

- 博士論文 -

リニアアクチュエータアレイによる分散マ
ニピュレーション法の新しい制御論理に関
する研究

Development of Novel Control Logic for Distributed
Manipulation by Linear Actuator Array

指導教官：堀 洋一 教授

東京大学大学院 工学系研究科 電気工学専攻

97112 及部 七郎斎

2002年12月

内容梗概

本論文は、近年注目を集めている新しい物体の操作法である分散マニピュレーション法をリニアアクチュエータを、アレイ状に並べ実現する場合の一般的な制御論的枠組みを提案する。

分散マニピュレーションは多数の支点を通じて対象物に作用し操作する方法で、多数のアクチュエータで実現される。近年のマイクロマシン技術により多数のアクチュエータが簡単で、低コストで実現できるようになり注目されてきた新しく興味深い概念である。このマニピュレーション法は、きめ細かなマニピュレーションが可能、耐故障性の高いマニピュレーションが可能、パワフルなマニピュレーションが可能の特長を持つ。

従来、この分散マニピュレーションはマイクロマシン技術の一つとして注目されていたためか、いかにこの分散マニピュレーションを実現するデバイスを製作するかということを中心に研究が行われてきた。また、分散マニピュレーションという概念自体が新しいため、体系的な研究はほとんどなく、体系的な研究が強く必要とされている。

これに対して、本論文では、分散マニピュレーションの基本三動作を、『つかむ』『計る』『運ぶ』と定義し、いかに作るのではなく、いかに使うかという新しい視点から、制御工学の知見をふんだんに用い、それぞれの動作の実現を目指す。

『計る』では外乱オブザーバアレイの提案し、これにより、分散マニピュレーションにおいて必ず必要な、マニピュレーション対象物の情報、質量、重心位置、慣性モーメントの推定が可能になる。

『つかむ』では面インピーダンス制御を提案し、可変ホームポジション構造からマニピュレーション対象物の形状に合わせた把持、荷重分散的な把持を実現する。また、インピーダンスパラメータを時変させることにより、加速度の時間微分値を抑制し、衝撃吸収効果のある面を実現する。

『運ぶ』では自律分散システムの視点を導入し、各アクチュエータの自律性による局所的な制御とアクチュエータ群全体を制御する集中系の二つを組み合わせた自律分散集中複合系という、ボトムアップとトップダウンの双方向アルゴリズムの提案を行う。これによりマニピュレーション対象物の搬送を可能にする。

そして、さらに最後では、分散マニピュレーションの基本三動作『計る』『つかむ』『運ぶ』を、有機的に結合し、『計りながらつかむ』、『つかみながら運ぶ』そして『計りながらつかみ、そして運ぶ』という複合動作を実現する。

目次

第 1 章	序論	1
1.1	分散マニピュレーションの定義	1
1.2	分散マニピュレーションをめぐる研究状況	4
1.2.1	研究の分類方法	4
1.2.2	トップダウン方式	5
1.2.3	ボトムアップ方式	17
1.2.4	両方からのアプローチ - ボトムアップとトップダウンそれぞれから -	19
1.2.5	まとめ	26
1.3	本論文の構成 - 分散マニピュレーション研究へのアプローチ -	27
1.3.1	分散マニピュレーションの基本三動作『計る』『つかむ』『運ぶ』	27
1.3.2	本論文題目に関する考察	29
第 2 章	分散マニピュレーション基本要件	32
2.1	まえがき	32
2.2	空間周波数による制約	33
2.3	力密度による制限	35
2.4	アクチュエータの基本構造	36
2.5	無駄時間による制限	37
2.6	アルゴリズムのタイプ	39
2.7	分散マニピュレーションシステムに関する幾つかの現実的な仮定	39
2.8	あとがき	42
第 3 章	計る	43
3.1	まえがき	43
3.2	対象物の力学的性質	43
3.3	外乱オブザーバによる外乱推定とそのアレイ化	44
3.4	外乱分布から質量、重心位置、慣性モーメントの推定	45
3.5	x 軸、 y 軸周りの慣性モーメント	47
3.6	推定誤差	48
3.6.1	アクチュエータ間隔が有限に伴う誤差	49
3.6.2	外乱オブザーバの推定誤差の影響	50
3.7	シミュレーションと実験結果	51
3.8	あとがき	52
第 4 章	つかむ	57
4.1	まえがき	57
4.2	面インピーダンス制御の提案 - インピーダンス制御基本式の導出 -	57
4.3	荷重分散	59
4.3.1	可変ホームポジション決定式の導出	59
4.3.2	可変ホームポジション (2次元への拡張)	62
4.3.3	シミュレーションと実験	66

第 0 章 目次

4.4	衝撃吸収効果 [24], [25]	69
4.4.1	Jerk の 2 乗積分最小化による衝撃吸収	69
4.4.2	インピーダンスパラメータの設計 ($m_d = 0$ の場合)	76
4.4.3	インピーダンスパラメータの設計 ($m_d \neq 0$ の場合)	77
4.5	あとがき	81
4.5.1	荷重分散	82
4.5.2	衝撃吸収効果	83
第 5 章	運ぶ	84
5.1	まえがき	84
5.2	物体搬送システムモデル	84
5.3	場の概念と自律分散集中複合系	85
5.4	フィードフォワードによる方法	87
5.4.1	モジュール群の振る舞いと境界条件の設定	87
5.4.2	シミュレーションと実験結果	89
5.5	フィードバックによる方法	90
5.5.1	モジュール群の振る舞いと境界条件の設定	95
5.5.2	負荷外乱を用いた搬送物体の位置の推定	96
5.5.3	搬送物体のフィードバック位置制御コントローラ的设计	97
5.5.4	シミュレーション結果	98
5.6	分散マニピュレーション基本要件の無駄時間制限	98
5.6.1	無駄時間による不安定化	98
5.6.2	スミス法を代表とする無駄時間補償器	101
5.6.3	スミス法の外乱補償	104
5.6.4	無駄時間補償器による安定化	106
5.7	あとがき	108
第 6 章	各動作の有機的な結合	109
6.1	まえがき	109
6.2	計りながらつかむ - 局所的な荷重分散から大域的な荷重分散 -	110
6.2.1	つかむ動作単独の場合の問題点	110
6.2.2	対象物の力学的な安定と荷重分散	112
6.2.3	局所的荷重分散領域作成法	116
6.2.4	シミュレーション結果	118
6.3	つかみながら運ぶ - 搬送方向だけ大域的荷重分散モードの解放 -	121
6.3.1	大域的荷重分散の制御目標の拡張	121
6.3.2	シミュレーション結果	124
6.4	計りながらつかみ、そして運ぶ	130
6.5	あとがき	130
第 7 章	結論	136
第 8 章	今後の展望	145
	謝辞	147
付録 A	可変ホームポジション (荷重分散の数理的な意味) - 『つかむ』 -	149
付録 B	製作した実験装置	151
	研究業績	157

第 1 章

序論

1.1 分散マニピュレーションの定義

分散マニピュレーションは [1][†] 『多数のアクチュエータにより多数の支点を通じて対象物に作用し操作する方法』と定義される。この言葉は 1999 年のロボットの分野の最も権威ある国際会議のうちの一つである The IEEE International Conference on Robotics and Automation(ICRA) で分散マニピュレーションのワークショップが開かれたことにより、制御分野でもメジャーとなった。

Fig.1.1 のように多数の人がある人を胸上げをして搬送や回転操作をしているところはまさに分散マニピュレーションである。この手法により物体搬送をはじめ物体を操作する試みが様々なところで考えられている。

このマニピュレーション法は、

1. 多数のアクチュエータで作用するため、きめ細かな操作が可能である (きめ細かなマニピュレーション)。
2. 少数のアクチュエータの故障はマニピュレーションに問題を及ぼさない (耐故障性のあるマニピュレーション)。
3. 個々が低出力のアクチュエータであっても、多数の支点で支えるため、大型で重い物体のマニピュレーションが可能である (パワフルマニピュレーション)。

の特長をもっている。

ハワイのマウナケア火山にある光学望遠鏡『すばる』は、世界最大級の反射鏡を持ち、その反射鏡は多数のアクチュエータにより、多数の支点を通じて支えられている。反射鏡に歪みが出たときはその多数のアクチュエータできめ細かな歪み補正を行っている [2], [3], [4]。これは、分散マニピュレーションのきめ細かなマニピュレーションの利点を利用している。文献 [5], [6],[7], [8] の Autonomous Distributed Micromachines (ADM) は LSI などの半導体

[†]Distributed manipulation effects motion on objects through a large number of points of contact.



Fig.1.1: 分散マニピュレーションの概念を的確に示す写真

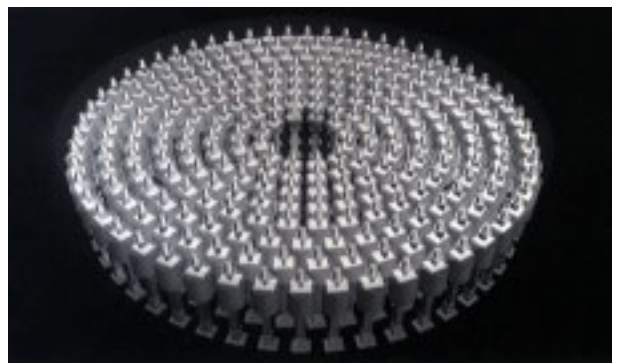


Fig.1.2: きめ細かなマニピュレーションの一例としてハワイのマウナケア火山にある光学望遠鏡すばる

加工技術を用いて、多数のマイクロアクチュエータを製作し、そのアクチュエータで相対的に巨視的なタスクを実現する装置である。Fig.1.3 にその概念図を示す。シリコンウェハ上にアクチュエータ、マイクロプロセッサ、センサを集積し、それらにより物体を搬送する。個々のアクチュエータセルは自分でマイクロプロセッサなどを持っており、自分で自分のフィードバック制御ができる。また、そのマイクロプロセッサで自律しており、隣接する他のアクチュエータセルと協力することにより分散マニピュレーションを可能とする。

各アクチュエータが自律することにより、いくらかのアクチュエータが故障した場合でも物体搬送に影響を及ぼさないアルゴリズムを提案し、耐故障性のあるマニピュレーションを実現している。

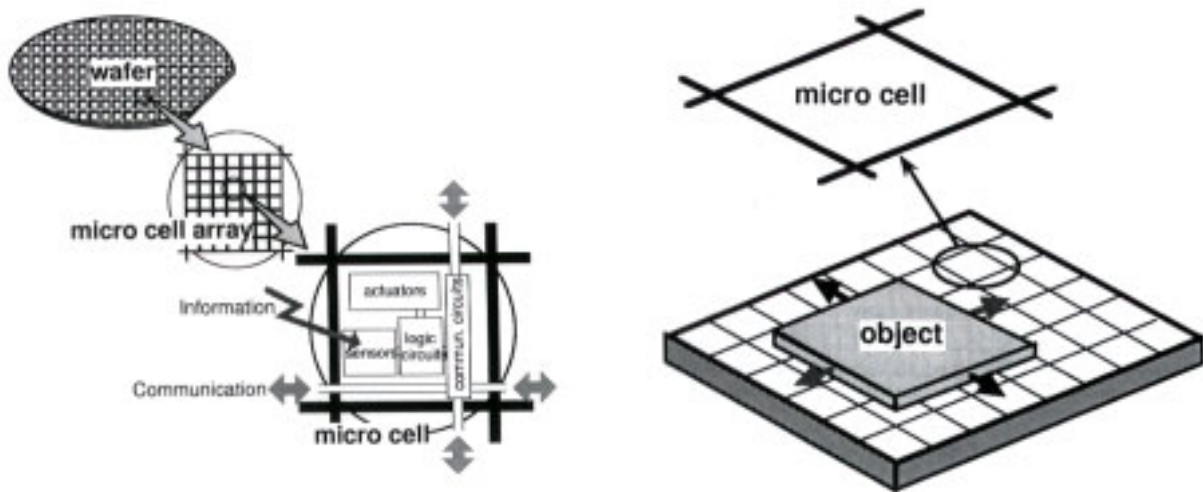


Fig.1.3: 耐故障性の高いマニピュレーションの一例として ADM

文献 [9] は多数のアクチュエータで自らを支えるマイクロロボットの試作報告論文である。自重の 30 倍程度重さの物体の搬送が可能だと報告している (Fig.1.4)。個々のアクチュエータは非力であるが、多数のアクチュエータで操作する、分散マニピュレーションならではの、これはパワフルマニピュレーションの一例である。

この分散マニピュレーションのコンセプトは、言い換えれば、高出力のアクチュエータを一つ用意するのではなく、低出力でシンプルな構造のアクチュエータを複数用意しそれらを複合し高出力を得る方法である。一般的に考えれば、低出力なアクチュエータを複数用意する方が高出力のアクチュエータを一つ用意するよりコスト的には割高である。最近なぜこのようなコンセプトが注目されるようになったかといえば、やはりマイクロマシン技術により多数のアクチュエータの集積が低コストで簡単に可能となったからだろう。

この分散マニピュレーションの研究は始まったばかりである。すばるの例にあるように、コスト的に割高でも分散マニピュレーションでしかできないことはやればいい。良くも悪くも、この分散マニピュレーション法がどのような特徴を持ち、何が可能で、何が不可能

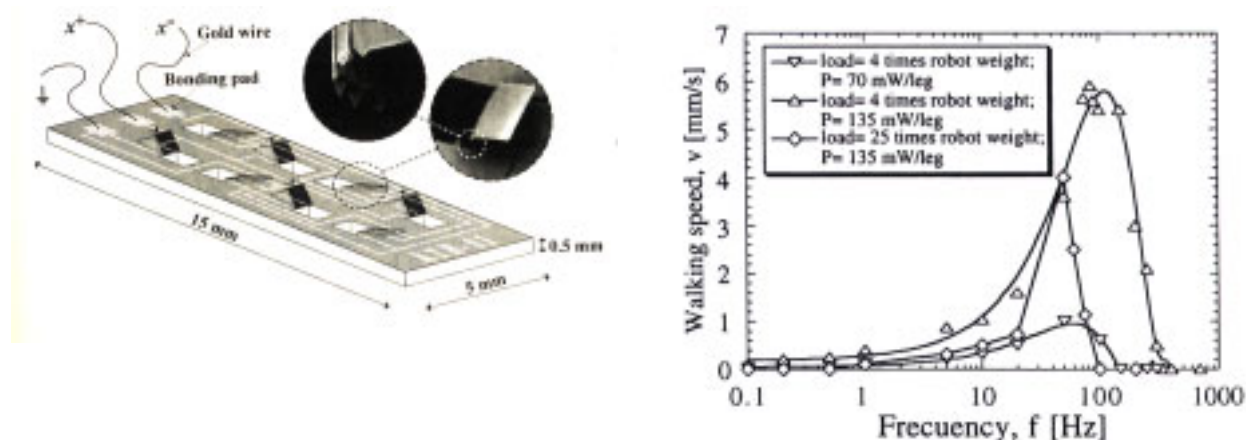


Fig.1.4: パワフルマニピュレーションの一例としてのマイクロウォーキングロボット

なのかを検証する段階にあるといえる。

多くの分散マニピュレーションの研究は、そもそも分散マニピュレーションの概念がマイクロマシン技術により安く、高密度のアクチュエータが製作できるようになったから始まったという経緯からか、マイクロマシンのデバイスづくりの視点から、いかに作るかに重点が置かれている。制御に携わる人間として、この分散マニピュレーションを制御の視点から、いかに分散マニピュレーションデバイスを使うか、制御をするにおいて何が問題となるかを検証する。

1.2 分散マニピュレーションをめぐる研究状況

1.2.1 研究の分類方法

研究が始まったばかりのこの分散マニピュレーションの研究動向に関して説明を行う。少ないながらも様々な研究が行われているわけであるが、様々な仮定やアプローチがあるものの、その動向から分散マニピュレーション研究の一般的なあり方などを分析することが本節の目的である。

研究の分類方法として、アルゴリズムの構造の違いに注目し、ボトムアップ、トップダウン方式という2つの分類から紹介を行う。

本節では、ボトムアップ方式、トップダウン方式という言葉を用いる。曖昧な言葉であるので、前もって使い方について示し定義しておく。

分散マニピュレーションはアクチュエータセル単体とシステムを構成する多数のアクチュエータセルの二つのレベルがある。どのレベルでルールを決め、システムを起動するかでボトムアップ方式とトップダウン方式の2つの方式による分類が成り立つ。

- ボトムアップ方式

ボトムアップ方式は個々のアクチュエータセルであるルールを決めておく。そのルールが多数のアクチュエータセルで構成される全体に効果を及ぼす。例えば、この部屋にいる人は人に殴られたら必ず、誰かれ構わず周りの人を殴り八つ当たりをするというルールを持っているとしよう。その場合、何らかのきっかけで誰かがある人を殴ってしまったら、少しの時間の後、この部屋では大乱闘が繰り広げられることになる（ケンカはやめましょう！）。

- トップダウン方式

トップダウン方式はボトムアップ方式の逆であり、多数のアクチュエータでの全体のタスクが何より先に決定される。

1.2.2 トップダウン方式

トップダウン方式による分散マニピュレーションへのアプローチとして、既に紹介した国立天文台の光学望遠鏡の鏡支持装置のPMA(Primary Mirror Actuator)の詳細、MDMS(Modular Distributed Manipulator System)、PVF(Programmable Vector Fields)を紹介する。MDMSでは最終的に搬送物体の位置制御を実現している。また、PVFでは力場中での搬送物体の安定状態を解析し、どのような力場が搬送に適しているか考察している。

すばるの鏡支持装置 PMA

ハワイのマウナ火山に日本の国立天文台ハワイ観測所がある。そこには『すばる』と名付けられた直径8[m]を超える超大型反射鏡を擁する光学望遠鏡がある。光学天体望遠鏡は遠くの天体を観測するために、天体から放射、反射される弱い光を高密度に集光する必要があり、大型の反射鏡が必要であった。集光している様子を Fig.1.5 に示す。しかし大型の反射鏡は自身の自重で歪んでしまい、集光が上手くできず、Fig.1.6 に示すようになってしまう。そのため、今までは10[m]を超える超大型反射鏡はその自重から構造上実現不可能と考えられていた。

しかしながらここで逆の発想で、むしろ反射鏡を薄く軽くひずみやすい構造にして、Fig.1.7 にあるような多数のアクチュエータで支え、積極的に各アクチュエータで歪みを補正すれば、大型で歪みのない反射鏡が作れる。個々のアクチュエータの構造は Figs.1.8, 1.9 に示す。ここでポイントとなるのが多数のアクチュエータを用いて多数の支点を通じて反射鏡を面で支え、補正を行っている点である。まさに分散マニピュレーションである。

