

②

ロボットによる
押し作業に関する研究

1995年11月

栗栖 正充

謝辞

この研究を進めていく上で終始懇切丁寧な御指導と貴重な御助言を賜わり、多くの有益な議論を賜った京都大学工学部吉川恒夫教授に深い感謝の意を表します。

数々の貴重な議論と懇切な御指導を賜った京都大学工学部横小路泰義助教授に深い感謝の意を表します。また、多くの有益な御意見を下さった京都大学工学部杉江俊治助教授に深い感謝の意を表します。

この研究を進めるにあたって、多くの有益な議論と御協力を頂いた京都大学工学部山本穰助手、井村順一助手に深い謝意を表します。

実験装置の制作に関して貴重な御助言を頂いた筑波大学電子情報工学系油田信一教授に心から感謝いたします。また、多くの意見、協力を頂いた筑波大学電子情報工学系知能ロボット研究室の学生各位に感謝の意を表します。

最後に、貴重な討論をして頂き、数々の有益な意見を述べて頂いた京都大学工学部機械工学教室吉川研究室の学生各位、特にこの研究に御助力頂いた川井昌之氏、水野光政氏に感謝の意を表します。

目次

第1章 序論	1
1.1 ロボットによる押し作業と移動マニピュレータ	1
1.2 押し作業に関する従来の研究	3
1.3 移動マニピュレータに関する従来の研究	7
1.4 従来の研究の問題点	9
1.5 本論文の目的と内容	11
第2章 対象物押し操作に基づく摩擦力分布の推定法	15
2.1 緒言	15
2.2 対象物にはたらく力とモーメント	16
2.3 摩擦中心	19
2.4 摩擦力分布，摩擦中心位置の推定法	21
2.4.1 問題の定式化とその解	21
2.4.2 仮想支持点の設定	23
2.4.3 押し操作のアルゴリズム	25
2.5 数値例	26
2.5.1 仮想支持点の個数	26
2.5.2 対象物の形状	27
2.6 実験	31
2.6.1 実験装置	31

2.6.2	実験結果	33
2.7	結言	34
第3章	押し作業における対象物の軌道計画	35
3.1	緒言	35
3.2	問題設定	36
3.2.1	対象物と仮定	37
3.2.2	軌道計画問題	38
3.3	状態方程式の導出	39
3.3.1	運動学方程式	39
3.3.2	制約条件	41
3.3.3	系の状態方程式	43
3.4	最適制御問題と数値解	44
3.4.1	問題の定式化	44
3.4.2	最適制御入力の計算	44
3.4.3	数値例	45
3.5	障害物回避	47
3.5.1	状態量の制約	47
3.5.2	数値解法	48
3.5.3	数値例	48
3.6	結言	49
付録		51
3.A.1	クリッピング法を適用した Fletcher-Reeves 法	51
3.A.2	ペナルティ法	53
第4章	対象物の押し作業の最適計画	55
4.1	緒言	55
4.2	問題設定	56
4.2.1	対象物と仮定	56

4.2.2	押し作業の計画問題	57
4.3	押し操作による対象物の運動と手先の速度方向の制限	57
4.4	押し操作前後の対象物座標の関係	59
4.4.1	対象物が円弧経路をとる場合	60
4.4.2	対象物が直線経路をとる場合	60
4.5	押し作業計画問題の定式化	61
4.5.1	複数回の押し操作による初期点と目標点との関係	61
4.5.2	押し作業計画問題の定式化	62
4.6	解法例	63
4.7	数値例	65
4.8	結言	71
第5章	押し作業における対象物の軌道追従制御	73
5.1	緒言	73
5.2	押し作業における対象物の運動	74
5.3	擬似中心	77
5.4	軌道追従制御	79
5.4.1	制御則	80
5.4.2	手先速度の制限	81
5.5	シミュレーション	82
5.6	移動マニピュレータによる押し作業	85
5.6.1	移動マニピュレータの制御	85
5.6.2	移動マニピュレータの目標軌道	87
5.7	実験	87
5.7.1	実験装置	87
5.7.2	実験結果	89
5.8	結言	91

第6章 移動マニピュレータの動的軌道計画と動的制御	93
6.1 緒言	93
6.2 状態方程式	94
6.3 軌道計画	96
6.3.1 制約条件	96
6.3.2 評価関数	98
6.3.3 定式化	100
6.4 数値例	101
6.5 位置制御と力制御	106
6.5.1 位置制御	106
6.5.2 力制御	107
6.6 実験	108
6.7 結言	109
第7章 結言	113
参考文献	115
研究発表	125

第1章

序論

1.1 ロボットによる押し作業と移動マニピュレータ

高度な作業を高速かつ高精度に行うロボットの開発が進むにつれ、ロボットが様々な作業を人間に代わって行うようになってきた。このようなロボットには時として、平面上に置かれた物体を同じ面上の別の場所に移動させるという作業が要求される。これは自動化された組立作業や移動ロボットによる運搬作業に見みられることである。このような作業をロボットが行う場合、通常は対象とする物体を把持するなどして持ち上げた後、目標とする位置まで運搬し、さらにその位置に置く、という一連の動作を行わねばならない。しかしながら、物体がロボットの持つ手先効果器やロボット本体より大きい場合、または持ち上げるには重過ぎる場合、ロボットはそのような物体を扱うことができない。このような状況にもロボットを柔軟に対応させる一つの手段として、物体を押し移動させるという方法がある。物体を押し移動させるような作業は押し作業と呼ばれる。

押し作業とは、物体を押し動かすという動作を1回または複数回行うことにより、平面上に置かれた物体の位置や姿勢を目的に応じて変化させる作業である。押し作業を行う能力をロボットに与えることには様々な利点があ

る。物体を押しするために必要とする力は持ち上げるために必要とする力よりも小さいため、ロボットが作業全体に費やすエネルギーを節約することができる。移動ロボットなどの運搬作業においては、ロボットは本体よりも大きな物や重い物を扱うことが可能となる。また障害物が存在するために通り抜けることができないような場所でも、その障害物が移動可能な場合には、障害物を押し動かすことにより経路を確保することができる。組み立て作業においては、部品を手先効果器で押しながら特定の状態に追い込みつつ把持するという動作により、部品の位置や姿勢に関する情報に誤差がある場合でも素早く部品を把持することが可能となる。さらに部品を把持するという動作そのものを減少させることもできる。これらは作業時間の短縮につながる。

押し作業においては、押される物体がどのように動くか、ということが重要となる。押される物体の運動は物体の底面に生じる摩擦力に依存し、摩擦力の生じ方は物体を支える平面との接触状態に支配される。さらに接触状態は物体が動くことにより通常は変化する。したがって底面に生じる摩擦力は一定とならず、結果として同じように物体を押ししても物体の運動が同じになる保証はない。この運動の不確実性が押し作業における大きな問題となる。人間が物体を押し移動させる動作を考えると、例えば物体が予想とは異なる動きをしても、手や指先を微妙に操作し巧みに物体を操ることにより、望み通りに物体を押ししていく。これは視覚や力覚などの感覚機能から得られる情報を蓄積された知識や経験と融合することにより、物体にどのような運動が生じるかを予め予測し、どのように押せば望み通りに物体を滑らすことができるかを考えながら作業を進めているからであろう。このような物体の運動の不確実性に対して人間が行う動作をプログラムという形で記述し、ロボットに安定した押し作業を自律的に遂行させることが、押し作業に関する研究の課題である。

一方押し作業を運搬作業に適用することを考えると、移動しながら人間が行うような巧みな操り動作を実現するロボット、すなわち移動機能を有する多関節型のマニピュレータが必要となる。移動機能を有するマニピュレータ

には、その移動機構が脚型、クローラ型、車輪型と様々なものがあるが、押し作業が平坦な床面などの平面上で行うことを前提としていることや、現在工場などで最も多用されているロボットを使用するほうが実用化の面で有利であることなどを考慮すると、移動機構が車輪型であるものが押し作業には適していると考えられる。車輪型の移動ロボットに搭載されたマニピュレータは一般に移動マニピュレータと呼ばれる。なお脚型やクローラ型のものも移動マニピュレータと呼ばれることがあるが、本論文では移動マニピュレータは特に車輪型のものを指すものとする。

移動マニピュレータは作業範囲に制限を持たずより多様な作業が実現可能であるため、近年研究対象として注目を集めている。しかしながら移動マニピュレータは床に固定されたマニピュレータとは異なり、搭載するマニピュレータと台車との間の干渉による影響を考慮しなければならず、その制御はマニピュレータや移動ロボットを個別に制御する場合に比べて複雑になる。さらに多くの移動マニピュレータは冗長性を有するため、通常は与えられた作業に対して関節の自由度が余る。したがって、作業効率を高めるためには搭載したマニピュレータが常に操作性の良い姿勢をとれるように台車を移動させる必要があり、作業内容に応じた移動マニピュレータの軌道設計も重要となる。

1.2 押し作業に関する従来の研究

滑りが生じている物体にはたらく摩擦力の解析はMacMillan[1]によってなされているが、ロボットによる押し作業を提唱し、物体を押しという動作の解析を最初に行ったのはMason[2]である。以来押し作業に関して数多くの研究がなされてきた。以下では押し作業に関する従来の研究を展望する。なお押し作業において対象とする物体を以下では対象物と呼ぶ。また対象物を単に押し動かすだけの動作を押し作業と区別して押し動作と呼ぶことにする。

ロボットが押し作業を確実に遂行するためには、運動に不確実性が存在する対象物をいかに操作するかが重要となる。人間が押し作業で行う動作には知識や経験に基づいている部分があり、そのような動作をロボットに自律的に行わせるためには、人間の知識や経験に相当するものをプログラムというかたちで与えなければならない。このためには押し動作と対象物に生じる運動との関係を明確にしておく必要があり、そのような要求から押し動作と対象物の運動との関係の解析がなされている。

Mason[2]は動摩擦係数と静止摩擦係数が同じでかつ平面上において一様であるという仮定の下に、水平面上における押し動作の解析を行い、1点で押される対象物の回転方向が、摩擦中心と呼ばれる対象物底面の圧力分布の重心に位置する点と押し動作の方向および力の方向との関係によってのみ定まることを示している。同様の仮定の下に、Peshkinら[3]、Alexanderら[4]は1点で押される対象物の運動の回転中心の存在範囲に関して解析を行い、Lynch[5]は平板で対象物を押す場合について同様の解析を行っている。また、Phamら[6]は棒状の対象物を1点で押すときに対象物に生じる運動の回転中心の位置を求めるアルゴリズムを提案している。Goyalら[7]は対象物底面の圧力分布の重心にはたらく力およびモーメントと重心の速度および角速度とを関係づける閉曲面(Limit Surface)の存在を示し、Cutkoskyら[8]は対象物底面と平面が一様に接しているとき、この閉曲面が楕円体で近似できることを示している。また直方体形状の対象物を垂直な壁に沿わせて押すための押し力の条件を考察した研究[9, 10]も報告されている。なおこれらの研究では、対象物の慣性力が摩擦力に対して無視でき、対象物に加える力と摩擦力の合力は常に釣り合うという準静力学的な仮定に基づき解析を行っているが、この仮定が成立しない衝撃的な押し動作による対象物の運動を解析した研究も報告されている[11]。

押し動作による対象物の運動の解析が進むに伴い、それらの解析結果を利用することによって押し作業の実現を計る研究がなされた。これらの研究は押し作業の利用の仕方から大きく2つに分類できると考えられる。1つはロ

ボットの手先効果器などの正確な位置決めを行わずに対象物を操作するために押し作業を利用するものであり、もう1つはロボットよりも大きな対象物を扱うために押し作業を利用するものである。

多角柱形状の対象物の角を平板で直線的に押していくと、多くの場合対象物が回転しいつかは対象物の側面が平板と接する。また平板を側面に密着させて対象物を押す場合、対象物が平板に対して滑らない範囲では対象物の運動をある程度平板で制御できる。このような動作を利用し対象物を特定の状態に追い込むことにより、対象物の初期位置や初期姿勢に不確かさが存在する状況において、素早い対象物の把持動作や、把持を行わない対象物の整列、位置決め動作を実現することが前者の研究の目的である。これらの研究は対象物の状態に関するフィードバック制御系を構成せず、押し作業をオープンループで遂行させるための手先効果器や押す道具の動作を計画することを課題としている。Balorda[12, 13]は平行2指ハンドで対象物を押すとき対象物がハンドに対してどのような状態で安定するかを解析し、対象物をそのような状態へ追い込むためのハンドの動作計画法を提案している。Akellaら[14]は平板の直線的な押し動作の組み合わせのみで対象物を移動させるための作業計画法を提案している。Lynchら[15]は平板を対象物の側面に密着させながら対象物を移動させるための作業計画法を提案している。また、Brost[16]は位置や姿勢に関して不確かさが存在する対象物を平行2指ハンドで確実に把持するためのアルゴリズムを、Goldberg, Raoら[17, 18]は平行2指ハンドで挿むという動作のみで対象物を一定の方向へ向けるためのハンドの動作計画法を提案している。Akellaら[19]は一枚の平板による押し動作で対象物を一定方向に向ける手法を提案している。この他にもコンベアベルト上を流れる対象物が、複数配置された平板または曲板の間を通り抜けた後すべて同じ姿勢に揃うようにするための平板または曲板の配置問題を扱った研究がある[20, 21]。

以上の計画問題に関する研究では対象物が多角柱形状であること、対象物を扱う道具またはロボットの手先効果器が対象物よりも大きいことが仮定され

ており、平らな側面を持たない対象物や手先効果器よりも大きい対象物を扱うことは考察されていない。これに対しロボットよりも大きな対象物を扱うために押し作業を利用することを考察した研究では、点接触による押し動作で対象物を操作する手法がとられている。1点で押される対象物の運動には不確定性が存在するため、点接触による押し作業をオープンループで行うことは極めて困難である。このため対象物の状態をなんらかのセンサで検出し、その情報をロボットの制御系にフィードバックすることにより逐次押し動作を決定するようなフィードバック制御系の構成、アルゴリズムの開発に焦点を置いているのがこれらの研究の特徴と考えられる。長田ら[22]は力覚センサを用いて対象物の姿勢を測定し、姿勢変化量から押し動作の方向を決定するアルゴリズムを提案している。また視覚システムを用いたマニピュレータによる押し作業を実現するためのビジュアルフィードバック制御に関して検討を行っている研究[23, 24, 25]も報告されている。Lynchら[26]は光学的に接触点の位置を検出できるロボット指を用い、姿勢を変えずに対象物を押すための制御系を構成している。これらの研究が作業台上に固定されたマニピュレータによる押し作業を対象としているのに対して、移動ロボットを用いた押し作業による対象物の運搬作業を対象としている研究もある。大川ら[27, 28, 29]は押し棒のついた移動ロボットに箱押し作業を行わせるための移動ロボットの制御則を提案しており、星谷ら[30]は2台の移動ロボットによる協調押し作業を実現するためのルールベースに基づく制御アルゴリズムを提示している。押し作業のための制御法を考察している研究が計画問題を扱っている研究と異なるところは、対象物をどのように移動させるかということや姿勢まで移動させようとしていることであり、ロボットよりも大きな対象物を扱ったもので、対象物の目標位置と目標姿勢の両方を指定した押し作業を実現している研究はまだ報告されていない。

一方ロボットが押し作業を確実に遂行することができたとしても、適用する状況を考慮しなければ、押し作業を導入することにより逆に作業全体の効

率を低下させてしまう危険がある。このため押し作業の効率性を考慮して、既存の作業に押し作業を組み込むための計画が必要となる。寺崎ら[31, 32]は押し作業を併用することによる組立作業の効率性を定量的に評価する方法を示し、押し作業を含めた組立手順の計画法を提案している。またChenら[33]は障害物を押し移動させるという動作を取り入れた移動ロボットの経路探索アルゴリズムを提案している。

1.3 移動マニピュレータに関する従来の研究

ロボットが行う作業が多様化するにしたがい、マニピュレータの作業範囲の拡大が要求されるようになってきている。このような要求に対して、通常使用されているマニピュレータを現在実用化が最も進んでいる車輪型移動ロボットに搭載し、マニピュレータに移動させながら作業を行わせるという試みがなされ、これに伴い移動マニピュレータの研究が行われるようになった。

マニピュレータを移動ロボットに搭載することで新たに生じる問題は双方の干渉による影響である。この干渉には、台車の走行誤差がマニピュレータの手先の位置決めに影響を及ぼす静的な干渉と、マニピュレータと台車の双方の運動が相互に影響を及ぼす動的な干渉とがある。特に動的な干渉では、マニピュレータを高速で動かす場合や重い対象物を把持させて作業させる場合に、マニピュレータの手先の目標軌道への追従性を著しく低下させる。これらの干渉による影響をいかに補償するかが移動マニピュレータの研究の中心的な課題の一つであり、これに関して種々の研究が報告されている。静的な干渉による影響を補償するものとしては、見浪ら[34, 35]が手先の目標軌道を台車から見た軌道に変換することで台車が設定された軌道からずれても手先を目標軌道に追従させる制御法を提案している。動的な干渉による影響を補償するものとしては、台車の振動が手先に与える影響をマニピュレータの関節のみで補償する方法[36, 37]や、マニピュレータの運動が台車に及ぼす影響や斜面における重力の影響を台車の駆動力のみで補償する方法[38, 39]

などが提案されている。またLiuら[40]は移動マニピュレータのモデル化誤差を考慮したロバスト制御則を提案している。しかしながらこれら研究では手先の軌道追従性の向上を目的としているため、台車の軌道制御に関しては考察されていない。これに対し、Yamamotoら[41, 42]は台車の前方に参照点を設け、マニピュレータと台車との動的な干渉を考慮しつつ手先と参照点をそれぞれの目標軌道に追従させる制御法を提案している。この制御法の特徴は参照点で台車を引っ張るようなかたちで制御を行うため、台車の持つ非ホロミックな拘束を考慮する必要がなくなる点である。

マニピュレータと台車の相互間の干渉を補償することに焦点をおいた研究では、台車が走行すべき軌道は予め与えられていることが仮定されている。これに対し与えられた手先の作業軌道に対して移動マニピュレータをどのように移動させるかという軌道設計に焦点をおいた研究がなされている。一般に移動マニピュレータに搭載されるマニピュレータは与えられる作業を遂行するに十分な自由度を持つことが仮定されている。このため移動マニピュレータは冗長な自由度を持つことになり、手先の作業軌道に対して各関節の軌道は一意には定まらずいくらかの自由度が残る。この残された自由度を台車の持つ非ホロミックな拘束を考慮しつついかに決定するかが、移動マニピュレータの軌道設計における課題となる。移動マニピュレータの軌道設計では、残る自由度に対する適当な拘束条件を新たに加える方法や、マニピュレータの作業性を指標とした評価関数を定義し、評価関数の最適化を計ることによって残る自由度を決定する方法などがある。前者には移動マニピュレータの拡張ヤコビ行列を利用する手法[43, 44]や、近似的な定式化により最適レギュレータを利用する手法[45]が提案されている。後者の方法では、各時刻において評価関数の偏微分値を用いることによりオンラインで軌道を生成する局所的最適化手法[46]と作業時間全体を通して評価関数を最小とする軌道をオフラインで計画する大域的最適化手法[47, 48]とがある。また、マニピュレータの関節にやわらかさを設定し、マニピュレータで台車を引っ張るようなかたちで移動マニピュレータを走行させる方法[49]も提案されている。

以上の研究は移動マニピュレータを組立作業などに用いられるマニピュレータの拡張として捉えたものであると考えられる。これに対し移動マニピュレータを移動ロボットの拡張として捉え、運搬作業における安定性を考察するという方向もある。これは手先に負荷を把持した移動マニピュレータを転倒させずに動作させることを目的とするもので、これには移動マニピュレータの重心の位置を制御する方法[50]や、系全体に作用するモーメントを指標とした安定規範に基づき、マニピュレータの運動で安定性を確保する制御法[51]などが提案されている。

またマニピュレータや移動ロボットの拡張として移動マニピュレータを捉えた研究とは異なるものとして、移動対象物に手先を押し付けながら移動マニピュレータを追従させる制御法[52]や、ドアの開閉と通り抜け動作の計画法[53]などが提案されている。

1.4 従来の研究の問題点

押し作業に関する従来の研究を概観すると、目標とする位置と姿勢まで対象物を移動させるような運搬作業として押し作業を捉えている研究が少いことに気付く。これは押される対象物の運動に不確実性が存在するため、位置と姿勢を同時に考慮して対象物を移動させることが難しいためであると考えられる。

対象物の位置と姿勢を共に考慮した押し作業では、Akellaら[14]やLynchら[15]により平板を用いた作業計画法が提案されているが、これらの方法では平板の大きさが対象物と同等もしくはそれ以上でなければならないという仮定を伴っている。これは押し作業を組立作業の中で利用することを前提としているので、扱う対象物が組立を行うマニピュレータの手先効果器よりも小さいという仮定が通常成り立つからである。しかしながら押し作業の応用範囲をより拡大するためには、ロボット本体よりも大きな対象物を扱えるようにする必要があり、特にロボットと対象物との相対的な大きさに制限を設

けないのであれば、対象物を1点で押す場合、すなわち点接触による押し動作を考慮しなければならない。しかしながら点接触による押し作業において位置と姿勢を共に考慮している研究はまだない。

対象物を1点で押す場合は位置と姿勢を個別に操ることができないため、任意の位置と姿勢へ対象物を移動させるためには、あらかじめ到達させることができる軌道または経路を計画しておく必要がある。しかし計画を行うためには押し動作と対象物に生じる運動との関係を定式化しておく必要がある。人間が平面上に置かれた物体を指で押して移動させる際、重心の位置や質量分布に関して先験的な知識を持たない場合は、物体を押しながらその際に生じる物体の運動を観察し、重心の位置や底面に生じる摩擦力を推定していると思われる。さらに推定によって得られた情報をもとに、物体を目標とする場所へ移動させるための経路を瞬時に計画し、巧みな操りを行い計画した経路に沿って物体を移動させていると考えられる。このような作業過程は押し作業を実行する上で重要であり、ロボットによる自律的な押し作業を実現するためには、推定、計画、制御というアプローチをとることが有効な手段であると考えられるが、そのようなアプローチをとっている研究はまだない。

一方移動マニピュレータの研究では、手先の位置や姿勢に関する軌道追従制御に焦点を置いているものが多く、環境との接触力を制御しなければならないような作業を対象としている研究がほとんどない。環境との接触を伴う作業、例えば研磨作業などでは、マニピュレータの手先の位置のみならず手先の力をも制御しなければならない。このような作業を高速、高精度に行わせるためには、動特性を考慮した制御則が必要と考えられる。さらに作業効率や手先の力出しやすさ、壁などの拘束面と台車との接触などを考慮するためには、マニピュレータの作業性や駆動力などを作業時間全体を通して最適化する大域的な軌道計画が必要であると思われる。しかしながら大域的な計画法では、手先の状態より移動マニピュレータの最適な位置と姿勢を求める手法[47]や、現在の状態から次の作業を開始する状態まで移行するための移動マニピュレータの軌道を求める手法[48]が提案されているが、手先の目

標軌道より移動マニピュレータの軌道を設計するような軌道計画法はまだ提案されていない。

1.5 本論文の目的と内容

本論文はロボットによる自律的な押し作業の確立、および移動マニピュレータの動的な制御とそのための軌道計画法の確立を目的とする。押し作業ではロボットよりも大きな対象物を扱うことができるように対象物は1点で押すものとし、運搬作業への押し作業の適用を考え、移動マニピュレータによる押し作業の実現を目指す。したがって水平面上に置かれた対象物を目標とする位置と姿勢まで、マニピュレータの手先で押して移動させることを作業目的とする。本論文ではこのような押し作業を実現する方法として、推定、計画、制御という段階を踏んだアプローチをとることにする。

ロボットが点接触による押し作業を確実に遂行するためには、1点で押される対象物の運動を正確に予測する必要がある。さらにそのためには対象物の底面に生じる摩擦力を知る必要がある。そこでまず対象物の底面に生じる摩擦力のモデル化を行い、そのモデル化した摩擦力を推定する方法を提案する。本論文ではこのモデル化した摩擦力を摩擦力分布と呼び、押し作業中は摩擦力分布が変化しないという仮定を設ける。摩擦力分布を知ることによって押し動作による対象物の運動を定式化することができる。

つぎに押し作業のための対象物の軌道を摩擦力分布を用いて計画する手法を提案する。この手法により、押す点を変更せずに対象物を移動させ得る対象物の位置と姿勢に関する時間軌道を求めることができる。さらに押す点を変更して複数回の押し動作を行うことにより対象物を移動させる作業を考え、そのような押し作業を摩擦力分布を用いて計画する手法を提案する。

実環境においては摩擦力分布が変化しないという仮定が成立しないため、計画通りに押し作業を実行するためには対象物の状態に応じてフィードバックを行う制御が必要となる。そこで軌道計画法によって生成された軌道に沿っ

て対象物を押すための制御法を提案する。さらに移動マニピュレータに押し作業を行わせる方法について述べ、移動マニピュレータを用いた押し作業の実験を行う。

最後に押し作業とは異なる作業への移動マニピュレータの適用例として、環境との接触による力を制御しなければならないような作業を想定し、そのような作業を効率良く行うための加速度の次元での移動マニピュレータの軌道計画法を提案する。また手先の位置と力および台車の位置を同時に制御する動的ハイブリッド制御法を提案する。

以下、本論文の構成について述べる。第2章では対象物を数回押すことによって摩擦分布を推定する方法を提案する。まず底面と床面とが有限個の点で接触しているものとし、各接触点にはたらく摩擦力の大きさの分布を摩擦分布として定義する。つぎに対象物が運動している際の静力学的関係式を記述し、対象物に加えている力と摩擦分布との関係式を導く。対象物から見た接触点の位置は変化しないという仮定の下に、この関係式より摩擦分布を求める方法を示す。さらに対象物と床面との接触点の位置を仮想的に定める方法、ならびに推定を行うための対象物を押すアルゴリズムについて述べ、最後に推定法の有効性を数値例および実験により示す。

押し動作による対象物の運動は摩擦分布を用いることにより定式化できる。第3章、第4章では対象物が移動しても摩擦分布は変化しないものとし、さらに摩擦分布は既知とする。第3章では押す点を変更せずに対象物を移動させるための対象物の軌道を摩擦分布を用いて計画する手法を提案する。まず対象物に関する運動学的拘束より対象物の運動学方程式を導出する。さらにアームの手先が対象物に対して滑らないための条件、手先の速度ならびに速度の変化率に関する制約などを定式化し、系の状態方程式を記述する。つぎに状態方程式と制約式に手先の道のり、速度の分散などを指標とした評価関数を加えることにより、軌道計画問題を最適制御問題に帰着させる。この最適制御問題を共役勾配法を用いて数値的に解くことにより対象物の軌道を求める。さらに軌道計画法の有効性を数値例により示す。最後に障

害物回避を含めた軌道計画法へと拡張し、その有効性を数値例により示す。

押す点を変更しない押し作業では、対象物を移動させる道のりが長くなったり、目標の場所まで移動させることができない場合があると考えられる。第4章では押す点を変更し複数回の押し動作で対象物を移動させる作業を、摩擦分布を用いて計画する手法を提案する。対象物から見た押す点の位置と速度の方向を固定すると、対象物の移動経路は直線経路または円弧経路となる。まず直線経路または円弧経路を与える押し動作を基本の押し操作とし、押し操作による対象物の変化量と目標位置、目標姿勢との関係を制約式として表す。つぎに対象物を押す回数、対象物が移動する道のり、角度の総変化量を指標とした評価関数を定義し、押し作業の計画問題を、制約式を満たし評価関数を最小にする押し操作の組み合わせを求めるとして問題として定式化する。最後に定式化された問題の具体的な解法について述べ、数値例により押し作業の計画例を示す。

第5章では第3章で提案する手法により計画された軌道に沿って対象物を移動させるための軌道追従制御法を提案する。まず対象物の運動を近似的に独立二輪駆動型移動ロボットの運動と同等に扱うことができる点が存在することを示し、その点の位置を摩擦分布を用いて求める方法を述べる。この点を利用し対象物を目標軌道に追従させる制御則を導く。さらに制御則の有効性をシミュレーションにより示す。また移動マニピュレータの制御則、目標軌道の与え方について述べ、最後に移動マニピュレータによる押し作業の実験を行い、提案する手法の有効性を検証する。

第6章では移動マニピュレータを動的に制御することを目的とし、加速度の次元での移動マニピュレータの軌道計画法を提案する。まず移動マニピュレータの状態方程式ならびに手先や台車に関する制約式を導出し、これらにマニピュレータの作業性や手先の操作力の方向などを指標とした評価関数を加え、軌道計画問題を最適制御問題として定式化する。最適制御問題を数値的に解くことにより移動マニピュレータの軌道を求め、軌道計画法の有効性を示す。つぎに手先の位置と力および台車の位置を同時に制御する動的ハイ

