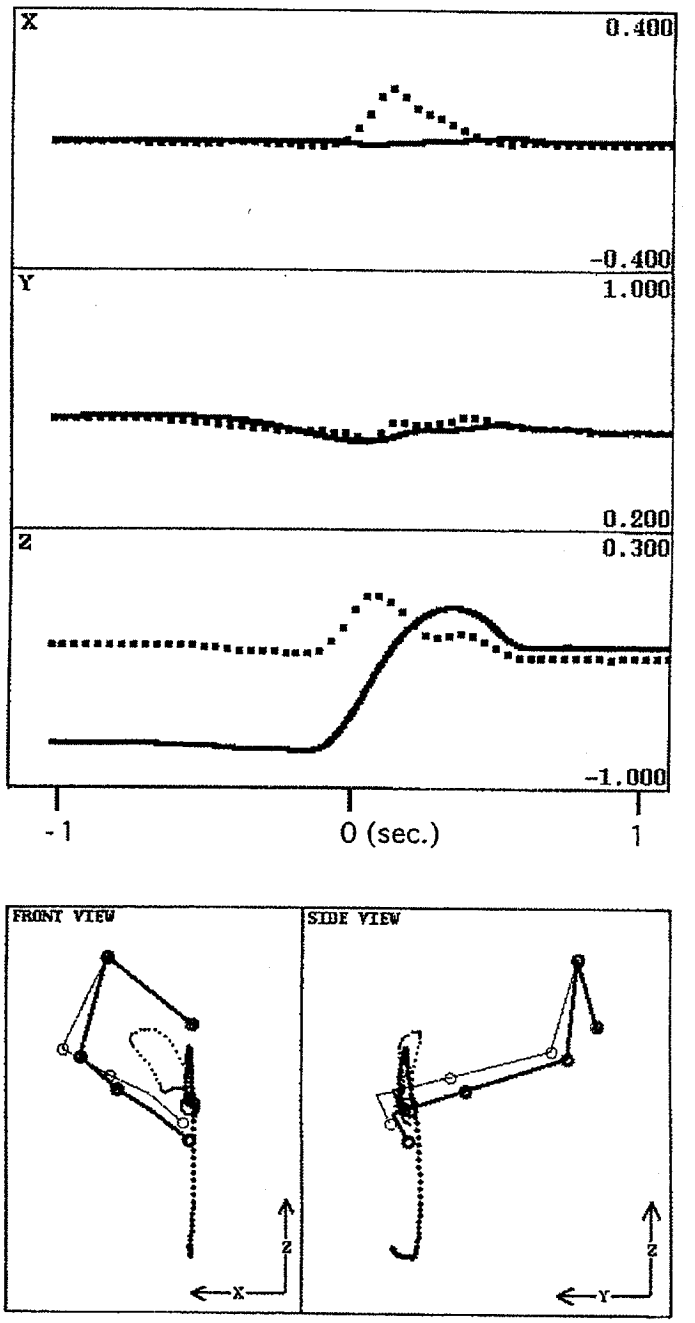


図 2.15: SARCOS アームによる 7 回学習後のけん玉の試行．図の説明は図 2.14 を参照．

2.4 SARCOS アームの数値モデルによるけん玉運動学習計算機シミュレーション

本節では，抽出された経由点がタスク表現として適切なものであるかを検討する．実機を用いたのでは、どのような動作となるか予測できず危険であるため、計算機シミュレーションを用いる．SARCOS アームの数値モデルによるけん玉運動学習計算機シミュレーションの概略を図 2.16 に示す．

2.4.1 SARCOS アームとけん玉の数値モデル

ここで制御対象をヒト腕と同じ 7 自由度を持つマニピュレータとする．数値実験では SARCOS Dextrous Slave Arm の数値モデルを用いた．

けん玉のボールと受け皿を数値計算により詳細に求めた研究があるが [19]，ここではけん玉のボールの軌道は以下に示すような簡単な質点系の力学モデルを用いて求めた．

$$\|\boldsymbol{F}\| = \left\| \begin{pmatrix} f_{\mathrm{x}} \\ f_{\mathrm{y}} \\ f_{\mathrm{z}} \end{pmatrix} \right\| = \begin{cases} K(l - l_0) & \text{if } l \geq l_0 \\ 0 & \text{otherwise} \end{cases}, \tag{2.33}$$

$$l = \sqrt{(x_{\mathrm{h}} - x)^2 + (y_{\mathrm{h}} - y)^2 + (z_{\mathrm{h}} - z)^2}, \tag{2.34}$$

$$m\ddot{x} = -f_{\mathrm{x}} = -\|\boldsymbol{F}\| \, (x_{\mathrm{h}} - x)/l, \tag{2.35}$$

$$m\ddot{y} = -f_{\mathrm{y}} = -\|\boldsymbol{F}\| \, (y_{\mathrm{h}} - y)/l, \tag{2.36}$$

$$m\ddot{z} = -f_z - mg = -\|\boldsymbol{F}\| \, (z_{\mathrm{h}} - z)/l - mg, \tag{2.37}$$

ここで K は糸の弾性係数を表す．$\boldsymbol{F}$ は糸にかかる力のベクトルを表す．l と l_0 は糸の長さと，自然長をそれぞれ表す．$x_{\mathrm{h}}, y_{\mathrm{h}}, z_{\mathrm{h}}$ はけんに取り付けられた糸の根本の三次元位置を表す．m はボールの質量，g は重力加速度を表す．x, y, z はボールの三次元位置を表す．$m\ddot{x}, m\ddot{y}, m\ddot{z}$ はボールの加速度を表す．これらの方程式はオイラー法で数値積分される．時間刻みは 2ms とした．

2.4.2 タスクの表現と経由点修正

実現されたタスクを $\mathbf{T}$ で表す．けん玉の成功のために重要と考えられるタスク表現 $\mathbf{T}$ を以下に示す．

$$\mathbf{T} = \begin{pmatrix} z_{\mathrm{peak}} \\ x_{\mathrm{peak}} \\ y_{\mathrm{peak}} \end{pmatrix}. \tag{2.38}$$

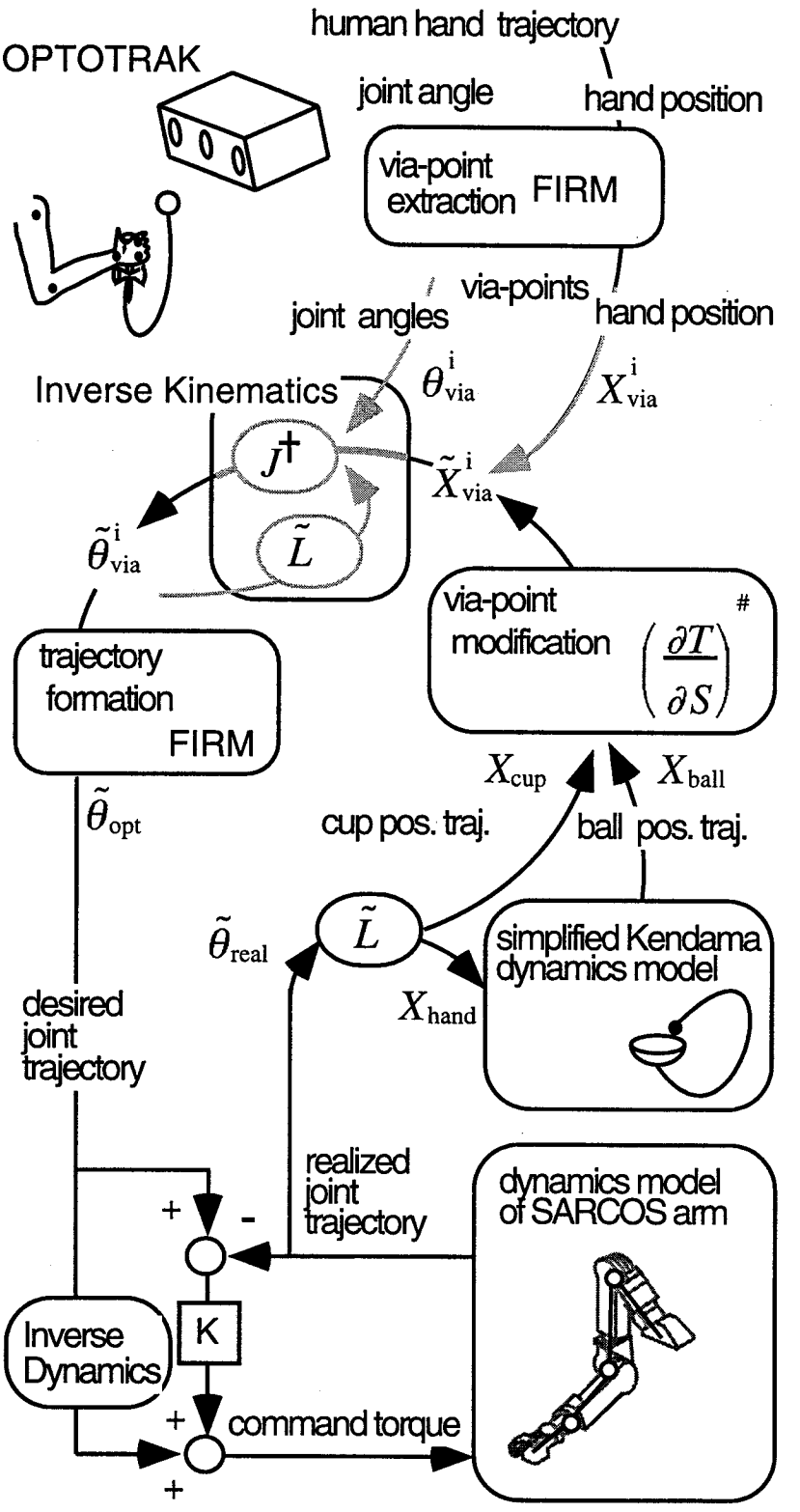


図 2.16: けん玉学習シミュレーションシステムの模式図.

ここで $z_{\mathrm{peak}}, x_{\mathrm{peak}}, y_{\mathrm{peak}}$ はそれぞれ，ボールが最も上がったときの初期位置からの高さ，左右のずれ，前後のずれを表す．$\mathbf{T}$ をけん玉成功に近付けるためには，ボールが適切な高さに上がり，受け皿の中心にちょうど入るような角度で投げ上げられればよい．$\mathbf{T}$ をタスク目標 $\mathbf{T}_{\mathrm{desired}}$ に近付けるために，本数値実験では学習スキームとして広義ニュートン法を用いた．N 個の経由点の集合を

$$\mathbf{S} = \{\tilde{\mathbf{X}}^1_{\mathrm{via}}, \tilde{\mathbf{X}}^2_{\mathrm{via}}, \cdots, \tilde{\mathbf{X}}^N_{\mathrm{via}}\} \tag{2.39}$$

とする．

n 回目の学習での経由点の修正は

$$\mathbf{S}_{n+1} = \mathbf{S}_n + (\partial \mathbf{T}_n / \partial \mathbf{S}_n)^{\#} \mathbf{T}_{\mathrm{gain}} (\mathbf{T}_{\mathrm{desired}} - \mathbf{T}_n) \tag{2.40}$$

となる．ここで $\#$ は単純正則化一般逆行列 [71]

$$(\partial \mathbf{T} / \partial \mathbf{S})^{\#} = A^{\#} = (A^T A + kI)^{-1} A^T \tag{2.41}$$

を用いた．これは

$$\|I - AA^{\#}\|^2 + k\|A^{\#}\|^2 \tag{2.42}$$

を最小にする逆変換行列であり，第 1 項は逆変換が正確に行われることを要求し，第 2 項は一度の修正が大きくならないことを要求する．本数値実験では $k = 1.0$ とした．$\mathbf{T}_{\mathrm{gain}}$ は，けん玉運動では前後左右の運動に比べて上下方向の運動が大きいので誤差の収束量をそろえるために付加したもので，以下のように設定した．

$$\mathbf{T}_{\mathrm{gain}} = \begin{pmatrix} 0.2 & 0 & 0 \\ 0 & 1.0 & 0 \\ 0 & 0 & 1.0 \end{pmatrix}. \tag{2.43}$$

手先位置の経由点の初期値はヒトのけん玉運動軌道から抽出した経由点

$$\mathbf{S}_0 = \{\mathbf{X}^1_{\mathrm{via}}, \mathbf{X}^2_{\mathrm{via}}, \cdots, \mathbf{X}^N_{\mathrm{via}}\} \tag{2.44}$$

とした．

$\partial \mathbf{T}_n / \partial \mathbf{S}_n$ を解析的に求めることは困難なので，$\mathbf{S}_0$ の振幅を α 倍した軌道のまわりで各経由点を $+\Delta\mathbf{X}^i_{\mathrm{via}}$ および $-\Delta\mathbf{X}^i_{\mathrm{via}}$ ずつ動かした時の $\mathbf{T}_{\mathrm{real}}$ の変化 $\Delta\mathbf{T}_{\mathrm{real}}$ を観測することによって求めた

$$\frac{\Delta \mathbf{T}_{\mathrm{real}}}{\Delta \mathbf{X}^i_{\mathrm{via}}} = \frac{\mathbf{T}^+_{\mathrm{real}} - \mathbf{T}^-_{\mathrm{real}}}{2\Delta \mathbf{X}^i_{\mathrm{via}}} \tag{2.45}$$

で代用した．これはタスクを経由点で数値微分したものとなり，各学習毎に得られた軌道 $\tilde{\mathbf{X}}_{\mathrm{via}}$ のまわりで毎回測定することが望ましいが，$\mathbf{S}_0$ の振幅を α 倍した軌道が最適な動きに充分近

いものと仮定して，すべての学習に上で求めたものを用いた．ボールが上がるとき，最も高く上がりきる前にSARCOSアームの腕部分や受け皿などに当たってしまうと正確な $\Delta \mathbf{T}_{\mathrm{real}}$ が得られないので，α はボールが受け皿よりも十分低い高さに上がるように決めた．各経由点のうち，数値微分を求めたのは三次元位置のみで，手先の向きに関しては修正を行わないものとした．

2.4.3 複数の被験者のデモンストレーション

図2.17 に学習前のけん玉の試行を示す．図2.17左 はボールと受け皿の三次元位置の時間変化を表す．図2.17右 の左は前面から，右は左側面からみたSARCOSアームの動きとボールの動きを示す．SARCOSアームの手先の軌道はヒトのそれと近いものとなっているが，前向き制御に用いた逆ダイナミクスとフィードバック制御の不完全性やけんの握り方の違い等によってボールがうまくあがらず，けん玉は成功しない．図2.18 に5回学習後のけん玉の試行を示す．ボールが適切に上がって，けん玉が成功するようになっている．

図2.19，図2.20に別の被験者が行ったけん玉運動データを用いた5回学習後のけん玉の試行を示す．図2.19 ではけん玉成功に至っているが，図2.20 ではけん玉は成功していない．ボールが受け皿の真上に落ちてくるというタスク目標としてはほぼ収束しているものの，ボールを受け止める時刻でのボールと受け皿の相対速度が大きすぎるため，ボールが受け皿の外へこぼれてしまっている．ヒトのけん玉運動軌道データを収集する際，成功か失敗かの情報は記録していないが，もとのヒトの試行でけん玉が成功していないものと思われる．同じ被験者で，けん玉が成功したと思われる別のデータを用いるとSARCOSアームでも成功した．もとのヒトの試行でけん玉が失敗した場合のデータでも学習によりSARCOSアームにけん玉を成功させるには，ボールを受け止める時のボールと受け皿の相対速度を小さくするというタスク目標を新たに設定する必要がある．

2.4.4 経由点の数を変えた場合

経由点の個数は少なすぎると観測されたヒト腕の運動と再構成された最適軌道の誤差が大きくなり，もとのヒト腕の運動軌道を精度良く近似できない．逆に制御すべき経由点の個数が多すぎると問題が不必要に複雑になる．これまでは，経由点の個数を7に固定していたが，7個が最適であるとは限らない．本節では経由点の個数を変えた場合を調べる．

経由点の個数が4個の場合を図2.22左 に示す．ボールが上がる途中で手先に当たってしまい，けん玉は成功に至らない．上がってくるボールを避けるために手先を右 (x の正方向) に振る動作がうまく再現されていないからと考えられる．

経由点の個数が5個から9個の場合は5回以内の学習で，いずれもけん玉成功に至っていた．

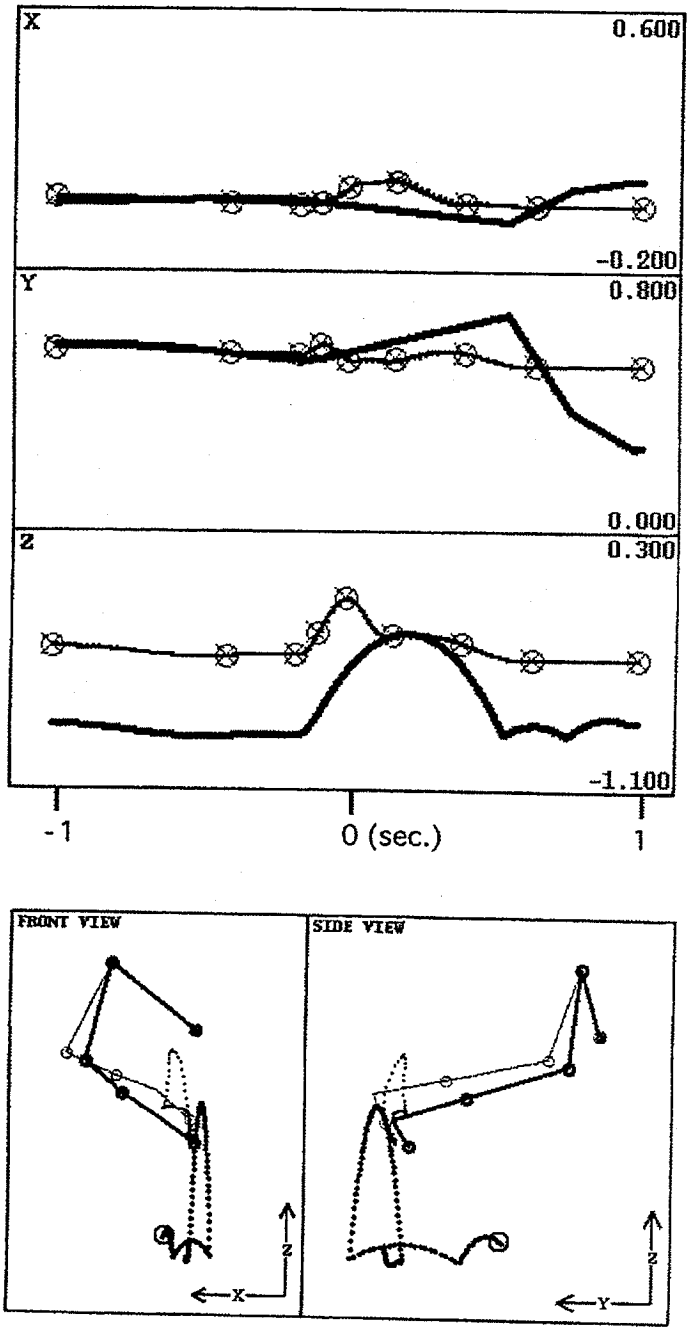


図 2.17: 学習前のけん玉の試行，被験者 H.M.. 上：実線はボールの三次元位置の時間変化，破線は受け皿の三次元位置の時間変化をそれぞれ示す．x は右方向正，y は前方向正，z は上方向正を表す．縦軸の単位はメートル．下：SARCOS アームの数値モデルの姿勢の三次元表示．細い線図は初期姿勢，太い線図は最終姿勢をそれぞれ表す．大きい丸はボールを表す．小さい丸は SARCOS アームの各関節を表す．

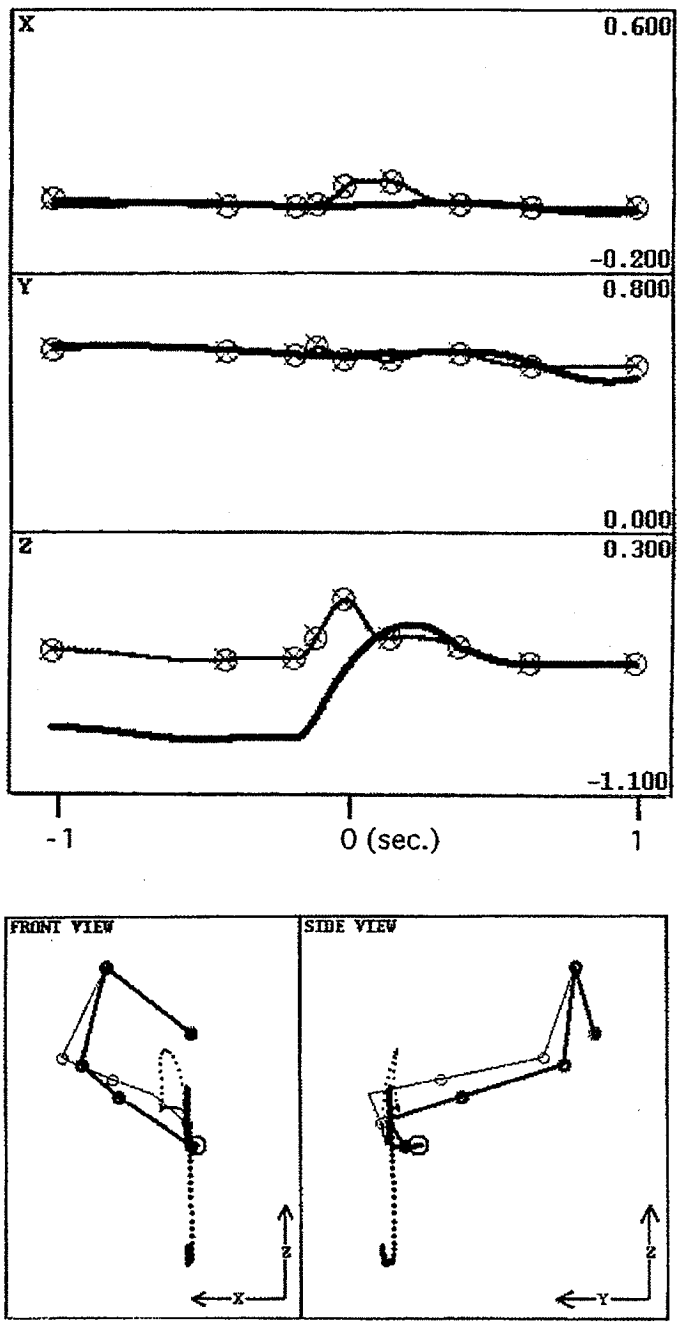


図 2.18: 5 回学習後のけん玉の試行，被験者 H.M.. 図の説明は図 2.17 を参照.

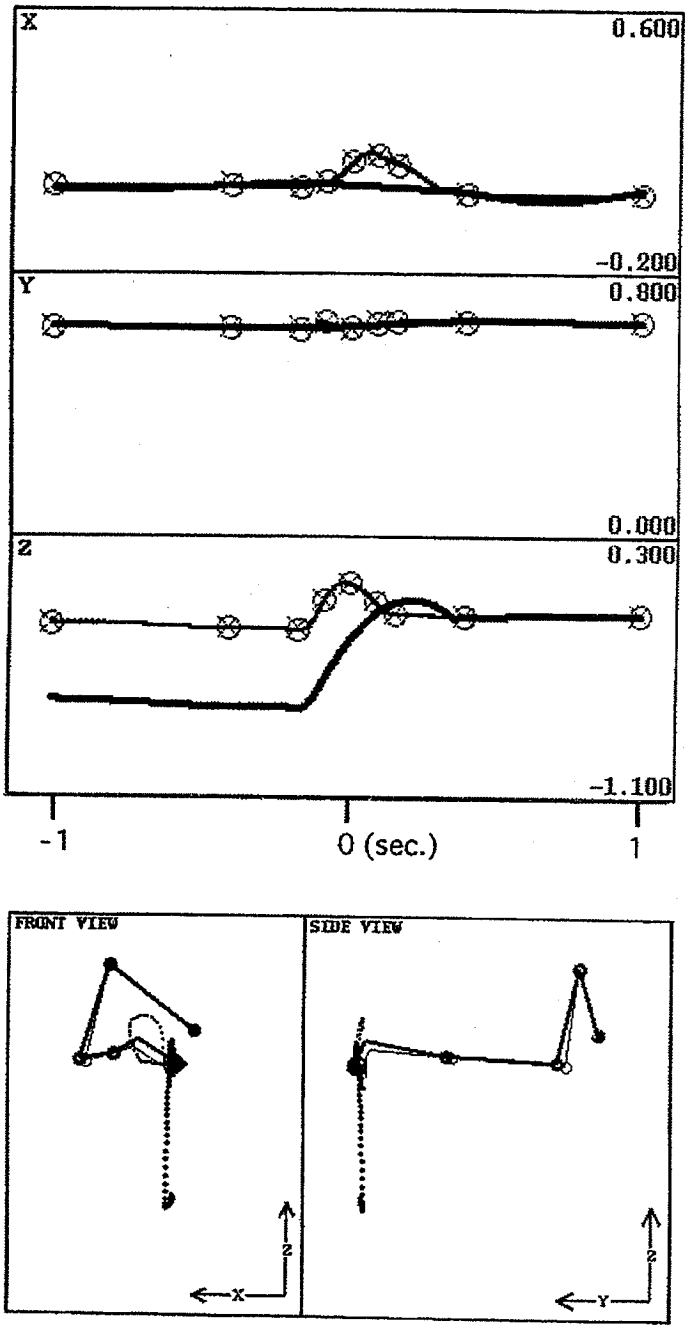


図 2.19: 5 回学習後のけん玉の試行，被験者 H.W.. 図の説明は図 2.17 を参照.

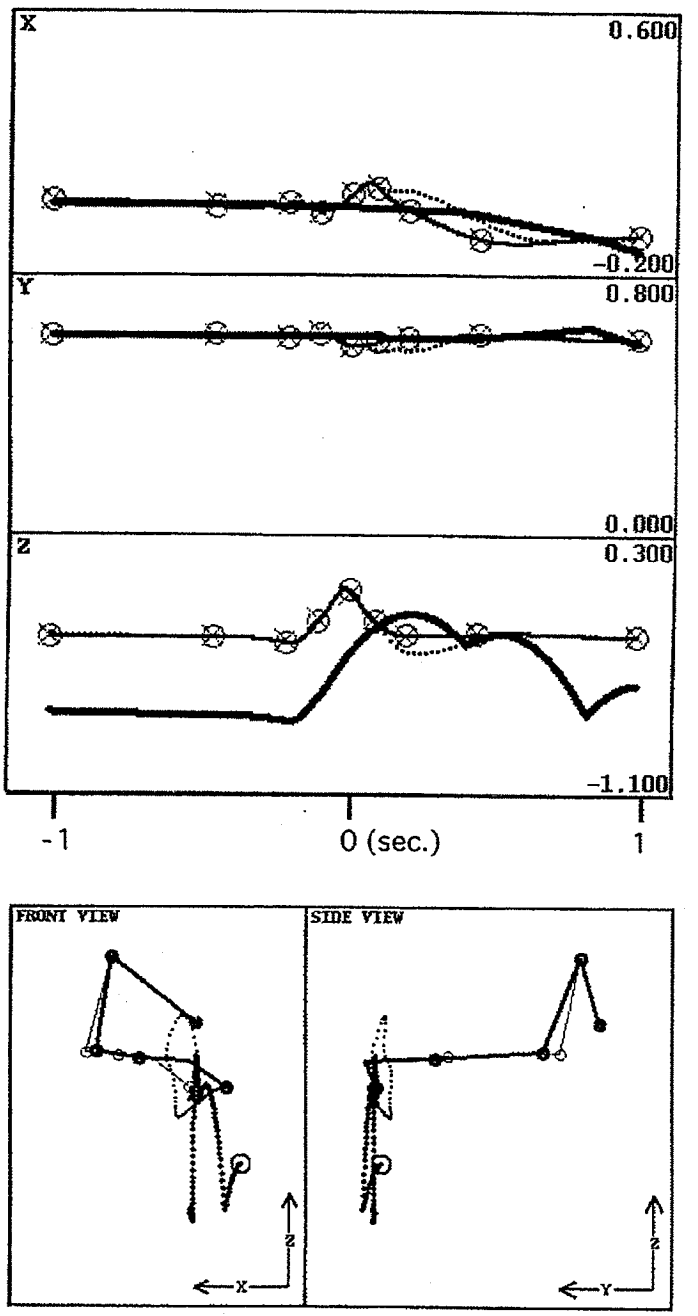


図 2.20: 5 回学習後のけん玉の試行，被験者 R.O.. 図の説明は図 2.17 を参照.