Fig.1.10 は鏡を支える PWA システムのブロック線図である。この鏡の鏡面形状を保持する制御ループは下記 1、2 の二重の力制御のループと、更にその外側に形成される波面補正ループ、下記 3 の三重の構成となっている。

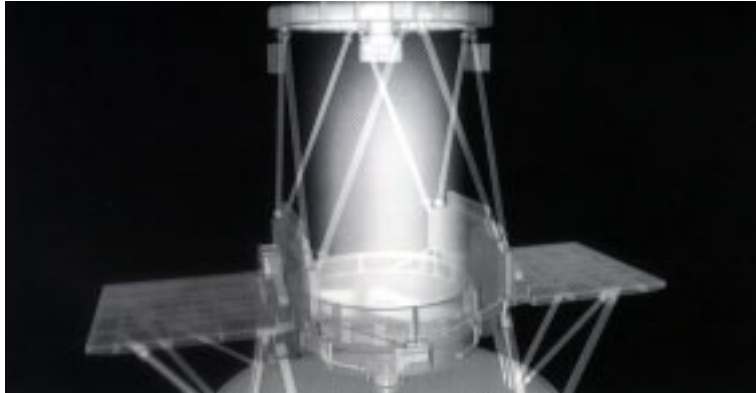


Fig.1.5: 光学望遠鏡のイメージ図。光学望遠鏡は反射鏡により光を集め、遠い天体を観測する。



Fig.1.6: 反射鏡の歪みにより、光が集光できず乱されている様子

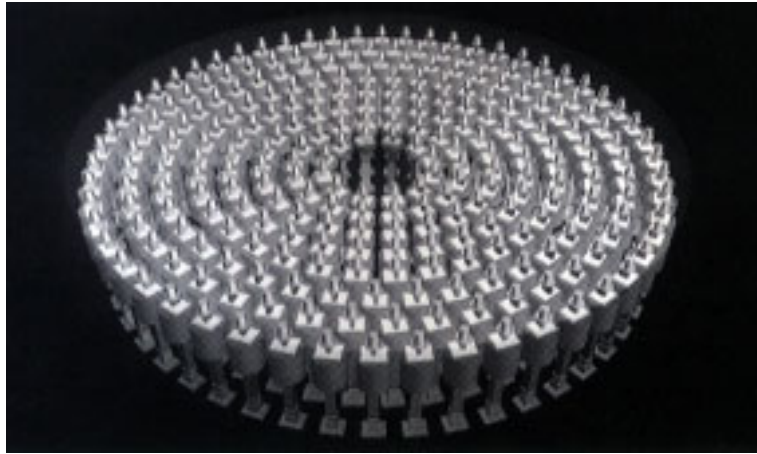


Fig.1.7: 反射鏡を支えるアクチュエータ群



Fig.1.8: すばるの反射鏡を支えるアクチュエータ単体その1

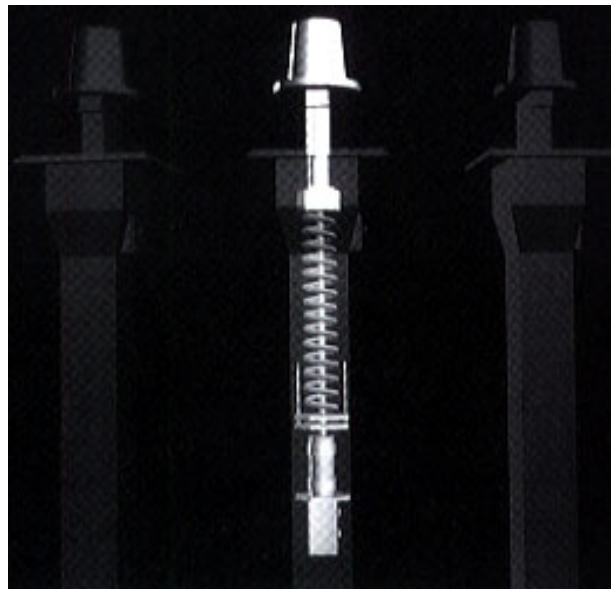


Fig.1.9: すばるの反射鏡を支えるアクチュエータ単体その2

1. PMA の個々に配置された局所的なアクチュエータ自身の力制御ループ
2. 鏡の外縁付近の 3 カ所の固定点機構に装備された力センサ（固定点反力センサ）の検出値を基に PMA 全体への力配分値を補正する固定点反力フィードバックループ
3. 入射光をモニタして波面誤差を検出し、それに基づいて波面形状を補正する光学的波面センサーフィードバックループ

上記 1 は各 PMA ごとの鏡面支持力を常時正確に発生・維持するものである。各点の支持力指令値は、望遠鏡の高度角に応じ、さらに個々のアクチュエータの取り付け位置や特性に応じた補正を加味して上位のコントローラから与えられる。すなわち、分散マニピュレーション的鏡支持を実現する根幹部分である。

上記 2 は、風圧などの外乱力による鏡面変形への対応である。PMA は鏡と弱く結合しているため、風圧荷重などの反力が作用してもそれに対する反力を生成することがない。したがって、風圧荷重に対する反力は、比較的剛性の高い固定点機構部に発生する。

上記 3 が、本研究をトップダウン方式に分類した理由で、様々な環境変動に対しても高い鏡面精度を維持する手段として、天体からの光を用いて鏡面形状を検出・補正するループが 3 である。

ジャック・ハルトマン方式による波面センサによって、鏡面のひずみが波面収差として検知される。その波面収差から鏡面誤差が決定され、その情報に基づいて鏡面形状を補正する。

すなわち、鏡面誤差を小さくするように各アクチュエータ PMA の指令値が決定される、トップダウン方式の構造を有している。

Figs.1.11, 1.12 には上記 3 の制御を行った場合と行わない場合との実験結果を示した。Fig.1.11 が制御を行わない場合で、Fig.1.12 が制御を行った場合である。単位は分からないが、鏡面誤差が一桁程度小さくなり、上記 3 の制御の効果の大きさがうかがえる。

MDMS - Modular Distributed Manipulator System - [10], [11], [12]

MDMS はカーネギーメロン大学の研究グループによって開発された分散マニピュレーション型の物体搬送システムである。各アクチュエータセルは x, y 方向にそれぞれ速度制御され回転するホイールがあり、そのアクチュエータセルを Fig.1.13 のように 2 次元平面に多数並べることにより、Fig.1.14 のような自在な物体搬送を実現しようとしている。物体搬送といえば、ベルトコンベアやロボットマニピュレーションによる方法が思いつく。ベルトコンベアの場合は、搬送方向が一方向だから間違えて搬送した物体を取り戻したりするのが難しい。またロボットマニピュレーションによる物体搬送は自由度はあるものの、重

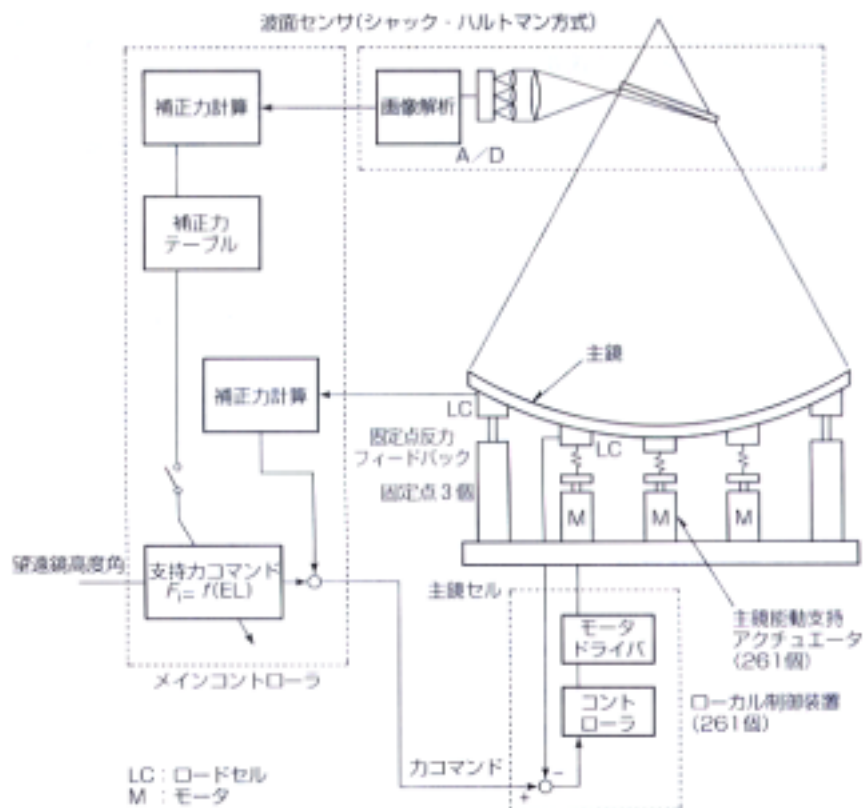


Fig.1.10: すばるのPWAシステムのブロック線図

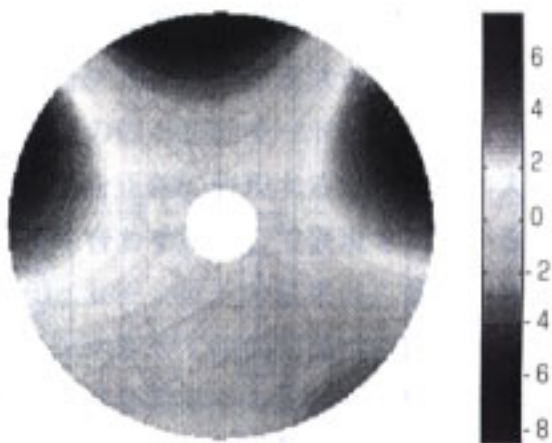


Fig.1.11: 光センサによる補正を行わない場合の鏡表面の歪み

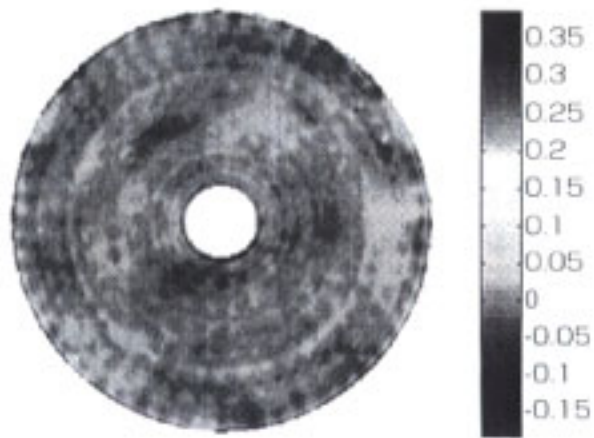


Fig.1.12: 光センサにより補正を行った場合の鏡表面の歪み

いものが運べない、長距離には向いていないなどの問題がある。それを解決する搬送システムの構築が MDMS の目的である。

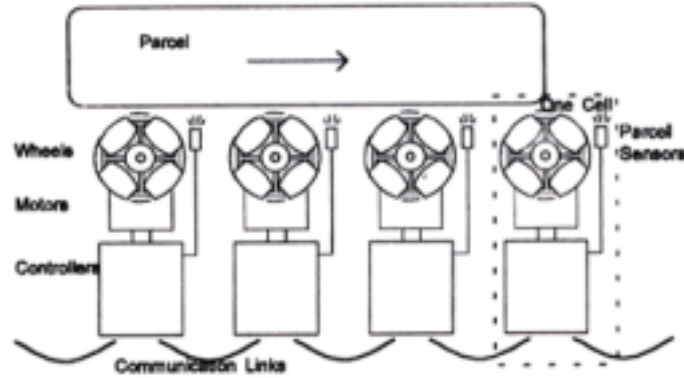


Fig.1.13: MDMS のアクチュエータセル

基礎方程式群 MDMS 上で搬送物体が受ける力 f_x, f_y とトルク τ は、MDMS と対象物との間の干渉モデルを Hooke の法則で決まるようなバネを仮定すると、

$$\begin{aligned} \vec{f} &= [f_x f_y]^T \\ &= \mu \mathbf{V} \vec{N}^T - \mu \dot{\vec{X}}_{cm} W \\ &= \mathbf{k}_s \vec{X}_{cm} + \vec{f}_0 - \mu W \dot{\vec{X}}_{cm} \end{aligned} \quad (1.1)$$

となる。(1.1) はとてもシンプルな形をしているが、MDMS の提案者は論文中でこの関係式を求めたのが自分の大きな仕事の一つであるといっている通り、この関係式を導出するまでは大変な工夫と変形が施されている。

ただし、ここで \mathbf{k}_s は MDMS と搬送物体との間に仮定したバネのバネ定数行列であり、 \vec{X}_{cm} は搬送物体の重心位置、 \vec{f}_0 はオフセットな力である。また、 μ, W はそれぞれ MDMS と搬送物体間の摩擦係数と搬送物体の重量である。

回転方向の力は、 i 番目のセルの出すトルクを τ_i とすると、

$$\begin{aligned} \tau_i &= \mu (\vec{X}_i - \vec{X}_{cm}) \times \left(\vec{V}_i - \dot{\vec{X}}_{cm} + \omega \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} (\vec{X}_i - \vec{X}_{cm}) \right) N_i \\ &= \mu (\vec{X}_i \times \vec{V}_i) N_i - \mu (\vec{X}_{cm} \times \vec{V}_i) N_i + \mu \dot{\vec{X}}_{cm} \times (\vec{X}_i - \vec{X}_{cm}) N_i \end{aligned} \quad (1.2)$$

である。ここで $\vec{X}_i, \vec{V}_i, N_i$ はそれぞれ i 番目のセルの位置、ホイール速度、垂直抗力である。全てのセルから対象物が受けるトルクは i について和を取れば、

$$\tau = \sum_i \tau_i$$

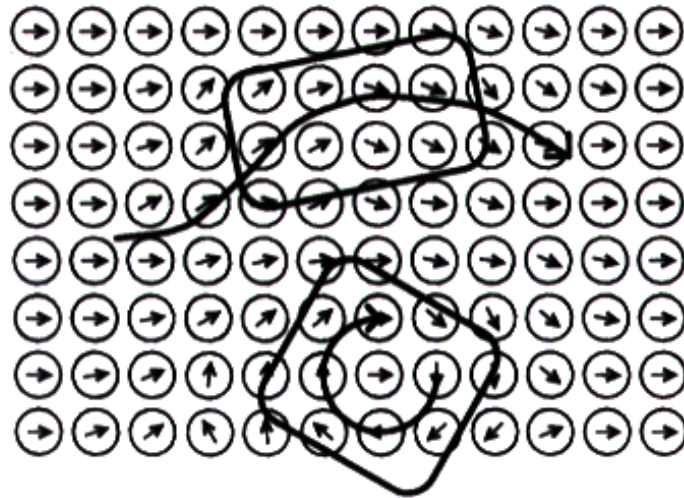


Fig.1.14: MDMS の動作概念図

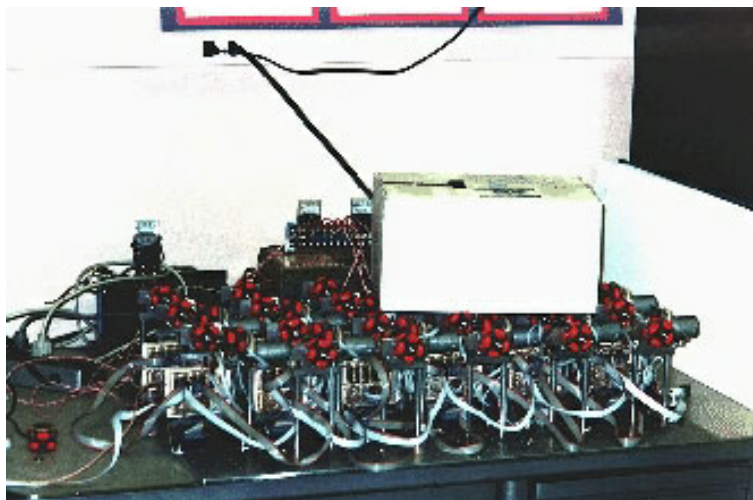


Fig.1.15: MDMS 実験装置の外観

$$\begin{aligned}
 &= \tau_0 + \vec{k}_{sT} \vec{X}_{cm} - (\vec{f}_0 + \mathbf{k}_s \vec{X}_{cm}) \times \vec{X}_{cm} \\
 &\quad - \mu\omega (\vec{\chi} \vec{N}^T - W \vec{X}_{cm}^T \vec{X}_{cm})
 \end{aligned} \tag{1.3}$$

$$\vec{\chi} = [\vec{X}_1^T \vec{X}_1 \dots \vec{X}_i^T \vec{X}_i \dots]^T \tag{1.4}$$

$$\vec{R} = [\vec{X}_1 \times \vec{V}_1 \dots \vec{X}_i \times \vec{V}_i \dots]^T \tag{1.5}$$

となる。

フィードフォワード 搬送物体を目標位置まで運ぶことが MDMS の仕事である。そのために MDMS は搬送物に働く希望の力とトルクを発生させなくてはならない。(1.1) のフィードフォワードの問題は具体的には各セルの位置と k_s, \vec{f}_0 が与えられたとき、各セルのホイールスピードを与える問題に帰着される。 k_s, \vec{f}_0 は、

$$\begin{bmatrix} f_{0x} & k_{sxx} & k_{sxy} \\ f_{0y} & k_{syx} & k_{syy} \end{bmatrix} = \mu W V B^T (B B^T)^{-1} \tag{1.6}$$

$$B = \begin{bmatrix} 1 & \dots & 1 \\ x_1 & \dots & x_n \\ y_1 & \dots & y_n \end{bmatrix} \tag{1.7}$$

で書くことができ、 $V = [\vec{V}_x^T \vec{V}_y^T]^T$ であるから、

$$\begin{bmatrix} f_{0x} \\ k_{sxx} \\ k_{syy} \\ f_{0y} \\ k_{syx} \\ k_{syy} \end{bmatrix} = \mu W \left[\begin{array}{c|c} (B B^T)^{-1} B & \mathbf{0} \\ \hline \mathbf{0} & (B B^T)^{-1} \end{array} \right] \begin{bmatrix} \vec{V}_x^T \\ \vec{V}_y^T \end{bmatrix} \tag{1.8}$$

$$\begin{bmatrix} \vec{V}_x^T \\ \vec{V}_y^T \end{bmatrix} = \frac{1}{\mu W} \left[\begin{array}{c|c} (B B^T)^{-1} B & \mathbf{0} \\ \hline \mathbf{0} & (B B^T)^{-1} B \end{array} \right]^\dagger \begin{bmatrix} f_{0x} \\ k_{sxx} \\ \mathbf{0} \\ f_{0y} \\ \mathbf{0} \\ k_{syy} \end{bmatrix} \tag{1.9}$$

$$= \frac{1}{\mu W} \left[\begin{array}{c|c} \mathbf{B}^T & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{B}^T \end{array} \right] \begin{bmatrix} f_{0x} \\ k_{sxx} \\ 0 \\ f_{0y} \\ 0 \\ k_{syy} \end{bmatrix} \quad (1.10)$$