ブリッド制御法を提案し，軌道計画法によって生成した軌道を用いて実験を行い，軌道計画法の有効性ならびに動的ハイブリッド制御法の有効性を確認する．

最後に第7章で本論文で得られた結果をまとめ，今後の課題について述べる．

第2章

対象物押し操作に基づく 摩擦力分布の推定法

2.1 緒言

Mason[2]は対象物を押す力が摩擦中心と呼ばれる対象物底面の圧力分布の重心に位置する点を通るとき，対象物が並進することを示している．さらに押す力の方向や押す道具の速度の方向と摩擦中心の位置関係により対象物の回転方向が定まることを示している．従来の点接触の押し作業における制御はこの結果を基礎にしており，摩擦中心の位置は底面の図心の位置としている[23, 24, 25, 27, 28, 29]．しかしながら対象物の質量分布が一様でない場合や，対象物底面と床面との間の摩擦係数が一様でない場合は，摩擦中心の位置は図心の位置とは異なるため，そのような場合はなんらかの方法で摩擦中心を推定しなければ良好な結果は望めなくなると思われる．

また対象物を1点で押す動作は基本的に不安定であり，対象物の位置だけでなくその姿勢をも制御するためには，押し動作による対象物の動きをある程度特定する必要がある．対象物の慣性力が底面に生じる摩擦力に比べて無視できる場合は対象物の動きは摩擦力に支配される．仮に対象物が動いても

この摩擦力が変動しないものとする、対象物の底面に生じる摩擦力は対象物に固有なものとなり、押し動作と対象物に生じる運動との関係は一意に定まることになる。このような仮定は厳密には成立しないが、凹凸や汚れの少ない平面上においては摩擦力の変動も小さいと考えられるため、対象物の底面に生じる摩擦力を知ることは、押し作業における作業計画を立てる際や実行時における制御系を構成する際に有力な情報を与えることになると考えられる。しかしながら、ロボットがオンラインで得るセンサデータから対象物に関する固有な情報を引き出し、それを押し作業に結び付けようとしている研究はない。

本章では対象物の底面に生じる摩擦力をモデル化するものとして摩擦力分布を定義し、摩擦力分布とその重心に位置する摩擦中心の位置を推定する方法を提案する。この推定は力覚センサを取り付けたアームで対象物を数回押し、対象物に生じる運動の回転中心と加えた力を測定することにより行う。まず対象物と床面が有限個の点で接触していると仮定し、各接触点にはたらく摩擦力の大きさの分布を摩擦力分布として定義する。つぎに対象物が運動をしている際の静力学的関係式を記述する。摩擦力分布はこの関係式を解くことにより得られる。さらに対象物と床面との接触点の位置を仮想的に定める方法と、推定を行うための対象物を押すアルゴリズムについて述べ、最後に推定法の有効性を数値例および実験により示す。

2.2 対象物にはたらく力とモーメント

摩擦のある水平な床面上に置かれた対象物に外力が加わったとき、その対象物の底面にはたらく力、およびモーメントの関係式を導く。まず以下の仮定を設ける。

【仮定2.1】 対象物の底面にはたらく摩擦はクーロン摩擦であり、静止摩擦係数と動摩擦係数は等しい。

【仮定2.2】 対象物と床面との接触状態は有限個の点による接触で近似できる。なおそれらの接触点を以下では支持点と呼び、その個数を n とする。

【仮定2.3】 対象物が床面上を移動しても、対象物から見た支持点の位置は変化しない。

【仮定2.4】 対象物と床面との間の摩擦係数は支持点によって異なってもよいが時間不変である。

【仮定2.5】 外力は水平に、かつ床面に十分近い位置で対象物に加えられる。

【仮定2.6】 対象物の速度は十分低速で、慣性力は摩擦力に対して無視できる。

また床面上に基準座標系 $\Sigma_U(O_U - X_U Y_U)$ を、対象物の底面に対象物座標系 $\Sigma_O(O_O - X_O Y_O)$ を設定し、以下の要素を定義する (Fig.2.1)。

${}^O p_i$: i 番目の支持点の位置

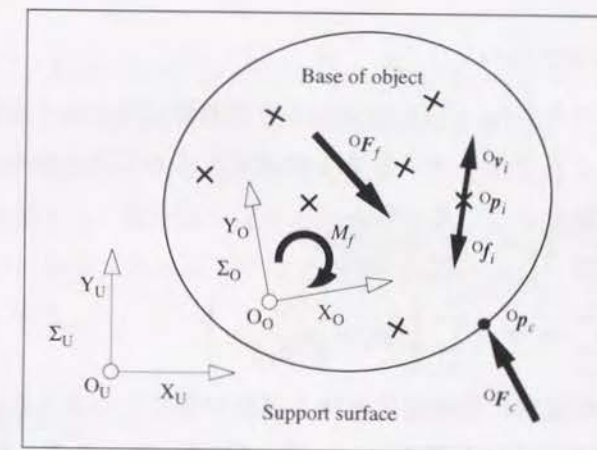


Fig. 2.1 Reference frame Σ_U and Object frame Σ_O

${}^O\mathbf{v}_i$: ${}^O\mathbf{p}_i$ における対象物の速度

a_i : ${}^O\mathbf{p}_i$ にはたらく摩擦力の大きさ

${}^O\mathbf{f}_i$: ${}^O\mathbf{p}_i$ にはたらく摩擦力

m_i : ${}^O\mathbf{f}_i$ による Σ_O の原点まわりのモーメント

${}^O\mathbf{F}_f$: 対象物にはたらく合摩擦力

M_f : 摩擦力による Σ_O の原点まわりの合モーメント

なお左上つき添字 O はその要素が Σ_O で表されていることを意味し、以後も特に断らない限り左上つき添え字はこの約束にしたがうものとする。

i 番目の支持点に働く摩擦力 ${}^O\mathbf{f}_i$, および ${}^O\mathbf{f}_i$ によるモーメント m_i はそれぞれ次式で表される。

$${}^O\mathbf{f}_i = -\frac{{}^O\mathbf{v}_i}{\|{}^O\mathbf{v}_i\|} a_i \quad (2.1)$$

$$m_i = {}^O\mathbf{p}_i \otimes \left\{ -\frac{{}^O\mathbf{v}_i}{\|{}^O\mathbf{v}_i\|} a_i \right\} \quad (2.2)$$

ここで $\|\cdot\|$ はベクトルのユークリッドノルムを表す。また \otimes は外積演算に相当する2次元ベクトルの演算を表す記号で、任意のベクトル $\mathbf{b} = [b_x, b_y]^T$, $\mathbf{c} = [c_x, c_y]^T$ に対して

$$\mathbf{b} \otimes \mathbf{c} = b_x c_y - b_y c_x \quad (2.3)$$

で定義されるものとする。式(2.1)(2.2)より対象物にはたらく合摩擦力 ${}^O\mathbf{F}_f$, および摩擦力による合モーメント M_f は次式によって表される。

$${}^O\mathbf{F}_f = \sum_{i=1}^n {}^O\mathbf{f}_i = -\sum_{i=1}^n \frac{{}^O\mathbf{v}_i}{\|{}^O\mathbf{v}_i\|} a_i \quad (2.4)$$

$$M_f = \sum_{i=1}^n m_i = -\sum_{i=1}^n \left\{ {}^O\mathbf{p}_i \otimes \frac{{}^O\mathbf{v}_i}{\|{}^O\mathbf{v}_i\|} a_i \right\} \quad (2.5)$$

Σ_U から見て対象物に姿勢変化を伴う運動が生じているとき、 Σ_O から見たその運動の回転中心の位置を ${}^O\mathbf{p}_r = [{}^Ox_r, {}^Oy_r]^T$, Σ_O の Σ_U に対する回転角を θ とすると、各支持点 ${}^O\mathbf{p}_i = [{}^Ox_i, {}^Oy_i]^T$ における対象物の速度の方向を表す単位ベクトルは次式で与えられる。

$$\frac{{}^O\mathbf{v}_i}{\|{}^O\mathbf{v}_i\|} = \text{sgn}(\dot{\theta}) D \frac{{}^O\mathbf{p}_i - {}^O\mathbf{p}_r}{\|{}^O\mathbf{p}_i - {}^O\mathbf{p}_r\|} \quad (2.6)$$

ただし

$$D = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (2.7)$$

である。

いま対象物のある1点 ${}^O\mathbf{p}_c = [{}^Ox_c, {}^Oy_c]^T$ に力 ${}^O\mathbf{F}_c = [{}^OF_{cx}, {}^OF_{cy}]^T$ を加えているとすると、仮定2.6よりつぎの関係式が成り立つ。

$${}^O\mathbf{F}_c = -{}^O\mathbf{F}_f \quad (2.8)$$

$$M_c = -M_f = {}^O\mathbf{p}_c \otimes {}^O\mathbf{F}_c \quad (2.9)$$

ここで M_c は ${}^O\mathbf{F}_c$ によるモーメントである。式(2.4)~(2.9)より

$$\text{sgn}(\dot{\theta}) {}^OF_{cx} = -\sum_{i=1}^n \frac{Y_i}{R_i} a_i \quad (2.10)$$

$$\text{sgn}(\dot{\theta}) {}^OF_{cy} = \sum_{i=1}^n \frac{X_i}{R_i} a_i \quad (2.11)$$

$$\text{sgn}(\dot{\theta}) M_c = \sum_{i=1}^n \frac{{}^Ox_i X_i + {}^Oy_i Y_i}{R_i} a_i \quad (2.12)$$

を得る。ただし $X_i = {}^Ox_i - {}^Ox_r$, $Y_i = {}^Oy_i - {}^Oy_r$, $R_i = \sqrt{X_i^2 + Y_i^2}$ である。 a_i は対象物底面の支持点 ${}^O\mathbf{p}_i$ にはたらく摩擦力の大きさなので、 (a_1, a_2, \dots, a_n) は対象物底面の摩擦力の大きさの分布を表す。この (a_1, a_2, \dots, a_n) を摩擦力分布として定義する。仮定2.1~2.6を考慮すると、この摩擦力分布は対象物に固有なものとなる。なお以下では $\mathbf{f} = [a_1, a_2, \dots, a_n]^T$ を摩擦力ベクトルと呼ぶことにする。

2.3 摩擦中心

対象物が姿勢を変えずに動くとき、各支持点の速度方向 $\frac{{}^O\mathbf{v}_i}{\|{}^O\mathbf{v}_i\|}$ は一定である。そこで

$${}^O e_v = \frac{{}^O v_i}{\|{}^O v_i\|} \quad (i = 1, 2, \dots, n) \quad (2.13)$$

とおくと、式(2.4)(2.5)は

$${}^O F_f = -{}^O e_v \sum_{i=1}^n a_i \quad (2.14)$$

$$M_f = - \left\{ \sum_{i=1}^n {}^O p_i a_i \right\} \otimes {}^O e_v \quad (2.15)$$

となる。ここで

$${}^O p_g = \frac{\sum_{i=1}^n a_i {}^O p_i}{\sum_{i=1}^n a_i} \quad (2.16)$$

なる ${}^O p_g$ を定義すると、式(2.15)は

$$M_f = {}^O p_g \otimes {}^O F_f \quad (2.17)$$

と書ける。これは対象物にはたらく合摩擦力 ${}^O F_f$ が ${}^O p_g$ に集中的にはたらくていとみなせることを意味する。この ${}^O p_g$ を摩擦中心と呼ぶ。式(2.8)(2.14)より対象物を押す力 ${}^O F_c$ は

$${}^O F_c = {}^O e_v \sum_{i=1}^n a_i \quad (2.18)$$

となるので、 ${}^O F_c$ の方向は対象物の移動方向と一致し、その作用線は ${}^O p_g$ を通っていることになる。言いかえれば、押す力 ${}^O F_c$ が ${}^O p_g$ を通るようにすれば、姿勢を変えずに対象物を押すことができる。式(2.16)から分かるように、摩擦中心は摩擦力分布の重心に位置しているため、その位置は対象物に固有となる。特に摩擦係数が一様の場合には、 $a_i = \mu n_i$ より

$${}^O p_g = \frac{\sum_{i=1}^n n_i {}^O p_i}{\sum_{i=1}^n n_i} \quad (2.19)$$

となり、Mason[2] の定義した摩擦中心と一致する。ただし n_i は i 番目の支持点にかかる垂直抗力、 μ は摩擦係数である。

2.4 摩擦力分布、摩擦中心位置の推定法

2.4.1 問題の定式化とその解

対象物をマニピュレータで数回押すことによって対象物の摩擦力分布および摩擦中心位置を推定する方法について述べる。

摩擦力分布を求めるには支持点の位置を知る必要があるが、床面上に置かれている対象物の支持点の位置を知ることは困難である。そこで本研究では視覚センサから得られる情報を用いて仮想的に支持点の位置を決定する。なお支持点の設定の仕方については後述する。支持点の位置を決定することによって、対象物に加える力およびその力によるモーメントと摩擦力分布との間に成立する線形な関係式が得られ、さらにその関係式より摩擦力分布を求めることができる。以下ではこれを示す。

まず推定に必要な情報を得るために対象物を押す動作を、以下では押し操作と呼ぶことにする。また押し操作による対象物の移動量は微小で、押し操作中は対象物に生じる運動の回転中心の位置は変わらないものとする。1回の押し操作によって測定できる値は、対象物に加えた力 ${}^O F_{cj}$ 、各押し操作前の対象物とマニピュレータとの接点の位置 ${}^O p_{cj}$ 、および回転中心の位置 ${}^O p_{rj}$ とする。ここで各値の右下の添字 j は j 回目の押し操作で得られた値を意味する。また ${}^O p_{rj}$ は視覚センサまたは何らかのセンサで測定できるものとする。このときモーメント M_{cj} は次式で得られる。

$$M_{cj} = {}^O x_{cj} {}^O F_{cyj} - {}^O y_{cj} {}^O F_{cxj} \quad (2.20)$$

いま押し操作を m 回行ったとすると、式(2.10)~(2.12)より

$$\mathbf{F} = \mathbf{G} \mathbf{f} \quad (2.21)$$

を得る。ただし

$$\mathbf{F} = [\text{sgn}(\dot{\theta}_1) {}^O F_{cx1}, \text{sgn}(\dot{\theta}_1) {}^O F_{cy1}, \text{sgn}(\dot{\theta}_1) M_{c1}, \dots, \text{sgn}(\dot{\theta}_m) {}^O F_{cxm}, \text{sgn}(\dot{\theta}_m) {}^O F_{cym}, \text{sgn}(\dot{\theta}_m) M_{cm}]^T \quad (2.22)$$

$$G = \begin{bmatrix} -\frac{Y_{11}}{R_{11}} & \cdots & -\frac{Y_{n1}}{R_{n1}} \\ \frac{X_{11}}{R_{11}} & \cdots & \frac{X_{n1}}{R_{n1}} \\ \frac{o_{x1}X_{11} + o_{y1}Y_{11}}{R_{11}} & \cdots & \frac{o_{xn}X_{n1} + o_{yn}Y_{n1}}{R_{n1}} \\ \vdots & \ddots & \vdots \\ -\frac{Y_{1m}}{R_{1m}} & \cdots & -\frac{Y_{nm}}{R_{nm}} \\ \frac{X_{1m}}{R_{1m}} & \cdots & \frac{X_{nm}}{R_{nm}} \\ \frac{o_{x1}X_{1m} + o_{y1}Y_{1m}}{R_{1m}} & \cdots & \frac{o_{xn}X_{nm} + o_{yn}Y_{nm}}{R_{nm}} \end{bmatrix} \quad (2.23)$$

であり、 f は2.2節で定義した摩擦力ベクトルである。 G の成分は支持点の位置と各回転中心の位置で構成される定数値となる。したがって F と f とは線形な関係にある。

測定値 F に対する式(2.21)の解 \hat{f} は次式で与えることができる。

$$\hat{f} = G^+ F \quad (2.24)$$

ここで G^+ は G の疑似逆行列 [54] である。特に G が正則行列となる場合は G^+ は逆行列 G^{-1} となる。 \hat{f} は式(2.21)を満たす f が一意ではない場合、式(2.21)を満たす f のうち $\|f\|$ を最小とする解を与え、式(2.21)を満たす f が存在しない場合、 $\|F - Gf\|$ を最小とする解を与える。ここでは \hat{f} を f の推定値とする。このとき摩擦中心位置の推定値 ${}^o\hat{p}_g$ は式(2.16)より

$${}^o\hat{p}_g = \frac{X^T \hat{f}}{e_n^T \hat{f}} = \frac{X^T G^+ F}{e_n^T G^+ F} \quad (2.25)$$

で得られる。ただし

$$X = \begin{bmatrix} o_{x1} & o_{x2} & \cdots & o_{xn} \\ o_{y1} & o_{y2} & \cdots & o_{yn} \end{bmatrix}^T \quad (2.26)$$

$$e_n = [1, 1, \dots, 1]^T \in \mathbf{R}^n \quad (2.27)$$

である。

2.4.2 仮想支持点の設定

対象物と床面との実際の接触状態、すなわち支持点の位置を視覚センサで知ることは難しい。そこで本研究では仮想的な支持点を対象物底面内に均等に設定することにする。なおこれらの支持点を以下仮想支持点と呼ぶ。

対象物の底面形状が既知である場合、または対象物を真上から見ることによって底面形状を把握できるのであれば、底面を格子状に分割し、各小領域の図心に仮想支持点を設定する。しかし、対象物を真上から見れない場合や移動ロボットの視覚センサのように取り付けの高さが限られている場合には、一般に底面形状を正確に把握することは困難となる。このような場合には、視覚センサから得る対象物に関する情報から底面形状を仮定し、その仮定した形状に対して仮想支持点を設定することにする。例えば、扱う対象物の底面形状が矩形であることのみが予め情報として与えられている場合には以下のように設定を行う。ただし対象物の側面は床面に対して垂直で、少なくとも1つの側面の最下部はすべて視覚センサで確認できるとする。

まず視覚センサの視野内の対象物側面の数によって2つのケースに分ける。

ケース1: 視野内の側面の数が1面の場合 (Fig.2.2(a)).

この場合は底面形状を見えている側面下端の辺(長さ l_x)を一辺とする正方形と仮定する。さらに見えている側面下端の辺が X_O 軸と一致するように対象物座標系 Σ_O を底面に設定する (Fig.2.2(b)).

ケース2: 視野内の側面の数が2面の場合 (Fig.2.3(a)).

この場合は底面形状が把握できることになる。このとき Σ_O は近い方の側面下端の辺が X_O 軸と一致するように設定する。 (Fig.2.3(b)).

つぎに仮定した底面を格子状に分割し、各小四辺形の図心に支持点を設定する。なお Fig.2.2(b), Fig.2.3(b) は支持点の数を9個とした例である。

上述の例は、対象物の底面形状が矩形であるという情報が与えられている場合であるが、任意の底面形状に対しても、視覚センサで見える部分の形状

を多角形で近似し、見えない部分の形状を正方形と仮定して、三つの領域を結合したものを底面として扱う方法などが考えられる。

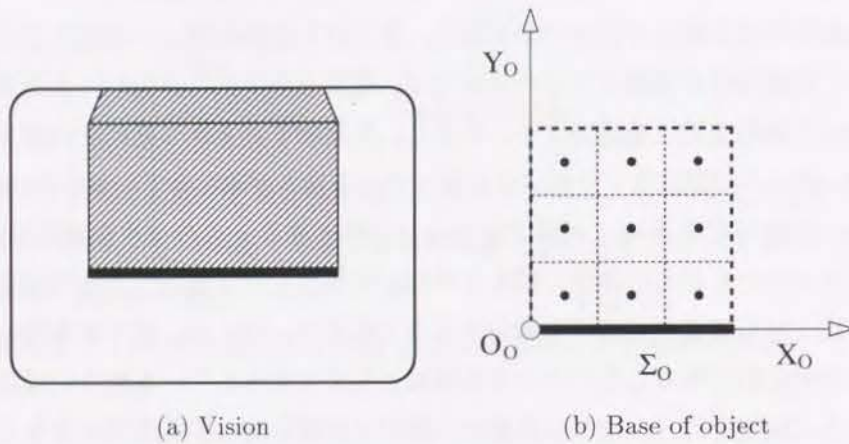


Fig. 2.2 Object and virtual support points (case 1)

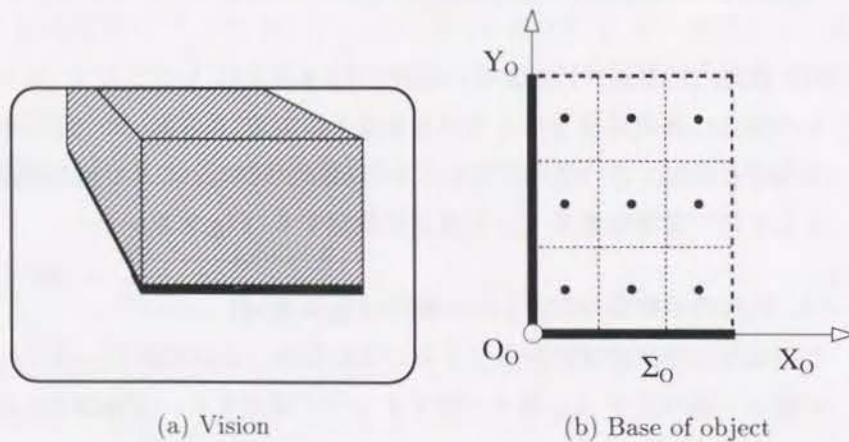


Fig. 2.3 Object and virtual support points (case 2)

2.4.3 押し操作のアルゴリズム

摩擦力分布の推定に必要な情報を得るために対象物に対して押し操作を行うわけであるが、良好な推定値を得るためには適切なアルゴリズムが必要である。以下では例として、前節で述べた矩形底面の例に対するアルゴリズムについて述べる。

アルゴリズム

押し操作はすべて X_0 軸を含む側面に対して行うものとし、押す面を以下では対象面と呼ぶ。また押し操作の際は押す方向が面の法線と一致するように行う。

<ステップ1>

対象面下端中央の点に対して押し操作を行う。このとき対象物には直進、回転のいずれかが生じる。

<ステップ2>

(1) 1回目の押し操作で対象物に回転が生じない場合は、押した位置より左右に $\frac{1}{4}l_x$ だけ離れた点に対してそれぞれ押し操作を行う。この2回の押し操作で得た測定値を用い、2.4.1節で述べた計算法で摩擦力分布と摩擦中心位置を求め、その値を推定値として終了する。

(2) 右(左)回転が生じた場合は、対象面の右(左)端から $\frac{1}{10}l_x$ 離れた点に対して押し操作を行い、それによって得た測定値と<ステップ1>で得た測定値とから摩擦力分布と摩擦中心位置を求める。

<ステップ3>

押す方向が前ステップで求めた摩擦中心を通るように対象面を押す。

<ステップ4>

(1) <ステップ3>で対象物に回転が生じた場合は、そのときに得た測定値とそれまでの押し操作で得た測定値とから摩擦力分布と摩擦中心位置を求め<ステップ3>へもどる。

(2) 対象物に回転が生じない場合は、前ステップで求めた摩擦係数分布と摩擦中心位置の値を推定値とし終了する。

ただしここでいう回転が生じないとは次の条件を満たす場合とする。

$$\text{【条件1】 } \frac{l_x}{\|O_{p_c} - O_{p_r}\|} < 0.2$$

以上のアルゴリズムに従い順次押し操作を行うことにより摩擦係数分布および摩擦中心位置を推定する。

2.5 数値例

2.5.1 仮想支持点の個数

同形状の対象物に対し、設定する仮想支持点の数 n を 4, 9, 16, 25 個と変えて推定を行い、摩擦中心位置の推定値の違いを比較する。対象物の底面形状は幅 $l_x = 0.6$ [m]、奥行き $l_y = 1$ [m] の長方形とし、摩擦係数分布を Fig.2.4 のように与えた。仮想支持点の設定はケース 1 の場合とした。すなわち底面形状を一辺が l_x の正方形と仮定して仮想支持点を設定している。推定の終了は条件 1 を満たすか、押し操作の回数が 10 回となったときとした。また結

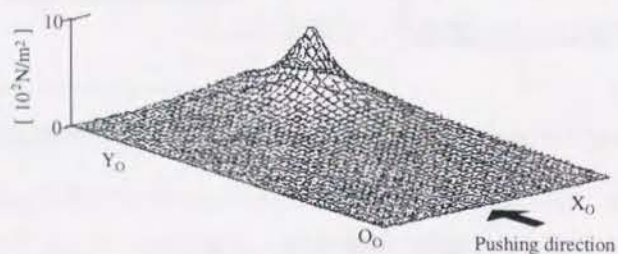


Fig. 2.4 Given friction distribution

果の評価は次式で定義したパーセント位置誤差 e_x , e_y を用いる。

$$e_x = \frac{|O_{\hat{x}_g} - O_{x_g}|}{l_x} \times 100 \quad (2.28)$$

$$e_y = \frac{|O_{\hat{y}_g} - O_{y_g}|}{l_y} \times 100 \quad (2.29)$$

ここで $O_{\hat{x}_g}$, $O_{\hat{y}_g}$ は摩擦中心位置の推定値, O_{x_g} , O_{y_g} は摩擦中心位置の真値である。

推定の結果を Fig.2.5 に示す。仮想支持点の数 n が 4 個の場合は誤差 e_x が、9 個の場合では誤差 e_y がある程度以上小さくなっていない。これに対して仮想支持点の数が 16, 25 個の場合は、どちらも 3 回目の押し操作で誤差 e_x , e_y とも小さくなっており、非常に良い推定結果を得ている。これより設定する仮想支持点の個数が少なすぎると良い推定値が得られないが、設定する個数がある程度以上だと推定値には違いがでなくなるようである。しかしながら、仮想支持点の個数を多くすると計算時間が増加することになるので、設定には適当な個数を選ぶ必要がある。

2.5.2 対象物の形状

つぎに形状の異なる対象物 3 種類に対して推定を行う。対象物の底面の寸法を Table 2.1 に、各対象物に与えた摩擦係数分布を Fig.2.6 に示す。仮想支持点の数は 25 個とした。推定の終了条件は 2.5.1 節と同様とした。

Table 2.1 Parameters of objects

	l_x [m] \times l_y [m]
Object A	1 \times 1
Object B	0.6 \times 1
Object C	1 \times 0.6

推定に要した押し操作の回数 m および位置誤差 e_x , e_y を Table 2.2 に

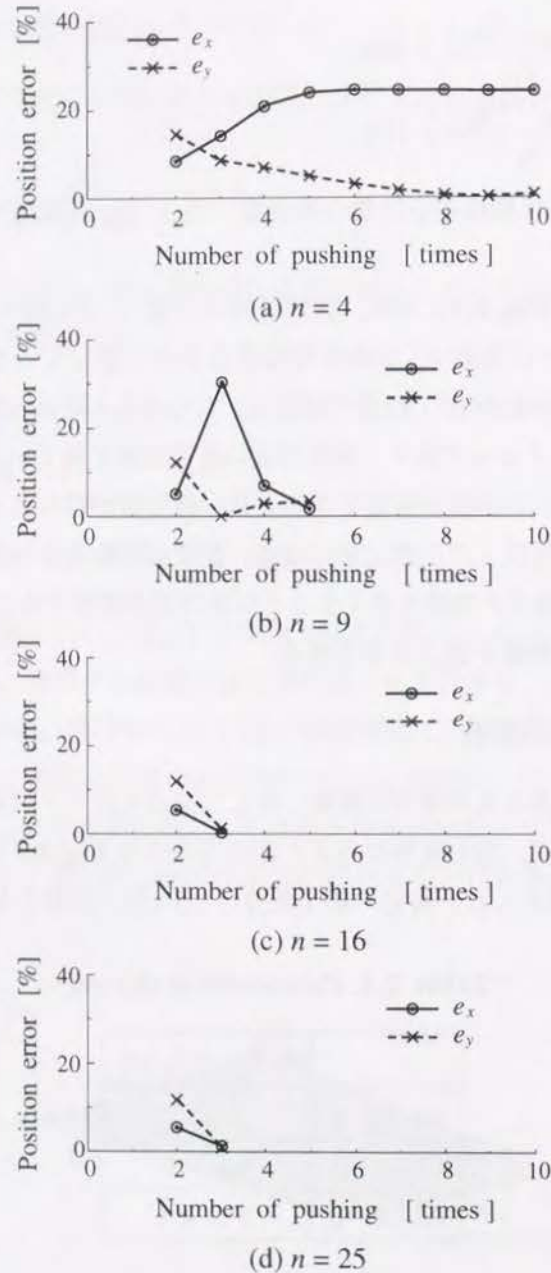


Fig. 2.5 Numerical examples

示す。また各対象物の推定した摩擦力分布を Fig.2.7~Fig.2.9 にそれぞれ示す。ただし、図では推定した摩擦力分布を与えた摩擦力分布と比較するために、各摩擦力を底面を分割した小四辺形の面積で除してある。

Table 2.2 Simulation results

	Case 1		Case 2	
	m [times]	e_x, e_y [%]	m [times]	e_x, e_y [%]
Object A	3	0.23 , 0.02	3	0.23 , 0.02
Object B	4	0.00 , 0.01	3	0.05 , 0.45
Object C	3	0.27 , 0.45	3	0.13 , 0.98