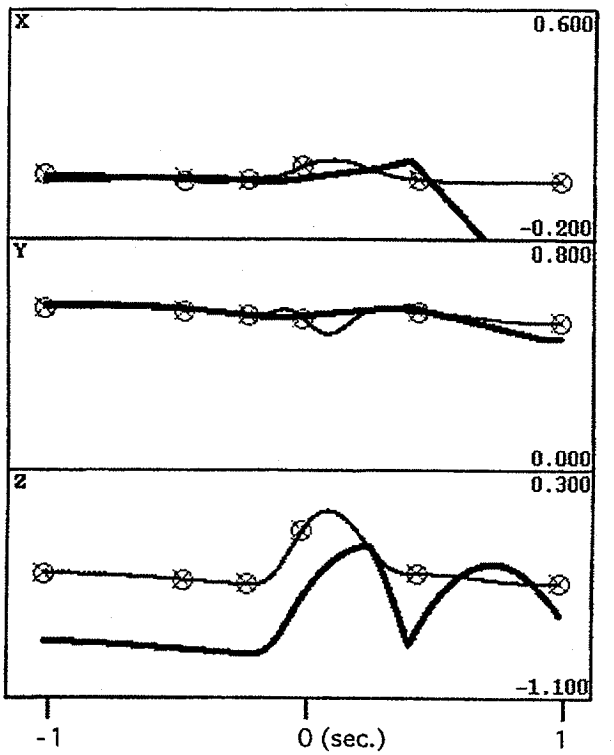


図 2.21: 経由点の数を 4 個とした場合の学習後のけん玉の試行．図の説明は図 2.17 を参照．

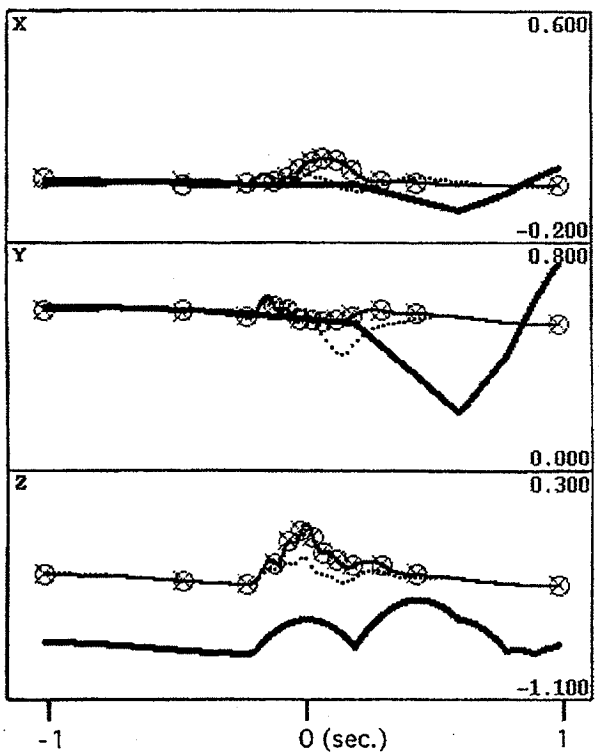


図 2.22: 経由点の数を 11 個とした場合の学習後のけん玉の試行．図の説明は図 2.17 を参照．

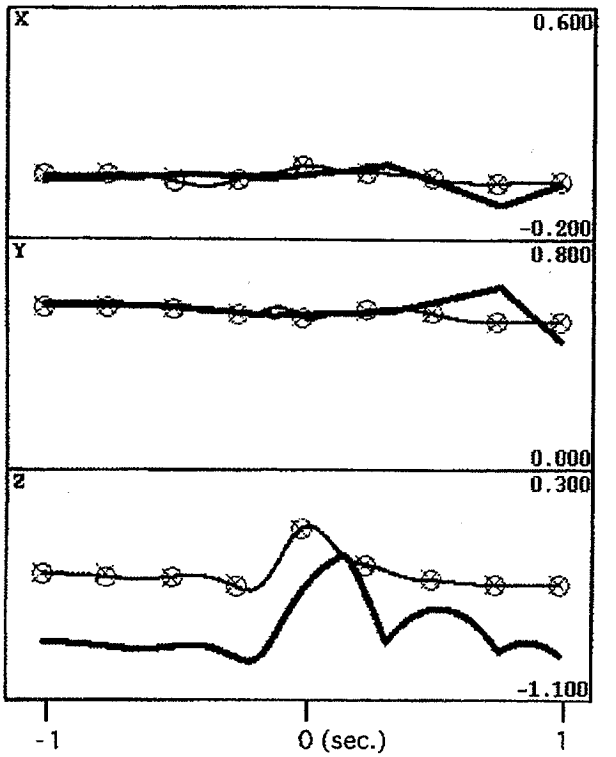


図 2.23: 経由点を等時間間隔にした場合の学習後のけん玉の試行．図の説明は図 2.17 を参照．

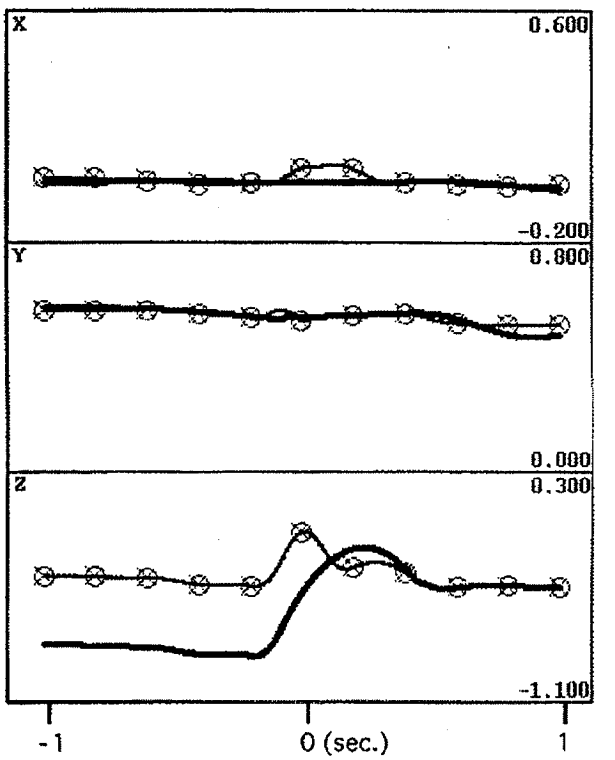


図 2.24: 経由点を等時間間隔にした場合の学習後のけん玉の試行．経由点の数を増やした場合の学習後のけん玉の試行．図の説明は図 2.17 を参照．

経由点の個数が10個の場合はけん玉成功に至っていたが，学習回数は10回必要とした．

経由点の個数が11個の場合を図2.22右 に示す．目標軌道がぎざぎざになってしまい，実現軌道の誤差もかなり大きくなっていてけん玉は成功しない．原因は未検討であるが，$\partial \mathbf{T}_n/\partial \mathbf{S}_n$ を求める際 $\mathbf{S}_0$ の振幅を α 倍した軌道のまわりで求めたので，正確な $\partial \mathbf{T}_n/\partial \mathbf{S}_n$ でないことが考えられる．また個々の経由点が互いに独立でなく互いに干渉しており，経由点の個数が多すぎると時間的に隣あう経由点どうしの時間間隔が狭くなり，経由点同士の相互作用が無視できなくなった可能性もある．

経由点の最適な個数を決定する合理的な方法は検討していないが，タスク成功のために要求される制御と測定の精度と，もとのヒトの運動軌道と再構成された最適軌道との誤差等を考慮に入れて経由点の最適な個数を決定する必要があると考えられる．

2.4.5 等時間間隔の経由点の場合

FIRMでは，もとのヒトの運動軌道と再構成された最適軌道との二乗誤差の大きい時刻付近で経由点が抽出される．本節ではFIRMの経由点抽出方法が有効であるかどうかを検討する．

図2.23 に等時間間隔で経由点を抽出し，経由点の個数を7個とした場合を示す．上がろうとするボールを手先がよけ切れずけん玉は成功していない．

図2.24 に等時間間隔で経由点を抽出し，経由点の個数を9個とした場合を示す．この場合にはけん玉は成功に至っている．

等時間間隔で経由点を抽出した場合，FIRMで抽出した場合に比べて経由点の個数が多い場合はけん玉は成功している．しかしFIRMで抽出した場合と同程度の経由点の個数の場合はけん玉は成功しない．

今回の実験では測定したヒト腕の運動軌道データを手先が最も高くなる (z 正方向のピーク) 時刻が運動時間全体の中心になるように揃えている．したがって，経由点を等時間間隔で抽出した場合，経由点の個数が偶数ではもとのヒトの運動軌道からの誤差が大きくなる．上の理由から，経由点の個数が偶数の場合は学習不可能であった．経由点の個数を9個とした場合はけん玉は成功している．この場合は，たまたま運動の特徴をよく表す時刻の近傍に経由点が選ばれたと考えられる．

2.4.6 経由点によるタスクの可操作性

本研究では経由点を制御変数とみなしタスク目標の達成を試みているが，操作能力のひとつの目安として

$$\sqrt{\det AA^T} \tag{2.46}$$

表 2.1: 各条件での可操作度.

経由点の個数	$\sqrt{\det AA^T}$	
	FIRM	等時間間隔
3	0.09	6.61
4	102.99	0.00
5	62.63	51.35
6	93.87	0.00
7	43.33	46.36
8	41.60	0.46
9	73.30	39.21
10	40.34	7.23
11	99.49	78.67

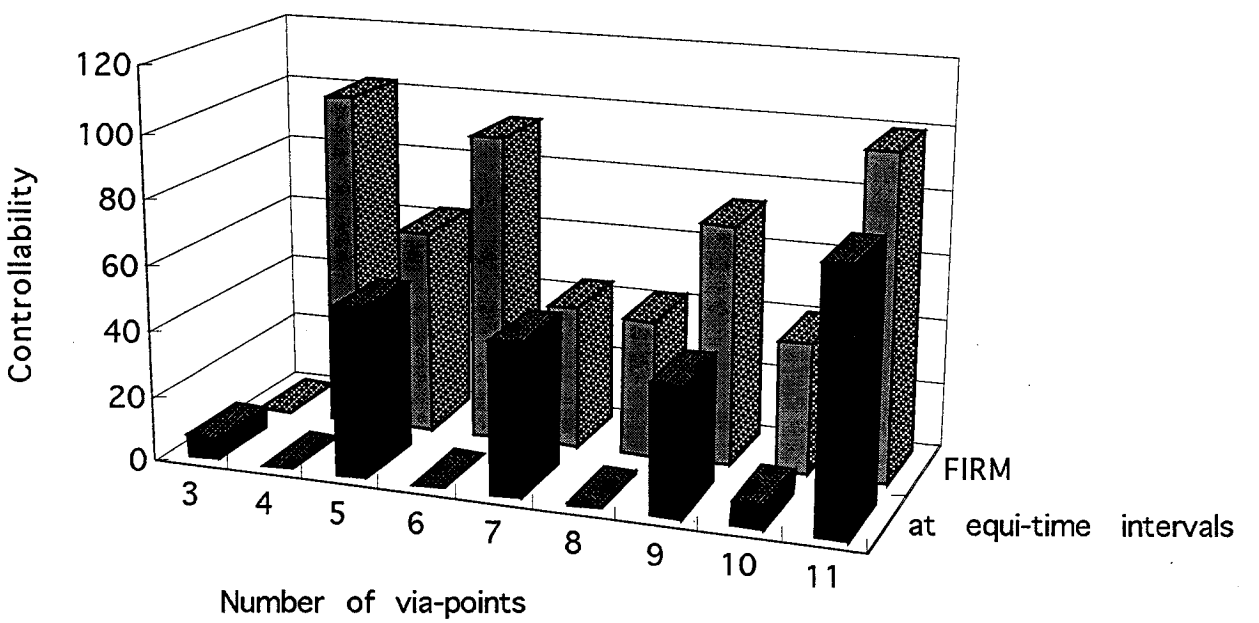


図 2.25: 各条件での可操作度.

を経由点によるタスクの可操作度と考えることとする．ただし $A = \partial T_0/\partial S_0$ である．$\sqrt{\det AA^T}$ の値が大きいほうがタスクの操作性が高いと考えられる．表 2.1 に各条件での $\sqrt{\det AA^T}$ の値を示す．

FIRM を用いて経由点を抽出した場合，経由点の個数が 3 個の場合は可操作度が非常に小さい値となっている．経由点の個数が少なすぎてもとのヒトの運動軌道をよく再現できず，経由点を修正してもボールの動きを制御することは非常に困難である．また $\sqrt{\det AA^T}$ の値が非常に小さいので，式 (2.40) の単純正則化一般逆行列において，逆変換が正確に行われない．上記の理由から，経由点の個数が 3 個の場合はけん玉を成功させる学習が不可能である．経由点の個数が 4 個の場合は可操作度が大きい値となっていて，ボールをうまく上げるというタスク目標そのものはうまく学習可能である．しかし，もとのヒト運動軌道のうち手先を左 (x の負方向) に振る動作がうまく再現されず，けん玉は成功しなかった．経由点の個数が 5 個から 10 個の場合はけん玉成功に至っていた．経由点の個数が 11 個の場合は可操作度は高い数値を示しているものの，学習の結果得られた目標軌道が振動してよい制御ができず，けん玉成功に至らなかった．これは経由点の個数が必要以上に多すぎ，高精度に $\partial T_n/\partial S_n$ を求められなかったこと等の理由が考えられる．

今回の実験では前節で述べたように，経由点を等時間間隔で抽出した場合，経由点の個数が偶数ではもとのヒトの運動軌道からの誤差が大きくなる．このことは，表 2.1 に示すように経由点の個数が偶数の場合は可操作度が極端に小さくなっていることに表れている．

以上の結果から，FIRM によって抽出された少数の経由点は運動の特徴をよく表現しているものと考えられる．

本章では，最適化原理に基づく運動パターン認識の理論がロボティクスにおける見まねによる運動学習に適用できることを示した．本実験では本方法の可能性を調べるためにけん玉運動を採用したが，次章ではより複雑なタスクとしてテニスサーブを採り上げる．

これまでは視覚フィドバックを積極的には用いていない，すなわち，一回の試行を行なった後，オフラインでタスク誤差を計測するためにのみ視覚フィドバックを用いてきた．人間がけん玉を行なうときには，玉が飛んでいる間に視覚による情報を用いて玉を受け止めるための軌道を修正している．生物はこのような修正をどのようにして行なっているのであろうか? サルと人間の腕の到達運動に関してさまざまな研究が行なわれている (例えば [21])．また到達運動途中で目標が突然変更された時の応答を説明するための理論的なモデルもいくつか提案されている (例えば [51, 16, 14, 25, 32])．これらの理論的な研究の成果を，視覚フィードバックを我々のモデルに組み込むことが今後必要と考える．

第 3 章

連続運動への拡張

3.1 はじめに

本研究では，抽象的表現として経由点を用いたタスクレベル学習を試みている．第 2 章では，第 1 章の枠組を用いてロボットにダイナミックな運動であるけん玉を行わせることに成功した [38, 58, 59, 60, 53, 54, 57, 61]．けん玉実験では簡単なタスク表現が可能な最も簡単な運動を選び，タスク目標を達成させることができた．しかし第 1 章の枠組が汎用性をもち得るかどうかを確かめるためには，連続する複数の動作が含まれる階層的な運動や，非線形性の強い制御対象を扱う場合など，種々の運動における学習可能性を調べる必要がある．

本章以降では最適化原理に基づく軌道計画と経由点の枠組を用いた見まねによるロボットの運動学習について汎用性と学習可能性に関して調べる．そのためより複雑で困難な運動としてテニスサーブと振り子の振り上げの運動学習を試みる．

第 3.2 節の実ロボットによるテニスサーブ学習実験では，時間的に連続する二つのサブタスク (ボールの投げ上げと打撃) が相互に強く影響しあうため，けん玉に比べて学習が困難である．

第 3.2 節の実ロボットによるテニスサーブ学習実験では，経由点を試行錯誤で決定した．その選択は直観的で根拠のないものであった．第 3.3 節のロボットの数値モデルによるテニスサーブ学習シミュレーション実験では，より合理的な経由点の選択方法について検討する．

3.2 テニスサーブロボット実験

図 3.1 にテニスサーブ実験の概略を示す．ここで行なうテニスサーブは，ラケットの柄に取り付けたカップにのせたボールを放り上げ，落ちてきたボールを軽く打って約 2 メートル離れたゴールに入れる，という作業である．

本実験で行なうテニスサーブでは時間的に連続するボールの投げ上げと打撃の二つの動作が相互に強く影響しあうため，けん玉に比べて学習が困難である．そのため，テニスサーブ成功のために以下の方策を行った．

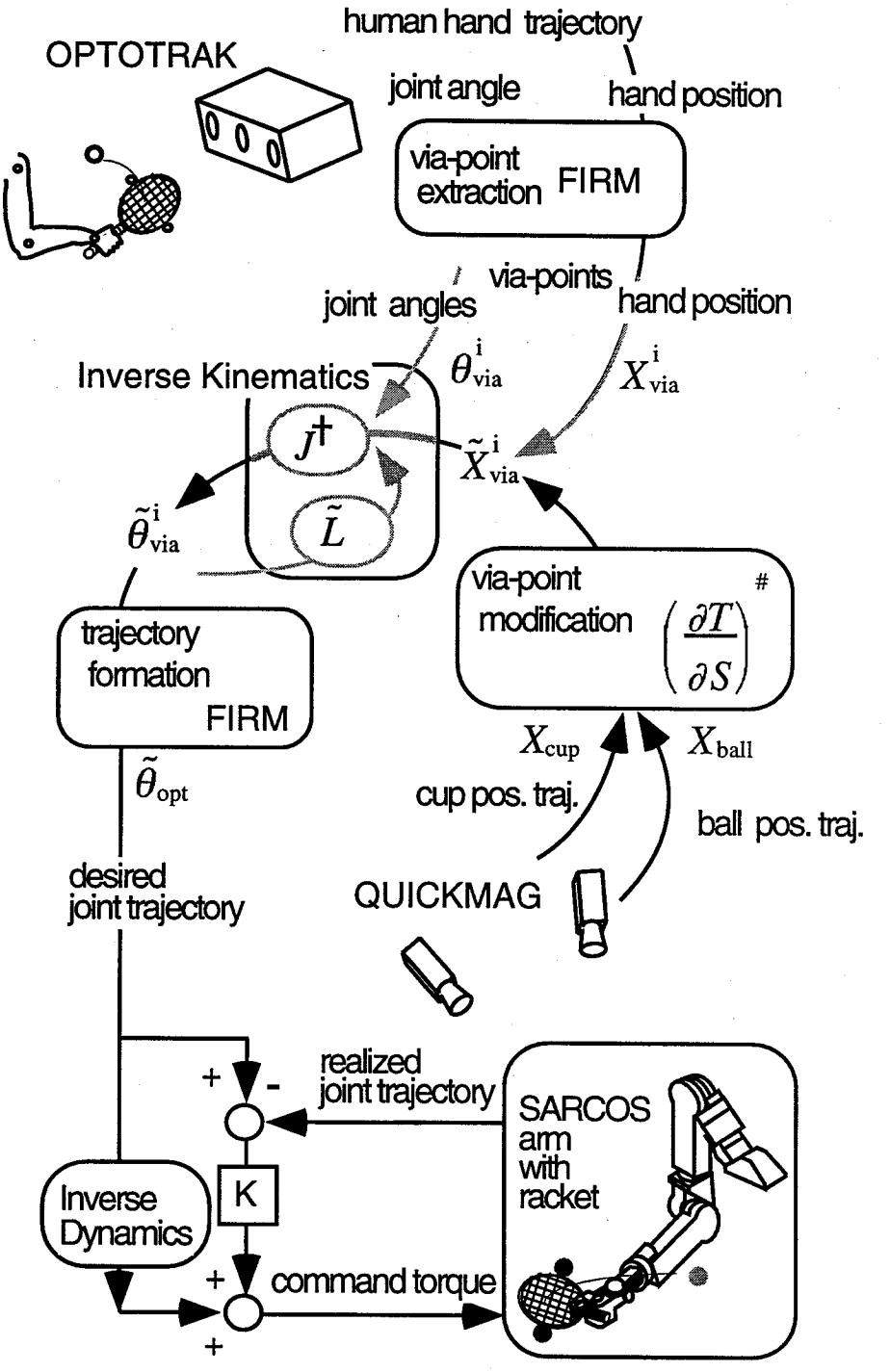


図 3.1: テニスサーブ学習実験システムの概略.

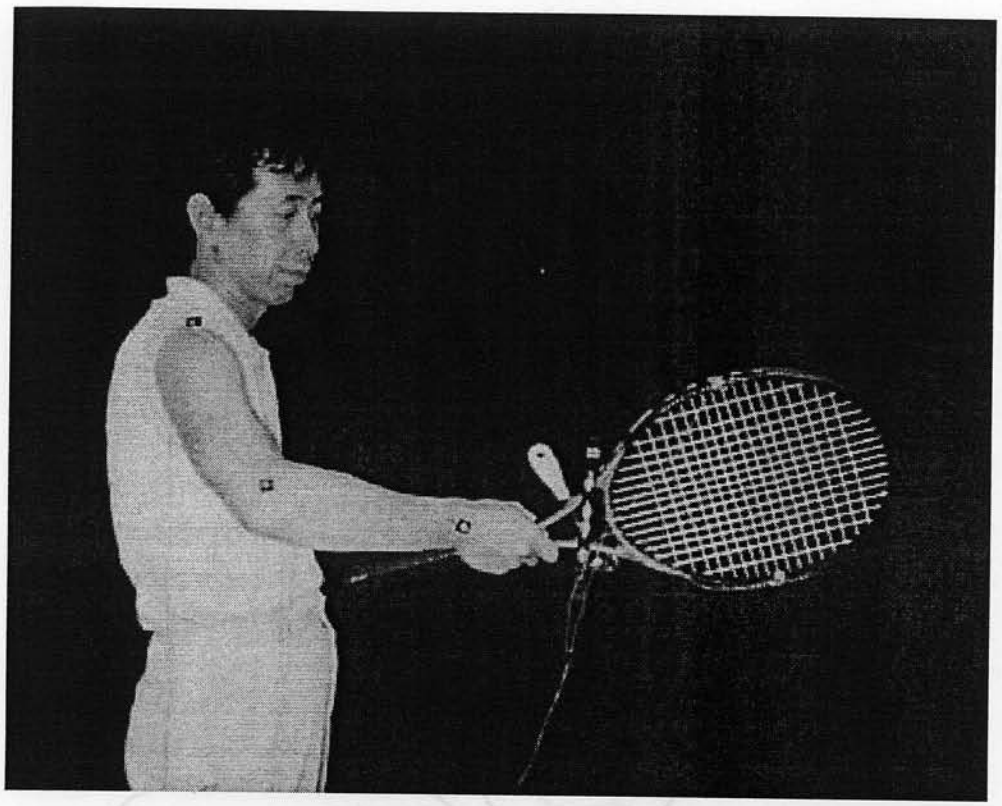


図 3.2: テニスラケットを持った被験者.

1. タスク目標をサブタスク 1 とサブタスク 2 に分ける.

2. 学習を 2 段階に分ける.

3. サブタスク間の相互影響を避けるため，学習させる経由点を制限する.

第 1 段階でボールを放りあげるサブタスク 1 を学習し，第 2 段階でボールをゴールの方向に打つというサブタスク 2 を学習する.

3.2.1 運動軌道計測と経由点抽出

図 3.2，図 3.3 で示すように，テニスサーブを行なうときの被験者の肩，肘，手首，ラケット面の両端の合計 5 箇所に赤外線 LED マーカーを取り付け，三次元位置計測装置 (OPTOTRAK) で計測した．本実験の計測条件では計測精度は 1mm 程度で，サンプリング周波数は 250Hz である．次にマーカーの位置軌道から，ラケット面中心の位置と向きの軌道を計算した．さらに，ラケット面中心の位置の軌道を数値微分し速度と加速度の軌道を得る．図 3.5 にラケット面中心の位置，速度，加速度の軌道を示す．ラケットでボールを打った瞬間ラケットに衝撃が加わるので，加速度が瞬間的に大きくなる時間をボールを打つべき時刻とした．上で得られたラケット面中心の位置と向きの軌道から経由点を抽出した．本論文では和田ら [83, 85, 84, 86] の FIRM(Forwa

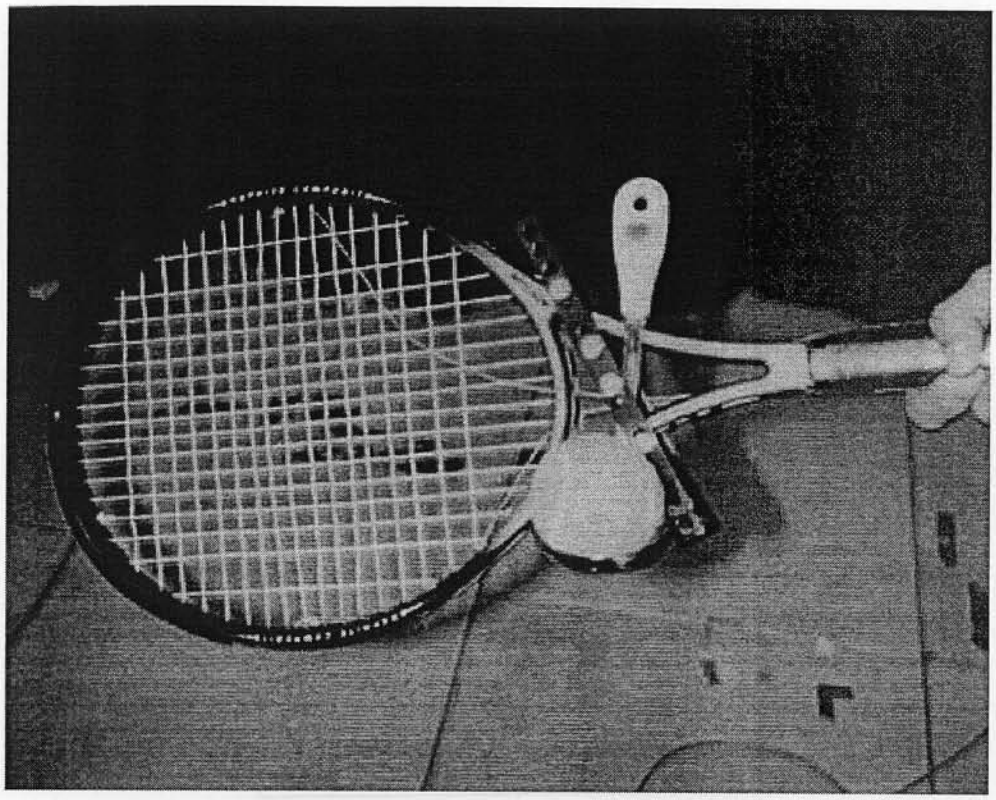


図 3.3: テニスラケットの拡大図.