ただし、ここで非正方行列 A があつたときその Penrose pseudo-inverse $A^\dagger = A^T(AA^T)^{-1}$ である。(1.10) は搬送物体の挙動を決める \vec{f}_0, k_s から各セルのホイール速度が順方向で決定できる関係式となっている。これにより、望み通りの挙動を得るためのホイール速度が決定できる。ここで得られたホイール速度が各セルでのホイール速度指令値となる。

フィードバック MDMS から搬送物体が得る力は (1.1) で与えられた。(1.1) の減衰項を無視すれば、

$$\vec{f} = \vec{f}_0 = \mu W \mathbf{V} \mathbf{B}^T (\mathbf{B} \mathbf{B}^T)^{-1} \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (1.11)$$

で書ける。先のフィードフォワード手法のときと同様の方法で Penrose pseudo の疑似逆行列を用いれば、(1.11) は容易に、

$$\begin{bmatrix} \vec{V}_x \\ \vec{V}_y \end{bmatrix} = \frac{1}{\mu W} \left[\begin{array}{c|c} \mathbf{B}_T & \mathbf{0}_n \\ \hline \mathbf{0}_n & \mathbf{B}_T \end{array} \right] \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_{0x} \\ f_{0y} \end{bmatrix} \quad (1.12)$$

(1.12) を用いれば、任意の力を与える各セルのホイール速度を計算できる。これにより、得られた力 f_x から、搬送物体の運動方程式は下記の (1.13) となる。

$$m\ddot{x}_{cm} = f_x - b\dot{x}_{cm} \quad (1.13)$$

さて、この運動方程式 (1.13) に対して、Fig.1.16 のように位置のフィードバックをかけることにより、搬送物体の運動方程式は、もともとの (1.13) から

$$\begin{aligned} m\ddot{x}_{cm} &= K_{fx}(x_r - x_{cm}) - \mu W \dot{x}_{cm} \\ m\ddot{x}_{cm} + \mu W \dot{x}_{cm} + K_{fx}x_{cm} &= K_{fx}x_r \end{aligned} \quad (1.14)$$

上記の (1.14) となる。伝達関数表現では、(1.16) のようになる。(1.16) 中の K_{fx} の調節により、搬送物体の位置から目標位置までの特性を調整できる。

$$\frac{X_{cm}(s)}{X_r(s)} = \frac{K_{fx}}{ms^2 + \mu Ws + K_{fx}} \quad (1.15)$$

$$\frac{X_{cm}(s)}{F_{dx}(s)} = \frac{1}{ms^2 + \mu Ws + K_{fx}} \quad (1.16)$$

MDMS のフィードバックループのブロックダイアグラムは Fig.1.16 となる。また Fig.1.17

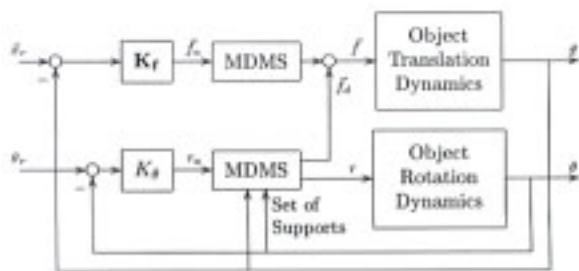


Fig.1.16: MDMS のフィードバックのブロック線図

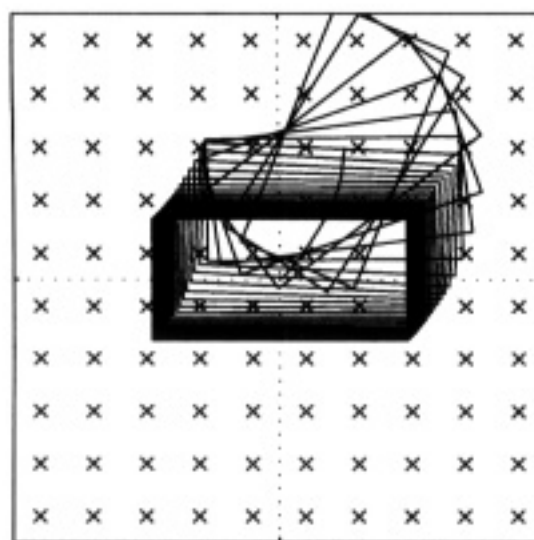


Fig.1.17: フィードバック位置制御によるシミュレーション結果

にフィードバックコントローラにより搬送物体の位置制御を施した場合の搬送物体の挙動を示す、シミュレーション結果を示す。論文には時間応答などやゲインとの関係などが書いていなかったのが残念である。

PVF - Programmable Vector Fields - [13], [14], [15], [16]

Programmable Vector Fields(PVF) の研究は Karl F. Böhringer らのグループによって行われている。Karl F. Böhringer らは MEMS デバイスの研究者でもあり、その立場から分散マニピュレーションのアルゴリズム研究に貢献している。Karl F. Böhringer らは MEMS デバイスのアクチュエータセルは非常に脆く、またセンサをつけたり、自律するといった高度は判断を行うことができるようなものの製作は難しい、という立場をとっている。そんな MEMS のアクチュエータセルでも実現可能な分散マニピュレーションのアルゴリズムが必要であると主張している。

PVF では、2次元平面内に高密度にアクチュエータセルを集積し、各アクチュエータセルは任意の力を出すことができる。そのときそれらアクチュエータセルにより生成される力場から搬送物はどのような影響を受けるかを解析している。

さて、アクチュエータセルにより生成される力場 $f = f(x, y)$ とし、搬送物体の形状を示す関数 $w(x, y)$ を定義する。ただし、搬送物体の大きさは有限であるから $W = \int_{\mathbf{R}^2} w(\mathbf{p})d\mathbf{p} < 0$ である。搬送物体の受ける力 F は、

$$F = \int_{\mathbf{R}^2} w(\mathbf{p})\mathbf{f}(A_\theta\mathbf{p} + \mathbf{t})d\mathbf{p} \quad (1.17)$$

であり、搬送物体が受けるモーメント M は、

$$M = \int_{\mathbf{R}^2} w(\mathbf{p})(A_\theta\mathbf{p}) \times \mathbf{f}(A_\theta\mathbf{p} + \mathbf{t})d\mathbf{p} \quad (1.18)$$

である。ただしここで、 A_θ は回転行列で、

$$A_\theta = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (1.19)$$

である。搬送物体の安定条件は、

$$F = \mathbf{0} \quad (1.20)$$

$$M = \mathbf{0} \quad (1.21)$$

$$(1.22)$$

である。 f として楕円型の力場 (1.23) を考える (Figs.1.18-a,1.18-b 参考)。

$$\mathbf{f}(x, y) = [-\alpha x \quad -\beta y]^T \quad (1.23)$$

このとき、力の平衡点 q は

$$F = \int_{\mathbf{R}^2} w(\mathbf{p})\mathbf{f}(A_\theta\mathbf{p} + \mathbf{t})d\mathbf{p} \quad (1.24)$$

$$= [-\alpha Wx \quad -\alpha Wy] \quad (1.25)$$

$$= \mathbf{0} \quad (1.26)$$

から $q = [0, 0, \theta]$ を得る。また、

$$M = \int_{\mathbf{R}^2} w(\mathbf{p})(A_\theta\mathbf{p} + \mathbf{t})d\mathbf{p} \quad (1.27)$$

$$= (\alpha - \beta) \left\{ \frac{\sin 2\theta}{2} \int (x^2 - y^2)w(x, y)dxdy \right\} \mathbf{k} \quad (1.28)$$

$$+ (\alpha - \beta) \left\{ \cos 2\theta \int xyw(x, y)dxdy \right\} \mathbf{k} \quad (1.29)$$

$$= \mathbf{0} \quad (1.30)$$

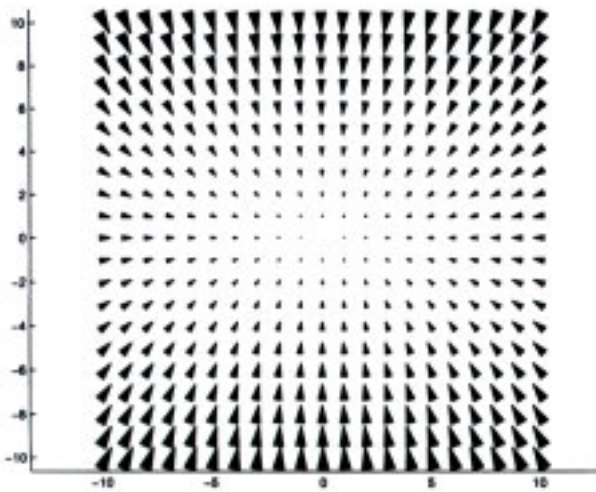


Fig.1.18-a: PVF のカベクトル

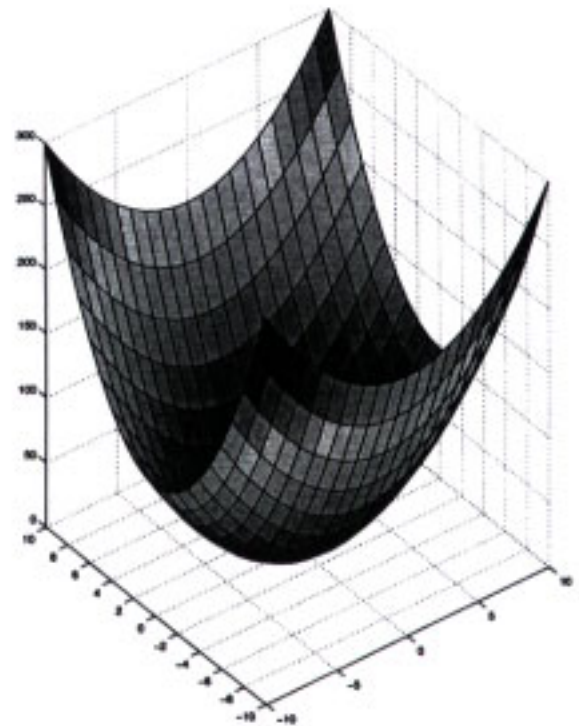


Fig.1.18-b: PVF により生成されるポテンシャル

Fig.1.18: Elliptic force field $\mathbf{f}(x, y) = [-\alpha x \quad \beta y]^T$

$$\frac{s_{20} - s_{02}}{2} \sin 2\theta + s_{11} \cos 2\theta = 0 \quad (1.31)$$

ただしここで、

$$s_{mn} = s_{mn}(w) = \int_{\mathbf{R}_2} x_m y_n w(x, y) dx dy \quad (1.32)$$

であり、(1.31) が満たされるためには、 $[\frac{s_{20} - s_{02}}{2} \quad s_{11}]^T$ と $[\sin 2\theta \quad \cos 2\theta]^T$ が直交することが条件となる。

- **SYMMETRY:** $s_{11} = s_{20} - s_{02} = 0$

このとき任意の $\theta \in [0, 2\pi)$ で式 (1.31) が満たされる。これは物体の元の形状で安定位置が決まるので、もしこの状態に陥ってしまうと楕円型の力場では対象物を回したりといったことを不可能になる。

- **ASYMMETRY:** $\theta = \theta_0, \theta_0 + \pi$

θ_0 は搬送物の形状と楕円型の力場のパラメータ α, β に依存し決まる。しかしながら $w(p)$ が与えられれば、任意の θ_0 を得る α, β を計算し容易に対象物を操作できる。

すでに示したとおり、PVF ではアクチュエータセルは脆く、センサも持たず、高度な判断はできないという立場である。そんなアクチュエータセルにより発生される力場で分散マニピュレーションを行うときは搬送物はただ安定状態に向かい移動する。すなわち安定点がどこにある力場を生成するかが重要となり、それが搬送物体の操作法となる。ここでは、楕円型の力場のみの解析だけであったが、他の力場での解析や MEMS での搬送実験なども論文には盛り込まれていることを付記しておく。

1.2.3 ボトムアップ方式

ここまでは、主に分散マニピュレーションに対するアプローチとしてトップダウン的な方法によるアプローチのものを紹介した。本節では、前節までとは違い、ボトムアップ方式によるアプローチの論文を紹介する。

ADM - Autonomous Distributed Micromachines -

Autonomous Distributed Micromachines (ADM) は LSI などの半導体加工技術を用いて、多数のマイクロアクチュエータを製作し、そのアクチュエータで相対的に巨視的なタスクを実現する装置で、小西らのグループで主に研究が行われている。個々のアクチュエータセルは自分でマイクロプロセッサなどを持っており、必要があれば自分で自分のフィードバック制御などが可能である。また、そのマイクロプロセッサで自律しており、隣接する他のアクチュエータセルと協力することにより分散マニピュレーションを可能とする。Figs.1.19,1.20 にその概念図を示す。シリコンウェハ上にアクチュエータ、マイクロプロセッサ、センサを集積し、それらにより物体を搬送する。個々のアクチュエータは小型で非力でストロークも小さいけれども、多数を平面上に並べることにより比較的重いものや広範囲の移動も可能になる。

搬送アルゴリズム ボトムアップ方式による、ADM の物体搬送アルゴリズムは、

1. まずはじめに搬送目標位置のアクチュエータセルを算出する。
2. 算出されたアクチュエータセルは隣接する 8 つのモジュールに対して”TowardMe”シグナルを送る。
3. あるセルが”TowardMe”シグナルを受信したら、そのセルはシグナルを受信した方向に物体を運ぶように外力を発生する。ただし、複数のセルから”TowardMe”シグナルを受信したら何もしない。
4. ”隣接し、シグナルを送信してきたセルと反対側の 3 つのセルに対して”TowardMe”シグナルを送る。

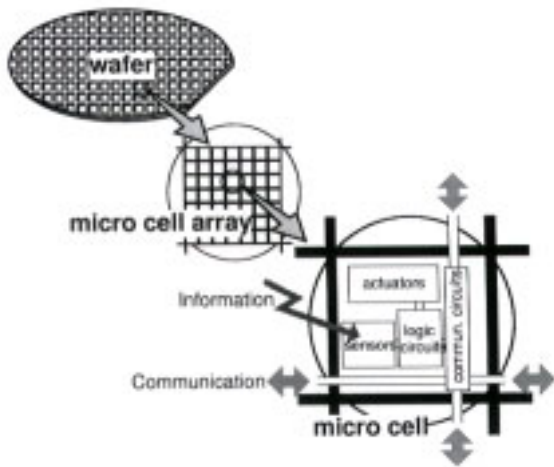


Fig.1.19: ADM の基本構成概念図その 1

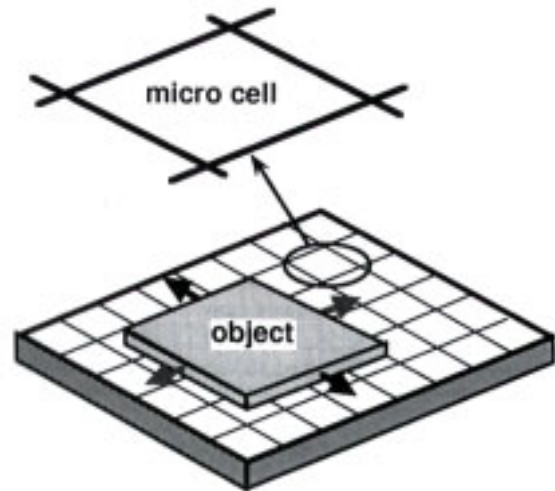


Fig.1.20: ADM の基本構成概念図その 2

5. 物体搬送中は 3, 4 のステップを繰り返す。

の通りである (Fig.1.21 参考)。

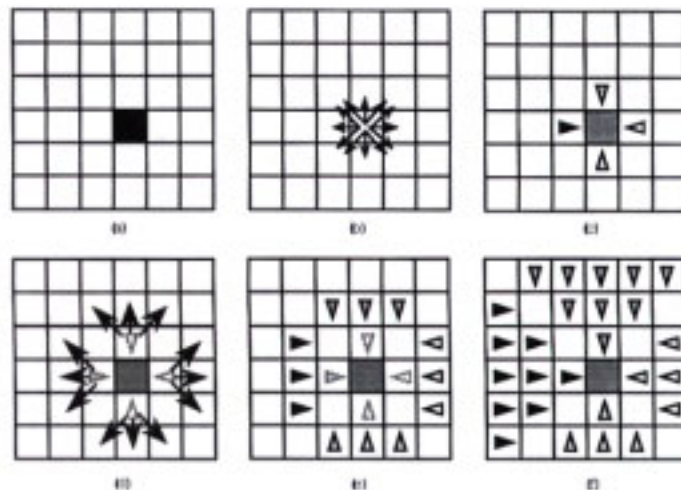


Fig.1.21: ADM の基本動作

耐故障性ロジック マイクロマシンアクチュエータは普通のアクチュエータに比べ壊れやすい。そのため耐故障のアルゴリズムを内包することが望ましい。故障を

- 2,3 個のアクチュエータが壊れた。
- アクチュエータが壊れただけ。ただし、その範囲は広範囲に及ぶ。

- アクチュエータとコントローラの両方が壊れる。

と分類する。これにより、故障から回復するアルゴリズムと故障を回避するアルゴリズムは、

1. 故障回復ロジック: (Fig.1.22 参照)

- "TowardMe" シグナルを受信していないアクチュエータセルは故障回復ロジックが動作している。
- 故障回復ロジックが動作しているとき、アクチュエータセルは右か左かのどちらかがアクティブであればそれと同じ方向でアクティブとなる。

2. 故障回避ロジック: (Fig.1.23 参照)

- "TowardMe" シグナルを受信していないアクチュエータセルは故障回避ロジックが動作している。
- 故障回避ロジックが動作しているとき、アクチュエータセルは自分に向かってアクティブとなっていないアクティブなアクチュエータセルの方向にアクティブとなる。

で得ることができる。Figs.1.24,1.25 に故障回復と故障回避のシミュレーション結果をそれぞれ示す。良好な結果を示しているのがわかる。

1.2.4 両方からのアプローチ - ボトムアップとトップダウンそれぞれから

-

ここまではボトムアップ、トップダウンそれぞれの立場から分散マニピュレーションにアプローチしている研究例について紹介した。本節ではボトムアップ、トップダウンの双方から、それぞれの長所、短所を理解し、アプローチしている研究について紹介する。

FTS - Flexible Transfer System - [17], [18]