推定はほぼすべて3回目の押し操作で条件1を満たして終了し、摩擦中心位置の推定値も真値に近い結果を得ている。推定した摩擦力分布は対象物A、対象物Cに関してはケース1、ケース2とも与えた摩擦力分布に似た形状になっている。また対象物Bのケース2の場合も同様である。しかしながら、対象物Bのケース1の場合は推定した摩擦力分布と与えた摩擦力分布とは全く違った形状となっている。これは底面を正方形と仮定したために仮定した底面が真の底面より小さくなり、実際には摩擦力が生じているにもかかわらず仮想支持点が設定されない領域ができたためである。しかしながら、このように隣接する支持点における摩擦力の大きさの差が極端に大きくなるような場合は、仮定する底面の奥行きを増加させることで対処することができると考えられる。

以上の結果より対象物の底面形状が矩形である場合、提案する摩擦力分布、摩擦中心位置の推定法が有効であると言える。



Fig. 2.6 Given friction distributions

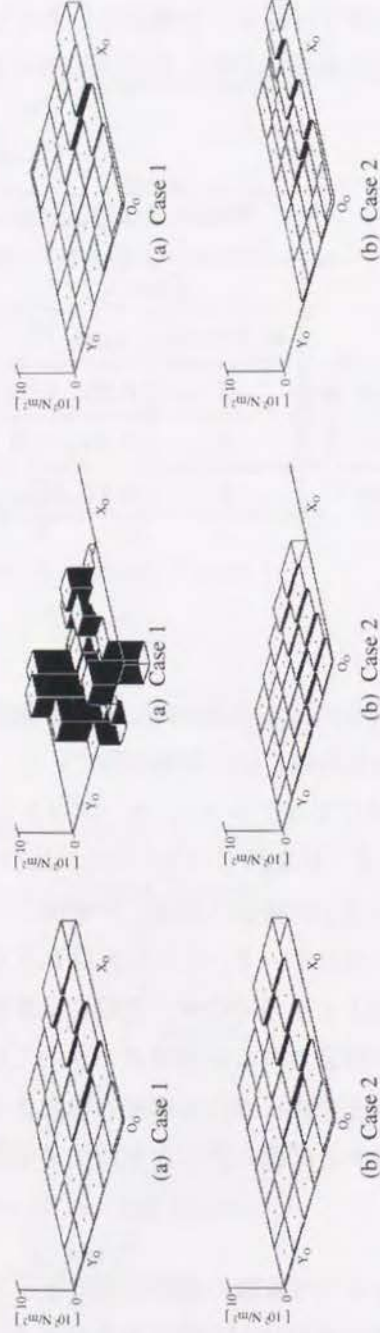


Fig. 2.7 Estimated friction distributions (object A)

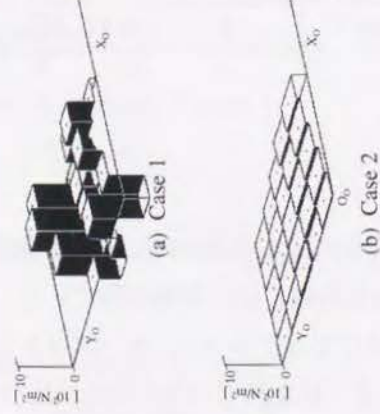


Fig. 2.8 Estimated friction distributions (object B)

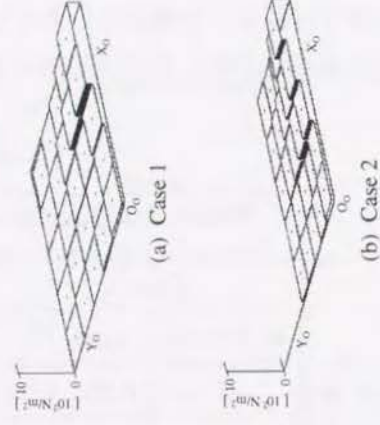


Fig. 2.9 Estimated friction distributions (object C)

2.6 実験

2.6.1 実験装置

推定法の有効性を実験により検証する。実験で用いた腕部および対象物の外観を Fig.2.10 に示す。腕は水平1自由度の直動関節を持ち、関節の根元に取り付けたレバーを手で動かすことにより動作範囲 1.5×10^{-2} [m] の伸縮運動を行わせることができる。手先には力覚センサを取り付けており、これにより対象物に加える力を測定する。実験で使用した対象物は質量 0.5[kg] のアクリル樹脂製の直方体で、1.5[kg] の分銅をはめることにより摩擦中心の位置を変えてある (Fig.2.11)。実験は方眼紙を張り付けた水平な平板の上で行った。回転中心の位置は以下の手順で求めた (Fig.2.12)。まず対象物底面の4つの角のうち2つを選び、その2点の移動前後の位置 (点 P, Q, P', Q') を方眼紙の目盛りから測定する。つぎに線分 PP', QQ' の垂直二等分線をそれぞれ求める。回転中心はこの2本の直線の交点 R で与えられる。なお本実験では回転中心の位置を手作業で求めているが、底面の2つの角の動きを視覚センサで捕らえることによりこの作業をロボットに行わせることは可能であるとえられる。

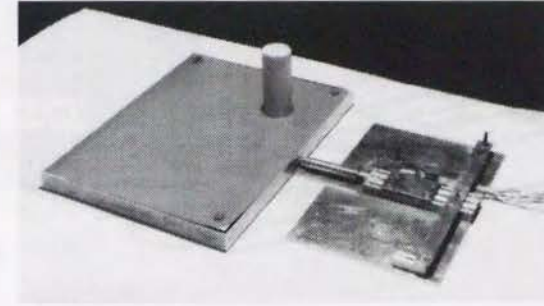


Fig. 2.10 Overview of manipulator and object used for experiment

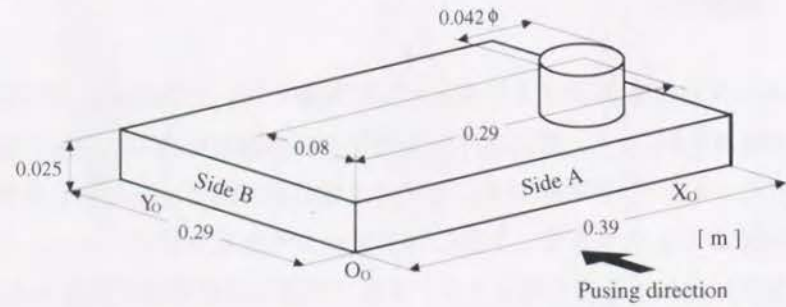


Fig. 2.11 Object used for experiment

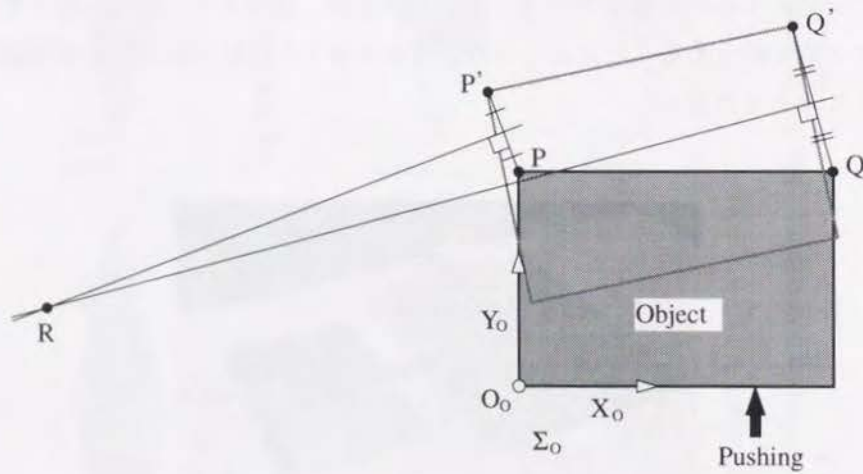


Fig. 2.12 Rotation center

2.6.2 実験結果

押し操作は対象物の側面Aに対して行い、押し操作量は 2×10^{-3} [m] とした (Fig.2.11 参照). 対象物の底面形状の仮定はケース2の場合、すなわち底面形状が把握できる場合とし、仮想支持点は25個とした. 以上の設定のもとで、2.4.3節で述べたアルゴリズムに従い実験を行ったところ、3回目の押し操作で対象物に回転が生じなくなった. すなわち2回目の押し操作後に求めた値が推定値となる. さらに対象物の側面Bを押し方向が摩擦中心を通るように押ししたところ、対象物はほとんど回転しなかった. これより摩擦中心位置の推定値は実用可能な程度の精度を持っていると言える. このときの推定した摩擦力分布を Fig.2.13 に示す. 図より分銅をはめている部分の摩擦力が最も大きくなっていることが分かる. なお周辺部分の摩擦力が大きくなっているのは、実験で使用した対象物底面が完全な平面ではなく、実際には周辺部分にも荷重が大きかったためであると思われる. したがって推定した摩擦力分布は実際の摩擦力分布に近いものと考えられる.

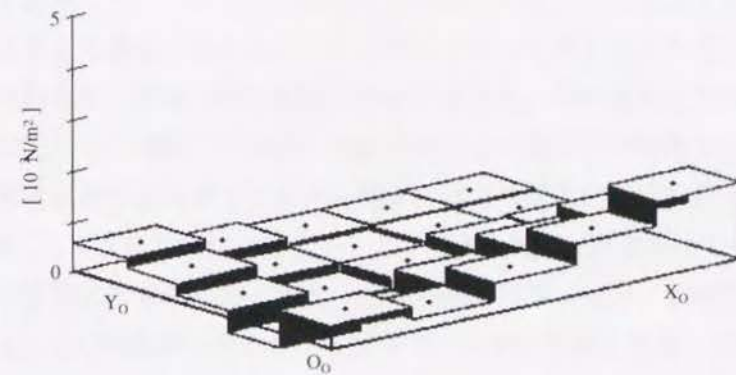


Fig. 2.13 Estimated friction distribution obtained from experiment

2.7 結言

本章では対象物の底面に生じる摩擦力をモデル化するものとして摩擦力分布を定義し、数回の押し操作で対象物の摩擦力分布および摩擦中心の位置を推定する方法を提案した。また底面の形状が矩形である対象物に対し、摩擦力分布を構成する際に必要となる底面の支持点の位置を仮想的に設定する方法について述べ、推定を行うためのアルゴリズムを示した。さらに提案する推定法が有効であることを数値例により示し、実験を行うことにより対象物の摩擦中心位置が実用可能な程度の精度で推定できることを示した。

なお Lynch[55] は後に、力覚センサを用いず類似した方法により摩擦力分布に相当するパラメータを推定する方法を提案している。この方法では押す力を測定せず、摩擦力の総和を1とすることにより、摩擦力の相対的な大きさの分布を推定している。

第3章

押し作業における 対象物の軌道計画

3.1 緒言

前章では対象物の底面に生じる摩擦力をモデル化した摩擦力分布を推定する方法を提案した。この摩擦力分布を用いることにより対象物を押した際に対象物に生じる運動を求めることが可能となる。しかしながら逆に対象物に望みの運動を生じさせる押し動作が決定できても、押す道具と対象物との間に滑りが発生し、実際には望みの運動を生じさせることができない場合がある。対象物が押す道具よりも大きい場合にはこの現象が起こることが多くなり、結果として対象物に生じさせることができる運動が限定されてしまう。したがって位置と姿勢の両方を考慮して対象物を移動させることは容易ではなく、あらかじめ対象物をどのように移動させるかを計画しておく必要がある。

対象物を任意の位置と姿勢へ押し移動させるための計画問題を扱った研究は Akellaら [14] や Lynchら [15] により行われているが、これらの研究では押す道具として平板を用い、対象物は凸多角柱形状で平板よりも小さいという前提を設けているため、平板よりも大きな対象物や平らな側面を持たない

対象物は扱うことができない。ロボットよりも大きな対象物などを扱う場合には、対象物を1点で押す方法が有効であるが、点接触による押し作業を対象とした研究では対象物を押すためのアルゴリズムや制御系の構成を議論したもののみで、対象物の位置と姿勢の両方を考慮するための計画問題を扱ったものはない。

本章では対象物上の1点を押すことによって対象物を目標とする位置と姿勢まで移動させる作業を考え、そのための対象物の軌道を摩擦力分布を用いて計画する一手法を提案する。提案する手法では点接触で対象物を押す方法を用いるため、対象物の大きさや形状に制限はない。まず対象物に関する運動学的拘束を摩擦力分布を用いて記述し、対象物の運動学方程式を導出する。さらに手先と対象物とに滑りが生じないための条件、手先の速度や速度の変化率などに関する制約を状態量、制御量に関する制約式として定式化し、系の状態方程式を記述する。これらの制約式と状態方程式に、手先の道のり、速度の分散などを指標とした評価関数を付加し、軌道計画問題を制約条件付きの最適制御問題に帰着させる。対象物の軌道はこの最適制御問題を数値的に解くことにより得られる。さらに障害物が存在する場合において、障害物と対象物との非接触条件を状態量の制約として表現し、障害物回避を含めた対象物の軌道計画へと拡張する。また数値例により提案する手法の有効性を示す。

3.2 問題設定

本章で扱う軌道計画とは、Fig.3.1のように初期点（初期位置，初期姿勢）で静止している対象物を目標点（目標位置，目標姿勢）で静止させるように1本のアームで押して移動させる作業を考えたとき、対象物を押す位置（以下作用点と呼ぶ）において対象物と手先との間に滑りが生じないようにしながら目標点まで対象物を移動させるための、対象物の位置と姿勢に関する時間軌道を計画することである。

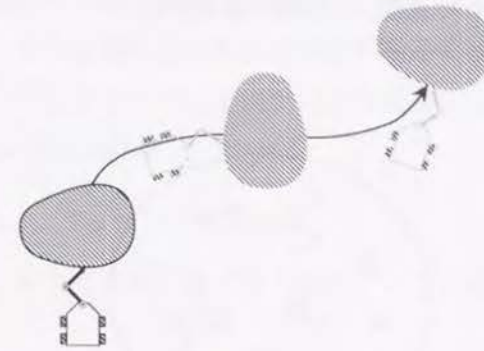


Fig. 3.1 Pushing Operation

3.2.1 対象物と仮定

押し作業のための軌道計画を行うにあたり、まず前もって与えられる対象物に関する情報や仮定について述べておく。

対象物に関して与えられている情報は、形状、摩擦力分布および対象物を押すアームの手先と対象物との間の摩擦係数とする。またこのとき仮定2.1~2.6 (2.2節参照) は成立しているものとする。仮定2.3, 2.4より摩擦力分布は対象物が移動しても変化しないことになる。さらに以下の仮定を設ける。

【仮定3.1】 対象物とアームの手先は摩擦あり点接触しているものとし、手先は対象物にモーメントは加えられないものとする。

ここで摩擦中心に原点を持ち対象物に固定された座標系 Σ_O から見た各要素を以下のように定義する (Fig.3.2)。

${}^O p_i$: i 番目 ($i = 1, \dots, n$) の支持点の位置

${}^O v_i$: ${}^O p_i$ における対象物の速度

a_i : ${}^O p_i$ にはたらく摩擦力の大きさ

${}^O p_c$: 対象物と手先との接点

${}^O v_c$: ${}^O p_c$ における対象物の速度
 μ : ${}^O p_c$ における対象物, 手先間の摩擦係数

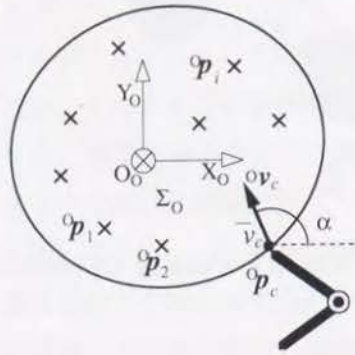


Fig. 3.2 Object frame Σ_O

摩擦中心は摩擦力分布の重心に位置することより以下の式が成立する.

$$\frac{\sum_{i=1}^n a_i {}^O p_i}{\sum_{i=1}^n a_i} = \mathbf{o} \quad (3.1)$$

未知対象物を扱う場合, 対象物が大きく把持して持ち上げることができなければその動特性を知ることはできない. しかし摩擦力分布は視覚センサなどで対象物の形状を把握すれば, 2章で提案した方法により推定することは可能である. したがって本章では摩擦力分布は既知とするが, 対象物の慣性力は仮定1.6が成立しているものとして考慮しない.

3.2.2 軌道計画問題

仮定2.6より対象物の運動は運動学的拘束によって記述されるので, 対象物を制御する量は作用点における手先の速度 ${}^O v_c$ となる. しかし手先の速度

が不連続であったりその変化率が大きいと, 仮定2.6が成立しないばかりでなく実際に押し操作を実現することが困難となる. よって計画の段階では, 手先の速度の変化率も考慮するために加速度の次元で考える. ここで後の式展開を簡単にするためにまず ${}^O v_c$ を以下のように速さ $\bar{v}_c (= \| {}^O v_c \|)$ と Σ_O の x 軸とのなす角 α とに分ける.

$${}^O v_c = \bar{v}_c {}^O n_\alpha, \quad {}^O n_\alpha = [\cos \alpha, \sin \alpha]^T \quad (3.2)$$

さらに \bar{v}_c および α の変化率を表す量を以下のように定義する.

$$a_{\bar{v}} = \dot{\bar{v}}_c \quad (3.3)$$

$$a_\alpha = \dot{\alpha} \quad (3.4)$$

以上により対象物の軌道計画問題は, 作用点の位置が与えられたとき, 対象物を初期点から目標点まで移動させるための手先の速さの変化率 $a_{\bar{v}}(t)$ と手先の速度方向の変化率 $a_\alpha(t)$ とを3.2.1節で述べた条件のもとで求めるという問題と捉えることができる.

3.3 状態方程式の導出

3.3.1 運動学方程式

まず系の状態方程式を導く準備として対象物の運動学方程式を導出する. 床面に固定された基準座標系を Σ_U とする. Σ_U から見た Σ_O の原点の位置および姿勢角を ${}^U p_o, \theta$, 作用点の位置を ${}^U p_c$ とすると次式が得られる.

$${}^U p_o = {}^U p_c - {}^U R_O {}^O p_c \quad (3.5)$$

ただし ${}^U R_O$ は

$${}^U R_O = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (3.6)$$

で表される Σ_U と Σ_O との間の回転行列である. 式(3.5)を時間微分すると次式を得る.

$${}^U\dot{\mathbf{p}}_o = {}^U\dot{\mathbf{p}}_c - \dot{\theta} {}^U\mathbf{R}_O D {}^O\mathbf{p}_c \quad (3.7)$$

ここで

$$D = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (3.8)$$

である。さらに対象物と手先との間には滑りが生じない場合を考えるので

$${}^U\dot{\mathbf{p}}_c = {}^U\mathbf{R}_O {}^O\mathbf{v}_c \quad (3.9)$$

となる。したがって式(3.2)(3.7)(3.9)より以下の運動学方程式を得る。

$${}^U\dot{\mathbf{p}}_o = \bar{v}_c {}^U\mathbf{R}_O {}^O\mathbf{n}_\alpha - \dot{\theta} {}^U\mathbf{R}_O D {}^O\mathbf{p}_c \quad (3.10)$$

ただし $\dot{\theta}$ は \bar{v}_c と α の関数になっており、その関数形は摩擦係数分布によって定まる。そこでつぎにこの関数について述べる。

仮定2.6および仮定3.1より、摩擦係数による ${}^O\mathbf{p}_c$ まわりのモーメント M_{fc} は0となるので、以下の関係式を得る。

$$\begin{aligned} M_{fc} &= \sum_{i=1}^n \left[-\left\{ D({}^O\mathbf{p}_i - {}^O\mathbf{p}_c) \right\}^T \frac{{}^O\mathbf{v}_i}{\|{}^O\mathbf{v}_i\|} a_i \right] \\ &= 0 \end{aligned} \quad (3.11)$$

Σ_U から見た i 番目の支持点の位置を ${}^U\mathbf{p}_i$ とすると

$${}^O\mathbf{v}_i = {}^U\mathbf{R}_O^T {}^U\dot{\mathbf{p}}_i \quad (3.12)$$

である。さらに式(3.10)と同様の展開により次式を得る。

$${}^U\dot{\mathbf{p}}_i = \bar{v}_c {}^U\mathbf{R}_O {}^O\mathbf{n}_\alpha + \dot{\theta} {}^U\mathbf{R}_O D ({}^O\mathbf{p}_i - {}^O\mathbf{p}_c) \quad (3.13)$$

ここで式(3.12)(3.13)を式(3.11)に代入して整理するとつぎの関係式を得る。

$$\begin{aligned} \sum_{i=1}^n \frac{A_i \omega + B_i}{\sqrt{A_i \omega^2 + 2B_i \omega + 1}} a_i &= 0 \\ \omega &= \frac{\dot{\theta}}{\bar{v}_c} \\ A_i &= ({}^O\mathbf{p}_i - {}^O\mathbf{p}_c)^T ({}^O\mathbf{p}_i - {}^O\mathbf{p}_c) \\ B_i &= \left\{ D({}^O\mathbf{p}_i - {}^O\mathbf{p}_c) \right\}^T \mathbf{n}_\alpha \end{aligned} \quad (3.14)$$

押し動作では対象物の慣性力を考慮しないのであれば、作用点における速度の大きさを固定した場合その速度方向により対象物の回転角速度は一意に定まる。すなわち $\frac{\dot{\theta}}{\bar{v}_c}$ は α に対して一意に定まる。したがって式(3.14)より $\dot{\theta}$ は \bar{v}_c と α の関数として以下のように表現できる。

$$\dot{\theta} = \bar{v}_c \omega(\alpha) \quad (3.15)$$

上式の $\omega(\alpha)$ が摩擦係数分布に支配されることは式(3.14)より容易に理解できよう。ただし関数 $\omega(\alpha)$ は α に関する陽関数の形で表現することができないため、実際には α に対する $\omega(\alpha)$ の値は式(3.14)を用いて数値的に求めなければならない。

以上をまとめると対象物の運動学方程式として次式を得る。

$$\begin{aligned} {}^U\dot{\mathbf{p}}_o &= \bar{v}_c \{ {}^U\mathbf{R}_O {}^O\mathbf{n}_\alpha - \omega(\alpha) {}^U\mathbf{R}_O D {}^O\mathbf{p}_c \} \\ \dot{\theta} &= \bar{v}_c \omega(\alpha) \\ \dot{\bar{v}}_c &= a_{\bar{v}} \\ \dot{\alpha} &= a_\alpha \end{aligned} \quad (3.16)$$

なお点接触の押し動作では式(3.15)のような速度に関する拘束式が存在し、式(3.16)を解析的に積分して対象物の位置と姿勢を時間に関する関数として表現することはできない。すなわち点接触の押し動作による対象物の運動は非ホロミックな拘束を持つことになる。

3.3.2 制約条件

運動学方程式の導出過程では作用点において対象物と手先との間には滑りが生じないとしていたが、この条件が常に満たされるようにするには手先の速度方向 α に制約を設けなければならない。また、押し動作で対象物を移動させるということ考えると、速度の大きさ \bar{v}_c などにも制約を設けなければならない。以下では状態量となる \bar{v}_c , α , 制御量となる $a_{\bar{v}}$ に関する制約について述べる。

まず \bar{v}_c と $a_{\bar{v}}$ に関しては, 対象物を押し移動させること, さらに対象物の移動速度およびその変化率があまり大きくならほうが望ましいことを考慮して以下の制約条件を設ける.

$$0 \leq \bar{v}_c \leq \bar{v}_{max} \quad (3.17)$$

$$|a_{\bar{v}}| \leq a_{\bar{v}max} \quad (3.18)$$

ここで \bar{v}_{max} および $a_{\bar{v}max}$ はそれぞれ速さとその変化率の最大値である.

つぎに α に関してであるが, 作用点で対象物と手先間に滑りが生じないためには, 手先が対象物に加える力 ${}^O\mathbf{F}_c$ が作用点における摩擦円錐内になければならない. Σ_O から見た作用点における対象物の内向き単位法線ベクトルを ${}^O\mathbf{n}_c$ とすると, ${}^O\mathbf{F}_c$ が摩擦円錐内にある条件は以下の式で表される.

$${}^O\mathbf{n}_c^T \frac{{}^O\mathbf{F}_c}{\|{}^O\mathbf{F}_c\|} \geq \frac{1}{\sqrt{1+\mu^2}} \quad (3.19)$$

ただし

$${}^O\mathbf{F}_c = \sum_{i=1}^n \frac{{}^O\mathbf{v}_i}{\|{}^O\mathbf{v}_i\|} a_i \quad (3.20)$$

である. 対象物の慣性力による影響は無視できるとしているので, 押し動作において対象物と手先との間に滑りが生じないときは, 手先が対象物に加える力と手先の速度方向は一対一に対応し, 摩擦円錐の境界に対して α の境界 α_{min} , α_{max} が定まることになる. 式(3.12)(3.13)を式(3.19)に代入し式(3.14)と連立させて解くと, 摩擦円錐に対応する α の範囲として次式を得る.

$$\alpha_{min} \leq \alpha \leq \alpha_{max} \quad (3.21)$$

上式が α に関する制約条件となる. なお式(3.21)で表される速度方向の範囲は摩擦円錐に対して速度円錐と呼ばれる [2].

3.3.3 系の状態方程式

まず以下の量を定義する.

$$\beta = \frac{\alpha_{max} + \alpha_{min}}{2} \quad (3.22)$$

ここで $\boldsymbol{\xi} = [\xi_1, \dots, \xi_5]^T = [{}^U\mathbf{p}_o^T, \theta, \bar{v}_c, \gamma (= \alpha - \beta)]^T$ を状態変数, $\mathbf{u} = [u_1, u_2]^T = [a_{\bar{v}}, \dot{\gamma} (= a_{\alpha})]^T$ を制御入力とすると式(3.16)(3.22)より系の状態方程式は以下ようになる.

$$\dot{\boldsymbol{\xi}} = \mathbf{f}(\boldsymbol{\xi}, \mathbf{u}) = [f_1(\boldsymbol{\xi}, \mathbf{u}), \dots, f_5(\boldsymbol{\xi}, \mathbf{u})]^T \quad (3.23)$$

ただし

$$\begin{aligned} f_1(\boldsymbol{\xi}, \mathbf{u}) &= \xi_4 \{ \cos(\xi_3 + \xi_5 + \beta) \\ &\quad + ({}^Ox_c \sin \xi_3 + {}^Oy_c \cos \xi_3) \omega(\xi_5 + \beta) \} \\ f_2(\boldsymbol{\xi}, \mathbf{u}) &= \xi_4 \{ \sin(\xi_3 + \xi_5 + \beta) \\ &\quad + (-{}^Ox_c \cos \xi_3 + {}^Oy_c \sin \xi_3) \omega(\xi_5 + \beta) \} \\ f_3(\boldsymbol{\xi}, \mathbf{u}) &= \xi_4 \omega(\xi_5 + \beta) \\ f_4(\boldsymbol{\xi}, \mathbf{u}) &= u_1 \\ f_5(\boldsymbol{\xi}, \mathbf{u}) &= u_2 \end{aligned} \quad (3.24)$$

である. また状態変数の制約式は

$$g_{sj}(\mathbf{u}, t) \geq 0 \quad (j = 1, \dots, 4) \quad (3.25)$$

$$\begin{aligned} g_{s1}(\mathbf{u}, t) &= \xi_4 \\ g_{s2}(\mathbf{u}, t) &= -\xi_4 + \bar{v}_{max} \\ g_{s3}(\mathbf{u}, t) &= \xi_5 + \gamma_{max} \\ g_{s4}(\mathbf{u}, t) &= -\xi_5 + \gamma_{max} \end{aligned}$$

となり, 制御入力の制約式は

$$|u_1| \leq a_{\bar{v}max} \quad (3.26)$$

となる. ここで γ_{max} は状態変数 ξ_5 の最大値であり次式で定義される.

$$\gamma_{max} = \frac{\alpha_{max} - \alpha_{min}}{2} \quad (3.27)$$

3.4 最適制御問題と数値解

3.4.1 問題の定式化

軌道計画問題を定式化する. 時間 $0 \leq t \leq t_f$ の間に対象物の状態を初期状態 $\xi_0 = [\xi_{01}, \dots, \xi_{05}]^T$ から目標状態 $\xi_f = [\xi_{f1}, \dots, \xi_{f5}]^T$ まで移すことを考えるとき, 作業の効率性などの点からアームの手先の道のりや速度の分散はできるだけ小さい方が望ましい. そこでまずそれらを評価する関数として次式を定義する.

$$L(\xi, \mathbf{u}) = w_l \xi_4 + w_v \xi_4^2 + w_\theta \xi_5^2 \quad (3.28)$$

つぎに終端時刻における状態を評価する関数として次式を定義する.

$$\Phi(\xi(t_f)) = \sum_{i=1}^5 w_{p_i} \{\xi_i(t_f) - \xi_{f_i}\}^2 \quad (3.29)$$

ただし w_{p_i} ($i = 1, \dots, 5$), w_l , w_v , w_θ は重み係数である. 式(3.28)(3.29)より軌道計画における評価関数として以下の式を用いる.

$$J(\mathbf{u}) = \Phi(\xi(t_f)) + \int_0^{t_f} L(\xi, \mathbf{u}) dt \quad (3.30)$$

これにより対象物の軌道計画問題は, 状態方程式が式(3.23)で与えられる系について, 初期条件 $\xi(0) = \xi_0$, および式(3.25)(3.26)で与えられる制約条件を満たし, 式(3.30)を最小にする制御入力 $\mathbf{u}(t)$ を求めるという最適制御問題に帰着する.