Inverse Relaxation Model) を用いて経由点を抽出した．最適軌道は簡単のため躍度最小軌道を用いた．躍度最小軌道は $\int_t (\mathrm{d}\ddot{x}/\mathrm{d}t)^2 \mathrm{d}t$ が最小となる軌道である．これは解析的に解けて時間に関する 5 次方程式となる．

図 3.5 に抽出された経由点と経由点から再構成されたラケット面中心の三次元位置と向きの最適軌道を示す．x, y, z はそれぞれ右方向正，前方向正，上方向正，ϕ, θ, ψ 成分は YZY オイラー角 [8] を表す．

3.2.2 手先座標から関節角座標への変換

第 2 章の 2.3.3 節の方法と同様にラケット面中心の三次元位置と向きの最適軌道から関節角の最適軌道に変換し，SARCOS アーム (SARCOS Dextrous Slave Arm) に関節角目標軌道として与える．SARCOS アームは人間の腕に似た 7 関節の油圧駆動の腕型ロボットである．SARCOS アームの試行の結果は 2 台のカラーカメラで特定の色の面積重心の三次元位置を計測する三次元位置計測装置 (QUICKMAG) で計測した．サンプリング周波数は 60Hz で，精度は本実験の設定では 5 ～ 10mm 程度である．図 3.6 に SARCOS アームの手先に取り付けたテニスラケットを示す．計測点はラケット面中心とボールの位置，ゴールの位置である．

図 3.7 と図 3.14左 に 1 回目の試行を示す．ボールがうまく上がらず，空振りとなってボール

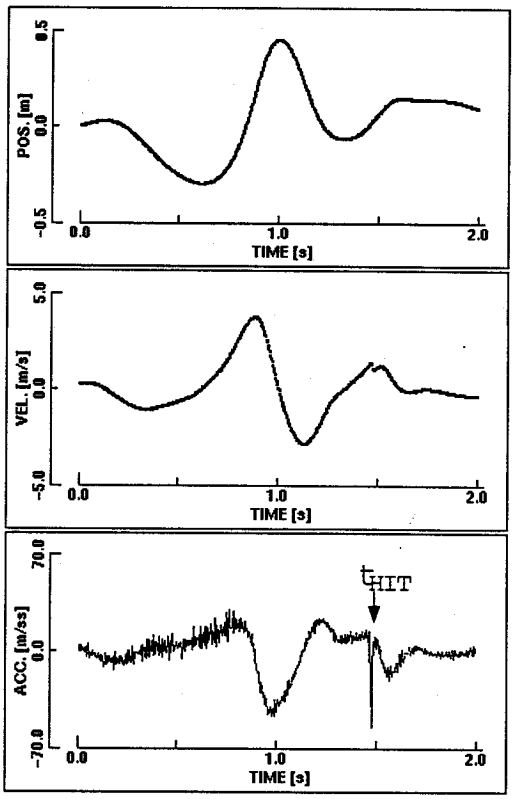


図 3.4: ラケット面中心の位置，速度，加速度の軌道．

を打つことができない．

3.2.3 サブタスク 1 の学習

図 3.8 にテニスサーブ学習のタスク表現の模式図を示す．ラケット面中心でボールを打つようにするためボールを放り上げる付近の経由点 $\mathbf{S}_{\mathrm{H}}$ を修正する．目標タスク $\mathbf{T}_{\mathrm{Hd}}$ はボールを打つべき時刻のラケット面中心の位置，実現タスク $\mathbf{T}_{\mathrm{H}}$ はボールを打つべき時刻のボールの位置をそれぞれ以下のように表す．

$$\begin{aligned} \mathbf{S}_{\mathrm{H}} &= (x_2, y_2, z_2, x_3, y_3, z_3)^{\mathrm{T}} \\ \mathbf{T}_{\mathrm{Hd}} &= (x_{\mathrm{Hd}}, y_{\mathrm{Hd}}, z_{\mathrm{Hd}})^{\mathrm{T}} \\ \mathbf{T}_{\mathrm{H}} &= (x_{\mathrm{H}}, y_{\mathrm{H}}, z_{\mathrm{H}})^{\mathrm{T}} \end{aligned}$$

$n+1$ 回目の試行の経由点を広義ニュートン法により以下のように求める．

$$\mathbf{S}_{\mathrm{H}}^{n+1} = \mathbf{S}_{\mathrm{H}}^{n} + J_{\mathrm{H}}^{\#} B_{\mathrm{H}} (\mathbf{T}_{\mathrm{Hd}}^{n} - \mathbf{T}_{\mathrm{H}}^{n}) \tag{3.1}$$

B_{H} は学習を安定させるためのゲイン行列で

$$B_{\mathrm{H}} = \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix} \tag{3.2}$$

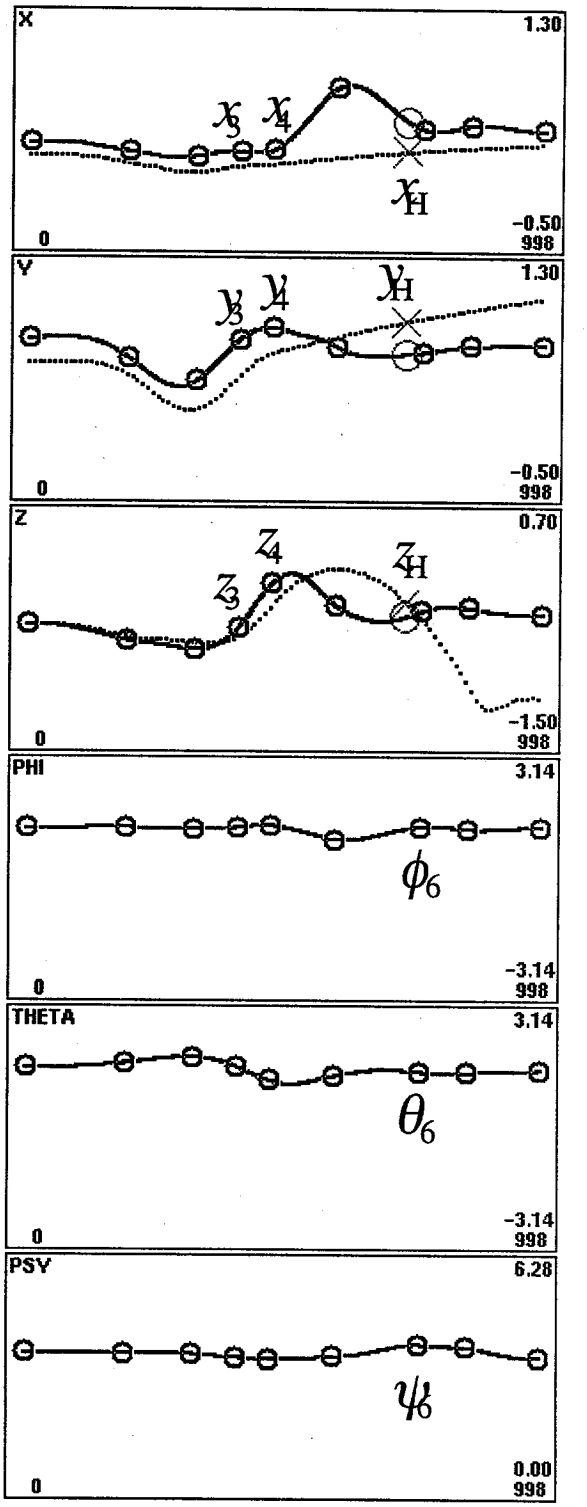


図 3.5: 経由点と最適軌道．x, y, z はそれぞれ右方向正，前方向正，上方向正，ϕ, θ, ψ 成分は YZY オイラー角を表す．小さい ∘ は経由点，実線は最適軌道 (躍度最小軌道)，点線はボールの軌道をそれぞれ示す．大きい ◯ はボールを打つべき時間のラケット面中心の位置，× はそのときのボールの位置をそれぞれ示す．

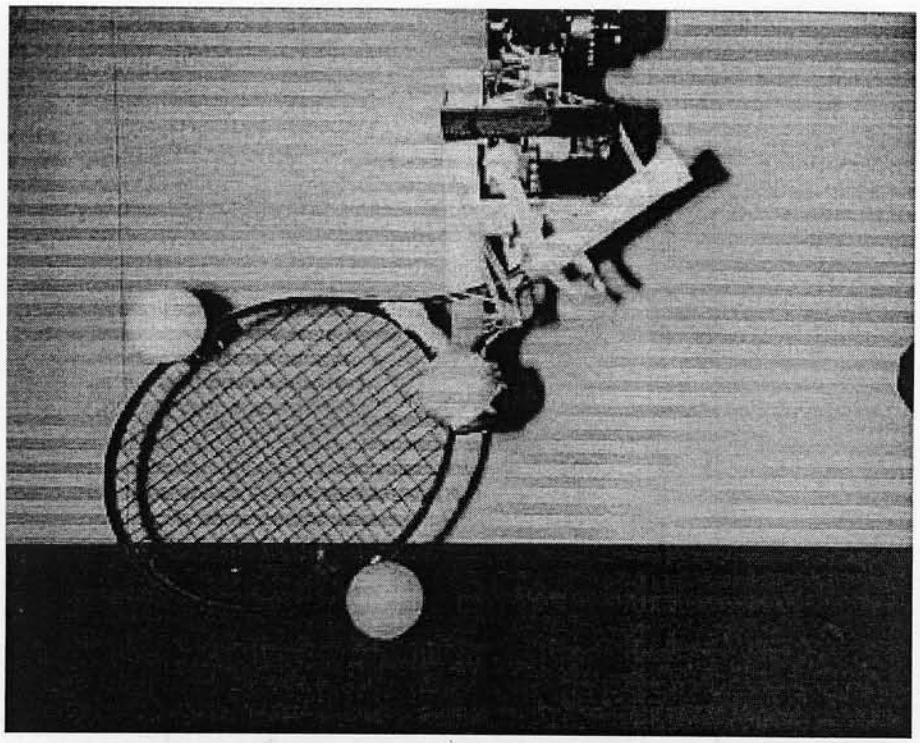


図 3.6: SARCOS アームの手先に取り付けたテニスラケット．ラケット面中心の位置を QUICKMAG で測定するために，ラケットに色付きマーカーを付けてある．ラケットの根本のカップにテニスのボールがのせてある．

とした．ヤコビ行列 J_{H} を解析的に求めることは困難なので，図 3.9 で示すように最初の試行の軌道のまわりで経由点に微小な摂動 $\pm\delta x_2, \cdots, \pm\delta z_3$ を順に与えたときのタスクの変化 δT_{H} を観測して以下のように求めた．

$$J_{\mathrm{H}} = \frac{\delta \mathbf{T}_{\mathrm{H}}}{\delta \mathbf{S}_{\mathrm{H}}} = \begin{pmatrix} \frac{\delta x_{\mathrm{H}}}{\delta x_2} & \cdots & \frac{\delta x_{\mathrm{H}}}{\delta z_3} \\ \frac{\delta \dot{y}_{\mathrm{H}}}{\delta x_2} & \cdots & \frac{\delta y_{\mathrm{H}}}{\delta z_3} \\ \frac{\delta z_{\mathrm{H}}}{\delta x_2} & \cdots & \frac{\delta z_{\mathrm{H}}}{\delta z_3} \end{pmatrix} \tag{3.3}$$

図 3.10 と図 3.14中 に 25 回目 (J_{H} を求めるための 12 回の試行も含む) の試行を示す．ボールがうまく上がってラケットの中心に当たるようになっている．

3.2.4 サブタスク 2 の学習

ボールがラケット面中心に当たるようになったところで打ったボールがゴールの方向に飛ぶようにラケット面の向き $\mathbf{S}_{\mathrm{G}}$ を修正する．目標タスク $\mathbf{T}_{\mathrm{Gd}}$ はゴールの位置，実現されたタスク $\mathbf{T}_{\mathrm{G}}$ はゴールの高さまでボールが落ちたときのボールの位置をそれぞれ示す．

$$\mathbf{S}_{\mathrm{G}} = (\phi_5, \theta_5, \psi_5, \phi_6, \theta_6, \psi_6)^{\mathrm{T}}$$

$$\mathbf{T}_{\mathrm{Gd}} = (x_{\mathrm{Gd}}, y_{\mathrm{Gd}})^{\mathrm{T}}$$

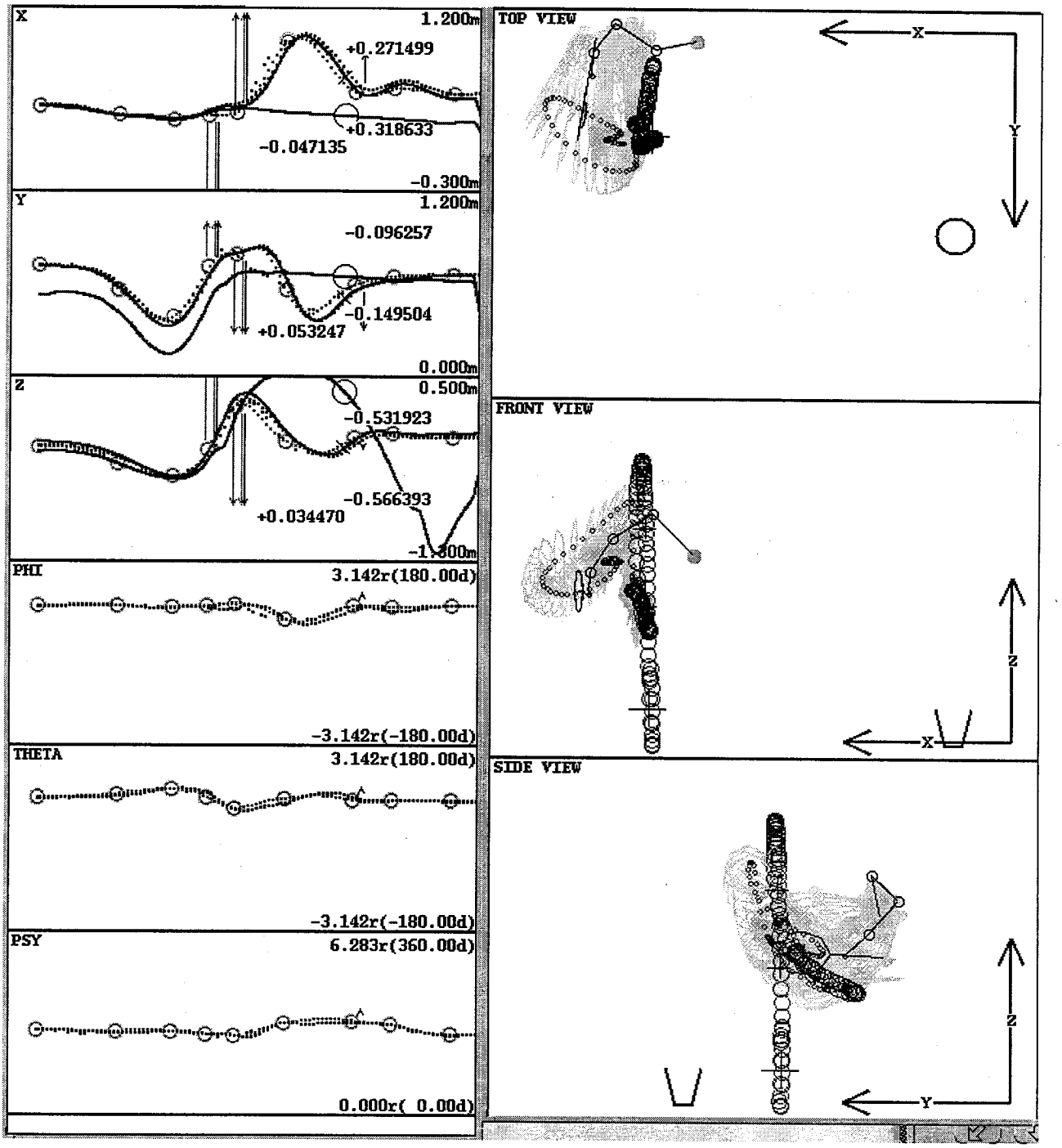
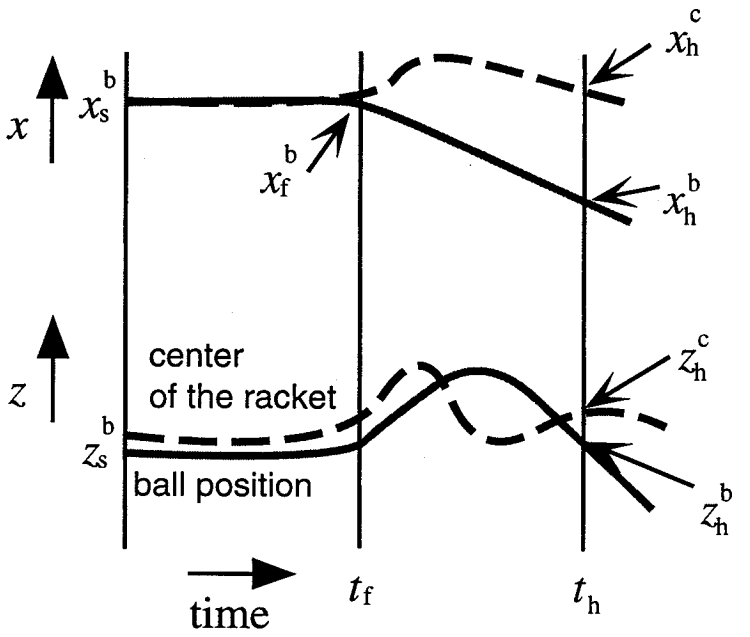


図 3.7: 学習前のテニスサーブの試行．左: テニスラケット面中心の三次元位置と向きの軌道と経由点，右: ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡 (上から上面，前面，側面図をそれぞれ示す) を示す．右図において，ゴールの位置を，上面図では ○，前面と側面図では ⊔ で示す．



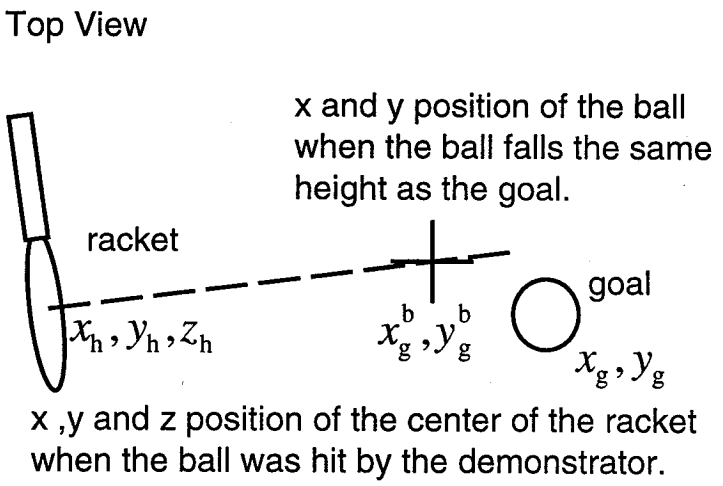


図 3.8: テニスサーブ学習のタスク表現の模式図．上: ラケット面中心 (破線) とボール (実線) の三次元位置の時間変化．下: ラケットとボール，ゴールとボールの落下位置の関係．

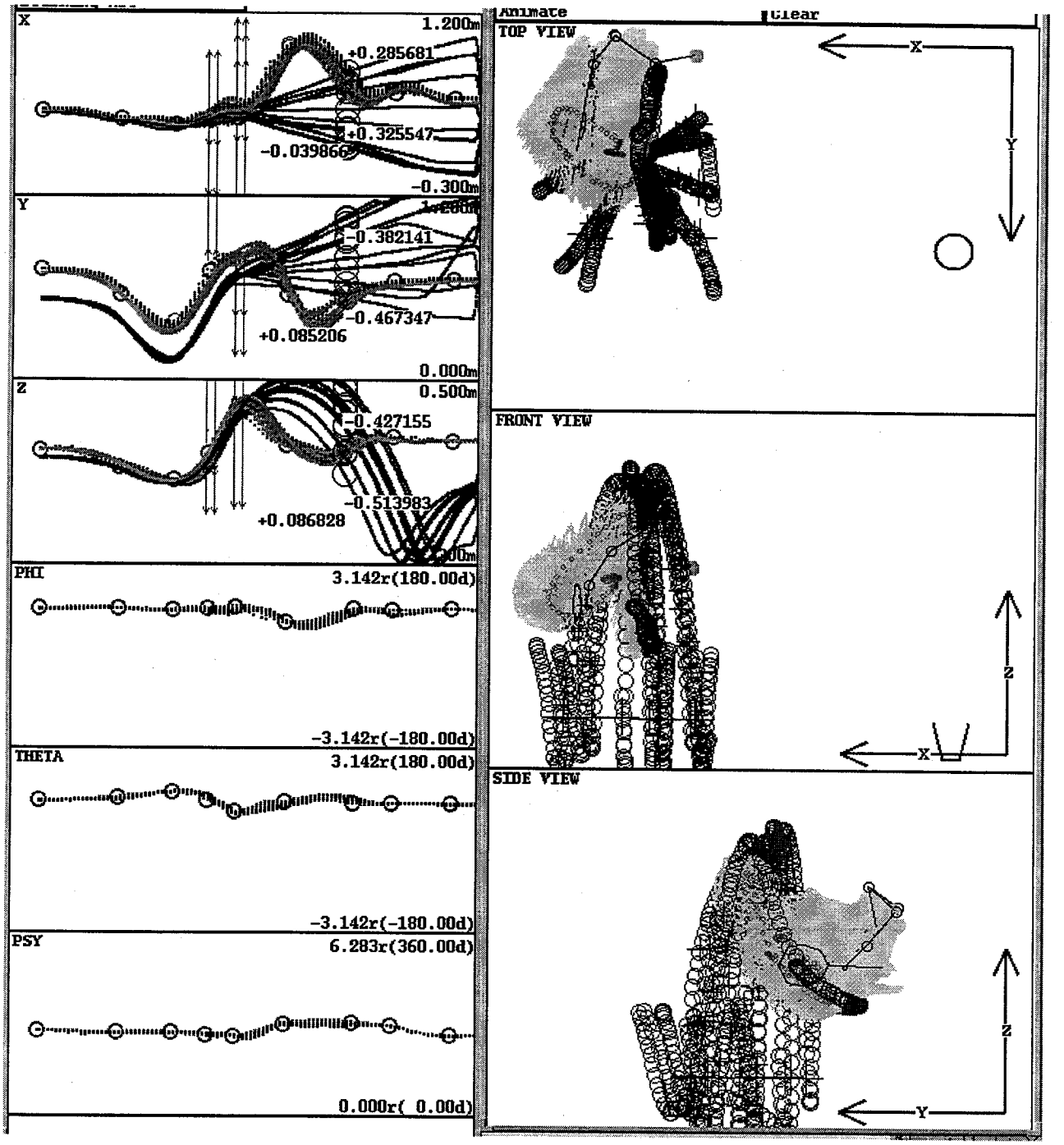


図 3.9: 第 1 段階のヤコビ行列推定のためのテニスサーブの試行．左: テニスラケット面中心の三次元位置と向きの軌道と経由点，右: ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡 (上から上面，前面，側面図をそれぞれ示す) を示す．右図において，ゴールの位置を，上面図では ○，前面と側面図では ⊔ で示す．

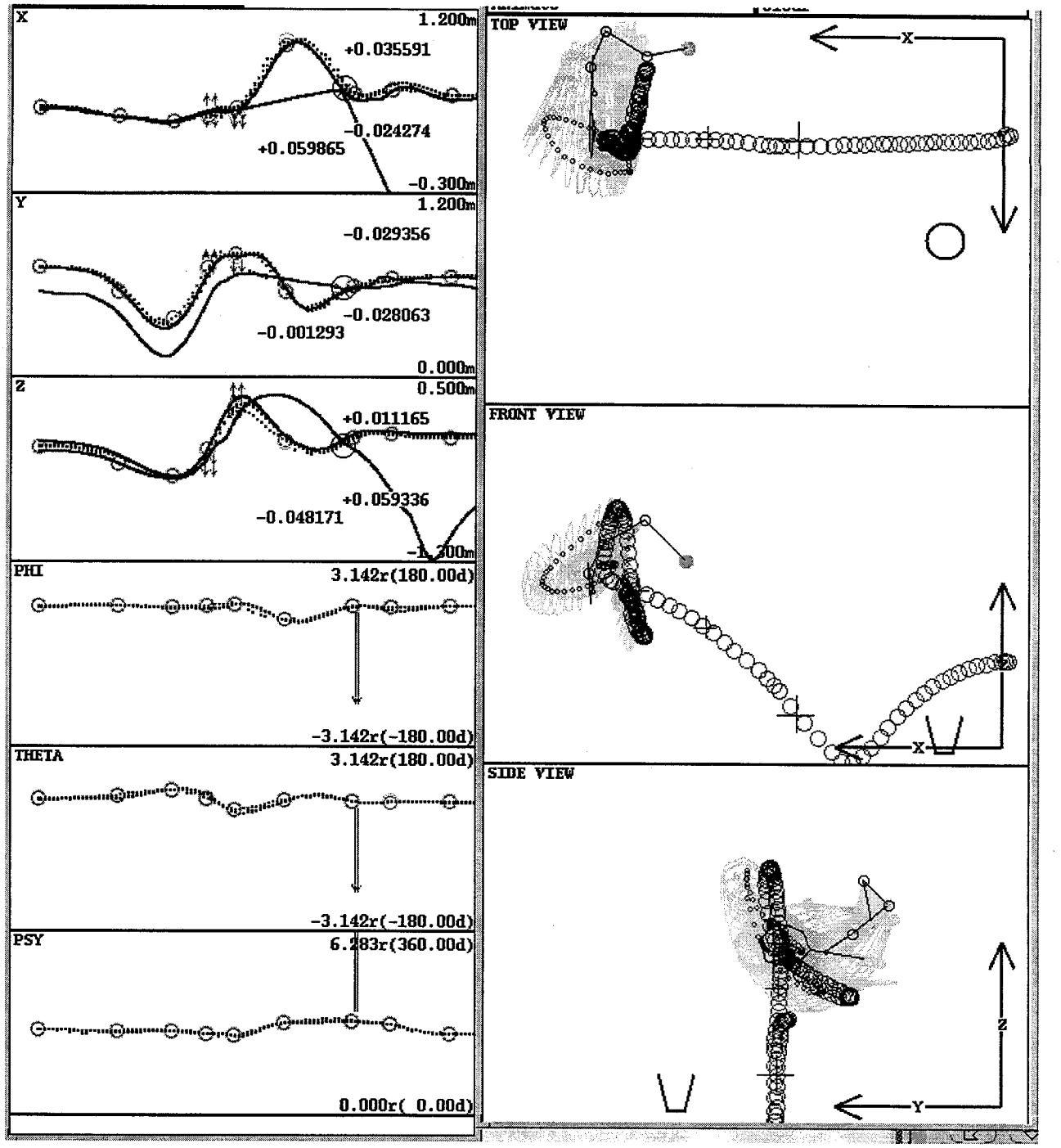


図 3.10: 第 1 段階学習後のテニスサーブの試行．左: テニスラケット面中心の三次元位置と向きの軌道と経由点，右: ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡 (上から上面，前面，側面図をそれぞれ示す) を示す．右図において，ゴールの位置を，上面図では ◯，前面と側面図では ⊔ で示す．

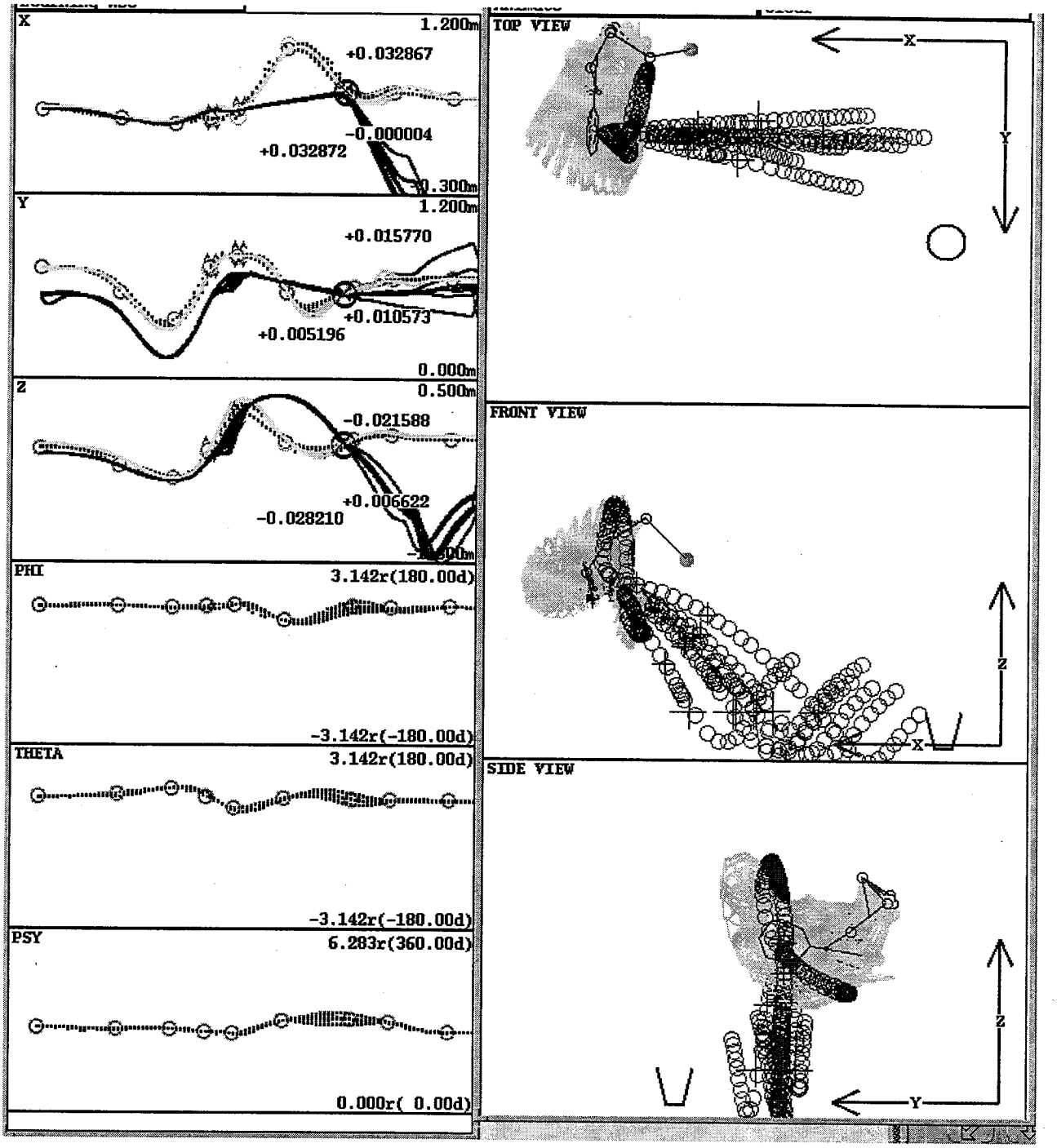


図 3.11: 第2段階のヤコビ行列推定のためのテニスサーブの試行．左: テニスラケット面中心の三次元位置と向きの軌道と経由点，右: ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡 (上から上面，前面，側面図をそれぞれ示す) を示す．右図において，ゴールの位置を，上面図では ○，前面と側面図では ⊔ で示す．

$$\mathbf{T}_{\mathrm{G}} = (x_{\mathrm{G}}, y_{\mathrm{G}})^{\mathrm{T}}$$

J_{G} は経由点に微小な摂動 $\pm\delta\phi_6, \cdots, \pm\delta\psi_6$ を順に与えたときのタスクの変化 $\delta\mathbf{T}_{\mathrm{G}}$ を観測して以下のように求めた.

$$J_{\mathrm{G}} = \frac{\delta\mathbf{T}_{\mathrm{G}}}{\delta\mathbf{S}_{\mathrm{G}}} = \begin{pmatrix} \frac{\delta x_{\mathrm{G}}}{\delta\phi_6} & \frac{\delta x_{\mathrm{G}}}{\delta\theta_6} & \frac{\delta x_{\mathrm{G}}}{\delta\psi_6} \\ \frac{\delta y_{\mathrm{G}}}{\delta\phi_7} & \frac{\delta y_{\mathrm{G}}}{\delta\theta_6} & \frac{\delta y_{\mathrm{G}}}{\delta\psi_6} \end{pmatrix} \tag{3.4}$$

図 3.11 に第 2 段階のヤコビ行列推定のためのテニスサーブの試行を示す.