FTS(Flexible Transfer System) は Fig.1.26 のような 25 個の物体搬送ユニット (アクチュエータセル) を 2 次元平面内に 5×5 で並べた装置である。各アクチュエータセルは、

- 自分の上に搬送物があるかないか判断でき、
- 東西南北の方向に搬送物を運ぶことができる。

FTS の目的は、Fig.1.27 のように FTS に投入された搬送物体を最適なルートで目標位置まで搬送することにある。アクチュエータセルが故障した場合はそのアクチュエータセルを

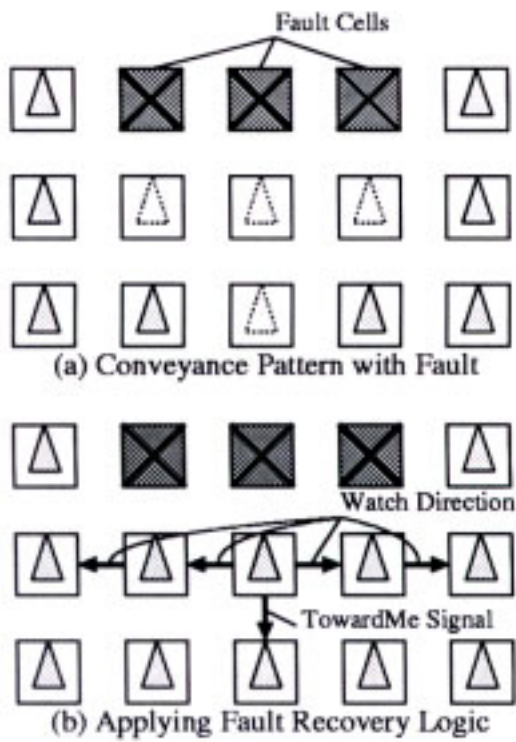


Fig.1.22: 故障回復ロジック

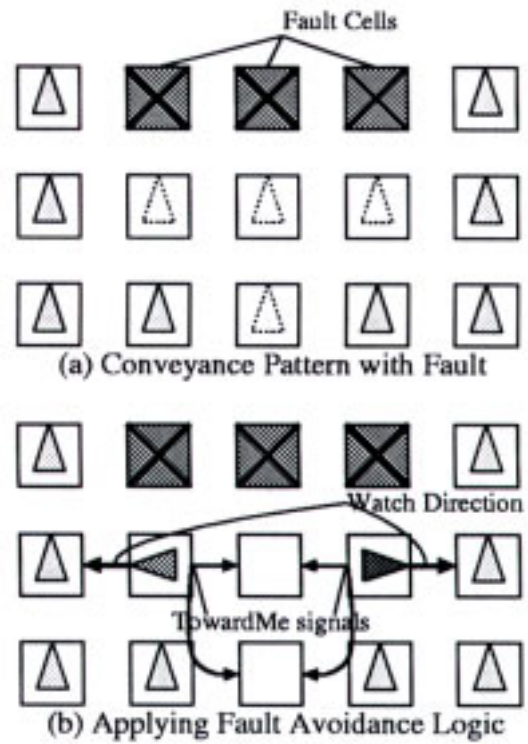


Fig.1.23: 故障回避ロジック

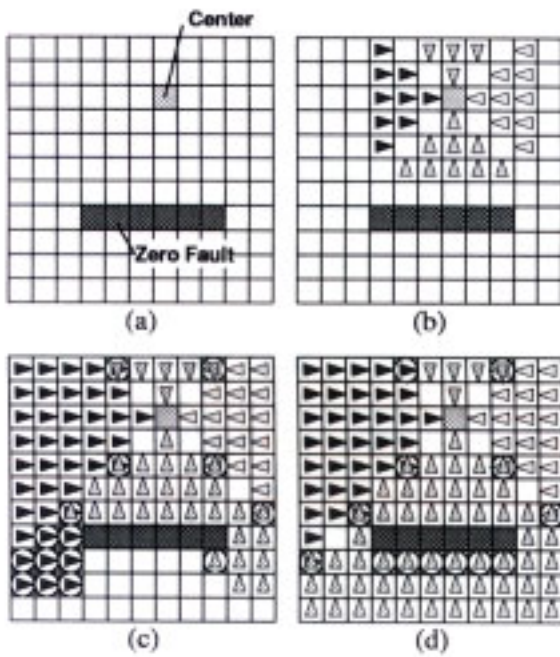


Fig.1.24: 故障回復ロジックのシミュレーション結果

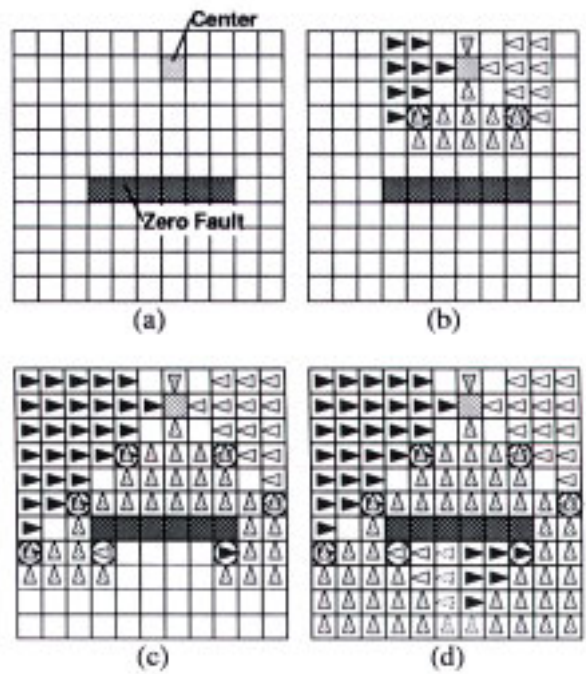


Fig.1.25: 故障回避ロジックのシミュレーション結果

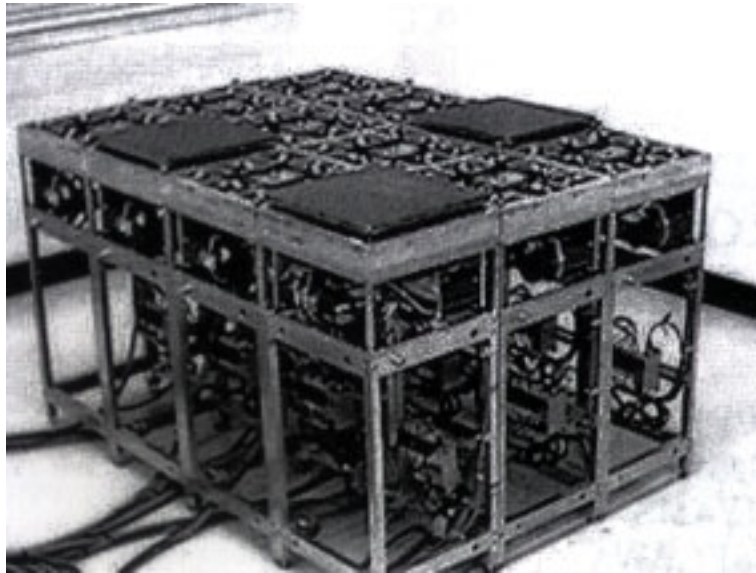


Fig.1.26: FTS 外観

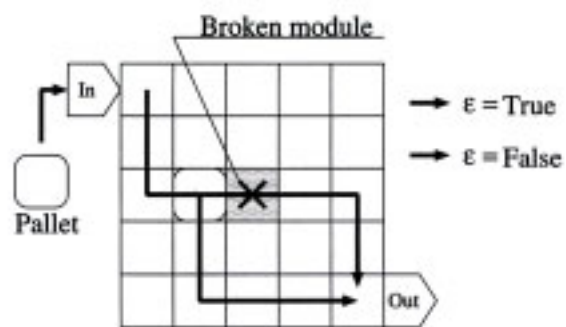


Fig.1.27: FTS の基本動作

避けるルートを選択しなくてはならない。この FTS の問題はパスプランニングの問題である。

この FTS ではあるアクチュエータセルの故障は隣のアクチュエータセルのみ知ることができる仮定している。すなわち、隣のアクチュエータが壊れていたら、今までと違ったパスに搬送物体を運ぶことを、各アクチュエータセルは判断しなくてはならない。また、各アクチュエータセルは 25 個の全体に対する情報は持っていない、搬送物体の搬送目標位置もわからない。そのため、全体を総括し最適なパスプランニングをするトップダウン方式のアルゴリズムの設計と、各アクチュエータセルは隣接するアクチュエータセルの故障を検知し、ローカルな判断でフレキシブルにパスプランニングを行う、ボトムアップ方式のアルゴリズムの設計の二つのアプローチが必要となる。

例えば、Fig.1.27 で、はじめにトップダウン方式による方法で最適なパスプランニングを行っておく (Broken module の上を通過しているパス)。あるアクチュエータセルが壊れたら、そこを避ける判断を Broken module の隣の丸く印のついているアクチュエータセルがローカルな情報を基にしてボトムアップ方式により行う。FTS では、このトップダウン方式とボトムアップ方式の双方からのパスプランニングの方法として、トップダウン方式は遺伝的アルゴリズム (GA) による方法を、ボトムアップ方式は学習オートマトン (LA) による方法を採用している。

LA によるボトムアップ方式によるパスプラン

LA によるパスプランを考えるに当たり、まずはじめに幾つかの関数などを定義する。アクチュエータセルは $p = (x, y)$ にあるとし、隣接するアクチュエータセルの集合を、

$$p(i) = \{(x + 1, y), (x, y - 1), (x - 1, y), (x, y + 1)\} \quad (1.33)$$

で定義する。 $p(1), p(2), p(3), p(4)$ がそれぞれ隣接アクチュエータセルとなる。次に、アクチュエータセル p の時刻 t での内部状態変数 $h_p(t)$ を以下で定義する。

$$h_p(t) = \left[b_{p(1)} \quad b_{p(2)} \quad b_{p(3)} \quad b_{p(4)} \right]^T \quad (1.34)$$

ただし、ここで $b_{p(1)}, b_{p(2)}, b_{p(3)}, b_{p(4)}$ は、それぞれ $p(1), p(2), p(3), p(4)$ 方向のアクチュエータセルへの搬送物体移送確率である。当然、必ずどこかに移動させるから、(1.35) のようにその和は 1 となる。

$$\|h_p(t)\| = \sum_{i=1}^4 |b_{p(i)}| = 1 \quad (1.35)$$

次に報酬変数を (1.36) で定義する。 r_p はアクチュエータセル p が隣接するアクチュエータセル $p' \in p(i)$ に搬送物体を移動させたとき受け取る報酬である。

$$r_p(t + 1) = (1 - \theta_r)r_p(t) + \theta_r r_{p' \in p(i)} \quad (\theta_r = const.) \quad (1.36)$$

(1.36) で定義した報酬変数を基にして、各アクチュエータセルの内部状態変数を (1.37) で更新する。ただし、ここで A はアクチュエータセル p が搬送物体を渡すセルで $A \in p(i)$ である。

$$h_p(t+1) = Lh_p(t) \quad (1.37)$$

ただし、ここで L は学習オートマトンで L_{R-I} メソッドで定義される方法で (1.38) で定義する。

これにより各アクチュエータセルは自己組織化し、搬送が可能となる。

$$L \equiv b_{p(i)}(t+1) = \begin{cases} (1 - \theta_v r_A) b_{p(i)}(t) + \theta_v r_A & (i = A) \\ (1 - \theta_v r_A) b_{p(i)}(t) & (i \neq A) \end{cases} \quad (1.38)$$

$(\theta_v = const.)$

GA によるトップダウン方式によるパスプラン GA によるトップダウン方式によるパスプランは多数あるルートのうち最短のものを GA により探し出す問題である。トップダウン方式では仮定通り、各アクチュエータセルの故障は考慮されないため、一度最初にパスプランを行えばそのままである。GA によるルート検索の手続きは Fig.1.28 で、標準的な方法である。

```

begin
  i ← 0
  path initialization
  repeat
    evaluation
    selection
    mutation
  i ← i + 1
  until i = end of generation
end

```

(1.39)

Fig.1.28: GA ルール

FTS シミュレーション結果 FTS のシミュレーション結果全てで共通で、縦軸は搬送能力である（詳細は論文には書いてないが、単位時間当たりの搬送量と思われる）。横軸は時刻で、10000step までの時刻では全てのアクチュエータセルは故障することもなく、正常に動

作している。10000step から 30000step までは 25 個のアクチュエータセルのうち 3 つが故障し、30000step から故障したアクチュエータセルは故障から回復し、全てのアクチュエータセルは正常な動作に戻る。Fig.1.29 に FTS に LA (ボトムアップ形式) だけを適用した場合の物体搬送のシミュレーション結果を示す。10000step を越えたところでアクチュエータセルの故障により、搬送能力が著しく低下している。しかしながら、その後、LA により搬送パスは変更され、搬送能力は若干低下しながらも復元されているのが Fig.1.29 から読みとることができる。30000step を越えたところでアクチュエータセルは故障から回復し、全てのアクチュエータセルは正常動作に戻るから、搬送能力も元に戻るはずであるが、LA だけの方法では元の状態に戻ることができないことがわかる。

Fig.1.29 の LA だけの搬送シミュレーション結果から、LA だけのボトムアップ方式のアルゴリズムだけではアクチュエータセルが故障から回復した際に問題があることがわかった。Figs.1.30,1.31 に LA によるボトムアップ方式と GA によるトップダウン方式の双方からのアプローチによるシミュレーション結果を示す。Fig.1.30 は搬送能力を示し、Fig.1.31 は LA と GA のどちらのパスプランが最終的に採用されたかを示している。Fig.1.30 から 10000step を越えたところでは LA によるパスプランが支配的になり、故障を補償している。また、30000step を越えたところでは今まで LA によるパスプランが支配的だったのが、GA によるパスプランが支配的になり、LA だけアルゴリズムでは問題となったアクチュエータセルが故障から回復しても、搬送能力は回復しない問題が解決されている。Fig.1.31 から同様の事情が容易に理解することができる。

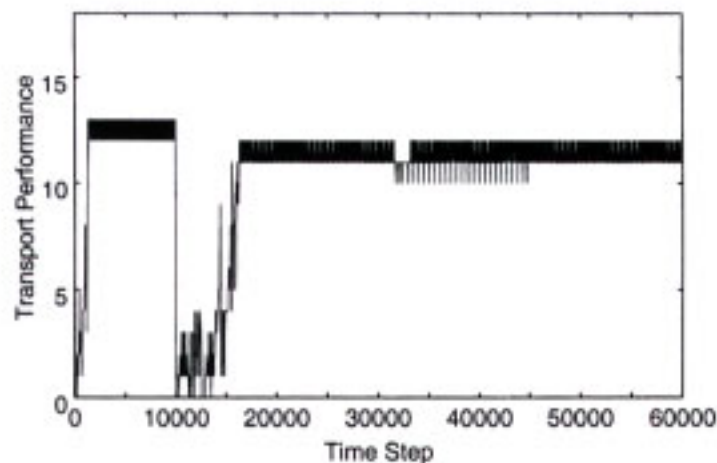


Fig.1.29: LA だけを施した場合の搬送性能

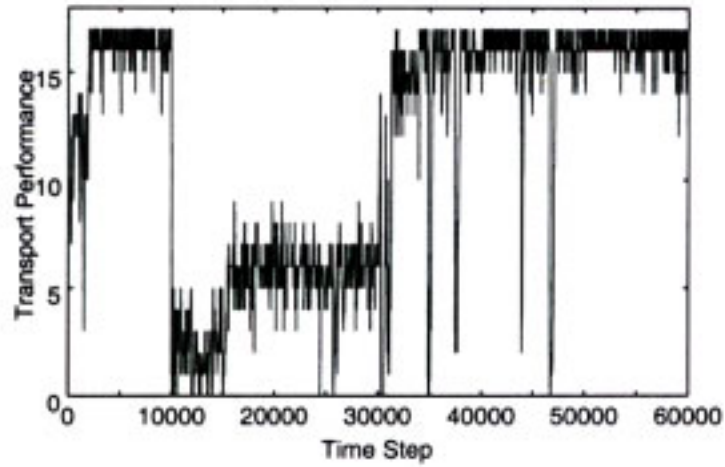


Fig.1.30: LA と GA の複合学習を用いた場合の搬送性能

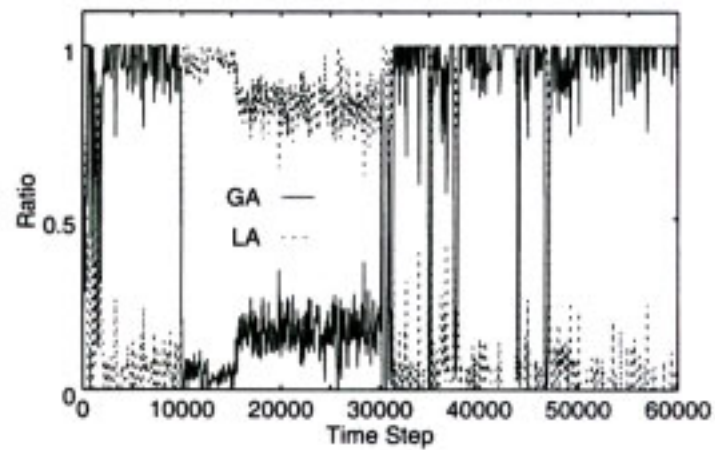


Fig.1.31: LA によって決定されるパスと GA により決定されるパスの選択比

1.2.5 まとめ

本節では分散マニピュレーション法の幾つかの研究例を紹介した。分散マニピュレーションに対するアプローチとして、トップダウン方式とボトムアップ方式の2つの方式による分類からそれぞれについて紹介を行った。また、最後にはトップダウン方式とボトムアップ方式の両方から分散マニピュレーションのアルゴリズムを提案している論文も紹介した。それぞれの紹介内容には、

- トップダウン方式

- MDMS (Modular Distributed Manipulator System)

MDMS は速度制御された回転するホイール (アクチュエータセル) を 2次元平面内に並べ、それにのせられた搬送物体の位置制御を実現した。搬送物体とホイールとの間に摩擦モデルを考え、搬送物体が受ける力を導出している。この式の導出は一般性が高く、分散マニピュレーションの視点から見て興味深い。MDMS のアクチュエータセルの数は比較的少ないと仮定され、一つの全てのアクチュエータセルを総括するシステムにより搬送物体の位置制御系が実現されていた。

- PVF (Programmable Vector Fields)