3.4.2 最適制御入力の計算

系の状態方程式が非線形である場合, 最適制御入力を解析的に求めるのは非常に難しい. そこで本研究では最適制御入力を共役勾配法の一つである Fletcher-Reeves 法 [56] を用いて計算機により数値的に求める. その際, 制御入力の制約に関してはクリッピング法 [57] を適用する (付録 A1 参照). また状態変数の制約に関しては亀山らによって開発されたペナルティ法 [58] を

用いる (付録 A2 参照). ここで用いたペナルティ法の特徴は初期制御入力を任意に選べることにある. なお初期制御入力は $\mathbf{u}(t) = \mathbf{0}$ とした.

以上により最適制御問題の解として最適制御入力 $\mathbf{u}(t)$ を求め, これにより軌道計画の解として対象物の軌道 $U_{p_o}(t)$, $\theta(t)$ を得る.

3.4.3 数値例

提案する軌道計画法の有効性を数値例により検証する. 数値例では重さ 1 [kg], 0.2×0.3 [m] の矩形底面を持つ対象物を想定した. Fig.3.3 に対象物の摩擦係数分布を示す. 床面の支持点の個数は 25 個とした. 初期状態は $\xi_0 = [0, 0, 0, 0, 0]^T$ ([m],[m],[rad],[m/s],[rad/s]) とし, 目標状態は $\xi_f = [0, 1, -\pi, 0, 0]^T$ とした. また作用点は ${}^O p_c = [0, -0.106]^T$ [m] とし, $\mu = 0.2$ とした. 以上の条件のもとに $t_f = 10$ [sec], 時間刻み幅 100 [msec], $\bar{v}_{max} = 0.3$ [m/sec], $\bar{a}_{v_{max}} = 0.3$ [m/sec²], w_{p_i} ($i = 1, 2, 3, 4, 5$) = 10000, $w_l = w_v = w_\theta = 1$ として計算を行った. この結果を Fig.3.4 に示す. 図において, 対象物の中に描かれている 3 本の実線のうちの両端の 2 本の線分によって挟まれる領域は手先速度方向の許容範囲 (速度円錐) を表し, その中の線分が手先速度方向を示す. また, 初期点から目標点まで続く点線は対象物の摩擦中心の軌跡を表す. これより対象物が目標点へ到達する軌道が得られていることがわかる.

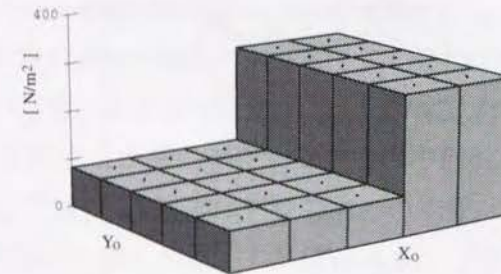


Fig. 3.3 Friction distribution

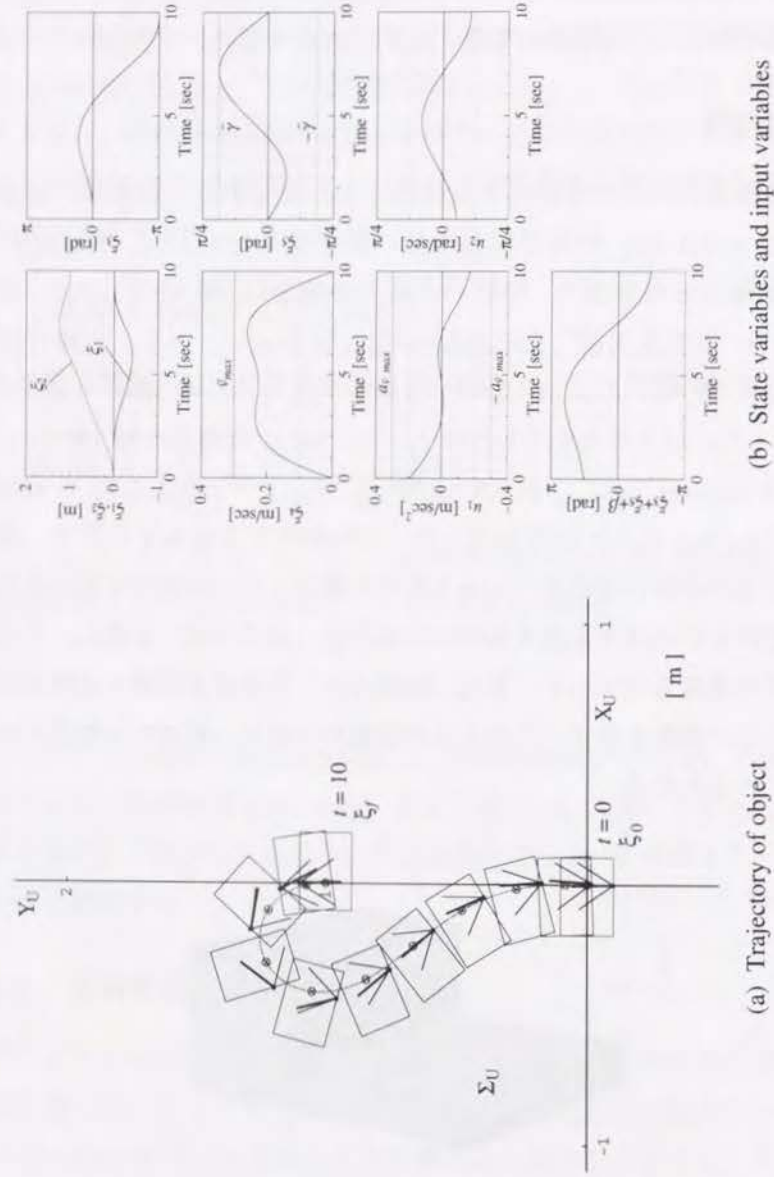


Fig. 3.4 Numerical example (A)

また形状、摩擦中心の位置がともに同じで摩擦力分布が周縁部に偏っている対象物、および形状が同じで摩擦中心の位置が対象物座標から見て x 軸の負の方向に 0.01 [m] ずれた対象物を想定し、上述の例と同様の条件で計算を行った。前者の対象物では Fig.3.4 に示す軌道よりもわずかに大きな円弧を描いて目標点に到達する軌道が得られた。これは摩擦力分布が周縁に偏っているため、対象物に生じる回転運動の曲率半径が大きくなるからである。また後者の対象物では、時計回りに回転するときの円弧軌道の曲率半径がさらに大きくなる軌道が得られた。これは摩擦中心の位置が摩擦円錐の左側の境界に近くなっているため、手先が滑らない範囲では対象物を時計回りに回転させるための回転半径を小さくできないためである。これらの結果は直感的に予想する結果と非常に良く一致していると言える。

3.5 障害物回避

本節では初期点と目標点との間に障害物が存在する場合の軌道計画法について述べる。

3.5.1 状態量の制約

初期点と目標点との間に障害物が存在すると、対象物の軌道、すなわち状態量に制約が加わることになる。以下では移動させようとする対象物が障害物に接触しないために必要な状態量の制約を定式化する。ただしアームなしそれを搭載する車自体との接触については、ここでは考慮しないこととする。また障害物の個数、形状および位置は与えられているものとする。

いま初期点と目標点との間に存在する障害物の個数を l とする。また ${}^U\mathbf{p}_{bq} = [{}^Ux_{bq}, {}^Uy_{bq}]^T$ を q 番目の障害物の関心の位置、 r_{bq} を ${}^U\mathbf{p}_{bq}$ を中心とした障害物を含む円の最小半径とする (Fig.3.5)。このとき対象物が q 番目の障害物と接触しないための条件は以下の式で与えられる。

$$\|{}^U\mathbf{p}_o - {}^U\mathbf{p}_{bq}\|^2 \geq (r_{bq} + r_o + r_w)^2 \tag{3.31}$$

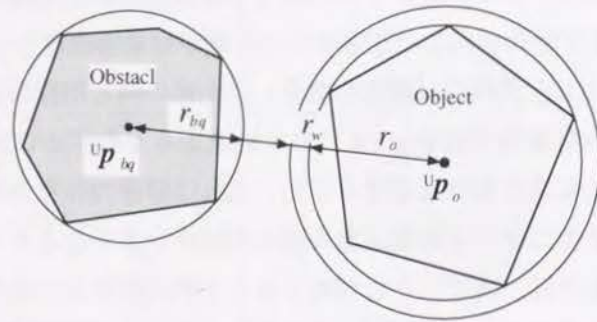


Fig. 3.5 Radius of object and obstacle

ただし r_o は摩擦中心を中心とした対象物を含む円の最小半径であり, r_w は安全余裕である. これにより状態量の制約式として次式を得る.

$$g_{bq}(\boldsymbol{\xi}) \geq 0 \quad (q = 1, \dots, l) \quad (3.32)$$

$$g_{bq}(\boldsymbol{\xi}) = (\xi_1 - Ux_{bq})^2 + (\xi_2 - Uy_{bq})^2 - (r_{bq} + r_o + r_w)^2$$

3.5.2 数値解法

障害物回避を含めた対象物の軌道計画問題は, 3.4.1節で述べた最適制御問題に式(3.32)を付加することによって定式化される. この定式化された最適制御問題はペナルティ法において式(3.32)に対するペナルティ関数を作ることにより3.4.2節と同様にして数値的に解くことができる.

3.5.3 数値例

障害物回避を含めた軌道計画法の有効性を数値例により検証する. 障害物は $U\mathbf{p}_{b1} = [-0.25, 0.5]^T$, $r_{b1} = 0.15$, $U\mathbf{p}_{b2} = [0.2, 1.2]^T$, $r_{b2} = 0.15$ とし, 初期状態を $\boldsymbol{\xi}_0 = [0, 0, 0, 0, 0]^T$, 目標状態を $\boldsymbol{\xi}_f = [0, 1.8, 0, 0, 0]^T$ とした. また $r_w = \frac{1}{10}r_o$ とした. その他の値は3.4.3節で挙げた数値例と同じ値を用い

た. 以上の条件のもとに対象物を移動させる軌道を計算させた. この結果を Fig.3.6 に示す. これより対象物が障害物に接触せずに目標点へ到達する軌道が得られていることが分かる.

3.6 結言

対象物上の1点を押すことによって対象物を目標とする位置と姿勢まで移動させるという押し作業において, 押す点を変更せずに移動させるための対象物の位置と姿勢に関する時間軌道を摩擦力分布を用いて計画する一手法を提案した. まず摩擦力分布を用いて対象物の運動学方程式を導出した. つぎに手先の速度や速度の方向, 速度の変化率などに関する制約式を定式化し, 系の状態方程式を記述した. これらの制約式と状態方程式に, 手先の道り, 速度の分散などを評価する関数を付加し, 軌道計画問題を最適制御問題として定式化した. さらに定式化された問題に対する数値的解法について述べ, 提案する手法の有効性を数値例により示した. また障害物回避を含めた軌道計画法への拡張を示し, その有効性を数値例により示した.

本章では作用点の選び方については考慮していないが, 提案した手法により得られる対象物の軌道は与える作用点の位置に依存するため, 道りが最短となるような軌道を得るためには, さらに作用点の位置に関して最適化を計るアルゴリズムが必要である.

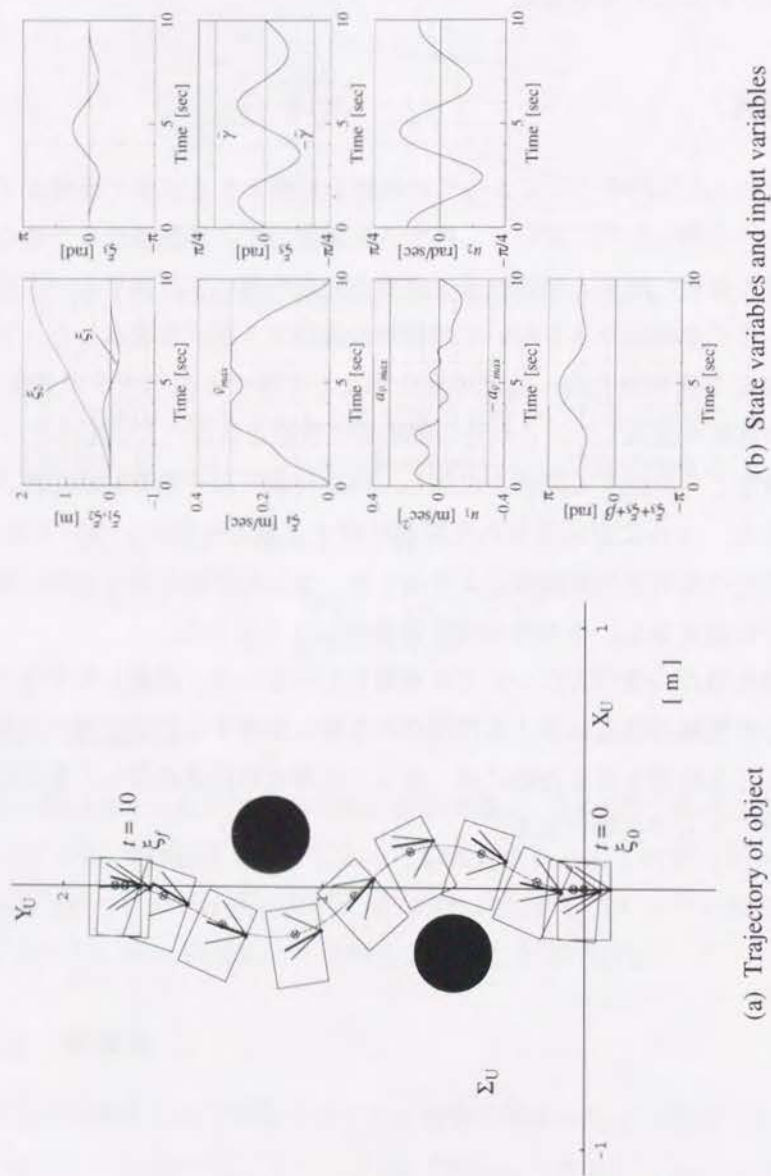


Fig. 3.6 Numerical example (B)

付録

3.A.1 クリッピング法を適用した Fletcher-Reeves 法

制御入力に制約がある最適制御問題はクリッピング法を用いることにより数値的に解くことができる。以下ではクリッピング法を適用した Fletcher-Reeves 法による最適制御計算アルゴリズム [57] において入力を2次元とした場合について述べる。

状態方程式が式 (3.23) で与えられる系について，初期条件

$$\xi(0) = \xi_0 \quad (3.A1.1)$$

および式 (3.26) で与えられる制約条件を満たし，式 (3.30) を最小にする制御入力 $\mathbf{u}(t)$ を求める。

乗数ベクトル $\lambda(t) = [\lambda_1(t), \dots, \lambda_5(t)]^T$ を導入して

$$H = -L(\xi, \mathbf{u}) + \lambda^T f(\xi, \mathbf{u}) \quad (3.A1.2)$$

と置くと，オイラーの方程式，横断性条件から以下の式を得る。

$$\dot{\lambda}(t) = -\frac{\partial H}{\partial \xi} \quad (3.A1.3)$$

$$\lambda(t_f) = -\frac{\partial \Phi}{\partial \xi(t_f)} \quad (3.A1.4)$$

[アルゴリズム]

- i) 飽和のない，すなわち $|u_1| \leq a_{v \max}$ である初期入力 $\mathbf{u}_0(t)$ ($0 \leq t \leq t_f$) を選ぶ。
- ii) 式 (3.23)(3.A1.1) を用いて， $\xi_0(t) = \xi(t; \mathbf{u}_0)$ を得る。 $\mathbf{u}_0(t)$ ， $\xi_0(t)$ を用いて，式 (3.A1.3)(3.A1.4) を $t = t_f$ から $t = 0$ まで逆時間に解き， $\lambda_0(t) = \lambda(t; \mathbf{u}_0)$ を得る。これより勾配 $J_{\mathbf{u}_0}$ を計算し，単位勾配方向ベ

クトルを $\mathbf{s}_0 = -\mathbf{J}_{\mathbf{u}0} / \|\mathbf{J}_{\mathbf{u}0}\|$ とする. ここで $\mathbf{J}_{\mathbf{u}i} = [J_{u_1i}, J_{u_2i}]^T$, $\|\mathbf{J}_{\mathbf{u}i}\|$ は次式で与えられる.

$$\mathbf{J}_{\mathbf{u}i} = -\frac{\partial H}{\partial \mathbf{u}_i} \quad (3.A1.5)$$

$$\|\mathbf{J}_{\mathbf{u}i}\| = \sqrt{\int_0^{t_f} \{J_{u_1i}^2(t) + J_{u_2i}^2(t)\} dt} \quad (3.A1.6)$$

$i=0$ と置いてステップ iv) へいく.

- iii) $|u_{1i}| > a_{\bar{v}max}$ かつ $u_{1i} J_{u_1i} < 0$ となる時間区間を飽和区間 S_i とし, 残りの時間区間を U_i とする. さらに $i \geq 1$ に対する単位勾配方向ベクトル \mathbf{s}_i を次式により定める.

$$\mathbf{s}_i(t) = -\frac{\mathbf{J}_{\mathbf{u}i}}{\|\mathbf{J}_{\mathbf{u}i}\|} + \zeta_i \begin{bmatrix} \delta(t) & 0 \\ 0 & 1 \end{bmatrix} \mathbf{s}_{i-1}(t) \quad (3.A1.7)$$

$$\delta(t) = \begin{cases} 1 & t \in U_i \\ 0 & t \in S_i \end{cases}$$

ただし ζ_i は

$$\zeta_i = \begin{cases} \zeta^o & \zeta^o < \zeta^m \text{ のとき} \\ \rho \zeta^m & \zeta^o \geq \zeta^m \text{ のとき} \\ & (\rho \text{ は定数で } 0 < \rho < 1) \\ 0 & I_2 = 0 \text{ または } I_3 = 0 \end{cases} \quad (3.A1.8)$$

で与えられ, さらに

$$\begin{aligned} \zeta^o &= I_1/I_2, \quad (I_2 \neq 0), \quad \zeta^m = I_1/I_3, \quad (I_3 \neq 0) \\ I_1 &= \int_{U_i} \mathbf{J}_{\mathbf{u}i}^T \mathbf{J}_{\mathbf{u}i} dt, \quad I_2 = \int_{U_i} \mathbf{J}_{\mathbf{u}i-1}^T \mathbf{J}_{\mathbf{u}i-1} dt, \\ I_3 &= -\int_{U_i} \mathbf{J}_{\mathbf{u}i}^T \mathbf{J}_{\mathbf{u}i-1} dt \end{aligned}$$

である.

- iv) まず $\mathbf{u}_{i+1}(t; \kappa) = [u_{1(i+1)}(t; \kappa), u_{2(i+1)}(t; \kappa)]^T$ を以下の式で与える.

$$u_{1(i+1)}(t; \kappa) = \text{sat}(u_{1i}(t) + \kappa s_{1i}(t)) \quad (3.A1.9)$$

$$u_{2(i+1)}(t; \kappa) = u_{2i}(t) + \kappa s_{2i}(t) \quad (3.A1.10)$$

ここに関数 $\text{sat}(x)$ は

$$\text{sat}(x) = \begin{cases} a_{\bar{v}max} & x \geq a_{\bar{v}max} \\ x & |x| \leq a_{\bar{v}max} \\ -a_{\bar{v}max} & x \leq -a_{\bar{v}max} \end{cases} \quad (3.A1.11)$$

である. $J(\mathbf{u}_{i+1}(t; \kappa))$ を最小にする κ を求め, これを κ_i とし, $\mathbf{u}_{i+1}(t) = \mathbf{u}_{i+1}(t; \kappa_i)$ と置く.

- v) $\xi_{i+1}(t) = \xi(t; \mathbf{u}_{i+1})$, $\lambda_{i+1}(t) = \lambda(t; \mathbf{u}_{i+1})$ を ii) と同様に計算して, $J(\mathbf{u}_{i+1})$, $\mathbf{J}_{\mathbf{u}i+1}$ を計算する.
- vi) $|J(\mathbf{u}_{i+1}) - J(\mathbf{u}_i)|$ が ϵ に十分近ければ停止. そうでなければ $i = i+1$ としてステップ iii) へいく.

3.A.2 ペナルティ法

状態変数の制約を含む最適制御問題はペナルティ法 [58] によって数値的に解くことができる. 以下にこのペナルティ法について簡単に紹介する.

まず状態変数の制約式 (3.25) に対して以下のペナルティ関数を定義する.

$$\pi_j(g_{sj}, \sigma_k) = \begin{cases} \left[g_{sj}(\xi) - \frac{\sigma_k}{2} h_j(t) \right]^2 + 3 \left[\frac{\sigma_k h_j(t)}{2} \right]^2, & g_{sj}(\xi) \leq 0 \\ \frac{[\sigma_k h_j(t)]^3}{g_{sj}(\xi) + \sigma_k h_j(t)}, & g_{sj}(\xi) \geq 0 \end{cases} \quad (j = 1, \dots, 4) \quad (3.A2.1)$$

ここで添字 k は反復計算において k 回目の反復を意味する. また $\sigma_0 > \sigma_1 > \dots$ で $\lim_{k \rightarrow \infty} \sigma_k = 0$, $h_j(t)$ はラグランジュ乗数に対応する関数であり $[0, t_f]$ において正値をとるものとする. ただし本章では $h_j(t) \equiv 1$, $\sigma_0 = 0.1$, $\sigma_{k+1} = \frac{1}{10}\sigma_k$ とした. このペナルティ関数を用いて新しい評価関数を

$$J_{1k} = J + \frac{1}{\sigma_k} \sum_{j=1}^4 \int_0^{t_f} \pi_j(g_{sj}, \sigma_k) dt \quad (3.A2.2)$$

とする. 計算法は, まず $k=0$ としてクリッピング法を適用した Fletcher-Reeves 法により最適制御入力 u_0 を求める. つぎにこの u_0 を初期値として $k=1$ に対する最適制御入力 u_1 を求める. この繰り返しを $|u_{k+1} - u_k|$ が十分小さくなり, かつ状態量の制約が満たされるようになるまで続ける. ただし各 k に対する最適制御入力 u_k を求める際に必要となる乗数ベクトル λ の計算には以下の式を用いる.

$$\dot{\lambda}(t) = -\frac{\partial H}{\partial \xi} - \sum_{j=1}^4 \begin{cases} \left[h_j(t) - \frac{2}{\sigma_k} g_{sj}(\xi) \right] \frac{\partial g_{sj}(\xi)}{\partial \xi}, & g_{sj}(\xi) \leq 0 \\ \frac{\sigma_k^2 h_j(t)^3}{[g_{sj}(\xi) + \sigma_k h_j(t)]^2} \frac{\partial g_{sj}(\xi)}{\partial \xi}, & g_{sj}(\xi) \geq 0 \end{cases} \quad (3.A2.3)$$

第4章

対象物の押し作業の最適計画

4.1 緒言

前章で提案した手法により, 押す点を変更せずに対象物を目標点まで移動させることができる軌道を求めることが可能となる. この軌道に対する押し作業では押す点を変更する動作がなく連続的な動作で対象物を移動できるので, 作業が俊敏に遂行できるという利点がある. しかしながら押す点を変更しないということは対象物に生じさせることができる運動の範囲を大きく制限することに等しい. したがって目標点によっては得られる軌道の道のりが非常に長くなり, 作業全体の効率が低下する場合がある. また作業空間の制限などにより目標点に到達させることができる軌道が存在しない場合も考えられる. このような状況に対しては, 押す点を変更し単純な動作の組み合わせで対象物を順次移動させていくという押し作業が有効であると考えられる. 単純な押し動作の組み合わせによる押し作業は, 1つの動作ごとに確認をとりながら作業を進めることができ, かつ各動作が実現しやすいという利点がある. 複数回の単純な押し動作によって押し作業を実現するための計画問題

を扱ったものには、Akellaら[14]が提案する平板を用いた押し作業の作業計画法がある。これは平板を直線的に移動させる動作を複数回行うことによって対象物を任意の位置と姿勢へ移動させることを実現させるための、平板の動作方向と移動量を計画するものであるが、平板が対象物よりも大きくなければならないという欠点がある。

本章では対象物上の1点を押し対象物を目標点へ移動させる作業に対し、押す点を変更し複数回の押し動作で作業を実現させることを考え、そのための押し作業を摩擦力分布を用いて計画する手法を提案する。提案する手法は以下のようなものである。まず対象物からみた手先と対象物の接点の位置および速度の方向を一定とすることにより、対象物の移動経路を直線経路および円弧経路のみに限定する。つぎに接点を変更することで対象物がとる直線または円弧の経路を1つまたは複数組み合わせることにより対象物を目標点へ移動させるための経路の候補を与える。それらのうち、対象物を押す回数、対象物が移動する道のり、角度の総変化量を指標とした評価関数を最小にする経路を最適経路とし、最適経路を与える押し動作の組み合わせを最適な押し作業とする。

4.2 問題設定

4.2.1 対象物と仮定

3章と同様に対象物に関して与えられている情報は、形状、摩擦力分布および対象物を押すアームの手先と対象物との間の摩擦係数とする。また仮定2.1~2.6(2.2節参照)および仮定3.1(3.2.1参照)が成立しているものとする。したがって摩擦力分布は対象物が移動しても変化しないものとし、対象物の運動に関しては準静力学的な仮定の下に議論を行う。さらに本章では作業領域内に障害物はないものとする。

4.2.2 押し作業の計画問題

単純な押し動作の組み合わせで押し作業を実現することに対し、本章では作業の計画問題を解くにあたり以下のような手法をとる。

対象物と手先が摩擦あり点接触をしているとき、手先が対象物に加える力が手先と対象物との間の最大静止摩擦力を越えると手先は対象物に対して滑り出す。各押し動作の間は手先が滑らないように押すことを考える。すなわち手先は対象物に最大静止摩擦力を越える力は加えないようにし、対象物を押す点(以下では作用点と呼ぶ)を変更するときは、手先を対象物より一度離してから変更する。また1回の押し動作中は対象物からみた作用点の速度方向が一定となるように押すこととする。本章ではこのように対象物を押す動作を押し操作と呼ぶことにする。押し操作による対象物の運動は、作用点の速度方向が対象物の摩擦中心を通るときには姿勢変化を伴わない直進運動となり、それ以外の場合は1つの回転中心まわりをまわる回転運動となる。しかしながら通常1回の直進運動または回転運動だけでは対象物を目標の位置および姿勢に移動させることができない。そこで作用点を変更して数回の押し操作を行い、その経路を組み合わせることで目標点(目標位置、目標姿勢)へ移動させることを考える。また対象物が移動する道のりや対象物の姿勢角の総変化量は小さい方がよく、押す回数も少ない方が望ましい。そこでそれらを評価する関数を定義し、経路を決定する際に用いる。

以上より押し作業の計画問題は、対象物を目標点まで移動させるための円弧経路と直線経路の組み合わせを、対象物の移動する道のり、姿勢角の変化、および押す回数をできるだけ少なくするように選択するという問題となる。

4.3 押し操作による対象物の運動と手先の速度方向の制限

本章における計画問題では、押し操作による対象物の運動の回転中心の位置と作用点の速度の方向との関係が重要となる。また、手先が対象物に対し

て滑らないようにしなければならないので、作用点の速度の方向と作用点にはたらく力との関係も重要となる。以下ではこれらの関係について述べる。

まず対象物に固定された座標系を Σ_O とし、 Σ_O から見た作用点の位置を ${}^O p_c$ 、作用点における対象物の速度を ${}^O v_c$ 、押し操作による対象物の運動の回転中心の位置を ${}^O p_r$ とする (Fig.4.1)。

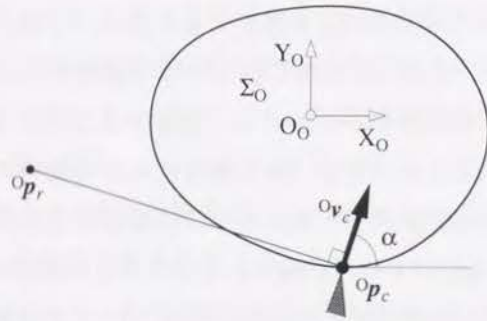


Fig. 4.1 Object frame Σ_O

作用点における対象物の速度の方向が摩擦中心に向いているとき対象物は直進運動を行うが、それ以外の場合は回転運動を行う。対象物が回転運動を行っているときの対象物の角速度を $\dot{\theta}$ とすると、そのときの作用点における対象物の速度は

$${}^O v_c = \dot{\theta} D ({}^O p_c - {}^O p_r) \quad (4.1)$$

と表される。ただし

$$D = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (4.2)$$

である。一方、 ${}^O v_c$ を大きさ \bar{v}_c と Σ_O の x 軸となす角度 α を用いて

$${}^O v_c = \bar{v}_c {}^O n_\alpha, \quad {}^O n_\alpha = [\cos \alpha, \sin \alpha]^T \quad (4.3)$$