$n+1$ 回目の試行の経由点を広義ニュートン法により以下のように求める.

$$\mathbf{S}_{\mathrm{G}}^{n+1} = \mathbf{S}_{\mathrm{G}}^{n} + J_{\mathrm{G}}^{\#} B_{\mathrm{G}} (\mathbf{T}_{\mathrm{Gd}} - \mathbf{T}_{\mathrm{G}}^{n}) \tag{3.5}$$

$$B_{\mathrm{G}} = \begin{pmatrix} \alpha & 0 \\ 0 & \alpha \end{pmatrix}$$

$$\alpha = \begin{cases} 0.4 & \text{if } d < 0.05 \\ 0.8 - 8d & \text{if } d > 0.05 \ \& \ d < 0.1 \\ 0 & \text{otherwise} \end{cases}$$

$$d = |\mathbf{T}_{\mathrm{Hd}} - \mathbf{T}_{\mathrm{H}}| \tag{3.6}$$

ここで d はボールを打つべき時刻でのラケット面中心とボールの距離である. $d < 0.1$ であればボールがラケット面中心に当たったと判定しラケットの向きを修正するが, そうでなければラケットの向きは修正しない.

図 3.12 に第 2 段階学習中のテニスサーブの 16 回分の試行を重ねて示す. ボールは徐々にゴールの方向に飛ぶようになっている. 図 3.13 と図 3.14右 に第 2 段階学習後のテニスサーブの試行を示す. ボールはゴールに入っている.

図 3.15 に 65 回試行中の, ボールを打つ時刻のボールとラケット面中心との距離 $|\mathbf{T}_{\mathrm{Hd}} - \mathbf{T}_{\mathrm{H}}^{n}|$, ゴールの高さまでボールが落ちたときのボールとゴールの距離 $|\mathbf{T}_{\mathrm{Gd}} - \mathbf{T}_{\mathrm{G}}^{n}|$ の試行毎の変化を示す. 図 3.15 において, 縦軸は距離 [m], 横軸は試行回数を示す. ○ は $|\mathbf{T}_{\mathrm{Gd}} - \mathbf{T}_{\mathrm{G}}^{n}|$, × は $|\mathbf{T}_{\mathrm{Hd}} - \mathbf{T}_{\mathrm{H}}^{n}|$ をそれぞれ示す. 横軸の下の $*$,+ は, 1 段目は J_{H}, 2 段目は J_{G} に関して, ヤコビ行列を推定するために摂動を与えた試行であることをそれぞれ示す. 当たり損ねや, 計測の失敗のために J_{G} 推定の摂動を与えた試行がとびとびになっている. 図 3.14右 に 60 回目 ($J_{\mathrm{H}}, J_{\mathrm{G}}$ を求めるための試行も含む) の試行を示す. ゴールの方向にボールが跳ね返されるようになっている.

3.3 経由点の選択

前節のロボット実験では学習によって修正する経由点を実験者が試行錯誤で選んだ. その際, 経由点の選択は直観的で根拠のないものであった. 本節では, より客観的な経由点の選択方法を考える. 本節の実験はすべて計算機シミュレーションにより行なった.

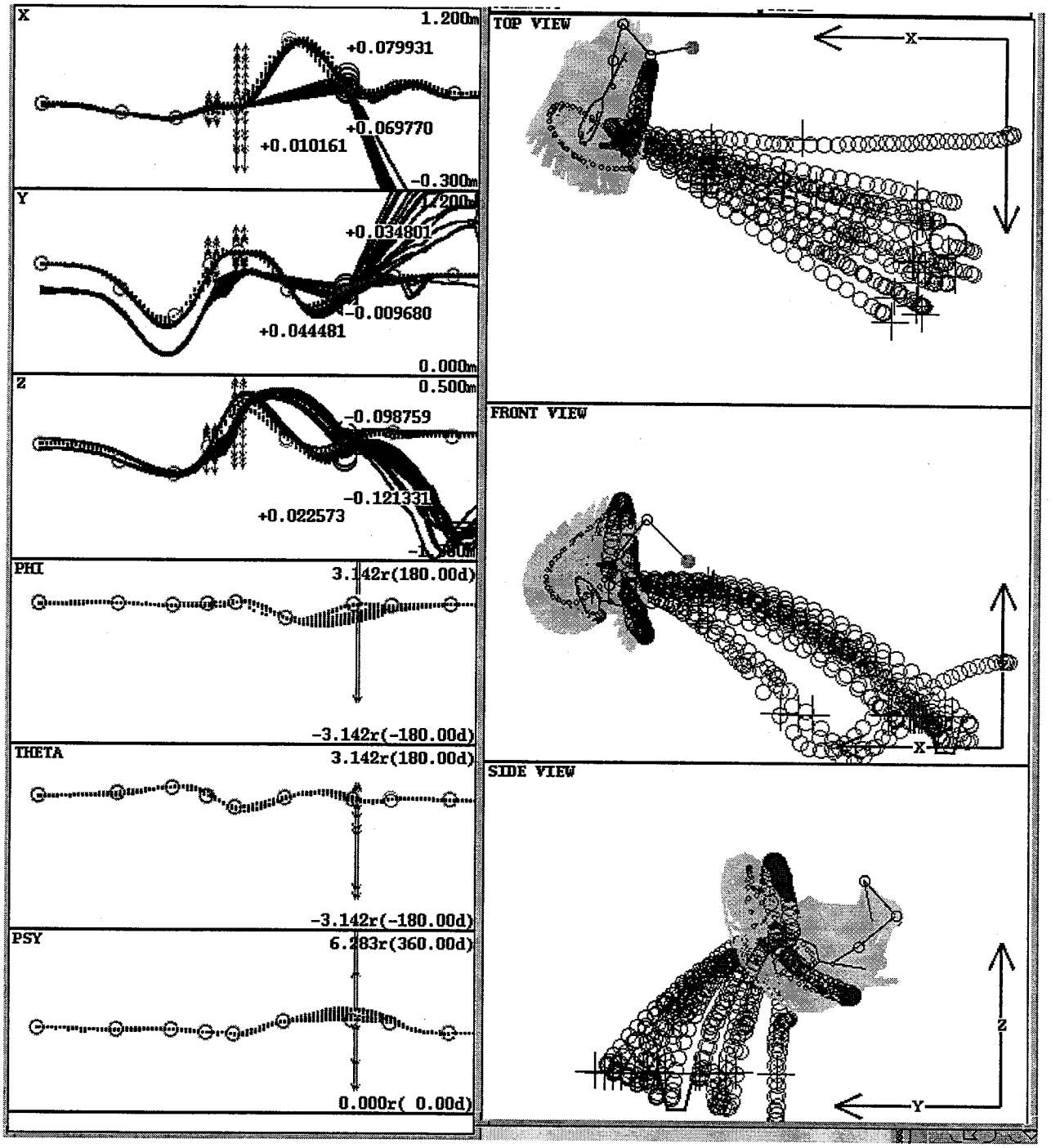


図 3.12: 第 2 段階学習中のテニスサーブの試行．16 回分の試行を重ねて示す．左: テニスラケット面中心の三次元位置と向きの軌道と経由点，右: ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡 (上から上面，前面，側面図をそれぞれ示す) を示す．右図において，ゴールの位置を，上面図では ◯，前面と側面図では ⊔ で示す．

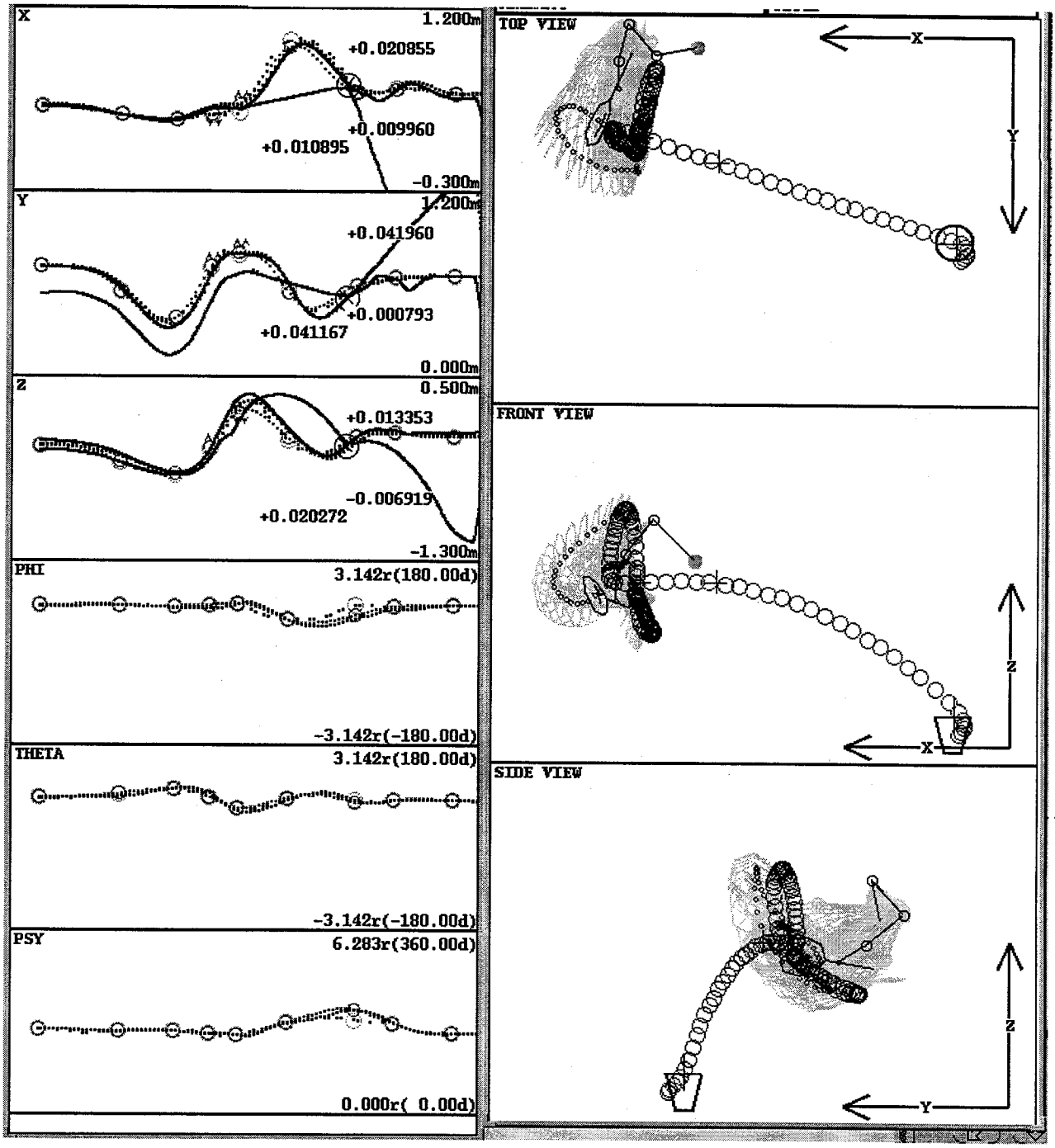


図 3.13: 第 2 段階学習後のテニスサーブの試行．左: テニスラケット面中心の三次元位置と向きの軌道と経由点，右: ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡 (上から上面，前面，側面図をそれぞれ示す) を示す．右図において，ゴールの位置を，上面図では ○，前面と側面図では ⊔ で示す．

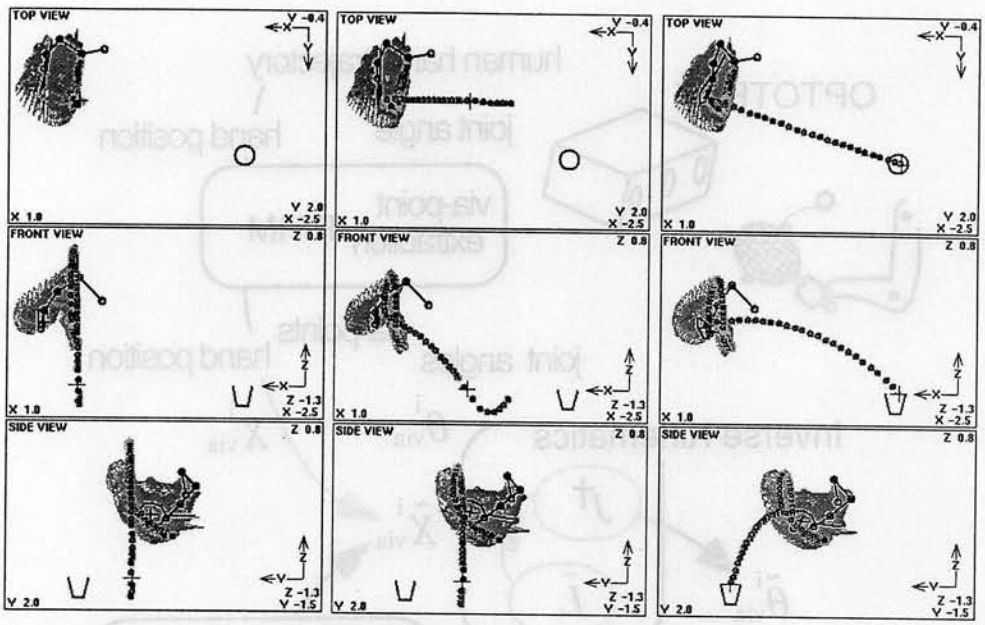


図 3.14: テニスサーブの試行．左からそれぞれ 1, 25, 60 回目の試行．ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡を示す．上から上面，前面，側面図をそれぞれ示す．ゴールの位置を，上面図では ○，前面と側面図では ⊔ で示す．

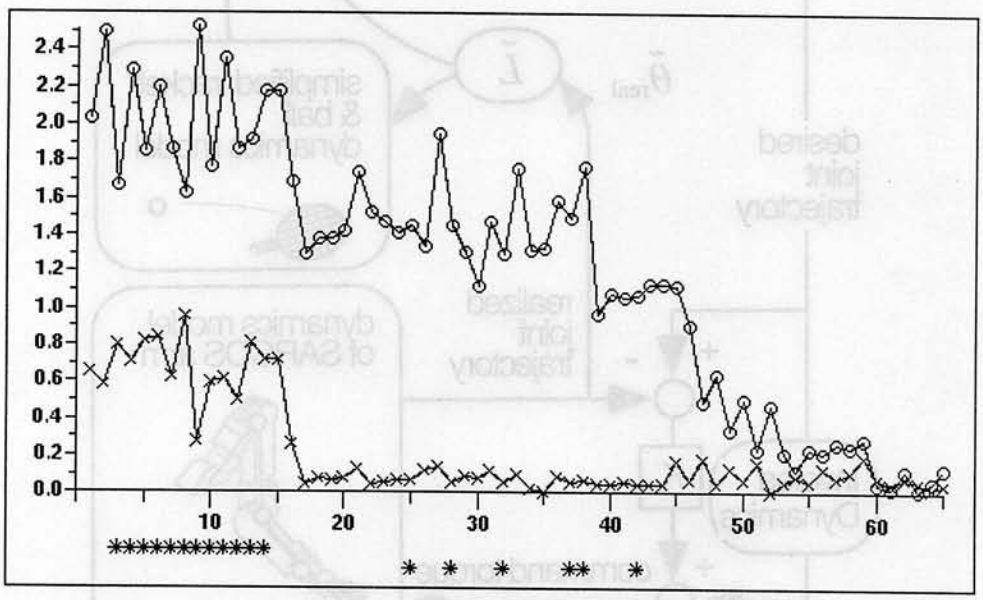


図 3.15: テニスサーブの 65 回試行中の誤差の変化．× はボールを打つ時刻のボールとラケット面中心との距離 $|\mathbf{T}_{\mathrm{Hd}} - \mathbf{T}_{\mathrm{H}}^{n}|$ の試行毎の変化を示す．○ はゴールの高さまでボールが落ちたときのボールとゴールの距離 $|\mathbf{T}_{\mathrm{Gd}} - \mathbf{T}_{\mathrm{G}}^{n}|$ の試行毎の変化を示す．縦軸は距離 [m]，横軸は試行回数を示す．横軸の下の $*$,$+$ は，1 段目は J_{H}，2 段目は J_{G} に関して，ヤコビ行列を推定するために摂動を与えた試行であることをそれぞれ示す．

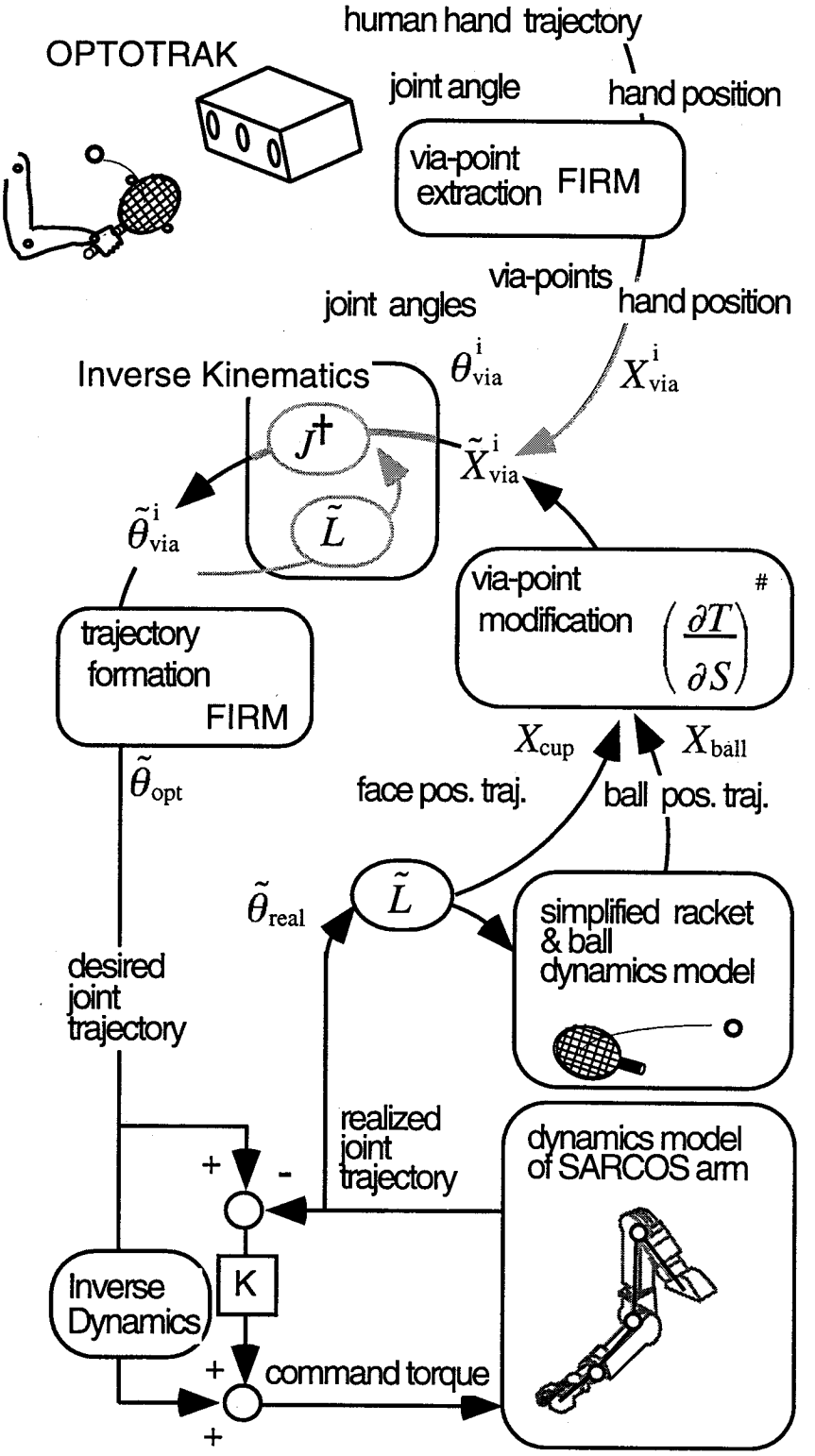


図 3.16: テニスサーブ学習シミュレーション実験システムの概略.

3.3.1 サブタスク 1 における経由点の選択

まず第 1 番目の経由点から終点までの位置と向きの成分 ($6 \times 8 = 48$ 個) に摂動を加え，ヤコビ行列を以下のように求める．

$$\frac{\delta \mathbf{T}_{\mathrm{H}}}{\delta \mathbf{S}} = \begin{pmatrix} \frac{\delta x_{\mathrm{H}}}{\delta x_1} & \frac{\delta x_{\mathrm{H}}}{\delta y_1} & \cdots & \frac{\delta x_{\mathrm{H}}}{\delta \psi_8} \\ \frac{\delta x_{\mathrm{H}}}{\delta y_1} & \frac{\delta y_{\mathrm{H}}}{\delta y_1} & \cdots & \frac{\delta y_{\mathrm{H}}}{\delta \psi_8} \\ \frac{\delta z_{\mathrm{H}}}{\delta x_1} & \frac{\delta z_{\mathrm{H}}}{\delta y_1} & \cdots & \frac{\delta z_{\mathrm{H}}}{\delta \psi_8} \end{pmatrix}$$

$$= (\mathbf{h}_1, \cdots, \mathbf{h}_{48}) \tag{3.7}$$

このときヤコビ行列の第 j 列の縦ベクトルのノルム

$$\|\mathbf{h}_j\| \tag{3.8}$$

を，その経由点の成分の変化がサブタスク 1 に及ぼす効果と考えることとする．例えば第 8 列目の縦ベクトルのノルム

$$\|\mathbf{h}_8\| = \sqrt{\left(\frac{\delta x_{\mathrm{H}}}{\delta y_2}\right)^2 + \left(\frac{\delta y_{\mathrm{H}}}{\delta y_2}\right)^2 + \left(\frac{\delta z_{\mathrm{H}}}{\delta y_2}\right)^2} \tag{3.9}$$

は 2 番目の経由点の y 成分がサブタスク 1 に及ぼす効果である．この値が大きい経由点の成分からいくつかを修正すべき経由点として選べばよい．

図 3.17上 において，横軸は列の番号 j を，縦軸は $\|\mathbf{h}_j\|$ の値をそれぞれ示す．図 3.17上 から，ヤコビ行列の第 13, 14, 15, 19, 20, 21 列，すなわち第 3, 4 番目の経由点のうち位置に関する成分を抜きだし，

$$\mathbf{S}_{\mathrm{H}} = (x_3, y_3, z_3, x_4, y_4, z_4)^{\mathrm{T}} \tag{3.10}$$

を修正する経由点として選べばよい．

3.3.2 サブタスク 2 における経由点の選択

3.3.1 節で選んだ経由点成分を用いて第 1 段階のシミュレーション学習を行ない，ボールがラケット面中心に当たるようになったところで，この軌道のまわりにおけるサブタスク 2 に関するヤコビ行列を以下のように求める．

$$\frac{\delta \mathbf{T}_{\mathrm{G}}}{\delta \mathbf{S}} = \begin{pmatrix} \frac{\delta x_{\mathrm{G}}}{\delta x_1} & \frac{\delta x_{\mathrm{G}}}{\delta y_1} & \cdots & \frac{\delta x_{\mathrm{G}}}{\delta \psi_8} \\ \frac{\delta y_{\mathrm{G}}}{\delta x_1} & \frac{\delta y_{\mathrm{G}}}{\delta y_1} & \cdots & \frac{\delta y_{\mathrm{G}}}{\delta \psi_8} \end{pmatrix}$$

$$= (\mathbf{g}_1, \cdots, \mathbf{g}_{48}) \tag{3.11}$$

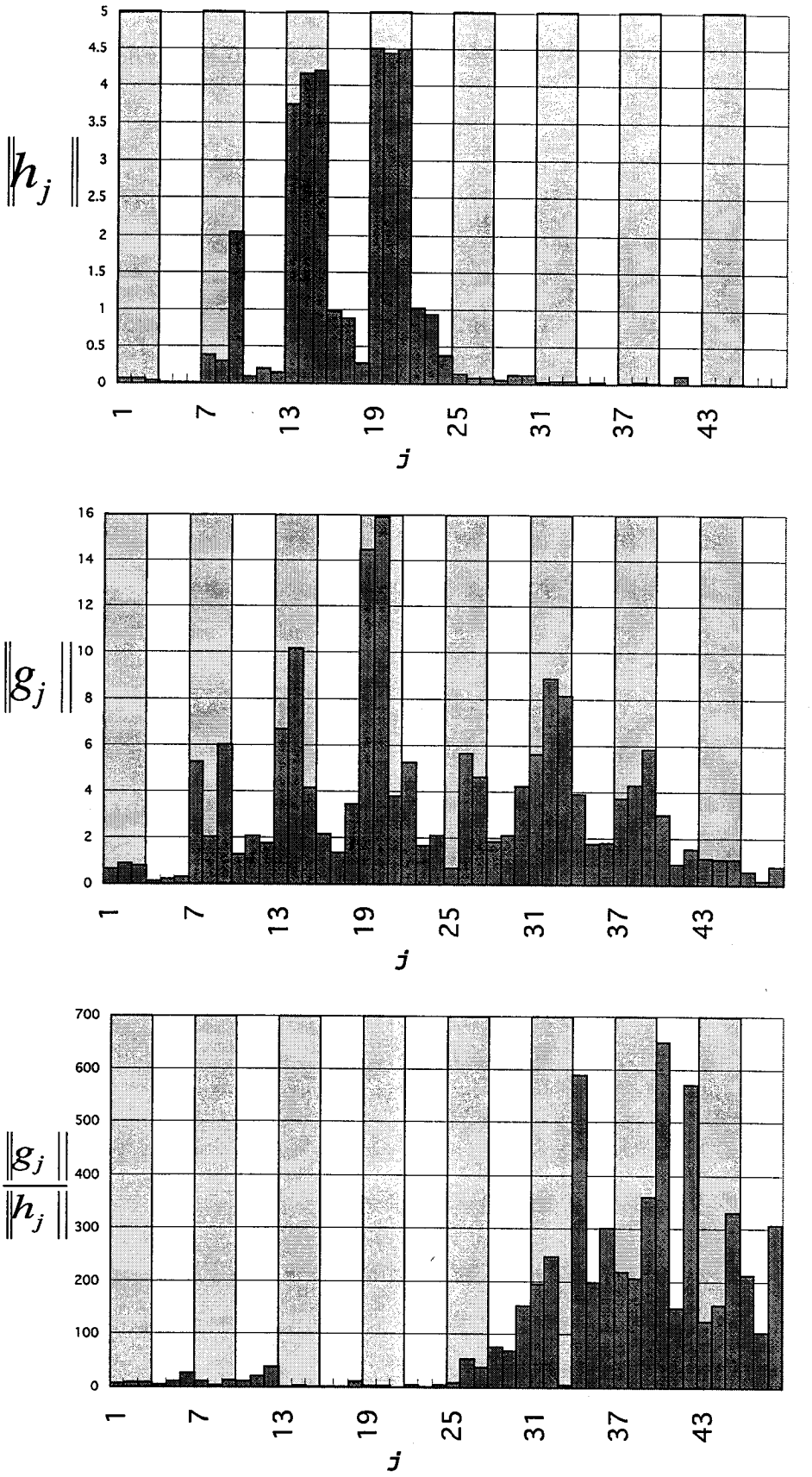


図 3.17: サブタスクに対する経由点修正の効果.