PVF は MEMS での実現を考えから高密度でのアクチュエータセルの集積を仮定し、MDMS とは異なった連続系近似のアプローチが採られている。各アクチュエータセルが任意の力が出せることを仮定し、その力場中で搬送物体がどのような挙動を示すか解析を行った。各アクチュエータセルは壊れやすく、センサも積載していない大変シンプルなものを仮定していた (マイクロマシンで作るとそれほど複雑なものは作れないと筆者はいつている)。

- ボトムアップ方式

- ADM (Autonomous Distributed Micromachine)

ADM では各アクチュエータセルは比較的高度な判断をすることができ、センサなども持っていることを仮定している (同じ MEMS での実現でも PVF とは立場が違う)。各アクチュエータセルが各々判断することにより、システム全体として耐故障と故障回避を実現した。

- トップダウン方式とボトムアップ方式双方から

- FTS (Flexible Transfer System)

FTS は分散マニピュレーションの定義から若干はずれるが、パスプランニングの方法としてトップダウン方式は GA を用いボトムアップ方式は LA を用い、そ

れらを上手く組み合わせている。アクチュエータセルの故障に対してもフレキシブルに対応し、パフォーマンスもあまり下げることのないシステムが実現された。

であった。

各システムにより条件は異なるものの、システムの構成だけを考えれば、

- 各アクチュエータセルの自律の度合い
- システムを構成するアクチュエータセルの数

がトップダウン方式のアルゴリズムを考えるか、ボトムアップ方式のアルゴリズムを考えるかのキーとなっている。各アクチュエータセルの自律が高く、アクチュエータセルの数が多ければ、ボトムアップ方式のアルゴリズムがよく適している。逆に、各アクチュエータセルの自律の程度が低く、またアクチュエータセルの数が少なければ、ボトムアップ方式により全てのアクチュエータセルを統一的に管理するのがよい。

アルゴリズムの設計のし易さをだけ考えれば、

- トップダウン方式は、システムの設計方法がわかりやすい。
- ボトムアップ方式は、各アクチュエータセルで決めるルールが全体にどんな影響を及ぼすか予測しにくく、設計が難しい。

であり、トップダウン方式が有利である。

これらはシステムの構成とアルゴリズム設計の難しさは、トレードオフの関係にあることを示している。多数のアクチュエータセルによりきめ細かな分散マニピュレーションを実現しようとするれば、複雑なアルゴリズムを考えなくてはならないと思われる。アクチュエータセルが少なく、アルゴリズムの設計が容易である場合は、きめ細かな分散マニピュレーションを実現するのは難しいだろう。

このトレードオフの関係を克服する方法として、トップダウン方式とボトムアップ方式の双方からのアプローチは第 3 の方法として大きな意味を持つと思われる。

1.3 本論文の構成 - 分散マニピュレーション研究へのアプローチ -

1.3.1 分散マニピュレーションの基本三動作『計る』『つかむ』『運ぶ』

マニピュレーションは分散マニピュレーションであれ、普通の集中マニピュレーションであれ（普通は集中は付けずただマニピュレーションという、なぜならマニピュレーシ

ンといえは普通集中マニピュレーションであるから。) 対象物を操作することが目的である。分散であるか集中であるかはマニピュレータ側の構成の違いであり、方法論の違いである。

そういった意味で、分散であれ集中であれ、マニピュレーションそのものに普遍の動作、基本となる動作があると考えられる。工場で働くロボットマニピュレータは、基本となる動作の繰り返しのシーケンスより制御され、複雑な動作を実現している。我々があるものをある場所に移動させることを考えてみると、まず右手でつかみ、対象物のだいたいの重さを計り、力をおおざっぱに計算し、目標位置まで搬送する。この場合、基本となる動作は、計る、つかむ、運ぶである。これら、基本となる動作の組み合わせにより、我々は対象物を移動させる。

マニピュレーションにおいては、

- 基本動作の正確な実現
- その組み合わせ、複合動作

が重要となる。

そこで、分散マニピュレーションの基本三動作として、

- 計る
対象物のマニピュレーションに必要な力学的な情報を計測することが目的で、具体的には、重心位置、質量、慣性モーメントの計測を行う。
- つかむ
対象物の把持を行う。移動させるといったマニピュレーションにはしっかりした把持が必要である。
- 運ぶ
対象物を目標位置まで搬送するというマニピュレーションに最も必要な動作である。

考え、これらの組み合わせにより任意のマニピュレーションが実現できると考える。これらの基本動作の正確な実現が本論文の第一の目的である。

次に複合動作として、

- 計りながらつかむ
計測した情報をもとにして、バランス良くしっかり把持する。
- つかみながら運ぶ
運ぶ方向に応じて把持の仕方を変える。

- 計りながらつかみ、そして運ぶ
さらに上記二つの複合動作

を考え、それらの実現を本論文の第二の目的とする。これら二つの目的の達成から、分散マニピュレーションの一般的で包括的な考察とする。

本論文の構成を Fig.1.32 に示す。

まずはじめに、分散マニピュレーションという特殊なマニピュレーションそれ自身を持っている特徴について考察する。前節の分散マニピュレーションをめぐる研究状況で紹介した、トップダウン方式、ボトムアップ方式といったアルゴリズムのタイプも分散マニピュレーションならではの。また、空間周波数による制限、アクチュエータの構造、無駄時間による制約、力密度による制約といった事柄も分散マニピュレーションならではの。これらを、分散マニピュレーションの基本要件とする。

次に、3章、4章、5章では基本動作の実現を目指す。3章は、基本三動作の内の計るで、キーとなる技術は外乱オブザーバレイである。4章ではつかむの実現を目指すし、キーとなる技術は面インピーダンス制御である。5章は、運ぶの実現を自律分散集中複合系というキー技術から目指す。

6章では、論文第二の目的である複合動作の実現を目指す。二つの動作の結合をした上級二動作、計りながらつかむ、つかみながら運ぶをそれぞれ 6.2 章、6.3 章で、さらにそれらを複合したはかりながらつかみ、そして運ぶは 6.4 章で考察する。

7章では本論文の結論を述べ、8章では今後の分散マニピュレーション研究の残された課題と展望を読む。

1.3.2 本論文題目に関する考察

本論文題目は『リニアアクチュエータアレイによる分散マニピュレーション法の新しい制御論理に関する研究』としたが、分散マニピュレーションという言葉の利用には少なからず考察が必要だった。

そもそも、分散マニピュレーションの研究は、海外で中心に行われており、また国内でもその発表がほとんど国際的な場、すなわち国際会議である。これは前節の分散マニピュレーションの研究状況で参照している論文もほとんど全て国際的な場で発表された内容であることからもうかがえる。そのため参考となる文献は英語である。論文中では、DISTRIBUTED MANIPULATION という言葉が一般的に使われている。この言葉にどのような日本語訳を当てるかが問題であった。日本国内で、日本語の論文で対応する用語を調べてみたが適切なものが見つからなかった。

そこで、DISTRIBUTED MANIPULATION 研究の中心的な人物であり、参考文献 [5], [6] の著者である小西氏に問い合わせてみたところ、分散マニピュレーションまたは分散型マ

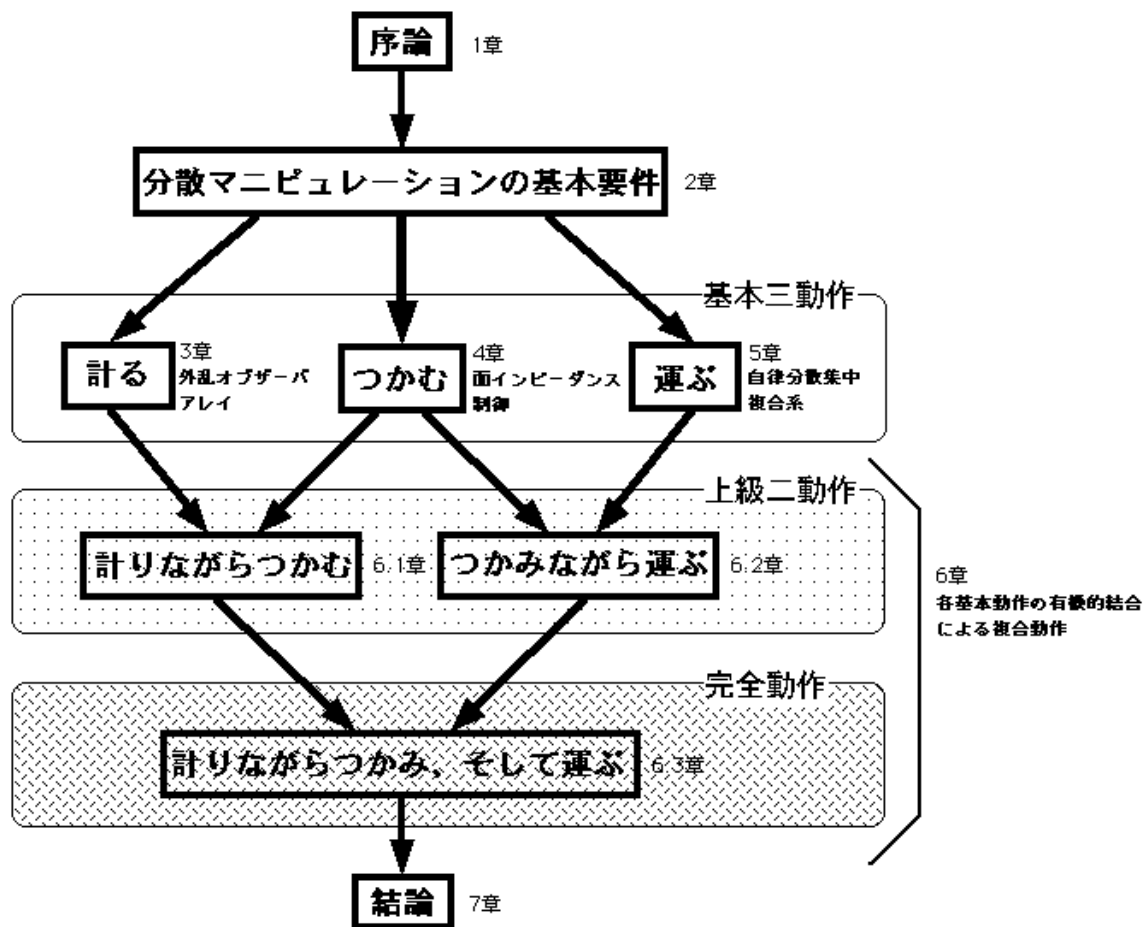


Fig.1.32: 本論文の構成チャート

ニピュレーションが適切だろうという回答を頂いた。分散型という型という言葉を入れる理由として、マニピュレーションするシステム側が分散しているのであって、マニピュレーションされるもの、マニピュレーション対象物が分散しているのではないということに起因しているとのことである。

言葉のゴロの良さから、分散マニピュレーションを DISTRIBUTED MANIPULATION の日本語訳として採用することとし、この言葉が日本国内でも一般的な言葉となればという思いを込め、本博士論文の論文題目とする。本論文が分散マニピュレーションを国内での研究の普及の礎となれば本望である。

第 2 章

分散マニピュレーション基本要件

2.1 まえがき

本章では、分散マニピュレーションの基本要件について考察する。分散マニピュレーション法はマニピュレーション法としても非常に特殊である。そのため、そのマニピュレーション法そのものが固有に持っている要件、分散マニピュレーション法を考える上で外すことのできないポイントとなる点が幾つかある。

さて、分散マニピュレーション法は『多数のアクチュエータにより多数の支点を通じて対象物に作用し操作する方法』と定義される。この定義から、

- 多数のアクチュエータであるがそのアクチュエータ間隔がどの程度必要かということから、空間周波数による制約を受ける。
- 各アクチュエータの発生できる力とアクチュエータ間隔から決まる力密度による制限。
- それぞれのアクチュエータは対象物にどのように作用するのかという、多数のアクチュエータの一つ一つはどのような構造なのかというアクチュエータの構造に関する条件もある。
- アクチュエータが多数であるがため一つ一つのアクチュエータに指令を送ったり、その指令を計算するのに時間がかかる。これは無駄時間としてシステムに内蔵する。無駄時間による制約が存在する。
- それぞれのアクチュエータがどのように働くのかという点から、アルゴリズムのタイプをどうするかという視点が存在する。

以下では、これら、空間周波数による制約、力密度による制限、アクチュエータの構造、無駄時間による制約、アルゴリズムのタイプ、5つの分散マニピュレーション基本要件を検討する。

また、分散マニピュレーション基本要件を考慮しながら、基本三動作や連携動作を実現するとき、この分散マニピュレーション基本要件に関して、

- マニピュレーション対象物ほどの程度の空間周波数を持ち分布するのか。それは時変かどうか。
- 発生できる力密度はどの程度か。無限ではないことは確かである。
- 無駄時間はどの程度か。1秒や2秒もあつたら安定化は難しいすぎるし、非現実的な量だろう。
- 序論内の第 1.2 章、分散マニピュレーションをめぐる研究状況で分類したように、既存の研究でもトップダウン方式、ボトムアップ方式など、アルゴリズムのタイプはいろいろあるが、どうするか。
- アクチュエータはリニアドライブタイプなのか、それとも回転型のモータタイプなのか、アクチュエータ構造はどのようになっているのか。

等の問題が生じる。条件を決め固定する必要があるだろう。どんなシステムでも何でもできるわけではない。幾つかの条件下でのみ成立するはずである。その条件があまりにシステムの成立条件を狭めるようになっては困るが、その条件があるおかげで、システムの理論的な展開や構築を大いに助けることがある。一般性をあまり失わない程度で条件、現実的な仮定をつけることはむしろ必要であるかもしれない。

本章最後では、幾つかの現実的な仮定を行い、次章以降の基本動作や連携動作の実現の理論的な展開等を助けることにする。

2.2 空間周波数による制約

ここで $f(x, y)$ は対象物の分布に関する情報と一般的な形で定義しておく。例えば、質量分布や表面形状の分布、色の分布などが一例である。それらの情報を、空間的にある間隔 (x 方向が Δx で y 方向が Δy) で配置されたアクチュエータに付随するセンサや外乱オブザーバでサンプリングし、 $f(x, y)$ は再現される。アクチュエータ間隔 (センサ間隔) が広ければ、欠落する情報は多いと考えられる。本節ではそれを定量的に考察し、分散マニピュレーションに求められるアクチュエータ (センサ) の間隔等の配置、空間周波数に関する考察を行う。

$f(x, y)$ を x 方向、 y 方向それぞれ Δx , Δy でサンプリングする関数 $comb(x, y)$ を

$$comb(x, y) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \delta_d(x - k_1\Delta x, y - k_2\Delta y) \quad (2.1)$$

で定義する。ただし、 δ_d はデルタ関数である。後の方で δ という関数が出てくるので、それと区別するために δ_d としてある。各アクチュエータでサンプリングされた信号 $f_s(x, y)$ は、

$$f_s(x, y) = f(x, y) \text{comb}(x, y) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} f(k_1 \Delta x, k_2 \Delta y) \delta_d(x - k_1 \Delta x, y - k_2 \Delta y) \quad (2.2)$$

である。2次元のフーリエ変換は、

$$F(\mu, \nu) = \mathcal{F}\{f(x, y)\} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(\mu x + \nu y)} dx dy \quad (2.3)$$

で定義される。また、

$$\mathcal{F}\{\text{comb}(x, y)\} = \frac{1}{\Delta x \Delta y} \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \delta_d\left(\mu - \frac{k_1}{\Delta x}, \nu - \frac{k_2}{\Delta y}\right) \quad (2.4)$$

である。結局、

$$\begin{aligned} F_s(\mu, \nu) &= \mathcal{F}\{f_s(x, y)\} = \mathcal{F}\{f(x, y)\} * \mathcal{F}\{\text{comb}(x, y)\} \\ &= \frac{1}{\Delta x \Delta y} \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} F\left(\mu - \frac{k_1}{\Delta x}, \nu - \frac{k_2}{\Delta y}\right) \end{aligned} \quad (2.5)$$

$F_s(\mu, \nu)$ は (2.5) からわかるように、 μ 軸方向に $\frac{1}{\Delta x}$ 、 ν 軸方向に $\frac{1}{\Delta y}$ の間隔で繰り返される周期関数となる。 $F_s(\mu, \nu)$ は周期的に繰り返される $F(\mu, \nu)$ で構成され、サンプル間隔 $\Delta x, \Delta y$ が下記の (2.6) を満たせば、 $F(\mu, \nu)$ は互いに重なり合うことなく、元の形を正確に復元できる。

$$\Delta x \leq \frac{1}{2\omega_{\mu}^{\max}}, \quad \Delta y \leq \frac{1}{2\omega_{\nu}^{\max}} \quad (2.6)$$

(2.6) は 2次元版のサンプリング定理で、元々の関数 $f(x, y)$ は $\omega_{\mu}^{\max}, \omega_{\nu}^{\max}$ 以上の周波数成分を持たず、帯域制限されていることを意味する。

(2.7) の 2次元の窓関数状のフィルタを通過させることにより、 $f(x, y)$ は (2.8) で復元できる。

$$H_R(\mu, \nu) = \Delta x \Delta y \text{rect}(\Delta x \mu, \Delta y \nu) = \begin{cases} \Delta x \Delta y & |\mu| \leq \frac{1}{2\Delta x}, \quad |\nu| \leq \frac{1}{2\Delta y} \\ 0 & \text{otherwise} \end{cases} \quad (2.7)$$

$$\begin{aligned} f(x, y) &= \mathcal{F}^{-1}\{F(\mu, \nu)\} = \mathcal{F}^{-1}\{F_s(\mu, \nu) H_R(\mu, \nu)\} \\ &= \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} f(k_1 \Delta x, k_2 \Delta y) \frac{\sin\left[\frac{\pi}{\Delta x}(x - k_1 \Delta x)\right]}{\frac{\pi}{\Delta x}(x - k_1 \Delta x)} \frac{\sin\left[\frac{\pi}{\Delta y}(y - k_2 \Delta y)\right]}{\frac{\pi}{\Delta y}(y - k_2 \Delta y)} \\ &= \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} f(k_1 \Delta x, k_2 \Delta y) \text{sinc}\left(\frac{x}{\Delta x} - k_1, \frac{y}{\Delta y} - k_2\right) \end{aligned} \quad (2.8)$$

(2.6), (2.8) から、対象物に関する情報はアクチュエータ間隔 $\Delta x, \Delta y$ でサンプリングすることにより、含まれる情報としては $\frac{1}{2\Delta x}, \frac{1}{2\Delta y}$ 以下の周波数帯域に帯域制限され、ローパスフィルタを通過したかようになってしまう。

また、得られる対象物の分布に関する情報はアクチュエータ間隔の広い方によって支配される。すなわち、仮に x 軸方向に極めて密にアクチュエータを配置しても、 y 軸方向には疎であったら、 y 軸方向のアクチュエータ間隔でサンプリングしたことと同じことである。分散マニピュレーションでは、対象物を計ったり、運んだりといった、動的な操作を加えることが前提となるため、基本的には x, y 軸方向に空間周波数指向性を持たせても意味がない。