と表すと、 $\dot{\theta}$ と \bar{v}_c および α との関係は次式で与えられる (3.3.1参照)。

$$\dot{\theta} = \bar{v}_c \omega(\alpha) \quad (4.4)$$

ここで $\omega(\alpha)$ は α の関数であり、その関数形は摩擦力分布によって定まる。したがって式(4.4)を用いて式(4.1)を ${}^O p_r$ について解くと次式を得る。

$${}^O p_r = {}^O p_c + D \frac{{}^O n_\alpha}{\omega(\alpha)} \quad (4.5)$$

上式より作用点の位置と速度の方向が決まれば、そのとき対象物が行う運動の回転中心の位置を求めることができる。

つぎに作用点の速度の方向と作用点にはたらく力との関係であるが、一般には速度の方向と力との方向とは異なる。3.3.2節で述べたように、対象物の慣性力による影響が無視でき、かつ作用点に対象物に対して動かないときは、作用点の速度の方向 α とそこにはたらく力 ${}^O F_c$ は一対一に対応し、 ${}^O F_c$ の方向を定めれば α が定まる。対象物と手先が滑らないためには対象物に加える力が最大静止摩擦力を越えてはならないが、そのときの摩擦円錐の境界に対して α の境界が定まり、摩擦円錐の範囲に対して α の範囲、すなわち速度円錐が定まる。したがって作用点における Σ_O から見た押し操作の速度方向は、速度円錐内になければならない。

4.4 押し操作前後の対象物座標の関係

4.2.2節で、1回の押し操作では対象物から見た作用点の位置 ${}^O p_c$ および作用点の速度方向 α を一定に保つという条件を設けた。これにより対象物は速度方向が摩擦中心に向いている場合は直進経路をとり、それ以外の場合は式(4.5)で表される点 ${}^O p_r$ を中心とする円弧経路をとる。以下では、それぞれの経路をとったときの押し操作の前後の位置と姿勢の関係を回転中心の位置 ${}^O x_r$ 、その回りの回転角度 θ 、および直進量 l を用いた同次変換行列で表す。

4.4.1 対象物が円弧経路をとる場合

移動前の対象物座標系を Σ_{O_A} , 移動後の対象物座標系を Σ_{O_B} とする. Σ_{O_A} から見た回転中心の位置を ${}^O p_{r_A} = [{}^O x_{r_A}, {}^O y_{r_A}]^T$ とし, その回りに角 θ だけ回転するとき, Σ_{O_A} から Σ_{O_B} への同次変換行列 ${}^A T_B$ は, Fig.4.2 のように, Σ_{O_A} を Σ_{O_A} に関して $-{}^O p_{r_A}$ だけ並進したのちに θ だけ回転した座標系を $\Sigma_{O_{A'}}$ とし, さらに $\Sigma_{O_{A'}}$ を $\Sigma_{O_{A'}}$ に関して ${}^O p_{r_A}$ だけ並進させるという合成変換になり, その同次変換行列は以下のように表される.

$$\begin{aligned} {}^A T_B &= \begin{bmatrix} 1 & 0 & {}^O x_{r_A} \\ 0 & 1 & {}^O y_{r_A} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -{}^O x_{r_A} \\ 0 & 1 & -{}^O y_{r_A} \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos \theta & -\sin \theta & {}^O x_{r_A}(1 - \cos \theta) + {}^O y_{r_A} \sin \theta \\ \sin \theta & \cos \theta & -{}^O x_{r_A} \sin \theta + {}^O y_{r_A}(1 - \cos \theta) \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (4.6)$$

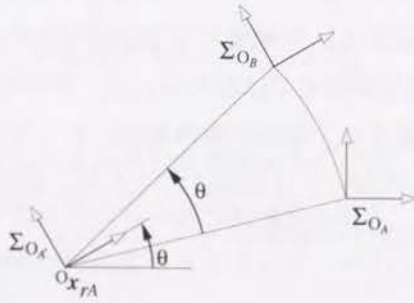


Fig. 4.2 Frames Σ_{O_A} , $\Sigma_{O_{A'}}$, and Σ_{O_B}

4.4.2 対象物が直線経路をとる場合

直進についても, 移動前の対象物座標系を Σ_{O_A} , 移動後の対象物座標系を Σ_{O_B} とし, Σ_{O_A} から見た対象物の直進量を ${}^O l_A = [{}^O l_{x_A}, {}^O l_{y_A}]^T$ とすると, Σ_{O_A} と Σ_{O_B} との関係を表す同次変換行列は次式のようになる.

$${}^A T_B = \begin{bmatrix} 1 & 0 & {}^O l_{x_A} \\ 0 & 1 & {}^O l_{y_A} \\ 0 & 0 & 1 \end{bmatrix} \quad (4.7)$$

4.5 押し作業計画問題の定式化

前節では1回の押し操作の前後の対象物の位置および姿勢の関係を同次変換行列で表した. 以下ではそれらを用いて初期点(初期位置, 初期姿勢)と目標点(目標位置, 目標姿勢)との関係を同次変換行列で表し, それより得られる制約条件を用いて押し作業の計画問題を定式化する.

4.5.1 複数回の押し操作による初期点と目標点との関係

まず対象物を押す回数(作用点の変更回数)を N 回とする. さらに初期点における対象物座標系を Σ_{O_0} とし, j 回目 ($j = 1, \dots, N$) の押し操作によって移動した後の対象物座標系を Σ_{O_j} とする. このとき $\Sigma_{O_{j-1}}$ と Σ_{O_j} との関係を表す同次変換行列を ${}^{j-1} T_j$ とすると, Σ_{O_0} と N 回目の押し操作後の対象物座標系 Σ_{O_N} との関係を表す同次変換行列 ${}^0 T_N$ は

$${}^0 T_N = {}^0 T_1 {}^1 T_2 \dots {}^{N-1} T_N \quad (4.8)$$

となる.

j 回目の押し操作が, $\Sigma_{O_{j-1}}$ から見て回転中心 ${}^O p_{r_j} = [{}^O x_{r_j}, {}^O y_{r_j}]^T$ の回りに角度 θ_j だけ回転させる押し操作であるとき, ${}^{j-1} T_j$ は次のように表せる.

$${}^{j-1} T_j = \begin{bmatrix} \cos \theta_j & -\sin \theta_j & {}^O x_{r_j}(1 - \cos \theta_j) + {}^O y_{r_j} \sin \theta_j \\ \sin \theta_j & \cos \theta_j & -{}^O x_{r_j} \sin \theta_j + {}^O y_{r_j}(1 - \cos \theta_j) \\ 0 & 0 & 1 \end{bmatrix} \quad (4.9)$$

また j 回目の押し操作が, $\Sigma_{O_{j-1}}$ から見て ${}^O l_j = [{}^O l_{x_j}, {}^O l_{y_j}]^T$ だけ直進させる押し操作であるならば, ${}^{j-1} T_j$ は次のように表せる.

$${}^{j-1} T_j = \begin{bmatrix} 1 & 0 & {}^O l_{x_j} \\ 0 & 1 & {}^O l_{y_j} \\ 0 & 0 & 1 \end{bmatrix} \quad (4.10)$$

なお j 回目の押し操作が直進であるときは $\theta_j = 0$ とする。

以上より N 回の押し操作による同次変換行列 ${}^0\mathbf{T}_N$ は、式(4.8)~(4.10) より次のようになる。

$${}^0\mathbf{T}_N = \begin{bmatrix} \cos \theta_N^0 & -\sin \theta_N^0 & {}^0x_N^0 \\ \sin \theta_N^0 & \cos \theta_N^0 & {}^0y_N^0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.11)$$

ここで θ_k^0 は

$$\theta_k^0 = \sum_{j=0}^k \theta_j \quad (4.12)$$

であり、 $\theta_0 = 0$ とする。また ${}^0x_N^0$ 、 ${}^0y_N^0$ は次式で表される。

$${}^0x_N^0 = \sum_{\text{rot}} (G_j {}^0x_{rj} - H_j {}^0y_{rj}) + \sum_{\text{trans}} {}^0l_{xj} \quad (4.13)$$

$${}^0y_N^0 = \sum_{\text{rot}} (H_j {}^0x_{rj} + G_j {}^0y_{rj}) + \sum_{\text{trans}} {}^0l_{yj} \quad (4.14)$$

ただし総和記号の rot は j 回目の押し操作が回転のものについての総和をとり、trans は直進のものについての総和をとるという意味である。また、 G_j 、 H_j 、 ${}^0l_{xj}$ 、 ${}^0l_{yj}$ はそれぞれ

$$G_j = \cos \theta_{j-1}^0 - \cos \theta_j^0 \quad (4.15)$$

$$H_j = \sin \theta_{j-1}^0 - \sin \theta_j^0 \quad (4.16)$$

$${}^0l_{xj} = ({}^0l_{xj} \cos \theta_{j-1}^0 - {}^0l_{yj} \sin \theta_{j-1}^0) \quad (4.17)$$

$${}^0l_{yj} = ({}^0l_{xj} \sin \theta_{j-1}^0 + {}^0l_{yj} \cos \theta_{j-1}^0) \quad (4.18)$$

である。

4.5.2 押し作業計画問題の定式化

Σ_{O_0} から見た対象物の目標点を ${}^0\mathbf{p}_d = [{}^0x_d, {}^0y_d]^T$ 、 θ_d とすると、対象物を N 回の押し操作で目標点まで移動させるためには

$${}^0x_N^0 = {}^0x_d \quad (4.19)$$

$${}^0y_N^0 = {}^0y_d \quad (4.20)$$

$${}^0\theta_N^0 = {}^0\theta_d \quad (4.21)$$

という制約条件を満たさねばならない。また対象物を目標点まで移動させるとき、対象物を移動させる道のりはできるだけ短い方がよく、さらに対象物の姿勢角の総変化量、作用点の変更回数も少ない方が望ましい。したがって最適な経路を決定するために用いる評価関数として次式を定義する。

$$J = \sum_{\text{rot}} |\theta_j| \sqrt{({}^0x_{rj})^2 + ({}^0y_{rj})^2} + \sum_{\text{trans}} \sqrt{({}^0l_{xj})^2 + ({}^0l_{yj})^2} \quad (4.22)$$

$$+ w_1 \sum_{j=1}^N |\theta_j| + w_2 N \quad (4.23)$$

ここで w_1 、 w_2 はそれぞれ角度の総変化量と押す回数に対する重みである。

以上より対象物の押し作業の計画問題は、式(4.19)~(4.21)の制約条件を満たし、式(4.22)で表される評価関数 J を最小化する回転中心の位置 ${}^0\mathbf{p}_{rj}$ 、回転角 θ_j 、および直進距離 0l_j の組を求めるとして定式化できる。

4.6 解法例

前節では円弧および直線の経路を組み合わせて対象物を目標点まで移動させる押し作業の計画問題の定式化を行った。以下ではその問題を解く一手法について具体的に述べる。

定式化された問題を解くためには、各押し操作における対象物の回転中心の位置 \mathbf{p}_{rj} および回転角 θ_j 、または直進距離 0l_j が必要となるが、 ${}^0\mathbf{p}_{rj}$ とそれを実現する作用点の位置 ${}^0\mathbf{p}_{cj}$ 、および ${}^0\mathbf{p}_{cj}$ における対象物の速度の方向 α_j との間には、式(4.5)で示されるような摩擦力分布に依存する複雑な関係があるため、問題の解を解析的に求めるのは非常に難しい。そこでここでは離散的にとった有限個の作用点と速度方向についてのみ回転中心の位置または直進方向との関係をあらかじめ求めておき、押し作業においてはそれらの作用点のみを用いて押し操作を行うという手法をとる。これにより対象物

の移動経路はあらかじめ定めた作用点に対する円弧経路または直線経路の組み合わせのみに制限されるが、作用点と回転中心の関係は既知なものとなる。また姿勢角についても離散化し、対象物は有限個の姿勢のみをとることができるものとする。したがって、それらの姿勢角と回転の作用点の組み合わせにより角度の制約式(4.21)を満たす組を求め、その組に対して直進の押し操作による直線経路の長さを位置に関する制約式(4.19)(4.20)を満たし、評価関数 J を最小にするように計画することで、対象物を目標点へ移動させる経路を求める。このように本手法では、対象物のとれる姿勢を離散化して経路の候補を有限個のものとするにより、制約式(4.19)~(4.21)を満たす解を求める。ただし以下では対象物の底面形状は多角形とし、作用点における速度の方向は、対象物と手先との間の摩擦係数にかかわらず手先を滑らさずに押せるように、対象物の辺に垂直に力が加わる方向の1方向のみとする。したがって直進のための作用点は摩擦中心から各辺におろした垂線の足となる。垂線の足が辺上に無ければその辺に対する直進の作用点はとれないので、その方向には対象物を直進させることができないものとする。なお対象物を回転させるための作用点を回転の作用点と呼び、直進させるための作用点を直進の作用点と呼ぶことにする。

以上の前提のもとに、まず回転の押し操作の回数の最大値 N_{Rmax} および対象物の姿勢角がとることのできる値の組を与えておく。与えられた N_{Rmax} に対して回転の押し操作の回数 N_R を $1, \dots, N_{Rmax}$ の間で変化させ、それぞれの N_R にたいして最後の回転の押し操作 (N_R 回目) で姿勢が目標値になるような姿勢角の組み合わせをすべて選ぶ。ただし初期姿勢と目標姿勢に変化がないときには、姿勢変化回数に0回も許し、 N_R を $0, \dots, N_{Rmax}$ の間で変化させて計画を行う。さらにそれぞれの姿勢角の組み合わせに対して、 j 回目 ($j = 1, \dots, N_R$) にどの作用点を押ししてその姿勢まで変化させるか、という作用点の組み合わせをすべて選び出す。つぎに各回転の押し操作の間にすべての直進の作用点に対する押し操作を考え、その中から式(4.19)および式(4.20)の制約条件を満たし直進距離の総和を最小とする直進経路の組合

せを線形計画法を用いて求める。さらに得られた経路に対する評価関数 J の値を求める。最後に姿勢角と回転の作用点の全ての組み合わせに対して以上の手順で評価関数 J の値を求め、その中で値が最小になるものを最適経路とする。

なお評価関数 J には回転角度および作用点の変更回数の評価も含まれているが、上示した解法では直進経路の組合せを求める段階では、直進距離の総和のみを評価した線形計画法を用いている。以下では直進距離の総和のみを評価して得た解が、姿勢角と回転の作用点の1つの組に対して評価関数 J を最小にすることを示す。

直進経路は回転角度には無関係であるから回転角の評価には無関係である。また1回の直進で目標位置へ移動させる経路と2回の直進により目標位置へ移動させる経路を比較すると、三角不等式から必ず1回の直進による経路が短くなる。線形計画法では直進経路の道のりを最小とする解が得られるので、得られた解は押し操作回数に関しても最小となる。したがって直進経路を線形計画法で求めた解は、与えられた姿勢角と回転の作用点の組に対して評価関数 J を最小にする。

4.7 数値例

提案する手法の有効性を数値例により示す。対象物はFig.4.3のような三角形とした。また支持点は15個とし、それぞれの点での摩擦力の大きさは全て1とした(Fig.4.4(a))。回転の作用点には各辺を二等分したそれぞれの線分の中点を選んだ(Fig.4.4(b))。対象物は三角形なので回転の作用点は6点 ($R_1 \sim R_6$) となる。直進の作用点については、摩擦中心からおろした垂線の足が頂点と重なり作用点を選べない辺が1つ存在するため、その数は2点 (T_1, T_2) となる。Table 4.1に回転の作用点の位置 ${}^O p_c$ 、およびそれに対する回転中心の位置 ${}^O p_r$ と回転方向を、Table 4.2に直進の作用点の位置 ${}^O p_c$ 、および直進方向 α をそれぞれ示す。なおTable 4.1右端の欄のCWは時計ま

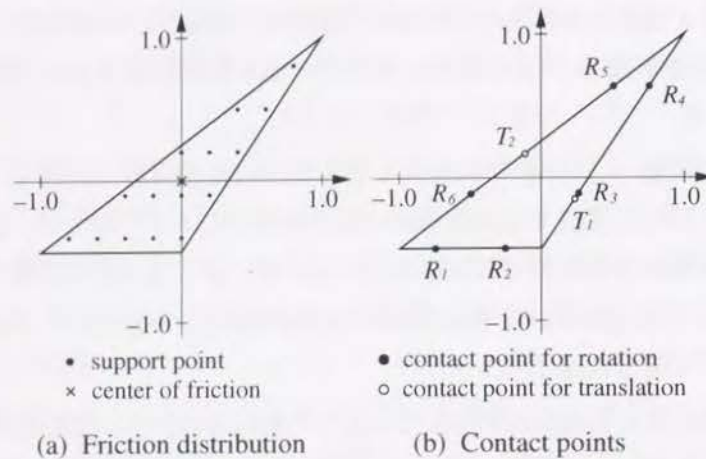


Fig. 4.3 Object

Table 4.1 Contact points for rotation

	${}^o p_c$	${}^o p_r$	
R_1	(-0.750, -0.500)	(0.379, 0.090)	CW
R_2	(-0.250, -0.500)	(0.911, 0.114)	CW
R_3	(0.250, -0.125)	(-1.005, -1.429)	CCW
R_4	(0.750, 0.625)	(-0.238, -0.317)	CCW
R_5	(0.500, 0.625)	(0.345, -0.297)	CW
R_6	(-0.500, -0.125)	(0.509, 0.427)	CCW

Table 4.2 Contact points for translation

	${}^o p_c$	α [deg]
T_1	(-0.231, -0.154)	146.33
T_2	(-0.120, 0.160)	306.88

わり, CCWは反時計まわりの回転が生じることを表す。

対象物の初期位置は $[0, 0]^T$, 初期姿勢は $0[\text{deg}]$ とし, 目標位置は $[10, 10]^T$, 目標姿勢は $80[\text{deg}]$ とした. また回転の押し操作回数の最大値 N_{Rmax} は 4 とした. 対象物がとることのできる姿勢角は, 目標姿勢角度の $1/4$ である $20[\text{deg}]$ ごととした. 以後この姿勢角の変化量の最小値を単位角と呼ぶ. 姿勢角の範囲は, 目標姿勢の反対方向への回転を含むときに評価関数がよくなる場合も考慮し, 初期姿勢から目標姿勢の反対方向へ1単位角回転したところから, 目標姿勢より1単位角進んだところまでとした. すなわち対象物がとることのできる姿勢角は $-20, \dots, 100[\text{deg}]$ の範囲で $20[\text{deg}]$ ずつの角度となる.

以上の条件のもとで, 評価関数の重み w_1, w_2 を, ケース1: $w_1 = 0, w_2 = 0$, ケース2: $w_1 = 10, w_2 = 0$, およびケース3: $w_1 = 0, w_2 = 10$ のように変えて計画を行った. ケース1は対象物の道のりを最小にすること, ケース2は道のりを小さくすることと角度の変化を抑えること, ケース3は道のりを小さくすることと作用点の変更回数を抑えることにそれぞれ対応する.

ケース1からケース3に対して得られた最適経路を Fig.4.4~Fig.4.6にそれぞれ示す. また各計画結果において選択された作用点と, それに対する対象物の姿勢変化量または直進量を Table 4.3~Table 4.5に示す. 図では, 初期点および目標点における対象物は太い実線で, 各押し操作後の対象物は細い実線で示しており, それぞれの押し操作における作用点は押し操作前の対象物上に点で示した. また摩擦中心の経路を点線で表し, 押し操作の順番を表す数字を記入してある. さらに回転の押し操作に対しては, 作用点にもっとも近い対象物頂点の動きを矢印つきの円弧で示した.

各経路における道のりは, ケース1が14.570, ケース2およびケース3が21.645となった. 3つの結果を比較すると, ケース1では距離を短くするために, はじめに目標位置まで移動させ安い角度に姿勢を変更した後に目標位置付近まで移動させ, その後位置と姿勢を調整するという経路をとっており, 押し操作の回数や姿勢角の変化量も大きくなっていることがわかる. それに

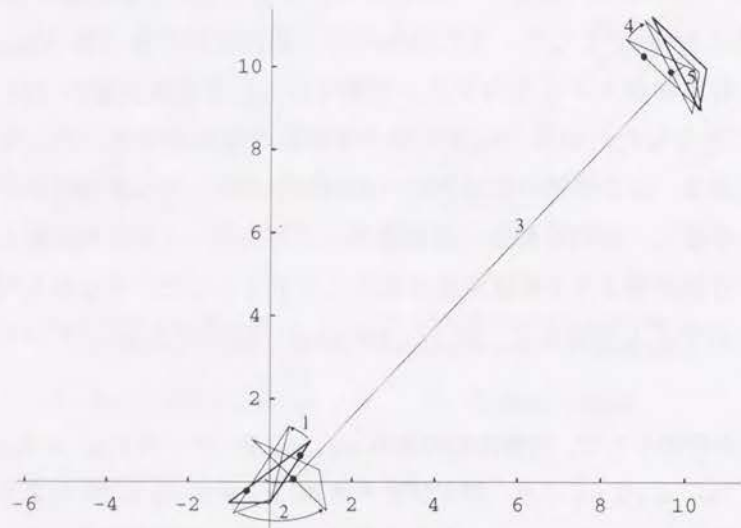


Fig. 4.4 Path of object (case 1 : $w_1 = 0, w_2 = 0$)

Table 4.3 Contact points and values of rotation or translation
(case 1 : $w_1 = 0, w_2 = 0$)

No.	Contact point	Value
1	R_4	$\theta = 20.000$
2	R_6	$\theta = 80.000$
3	T_2	$l = 13.152$
4	R_5	$\theta = -20.000$
5	T_2	$l = 0.192$

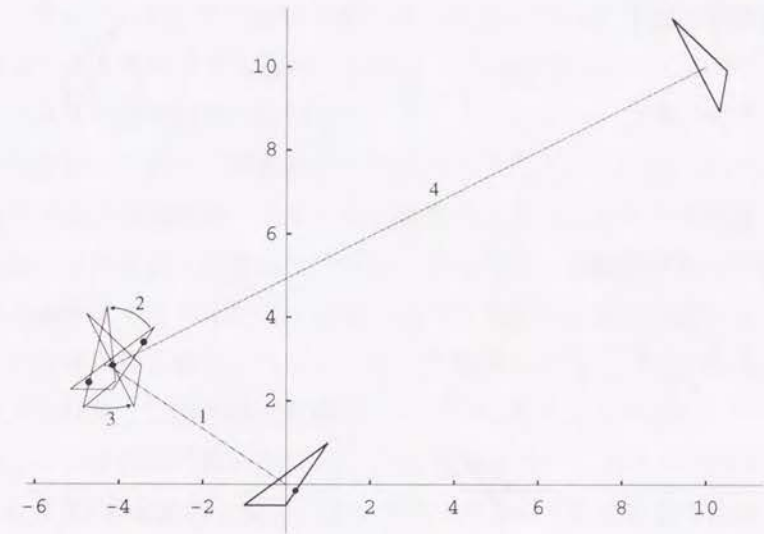


Fig. 4.5 Path of object (case 2 : $w_1 = 10, w_2 = 0$)

Table 4.4 Contact points and values of rotation or translation
(case 2 : $w_1 = 10, w_2 = 0$)

No.	Contact point	Value
1	T_1	$l = 4.989$
2	R_4	$\theta = 40.000$
3	R_6	$\theta = 40.000$
4	T_2	$l = 15.655$

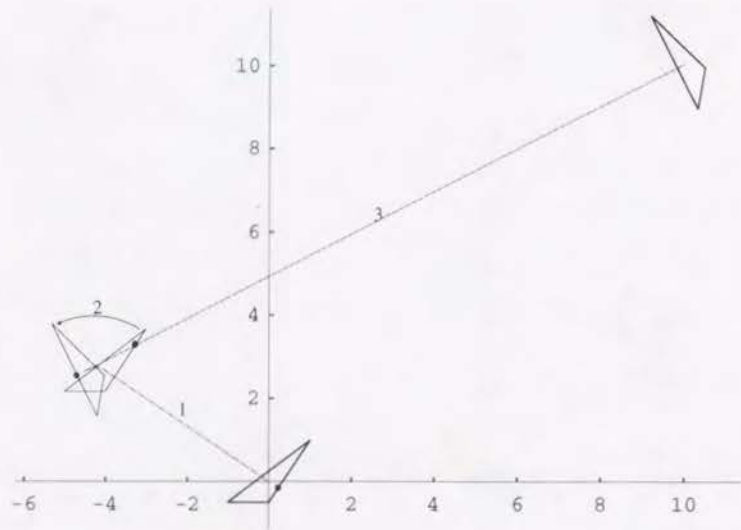


Fig. 4.6 Path of object (case 3 : $w_1 = 0, w_2 = 10$)

Table 4.5 Contact points and values of rotation or translation
(case 3 : $w_1 = 0, w_2 = 10$)

No.	Contact point	Value
1	T_1	$l = 4.816$
2	R_4	$\theta = 80.000$
3	T_2	$l = 16.275$

対してケース2では、角度の変化量を抑えるということから同方向の回転だけで目標を達成する経路をとっているが、そのため道のりはケース1よりかなり長くなっている。また回転の押し操作を2回に分け、1回の回転のみで目標を達成できる経路よりも道りが小さくなる経路をとっている。ケース3ではケース2とほぼ同様の経路が得られているものの、姿勢の変化を1回の回転のみで行っており、押し回数を抑えていることがわかる。なお $w_1 = 10, w_2 = 10$ とした場合はケース3と同じ結果となる。これらの結果より、組立作業において作業台上の部品を移動させるような、比較的作用点の変更が容易である場合はケース1のような重み付けを選択し、移動ロボットでロボットよりも大きな物を移動させるような、作用点を変更しにくい作業においてはケース3のような重み付けを選択すればよいと考えられる。

なおここでは作用点数を8個、1つの作用点について速度方向は1通り、姿勢角は6通り、回転の押し操作回数を最大4回として計画を立てたが、作用点数や姿勢角がとれる値などを多くして探索を行うと、押し操作の回数を抑えつつ道りを小さくするようなさらによい経路が求められる可能性がある。

4.8 結言

本章では対象物上の1点を押して対象物を目標点へ移動させる作業において、複数回の押し操作による作業を計画する一手法を提案した。まず直線経路と円弧経路の組み合わせで対象物の移動経路を構成することを考え、それを実現する押し操作の組み合わせから、対象物を押す回数、対象物が移動する道のり、角度の総変化量を指標とした評価関数を最小にする組み合わせを求めるといった問題として、押し作業の計画問題を定式化した。また作用点の位置および対象物の姿勢に離散的な値のみを許した場合について定式化した問題の具体的な解法について述べ、さらにその有効性を示す数値例を挙げた。

第5章

押し作業における 対象物の軌道追従制御

5.1 緒言

対象物を1点で押すという動作は基本的に不安定であり，単に対象物を直進させる作業でもロボットに自律的に行わせることは容易ではない．点接触によって対象物を目標とする位置まで押すための制御法を検討した研究は多数報告されている [23, 24, 25, 27, 28, 29]．しかしながら提案されている制御則は，押している点と対象物の重心および目標位置とをそれぞれ結んだ2つの直線のなす角により，つぎの押し動作の方向を決定するという比例制御に基づくものがほとんどである．このため重心の位置を目標とする位置へ移動させることはできても，同時に対象物の姿勢を制御することはできない．

第3章では摩擦分布が一定であるという仮定の下に，押す点を変更せずに対象物を目標とする位置と姿勢まで移動させるための対象物の軌道を計画する手法を提案した．また第4章では同様の仮定の下に，複数回の押し操作で対象物を目標とする位置と姿勢まで移動させるための作業計画法を提案した．しかしながら実環境での押し作業では，摩擦分布の推定誤差や摩擦力

分布が押し作業中に変化するという計画の段階では現れない不確定な要素が存在する。したがって与えられた軌道や経路に沿って安定に対象物を押すためにはなんらかの制御が必要となる。

本章では第3章で提案した手法により計画した軌道に対象物を追従させるための一手法を提案する。提案する手法では、近似的に対象物の運動を独立二輪駆動型移動ロボットの運動と同等に扱うことができる点を摩擦力分布より求め、その点に移動ロボットの速度フィードバック則 [59] を適用することにより対象物の運動を制御する方法をとる。これにより対象物の位置と姿勢を計画した軌道に追従させることが可能となる。提案する手法の有効性はシミュレーションによって確認する。さらに移動マニピュレータによる押し作業の一手法を示し、実験により有効性の検証を行う。

なお本章でも対象物の慣性力は摩擦力に対して無視できるものとし考慮しない。

5.2 押し作業における対象物の運動

対象物の動特性が無視できる場合、水平面上におかれた対象物を押した際に生じる対象物の運動は摩擦力分布に支配される。特に対象物を押す点（以下、作用点と呼ぶ）を固定した場合、対象物の運動は非ホロノミックな拘束を持つこととなり、対象物の位置や姿勢を自由に制御することはできない。同様に非ホロノミックな拘束をもつものに車輪駆動型の移動ロボットが挙げられる。以下では作用点を固定した場合の対象物の運動と独立二輪駆動型移動ロボットの運動との違いについて考察する。

Fig.5.1に独立二輪駆動型の移動ロボットを示す。このような移動ロボットを制御する場合、ロボットの動特性を考慮しないのであれば、一般に中心点 O における並進速度 v と回転速度 $\dot{\theta}$ を制御量とする。このように並進速度と回転速度を独立に扱えるのは機構上の特性によるものであり、車輪をどのように回転させてもロボットの運動の回転中心は車軸と一致する直線上にあ