第3.3.1節と同様に $\|\mathbf{g}_j\|$ を求めたものが図3.17中である．$\|\mathbf{g}_j\|$ の値が大きいものを修正すべき経由点として選び学習の計算機シミュレーションを行なったところ，学習は不安定となり収束しなかった．この場合，第3.3.1節で選んだ経由点成分も含まれるので，サブタスク2のための経由点修正の影響がサブタスク1にも及んだためと考えられる．

学習を安定に収束させるため，サブタスク2のみに経由点修正の効果が現れ，サブタスク1に影響の少ない経由点の選択をしたい．そこで，

$$\frac{\|\mathbf{g}_j\|}{\|\mathbf{h}_j\|} \tag{3.12}$$

を求めたものが図3.17下である．図3.17下から，ヤコビ行列の第34, 40, 42列，すなわち第6, 7番目の経由点のうち向きに関する成分の一部を選び，修正する経由点を

$$\mathbf{S}_\mathrm{G} = (\phi_6, \phi_7, \psi_7)^\mathrm{T} \tag{3.13}$$

とすれば，サブタスク1に悪影響を与えず，しかもサブタスク2の学習が安定して収束すると考えられる．この経由点の組み合わせで学習の計算機シミュレーションを行なったところ，学習は安定して収束した．この結果は実ロボットによる実験で直観的に選んだ経由点と定性的に一致している．

第 4 章

動作や環境の変動への適応

4.1 はじめに

本研究では，抽象的表現として経由点を用いたタスクレベル学習を試みている．第 2 章では，第 1 章 1.5 節の枠組を用いてロボットにダイナミックな運動であるけん玉を行わせることに成功した [38, 58, 59, 60, 53, 54, 57, 61].

けん玉実験では簡単なタスク表現が可能な最も簡単な運動を選び，タスク目標を達成させることができた．しかし第 1 章 1.5 節の枠組が汎用性をもち得るかどうかを確かめるためには，連続する複数の動作が含まれる階層的な運動や，非線形性の強い制御対象を扱う場合など，種々の運動における学習可能性を調べる必要がある．連続する複数の動作が含まれる階層的な運動に関しては，第 3 章でロボットにテニスサーブを行わせることに成功した [55, 62].

本章では最適化原理に基づく軌道計画と経由点の枠組を用いた見まねによるロボットの運動学習について汎用性と学習可能性に関して調べる．そのためより複雑で困難な運動として振り子の振り上げの運動学習を試みる．4.3 節の計算機シミュレーションによる振り子の振り上げ実験では，操作対象の振り子が強い非線形性を持っている．

Atkeson らは，人間のデモンストレーションからロボットに振り子の振り上げ動作を行なわせている [4]．彼らはタスクモデルの学習と最適制御の組合せで，少ない試行回数で振り上げ動作を成功させた．彼らのモデルでは，タスクの物理的性質に関する先見的な知識に基づくモデルとして制御対象である振り子の内部モデルを用いているが，我々のモデルでは制御対象の内部モデルは陽には用いない．(ヤコビ行列 $J^{\#}$ が間接的な内部モデルであると考えてよい).

第 2 章と第 3 章では，被験者が行なった見本の運動をロボットに与える初期の運動軌道とした．このとき，学習の間 $J^{\#}$ は大きく変化しないという仮定のもとに学習前の運動軌道の近傍で見積もった $J^{\#}$ を以降の学習にも用いていた．しかし，振り子などの非線形性の強い制御対象を扱う場合には学習の進行とともに $J^{\#}$ が大きく変化し，第 2 章と第 3 章と同様には学習を行なえない．そこで本章ではヤコビ行列 $J^{\#}$ を学習の間，自動的に校正する方法を提案する．

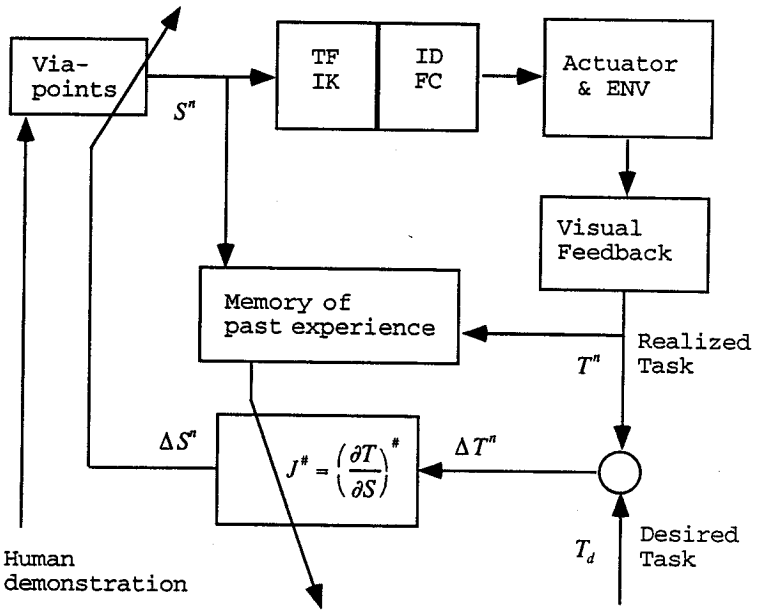


図 4.1: ヤコビ行列自動校正システムのブロック図.

4.2 ヤコビ行列の自動校正

本論文で述べてきた広義ニュートン法による学習では，学習前に，まず最初の試行で経由点に微小な摂動を与えてヤコビ行列 J の推定値を求めるが，制御対象の非線形性が強い場合は学習が進むにつれ真の J そのものが変化する．また, 実際の応用においてはアクチュエータの動作のばらつきや測定系のノイズの影響が無視できない場合も起こり得る．したがって，最初の推定値をそのまま用いたのでは学習が遅くなったり，最悪の場合発散してしまう．

制御対象の非線形性が強い場合やアクチュエータの動作のばらつきや測定系のノイズの影響が無視できない場合，安定に学習を収束させるために学習の途中で J を推定しなおしたいが，再推定の時期を自動的に決定することは困難である．また再推定のためだけに摂動を与えることは試行回数がいたずらに増加するので好ましくない．そこで図 4.1 で示すように学習途中の経由点と実現されたタスクを記憶しておきヤコビ行列を自動的に校正する．

n 回目で得られる経由点とタスクをそれぞれ

$$\mathbf{S}^n = \{x_1^n, \cdots, x_L^n\}, \tag{4.1}$$

$$\mathbf{T}^n = \{\tau_1^n, \cdots, \tau_M^n\} \tag{4.2}$$

とする．このとき，$n-m$ 回目から n 回目までの m 回の試行で得られた経由点と実現されたタスクの集合を

$$\{\mathbf{S}^{n-m}, \mathbf{T}^{n-m}\}, \cdots, \{\mathbf{S}^n, \mathbf{T}^n\} \tag{4.3}$$

とする．このうち任意の 2 試行 (例えば i 番目と j 番目) の差分を

$$\{\delta\mathbf{S}, \delta\mathbf{T}\} = \{\mathbf{S}^i - \mathbf{S}^j, \mathbf{T}^i - \mathbf{T}^j\} \tag{4.4}$$

のように求める．$\{\delta\mathbf{S},\ \delta\mathbf{T}\}$ のうち $\|\delta\mathbf{S}\|$ と $\|\delta\mathbf{T}\|$ が充分大きく，かつ互いに一時独立になるようなものを K 組み選んで行列 $\mathcal{X}$, $\mathcal{T}$ を作る．

$$\mathcal{X} = \begin{pmatrix} \delta x_1^1 & \cdots & \delta x_L^1 \\ \vdots & & \vdots \\ \delta x_1^K & \cdots & \delta x_L^K \end{pmatrix}, \tag{4.5}$$

$$\mathcal{T} = \begin{pmatrix} \delta \tau_1^1 & \cdots & \delta \tau_M^1 \\ \vdots & & \vdots \\ \delta \tau_1^K & \cdots & \delta \tau_M^K \end{pmatrix} \tag{4.6}$$

行列 $\mathcal{X}$, $\mathcal{T}$ とヤコビ行列 J には

$$\mathcal{T}^{\mathrm{T}} = J\mathcal{X}^{\mathrm{T}}$$

の関係が成り立つ．

$$(AB)^{\mathrm{T}} = B^{\mathrm{T}}A^{\mathrm{T}}$$

から

$$\begin{aligned} \mathcal{T} &\approx \mathcal{X}J^{\mathrm{T}}, \\ J^{\mathrm{T}} &\approx \mathcal{X}^{\#}\mathcal{T} \end{aligned}$$

と変形でき，これをさらに変形すると式 (4.7) のようにヤコビ行列が推定される．

$$J = \left(\mathcal{X}^{\#}\mathcal{T}\right)^{\mathrm{T}} \tag{4.7}$$

上に述べた手順を用いると，過去に学習によって修正された経由点と実現されたタスクの蓄積からヤコビ行列を自動的に校正することができる．

4.3 振り子の振り上げ

本節では非線形性の強い制御対象の運動学習の例として，振り子の振り上げ動作の学習に関して述べる．ここで行なう振り子の振り上げ動作の運動学習システムの模式図を図 4.2 に示す．振り子には水平まわりに自由に回転する軸が一方に付いている．回転軸を水平方向に振り，下に垂れていた振り子を振り上げて上向きに止めることが運動の目標である．

例えば，回転軸を横に動かそうとする力がかかった時，振り子が下向きの時と上向きの時では，振り子が回転しようとする方向は逆になる．また振り子が真横を向いている時は，回転軸にかかる力は振り子の回転にまったく影響しない．

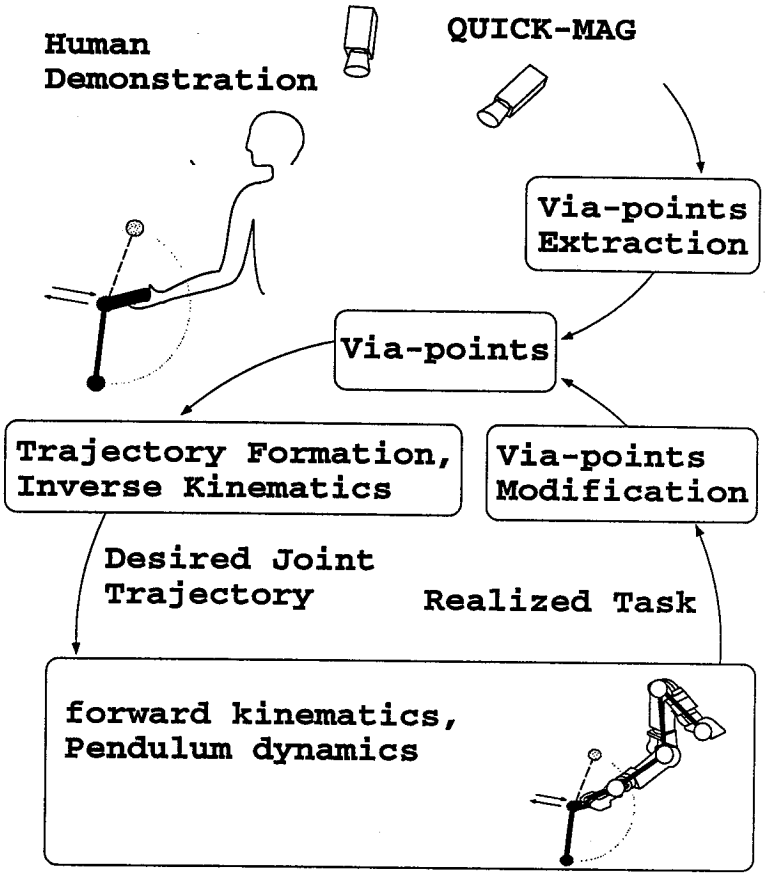


図 4.2: 振り子振り上げの運動学習システムの概略.

4.3.1 運動軌道計測と経由点抽出

人間の手本の運動軌道は Prof. Atkeson に提供して頂いた．運動軌道計測は QUICKMAG で行なわれ，マーカは振り子の上端と下端に取り付けられた [4]．人間の手本では水平方向以外の手先位置もわずかに変化しているが，本実験では水平方向のみの位置軌道から経由点を抽出した．

ロボットの数学モデルは簡単のため SARCOS アームの順キネマティクス方程式のみとし，運動方程式は用いない．すなわち関節角実現軌道が目標軌道に完全に一致する理想的なモデルである．振り子は式 (4.8) で示すような回転軸の摩擦が無く質点が先端に集中した理想的な振り子の力学モデルを用いた [4]．角度は振り子が真下を向いている時が $-\pi$，真上を向いている時が 0 で，向かって反時計まわりが正である．

$$\dot{\theta}_{k+1} = \dot{\theta}_k + \frac{\Delta}{l}\left(g\sin(\theta_k) - \ddot{x}_k\cos(\theta_k)\right) \tag{4.8}$$

ここで θ は振り子の角度，$\dot{\theta}$ は振り子の角速度をそれぞれ表す．角度は，振り子が真下を向い

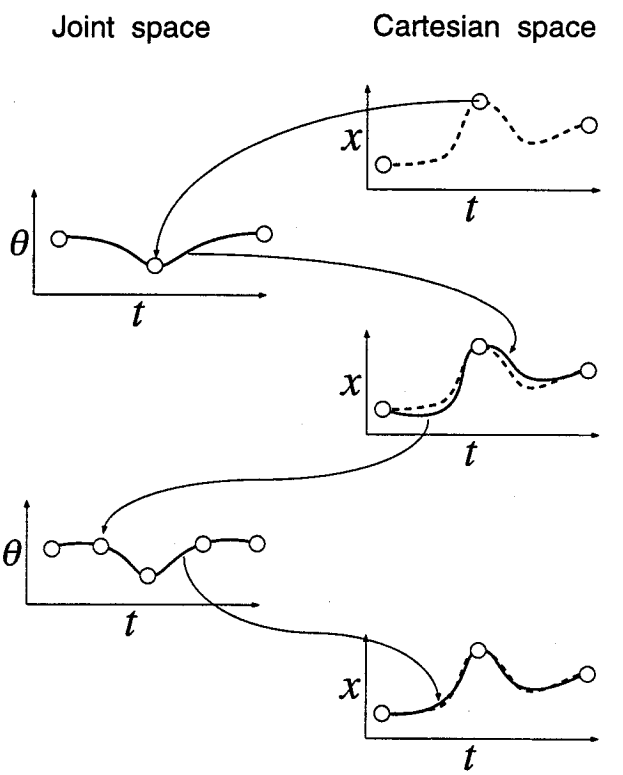


図 4.3: 関節角軌道を求める手順.

ている時が $-\pi$, 真上を向いている時が 0 で，向かって反時計まわりが正である．$\ddot{x}$ は手先位置の加速度で向かって左向きが正である．$\Delta = 0.002s$ は刻み時間，$g = 9.8m/s^2$ は重力加速度，$l = 0.29m$ は振り子の長さ，をそれぞれ表す.

4.3.2 逆キネマティクス

手先座標から関節角座標へ変換するとき，以下の二つの方法が最も簡単である.

(1) 手先の経由点から手先軌道を生成し，手先軌道からすべてのサンプリング点で逆キネマティクス変換を行ない，関節角軌道を求める.

(2) 手先の経由点から逆キネマティクス変換により関節角の経由点を求め，関節角の経由点から関節角軌道を生成する.

方法 (1) は正確な手先軌道を得ることができるが計算コストが高い．方法 (2) は計算コストは非常に低いが，手先座標の経由点を正確に通過することしか保証されないため，関節角軌道から得られた手先軌道は誤差が大きい.

第 2 章のけん玉実験と第 3 章のテニスサーブ実験では，方法 (2) を用いた．けん玉とテニスサーブ実験では，手先の軌道の誤差はそれほどタスクの達成に影響しなかった．しかし，本章

で行なう振り子の振り上げでは誤差が大きいと回転軸が水平にならない場合が発生し，手先軌道の誤差をなるべく小さくする必要が出て来た．方法 (1), (2) はそれぞれ長短があるが，ここでは，比較的計算コストが低く，方法 (2) より誤差が小さくなるように関節角空間で，経由点を追加する方法を用いる．

1. 手先の経由点から手先軌道を生成する．
2. 手先の経由点から逆キネマティクス変換により関節角の経由点を求める．
3. 関節角の経由点から関節角軌道を生成する．
4. 関節角軌道から順キネマティクス変換により手先軌道を得る．
5. 1 で得た手先軌道と 4 で得た手先軌道を比較し，誤差の大きいところに関節角の経由点を追加する．
6. 誤差が充分小さければ終了する．そうでなければ手順 3 に戻る．

L を順キネマティクス方程式とし，広義ニュートン法を用いて関節角の経由点を以下のように求める．

$$\boldsymbol{\theta}^{i+1} = \boldsymbol{\theta}^{i} + \left(\frac{\partial \mathrm{L}}{\partial \boldsymbol{\theta}^{i}}\right)^{\#} (\mathrm{S} - \mathrm{L}(\boldsymbol{\theta}^{i})) \tag{4.9}$$

4.3.3 経由点修正

図 4.4 に 1 回目の試行を示す．図 4.4左 は振り子の動きを示す．図 4.4右上 は回転軸の水平位置の時間変化を示す．破線と点線は手先と振り子先端の水平位置の時間変化を，丸は経由点をそれぞれ示す．図 4.4右下 は振り子の角度と加速度の軌道とタスクの設定をそれぞれ示す．点線は手本の振り子の，実線はロボットによって実現された振り子の角度と角速度の時間変化をそれぞれ示す．この試行では経由点はまだ修正されていない．図からわかるように，手本から得た経由点をそのまま用いてロボットに試行させた場合，振り子は下で揺れるだけで上向きにはならない ($\frac{\pi}{2}$ にさえならない)．

振り子を振り上げるという目標を達成するためのタスクの設定はどのようにでもできるが，本実験では最も直観的なものとして以下のようなタスクを設定した．実現タスクと目標タスクを，

$$\mathbf{T}_{\mathrm{d}} = \begin{pmatrix} \hat{\theta} \\ \hat{\omega} \end{pmatrix}, \tag{4.10}$$

$$\mathbf{T} = \begin{pmatrix} \theta \\ \omega \end{pmatrix} \tag{4.11}$$

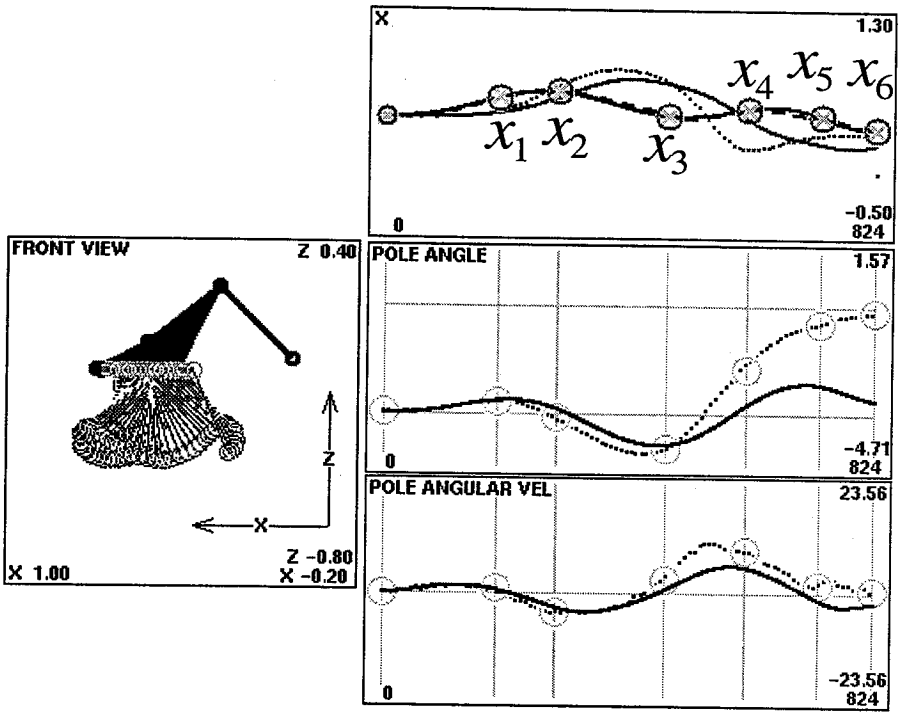


図 4.4: 学習前の振り上げ動作の試行.

と表すこととする. $\hat{\theta},\hat{\omega}$ はそれぞれ終点に一致する時間の手本の振り子の角度と角速度を表す. θ,ω はそれぞれ終点に一致する時間の振り子の角度と角速度を表す.

学習によって修正する経由点を

$$\mathbf{S} = \begin{pmatrix} x_1 \\ \vdots \\ x_6 \end{pmatrix}$$

とし, $n+1$ 回目の試行の経由点を広義ニュートン法により以下のように求める.

$$\mathbf{S}^{n+1} = \mathbf{S}^n + J^{\#} B(\mathbf{T}_\mathrm{d} - \mathbf{T}^n) \tag{4.12}$$

ここで, B は発散を防止するために収束量を調整するもので

$$B = \begin{pmatrix} 0.2 & 0 \\ 0 & 0.02 \end{pmatrix} \tag{4.13}$$

とした. ヤコビ行列 J を解析的に求めることは困難なので, まず 2 回目から 7 回目までの試行で, 経由点に微小な摂動 $\delta x_1,\cdots,\delta x_6$ を順に与え, このときの 1 回目の試行からのタスクの変化 δT からヤコビ行列の推定値 J を求める.

$$J = \frac{\delta T}{\delta S} = \begin{pmatrix} \frac{\delta\theta}{\delta x_1} & \cdots & \frac{\delta\theta}{\delta x_6} \\ & & \\ \frac{\delta\omega}{\delta x_1} & \cdots & \frac{\delta\omega}{\delta x_6} \end{pmatrix} \tag{4.14}$$

8 回目以降の試行で式 (4.12) により学習を行なう.

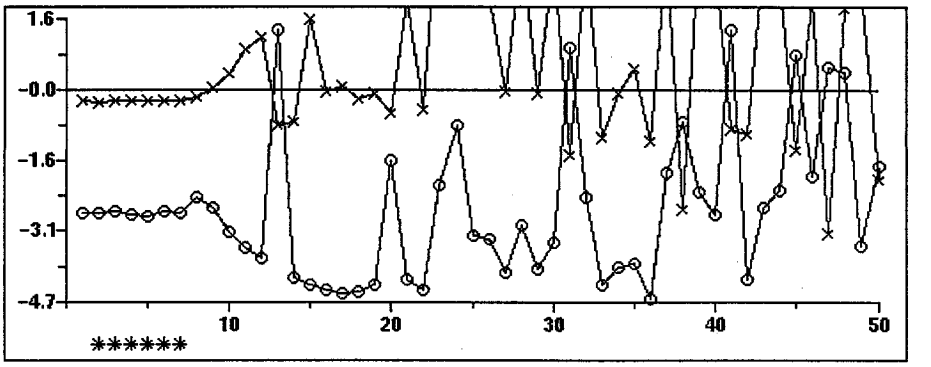


図 4.5: 50 回試行中の θ, ω の変化．最初に推定したヤコビ行列の推定値 J をそのまま学習に用いた場合．縦軸は角度，横軸は試行回数を，○ は θ を，× は $\omega \times 0.1$ をそれぞれ示す．横軸の下の $*$ はヤコビ行列の初期値を得るために摂動を与えた試行であることを示す．

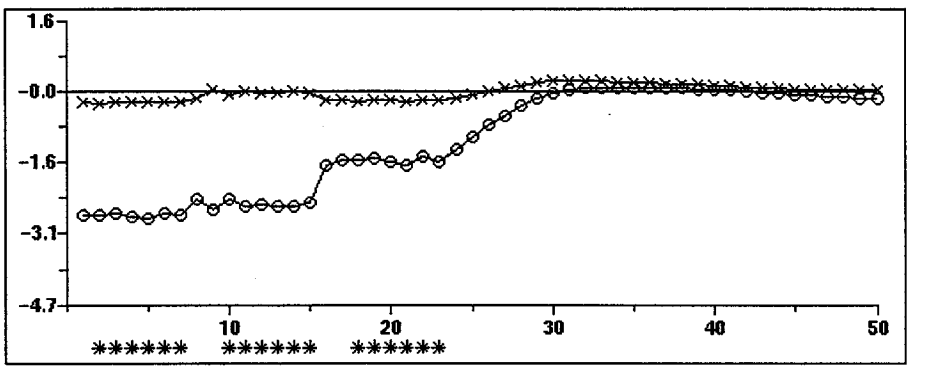


図 4.6: 50 回試行中の θ, ω の変化．学習途中でヤコビ行列を 2 回再推定した場合．縦軸は角度，横軸は試行回数を，○ は θ を，× は $\omega \times 0.1$ をそれぞれ示す．横軸の下の $*$ はヤコビ行列の初期値を得るために摂動を与えた試行であることを示す．

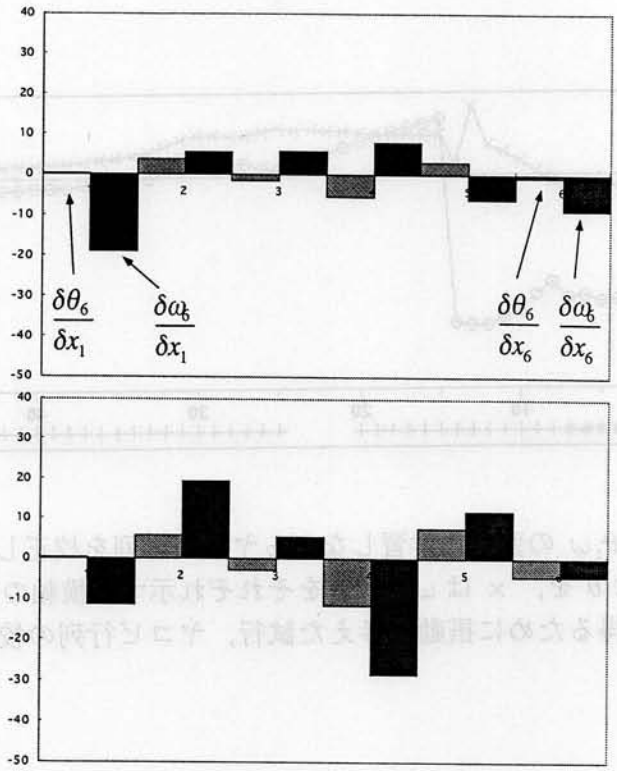


図 4.7: ヤコビ行列の最初の推定値 J_C と 3 度目の推定値.