アクチュエータ間隔は、空間周波数に関する 2 次元のサンプリング定理 [19] からわかるように、できるだけ等しく狭くし、対象物の最大空間分布周波数 $\omega_\mu^{\max}, \omega_\nu^{\max}$ に対して、

$$\Delta x \leq \frac{1}{2\omega_\mu^{\max}}, \quad \Delta y \leq \frac{1}{2\omega_\nu^{\max}}$$

を満たすように設計しなくてはならない。

2.3 力密度による制限

マニピュレーション対象物は $\rho(x, y, z)$ の密度で分布しているとする。このとき、 i, j 番目アクチュエータの担当する領域は、

$$\Omega_{ij} = \left\{ (x, y) : x_i - \frac{\Delta x}{2} \leq x \leq x_i + \frac{\Delta x}{2}, y_j - \frac{\Delta y}{2} \leq y \leq y_j + \frac{\Delta y}{2} \right\} \quad (2.9)$$

で、そのアクチュエータが、その領域に作用している力（表面力とか重力）より、少なくとも大きい力を発生できないとその微小領域を操作できないだろう。物体が比較的大きく、慣性力が支配的ならば、

$$g \int \left[\int_{\Omega_{ij}} \rho(x, y, z) dx dy \right] dz \simeq g \Delta x \Delta y \int \rho(x_i, y_j, z) dz \quad (2.10)$$

が i, j 番目のアクチュエータの担当領域の物体質量分となる。また物体が小さく、表面力が支配的ならば、

$$\int_{\Omega_{ij}} \mathbf{t} \cdot \mathbf{n} dx dy \simeq t_{ij} \Delta x \Delta y \quad (2.11)$$

が i, j 番目のアクチュエータの担当領域の表面積分である。ただしここで、 g は重力加速度とする。

2.4 アクチュエータの基本構造

全ての運動は、並進と回転を基本運動とし、それらの組み合わせで表すことができる。マニピュレーションにおいても同様で、最小の自由度は並進タイプのアクチュエータによる操作と回転タイプのアクチュエータのタイプによる操作である。

分散マニピュレーションは『多数のアクチュエータにより多数の支点を通じて対象物に作用し操作する方法である』で定義された。ここに出てくる多数のアクチュエータによる操作も最も基本は並進タイプか回転タイプであり、全てのマニピュレーションはその組み合わせである。

xy 平面に多数のアクチュエータが並んでいるとすると、回転タイプは F_x または F_y という対象物を x 軸、 y 軸方向といったアクチュエータの並びの方向に作用する力を発生し、操作することができる。これに対して、並進型のリニアドライブは、アクチュエータの並びの方向と違い、 $F_z (= F_i)$ という z 方向に力を発生することができる。

MDMS は、先の序論内の第 1.2 章で紹介したカーネギーメロン大学の研究グループによって開発された分散マニピュレーション型の物体搬送システムであった。各アクチュエータセルは x, y 方向にそれぞれ速度制御され回転するホイールがあり、そのアクチュエータセルを Fig.2.1 のように 2 次元平面に多数並べることで、Fig.2.2 のような自在な物体搬送を実現しようとしている。回転タイプのアクチュエータによる操作の分散マニピュレーション

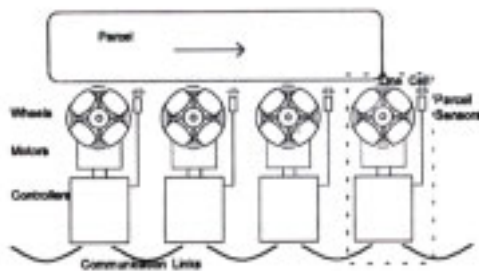


Fig.2.1: MDMS (1):MDMS のアクチュエータセル

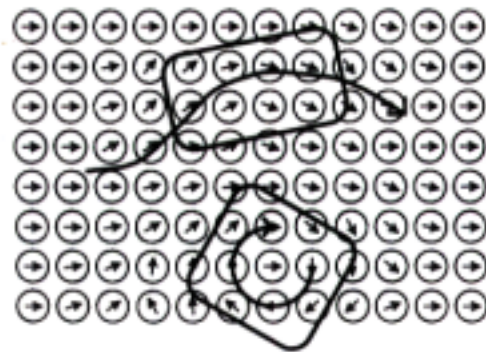


Fig.2.2: MDMS (2) : MDMS の動作

ンの研究は、MDMS により多くがなされている。

このように、対象物にどのような力を与え操作をするかが、アクチュエータの基本構造を決める上でのポイントとなる。

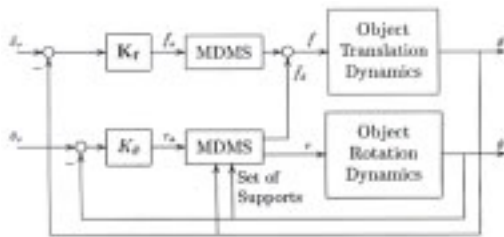


Fig.2.3: MDMS (3):MDMS のフィードバックシステムのブロック線図

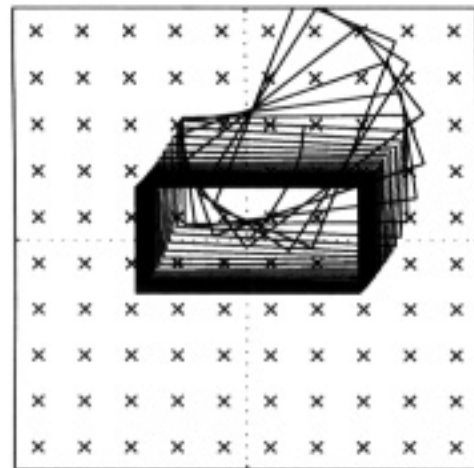


Fig.2.4: MDMS (4) : MDMS による物体搬送のシミュレーション結果

2.5 無駄時間による制限

分散マニピュレーションは多数のアクチュエータで対象物を操作することが大前提である。そのため何らかの形でその多数のアクチュエータは制御されている。もし仮に、非常に高速な計算機と非常に太いバスがあれば、多数あるアクチュエータも全て集中管理的に一点で管理できる。対象物の分布の状況などを知る必要があれば、一元的な管理が必要である。

一点集約方式の場合、各アクチュエータからの情報収集や集中管理系から各アクチュエータへの情報の配信に無駄時間を生ずる。さらに一点で各アクチュエータに指令を行うため、その指令値の計算にも無駄時間が生じる。

情報の配信に伴う無駄時間を τ_{d1} とし、計算に伴う無駄時間を τ_{d2} とすると、トータルの無駄時間が分散マニピュレーションに伴う無駄時間 τ_d となる。

$$\tau_d = \tau_{d1} + \tau_{d2} \tag{2.12}$$

まず情報配信に伴う無駄時間 τ_{d1} を考える。これはアクチュエータの並べ方に依存する。隣のアクチュエータにデータを渡すのに必要な時間を τ_p とし、全てのアクチュエータの数を n_t とすると、各アクチュエータが 2ch. のバスを持っているとすると、全てのアクチュエータからの情報収集には、直線上に並べるのが最も効率が良いものの、少なくとも、

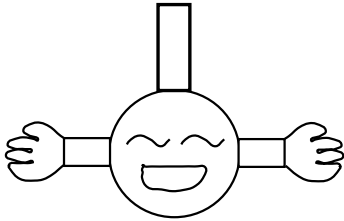


Fig.2.5: 2ch のバスを持っているアクチュエータ

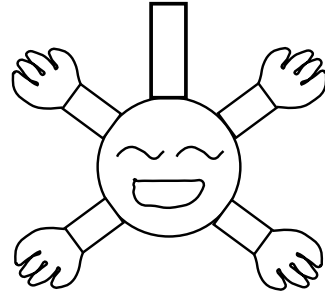
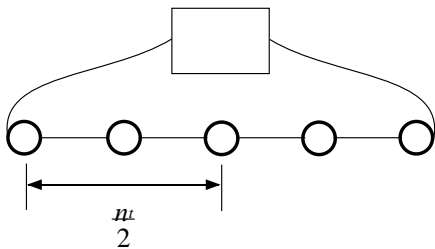
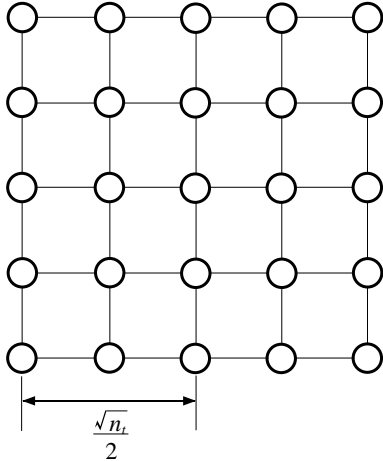


Fig.2.6: 4ch のバスを持っているアクチュエータ



$$\tau_{d1} = \frac{n_t}{2} \tau_p \quad (2.13)$$

の時間遅れを持つ。これは制御系設計に於いては、無駄時間として作用し、制御系の安定性を脅かす。また、4 ch. の場合は、正方形形状に並べるのが最も効率よく、



$$\tau_{d1} = \frac{\sqrt{n_t}}{2} \tau_p \quad (2.14)$$

であり、6 ch. の場合は、直方体形状に並べるのが最も効率よく、無駄時間は、

$$\tau_{d1} = \frac{n_t^{1/3}}{\sqrt{2}} \tau_p \quad (2.15)$$

で表すことができる。

例えば、アクチュエータは 4ch. のバスを持ち、1[MHz] のクロックで同期して動いており、隣のアクチュエータにデータを渡すのに 1 クロック、1[μs] の時間がかかるアクチュエータが 10000 個あった場合、 $\tau_{d1}=50[\mu s]$ となる。この情報配信に伴う無駄時間はそれほど大きくないようである。

多数にアクチュエータによる無駄時間を考える場合、各々のアクチュエータに出し指令値を計算するための無駄時間が支配的のようである。この無駄時間がどの程度かを見積もることが難しいが、非常に似たシチュエーションを参考にして考察する。

各々のアクチュエータで得られた情報を処理するのは、カメラで捉えた画像を処理するのに似ている。アクチュエータで捉えた情報が 1 ピクセルの色やその強さに対応すると考えると分かり易い。

ロボットの先にカメラを付けておく。そのカメラで目標物を捉え、その画像処理をした結果をもとにロボットを動かし目標物をトラッキングするビジュアルサーボと呼ばれるものがある。このビジュアルサーボにおいてもその画像処理に伴う無駄時間が問題となることがある。この無駄時間の量はもちろんだのような処理を行うかより異なるが、だいたい数十 [ms] 程度である。

したがって、 τ_{d2} もおおざっぱにいて、1[ms] でもなければ 10[s] でもなく、数十 [ms] 程度と考えるのが妥当であると思われる。

2.6 アルゴリズムのタイプ

アルゴリズムのタイプは、序論の分散マニピュレーションをめぐる研究状況で紹介したとおり、

- トップダウン方式
- ボトムアップ方式
- トップダウン方式とボトムアップ方式の混合

の 3 タイプが考えられる。

2.7 分散マニピュレーションシステムに関する幾つかの現実的な仮定

ここでは、以上までで考察した分散マニピュレーション基本要件の各項目に関して幾つかの仮定をする。

- 空間周波数による制約
マニピュレーション対象物は非圧縮性物質（剛体）とする。マニピュレーション対象物の密度分布は ρ で、

$$\rho = \rho(x, y, z) \neq \rho(x, y, z, t) \tag{2.16}$$

で、マニピュレーション対象物の空間周波数分布は時不変とする。またマニピュレーション対象物の空間的な変化は比較的穏やかで、アクチュエータ間隔 $\Delta x, \Delta y$ 程度では線形補間ができる程度とする。サンプリング定理により制限されるよりずっと低いところでのみの変化と仮定する。

- 力密度による制限

定常的には力密度に関する制限は受けないとする。各アクチュエータの力は十分大きく、各アクチュエータの担当領域に対して、物体が比較的大きく、慣性力が支配的ならば、

$$g \int \left[\int_{\Omega_{ij}} \rho(x, y, z) dx dy \right] dz \simeq g \Delta x \Delta y \int \rho(x_i, y_j, z) dz \ll f_{ij}^{\max} \quad (2.17)$$

が全てのアクチュエータについて満たされるとする。また物体が小さく、表面力が支配的であったとしても

$$\int_{\Omega_{ij}} \mathbf{t} \cdot \mathbf{n} dx dy \simeq t_{ij} \Delta x \Delta y \ll f_{ij}^{\max} \quad (2.18)$$

が満たされるとする。すなわち、定常的には対象物を自由に操作するだけの力をマニピュレータとして持っているとする。ただし、アクチュエータは無限に力が出せるわけではなく、アクチュエータの最大出力 f_{ij}^{\max} は有限の値であるので、瞬間的に非常に大きな力が必要な場合は考察を要することとする。これは、第4章で実現できるインピーダンスパラメータという点で考察する。

また、次のアクチュエータの基本構造から決まるのであるが、 x, y 方向に発生できる力 F_x, F_y は

$$\begin{aligned} F_x &= -\frac{\partial V}{\partial x} = -g \frac{\partial}{\partial x} \int \rho(x, y, z) z(x, y) dx dy dz \\ &\simeq -g \int \rho \frac{\partial z}{\partial x} dx dy dz \end{aligned} \quad (2.19)$$

$$\begin{aligned} F_y &= -\frac{\partial V}{\partial y} = -g \frac{\partial}{\partial y} \int \rho(x, y, z) z(x, y) dx dy dz \\ &\simeq -g \int \rho \frac{\partial z}{\partial y} dx dy dz \end{aligned} \quad (2.20)$$

でほぼ記述できる。これらの力に比べマニピュレータとマニピュレーション対象物との間の摩擦は小さいが、ゼロではなくマニピュレーション対象物は安定に静止できるとする。

- アクチュエータの基本構造

分散マニピュレーションのアクチュエータは並進タイプでリニアドライブで操作を行

う、Fig.2.7 のような、 xy の 2 次元平面内に z 方向にリニアドライブできるアクチュエータが多数並べてある状況を想定する。各アクチュエータは x, y のそれぞれの方向に $\Delta x, \Delta y$ の間隔で、 $x_{\min} \leq x \leq x_{\max}$ 、 $y_{\min} \leq y \leq y_{\max}$ の領域に並べられている。 i, j 番目のアクチュエータは $(x_i, y_j) = ((i - 1)\Delta x, (j - 1)\Delta y)$ の位置にあり、 z 方向の変位は、 $z_{ij} = z(x_i, y_j) = z((i - 1)\Delta x, (j - 1)\Delta y)$ であるとする。各々のアクチュ

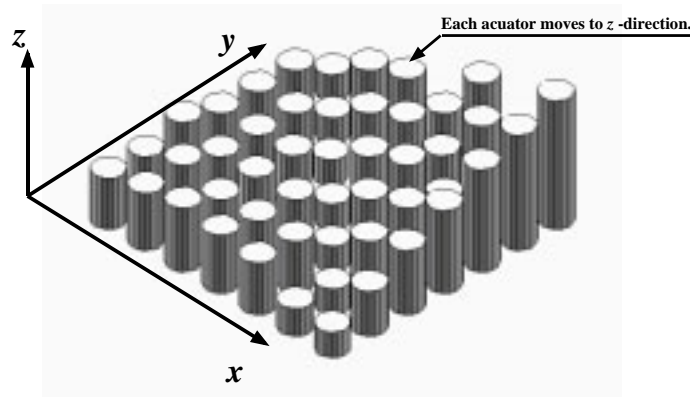


Fig.2.7: 採用する分散マニピュレータ

エータは z 方向に 1 自由度のみもつものに限定する。これは、 z 方向に変位ができるアクチュエータならば、隣接するアクチュエータとの間の勾配で、擬似的に x, y 方向の力も発生させることができるからである。少ない自由度であまりシステムを複雑化させることなく実現できるシステムを採用する。また、回転型アクチュエータ（モータ）タイプにより F_x, F_y を発生するタイプは、MDMS により研究がなされているので、研究に多くの可能性があるリニア型を採用する。

- 無駄時間による制限

無駄時間 τ_d は、情報配信に伴う無駄時間 τ_{d1} と、各アクチュエータに出す指令値の計算のための無駄時間 τ_{d2} の和であった。 τ_{d1} はそれほど大きくなくせいぜい数十 $[\mu s]$ であった。それに対して、 τ_{d2} は大きく支配的で、数十 $[ms]$ 程度は必要となった。

そこで、 τ_d としては $100[ms]$ 程度を仮定する。

- アルゴリズムのタイプ

アルゴリズムのタイプは 3 タイプあった。ボトムアップ方式、トップダウン方式、その混合の好きなものを利用できると仮定する。ただし、ボトムアップ方式は各アクチュエータから機能が発現されるため、局所的な情報しか得られないとし、トップダウン方式は情報を一元的に管理する集中系が存在し、対象物やアクチュエータ全体

に対する情報が得られる。しかし、情報配信や計算時間による無駄時間が生ずるとする。

2.8 あとがき

本章では、分散マニピュレーション法の基本要件について考察した。それらは分散マニピュレーション法がそのマニピュレーションであるが故に存在する本質的なものばかりであって、分散マニピュレーション法の一般的な枠組みを構築するためには外すことができない事柄ばかりであった。

それら基本要件に対して、次章以降の基本動作の実現において、幾つかはあまり一般性を失わない程度に仮定を行うこととし、論理の展開を助けることとした。

第 3 章

計る

3.1 まえがき

本章では基本三動作の一つ『計る』の実現を目指す。制御の分野では外乱オブザーバという技術が良く知られている。摩擦や負荷などの外乱を推定し、その推定値を用いて補償をすることにより、高い制御性能を出す技術である。

外乱量が推定できるこの外乱オブザーバはセンサの代用としても用いることができる。分散マニピュレーションでは多数のアクチュエータで作用するため、きめ細かなマニピュレーションが可能であることは既に述べた通りである。この外乱オブザーバの技術を使えば、容易にセンサレスで

きめ細かなマニピュレーション = きめ細かなセンシング

が実現できる。多数の外乱オブザーバが連携した外乱オブザーバアレイの出力より得られる外乱分布を用いて、マニピュレーション対象物の

- 質量
- 重心位置
- 慣性モーメント

などマニピュレーションに重要な情報の計測を行い、基本三動作の一つ『計る』の実現を目指す。

3.2 対象物の力学的性質

Figs.3.1,3.2 のように分散マニピュレータ上にマニピュレーション対象物があるとする。この対象物の密度は既に定義したとおり、 $\rho(x, y, z)$ で、質量 M 、重心位置 $[x_g, y_g, z_g]^T$ 、