るため、ロボットから見た中心点の速度方向は常に車軸と垂直になるからである。押し作業では作用点を固定した場合、対象物の運動を制御する量は作用点における速度であり、その大きさと方向の2つの量によって対象物の並進速度と回転速度を制御しなければならない。逆にいえば、その2つの量によって対象物の並進速度と回転速度とを独立に制御できるのであれば、既存の移動ロボットの制御則を用いることにより対象物の運動を制御できることになる。

ここで作用点における速度方向を変化させた際の対象物の運動の回転中心の存在範囲を調べてみる。Fig.5.2は作用点を固定し、作用点における速度方向を $\pi/100[\text{rad}]$ 刻みに変化させたときの対象物に生じる運動の回転中心をプロットした例であり、Fig.5.2(a), (b), (c)はそれぞれ作用点が異なる。図において、作用点から対象物の内側に伸びている2本の太い線分で挟まれる領域は速度円錐と呼ばれるものであり、作用点における速度の方向がこの範囲内にあるときは対象物を押す手先と対象物との間に滑りは生じない(3.3.2節参照)。この速度円錐は摩擦円錐に対応するもので、その領域は作用点における摩擦係数と対象物底面の摩擦力分布に依存する。また各対象物の中心にある点は摩擦中心を表す。黒い点は速度方向が速度円錐の内側にあるときの回転中心を表し、白い点は速度方向が速度円錐の外内にあるときの回転中心を表す。これらの図からわかるように、作用点を固定した場合では対象物の運動の回転中心は同一直線上になく、速度方向が常に一定となる点が存在

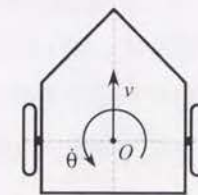


Fig. 5.1 Mobile roboto

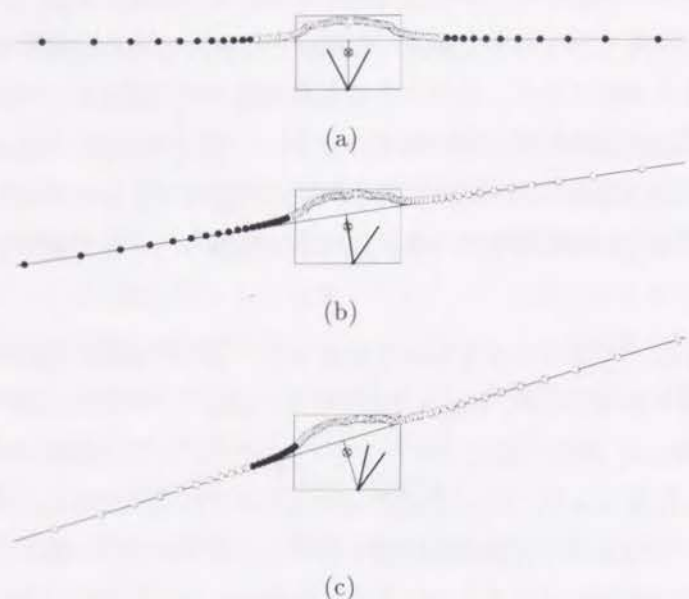


Fig. 5.2 Plots of rotation center

しないため、移動ロボットの中心点のように並進速度と回転速度を分離して扱える点が存在しない。しかしながら Fig.5.2(a) を見ると、速度方向が速度円錐内にあるときの回転中心がほぼ一直線上に並んでいることがわかる。すなわち Fig.5.2(a) のような場合、作用点の速度方向が速度円錐内にある限りにおいては、直線上の点の速度方向は近似的に一定とみなせる。さらにそのなかに、速度の大きさとその点における対象物の回転速度とを作用点における速度によって独立に制御することが可能となる点が1点存在する。以下ではこの点を擬似中心と呼ぶことにする。擬似中心における対象物の運動は移動ロボットの中心点における運動と同様の扱いができるため、擬似中心において独立二輪駆動型移動ロボットの制御則を適用することにより対象物の軌道制御が可能となる。

なお Fig.5.2(a) は摩擦中心から対象物の辺へおろした垂線の足を作用点とした場合であるが、このような場合は作用点において手先を滑らせずに対象

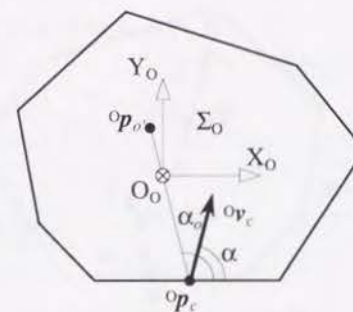
物に左右どちらの回転も生じさせることができ、制御の面から考えても理想的である。これに対し Fig.5.2(b) では手先が滑らない範囲では対象物に左回転しか生じさせることができず、Fig.5.2(c) ではさらにその回転中心の位置が限定されてしまう。以上のことを考慮し、以下では軌道計画の段階で摩擦中心が速度円錐の内側になるように作用点を選んでいるものとし、さらに押し作業を行う際は軌道計画時と同じ作用点を選ぶものとする。

5.3 擬似中心

本節では擬似中心における対象物の並進速度および回転速度と作用点の速度との関係、ならびに擬似中心の位置を求める。

原点を摩擦中心に持ち、 x 軸が作用点のある対象物の辺または作用点における対象物の接線と平行である対象物に固定された座標系を対象物座標系 Σ_O とし、 Σ_O から見た作用点および擬似中心の位置をそれぞれ ${}^O p_c$, ${}^O p_o$ とする (Fig.5.3)。また Σ_O から見た作用点の速度を ${}^O v_c$, ${}^O v_c$ が Σ_O の x 軸とのなす角を α とし、

$${}^O v_c = \bar{v}_c \begin{bmatrix} \cos \alpha \\ \sin \alpha \end{bmatrix} \quad (5.1)$$

Fig. 5.3 Object Frame Σ_O

とする。作用点の速度方向が摩擦中心に向くとき、対象物は回転運動を伴わない純粋な並進運動を行う。このとき回転中心が乗る直線 L_r 上の点における並進速度と作用点の速度は一致するため、 L_r は作用点と摩擦中心とを結んだ直線 L_g と直交していることが分かる (Fig.5.4)。さらに L_r と L_g との交点においては、幾何学的な関係より並進速度および回転速度と作用点の速度とが線形関係になる。この交点が擬似中心であり、対象物の回転速度および擬似中心の並進速度をそれぞれ $\dot{\theta}$, $\bar{v}_{o'}$, ${}^O p_{o'} - {}^O p_c$ と Σ_O の x 軸とのなす角を α_o , $l_p = \|{}^O p_{o'} - {}^O p_c\|$ とすると、以下の関係式が成立する。

$$\begin{bmatrix} \bar{v}_{o'} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \bar{v}_c \cos(\alpha - \alpha_o) \\ \frac{\bar{v}_c}{l_p} \sin(\alpha - \alpha_o) \end{bmatrix} = \mathbf{B} {}^O v_c \quad (5.2)$$

ただし

$$\mathbf{B} = \begin{bmatrix} \cos \alpha_o & \sin \alpha_o \\ \frac{1}{l_p} \sin \alpha_o & -\frac{1}{l_p} \cos \alpha_o \end{bmatrix} \quad (5.3)$$

である。押し作業においては対象物の運動の回転中心が作用点と一致することはなく、 l_p は0でない有限の値をとる。したがって $\det \mathbf{B}$ も0でない有限の値を持ち、 $[\bar{v}_{o'}, \dot{\theta}]^T$ に対して ${}^O v_c$ が一意に定まる。よって擬似中心における並進速度と回転速度は ${}^O v_c$ によって独立に与えることができる。

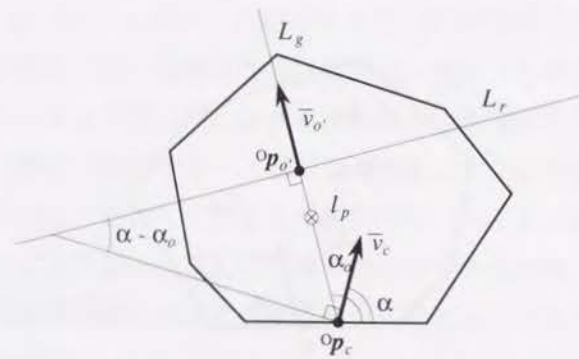


Fig. 5.4 Pseudo center

つぎに摩擦係数分布より擬似中心の位置を求める。擬似中心、摩擦中心および作用点が同一直線上にあることより次式が成立する。

$${}^O p_{o'} = {}^O p_c - \frac{l_p}{\|{}^O p_c\|} {}^O p_c \quad (5.4)$$

一方、作用点の速度と対象物の回転速度との関係は次式によって与えられる (3.3.1節参照)。

$$\dot{\theta} = \bar{v}_c \omega(\alpha) \quad (5.5)$$

ただし $\omega(\alpha)$ は α の関数であり、その関数形は摩擦係数分布によって定まる。 $\alpha = \alpha_o$ となるとき $\dot{\theta} = 0$ となるため、上式と式(5.2)の2行目の式とが $\alpha = \alpha_o$ の近傍において近似的に一致する。2式を $\alpha = \alpha_o$ の近傍においてそれぞれ1次近似すると式(5.2)の2行目の式より

$$\dot{\theta} = \frac{\bar{v}_c}{l_p} (\alpha - \alpha_o) \quad (5.6)$$

を得、式(5.5)より次式を得る。

$$\dot{\theta} = \bar{v}_c \Delta\omega(\alpha_o) (\alpha - \alpha_o) \quad (5.7)$$

ただし $\Delta\omega(\alpha_o) = \frac{\partial \omega}{\partial \alpha} |_{\alpha=\alpha_o}$ である。したがって $\Delta\omega(\alpha_o) = \frac{1}{l_p}$ となり、式(5.4)より擬似中心の位置は次式で与えられる。

$${}^O p_{o'} = \left(1 - \frac{1}{\Delta\omega(\alpha_o) \|{}^O p_c\|} \right) {}^O p_c \quad (5.8)$$

5.4 軌道追従制御

本節ではまず対象物の軌道追従制御を実現する制御則について述べ、つぎに作用点において対象物を押す手先が滑らないようにするための制限を設ける。

5.4.1 制御則

与えられる対象物の目標軌道は床面に固定された基準座標系 Σ_U から見た摩擦中心の位置 ${}^U\mathbf{p}_{od}(t)$, 速度 ${}^U\dot{\mathbf{p}}_{od}(t)$, 対象物の姿勢角 $\theta_d(t)$ および角速度 $\dot{\theta}_d(t)$ とする. また, 観測できる量は Σ_U から見た作用点における対象物の位置 ${}^U\mathbf{p}_c$ と速度 ${}^U\mathbf{v}_c$, および姿勢角 θ , 角速度 $\dot{\theta}$ とする.

5.2節で述べたように, 擬似中心において対象物は独立二輪駆動型移動ロボットと同様の扱いが可能となる. そこで擬似中心に Kanayama らが提案した移動ロボットの速度フィードバック則 [59] を適用する. この制御則は, 移動ロボットに固定された座標系から見た目標軌道に対する位置誤差と姿勢誤差をもとに速度のフィードバックを行うものであり, 実際の移動ロボットの軌道が目標軌道に一樣漸近収束することが示されている.

まず原点を擬似中心に持ち x 軸が作用点と摩擦中心とを結ぶ直線上にある座標系を $\Sigma_{O'}$ とし, この座標系から見た擬似中心の目標軌道に対する位置誤差 ${}^{O'}\mathbf{p}_e = [{}^{O'}x_e, {}^{O'}y_e]^T$ を考える (Fig.5.5). Σ_U から見た擬似中心の現在位置を ${}^U\mathbf{p}_{o'}$, 目標位置を ${}^U\mathbf{p}_{o'd}$, $\Sigma_{O'}$ から Σ_U への回転行列を ${}^UR_{O'}$ とすると, ${}^{O'}\mathbf{p}_e$ は次式で与えられる.

$${}^{O'}\mathbf{p}_e = {}^UR_{O'}^T ({}^U\mathbf{p}_{o'd} - {}^U\mathbf{p}_{o'}) \quad (5.9)$$

ここで姿勢角の誤差を $\theta_e = \theta_d - \theta$ とし, $\bar{v}_{o'}$, $\dot{\theta}$ に速度フィードバック則として

$$\begin{bmatrix} \bar{v}_{o'} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \bar{v}_{o'd} \cos \theta_e + K_x {}^{O'}x_e \\ \dot{\theta}_d + \bar{v}_{o'd} (K_y {}^{O'}y_e + K_\theta \sin \theta_e) \end{bmatrix} \quad (5.10)$$

を適用すると, 擬似中心における対象物の並進速度および回転速度を上式で得られる値とすることができれば, $\bar{v}_{o'd} > 0$ であるかぎり $\mathbf{e}_{o'} = [{}^{O'}\mathbf{p}_e^T, \theta_e]^T$ が \mathbf{o} に一樣漸近収束することが保証される. 通常押し作業においては, 作用点の速度が速度円錐内にありかつ 0 でないかぎり $\bar{v}_{o'd} > 0$ は成立する. また, 擬似中心における対象物の並進速度および回転速度と作用点の速度との関係は近似的に式(5.2)で与えられる. したがって作用点の速度 ${}^O\mathbf{v}_c$ を

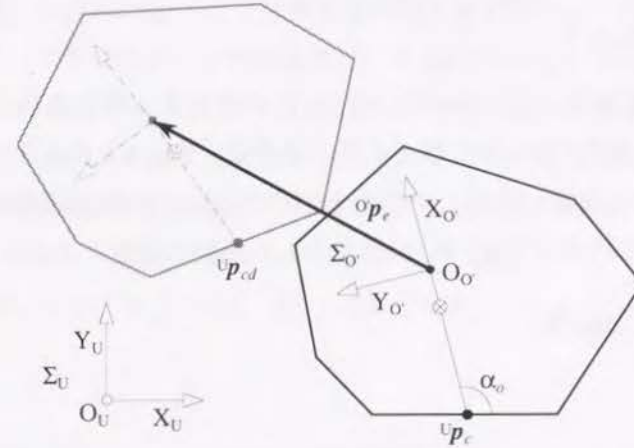


Fig. 5.5 Position error of pseudo center with respect to $\Sigma_{O'}$

$${}^O\mathbf{v}_c = \mathbf{B}^{-1}[\bar{v}_{o'}, \dot{\theta}]^T \quad (5.11)$$

とすれば, 対象物を目標軌道に追従させることが期待できる. このとき制御量, すなわち対象物を押す手先の速度 ${}^U\mathbf{v}_c$ は

$${}^U\mathbf{v}_c = {}^UR_{O'} {}^O\mathbf{v}_c \quad (5.12)$$

となる.

5.4.2 手先速度の制限

式(5.11)で得られる ${}^O\mathbf{v}_c$ は速度円錐内にある限りにおいては, 対象物を押す手先の速度を式(5.12)で与えられる ${}^U\mathbf{v}_c$ とすることによって実現できる. しかし ${}^O\mathbf{v}_c$ が速度円錐内でない場合, 式(5.12)で得られる ${}^U\mathbf{v}_c$ を手先の速度としても, 手先が対象物から離れたり対象物に対して滑ってしまうため, 対象物に望みの速度を実現させることができない. このような場合を避けるためには手先の速度に制限を設ける必要がある.

Σ_O から見た作用点における対象物側面の内向き単位法線ベクトルを ${}^O n_c$ とすると、手先が対象物から離れるのは

$${}^O n_c^T {}^O v_c < 0 \quad (5.13)$$

となるときである。したがって、式(5.13)が成立する場合は ${}^U v_c = \mathbf{o}$ とする。また ${}^O v_c$ が速度円錐の外に出てしまう場合は、**Fig.5.6**のように ${}^O v_c$ の端点から ${}^O n_c$ の延長線におろした垂線と端点に近い方の速度円錐の稜線との交点を新たな作用点の速度 ${}^O \tilde{v}_c$ の端点とし、手先の速度を次式で与える。

$${}^U v_c = {}^U R_O {}^O \tilde{v}_c \quad (5.14)$$

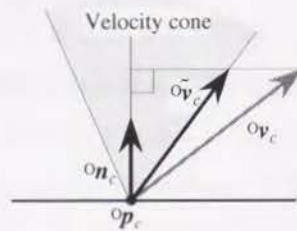


Fig. 5.6 Limitation of velocity

以上のように手先の速度を制限すると、目標軌道に対する追従性が悪くなる場合も起こり得るが、少なくとも手先が対象物から離れたり滑ったりすることによって対象物を制御できなくなる状況に陥ることは避けられると考えられる。

5.5 シミュレーション

提案した制御法の有効性をシミュレーションにより検証する。対象物の底面形状は 0.265×0.38 [m] の長方形とし、**Fig.5.7**に示す摩擦係数分布を仮定した。なお床面との接触点の個数は 25 個である。この対象物を 15[sec]

で ${}^U p_{od}(0) = [0, 0]^T$ ([m],[m]), $\theta_d(0) = 0$ [deg] から ${}^U p_{od}(15) = [1, -0.3]^T$ ([m],[m]), $\theta_d(15) = -180$ [deg] まで移動させる。作用点は摩擦中心から対象物の辺へおろした垂線の足とし、目標軌道は第3章で提案した手法により計画した。ただし作用点における摩擦係数は、軌道計画時には 0.15 としシミュレーションでは 0.25 としている。このように軌道計画時に摩擦係数の値を実際に想定される値よりも小さくしておくこと、軌道に対する制約は厳しくなるが、制御を行う際には計画された軌道に対する追従性の向上が期待できる。なおシミュレーションでは対象物の初期位置を $[0.1, 0]^T$ とし、フィードバックゲインを $K_x = 20$, $K_y = 128$, $K_\theta = 23$ とした。

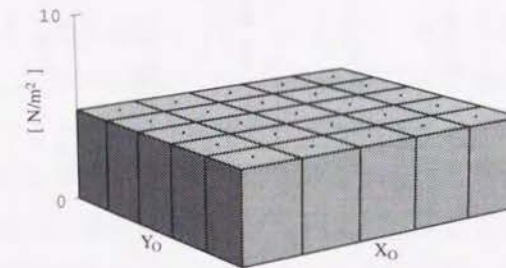


Fig. 5.7 Friction distribution

Fig.5.8に結果を示す。図において、対象物の中に描かれている3本の実線のうちの両端の2本の線分は速度円錐を表し、その中の線分が手先速度の方向を示す。また、初期位置から終端位置まで続く実線は対象物の摩擦中心の実現軌道を表し、点線は目標軌道を表す。図より対象物の軌道が目標軌道に速やかに収束していることがわかる。この結果より摩擦係数分布が変動しない場合においては、目標軌道の追従に対して提案する手法が有効であることが分かる。

なお上記の例では手先速度が制限内に納まっているが、初期位置の誤差を大きくしたり作用点における摩擦係数を小さくすると、手先速度が制限にかかり目標軌道に対する収束性は悪くなる。また他の目標軌道でも同様の現象

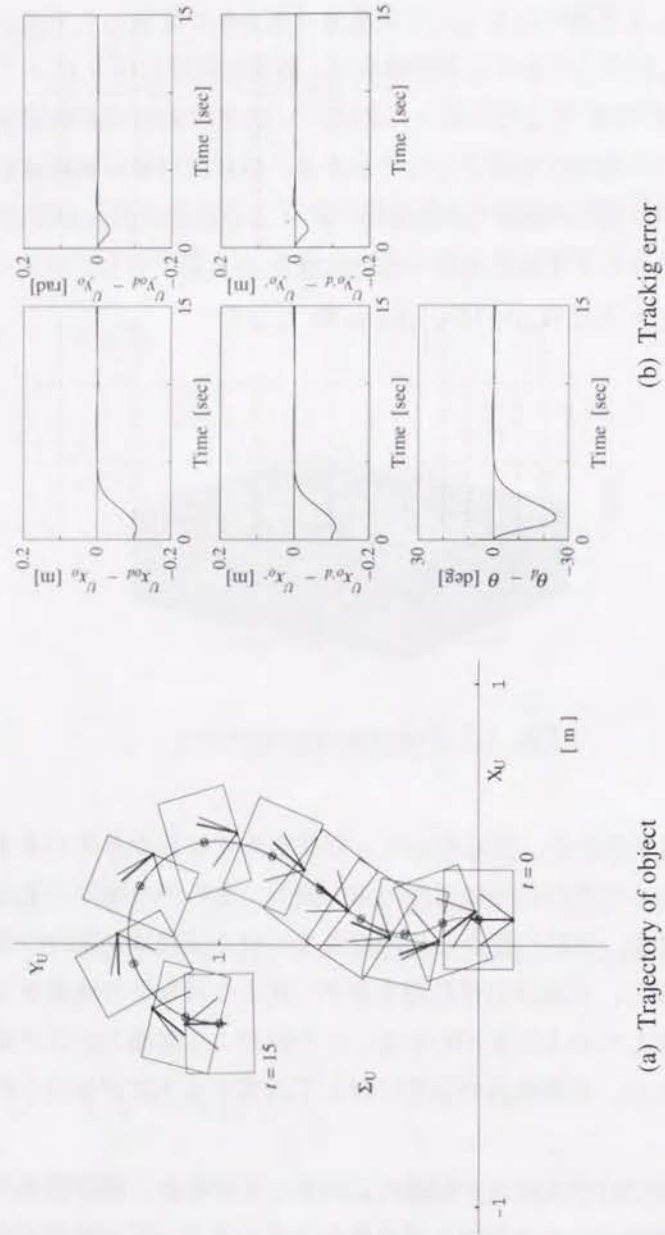


Fig. 5.8 Simulation result

が見られた。

5.6 移動マニピュレータによる押し作業

移動マニピュレータに対象物の押し作業を行わせるための一手法について述べる。ただし移動ロボットは独立二輪駆動型とし、搭載するマニピュレータは3次元 n 自由度のものとする。

5.6.1 移動マニピュレータの制御

まずマニピュレータには手先に対する位置と速度のフィードバック制御を行うことを考える。また移動ロボットは通常その制御対象として位置のほか姿勢を伴うが、ここでは移動ロボットの前方に参照点を設け、その点に対して位置と速度のフィードバック制御を行う方法[52]を採用する。この制御法では参照点で移動ロボットを引っ張るようなかたちとなり姿勢を制御することはできなくなるが、非ホロノミックな拘束がなくなるため制御が比較的簡単になるという特徴がある。以下ではこの方法を採用した移動マニピュレータの制御則について簡単に述べる。

基準座標系 Σ_U から見た参照点の位置を ${}^U p_r$ 、マニピュレータの手先の位置

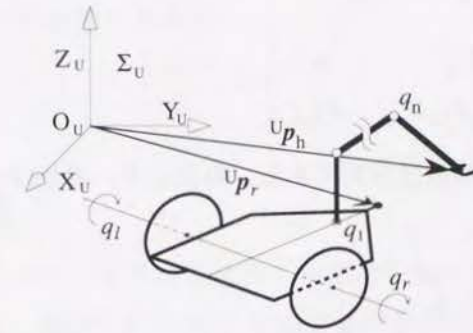


Fig. 5.9 Mobile manipulator

を ${}^U p_h$ とし、左右輪の回転角を表すベクトルを $q_p = [q_r, q_l]^T$ 、マニピュレータの関節角ベクトルを $q_m = [q_1, \dots, q_n]^T$ とする (Fig.5.9)。なお押し作業は2次元平面で考えているので、 ${}^U p_r$ は Z 座標成分を除いた2次元ベクトルとし、 ${}^U p_h$ の Z 座標成分は定数とする。このとき ${}^U r = [{}^U p_r^T, {}^U p_h^T]^T$ 、 $q = [q_p^T, q_m^T]^T$ とすると、参照点および手先の速度と関節角速度との関係は次式で与えられる。

$${}^U \dot{r} = J \dot{q} \quad (5.15)$$

ここで J は移動マニピュレータのヤコビ行列であり、次式で与えられる。

$$J = \begin{bmatrix} J_{rp} & \mathbf{0} \\ J_{hp} & J_{hm} \end{bmatrix} \quad (5.16)$$

ただし $J_{rp} = \partial {}^U p_r / \partial q_p^T$ 、 $J_{hp} = \partial {}^U p_h / \partial q_p^T$ はそれぞれ \dot{q}_p の ${}^U \dot{p}_r$ および ${}^U \dot{p}_h$ に関するヤコビ行列、 $J_{hm} = \partial {}^U p_h / \partial q_m^T$ は \dot{q}_m の ${}^U \dot{p}_h$ に関するヤコビ行列である。ここで ${}^U r$ の目標値からの誤差を ${}^U r_e$ とし、 ${}^U r$ が微小であるとする、関節座標空間における誤差は

$$q_e \simeq J^* {}^U r_e \quad (5.17)$$

$$\dot{q}_e = J^* \dot{{}^U r}_e \quad (5.18)$$

で与えられる。ただし

$$J^* = \begin{bmatrix} J_{rp}^{-1} & \mathbf{0} \\ -J_{hm}^+ J_{hp} J_{rp}^{-1} & J_{hm}^+ \end{bmatrix} \quad (5.19)$$

であり、 ${}^*+$ は * の疑似逆行列である。以上より、移動マニピュレータの制御則を次式で与える。

$$\tau = K_p q_e + K_v \dot{q}_e \quad (5.20)$$

ここで τ は q に対する駆動力で、 K_p および K_v はフィードバックゲイン行列である。

5.6.2 移動マニピュレータの目標軌道

マニピュレータの手先に関しては、式(5.12)で得る ${}^U v_e$ をその目標速度 ${}^U \dot{p}_{hd}$ とすればよいが、このときの目標位置は存在しない。そこで手先の目標位置を、各時刻における手先位置とその時の目標速度および制御時におけるサンプリング時間を用いて定めることにする。すなわち現在の手先位置から目標速度で1サンプリング時間移動した位置を目標位置にとる。したがってサンプリング時間を Δt とすると、目標位置 ${}^U p_{hd}$ は次式で与えられる。

$${}^U p_{hd} = {}^U p_h + {}^U \dot{p}_{hd} \Delta t \quad (5.21)$$

一方移動ロボットの参照点の目標軌道は基本的に任意に与えることが可能であるが、マニピュレータの操作性を考慮して以下のように定める。ただしマニピュレータの第一関節は鉛直方向に軸を持つ回転関節であり、軸は移動ロボットの中心点から離れた対称軸上にあるものとする。さらに参照点をこの第一関節の軸上にとる。このように参照点をとると、参照点の追従誤差がマニピュレータの制御に与える影響を最小にすることができる。

まず対象物座標系 Σ_O において、原点から作用点へ向かう直線の延長上に作用点から l_r 離れた点をとる (Fig.5.10)。つぎに対象物の目標軌道からその点の軌道を計算し、算出された軌道を参照点の目標軌道とする。ここで l_r はマニピュレータの可操作度 [54] が最大になるときの第一関節から手先までの $X_O Y_O$ 平面上での距離である。

5.7 実験

5.7.1 実験装置

移動マニピュレータによる対象物押し作業の実験を行い提案する手法の有効性を検証する。実験には平面2自由度並列駆動型のマニピュレータを搭載した独立二輪駆動型移動ロボットを使用した。Fig.5.11(a)に移動マニピュレータの諸元を、Fig.5.11(b)に概観をそれぞれ示す。移動ロボットの

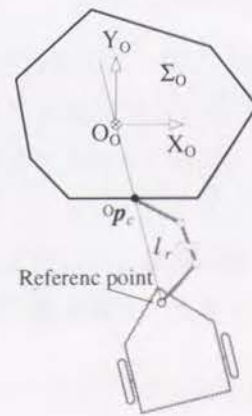


Fig. 5.10 Reference point of mobile manipulator

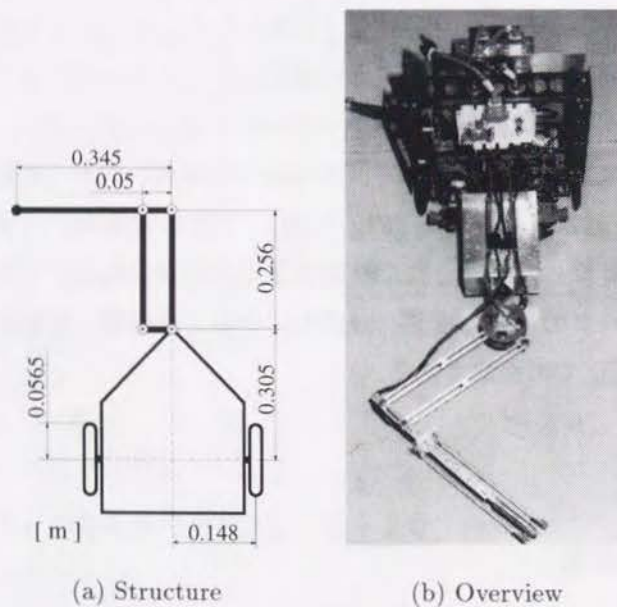


Fig. 5.11 Mobile manipulator used for experiments

車輪の回転角はオプティカルエンコーダによって検出され、回転速度および

位置と姿勢はエンコーダの出力から計算される。これらの処理はモトローラ製CPU68000を搭載したボードコンピュータで行われ、その値はDPM(Dual Port Memory)を介してホストコンピュータに取り込まれる。車輪の駆動は、ホストコンピュータからDPMを介して送られた司令値にしたがい、ボードコンピュータがモータを駆動することにより行われる。マニピュレータの各関節角はポテンシオメータからの電圧値を、角速度はポテンシオメータからの出力を微分回路に通した後の電圧値を、それぞれA/Dボードを介してホストコンピュータに取り込むことにより得られる。マニピュレータの各モータは、ホストコンピュータからの司令値がD/Aボード、ドライバアンプを介して出力されることにより駆動される。マニピュレータの手先には駆動力のない平行リンクとポテンシオメータが取り付けられており、これにより対象物の姿勢と姿勢角速度が測定できるようになっている。ホストコンピュータにはオーバードライブプロセッサ(CPU80486:66MHz)と数値演算プロセッサ(FPU80487)を搭載したNEC製PC-9801FAを使用した。