振り子の振り上げ動作では，振り子が下で振れるときと上で振れるときとではヤコビ行列が大きく異なり，最初に推定したヤコビ行列の推定値 J をそのまま学習に用いた場合，学習が発散した．図 4.5 に 50 回試行中の θ,ω の変化を示す．縦軸は角度，横軸は試行回数を，○ は θ,× は $\omega \times 0.1$ をそれぞれ示す．横軸の下の $*$ はヤコビ行列の初期値を得るために摂動を与えた試行であることを示す．

学習途中でヤコビ行列を 2 回再推定した結果，学習は安定に収束した．図 4.6 に 50 回試行中の θ,ω の変化を示す．縦軸は角度，横軸は試行回数を，○ は θ を，× は $\omega \times 0.1$ をそれぞれ示す．横軸の下の $*$ はヤコビ行列の初期値を得るために摂動を与えた試行であることを示す．

図 4.7上 は最初のヤコビ行列の推定値のグラフである．図 4.7下 は 3 度目の推定値である．図 4.7 から分かるように，ヤコビ行列の各成分は大きさ，符号の両方の変化がみられる．

式 (4.3) から式 (4.7) を用いて，学習しながらヤコビ行列を校正すると学習は収束した．図 4.10 に 50 回目の試行を示す．振り子は上を向くようになっている．

図 4.8 に 50 回試行中の θ,ω の変化を示す．縦軸は角度，横軸は試行回数を，○ は θ を，× は $\omega \times 0.1$ をそれぞれ示す．横軸の下の $*$，$+$ はそれぞれ，ヤコビ行列の初期値を得るために摂動を与えた試行，ヤコビ行列の校正を行なった試行，であることを示す．

図 4.9上 は最初のヤコビ行列の推定値をグラフ化したものである．図 4.9下 は 50 回目の試行

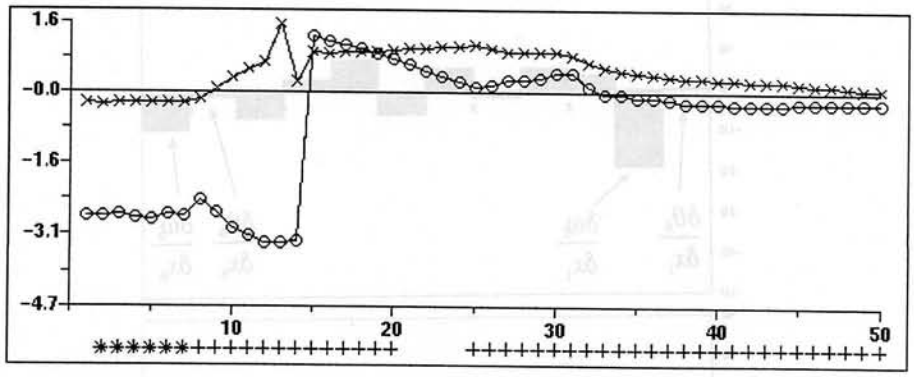


図 4.8: 50 回試行中の θ,ω の変化．学習しながらヤコビ行列を校正した場合．縦軸は角度，横軸は試行回数を，○ は θ を，× は $\omega \times 0.1$ をそれぞれ示す．横軸の下の $*$，$+$ はそれぞれ，ヤコビ行列の初期値を得るために摂動を与えた試行，ヤコビ行列の校正を行なった試行，であることを示す．

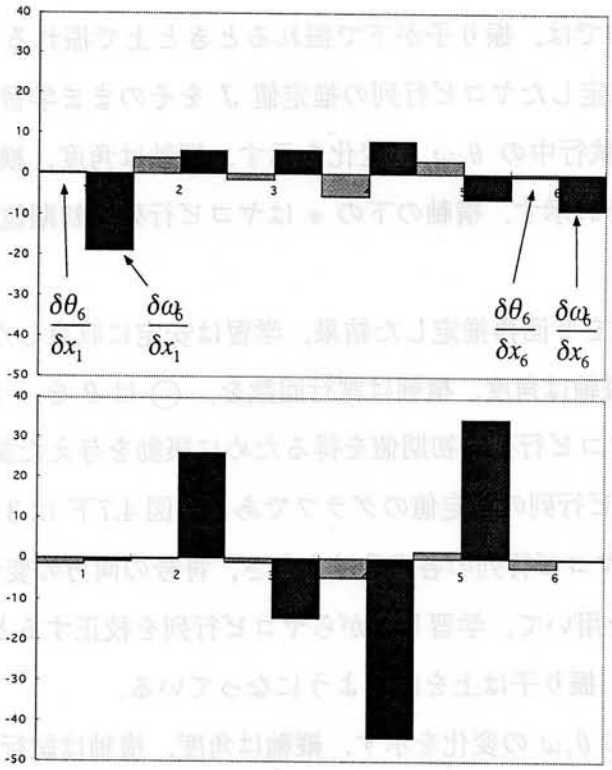


図 4.9: ヤコビ行列の最初の推定値 J_C と 50 回目の試行時の推定値．

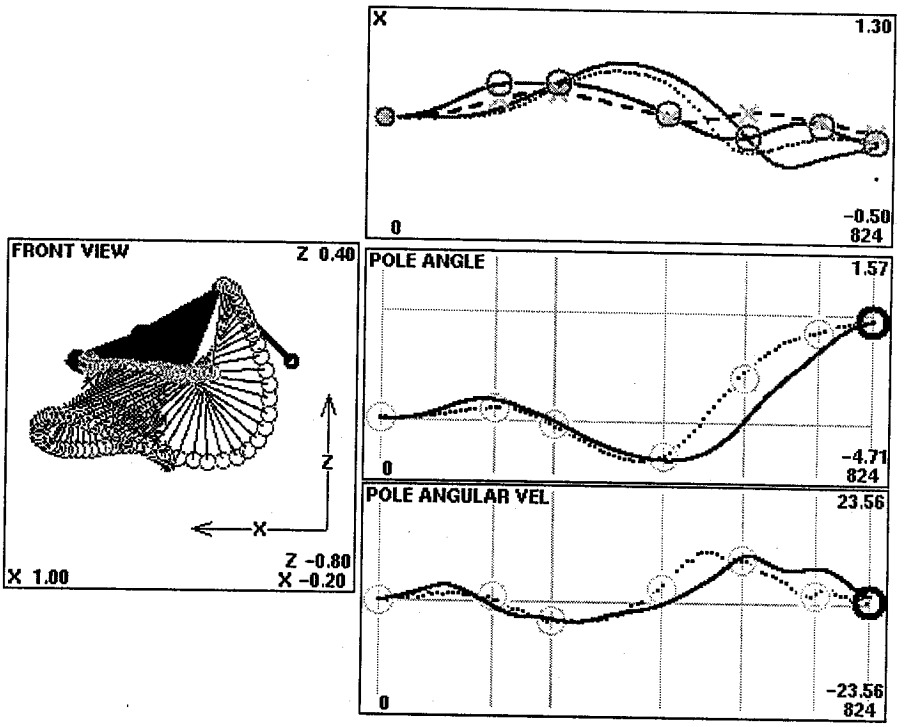


図 4.10: 50 回学習後の振り上げ動作の試行.

時のヤコビ行列の校正された値を示す. 図 4.8から分かるように, ヤコビ行列が自動的に校正されて学習が収束していることが分かる.

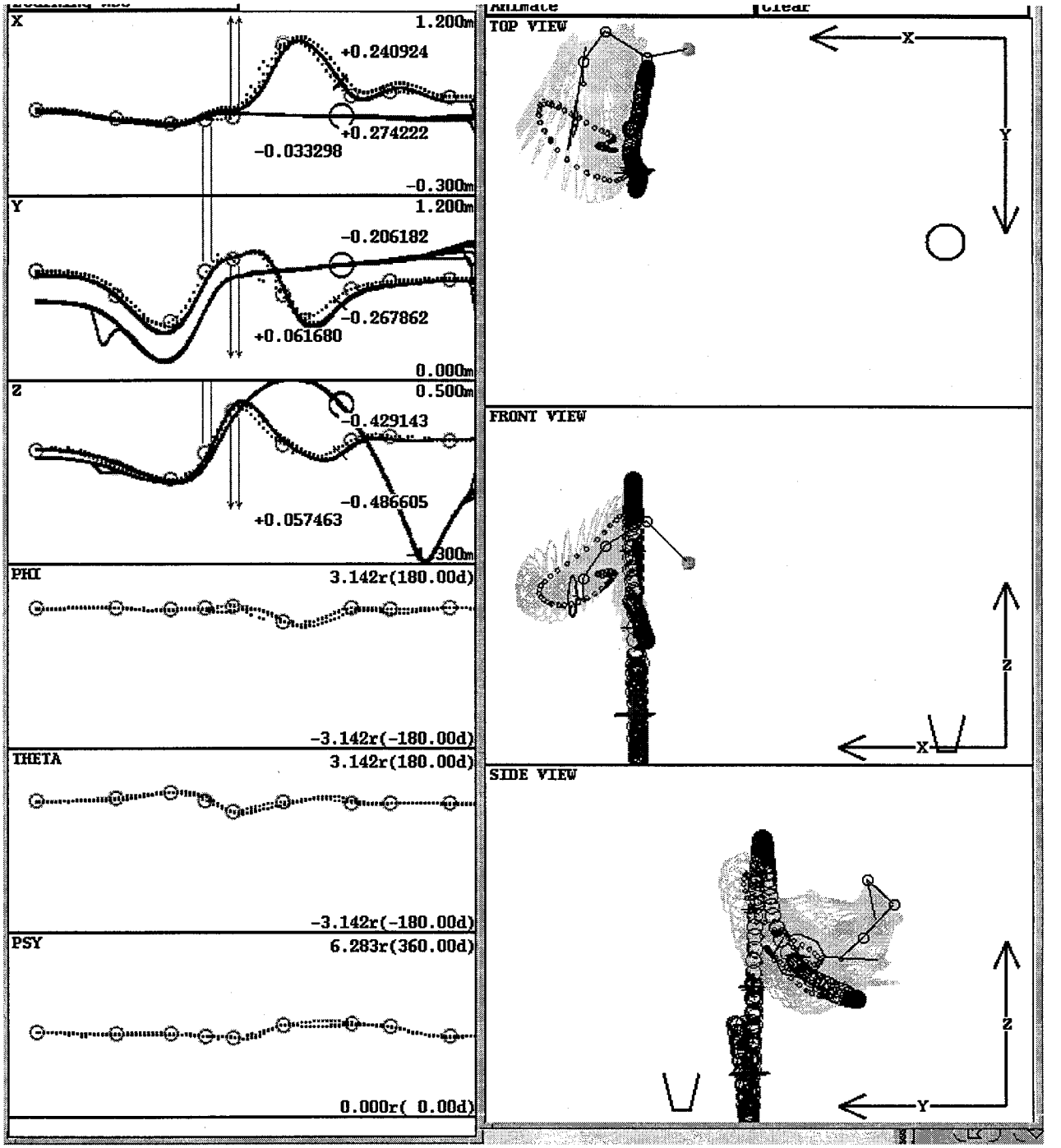


図 4.11: 学習前のテニスサーブの試行．7 回分の試行を重ねて示す．左: テニスラケット面中心の三次元位置と向きの軌道，右: ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡 (上から上面，前面，側面図をそれぞれ示す) を示す．右図において，ゴールの位置を，上面図では ◯，前面と側面図では ⊔ で示す．

4.4 テニスサーブにおけるヤコビ行列の自動校正

テニスサーブでは経由点とタスクの関係は比較的非線形性が弱い．したがってシミュレーションでは最初に推定したヤコビ行列のみを用いても学習は収束する．しかし，実際のロボットを

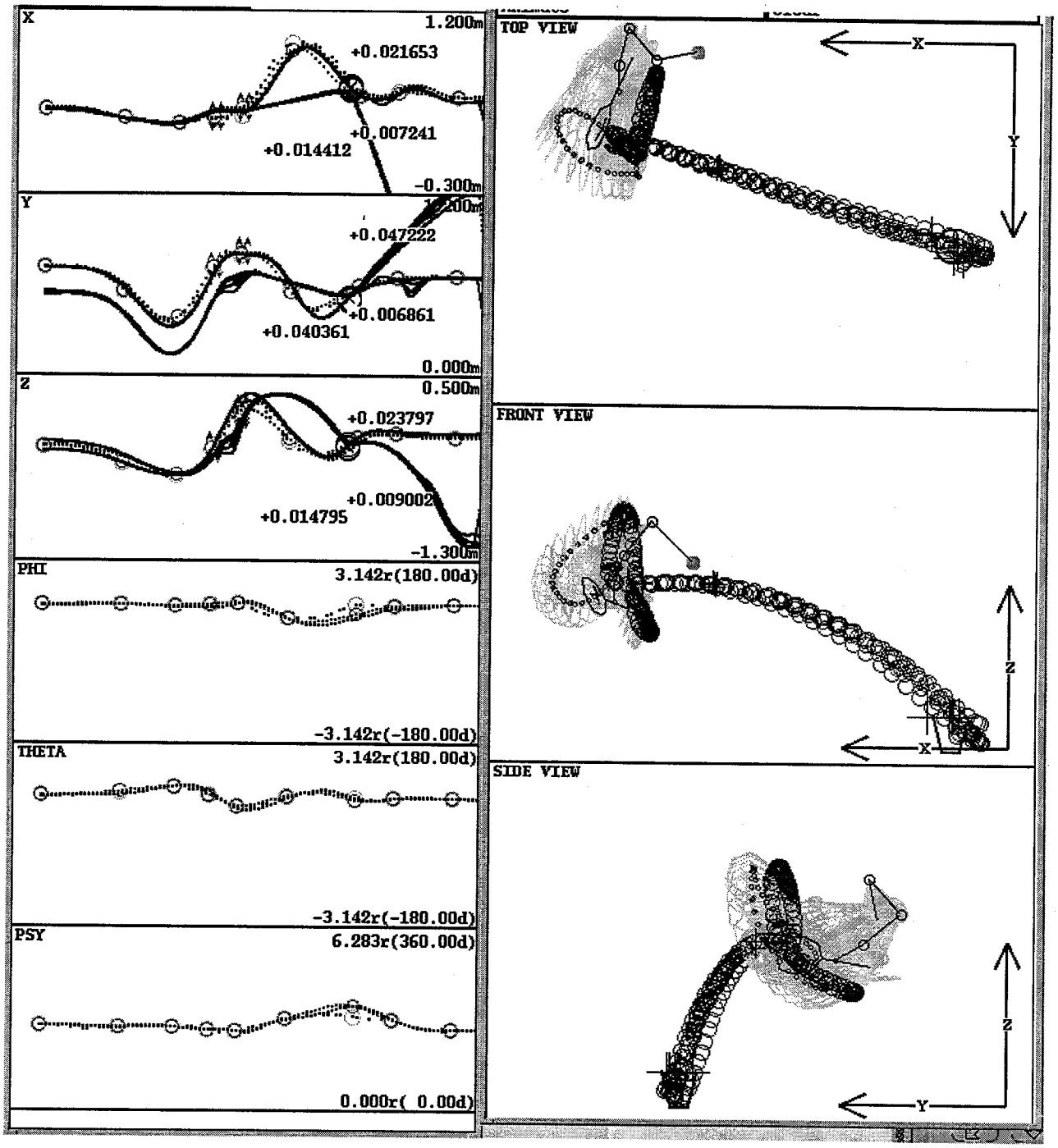


図 4.12: 第2段階学習後のテニスサーブの試行．5回分の試行を重ねて示す．左: テニスラケット面中心の三次元位置と向きの軌道，右: ボールを打つべき時刻の SARCOS アームの姿勢とボールの軌跡 (上から上面，前面，側面図をそれぞれ示す) を示す．右図において，ゴールの位置を，上面図では ○，前面と側面図では ⊔ で示す．

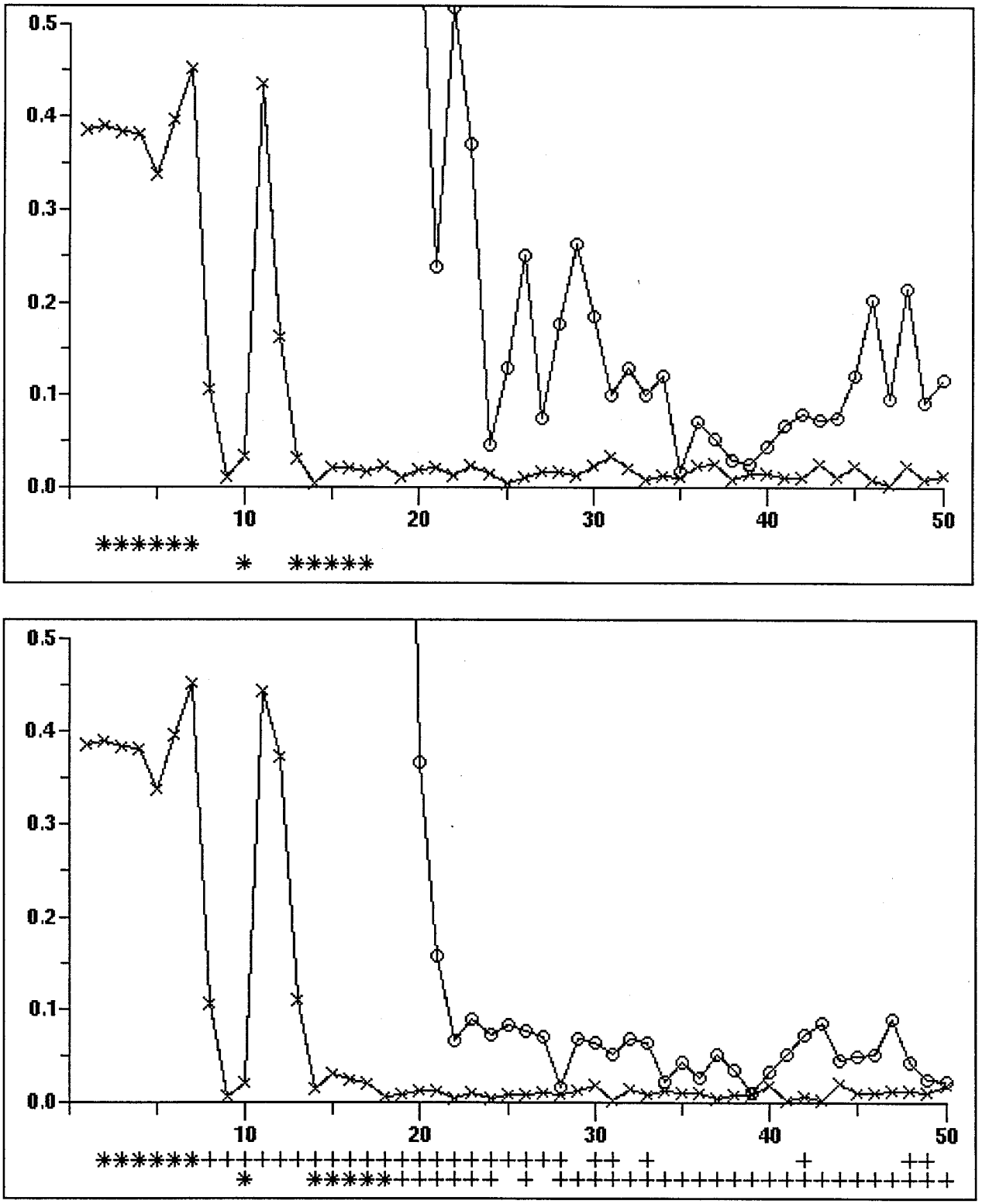


図 4.13: テニスサーブシミュレーションの 50 回試行中の誤差の変化．上: 最初に推定したヤコビ行列のみを用いた場合，下: 学習しながらヤコビ行列校正を行なった場合，(縦軸は距離 [m]，横軸は試行回数を示す．o は $|\mathbf{T}_{\mathrm{Gd}} - \mathbf{T}^{n}_{\mathrm{G}}|$，× は $|\mathbf{T}_{\mathrm{Hd}} - \mathbf{T}^{n}_{\mathrm{H}}|$ をそれぞれ示す．横軸の下の $*$，$+$ は，1 段目は J_{H}，2 段目は J_{G} に関して，それぞれ，ヤコビ行列の初期値を得るために摂動を与えた試行，ヤコビ行列の校正を行なった試行，であることを示す)．

用いた場合，第3章の図4.11，図4.12から分かるように，試行ごとにボールの飛び方にばらつきがある．ボールを放り上げる時にカップからはなれるボールの初速度，ラケット面に跳ね返る時の方向や速度にばらつきがあるためであると考えられる．特に，図4.12ではばらつきが著しい．これはラケット面とボールの表面の凹凸が大きく影響しているためであろう．また測定系のノイズの影響も含まれる．したがって，最初に推定したヤコビ行列には誤差が含まれると考えられる．

本節では，その方法を用いたヤコビ行列校正の効果をテニスサーブの運動に関して調べる．

本シミュレーション実験では実際のロボットで実験を行なう時と同じ程度にボールの飛び方にばらつきを与え，動作のばらつきやノイズに対するヤコビ行列校正の効果を調べる．

図4.13に，50試行中の，ボールを打つ時刻のボールとラケット面中心との距離 $|\mathbf{T}_{\mathrm{Hd}}-\mathbf{T}_{\mathrm{H}}^{n}|$，ゴールの高さまでボールが落ちたときのボールとゴールの距離 $|\mathbf{T}_{\mathrm{Gd}}-\mathbf{T}_{\mathrm{G}}^{n}|$ の試行毎の変化を示す．

最初に推定したヤコビ行列のみを用いて学習を進める場合より，ヤコビ行列を校正しながら学習を進めるほうが誤差の減少が速く，しかも安定していることがわかる．

第 5 章

今後の課題

5.1 はじめに

本研究では，脳の運動制御の計算理論のひとつである，最適化原理に基づく運動パターン認識の応用例として，見まねによる運動学習ロボットの開発を行なって来た．これまでにSARCOSアームにけん玉 (第2章) とテニスサーブ (第3章) を行わせることに成功した．また計算機シミュレーションにより振り子の振り上げ (第4章) も可能であることを確かめた．

けん玉はダイナミックでしかも簡単なタスク表現が可能な運動であり，単一のタスク目標を設定することにより成功させることができた．テニスサーブでは時間的に連続する二つの動作 (玉を投げ上げる動作と玉を打つ動作) が相互に強く影響しあうため，けん玉に比べて学習が困難である．振り子は非線形性の強い制御対象であり，学習の進行につれて振るまいが大きく変化するので学習が困難であった．テニスサーブ成功のため，タスク目標をサブタスク1とサブタスク2に分け，それぞれのサブタスクに対応して学習を2段階に分けた．またサブタスク間の相互影響を避けるため，ヤコビ行列を解析して学習で修正する経由点の選択を行なった．振り子の振り上げタスクでは，制御対象である振り子の動特性が学習の進行とともに変化する．学習を安定に収束させるためヤコビ行列を学習途中に再評価しなければならないが，再評価のタイミングの決定を自動的に行なうことは困難であるし，再評価のための試行は無駄である．そこで過去の学習による試行の履歴を利用して，学習の進行と同時にヤコビ行列を自動的に校正する方法を提案した．

これまで本論文で用いてきた学習スキーマは広義ニュートン法であった．しかし，解決すべき本質的な課題も多く残っている．また，どのような学習対象に対しても上の方法が有効であるとは限らない．今後，フィードバック誤差学習や強化学習，あるいはそれらを組み合わせた形の学習の枠組を取り入れて検証実験を進めて行く必要がある．本章では運動学習のための新たな枠組 (階層的強化学習の枠組) を構成するために解決すべき課題と，これから進むべきであると考える方向についてまとめる．また強化学習を取り入れるための前準備として振り子の振り上げの計算機シミュレーションに関して述べる．

5.2 本研究の枠組と問題点

これまで行なってきた見まねによる運動学習の枠組で，もっとも特徴的と考えられることは以下の2点である．

1. FIRM(Forward-Inverse Relaxation Model) を用いて経由点の抽出と最適軌道の再構成を行なう．

2. 経由点を制御変数とみなし，広義ニュートン法により学習を行なう．

これらの要素を用いてこれまで以下のように学習を行なっていた．

(1) FIRM を用いて人間の運動軌道から経由点の抽出を抽出する．

(2) タスクの成功に本質的な役割を果たすと考えられる経由点を制御変数とみなし，修正の方策を決める．

(3) すべての経由点から運動軌道全体を FIRM により運動前に再構成しておき，目標軌道としてロボットに与える．

(4) タスク目標が達成されるように経由点を修正する．

(5) 作業目標が達成されるまで (3), (4) を繰り返す．

このような枠組を用いて，これまで見まねによる運動学習をある程度実現してきた．しかし，この枠組では以下の問題点が未解決のまま残っている．

1. 運動を行なう前に全体の軌道生成を行なうので運動途中での軌道修正が不可能である．

2. サブゴールの設定が実験者の直観と試行錯誤によって与えられた．

3. 手本の運動を超えられない．手本より上手な運動はできない．

4. 白紙の状態からまったく新しい運動パターンは獲得できない．

これらの課題を解決するために，階層的強化学習の新しい枠組を構築する必要がある．

5.3 階層的強化学習の新しい枠組と今後の予定

これまで用いてきた枠組の中だけで新しい発展を望むことは困難なので，前節で述べた課題を解決するためには従来の枠組を再構築する必要がある．しかし，経由点と最適化原理に基づ

表 5.1: 階層的強化学習のための枠組.

階層	要素	目標	最適化原理	実現化のための道具
高次	ゴール, サブゴール	作業	$\sum_i J_i$	Multiple-paired forward-inverse models, 強化学習
中間	経由点, 運動時間	サブゴール	$\sum_i J_i$	経由点 強化学習
低次	運動指令, 制御対象の ダイナミクス, 運動時間, 目標位置	経由点 (x_i, x_{i+1})	$J_i = \int_{t_i}^{t_{i+1}} \dot{u}^2 dt$ $J_i = \int_{t_i}^{t_{i+1}} \dddot{x}^2 dt$	Local trajectory planner-controller 躍度 (トルク変化) 最小軌道など (Hoff-Arbib, リカレントネット, 最適制御則, 強化学習)

く軌道生成の要素は残したい．そこで，現時点で考え付く階層的強化学習のための枠組を表 5.1 に示す．

表 5.1 において，低次の階層での目標は中間の階層から与えられた経由点 x_i, x_{i+1} を通るようにコスト関数 J_i を最小とすることである．J_i としては躍度やトルク変化などを考える．中間の階層での目標は高次の階層から与えられたサブゴールを達成し，かつ $\sum J_i$ がなるべく小さくなるように経由点を定めることである．低次の階層から中間の階層へは運動誤差だけでなく J_i も伝達される．さらに J_i は高次の階層まで伝達され，$\sum_i J_i$ が最小となるように最終的なゴールが達成されればよい．結果として最終ゴールを達成することのできる軌道のうち，全体の運動軌道が最もなめらかとなる運動軌道が獲得されることが望まれる．

今後，上記の枠組を考慮しつつ，スクラッチからの運動系列獲得 (5.3.1 節)，サブゴール設定の自動化 (5.3.2 節)，local trajectory planner-controller の学習 (5.3.3 節)，などのテーマを行なっていく予定である．強化学習を取り入れる第 1 ステップとして，振り子の振り上げの計算機シミュレーションを行なった (5.4 節).

5.3.1 スクラッチからの運動系列獲得

これまで用いてきた枠組では，白紙の状態からまったく新しい運動パターンを獲得することはできない．教師の手本を基にして学習を行なうからである．教師なし学習のうち，運動系列を獲得する能力を持つもののひとつに強化学習がある [78]．強化学習では時々刻々の入力に対する望ましい出力を学習者に与えることなく，単に学習者の行動を評価し，強化信号 (報酬や罰) として学習者に与える．学習者は強化信号を用いてタスクに対する解を自ら獲得する．強化学習による運動制御では cart-pole 制御や起き上がり運動の獲得などが行なわれている [5, 11,

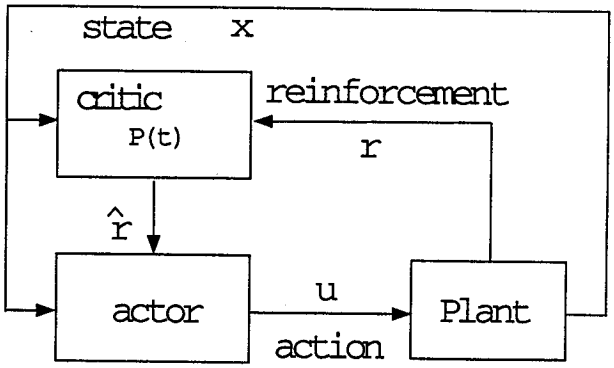


図 5.1: Actor-Critic 型強化学習のブロック線図.

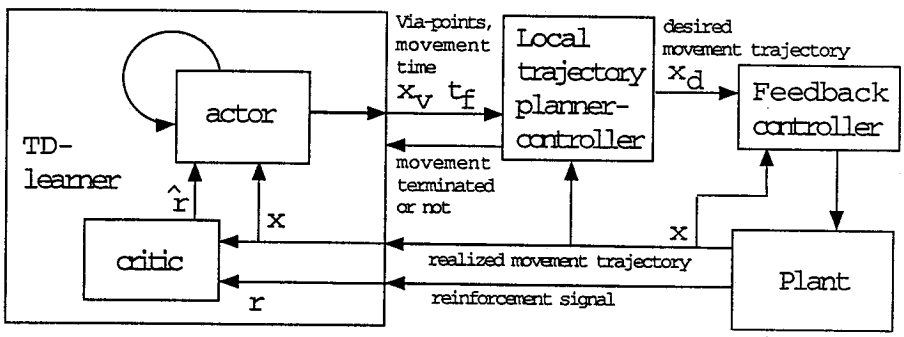


図 5.2: 経由点を用いた TD 学習.

12, 66]. これらのモデルでは, 図 5.1 に示す actor-critic 型の強化学習によって時々刻々の入力に対して最適なフィードバック制御を行なう.

今後, 経由点表現と局所的な軌道生成, 強化学習を組み合わせてスクラッチからの運動系列獲得の可能性を検討していく. 具体的には以下のように進める予定である.

方法 1: 図 5.2 に示すように経由点表現を用いて強化学習を行なう. アクターの出力は次の経由点の位置と時刻. 経由点と経由点の間の時間は変動するが, 等時間間隔と仮定して TD-learning (Temporal Difference learning) を行なってもうまくいくかどうかも確かめる (5.4 節).