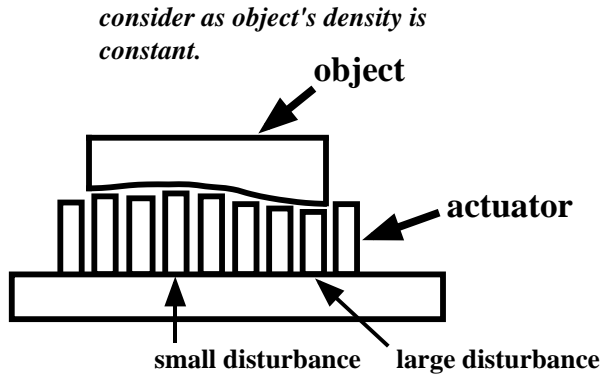


Fig.3.1: 負荷がかかっているときのアクチュエータ (1)

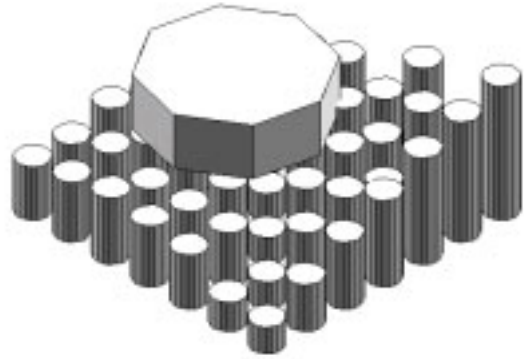


Fig.3.2: 負荷がかかっているときのアクチュエータ (2)

z 軸周りの慣性モーメント J_z はそれぞれ

$$M = \int \rho(x, y, z) dx dy dz \quad (3.1)$$

$$\begin{bmatrix} x_g \\ y_g \\ z_g \end{bmatrix} = \frac{1}{\int \rho(x, y, z) dx dy dz} \begin{bmatrix} \int x \rho(x, y, z) dx dy dz \\ \int y \rho(x, y, z) dx dy dz \\ \int z \rho(x, y, z) dx dy dz \end{bmatrix} \quad (3.2)$$

$$J_z = \int [(x - x_g)^2 + (y - y_g)^2] \rho(x, y, z) dx dy dz \quad (3.3)$$

で定義される [20]。質量は密度を体積積分したものである。また慣性モーメントは重心周りの慣性モーメントを用いることとする。

3.3 外乱オブザーバによる外乱推定とそのアレイ化

マニピュレーション対象物が分散マニピュレータの上ののっているとき、各アクチュエータ側から見れば、のっていない場合と比べ、負荷による外乱が大きくなっている。

i, j 番目のアクチュエータで推定される負荷による外乱量がどのようになるか考えてみる。 i, j 番目のアクチュエータの負荷外乱 $d_{i,j} = d(x_i, y_j)$ は Fig.3.3 のように、 i, j 番目のアクチュエータのすぐ上の対象物の重さとなるだろう。これは対象物を (x_i, y_j) を中心にして、 $\Delta x, \Delta y$ の筒状の部分に対応する。(3.4) で計算できる。

$$\begin{aligned} d_{ij} &= g \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} dx \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} dy \int_{z_1}^{z_2} \rho(x, y, z) dz \\ &\simeq g \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} dx \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} dy \int_{z_1}^{z_2} \rho(x_i, y_j, z) dz = g \Delta x \Delta y \int_{z_1}^{z_2} \rho(x_i, y_j, z) dz \\ &\simeq \hat{d}_{ij} \end{aligned} \quad (3.4)$$

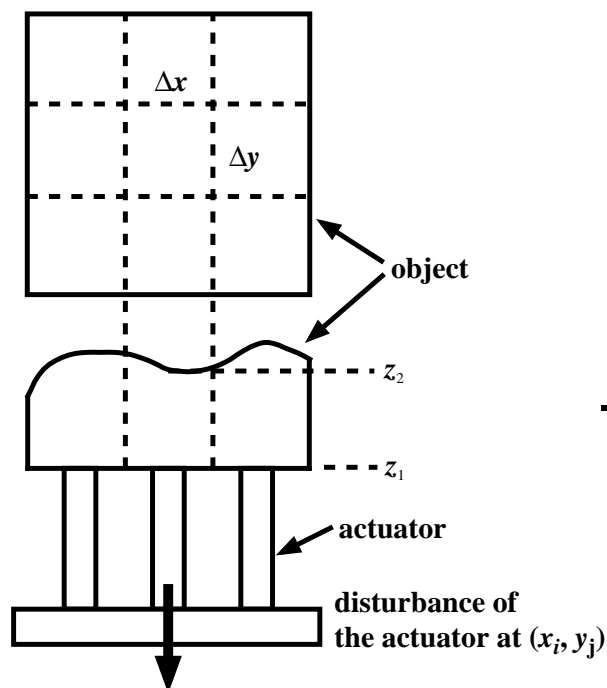


Fig.3.3: 外乱量の計算

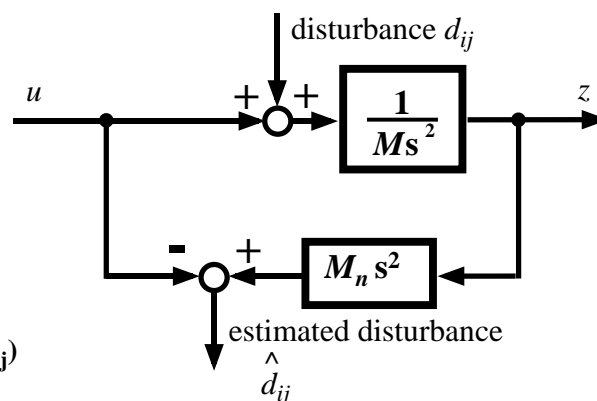


Fig.3.4: 外乱推定ブロック

(3.4) の負荷外乱は力センサなどを用いれば計測可能であるが、 i, j 番目のアクチュエータに、良く知られている Fig.3.4 のような外乱オブザーバ [21], [22] を取り付ければ、簡単に推定外乱 \hat{d}_{ij} を得ることができる。Fig.3.4 はノミナルモデル $\frac{1}{Ms^2}$ を持っていてノミナルモデルと実際の入力から $\frac{1}{Ms^2}$ を通して出てきた出力を比較し、その差は外乱に起因するものとして、外乱量を推定する。(3.4) の外乱はほとんど負荷外乱により生成されたと考えると、外乱オブザーバの出力は

$$\hat{d}_{ij} \simeq d_{ij} \tag{3.5}$$

とすることができ、(3.4) の最後の近似が成り立つ。

3.4 外乱分布から質量、重心位置、慣性モーメントの推定

前節では、 i, j 番目のアクチュエータに外乱オブザーバを付け、そのアクチュエータの上ののっている対象物の (x_i, y_j) を中心に $\Delta x, \Delta y$ の筒状の部分だけ切り取ったに対応する負荷が推定できるようになった。

この外乱オブザーバを全てのアクチュエータに適用し、通常、一つで使う外乱オブザーバをアレイ化する。これにより、各アクチュエータで、外乱オブザーバがそのアクチュエータの上に載っている対象物の (x_i, y_j) を中心に $\Delta x, \Delta y$ の筒状の部分だけ切り取ったに対応する負荷が推定できるから、対象物の密度を z 方向にだけ積分したものの情報が得られる。

(3.1)、(3.2)、(3.3)の重量 M 、重心位置 $[x_g \ y_g \ z_g]^T$ 、慣性モーメント J_z は、推定外乱 \hat{d} を用いて、(3.6)、(3.7)、(3.8) と書き直すことができる。

これらの式を用いれば、外乱オブザーバレイにより推定された外乱分布より容易に対象物の質量、重心位置、慣性モーメントの推定ができる。

得られた (3.6), (3.7), (3.8) をブロック線図で書き表すと Fig.3.5 となる。まずはじめに、質量を推定する。推定質量 \hat{M} は各アクチュエータで得られた外乱量を全てのアクチュエータで足し合わせ、重力加速度で割ればよい。次に重心位置 \hat{x}_g, \hat{y}_g は、各アクチュエータで得られた外乱量にそのアクチュエータの座標を掛け、それを全てのアクチュエータで足し合わせ、最後に先ほど得られた推定質量で割ることにより得られる。慣性モーメント \hat{J}_z は、(3.8) の通りで、推定重心からアクチュエータまでの距離の2乗にそのアクチュエータで推定された外乱量を掛け、全てのアクチュエータで足し合わせ、最後は推定質量で割ることにより、推定慣性モーメントが得られる。

$$\begin{aligned} M &= \int \rho(x, y, z) dx dy dz = \sum_{i,j} \int_{\Omega_{ij}} \rho(x, y, z) dx dy dz \\ &\simeq \sum_{i,j} \int_{\Omega_{ij}} \left[\int \rho(x_i, y_j, z) dz \right] dx dy = \sum_{i,j} \Delta x \Delta y \int \rho(x_i, y_j, z) dz \\ &= \frac{1}{g} \sum_{i,j} d_{ij} \simeq \frac{1}{g} \sum_{i,j} \hat{d}_{ij} = \hat{M} \end{aligned}$$

質量推定式

$$\hat{M} = \frac{1}{g} \sum_{i,j} \hat{d}_{ij} \quad (3.6)$$

$$\begin{aligned} \mathbf{R} &= [x_g \ y_g \ z_g]^T \\ &= \frac{1}{M} \begin{bmatrix} \int x \left[\int \rho(x, y, z) dz \right] dx dy \\ \int y \left[\int \rho(x, y, z) dz \right] dx dy \\ \int z \rho(x, y, z) dx dy dz \end{bmatrix} \simeq \frac{1}{M} \begin{bmatrix} \sum_{i,j} x_i \Delta x \Delta y \int \rho(x_i, y_j, z) dz \\ \sum_{i,j} y_j \Delta x \Delta y \int \rho(x_i, y_j, z) dz \\ \int z \rho(x, y, z) dx dy dz \end{bmatrix} \\ &= \frac{1}{M} \begin{bmatrix} \frac{1}{g} \sum_{i,j} x_i d_{ij} \\ \frac{1}{g} \sum_{i,j} y_j d_{ij} \\ \int z \rho(x, y, z) dx dy dz \end{bmatrix} \simeq \frac{1}{\hat{M}} \begin{bmatrix} \frac{1}{g} \sum_{i,j} x_i \hat{d}_{ij} \\ \frac{1}{g} \sum_{i,j} y_j \hat{d}_{ij} \\ \int z \rho(x, y, z) dx dy dz \end{bmatrix} = \begin{bmatrix} \hat{x}_g \\ \hat{y}_g \\ \hat{z}_g \end{bmatrix} \end{aligned}$$

重心推定式

$$\hat{x}_g = \frac{1}{g\hat{M}} \sum_{i,j} x_i \hat{d}_{ij} \quad (3.7)$$

$$\begin{aligned}
 J_z &= \int [(x - x_g)^2 + (y - y_g)^2] \rho(x, y, z) dx dy dz \\
 &\simeq \sum_{i,j} [(x_i - x_g)^2 + (y_j - y_g)^2] \left[\Delta x \Delta y \int \rho(x_i, y_j, z) dz \right] \\
 &= \frac{1}{g} \sum_{i,j} [(x_i - x_g)^2 + (y_j - y_g)^2] d_{ij} \simeq \frac{1}{g} \sum_{i,j} [(x_i - \hat{x}_g)^2 + (y_j - \hat{y}_g)^2] \hat{d}_{ij} = \hat{J}_z
 \end{aligned}$$

z 軸周りの慣性モーメント推定式

$$\hat{J}_z = \frac{1}{g} \sum_{i,j} [(x_i - \hat{x}_g)^2 + (y_j - \hat{y}_g)^2] \hat{d}_{ij} \quad (3.8)$$

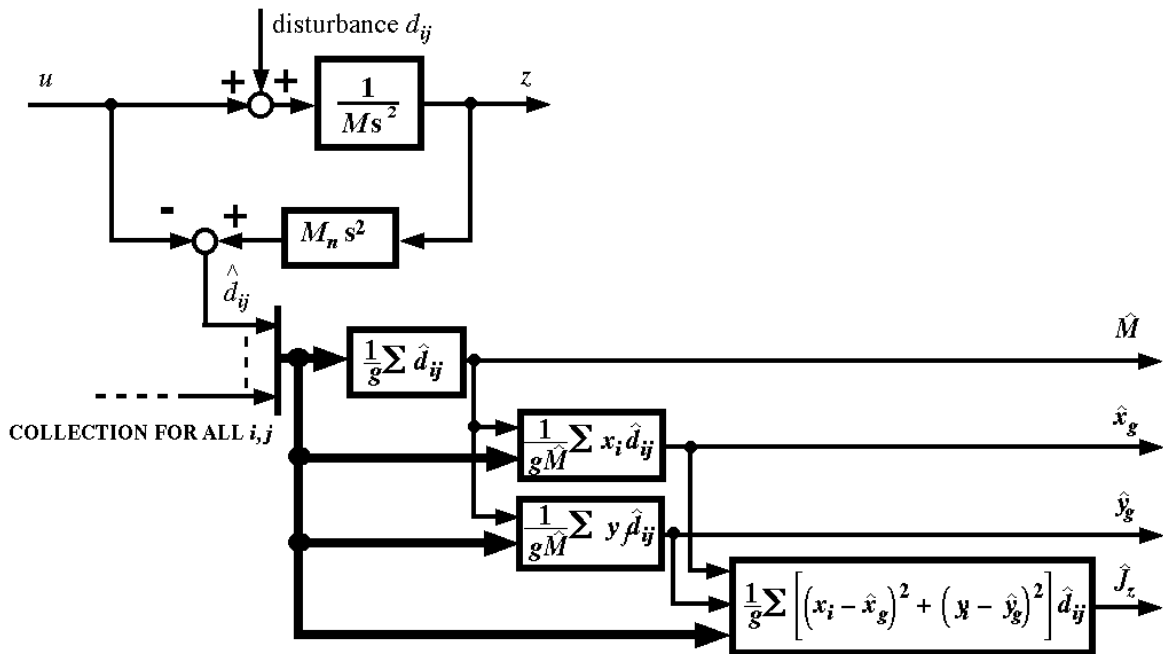


Fig.3.5: 外乱オブザーバレイによる対象物の質量、重心位置、慣性モーメントの推定ブロック線図

3.5 x 軸、 y 軸周りの慣性モーメント

z 軸周りの重心位置を基準とした対象物の慣性モーメントは求めることができた。 x 軸、 y 軸周りも対象物が十分薄いと近似できれば求めることができる ($z - z_g \simeq 0$)。この近似は大抵の場合成り立つ。なぜなら普通 xy 平面内に物体を置く場合、 z 軸周りに大きな慣性モーメントを持つように置くからである。そのように置かないと不安定な置き方となる。

x 軸、 y 軸周りの慣性モーメントは対象物が十分薄いとすれば、それぞれ、(3.9)、(3.10)で得ることができる。

$$\begin{aligned}
 J_x &= \int [(y - y_g)^2 + (z - z_g)^2] \rho(x, y, z) dx dy dz \\
 &\simeq \sum_{i,j} \int_{\Omega_{i,j}} [(y_j - y_g)^2 + (z - z_g)^2] \rho(x_i, y_j, z) dx dy dz \\
 &\simeq \sum_{i,j} \Delta x \Delta y \int (y_j - y_g)^2 \rho(x_i, y_j, z) dz = \frac{\Delta x \Delta y}{g} \sum_{i,j} (y_j - y_g)^2 d_{ij} \\
 &\simeq \frac{\Delta x \Delta y}{g} \sum_{i,j} (y_j - \hat{y}_g)^2 \hat{d}_{ij}
 \end{aligned}$$

x 軸周りの慣性モーメント推定式

$$\hat{J}_x = \frac{\Delta x \Delta y}{g} \sum_{i,j} (y_j - \hat{y}_g)^2 \hat{d}_{ij} \quad (3.9)$$

$$\begin{aligned}
 J_y &= \int [(x - x_g)^2 + (z - z_g)^2] \rho(x, y, z) dx dy dz \\
 &\simeq \frac{\Delta x \Delta y}{g} \sum_{i,j} (x_i - \hat{x}_g)^2 \hat{d}_{ij}
 \end{aligned}$$

y 軸周りの慣性モーメント推定式

$$\hat{J}_y = \frac{\Delta x \Delta y}{g} \sum_{i,j} (x_i - \hat{x}_g)^2 \hat{d}_{ij} \quad (3.10)$$

3.6 推定誤差

推定には必ず誤差というものがある。本節は外乱オブザーバレイによる質量推定誤差、重心位置推定誤差、推定誤差を定量的に見積もる。外乱オブザーバレイによる推定の誤差 Δ_e は、

- アクチュエータ間隔が有限に伴う誤差 Δ_{ac}
 アクチュエータ間隔が有限なため、そのアクチュエータ間隔でしか対象物の情報を得られない。そのため生じる誤差
- 外乱オブザーバ単体の推定誤差の影響 Δ_{obs}
 外乱オブザーバはセンサを使うことなく外乱量を推定する方法である。この外乱オブザーバは推定誤差を含む。そのため生じる誤差

を主に含む。すなわち、

$$\Delta_e = \Delta_{ac} + \Delta_{obs} \quad (3.11)$$

である。

3.6.1 アクチュエータ間隔が有限に伴う誤差

外乱オブザーバアレイにより計測された外乱分布を用いて、重量、重心位置、慣性モーメントの推定を行う場合、外乱オブザーバの推定外乱の誤差と、有限アクチュエータ間隔による誤差がある。本節では、アクチュエータ間隔が有限による誤差を定量的に評価する。

分散マニピュレーションを行う場合、多数の支点を通じてマニピュレーション対象物を操作するといっても、無限のアクチュエータで支えることは不可能である。そのためアクチュエータ間隔は有限の値で、 $\Delta x \neq 0, \Delta y \neq 0$ である。

アクチュエータ間隔が有限なために生じる質量誤差 $\Delta_{ac}M$ 、重心位置誤差 $\Delta_{ac}x_g$ 、 z 軸周りの慣性モーメントの誤差 $\Delta_{ac}J_z$ はそれぞれ (3.12), (3.13), (3.14) で与えられる。