5.7.2 実験結果

対象物には重さ0.15[kg]、大きさ0.265×0.38×0.12[m]の段ボール製直方体を使用し、底面の摩擦力分布はFig.5.7に示す分布を仮定した。また対象物と手先との摩擦係数、対象物の目標軌道、ゲイン K_x 、 K_y 、 K_θ はシミュレーションと同じ値とした。移動マニピュレータの制御に関するゲインは、 $K_p = \text{diag}[1, 1, 1.2, 1.2]$ 、 $K_v = \text{diag}[0.02, 0.02, 1, 1]$ である。以上の設定のもとで、対象物の初期位置を $[0, 0]^T$ ([m],[m])、初期姿勢を0[deg]、サンプリングタイムを5[msec]として実験を行った。結果の一例をFig.5.12に示す。図において、参照点の目標軌道および移動マニピュレータは点線で、参照点の実現軌道は実線で表した。また対象物などはFig.5.8(a)と同様に示した。

実験は凹凸の少ない水平な床面上で行ったが、完全な平面ではないため摩擦力分布は対象物の移動に伴って変動していると思われる。しかし姿勢角に対する追従性が少し悪いものの、対象物が目標軌道にはほぼ追従していること

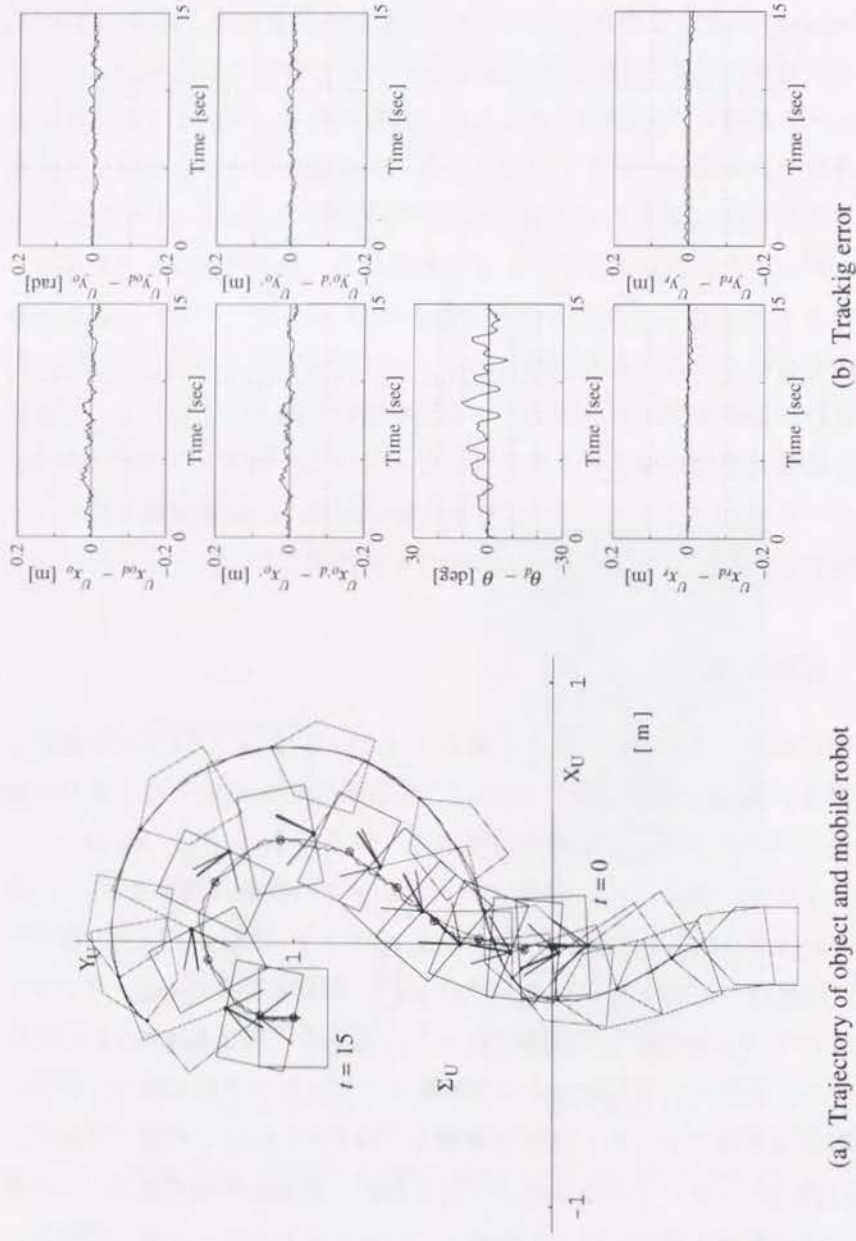


Fig. 5.12 Experimental result

がFig.5.12より分かる. この結果より, 摩擦力分布が変動しないという仮定が成り立たない実環境においても, 提案する手法が有効であることが分かる.

5.8 結言

本章では押し作業における対象物の軌道追従制御法を提案した. まず点接触による押し動作において, 対象物の運動を近似的に車輪型移動ロボットの中心点における運動と同様に扱うことができる点(擬似中心)が存在することを示し, 擬似中心の位置を摩擦力分布より求める方法について述べた. さらに擬似中心を利用して対象物の軌道追従制御を実現する制御則を導き, その有効性をシミュレーションにより示した. また移動マニピュレータで押し作業を行うための一手法を示し, 実機による実験を行うことで, 平坦な床面上で目標軌道に沿って対象物を押し移動させることが提案する手法で可能となることを示した.

第6章

移動マニピュレータの 動的軌道計画と動的制御

6.1 緒言

前章まではロボットによる自律的な押し作業の確立を目的とした議論を行った。提案した手法は床に固定されたような通常のマニピュレータにも適用できるものであるが、運搬作業などの広範囲な押し作業を考え、特に移動マニピュレータによる実現に焦点をおいた。逆の見方をすると押し作業は移動マニピュレータの1つの適用例であると言える。本章では押し作業とは異なる作業への移動マニピュレータの適用を考え、移動マニピュレータ自体の軌道計画および制御に焦点をおく。

移動マニピュレータに高速高精度な作業の遂行を要求する場合、搭載するマニピュレータと台車との間の動的な干渉を考慮した制御法が必要となる。移動マニピュレータの動的な制御法には、台車の振動による手先の軌道追従誤差をマニピュレータの関節駆動力で補償する方法[36, 37]、ニュートンオイラー法による逆動力学解法を用いてマニピュレータの運動による台車への影響や斜面における重力の影響を車輪の駆動力で補償する方法[38, 39]などが

提案されているが、これらの研究では手先の軌道追従性の向上を目的としているため、台車の目標軌道に対する追従制御に関しては考察されていない。これに対し Yamamoto ら [42] は、マニピュレータと台車との動的な干渉を考慮しつつ両者をそれぞれの目標軌道に追従させる制御法を提案している。しかしながら従来の動的な制御法を提案している研究では、手先の位置や姿勢の制御を目的としているものが多く、広範囲な研磨作業のように環境との接触による力を制御しなければならないような作業を対象としている研究はまだない。一方移動マニピュレータを動的に制御する場合、その制御性能を十分に発揮し、さらに作業の効率を計るためには、駆動力などを考慮できるような加速度の次元での大域的な軌道計画が必要と思われるが、そのような移動マニピュレータの軌道計画法はまだ報告されていない。

本章では移動マニピュレータを動的に制御することを目的とし、加速度の次元での移動マニピュレータの軌道計画法を提案する。また環境との接触による力を制御しなければならない作業を考慮し、手先の位置と力および台車の位置を同時に制御する移動ロボットの動的ハイブリッド制御法 [54] を提案する。さらに提案する計画法によって生成した軌道を用いて実験を行うことにより、軌道計画法の有効性ならびに動的ハイブリッド制御法の有効性を確認する。提案する軌道計画法は、与えられた手先の目標軌道から、駆動力やマニピュレータの操作性を指標とした評価関数を作業時間全体を通して最少とする台車の軌道を、車輪の角加速度空間で探索するものである。特に手先を壁などに押し付けて移動するような作業に対しては、手先の操作力方向を考慮する評価関数を加えることにより、マニピュレータが拘束面に力を加え易い姿勢となるような台車の軌道を生成する。

6.2 状態方程式

本章では Fig.6.1 に示すような 3 次元 n 自由度マニピュレータを搭載した独立二輪駆動型の移動マニピュレータを対象とする。

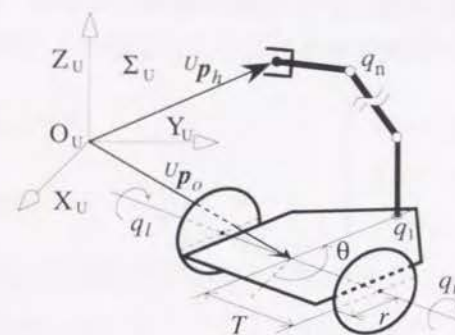


Fig. 6.1 Mobile manipulator

まず移動マニピュレータの状態方程式を記述しておく。車輪と台車の対称面との交点を台車の中心点とし、中心点を通り床面に平行な平面に XY 平面が一致するように基準座標系 Σ_U をとる。また Σ_U から見た台車の中心点を $U p_o$ 、台車の姿勢角を θ とする。ただし $U p_o$ は Z 座標成分を除いた 2 次元ベクトルである。ここで左右輪の回転角を表すベクトルを $q_p = [q_r, q_l]^T$ とし、車輪と床との間に滑りが生じないものすると以下の関係式を得る。

$$U \dot{p}_o = \begin{bmatrix} \frac{r}{2} \cos \theta & \frac{r}{2} \cos \theta \\ \frac{r}{2} \sin \theta & \frac{r}{2} \sin \theta \end{bmatrix} \dot{q} \quad (6.1)$$

$$\dot{\theta} = \frac{r}{2T} (\dot{q}_r - \dot{q}_l) \quad (6.2)$$

ただし T は台車の中心から車輪までの距離、 r は車輪の半径である。つぎにマニピュレータの関節角ベクトルを $q_m = [q_1, \dots, q_n]^T$ とし、 $q = [q_p^T, q_m^T]^T$ に対する駆動力を τ とすると、移動マニピュレータの運動方程式は次式のように表せる [42]。

$$M(q)\ddot{q} + h(q, \dot{q}) = \tau \quad (6.3)$$

ここで $M(q)$ は慣性行列、 $h(q, \dot{q})$ は遠心力、コリオリ力を表す。

以上より, $\xi = [{}^U p_o^T, \theta, q_m^T, \dot{q}^T]^T$ を状態変数とすると, 式(6.1)~(6.3)より系の状態方程式として次式を得る.

$$\dot{\xi} = \begin{bmatrix} \frac{r}{2}(\dot{q}_r + \dot{q}_l) \cos \theta \\ \frac{r}{2}(\dot{q}_r + \dot{q}_l) \sin \theta \\ \frac{r}{2T}(\dot{q}_r - \dot{q}_l) \\ \dot{q}_m \\ -M^{-1}h \end{bmatrix} + \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix} \tau \quad (6.4)$$

6.3 軌道計画

マニピュレータの手先の目標軌道が与えられた場合に, これを満足する台車の目標軌道を計画する方法について述べる. 移動マニピュレータは通常冗長性を持つため, 手先の軌道を与えたとしても台車の軌道は一意には定まらずいくらかの自由度が残る. この自由度を作業内容に応じて定めた評価関数の値が小さくなるように決めることが, 提案する軌道計画法の基本的な考えである. 以下ではまず系に存在する制約条件について述べる. つぎに評価関数を定義し, 軌道計画問題を最適制御問題として定式化する.

6.3.1 制約条件

6.3.1.1 手先の制約

移動マニピュレータでの作業では, その作業内容に応じてまず先に手先の軌道が与えられることが多いと考えられる. したがって, ここでは手先の目標軌道が時間関数 ${}^U p_h(t)$ として与えられているものとし, さらに ${}^U p_h(t)$ は時間に関して2回微分可能であるものとする. また与えられる作業が手先を壁に押し付けて移動するような作業である場合は, ${}^U p_h(t)$ は壁を表す拘束面上にあるものとする. このとき手先の位置は台車の位置と姿勢およびマニピュレータの関節角の関数として $f(\xi_p)$ ($\xi_p = [{}^U p_o^T, \theta, q_m^T]^T$) と表せる. 手先が常に目標軌道上になければならないことより, 各時刻において ${}^U p_h(t)$ と

$f(\xi_p)$ は一致していなければならない. これが手先に関する制約であり, その条件は状態量に関する制約式として以下のように与えられる.

$${}^U p_h(t) - f(\xi_p) = 0 \quad (6.5)$$

6.3.1.2 台車の制約

手先を壁に押し付けて移動するような作業の場合には, 壁すなわち手先の拘束面と台車とが接触しないようにする必要がある. 以下では拘束面と台車とが接触しないための台車の位置と姿勢に関する制約を考慮する. まず台車上に代表点を選び, 台車に固定した座標系 Σ_P から見たその位置を ${}^P p_{o'}$ ($\in \mathbf{R}^3$) とする. このとき基準座標系 Σ_U から見た代表点の位置 ${}^U p_{o'}$ は

$${}^U p_{o'} = {}^U R_P {}^P p_{o'} + \begin{bmatrix} {}^U p_o \\ 0 \end{bmatrix} \quad (6.6)$$

となる. ただし ${}^U R_P$ は Σ_P から Σ_U への回転行列である. さらに安全半径として r_s を定める. ここで拘束面と代表点との距離を表す関数を $d_c({}^U p_{o'})$ とすると, 拘束面と台車との非接触条件は

$$d_c({}^U p_{o'}) - r_s \geq 0 \quad (6.7)$$

となる (Fig.6.2). また拘束面と代表点との距離が $d_c({}^U p_{o'})$ のような形の関数で表せない場合は, 拘束面を平面や球状の仮想障害物との組み合わせで近似し, それらすべてとの非接触条件で台車に関する制約を表す. 以上をまと

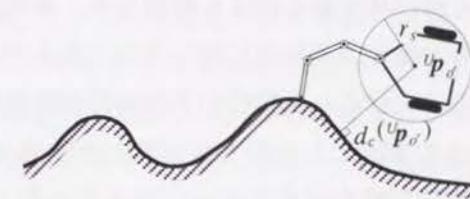


Fig. 6.2 Constraint surface and mobile manipulator

めると、拘束面と台車との非接触条件は状態量に関する制約式として次式のように表せる。

$$g(\xi_p) \geq 0 \quad (6.8)$$

6.3.2 評価関数

評価関数を定める指標としては、台車の移動する道のり、関節速度の分散、関節駆動力の分散をそれぞれ小さくすることが挙げられる。また台車の車輪に滑りが生じるのは、台車の姿勢の変化速度や左右輪の駆動力の差の急激な変動にも起因すると考えられるので、それらを抑えることも必要である。さらに、手先の操作性を良くするという点でマニピュレータの可操作度 [54] が常に大きくなっていることが望ましい。以上の指標が作業時間全体 $([0, t_f])$ を通してなるべく満たされるようにするために、評価関数を次式で定義する。

$$\Psi_p = \int_0^{t_f} \left[W_l \frac{r}{2} (q_r + q_l) + \dot{q}^T W_q \dot{q} + \tau^T W_\tau \tau + W_\omega \dot{\theta}^2 + W_n (\tau_r - \tau_l)^2 + W_j (\det \mathbf{J}_{hm} \mathbf{J}_{hm}^T)^{-\frac{1}{2}} \right] dt \quad (6.9)$$

ここで τ_r, τ_l は車輪の駆動力、 $\mathbf{W}_q, \mathbf{W}_\tau$ は重みを表す対角行列、 W_l, W_ω, W_n, W_j は重み係数である。また $\mathbf{J}_{hm} = \partial^U \mathbf{p}_h / \partial \mathbf{q}_m^T$ は $\dot{\mathbf{q}}_m$ の $^U \mathbf{p}_h$ に関するヤコビ行列であり、 $(\det \mathbf{J}_{hm} \mathbf{J}_{hm}^T)^{-\frac{1}{2}}$ はマニピュレータの可操作度である。

さらに力制御を必要とする作業では、搭載したマニピュレータが拘束面に力を加え易いような姿勢をとるようにすることを考え、上述の評価に加えて手先が拘束面に加える力の方向を評価する関数を定める。手先の操作力を評価する指標には操作力楕円体 [54] というものがある。本来操作力楕円体は手先が静止している場合の操作力の評価に用いるのであるが、力の方向を考慮するという意味からここではこれを操作力方向の評価指標に利用する。ただし力を制御する方向は 1 方向とし手先と拘束面との摩擦はないものとする。

マニピュレータの関節駆動力ベクトル τ_m の大きさの上限を定めたとき、手先の出し得る力の大きさが最大となる操作力の方向は、そのときの操作力楕円体のもっとも長い主軸の方向となる (Fig.6.3)。いまマニピュレータが

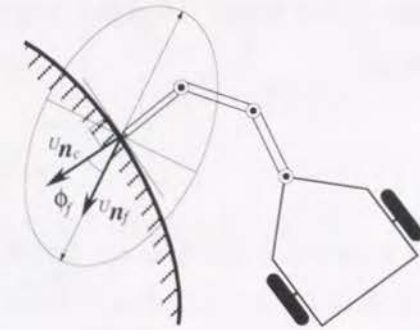


Fig. 6.3 Manipulating-force ellipsoid

特異姿勢でないものとする。台車から見たその軸の方向は $\mathbf{J}_{hm} \mathbf{J}_{hm}^T$ の最少固有値に対する固有ベクトル $^P \mathbf{n}_f$ で与えられる。ただし手先にもっとも近い関節から手先に向かう方向を正の向きとし、 $^P \mathbf{n}_f$ の大きさは 1 とする。このとき基準座標系から見た $^P \mathbf{n}_f$ は

$$^U \mathbf{n}_f = {}^U \mathbf{R}_P {}^P \mathbf{n}_f \quad (6.10)$$

となる。ここで手先が接触している点における拘束面の内向き単位法線ベクトルを $^U \mathbf{n}_c$ とし、 $^U \mathbf{n}_c$ と $^U \mathbf{n}_f$ とのなす角を ϕ_f とすると

$$\cos \phi_f = {}^U \mathbf{n}_c^T {}^U \mathbf{n}_f \quad (6.11)$$

となる。ただし $^U \mathbf{n}_c$ は手先の軌道と同様に時間の関数で与えられているものとする。 $\phi_f = 0$ となるときがもっとも理想的に手先が拘束面に力を加えることになり、力制御を行う場合には、作業時間全体を通してできるだけそのような状態になることが望ましい。これより、手先が拘束面に加える力の方向を評価する関数として次式を定義する。

$$\Psi_f = \int_0^{t_f} W_f (1 - \cos \phi_f) dt \quad (6.12)$$

ただし W_f は重み係数である。

以上より、位置制御のみで手先の力を考慮する必要のない作業に対する軌道計画では評価関数を

$$\Psi = \Psi_p \quad (6.13)$$

で与え、手先の力を制御する必要があるような作業に対する軌道計画では、評価関数に式(6.9)と式(6.12)をたし合わせた次式を用いる。

$$\Psi = \Psi_p + \Psi_f \quad (6.14)$$

6.3.3 定式化

軌道計画の問題は、状態方程式(6.4)、制約式(6.5)(6.8)および前節で述べた評価関数により最適制御問題として定式化できる。しかし状態方程式が非線形でさらに状態量に関する制約がある場合には、解析的に解を求めるのは難しく、計算機により数値的に解を求めることとなる。そのような場合、等式制約式を変形し状態方程式に代入することで式を整理し、問題を解き易くしておくという方法が多く用いられる。本手法でもまず問題を解き易くするために、等式で与えられる制約式(6.5)を用いて状態方程式(6.4)を整理する。式(6.5)を時間に関して2回微分し $\ddot{\mathbf{q}}$ について解くと次式を得る。

$$\ddot{\mathbf{q}} = \mathbf{J}_h^+ \{ \mathbf{U} \ddot{\mathbf{p}}_h - \dot{\mathbf{J}}_h \dot{\mathbf{q}} \} + \{ \mathbf{I}_{n+2} - \mathbf{J}_h^+ \mathbf{J}_h \} \mathbf{k} \quad (6.15)$$

ここで $\mathbf{J}_h = \partial \mathbf{f} / \partial \mathbf{q}^T$ は $\dot{\mathbf{q}}$ の $\mathbf{U} \dot{\mathbf{p}}_h$ に関するヤコビ行列である。また右辺第2項が冗長成分であり、 \mathbf{k} は $n+2$ 次元任意ベクトルである。ただしここではマニピュレータの関節はすべて回転関節であることを前提としている。ここで \mathbf{k} を新たな入力と見なし、式(6.15)を式(6.3)に代入したものをさらに式(6.4)に代入すると次式を得る。

$$\dot{\boldsymbol{\xi}} = \begin{bmatrix} \frac{r}{2}(\dot{q}_r + \dot{q}_l) \cos \theta \\ \frac{r}{2}(\dot{q}_r + \dot{q}_l) \sin \theta \\ \frac{r}{2T}(\dot{q}_r - \dot{q}_l) \\ \dot{q}_m \\ \mathbf{J}_h^+ \{ \mathbf{U} \ddot{\mathbf{p}}_h - \dot{\mathbf{J}}_h \dot{\mathbf{q}} \} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_{n+2} - \mathbf{J}_h^+ \mathbf{J}_h \end{bmatrix} \mathbf{k} \quad (6.16)$$

つぎに終端時刻 $t = t_f$ における目標状態を $\boldsymbol{\xi}_f$ とし、終端時刻の状態を評価する関数を以下のように定める。

$$\Phi(\boldsymbol{\xi}(t_f)) = \{ \boldsymbol{\xi}(t_f) - \boldsymbol{\xi}_f \}^T \mathbf{W}_p \{ \boldsymbol{\xi}(t_f) - \boldsymbol{\xi}_f \} \quad (6.17)$$

ただし \mathbf{W}_p は重みを表す対角行列である。この関数と式(6.13)または式(6.14)より以下のような拡張評価関数を作る。

$$\tilde{\Psi} = \Phi + \Psi \quad (6.18)$$

以上より軌道計画問題は、微分方程式(6.16)において、初期状態 $\boldsymbol{\xi}(0) = \boldsymbol{\xi}_0$ 、および制約式(6.8)を満たし、式(6.18)を最小にする入力 $\mathbf{k}(t)$ を求めるという最適制御問題として扱うことができる。第3章と同様、本章でもこの定式化された問題を Fletcher-Reeves 法にペナルティ法を組み合わせた方法を用いて計算機により数値的に解く。ただし対象とする作業が手先を壁に押し付けなければならないような作業ではなく、さらに環境や他の対象物と台車との接触を考慮する必要がない作業では、制約式(6.8)はなくなり、数値計算においてもペナルティ法を組み合わせる必要はなくなる。

6.4 数値例

提案する軌道計画法の数値例を示す。想定する作業は、与えられた拘束面に手先を押し付けながら移動マニピュレータを移動させる作業とする。Fig.6.4 使用する移動マニピュレータのモデルを、Table 6.1に各パラメータをそれぞれ示す。搭載するマニピュレータは平面2自由度並列駆動型のものとした。また拘束面はFig.6.5に示すように、2つの直線(line 1,2)と3つの円弧(arc

$= \text{diag}[1, 1, 1, 1]$, $W_l = 1$, $W_\omega = 2$, $W_n = 2$, $W_j = 4$ とした. 操作力方向の評価に関する重みは, ケース1: $W_f = 20$, ケース2: $W_f = 150$ とし, 重みの違いによる生成軌道の違いを比較することにする. 以上の設定のもとで, 移動マニピュレータの初期状態および目標状態をそれぞれ $\xi_0 = [0.346, -0.433, \frac{\pi}{2}, 0, \frac{\pi}{2}, 0, 0, 0, 0]^T$, $\xi_f = [0.346, 1.267, \frac{\pi}{2}, 0, \frac{\pi}{2}, 0, 0, 0, 0]^T$ ([m],[m],[rad],[rad],[rad],[rad/sec],[rad/sec],[rad/sec],[rad/sec]) として計算を行った.

ケース1の計算結果を Fig.6.7 に, ケース2の計算結果を Fig.6.8 にそれぞれ示す. Fig.6.7(a), Fig.6.8(a) において, 手先から拘束面の内側に伸びている2つの線分は, 点線が拘束面の法線ベクトルを, 実線が操作力楕円体の長

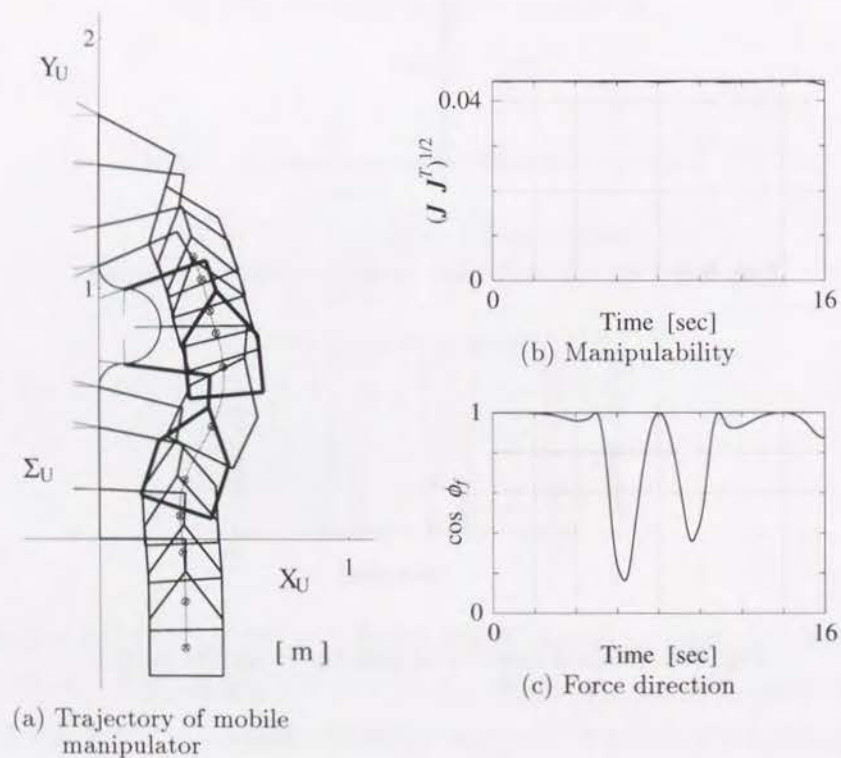


Fig. 6.7 Numerical example (case 1 : $W_f = 20$)

軸の方向をそれぞれ示す. ケース1とケース2を比較すると, 可操作度に関してはケース2の方が操作力方向の評価を良くするために若干小さくなってはいるものの, どちらも作業時間全体をとおしてほぼ最大値に近い値をとっている. しかし操作力の方向に関しては, 6.5秒と9.5秒付近で違いが見られる (Fig.6.7(a), Fig.6.8(a) において太線で示した部分). ケース1では操作力が拘束面の法線と垂直に近い方向となっており, 手先が拘束面に力を加えにくい状態になっているが, ケース2ではケース1に比べてかなり良い評価値を得ている. これらの結果より手先の操作力の方向を考慮することで, 拘束面に力を加え易い軌道が得られることがわかる.