方法 2: 図 5.3 に示すように Continuous TD-learning[11] [1]などを用いてある程度成功に近付くまで強化学習を行なう. 得られた運動軌道から経由点を抽出し, 広義ニュートン法で学習を行なう. 経由点表現に切替えることにより, 次元が減り探索空間を狭い範囲に限定できるので, より少ない試行回数で, 運動系列を獲得できると期待される.

[1]TD-learning は一般的に離散時間で定義されているが Continuous TD-learning は連続時間版の TD-learning である.

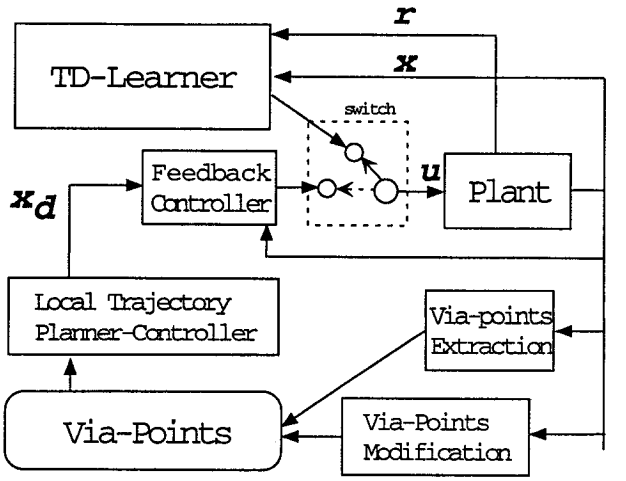


図 5.3: TD- 学習から経由点を用いた広義ニュートン法への切替え.

5.3.2 サブゴール設定の自動化

本論文では，これまでサブゴールの設定は実験者の経験的知識や試行錯誤によって行なわれてきた．今後，サブゴール設定が自動的に行なわれるためにはどのようにすればよいかを検討していく．

運動指令と結果の関係 (ダイナミクス) が質的に変化する場所，あるいは経由点をサブゴールに設定する．タスクのダイナミクスの解析には多重対内部モデル: MPFIM (Multiple-Paired-Forward-Inverse-Models)[87] を用いる．このモデルは複数の順モデルと逆モデル (線形の順モデルと局所的に定義された評価関数から決まる局所的な最適制御器) から，タスクのダイナミクスに応じて最適なモジュールを選択するものである．

手本の運動軌道があらかじめ与えられているときはタスクのダイナミクスが質的に変化したところがわかるので，MPFIM (Multiple-Paired-Forward-Inverse-Models) を用いてサブゴールをあらかじめ決定しておくことができると期待される．また，観測すべきダイナミクスをどのように設定するかということも，手本の運動軌道があらかじめ与えられているときは選択しやすい．

しかし手本の運動軌道が与えられていない場合には，タスクのダイナミクスが質的に変化するような状況に偶然遭遇するまではサブゴールを決定できない．また，手本の運動軌道が与えられていない場合は何を観測すべきダイナミクスとして選ぶかが分からない．

手本の運動軌道があらかじめ与えられているか否かに関係なく観測すべきダイナミクスの選択が困難な場合もあると考えられる．

具体例:(振り子の振り上げ) 図 5.4 に振り子の振り上げの場合の例を示す．振り子の振り上げでは振り子が下を向いている時と上を向いている時とでは回転軸に加わる制御入力と振り子が

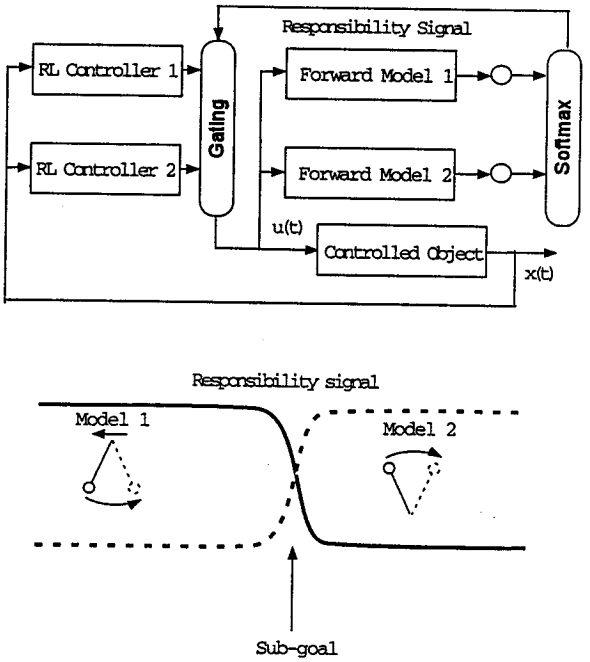


図 5.4: MPFIM (Multiple-Paired-Forward-Inverse-Models) を用いた振り子の振り上げのサブゴール設定.

回転しょうとする方向が異なる．振り子が真横を向いた時，MPFIM (Multiple-Paired-Forward-Inverse-Models) のサブモジュールの切り替えが起こると考えられる．したがって，このときの状態がサブゴールとして選ばれると予想される．

5.3.3 local trajectory planner-controller

従来使用してきたFIRMでは与えられた始点と経由点，終点を通る軌道を運動時間全体にわたって生成する．したがって視覚フィードバックなどを用いた運動途中での軌道修正が不可能であった．

前節で述べた強化学習と経由点を組み合わせる方法1, 2や視覚フィードバックによる制御を行なうために，局所的な時間幅での軌道生成を逐次行なう軌道計画器を構成したい．

Hoff-Arbib のモデルはこの目的にかなうモデルのひとつである．図5.5 と以下の式に Hoff-Arbib のモデルを示す．

$$\mathbf{q} = (x, v, a)^{\mathrm{T}}, \tag{5.1}$$

$$\mathbf{T} = (T, 0, 0)^{\mathrm{T}}, \tag{5.2}$$

$$\frac{d}{dt}\mathbf{q} = A\mathbf{q} + B\mathbf{T}, \tag{5.3}$$

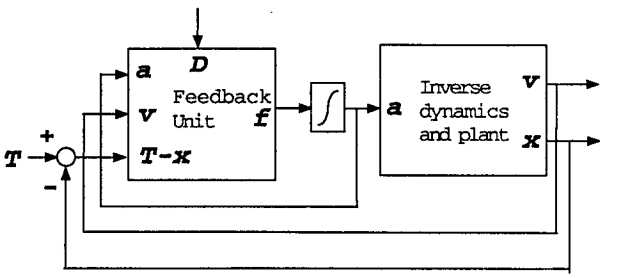


図 5.5: Hoff & Arbib の躍度最小軌道生成器.

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -60/D^3 & -36/D^2 & -9/D \end{pmatrix}, \tag{5.4}$$

$$B = \begin{pmatrix} 0 \\ 0 \\ 60/D^3 \end{pmatrix}, \tag{5.5}$$

$$D = t_f - t \tag{5.6}$$

Hoff-Arbib のモデルは現在の位置と速度から目標位置 T と到達時間 t_f を与えると躍度最小軌道 x を生成するフィードバックコントローラである．しかし Hoff-Arbib のモデルにはいくつかの不満がある．最適化基準として躍度最小が必ずしも正しいとは限らないことと，パラメータがあらかじめ組み込まれていることである．

最近，躍度最小基準もトルク変化最小基準も用いないサッカード眼球運動とヒト腕の運動軌道を再現できる新しいモデルが出てきた [23]．このモデルは運動指令の大きさに依存したノイズが運動指令に加えられるとき，運動後の分散が最も小さくなるような軌道が生成されるというものである．このモデルは非線形計画問題のなかの二次計画問題 (quadratic programming problem) に帰着され，MATLAB などの汎用数値計算ソフトウエアによって解を得ることができる．このモデルは強化学習を用いて解くことができる可能性がある．今後，上述の軌道計画モデルを強化学習の枠組で解くことを目標としたい．

5.4 スクラッチからの運動系列獲得; 経由点表現を用いた強化学習による振り子の振り上げ運動の獲得.

本節では経由点と強化学習を組み合わせた枠組を新たに提案し，それを用いて cart-pole 制御のための運動系列の学習が可能であることをシミュレーションにより示す．この新しい枠組では以下の要素を取扱う．

(1) 経由点と最適化原理に基づく軌道生成.

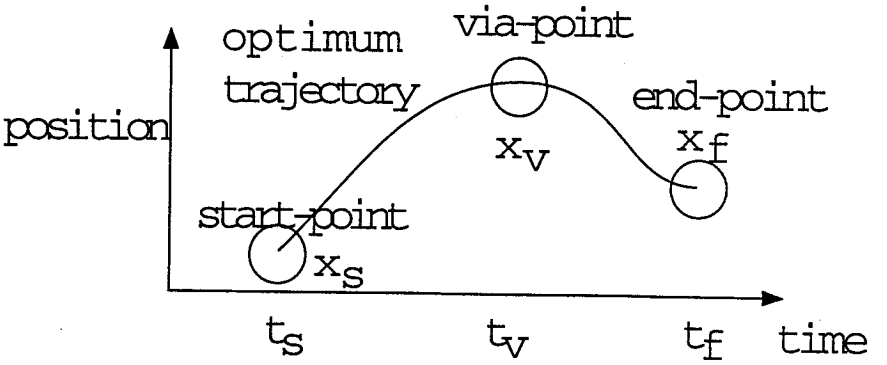


図 5.6: 経由点と最適軌道.

(2) 時間に関して局所的な軌道生成 (local trajectory planner-controller).

(3) 強化学習.

(4) サブゴール設定の自動化.

本稿ではまず第 1 ステップとして，これらのうち (2) と (3) に関して考える．(1) に関しては，本節では躍度最小軌道を仮定する．(4) に関しては，本節ではひとつのゴールのみで学習可能なタスクとして振り子の振り上げを選んだ.

5.4.1 経由点と最適軌道の時間局所的な生成.

図 5.6 に始点，終点とひとつの経由点と，再構成された最適軌道を示す．心理物理実験から示唆される人腕の最適軌道モデルとしては躍度最小軌道 (minimum jerk trajectory) やトルク変化最小軌道 (minimum torqe change trajectory) などが提案されている．始点 (運動の開始時)，経由点，終点 (運動の終了時) が与えられた時，これらの点を通る最適軌道計画問題を考える．始点，経由点，終点における位置，速度，加速度をそれぞれ $(x_s, \dot{x}_s, \ddot{x}_s)$, $(x_v, \dot{x}_v, \ddot{x}_v)$, $(x_f, \dot{x}_f, \ddot{x}_f)$ とする．始点と経由点間， 経由点と終点の時刻をそれぞれ t_v,t_f とする．最適軌道を躍度最小軌道 (5 次多項式) とすると， $\dot{x}_f = 0$, $\ddot{x}_f = 0$ とおいたとき $\dot{x}_v$, $\ddot{x}_v$ が容易に決定でき，最適軌道が計算できる.

複雑な運動に対しては複数の経由点が必要となるが，すべての経由点から大域的な最適軌道を計算する従来の FIRM をそのまま用いたのでは，運動途中での軌道修正が不可能である.

本節では運動途中での軌道修正を可能とするために図 5.7 に示すように，局所的な経由点と最適軌道の生成を繰り返して全体の運動を行なう．このとき，現時点とひとつ先の 2 つの経由点のみを用いたのでは，その経由点での速度と加速度を決定できない．仮にその経由点での速度と加速度を 0 とするとぎこちない動作になり不自然である．局所的な経由点を滑らかにつな

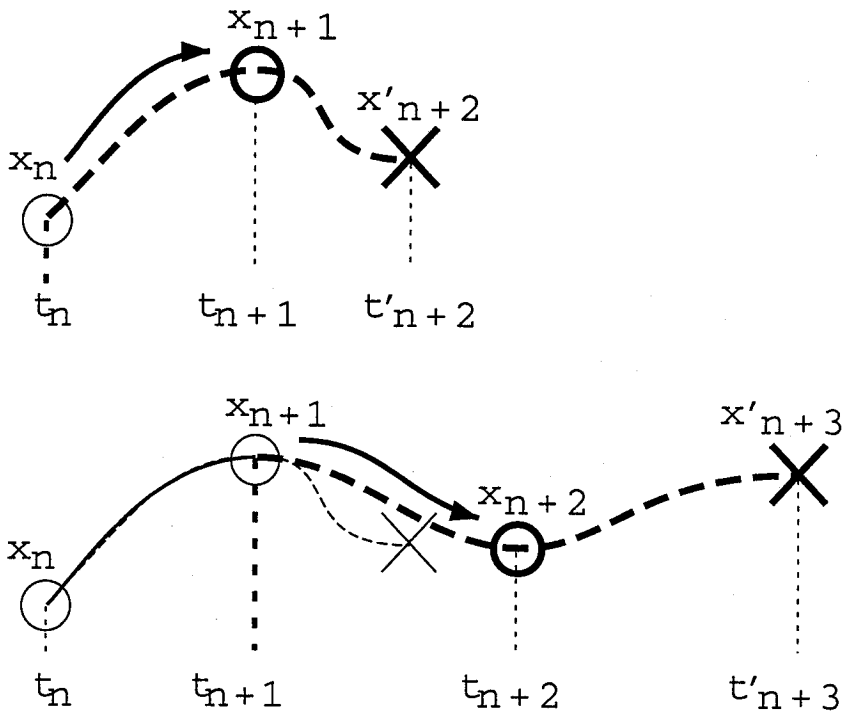


図 5.7: 局所的な経由点と最適軌道の生成: ○ は経由点の位置, × は仮の経由点の位置を示す. 破線は計画された軌道, 実線は実行された軌道, 細い破線は破棄された軌道をそれぞれ示す.

いでいくようにするために, 図 5.7上 に示すように, まず時刻 t_n における経由点の位置を x_n とするとき, ひとつ先の経由点の位置と時間 x_{n+1}, t_{n+1} を決め, 同時にふたつ先の経由点の位置と時間を暫定的に x'_{n+2}, t'_{n+2} とする. つぎに仮の経由点 x'_{n+2} における速度, 加速度を 0 として最適軌道 $(x_n \rightarrow x_{n+1} \rightarrow x'_{n+2})$ を生成する. 上で得られた最適軌道のうち前半部分 $(x_n \rightarrow x_{n+1})$ を目標軌道として制御を行なう. 時刻 t_{n+1} になった時点で (図 5.7下), すでに生成されている最適軌道 $(x_{n+1} \rightarrow x'_{n+2})$ を破棄して新たに経由点 x_{n+2}, x'_{n+3} を上記と同様の手順で生成する. このようにして次の経由点に移行するごとに, つぎつぎと経由点と最適軌道の生成を繰り返して運動を行なう.

5.4.2 経由点を用いた TD(Temporal Difference) 学習

actor-critic 型強化学習

本節で提案する経由点表現を用いた強化学習の枠組を図 5.8 に示す. このモデルは actor-critic 型の強化学習に経由点表現を組み込んだ形となっている. actor-critic 型の強化学習モデルは, actor と呼ばれる制御ネットワークと critic と呼ばれる評価ネットワークから成る. actor は状態入力に対するフィードバック制御出力を強化学習の原理により学習する. critic は今の制御対象の状態がどれぐらい失敗に近いか (あるいは成功に近いか) を示す評価関数を TD 誤差を最小

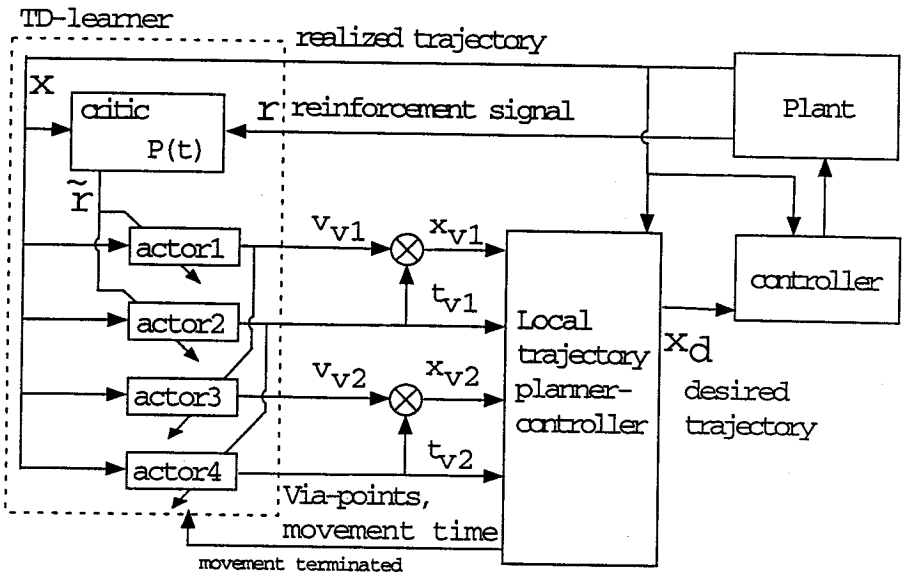


図 5.8: 経由点を用いた TD 学習のブロック線図.

化するように学習する.

critic の学習

n 番目の経由点の時刻を t_n と表す. ある状態変数と制御変数 $\mathbf{x}(t_n), x_{\rm d}(t_n)$ のもとで報酬 $R(t_n) = r(\mathbf{x}(t_{n-1}), u(t_{n-1}))$ を受け取るとき以下のように価値関数を定義する.

$$V(\mathbf{x}(t_n)) = R(t_n) + \gamma\, R(t_{n+1}) + \gamma^2\, R(t_{n+2}) + \cdots \tag{5.7}$$

$0 \leq \gamma \leq 1$ は割引率 (discount factor) である. ここで価値関数の推定値を $P(\mathbf{x}(t_n))$ とすると, 推定値の非整合性を表す TD 誤差 $\tilde{r}$ は以下のようになる.

$$\tilde{r}(t_n) = R(t_n) + \gamma\, P(\mathbf{x}(t_{n+1})) - P(\mathbf{x}(t_n)) \tag{5.8}$$

critic は TD 誤差 $\tilde{r}$ が 0 になるように学習を行なう.

actors

actor の関数近似器の出力を $z_i(\mathbf{x}(t_n))$ とする. actor の出力は以下のようにする.

$$v_{\rm v1}(\mathbf{x}(t_n)) = z_1(\mathbf{x}(t_n)) + z_{\rm p1}(t_n) \tag{5.9}$$

$$t_{\rm v1}(\mathbf{x}(t_n)) = z_2(\mathbf{x}(t_n)) + z_{\rm p2}(t_n) \tag{5.10}$$

$$v_{\rm v2}(\mathbf{x}(t_n)) = z_3(\mathbf{x}(t_n)) \tag{5.11}$$

$$t_{\rm v2}(\mathbf{x}(t_n)) = z_4(\mathbf{x}(t_n)) \tag{5.12}$$

v_{v1}, v_{v2} は単位時間あたりの経由点の移動量，t_{v1}, t_{v2} は経由点間の時間をそれぞれ表す．ただし $t_{v1}(\mathbf{x}(t_n)) < t_{\min}$ のときは $t_{v1} = t_{\min}$ とする．t_{v2} についても同様とする．$z_{p1}(t_n), z_{p2}(t_n)$ は探索を行なうための摂動である．これらから経由点の時間と位置を以下のように計算する．

$$t_{n+1} = t_n + t_{v1} \tag{5.13}$$

$$t'_{n+2} = t_{n+1} + t_{v2} \tag{5.14}$$

$$x_{n+1} = x_n + v_{v1}\, t_{v1} \tag{5.15}$$

$$x'_{n+2} = x_{n+1} + v_{v2}\, t_{v2} \tag{5.16}$$

actor1, 2 の学習

actor1 と actor2 に関しては，大きな $\tilde{r}$ を生じた制御出力が同じ状態で再び出力されるように学習する．学習のための入出力ペアーと教師信号は以下のようになる．

$$\mathbf{x}(t_{n-1}),\ z_1(\mathbf{x}(t_{n-1})),\ z_1(\mathbf{x}(t_{n-1})) + z_{p1}(t_{n-1})\, \tilde{r}(t_{n-1})$$

$$\mathbf{x}(t_{n-1}),\ z_2(\mathbf{x}(t_{n-1})),\ z_2(\mathbf{x}(t_{n-1})) + z_{p2}(t_{n-1})\, \tilde{r}(t_{n-1})$$

actor3, 4 の学習

図 5.7 において x_{n+2} と x'_{n+2} および t_{n+2} と t'_{n+2} がそれぞれ等しくなるように学習を行なうと，仮の経由点が正しく予測されるようになる．すなわち，学習のための入出力ペアーと教師信号は以下のようにする．

$$\mathbf{x}(t_{n-2}),\ z_3(\mathbf{x}(t_{n-2})),\ z_1(\mathbf{x}(t_{n-1})) \tag{5.17}$$

$$\mathbf{x}(t_{n-2}),\ z_4(\mathbf{x}(t_{n-2})),\ z_2(\mathbf{x}(t_{n-1})) \tag{5.18}$$

5.4.3 振り子の振り上げ運動学習数値実験

前節の経由点表現を用いた強化学習を図 5.9 のような cart-pole 型の振り子振り上げ問題に適用した．

下に垂れた ($\theta = -\pi$) 振り子を，回転軸を x 方向に動かして振り上げ，倒立状態 ($\theta = 0$) に保持することが運動の目標である．振り子の力学モデルは

$$\dot{\theta}_{i+1} = (1 - \alpha_1)\dot{\theta}_i + \frac{\Delta}{l}\left(g \sin(\theta_i) + \ddot{x}_i \cos(\theta_i)\right) \tag{5.19}$$

とした．ここで $\theta_i, \dot{\theta}_i, \ddot{\theta}_i, \ddot{x}_i$ はそれぞれ時刻 i における振り子の角度，角速度，角加速度，回転軸に対する横方向の加速度である．α_1 は粘性項，$\Delta = 1/60s$ は刻み時間，$g = 9.81m/s^2$ は重力加速度，$l = 0.5m$ は振り子の長さをそれぞれ示す．ここでは $\alpha_1 = 0.01$ とした．

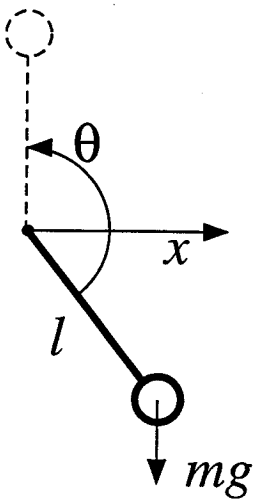


図 5.9: 振り子の力学モデル.

n 番目の経由点の時刻における状態変数は

$$\mathbf{x}(t_n) = \left\{\theta(t_n), \dot{\theta}(t_n), x(t_n), \dot{x}(t_n)\right\} \tag{5.20}$$

報酬は

$$r(t_n) = \begin{cases} (1+\cos(\theta(t_n)))/2 & \text{if } |\theta(t_n)| \leq \pi \\ 0 & \text{otherwise} \end{cases} \tag{5.21}$$

とした. 最長運動時間は 12 秒とした. ただし $|x| \geq 1$ のとき $r = -1$ とし終了とした.

critic と actor は Adaptive Gaussian Softmax Basis Function を用いた [65]. critic の出力は以下のように計算した. k 番めのユニットの活性化関数を

$$a_k(\mathbf{x}(t_n)) = \exp\left(\frac{||M_k(\mathbf{x}(t_n) - \mathbf{c}_k)||}{2}\right) \tag{5.22}$$

とする. ただし $\mathbf{c}_k$ は活性度の中心, M_k は活性化関数の形状を決定する行列である. Softmax 基底関数は以下のように与えられる.

$$b_k(\mathbf{x}(t_n)) = \frac{a_k(\mathbf{x}(t_n))}{\sum_{l=1}^{K} a_l(\mathbf{x}(t_n))} \tag{5.23}$$

ただし K は基底関数の数である. critic の出力は

$$P(\mathbf{x}(t_n)) = \sum_{k=1}^{K} w_k b_k(\mathbf{x}(t_n)) \tag{5.24}$$

と得られる. 誤差がある基準 $e_{\max}$ より大きく, すべてのユニットの活性度があるしきい値 $a_{\min}$ より小さければ, 新しいユニットを配置する. 新しいユニットは

$$\mathbf{c}_k = \mathbf{x}(t_n),\ M_k = \mathrm{diag}(\mu_i),\ w_k = z(\mathbf{x}(t_n))$$

で初期化する．

critic の学習は以下のような TD(λ) 学習を行なった．

$$\Delta w_k = 0.3\,\tilde{r}(t_{n-1})\,e_k(t_{n-1}) \tag{5.25}$$

$$e_k(t_n) = \gamma\,\lambda\,e_k(t_{n-1}) + b_k(\mathbf{x}(t_n)) \tag{5.26}$$

ここで $e_k(t_n)$ は各シナプスの履歴 (eligibility trace) を表す．critic の各パラメータは

$$\gamma = 0.8,\ \lambda = 1.0,\ M_k = \mathrm{diag}(\pi/4, \pi, 0.5, 0.5),$$

$$a_{\min} = 0.5,\ e_{\max} = 0.0001$$

とした．

actor の出力も critic と同様に計算される．ただし出力値の範囲を

$$-1 \leq v_{\mathrm{v1}}(\mathbf{x}(t_n)) \leq 1 \tag{5.27}$$

$$0.05 \leq t_{\mathrm{v1}}(\mathbf{x}(t_n)) \leq 1 \tag{5.28}$$

となるように制限した．actor1, 2 には探索を行なうための摂動 $z_{\mathrm{p1}}, z_{\mathrm{p2}}$ を以下のように与えた．

$$|z_{\mathrm{p1}}| < 0.3\,\exp(-3\,P(t_n)) \tag{5.29}$$

$$|z_{\mathrm{p2}}| < \exp(-2P(t_n)) \tag{5.30}$$

actor の各パラメータは

$$M_k = \mathrm{diag}(\pi/8, \pi/2, 0.5, 0.5),$$

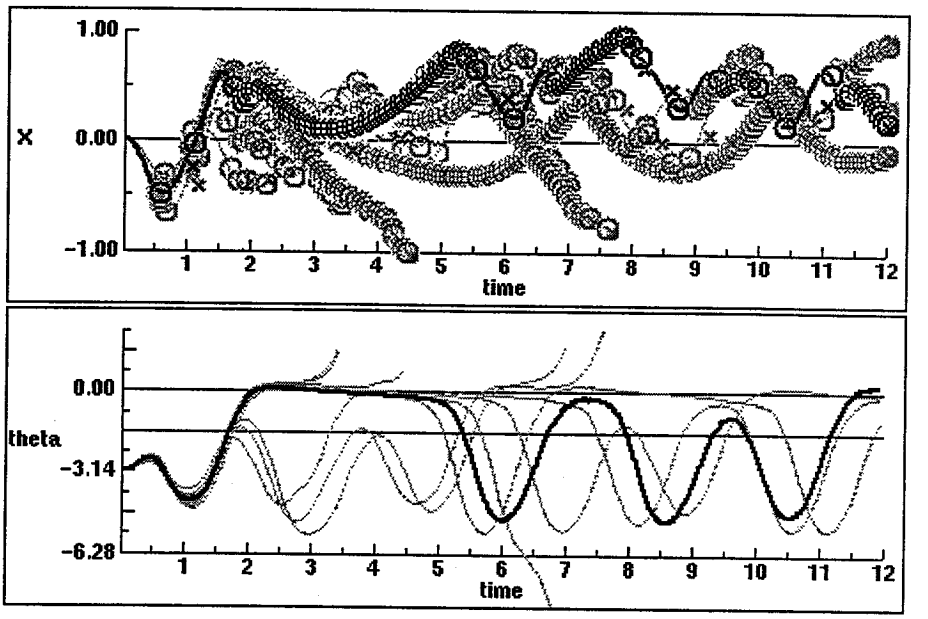
$$a_{\min} = 0.6,\ e_{\max} = 0.0001$$

とした．

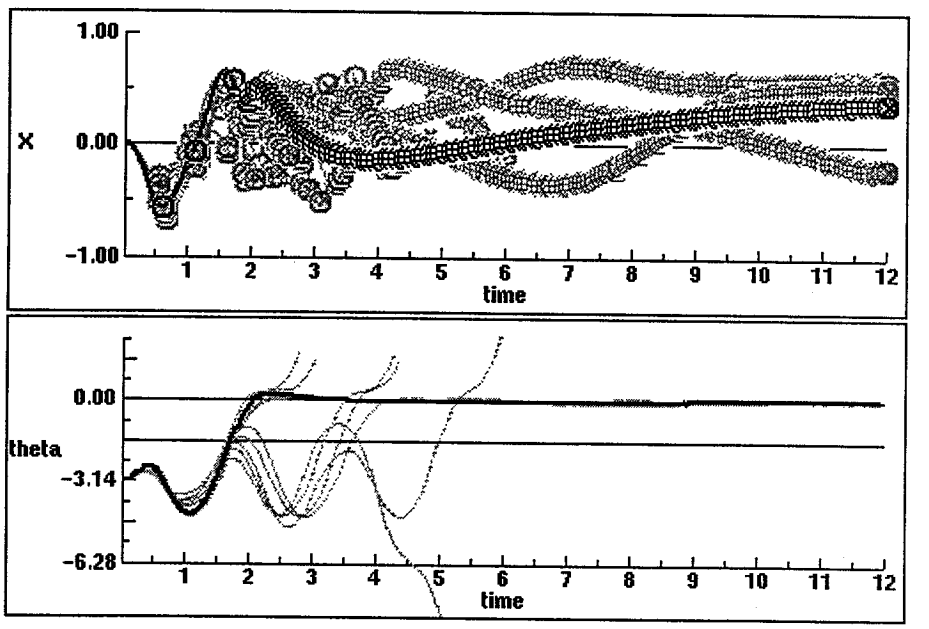
実験結果

図 5.10 に 1000 回試行後と 2000 回試行後の振り上げの様子を示す．予備的な振りで慣性を蓄えてから倒立状態に振り上げた後，保持に成功するようになっている．運動の始めの部分では経由点の時間間隔が長くとられフィードフォワード的に運動を行なっている．振り子が上のほうに振り上がった状態になると自動的に経由点の時間間隔が短くなり，細かいフィードバックを行なってバランスを保とうとするようになっている．

本節では経由点表現と強化学習を組み合わせた枠組が振り子の振り上げ運動獲得に適用可能であることを示した．しかし，2000 回程度と比較的多くの試行回数が必要であった．経由点間



(A)



(B)

図 5.10: (A)991~1000 回試行後の振り上げと (B)1991~2000 回試行後の振り上げ．太い線で示してある 1000 回目と 2000 回目の試行は摂動を与えていない．上から x, θ の時間変化をそれぞれ示す．x のグラフで ∘ は経由点，× は仮の経由点をそれぞれ示す．

の時間を 0.05 に固定した場合は非線形フィードバック則を獲得する従来の方式に非常に近いものとなるが，この場合も同程度の試行回数を必要とした．パラメータをファインチューニングすれば数100回程度の試行回数で成功できるようにできるかも知れない．しかし，現時点のシミュレーション結果からは，本研究で提案する枠組が工学的な応用面においては，従来の方法と比べて明かなアドバンテージをもつとはいい難い．本稿では状態変数の時間遅れは無いものとしたが，今後，時間遅れのある場合も調べる必要がある．生物のフィードバックループは時間遅れが大きくゲインが小さい．したがって高速で協調的な運動は前向き制御によって実行されなくてはならない．本稿で提案した枠組は，脳のモデルとして適していると期待される．今後，生物の振舞いを良く再現できる脳のモデルを目指し，状態変数の時間遅れが大きい場合や多自由度の運動制御などを調べていきたい．

第 6 章

まとめ

本論文では，最適化原理に基づく双方向性神経回路モデルを用いた見まねによる運動学習ロボットの検証実験を行ない，見まねによる運動学習の枠組の学習可能性と汎用性について考察した．以下に本論文の結論として各章ごとの要点を述べる．

6.1 けん玉

第 2 章では，けん玉の運動学習ロボットに関して述べた．けん玉は比較的単純な運動であった．経由点抽出のタスク表現としての適切さを評価した．しかし，見まねによる運動学習の枠組の学習可能性と汎用性を調べるために，次節の実験を行なう必要があった．

6.2 テニスサーブ

第 3 章では，テニスサーブの運動学習ロボットに関して述べた．本研究で行なったテニスサーブは，玉を放り上げる動作と，玉を打つ動作の 2 段階の学習が必要で，第 2 章で述べたけん玉の運動学習に比べて，学習が困難であった．その理由はサブタスク 1 の学習とサブタスク 2 の学習が互いに強い依存性を持っているためであった．この問題点を解決するために，最初にとった方策が，学習の制御変数として選ぶ経由点を制限することであった．しかし，その選びかたは実験者が試行錯誤で選び出したもので，根拠がないものであった．そこで，ヤコビ行列を解析することによって経由点の選択を適切に行なう方法を提案した．

6.3 振り子の振り上げ

第 4 章では，振り子の振り上げの運動学習ロボットに関して述べた．広義ニュートン法だけに限らず，ある軌道のまわりでの線形近似を用いることは非常に多い．学習の結果，軌道そのものが変化するので，当然，その軌道のまわりでの線形近似は最初のものとは異なってくる．第 2 章で述べたけん玉の運動学習や第 3 章で述べたテニスサーブの運動学習に関しては，軌道

の変化が比較的少ないので，最初の軌道のまわりで求めた線形近似をそのまま用い続けることが可能であった．しかし，振り子のような非線形性の強い制御対象の学習制御を行なうとき，最初の軌道のまわりで求めた線形近似をそのまま用い続けたのでは，学習が発散してしまう．振り子の場合には振り子が下で振れているときと，上で振れているときとでは，軸の動かし方に対して振り子の振るまいがまったく変わってしまうからである．学習の途中で線形近似を行なえばよいのであるが，線形近似を行なう時期を自動的に決定することは非常に困難である．また，線形近似を行なうためにだけ試行を行なうことは非常に効率が悪い．以上の問題を解決する為に学習の履歴を用いて，線形近似を行なう方法を提案した．

6.4 今後の課題

第5章では，第4章までの運動学習の枠組の問題点と今後の課題に関して述べた．第4章までの従来の枠組では以下の問題点が未解決のまま残っている．1) 運動を行なう前に全体の軌道生成を行なうので運動途中での軌道修正が不可能である．2) サブゴールの設定が実験者の直観と試行錯誤によって与えられた．3) 手本の運動を超えられない．手本より上手な運動はできない．4) 白紙の状態からまったく新しい運動パターンは獲得できない．

これらの課題を解決するために，階層的強化学習の新しい枠組を構築する必要がある．

スクラッチからの運動系列獲得の枠組の構築のために，まず第1ステップとして経由点と最適化原理に基づくモデルに強化学習を組み込んだモデルの数値実験を行なった．

謝辞

本研究を進めるに当たって，ご指導と助言をいただいたATR人間情報通信研究所川人光男室長，大阪大学基礎工学部生物工学科福島邦彦教授に感謝いたします．川人室長には神経回路モデルを研究する機会を与えていただき，大阪大学基礎工学部生物工学科4年在学中，大阪大学大学院基礎工学研究科博士前期課程在学中，ATR人間情報通信研究所在職中および科学技術振興事業団奉職中を通じて様々な知識と有益な議論を賜わりました．福島教授には大阪大学大学院基礎工学研究科博士後期課程在学中に様々な知識と有益な議論を賜わりました．また本論文の主査をしていただき，本論文をまとめる際にも貴重なご意見をいただきました．本論文の副査をしていただきました大阪大学基礎工学部生物工学科佐藤俊輔教授，中野馨教授の皆様に感謝いたします．最後に貴重な御意見をいただきました大阪大学基礎工学部生物工学科倉田耕治講師および庄野逸助手の皆様，大阪大学基礎工学部生物工学科の皆様，ATR人間情報通信研究所第3研究室の皆様，および科学技術振興事業団川人学習動態脳プロジェクトの皆様に感謝いたします．

参考文献

[1] E. W. Aboaf, C. G. Atkeson, and D. J. Reinkensmeyer. Task-level robot learning. In *Procceedings, IEEE International Conference on Robotics and Automation*, pp. 24–29, Philadelphia, April 1988.

[2] 有本卓. ロボット知能とタスク理解. システム／制御／情報, Vol. 37, No. 10, pp. 569–575, 1993.

[3] 浅田春比古. マニピュレーションの知能. 日本ロボット学会誌, Vol. 5, No. 6, pp. 487–494, 1987.

[4] C. G. Atkeson and S. Schaal. Robot learning from demonstration. In *International Conference on Machine Learning*, 1997.

[5] A. G. Barto, R. S. Sutton, and C. W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 13, pp. 834–846, 1983.

[6] D. J. Bennett. Torques generated at the human elbow joint in response to constant position errors imposed during voluntary movements. *Experimental Brain Research*, Vol. 95, pp. 488–498, 1993.

[7] D. J. Bennett, J. M. Hollerbach, Y. Xu, and I. W. Hunter. Time-varying stiffness of human elbow joint during cyclic voluntary movement. *Experimental Brain Research*, Vol. 88, pp. 433–442, 1992.

[8] J. J. Craig. *Introduction to robotics: Mechanics and control.* Addison-Wesley, Reading, MA, second edition edition, 1989.

[9] J. Decety, D. Perani, M. Jeannerod, V. Bettinardi, B. Tadary, R. Woods, J. C. Mazziotta, and F. Fazio. Mapping motor representations with positron emission tomography. *Nature*, Vol. 371, No. 13, pp. 600–603, Oct 1994.

[10] G. di Pellegrino, L. Fadiga, L. Fogassi, V. Gallese, and G. Rizzolatti. Understanding motor events: a neurophysiological study. *Experimental Brain Research*, Vol. 91, pp. 176–180, 1992.

[11] K. Doya. Temporal difference learning in continuous time and space. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems 8*, pp. 1073–1079. MIT Press, Cambridge, MA, USA, 1996.

[12] K. Doya. Efficient nonlinear control with actor-tutor architecture. In M. C. Mozer and M. I. Jordan, editors, *Advances in Neural Information Processing Systems 8*. MIT Press, Cambridge, MA, USA, 1997.

[13] R. Featherstone. *Robot dynamics algorithms.* Kluwer, Boston, 1987.

[14] J. R. Flanagan, D. J. Ostry, and A. G. Feldman. Control of trajectory modifications in target-directed reaching. *Journal of Motor Behavior*, Vol. 25, No. 3, pp. 140–152, 1993.

[15] T. Flash. The control of hand equilibrium trajectories in multi-joint arm movements. *Biological Cybernetics*, Vol. 57, pp. 257–274, 1987.

[16] T. Flash and E. Henis. Arm trajectory modifications during reaching towards visual targets. *Journal of Cognitive Neuroscience*, Vol. 3, No. 3, pp. 220–230, 1991.

[17] T. Flash and N. Hogan. The coordination of arm movements: An experimentally confirmed mathematical model. *Journal of Neuroscience*, Vol. 5, pp. 1688–1703, 1985.

[18] T. Flash and N. Hogan. The coordination of arm movements: An experimentally confirmed mathematical model. *Journal of Neuroscience*, Vol. 5, pp. 1688–1703, 1985.

[19] T. Fujii, T. Yasuda, S. Yokoi, and J. Toriwaki. A virtual pendulum manipulation system on a graphic workstation. In *Proceedings, 2nd IEEE International Workshop on Robot and Human Communication*, 1993.

[20] M. Fujita. Adaptive filter model of the cerebellum. *Biological Cybernetics*, Vol. 45, pp. 195–206, 1982.

[21] A. P. Georgopoulos, J. F. Kalaska, and J. T. Massey. Spatial trajectories and reaction times of aimed movements: Effects of practice, uncertainty, and change in target location. *Journal of Neurophysiology*, Vol. 46, No. 4, pp. 725–743, 1981.

[22] H. Gomi, Y. Koike, and M. Kawato. Human hand stiffness during discrete point-to-point multi-joint movement. In *Proceedings of IEEE Engineering in Medicine and Biology Society*, pp. 1628–1629, 1992.

[23] C. M. Harris and D. M. Wolpert. Signal-dependent noise determines motor planning. *Nature*, Vol. 394, No. 20, pp. 780–784, August 1998.

[24] 肥爪彰夫, 吉田雅彦, 吉川順偉, 竹内章, 北澤京介. 高次外乱オブザーバによるロボットアーム (とめけん) の加速度基準型高速運動制御. 日本機械学会ロボティクスメカトロニクス講演会論文集, pp. 1272–1277, 1994.

[25] B. Hoff and M. A. Arbib. Models of trajectory formation and temporal interaction of reach and grasp. *Journal of Motor Behavior*, Vol. 25, No. 3, pp. 175–192, 1993.

[26] 市田浩三, 吉本富士市. スプライン関数とその応用. 教育出版, 1979.

[27] K. Ikeuchi, M. Kawade, and T. Suehiro. Assembly task recognition with planar, curved and mechanical contacts. In *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 688–694, Atlanta, May 1993.

[28] 磯部倫明, 川人光男, 鈴木良次. 関節角座標および視覚座標における産業用マニピュレータの繰り返し学習制御. 信学技報, MBE87-134, pp. 241–248, 1988.

[29] M. Ito. Neurophysiological aspects of the cerebellar motor control system. *International Journal of Neurology*, Vol. 7, pp. 162–176, 1970.

[30] M. Ito. *The cerebellum and neural control.* Raven Press, New York, 1984.

[31] 伊藤宏司, 伊藤正美. 生体とロボットにおける運動制御. 計測自動制御学会, 1991.

[32] M. I. Jordan, T. Flash, and Y. Arnon. A model of the learning of arm trajectories from spatial deviations. *Journal of Cognitive Neuroscience*, Vol. 6, No. 4, pp. 359–376, 1994.

[33] S. B. Kang and K. Ikeuchi. Toward automatic robot instruction from perception — recognizing a grasp from observation. *IEEE Transaction on Robotics and Automation*, Vol. 9, pp. 432–443, 1993.

[34] M. Katayama and M. Kawato. Virtual trajectory and stiffness ellipse during muti-joint arm movement predicted by neural inverse models. *Biological Cybernetics*, Vol. 69, pp. 353–362, 1993.

[35] M Kawato. Motor theory of speech perception revisited from minimum-torque-change neural network model. In *Proceedings of 8th Symposium on Future Electron Devices*, pp. 141–150, 1989.

[36] M. Kawato. Optimization and learning in neural networks for formation and control of coordinated movement. In D. Meyer and S. Kornblum, editors, *Attention and Performance, XIV: Synergies in Experimental Psychology, Artificial Intelligence, and Cognitive Neuroscience - A Silver Jubilee*, pp. 821–849. MIT Press, Cambridge, MA, USA, 1992.

[37] M. Kawato, K. Furukawa, and R. Suzuki. A hierarchical neural-network model for control and learning of voluntary movement. *Biological Cybernetics*, Vol. 57, pp. 169–185, 1987.

[38] M. Kawato, F. Gandolfo, H. Gomi, and Y. Wada. Teaching by showing in kendama based on optimization principle. In M. Marinaro and P. G. Morasso (Eds.), editors, *Proceedings of the International Conference on Artificial Neural Networks*, pp. 6–29, Sorrento, Italy, 1994.

[39] M. Kawato, M. Isobe, and R. Suzuki. Coordinates transformation and learning control for visually-guided voluntary movement with iteration: A newton-like method in a function space. *Biological Cybernetics*, Vol. 59, pp. 161–177, 1988.

[40] 川人光男. 脳の計算理論. 産業図書, 1996.

[41] Y. Kobayashi, K. Kawano, A. Takemura, Y. Inoue, T. Kitamura, H. Gomi, and M. Kawato. Inverse-dynamics representation of complex spike discharges of purkinje cells in monkey cerebellar ventral paraflocculus during ocular following responses. In *Society for Neuroscience Abstracts*, Vol. 21, San Diego, 140, November 11-16 1995.

[42] 小池康春, 川人光男. 神経回路モデルを用いた表面筋電信号からの人腕の軌道生成. 信学技報, HC94-24, pp. 61–66, 1994.

[43] Y. Kuniyoshi. The science of imitation – towards physically and socially grounded intelligence –. *RWC Tech. Rep.*, Vol. TR-94001, pp. 123–124, 1994.

[44] Y. Kuniyoshi, M. Inaba, and H. Inoue. Learning by watching: extracting reusable task knowledge from visual observation of human performance. *IEEE Transaction on Robotics and Automation*, Vol. 10, No. 6, pp. 799–822, 1994.

[45] Y. Kuniyoshi, H Inoue, and M Inaba. Teaching by showing: Generating robot command sequences based on real time visual recognition of human pick and place actions. *JSRJ*, Vol. 9, pp. 295–303, 1991.

[46] 国吉康夫, 井上博允, 稲葉雅幸. 人間が実演して見せる作業の実時間視覚認識とそのロボット教示への応用. 日本ロボット学会誌, Vol. 9, No. 3, pp. 295–303, 1991.

[47] 国吉康夫. 自律ロボットにおける注意と視点：情報の分節・統合・共有. 人口知能学会研究会資料, Vol. SIG-FAI-9402-7(10/3-4), pp. 37–40, 1994.

[48] A. M. Liberman, F. S. Cooper, D. P. Shankweiler, and M Studdert-Kennedy. Perception of the speech code. *Psychological Review*, Vol. 74, pp. 431–461, 1967.

[49] A. M. Liberman and I. G Mattingly. The motor theory of speech perception revised. *Cognition*, Vol. 21, pp. 1–36, 1985.

[50] D. Marr. *Vision.* New York: W. H. Freeman & Company, 1982.

[51] L. Massone and E. Bizzi. A neural network model for limb trajectory formation. *Biological Cybernetics*, Vol. 61, No. 6, pp. 417–425, 1989.

[52] 三浦宏文, 井上博允, 下山勲, 井筒正夫, 光石衛, 竹中一起, 木村壮. 視覚をもつマニピュレータの動的制御 (ロボットによる「けん玉」の実現). 58 年度科学研究費補助金研究成果報告書, pp. 55–63, 1984.

[53] H. Miyamoto, F. Gandolfo, H. Gomi, S. Schaal, Y. Koike, R. Osu, E. Nakano, Y. Wada, and M. Kawato. A kendama learning robot based on a dynamic optimization theory. *4th IEEE International Workshop on Robot and Human Communication*, pp. 327–332, 1995.

[54] H. Miyamoto, F. Gandolfo, H. Gomi, S. Schaal, Y. Koike, R. Osu, E. Nakano, Y. Wada, and M. Kawato. A kendama learning robot based on a dynamic optimization principle. *International Conference on Neural Information Processing*, pp. 938–942, 1996.

[55] H. Miyamoto and M. Kawato. A tennis serve and upswing learning robot based on dynamic optimization theory. *Neural Networks*, Vol. 11, No. 7-8, pp. 1331–1344, 1998.

[56] H. Miyamoto, M. Kawato, T. Setoyama, and R. Suzuki. Feedback-error-learning neural network for trajectory control of a robotic manipulator. *Neural Networks*, Vol. 1, pp. 251–265, 1988.

[57] H. Miyamoto, S. Schaal, F. Gandolfo, H. Gomi, Y. Koike, R. Osu, E. Nakano, Y. Wada, and M. Kawato. A kendama learning robot based on dynamic optimization theory. *Neural Networks*, Vol. 9, No. 8, pp. 1281–1302, 1996.

[58] 宮本弘之, 五味裕章, 川人光男. 最適化原理に基づく見まねによるけん玉学習. pp. 453–459, 1985. 信学技報, NC20-5.

[59] 宮本弘之, F. Gandolfo, 五味裕章, Schaal, 小池康春, 大須理英子, 中野恵理, 和田安弘, 川人光男. 最適化原理に基づく見まねによるけん玉学習. 信学技報, NC94-143, pp. 223–230, 1995.

[60] 宮本弘之, 五味裕章, 川人光男. タスクレベルのロボット学習. システム / 制御 / 情報, Vol. 39, No. 9, pp. 419–426, 1995.

[61] 宮本弘之. ロボットへの教示方法. 計測と制御, 第 36 巻 9 号, pp. 627–630, 1997.

[62] 宮本弘之, 川人光男. 作業レベルのロボット学習のための見まねによる教示. 電子情報通信学会論文誌 D-II 分冊, 印刷中, 1998.

[63] 宮崎文夫. タスク理解と学習. システム／制御／情報, Vol. 37, No. 10, pp. 615–621, 1993.

[64] P. Morasso. Spatial control of arm movements. *Experimental Brain Research*, Vol. 42, pp. 223–227, 1981.

[65] J. Morimoto and K. Doya. Reinforcement learning of dynamic motor sequence: Learning to stand up. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, In press.

[66] 森本淳, 銅谷賢治. 強化学習による起き上がり運動パターンの獲得. 信学技報, NC97-28, pp. 25–32, 1997.

[67] 中村仁彦, 花房秀郎. 関節型ロボットアームの特異点低感度分解. 計測自動制御学会論文集, Vol. 20, No. 5, pp. 453–459, 1984.

[68] 浪花智英, 有本卓. 幾何学的拘束のあるロボットマニピュレータの学習制御. 計測自動制御学会論文集, Vol. 29, No. 4, pp. 411–418, 1993.

[69] 大道武生, 川内直人, 佐々木拓. 教示システムの現状と動向. 日本ロボット学会誌, Vol. 11, No. 1, pp. 31–35, 1993.

[70] 岡部佐規一, 神谷好承. 組立ロボットの最近の動向. 日本ロボット学会誌, Vol. 11, No. 1, pp. 58–61, 1993.

[71] 岡本良夫. 逆問題とその解き方. オーム社, 1992.

[72] D. I. Perrett, M. H. Harries, R. Bevan, S. Thomas, P. J. Benson, A. J. Mistlin, A. J. Chitty, J. K. Hietanen, and J. E. Ortega. Frameworks of analysis for the neural representation of animate objects and actions. *Journal of Experimental Biology*, Vol. 146, pp. 87–113, 1989.

[73] C. R. Rao and S. K. Mitra. *Generalized inverse of matrices and its applications.* New York; Wiley, 1971.

[74] 桜井明, 石井好, 高山文雄. スプライン関数入門. 東京電機大学出版局, 1981.

[75] S. Schaal, C. G. Atkeson, and S. Botros. What should be learned? In *Proceedings of Seventh Yale Workshop on Adaptive and Learning Systems*, pp. 199–204, 1992.

[76] B. Shepherd. Applying visual programming to robotics. In *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 707–712, Atlanta, May 1993.

[77] M. Shidara, K. Kawano, H. Gomi, and M. Kawato. Inverse-dynamics encoding of eye movement by purkinje cells in the cerebellum. *Nature*, Vol. 365, pp. 50–52, 1993.

[78] R. S. Sutton and A. G. Barto. *Reinforcement Learning, An Introduction.* A Bradford Book, MIT Press, Cambridge, MA, USA, 1998.

[79] S. Tachi, T. Maeda, R. Hirata, and H. Hoshino. A construction method of virtual haptic space. In *Proceedings of the ICAT'94 (4th International Conference on Artificial Reality and Tele-Existence)*, pp. 131–138, Tokyo, July 1994.

[80] T. Takahashi, H. Ogata, and S. Muto. A method for analyzing human assembly operations for use in automatically generating robot commands. In *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 695–700, Atlanta, May 1993.

[81] Y. Uno, M. Kawato, and R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement — minimum torque-change model. *Biological Cybernetics*, Vol. 61, pp. 89–101, 1989.

[82] Y. Uno, R. Suzuki, and M. Kawato. Minimum muscle-tension-change model which reproduces human arm movement. *Proceedings of the 4th Symposium on Biological and Physiological Engineering, in Japanese.*, pp. 299–302, 1989.

[83] Y. Wada and M. Kawato. A neural network model for arm trajectory formation using forward and inverse dynamics models. *Neural Networks*, Vol. 6, No. 7, pp. 919–932, 1993.

[84] Y. Wada and M. Kawato. A theory for cursive handwriting based on the minimization principle. *Biological Cybernetics*, Vol. 73, No. 1, pp. 3–13, 1995.

[85] Y. Wada, Y. Koike, E. V-Bateson, and M. Kawato. A computational model for cursive handwriting based on the minimization principle. In J.D. Cowan, G. Tesauro, and J. (eds.) Alspector, editors, *Advances in Neural Information Processing Systems 6*, pp. 727–734. Morgan Kaufmann, 1994.

[86] Y. Wada, Y. Koike, E. V-Bateson, and M. Kawato. A computational theory for movement pattern recognition based on optimal movement pattern generation. *Biological Cybernetics*, Vol. 73, No. 1, pp. 15–25, 1995.

[87] D. Wolpert and M. Kawato. Multiple paired forward and inverse models for motor control. *Neural Networks, Special Issue, in press*, 1998.

[88] 吉川恒夫. ロボット制御基礎論. コロナ社, 1988.

発表等

査読付き論文

- Miyamoto, H., Kawato, M., Setoyama, T., Suzuki, R.: Feedback-error-learning neural network for trajectory control of a robotic manipulator. Neural Networks, vol.1, pp.251–265 (1988).
- Miyamoto H, Schaal S, Gandolfo F, Gomi H, Koike Y, Osu R, Nakano E, Wada Y, Kawato M: A Kendama learning robot based on dynamic optimization theory. Neural Networks, vol.9, no.8, pp.1281–1302 (1996).
- Miyamoto H, Kawato M: A tennis serve and upswing learning robot based on dynamic optimization theory. Neural Networks, vol.11, no.7-8, pp.1311–1344 (1998).
- 宮本弘之, 川人光男: 作業レベルのロボット学習のための見まねによる教示. 電子情報通信学会論文誌 D-II 分冊, no.10, pp.2401–2410 (1998).

査読付き国際会議

- Miyamoto, H., Fukushima, K.: Recognition of spatio-temporal patterns by a multi-layered neural network model. Proceedings of the 1993 International Joint Conference on Neural Networks, pp.2267–2270 (1993).
- Miyamoto H, Gandolfo F, Gomi H, Schaal S, Koike Y, Osu R, Nakano E, Wada Y, Kawato M: A Kendama learning robot based on a dynamic optimization theory. 4th IEEE International Workshop on Robot and Human Communication, pp.327–332 (1995).
- Miyamoto H, Gandolfo F, Gomi H, Schaal S, Koike Y, Osu R, Nakano E, Wada Y, Kawato M: A Kendama learning robot based on a dynamic optimization principle. 1996 International Conference on Neural Information Processing, pp.938–942 (1996).

解説

- 宮本弘之, 五味裕章, 川人光男: タスクレベルのロボット学習. システム / 制御 / 情報, Vol.39, No.9, pp.419–426, 1995.
- 宮本弘之, 大須理英子, 中野恵理, 川人光男,Francesca Gandolfo, 五味裕章,Stefan Schaal, 小池康晴, 和田安弘: けん玉ロボットの開発. 発明, Vol.92 No.6, pp.44–49, 1995.
- 宮本弘之: 頭を使うけん玉ロボット. ATR ジャーナル, Vol.92, No.6, 1995.
- 宮本弘之, 川人光男: 最近のロボット教示方法 =見まねによる作業レベル学習ロボットの開発例=. ファクトリー・オートメーション, 7, 1996.
- 宮本弘之: ロボットへの教示方法. 計測と制御, 第 36 巻, 第 9 号, pp.627–630, 1997.

発表

- Kawato M, Gandolfo F, Gomi H, Wada Y, Miyamoto H: Teaching by showing for task level robot learning through movement pattern perception. In Proceedings of the International Conference on Neural Information Processing,Seoul, Korea, pp.17–20 October 1994.
- 川人, 宮本, 磯部, 鈴木: 作業座標における手の繰返し学習制御 – 広義ニュートン法のアルゴリズム. 信学技報, MBE85-91, pp.83–92, 1986.
- 宮本弘之, 川人光男, 鈴木良次: 中枢神経系モデルに基づく産業用マニピュレータの階層学習制御. 信学技報, MBE86-81, pp.17–24, 1987.
- 宮本弘之, 福島邦彦: 階層神経回路モデルによる時系列パターンの認識. 信学技報, NC89-83, pp.129–134, 1990.
- 宮本弘之, 福島邦彦: 聴覚末梢系の局所的特徴抽出機構の自己組織化モデル. 信学技報, NC90-139, pp.163–168, 1991.
- 宮本弘之, 五味裕章, 川人光男: 最適化原理に基づく見まねによるけん玉学習ロボット. 信学技報, NC94-71, pp.89–96, 1995.
- 宮本弘之,Francesca Gandolfo, 五味裕章,Stefan Schaal, 小池康晴, 大須理英子, 中野恵理, 川人光男: 最適化原理に基づく見まねによるけん玉学習. 信学技報, NC94-143, pp.223–230, 1995.

- 宮本弘之, 川人光男: みまねによる作業レベルの運動学習. 第1回 JSME ロボメカ・シンポジア講演論文集, pp.150–155, 1996.
- 宮本弘之: 頭を使うけん玉ロボット (見まねによる運動学習). 生理学技術研究会報告, 第19号, 1997.
- 琴坂信哉, 宮本弘之: 見まねによる運動学習. SICE 九州フォーラム'97, 1997.
- 宮本弘之: ロボットへの教示方法 – 見まねによる学習 –. 日本ファジー学会ファジー制御研究会特別講演会「とことん Intelligent」, 1998.