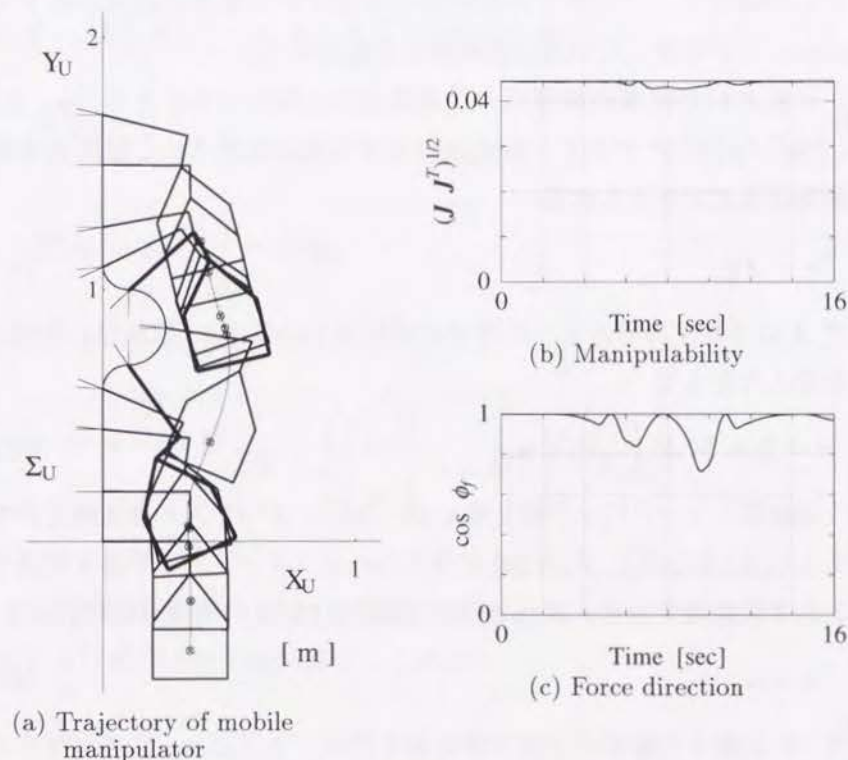


Fig. 6.8 Numerical example (case 2 : $W_f = 150$)

6.5 位置制御と力制御

本節ではまず移動マニピュレータの動的な位置制御法を紹介し、つぎに力制御の一手法としてそれを拡張した動的ハイブリット制御法を示す。

6.5.1 位置制御

Yamamotoら[42]は、台車の位置とマニピュレータの手先の位置を同時にそれぞれの目標軌道に追従させる動的な軌道追従制御法を提案している。この制御法は台車の位置制御に look-ahead control という制御法[60]を用いている。look-ahead control の特徴は台車の前方に参照点を設け、この点を目標軌道に追従させることを目的としている点であり、台車の姿勢を制御できなくなるが非ホロノミックな拘束がなくなるという利点がある。以下ではYamamotoらが提案した軌道追従制御法を紹介する。

Σ_U で表された台車の参照点の位置を表す2次元ベクトルを U_{p_r} とし、 $U_{\dot{r}} = [U_{p_r}^T, U_{p_h}^T]^T$ とすると、参照点および手先の速度 $U_{\dot{r}}$ と関節角速度 \dot{q} との関係は次式で与えられる。

$$U_{\dot{r}} = J\dot{q} \quad (6.19)$$

ここで J は移動マニピュレータのヤコビ行列である(5.6.1節参照)。いま u_r を新たな入力として

$$\tau = h + MJ^* [-\dot{J}\dot{q} + u_r] \quad (6.20)$$

という非線形フィードバック則を考える。ただし J^* は J の逆変換を表す行列である(5.6.1節参照)。式(6.20)で与えられるフィードバック側を移動マニピュレータに適用すると、式(6.3)および式(6.19)を時間微分した式より

$$U_{\ddot{r}} = u_r \quad (6.21)$$

という $U_{\ddot{r}}$ に関する線形かつ非干渉な系を得る。さらに u_r に適当なサーボ補償器を適用すれば、マニピュレータの手先と台車の参照点を同時に目標軌道に追従させる制御系が構成できる。

6.5.2 力制御

手先を壁などに押しつけながら移動マニピュレータを動かす作業を考えたとき、手先に対しては位置制御と力制御を同時に行わなければならない。以下では手先の位置と力および台車の位置をそれぞれの目標軌道に追従させるための移動マニピュレータの動的ハイブリット制御法を示す。ハイブリット制御法は、位置を制御すべき方向と力を制御すべき方向とを分離して制御を行う方法で、拘束面に垂直な方向すなわち法線方向には力制御を行い、法線と垂直な方向には位置制御を行う。

いま手先の拘束数を m とし、拘束された手先の軌道を表す $(3-m)$ 次元関数を $y_h = s(U_{p_h})$ とする。さらに手先の位置制御方向を表す行列を $E_p = \partial y_h / \partial U_{p_h}^T (\in R^{(3-m) \times 3})$ 、力制御方向を表す行列を $E_f (\in R^{m \times 3})$ とする。ただし $E = [E_p^T \ E_f^T]^T$ とするとき以下の2式を満足しているものとする。

$$E^U \dot{p}_h = \begin{bmatrix} \dot{y}_h \\ \mathbf{o} \end{bmatrix} \quad (6.22)$$

$$E^U \ddot{p}_h = \begin{bmatrix} \ddot{y}_h \\ \mathbf{o} \end{bmatrix} - \dot{E}^U \dot{p}_h \quad (6.23)$$

このとき u_1, u_2, u_3 を新たな入力として

$$\tau = h + MJ^* \begin{bmatrix} u_1 - \dot{J}_{rp}\dot{q}_p \\ E^{-1} \left\{ \begin{bmatrix} u_2 \\ \mathbf{o} \end{bmatrix} - \dot{E} J_h \dot{q} \right\} - \dot{J}_h \dot{q} \\ + J_h^T E_f^T u_3 \end{bmatrix} \quad (6.24)$$

という非線形状態フィードバック則を移動マニピュレータに適用すると、式(6.3)および式(6.19)を時間微分した式より

$$U_{\ddot{p}_p} = u_1 \quad (6.25)$$

$$\ddot{y}_h = u_2 \quad (6.26)$$

$$C f_h = u_3 \quad (6.27)$$

という線形かつ非干渉化された系を得る. ここで ${}^C f_h (\in R^m)$ は ${}^U p_h$ に原点を持ち各制御方向に座標軸を持つ座標系で記述された手先が拘束面に加える力である. さらに u_1, u_2 に適当な位置サーボ補償器を, u_3 に適当な力サーボ補償器を加えれば, マニピュレータの手先と台車の参照点を目標位置軌道に, マニピュレータの手先力を目標力軌道にそれぞれ追従させる制御系が構成できる.

6.6 実験

動的ハイブリッド制御法の有効性を実験により検証する. 手先の目標軌道は数値例と同じものを用い, 台車の参照点の目標軌道には, 数値例で得られたケース1およびケース2の軌道をそれぞれ参照点の軌道に変換したものを用いる. この2つの軌道に対する実験結果を比較することにより, 軌道計画で手先の操作力の方向を考慮することの有効性も確認する.

実験に用いた移動マニピュレータは平面2自由度並列駆動型のマニピュレータを搭載した移動ロボットで, 各パラメータは数値例で用いたものと同じである. 台車の参照点の位置はマニピュレータの第一関節の位置とした. 手先には力覚センサが取り付けられており, 手先の押し付け力を測定できるようになっている. さらに拘束面との摩擦を減らすために, 力覚センサの先にはベアリングを取り付けてある. サーボ補償器は以下のように選んだ.

$$u_1 = \ddot{p}_{rd} + K_{vr}(\dot{p}_{rd} - \dot{p}_r) + K_{pr}(p_{rd} - p_r) \quad (6.28)$$

$$u_2 = \ddot{y}_{hd} + K_{vh}(\dot{y}_{hd} - \dot{y}_h) + K_{ph}(y_{hd} - y_h) \quad (6.29)$$

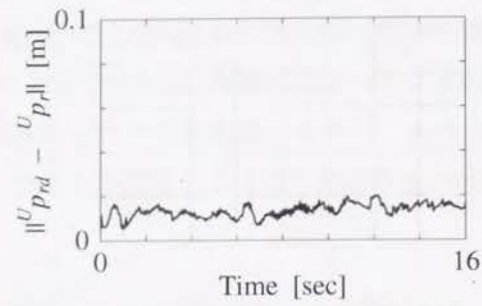
$$u_3 = {}^C f_{hd} + K_{pf}({}^C f_{hd} - {}^C f_h) + K_{if} \int_0^t ({}^C f_{hd} - {}^C f_h) dt \quad (6.30)$$

ここで添字 d はそれぞれの値に対する目標値を表す. 各ゲインは $K_{vr} = \text{diag}[2, 2]$, $K_{pr} = \text{diag}[40, 40]$, $K_{vh} = 24$, $K_{ph} = 200$, $K_{pf} = 1.1$, $K_{if} = 14$, とした. 以上の設定でサンプリングタイムを $5[\text{msec}]$ として実験を行った.

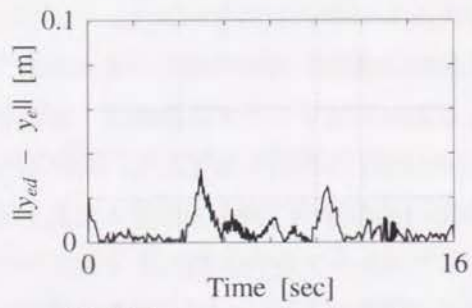
Fig.6.9にケース1の軌道に対する結果を, Fig.6.10にケース2の軌道に対する結果をそれぞれ示す. ケース2では参照点の位置, 手先の位置ならびに押し付け力が目標軌道にほぼ追従しており, 動的ハイブリッド制御法の有効性が現れている. しかしケース1では6.5秒付近で力の追従性が悪くなっている. これはその部分でアームが拘束面に力を加えにくい姿勢になっているためであると考えられる. これより軌道計画で手先の操作力の方向を考慮することが力制御を行う上で有効であることが確認できた.

6.7 結言

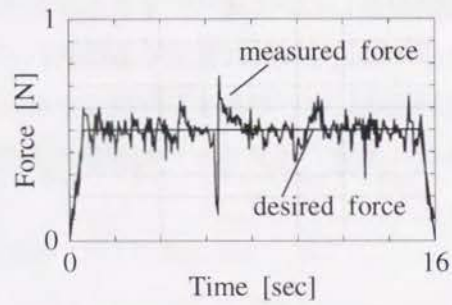
本章では移動マニピュレータの動的な制御法, およびそれに対応する移動マニピュレータの大域的な軌道計画法の確立に焦点をおき, まず, 与えられた手先の作業に対し移動マニピュレータを効率よく移動させるための台車の軌道を加速度の次で計画する手法を提案した. 特に手先力の制御を必要とする作業に対し, 手先の操作力の方向を評価する関数を軌道計画の評価関数に加えることで, マニピュレータが拘束面に力を加えやすい姿勢をとれるような軌道が生成できることを示した. つぎに手先の位置と台車の位置を同時に制御する動的軌道制御法について述べ, それを拡張することによって手先の位置と力および台車の位置を同時に制御する動的ハイブリッド制御法を導出した. さらに軌道計画法によって生成した軌道を用いて実験を行い, 軌道計画法の有効性ならびに動的ハイブリッド制御法の有効性を示した.



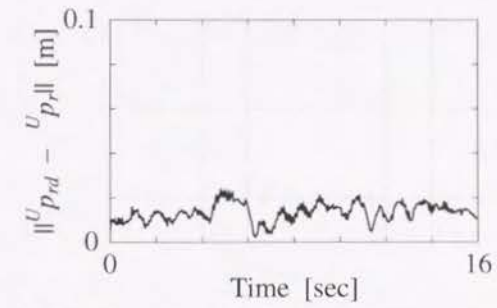
(a) Tracking error of reference point



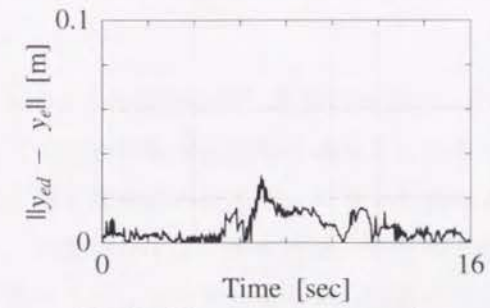
(b) Tracking error of armtip



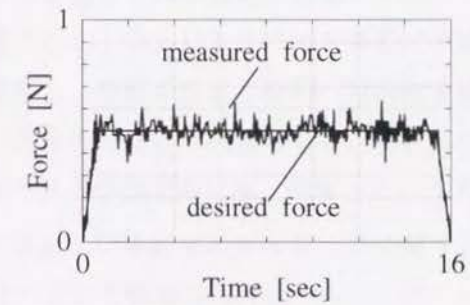
(c) Measured force

Fig. 6.9 Experimental result (case 1 : $W_f = 20$)

(a) Tracking error of reference point



(b) Tracking error of armtip



(c) Measured force

Fig. 6.10 Experimental result (case 2 : $W_f = 150$)

第7章

結言

本論文はロボットによる自律的な押し作業の確立，および移動マニピュレータの動的な制御とそのための軌道計画法の確立を目的として行った研究をまとめたものである．押し作業においては，ロボット自体よりも大きな対象物を扱うことを考え，対象物を1点で押す方法，すなわち点接触による押し作業を対象とし，移動マニピュレータによる押し作業の実現を目指した．点接触による押し作業では摩擦などの不確定な要素の存在により，対象物を直進させるだけでも困難となる．さらに1点で押される対象物の運動には非ホロノミックな拘束が存在し，任意の位置と姿勢へ対象物を移動させるためには適切な軌道ないし経路を与えなければならない．これに対し本論文では，摩擦力の推定，対象物の軌道や作業の計画，制御，というアプローチをとった．

第1章では押し作業および移動マニピュレータに関する従来の研究の問題点を指摘し，本論文の目的を明確にした．

第2章では対象物の底面に生じる摩擦力をモデル化するものとして摩擦力分布を定義し，数回の押し操作で摩擦力分布および摩擦中心の位置を推定する方法を提案した．また，摩擦力分布の構成要素である対象物底面の支持点の位置を仮想的に設定する方法について述べ，推定を行うための押し操作のアルゴリズムを示した．さらに，底面形状が矩形である対象物に対して，提

案する推定法が有効であることを数値例および実験により示した。摩擦係数分布は、対象物底面と床面との接触状態が変化しないという仮定の下では、対象物に固有のものとなる。これにより、押し動作により生じる対象物の運動は一意に定まり、さらにその運動を摩擦係数分布より求めることが可能となる。

第3章では摩擦係数分布が変動しないという仮定の下に、押し作業における対象物の軌道を摩擦係数分布を用いて計画する手法を提案した。さらにその手法を拡張することにより障害物回避も含めた軌道計画を行う手法を提案した。この手法により、対象物を1本のアームで押し移動させる際、押す点を変更せずに連続的な動作で対象物を目標の位置と姿勢まで移動させることができる対象物の軌道を計画できること、さらに、障害物が存在してもそれらに対象物が接触しない軌道を生成できることを数値例により示した。

前章で提案した軌道計画法では、対象物を押す点の選び方によっては得られる軌道が非常に長い経路であったり、環境の制限により目標点に到達する軌道が得られない場合も考えられる。これに対し第4章では、対象物から見た押す点の位置と速度の方向を一定とすることにより、対象物の移動経路を直線経路と円弧経路に限定し、それらの経路を与える押し操作を複数組み合わせることで対象物を目標とする位置と姿勢まで移動させる作業を摩擦係数分布を用いて計画する手法を提案した。さらにその有効性を数値例で示した。

第3章と第4章では摩擦係数分布が変動しないという仮定を設けていたが、実環境においては対象物の移動に伴い摩擦係数分布は変化すると考えられるため、押し作業を実現するためには計画通りに作業を遂行するための制御が必要となる。第5章では1点で押される対象物を近似的に独立二輪駆動型移動ロボットと同様に扱うことができる点、擬似中心の存在を示し、擬似中心を利用することにより、対象物を第3章で提案した計画法により得られる軌道に沿って移動させるための制御法を提案した。さらに押し作業を実現させるための移動マニピュレータの制御則、ならびに目標軌道の与え方について述べ、実機による実験により実環境においても対象物を軌道に沿って移動させることができることを示した。

第6章では加速度の次元での移動マニピュレータの軌道計画法を提案し、手先を壁に押しつけながら移動するような作業において、マニピュレータが力を加えやすい姿勢をとれるような軌道が生成できることを数値例で示した。さらに手先の位置と力および台車の位置を同時に制御するための移動マニピュレータの動的ハイブリッド制御法を提案し、生成された軌道を用いて実験を行うことにより、提案する制御法の有効性、ならびに軌道計画法の有効性を検証した。

今後の研究課題としては、以下に示す1)押し作業の斜面での適用、2)対象物の動特性を考慮した押し作業の計画と制御、などが挙げられる。

本論文では床面が水平であることを前提としているが、床面が傾いている場合は重力の影響により、対象物の姿勢によって摩擦係数分布が変化すると考えられ、第3章、第4章で提案した手法と同じ枠組みで計画を行うことはできなくなる。しかしながら運搬作業においては斜面を登らなければならない状況も考えられるため、より多様な作業に押し作業を導入するためには、1)のような拡張が必要と思われる。

さらに本論文では対象物の慣性力は摩擦係数に対して無視できるという仮定の下に準静力学的な展開を行ったが、より迅速な押し作業を行う際は対象物の動特性を考慮する必要があると考えられる。したがって2)で示す展開が今後必要となる。また動的な押し作業を移動マニピュレータに行わせるためには、それを実現するための移動マニピュレータの軌道設計や制御則が必要となり、第6章で示した軌道計画法と制御法に拡張を加えなければならない。

参考文献

- [1] W.D.MacMillan, "*Dynamics of Rigid Bodies*," New York: Dover, 1936.
- [2] M. T. Mason, "Mechanics and Planning of Manipulator Pushing Operations," *The International Journal of Robotics Research*, Vol. 5, No. 3, pp. 53-71, 1986.
- [3] M. A. Peshkin and A. C. Sanderson, "The Motion of a Pushed, Sliding Workpiece," *IEEE Journal of Robotics and Automation*, Vol. 4, No. 6, pp. 569-598, 1988.
- [4] J. C. Alexander and J. H. Maddocks, "Bounds on the Friction-Dominated Motion of a Pushed Object," *The International Journal of Robotics Research*, Vol. 12, No. 3, pp. 231-248, 1993.
- [5] K. M. Lynch, "The Mechanics of Fine Manipulation by Pushing," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 2269-2276, 1992.
- [6] D. T. Pham and S. H. Yeo, "Initial Motion of a Rectangular Object being Pushed or Pulled," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 1046-1050, 1990.

- [7] S. Goyal, A. Ruina, and J. Papadoulos, "Limit Surface and Moment Function Description of Planar Sliding," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 794-799, 1989.
- [8] S. H. Lee and M. R. Cutkosky, "Fixture Planning With Friction," *ASME Journal of Engineering for Industry*, Vol. 113, pp. 320-327, 1991.
- [9] M. T. Mason, "How to Push a Block Along a Wall," *NASA Conference on Space Telerobotics, Pasadena*, pp. 26-38, 1989.
- [10] H. Mayeda and Y. Wakatsuki, "Strategies for Pushing a 3D Block along a Wall," *Proceedings '91 IEEE/RSJ International Workshop on Intelligent Robots and Systems*, pp. 461-466, 1991.
- [11] W. H. Hung, E. P. Krotkov, and M. T. Mason, "Impulsive Manipulation," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 120-125, 1995.
- [12] Z. Balorda, "Reducing Uncertainty of Objects by Robot Pushing," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 1051-1056, 1990.
- [13] Z. Balorda, "Automatic Planning of Robot Pushing Operations," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 732-737, 1993.
- [14] S. Akella and M. T. Mason, "Posing Polygonal Object in the Plane by Pushing," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 2255-2262, 1992.
- [15] K. M. Lynch and M. T. Mason, "Controllability of Pushing," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 112-119, 1995.

- [16] R. C. Brost, "Automatic Grasp Planing in the Presence of Uncertainty," *The International Journal of Robotics Research*, Vol. 7, No. 1, pp. 3-17, 1988.
- [17] A. S. Rao and K. Y. Goldberg, "Orienting Generalized Polygonal Parts," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 2263-2268, 1992.
- [18] K. Y. Goldberg, "Orienting Polygonal Parts without Sensors," *Algorithmica*, Vol. 10, pp. 201-225, 1993.
- [19] S. Akella and M. T. Mason, "Parts Orienting by Push-aligning," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 414-420, 1995.
- [20] M. A. Peshkin and A. C. Sanderson, "Planning Robotic Manipulation Strategies for Workpieces that Slide," *IEEE Journal of Robotics and Automation*, Vol. 4, No. 5, pp. 524-531, 1988.
- [21] M. Brokowski, M. Peshkin, and K. Goldberg, "Curved Fences for Part Alignment," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 467-473, 1993.
- [22] 長田, 山浦, "力覚センサを用いたマニピュレータによる押し操作の研究," 第29回SICE学術講演会予稿集, pp. 483-484, 1990.
- [23] 沢田, 稲葉, 井上, "ビジュアルフィードバックを用いたマニピュレータによる押し操作の研究," 第5回日本ロボット学会学術講演会予稿集, pp. 599-600, 1987.
- [24] F. Gandolfo, M. Tistarelli, and G. Sandini, "Visual Monitoring of Robot Actions," *Proceedings '91 IEEE/RSJ International Workshop on Intelligent Robots and Systems*, pp. 269-275, 1991.

- [25] P. Franchi, F. Gandolfo, and et al., "Preliminary Experiments of Visuo-Motor Integration in Pushing Tasks," *Proceedings '91 IEEE/RSJ International Workshop on Intelligent Robots and Systems*, pp. 535-537, 1991.
- [26] K. M. Lynch, H. Maekawa, and K. Tanie, "Manipulation and Active Sensing by Pushing Using Tactile Feedback," *Proceedings '92 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1992.
- [27] S. Takagi and Y. Okawa, "Rule-Based Control of a Mobile Robot for the Push-a-Box Operation," *Proceedings IEEE/RSJ International Workshop on Intelligent Robots and Systems*, pp. 1338-1343, 1991.
- [28] Y. Okawa and K. Yokoyama, "Control of a Mobile Robot for the Push-a-Box Operation," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 761-766, 1992.
- [29] 大川, 横山, "移動ロボットによる箱押し作業の制御," 日本ロボット学会誌, Vol. 11, No. 1, pp. 148-155, 1993.
- [30] 星谷, 井上, 大川, "2台の移動ロボットによる荷物押し作業の制御," 第12回日本ロボット学会学術講演予稿集, pp. 363-364, 1994.
- [31] 寺崎, 長谷川, "平行2指ハンドによる滑らし操作を利用した知的物体操作のための動作計画," 日本ロボット学会誌, Vol. 12, No. 7, pp. 138-147, 1994.
- [32] 寺崎, 長谷川, "指先に回転機構を有する平行2指ハンドによる知的物体操作のための動作計画," ロボティクス・メカトロニクス講演会'95 講演論文集, 第B巻, pp. 1351-1354, 1995.

- [33] P. C. Chen and Y. K. Hwang, "Practical Path Planning among Movable Obstacles," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 444-449, 1991.
- [34] 見浪, 藤原, 拓植, "自律移動ロボット搭載型マニピュレータの位置・姿勢制御," 日本ロボット学会誌, Vol. 11, No. 1, pp. 156-164, 1993.
- [35] 見浪, 藤原, 拓植, "カメラフィードバックを用いた自律移動マニピュレータの位置・速度制御," 日本ロボット学会誌, Vol. 11, No. 2, pp. 93-101, 1993.
- [36] S. Dubowsky and A. B. Tanner, "A study of the Dynamics and Control of Mobile Manipulators Subjected to Vehicle Disturbances," *Proceedings IV International Symposium of Robotics Research*, 1987.
- [37] N. A. M. Hootsmans and S. Dubowsky, "Large Motion Control of Mobile Manipulators Including Vehicle Suspension Characteristics," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 2336-2341, 1991.
- [38] 藤原, 見浪, 拓植, "車輪型移動マニピュレータの非線形補償," 第10回日本ロボット学会学術講演会予稿集, pp. 525-528, 1992.
- [39] 藤原, 見浪, 西山, "斜面を走行する移動マニピュレータの重力補償," 第10回日本ロボット学会学術講演予稿集, pp. 529-532, 1992.
- [40] K. Liu, "Decentralized Continuous Robust Controller for Mobile Robots," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 1822-1827, 1990.
- [41] Y. Yamamoto and X. Yun, "Coordinating Locomotion and Manipulation of a Mobile Manipulator," *Proceedings of the 31st IEEE Conference on Decision and Control*, pp. 1-6, 1992.

- [42] Y. Yamamoto and X. Yun, "Modeling and Compensation of the Dynamic Interaction of a Mobile Manipulator," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 2187-2192, 1994.
- [43] H. Seraji, "An On-line Approach to Coordinated Mobility and Manipulation," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 28-35, 1993.
- [44] H. Seraji, "Configuration Control of Rover-Mounted Manipulators," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 2261-2266, 1995.
- [45] W. Miksch and D. Schroeder, "Performance-Function Based Controller Design for a Mobile Manipulator," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 227-232, 1992.
- [46] 森, 大隅, 小橋, 新井, "移動マニピュレータの軌道計画," 第11回日本ロボット学会学術講演会予稿集, pp. 1077-1078, 1993.
- [47] F. G. Pin and J. C. Culioli, "Multi-Criteria Position and Configuration Optimization for Redundant Platform/Manipulator Systems," *Proceedings IEEE/RSJ International Workshop on Intelligent Robots and Systems*, pp. 103-107, 1990.
- [48] W. F. Carriker, P. K. Khosla, and B. H. Krogh, "Path Planning for Mobile Manipulators for Multiple Task Execution," *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 3, pp. 403-408, 1991.
- [49] C.-C. Wang and V. Kumar, "Velocity Control of Mobile Manipulators," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 204-209, 1990.

- [50] 藤沢, 福田, その他, "マンロボット協調作業型マニピュレータの基礎的研究-重心制御によるマニピュレータ/ビークルシステムの制御-, " 第10回日本ロボット学会学術講演予稿集, pp. 537-538, 1992.
- [51] 黄, 菅野, 加藤, "ヴィークル搭載型マニピュレータの安定化と制御," -安定規範とマニピュレータによる補償運動-, Vol. 31, No. 7, pp. 861-870, 1995.
- [52] Y. Yamamoto and X. Yun, "Control of Mobile Manipulators Following a Moving Surface," *Proceedings IEEE International Conference on Robotics and Automation*, pp. 1-6, 1993.
- [53] K. Nagatani and S. Yuta, "Design a Behavior to Open a Door and to Pass Through a Door-way using a Mobile Robot Equipped with a Manipulator," *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 847-853, 1994.
- [54] 吉川, "ロボット制御基礎論," コロナ社, 1988.
- [55] K. M. Lynch, "Estimation of the Friction Parameters of Pushed Objects," *Proceedings '93 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 186-193, 1993.
- [56] 嘉納, "システムの最適理論と最適化," コロナ社, 1987.
- [57] V. H. Quintana and E. J. Davison, "Clipping-off Gradient Algorithms to Compute Optimal Controls with Constrained Magnitude," *The International Journal of Control*, Vol. 20, No. 2, pp. 243-255, 1974.
- [58] 亀山, 佐山, "状態制約を受ける最適制御問題に対するペナルティ法," 計測自動制御学会論文集, Vol. 10, No. 3, pp. 272-277, 1974.

- [59] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi, "A Stable Tracking Control Method for a Non-Holonomic Mobile Robot," *Proceedings '91 IEEE/RSJ International Workshop on Intelligent Robots and Systems*, pp. 1236-1241, 1991.
- [60] X. Yun and Y. Yamamoto, "Internal Dynamics of a Wheeled Mobile Robot," *Proceedings '93 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1288-1294, 1993.

研究発表

発表論文

1. T. Yoshikawa and M. Kurisu, "Identification of the Center of the Friction from Pushing an Object by a Mobile Robot", Proc. of 1991 IEEE/RSJ International Workshop on Intelligent Robots and Systems, pp.449-454, 1991.
2. 吉川, 栗栖, "対象物押し操作にもとづく摩擦力分布の推定法", 日本ロボット学会誌, Vol.10, No.5, pp.632-638, 1992.
3. M. Kurisu and T. Yoshikawa, "Trajectory Planning for an Object in Pushing Operation", Proc. of 1994 Japan-U.S.A. Symposium on Flexible Automation, pp.1009-1016, 1994.
4. 栗栖, 吉川, "押し作業における対象物の軌道計画", 日本ロボット学会誌, Vol.13, No.8, pp.1115-1121, 1995.
5. 吉川, 栗栖, 水野, "ロボットによる対象物の押し作業の最適計画", 日本ロボット学会誌(投稿中).
6. 栗栖, 吉川, "移動マニピュレータの軌道計画と制御", 日本機械学会誌(投稿中).
7. 栗栖, 吉川, "押し作業における対象物の軌道追従制御", 日本ロボット学会誌(投稿中).

口頭発表(*発表者)

1. 吉川,*栗栖,“移動ロボットによる対象物押し作業における摩擦中心位置の同定法”,第35回システム制御情報学会研究発表講演会,京都私学会館,1991年5月.
2. *吉川,栗栖,“押し作業における対象物の軌道計画”,第37回システム制御情報学会研究発表講演会,大阪国際交流センター,1993年5月.
3. 吉川,栗栖,*水野,“マニピュレータによる押し操作を利用した床面と対象物間の摩擦力分布の推定”,第4回インテリジェントFAシンポジウム,メルパルク OSAKA,1993年7月.
4. 吉川,栗栖,*川井,“マニピュレータを搭載した車輪型移動体の軌道計画と制御”,日本ロボット学会第4回ロボットシンポジウム,工学院大学,1994年5月.
5. *栗栖,吉川,“マニピュレータを搭載した独立二輪駆動型移動ロボットの軌道計画と制御”,第12回日本ロボット学会学術講演会,九州大学工学部,1994年11月.
6. *栗栖,吉川,“移動マニピュレータの軌道計画と動的ハイブリット制御”,日本機械学会ロボティクス・メカトロニクス演会'95,川崎市産業振興会館,1995年6月.