また、 x 軸周りの慣性モーメントの誤差 $\Delta_{ac}J_x$ 、 y 軸周りの慣性モーメントの誤差 $\Delta_{ac}J_y$ はそれぞれ、(3.15)、(3.16) で与えられる。

$$\begin{aligned} \Delta_{ac}M &= \int \rho(x, y, z) dx dy dz - \sum_{i,j} \Delta x \Delta y \int \rho(x_i, y_j, z) dz \\ &= \sum_{i,j} \int_{\Omega_{ij}} [\rho(x, y, z) - \rho(x_i, y_j, z)] dx dy dz \\ &\simeq \sum_{i,j} \int_{\Omega_{ij}} \left[\rho + \frac{\partial \rho}{\partial x} \delta x + \frac{\partial \rho}{\partial y} \delta y - \rho \right] dx dy dz \\ &= \sum_{i,j} \int_{-\frac{1}{2}\Delta x}^{\frac{1}{2}\Delta x} d\delta x \int_{-\frac{1}{2}\Delta y}^{\frac{1}{2}\Delta y} d\delta y \int dz \left[\frac{\partial \rho}{\partial x} \delta x + \frac{\partial \rho}{\partial y} \delta y \right] \\ &= 0 \end{aligned}$$

アクチュエータ間隔が有限なために生じる質量誤差

$$\Delta_{ac}M = 0 \quad (3.12)$$

$$\begin{aligned} \Delta_{ac}x_g &= \frac{1}{M} \int x \rho(x, y, z) dx dy dz - \sum_{i,j} \Delta x \Delta y \int x_i \rho(x_i, y_j, z) dz \\ &\simeq \frac{1}{M} \sum_{i,j} \int_{\Omega_{ij}} (x_i + \delta x) \left[\rho(x_i, y_j, z) + \frac{\partial \rho}{\partial x} \delta x + \frac{\partial \rho}{\partial y} \delta y \right] dx dy dz \\ &\quad - \frac{1}{M} \sum_{i,j} \int_{\Omega_{ij}} x_i \rho(x_i, y_j, z) dx dy dz \\ &= \frac{1}{12M} \Delta x^3 \Delta y \sum_{i,j} \int \frac{\partial \rho}{\partial x} dz \end{aligned}$$

アクチュエータ間隔が有限なために生じる重心位置誤差

$$\Delta_{ac}x_g = \frac{1}{12M} \Delta x^3 \Delta y \sum_{i,j} \int \frac{\partial \rho}{\partial x} dz \quad (3.13)$$

アクチュエータ間隔が有限なために生じる z 軸周りの慣性モーメントの誤差

$$\begin{aligned} \Delta_{ac}J_z &= \frac{1}{12}M(\Delta x^2 + \Delta y^2) - 2M(x_g\Delta_e x_g + y_g\Delta_e y_g) \\ &+ \frac{\Delta x\Delta y}{6} \left(\Delta x^2 \sum_{ij} x_i \int \frac{\partial \rho}{\partial x} dz + \Delta y^2 \sum_{ij} y_j \int \frac{\partial \rho}{\partial y} dz \right) \end{aligned} \quad (3.14)$$

アクチュエータ間隔が有限なために生じる x 軸周りの慣性モーメントの誤差

$$\Delta_{ac}J_x = \frac{1}{12}\Delta y^2 M - 2M y_g \Delta_e y_g + \frac{\Delta x \Delta y^3}{6} \sum_{ij} y_j \int \frac{\partial \rho}{\partial y} dz \quad (3.15)$$

アクチュエータ間隔が有限なために生じる y 軸周りの慣性モーメントの誤差

$$\Delta_{ac}J_y = \frac{1}{12}\Delta x^2 M - 2M x_g \Delta_e x_g + \frac{\Delta x^3 \Delta y}{6} \sum_{ij} x_i \int \frac{\partial \rho}{\partial x} dz \quad (3.16)$$

3.6.2 外乱オブザーバの推定誤差の影響

外乱オブザーバは推定誤差を含む。その推定誤差を (3.17) で定義する。そのとき、外乱オブザーバの推定誤差による、質量推定誤差 $\Delta_{obs}M$ 、重心推定誤差 $\Delta_{obs}x_g$ 、慣性モーメント推定誤差 $\Delta_{obs}J_z$ はそれぞれ (3.18), (3.19), (3.20) で表すことができる。

$$d_{ij} = \hat{d}_{ij} + \Delta d_{ij} \quad (3.17)$$

外乱オブザーバ単体の推定誤差が推定質量に与える影響

$$\frac{\Delta_{obs}M}{M} = \frac{\frac{1}{g} \sum_{ij} d_{ij} - \frac{1}{g} \sum_{ij} \hat{d}_{ij}}{M} = \frac{\sum_{ij} \Delta d_{ij}}{gM} \quad (3.18)$$

外乱オブザーバ単体の推定誤差が推定重心位置に与える影響

$$\frac{\Delta_{obs}x_g}{x_g} = \frac{-\frac{x_g}{M} \Delta_d M + \frac{1}{gM} \sum_{ij} x_i \Delta d_{ij}}{\frac{1}{gM} \sum_{ij} x_i d_{ij}} \quad (3.19)$$

外乱オブザーバ単体の推定誤差が z 軸周りの慣性モーメント推定に与える影響

$$\begin{aligned} \frac{\Delta_d J_z}{J_z} &= \frac{\frac{1}{g} \sum_{ij} [(x_i - x_g)^2 + (y_j - y_g)^2] \Delta d_{ij}}{\frac{1}{g} \sum_{ij} [(x_i - x_g)^2 + (y_j - y_g)^2] d_{ij}} \\ &= \frac{\sum_{ij} (x_i^2 + y_j^2) \Delta d_{ij} - 2gM(x_g \Delta_d x_g + y_g \Delta_d y_d) - (x_g^2 + y_g^2) g \Delta_d M}{\sum_{ij} (x_i^2 + y_j^2) d_{ij} - gM(x_g^2 + y_g^2)} \end{aligned} \quad (3.20)$$

外乱オブザーバ単体の推定誤差が x, y 軸周りの慣性モーメント推定に与える影響

$$\frac{\Delta_d J_x}{J_x} = \frac{\sum_{ij} (y_j - y_g)^2 \Delta d_{ij}}{\sum_{ij} (y_j - y_g)^2 d_{ij}} \quad (3.21)$$

$$\frac{\Delta_d J_y}{J_y} = \frac{\sum_{ij} (x_i - x_g)^2 \Delta d_{ij}}{\sum_{ij} (x_i - x_g)^2 d_{ij}} \quad (3.22)$$

3.7 シミュレーションと実験結果

重さ $M = 1.35[\text{kg}]$ 、長辺の長さ $a = 25.5[\text{cm}]$ 、短辺の長さ $b = 18[\text{cm}]$ 、厚さ $c = 5.5[\text{cm}]$ の直方体の疑似負荷（雑誌）を用意し、その疑似負荷の重量、重心位置、慣性モーメントを計測するシミュレーションと実験を行う（実験装置の詳細は付録を参照）。各アクチュエータはそれぞれ位置制御が施されており、適当な位置指令を入力し（5mm）、そのとき重量、重心位置、慣性モーメントが既知な負荷に対して、得られた外乱分布から提案手法により重量、重心位置、慣性モーメントを推定する。疑似負荷のパラメータ、シミュレーション、実験で用いるパラメータを Table 3.1 に示す。

実験風景を Fig.3.6 に、シミュレーション、実験結果を Table 3.2 に示す。また、シミュレーションにおける負荷外乱分布を Fig.3.7 に、実験における負荷外乱分布を Fig.3.8 に示す。また Table 3.2 のシミュレーション、実験結果には併せて推定誤差 Δ_e も示す。シミュレーションの推定誤差は Δ_e^{sim} で、実験の推定誤差は Δ_e^{ex} で示してある。

実験風景は Fig.3.6 に示した通りであるが、製作した実験装置の上に対象物（雑誌を 2 冊重ねガムテープで縛ったもの）を置いてある。大きさは 12 個のアクチュエータで支えられる程度である。Fig.3.7 は負荷外乱分布のシミュレーション結果であるが、これは理想的な結果といえるだろう。対象物（雑誌）がのせてある部分のみ負荷外乱が、のせてあるアクチュエータで全て同じ値が検出されている。また、対象物（雑誌）がのっていない部分は負荷外乱は完全にゼロとなっている。そのため、非常に綺麗な等高線が得られ、Fig.3.7 となる。

Fig.3.8 は、Fig.3.6 のように実験したときの負荷外乱の等高線である。そもそも実験装置の製作のスキルの問題で、分散マニピュレータ自体の表面がデコボコである。そのためデコの部分に当たる少しオフセットが大きく飛び出た部分は大きな力で対象物を支えることになる。それに対して、ボコの部分にあたるオフセットの小さい部分は比較的小さい力で対象物を支えることになる。また、対象物がない部分のアクチュエータも、ノイズ等の問題により非常に小さい負荷外乱を推定してしまう。そのため、必ずしもゼロになっている

わけではない。そういった理由から、Fig.3.6は必ずしも綺麗な等高線の分布とはなっていない。しかし、対象物がある部分にはもちろん多くの負荷外乱が推定されており、対象物の存在が推測できる結果となっている。

Table 3.2から、必ずしも正確な推定はできないものの、おおざっぱな計測は可能なことがわかる。また、外乱分布から物体の形状もほぼ推定可能なことも読みとることができる。

推定重心位置は推定した推定質量を用いて推定する。推定した結果を用いて推定する。さらに推定慣性モーメントは推定質量を用いて重心位置を推定し、さらにその結果を用いて推定するため、一般に推定質量より推定重心位置、推定慣性モーメントの方が誤差が大きい。Table 3.2にもその傾向が現れており、推定誤差の割合は推定慣性モーメントが一番大きい。

本手法は、必ずしも正確な推定はできないが、おおざっぱな推定は可能という結論をシミュレーションや実験結果から得ることができた。逆にいえば、体重計のように正確に計測はできない。この基本動作『計る』は計ることそれ自身も意味があるが、他の基本動作のサポート的な動作という意味合いがある。計ったデータをもとにして、つかんだり運んだりすることが目的である。そういった意味からも必ずしも正確な体重計のような計測が必要な訳ではない。

ただし、各アクチュエータが負荷量を測定するセンサを持っていれば、外乱オブザーバの代わりに正確な推定ができるので、推定質量、推定重心位置、推定慣性モーメントも正確な推定ができるという、本提案手法の拡張の可能性はある。

3.8 あとがき

本章では、分散マニピュレーションの一般的な知見を得るために、基本三動作の一つ『計る』の実現を目指した。多数あるアクチュエータに外乱オブザーバを組み込むことで、容易な形で外乱の推定が可能となり、外乱分布の計測ができるようになった。得られた外乱分布より、マニピュレーション対象物の、

1. 質量
2. 重心位置
3. 重心周りの慣性モーメント

を得る方法を提案し、シミュレーション、実験結果の両側面から提案手法の有効性を確認した。

Table3.1: シミュレーションと実験で用いたパラメータ

weight M	1.35 [kg]
long side a	25.5 [cm]
short side b	18 [cm]
thickness c	5.5 [cm]
Δx	0.065[m]
Δy	0.065[m]
x_{\min}	0
y_{\min}	0
x_{\max}	$7\Delta x = 0.455$ [m]
y_{\max}	$7\Delta y = 0.455$ [m]
x_g	$4.5\Delta x = 0.2925$ [m]
y_g	$3.0\Delta y = 0.1950$ [m]
J_z	1.09603×10^{-2} [kgm ²]

Table3.2: シミュレーションと実験結果

\hat{M}^{sim}	1.2887[kg]
$\Delta_e M^{sim}$	0.0613[kg]
$(\hat{x}_g^{sim}, \hat{y}_g^{sim})$	$(0.29179, 0.19536)$ [m]= $(4.489\Delta x, 3.006\Delta y)$
$(\Delta_e x_g^{sim}, \Delta_e y_g^{sim})$	$(7.1 \times 10^{-4}, -3.6 \times 10^{-4})$ [m]
\hat{J}_z^{sim}	1.1032×10^{-2} [kgm ²]
$\Delta_e J_z$	-7.17×10^{-3} [kgm ²]

Table3.3: 誤差

\hat{M}^{ex}	1.05289[kg]
$\Delta_e M^{ex}$	0.29711[kg]
$(\hat{x}_g^{ex}, \hat{y}_g^{ex})$	$(0.29682, 0.14926)$ [m]= $(4.566\Delta x, 2.296\Delta y)$
$(\Delta_e x_g^{ex}, \Delta_e y_g^{ex})$	$(-4.32 \times 10^{-3}, 4.574 \times 10^{-2})$ [m]
\hat{J}_z^{ex}	1.5963×10^{-2} [kgm ²]
$\Delta_e J_z^{ex}$	-5.0027×10^{-3} [kgm ²]



Fig.3.6: 実験風景

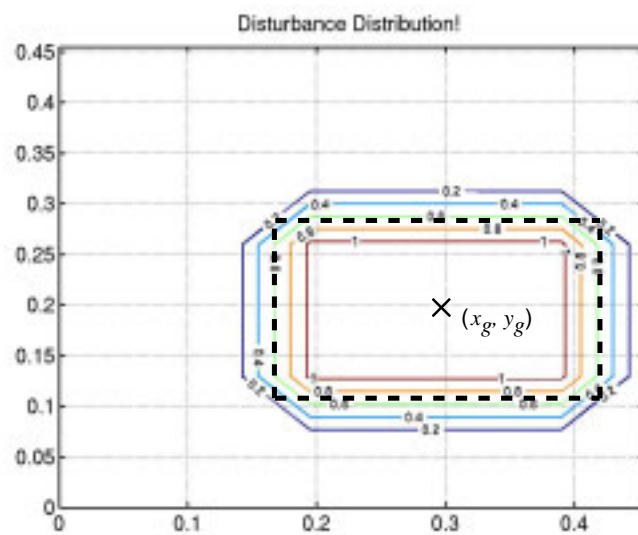


Fig.3.7: 外乱分布 (シミュレーション): [N]

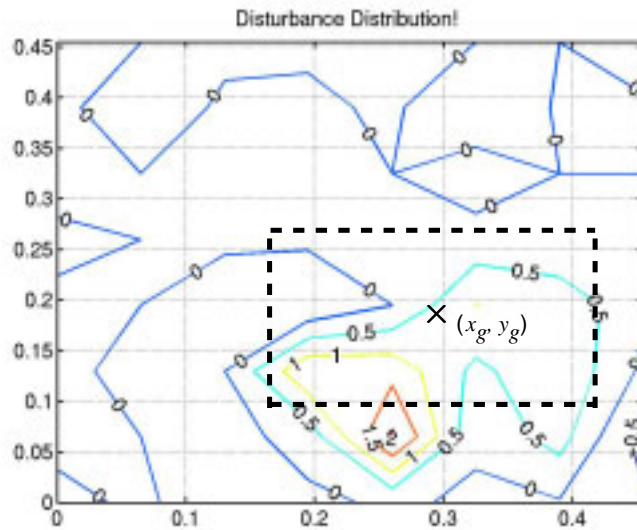


Fig.3.8: 外乱分布 (実験): [N]

結果より、必ずしも正確な値の計測は可能ではないが、おおざっぱな計測であれば十分効果を発揮する手法であるといえる。正確な計測が出来ない理由は、

1. アクチュエータ間隔が有限なことによる誤差
2. 外乱オブザーバの推定誤差

のためである。しかしながら、外乱オブザーバさえあれば適用できる手法であるため、簡単でおおざっぱな計測が必要なときは効果を発揮できる。また、外乱オブザーバの代わりにセンサを用いれば、外乱オブザーバによる推定誤差がなくなり、本手法でも正確な計測ができる。上記、二点の誤差に関しては定式化が可能である。

また、分散マニピュレーションにおいては分散マニピュレーション基本要件というものがあった。本章最後にその点について考察する。『計る』の実現において各項目は、

- 空間周波数による制約
対象物の分布に関する情報は空間周波数による制約を受け、アクチュエータ間隔より2倍より短い波長の対象物の分布に関する情報は観察不可能であった。計る動作においては、アクチュエータ間では対象物の分布は一定と、さらに厳しい近似を施しているため、アクチュエータ間隔に伴う誤差が生じる。
- 力密度による制限
外乱オブザーバによる外乱推定をベースとしているため、外力を出すわけではないから制限を受けない。

- 無駄時間による制限

各アクチュエータの外乱オブザーバから外乱分布を得るために外乱情報を集める。その際、無駄時間が影響するが、外乱情報を集めるのに時間がかかるだけで、安定性などには影響しない(その情報を用いて、フィードバックしているわけではないから)。

- アルゴリズムのタイプ

各アクチュエータで推定された負荷外乱を全て集める必要がある。したがって、アルゴリズムはトップダウン方式となる。

となる。

第 4 章

つかむ

4.1 まえがき

本章では分散マニピュレーションの二つ目の基本動作である『つかむ』の実現を目指す。つかむといっても、手で把持するのではなく、多数のアクチュエータで包み込むように対象物を支えるという意味である。

分散マニピュレーションのきめ細かなマニピュレーションの特長を生かし、高密度の点での支えはほとんど面として作用するだろうから、各アクチュエータにインピーダンス制御を施すことで、面インピーダンス制御を提案する。

面インピーダンス制御を行う上でのポイントは、例えばスポンジやアルファゲルのような衝撃吸収効果のある材料を、アクチュエータ群を用いて、分散マニピュレーションで実現する。対象物をときには柔らかく、ときには堅く、ソフトで自由に設定し、把持（つかむ）することが可能である。インピーダンスはソフトで自由に設定できるから、今までに材料としては、存在しなかったような衝撃吸収効果のある材料を擬似的に実現できる可能性がある。

もう一つのポイントは、やはりソフトで自由に設定できるから、表面形状（つら面）が任意に指定できる。対象物の形状にあわせ、フィッティングを行い、荷重分散を行っての把持ができる。

1. 対象物形状にあわせたフィッティング、荷重分散の実現、ホームポジション z_{ij}^d の設定
2. 疑似衝撃吸収材料の実現、インピーダンスパラメータ m_d, d_d, k_d の設定

4.2 面インピーダンス制御の提案 - インピーダンス制御基本式の導出 -

分散マニピュレーションの特徴である、きめ細かなマニピュレーションの特徴を生かし、各アクチュエータのインピーダンス制御から、アクチュエータ群としては、面にインピー

ダンスを持たせる面インピーダンス制御を提案する。

下記の定式化では、わかりやすさを重要視して、アクチュエータが 3 個並んでいる場合について定式化を行うが、2 次元平面内にアクチュエータが多数並んでいる場合でも、一般的な形で定式化できる。その詳細を次節に示す。

さて、各アクチュエータには位置制御が施されており、各アクチュエータの位置指令値を決定するインピーダンス制御の式は、

$$\begin{aligned}
 m_d s^2 z_1 + d_d s(z_1 - z_1^d) + k_d(z_1 - z_1^d) &= F_1 \\
 m_d s^2 z_2 + d_d s(z_2 - z_2^d) + k_d(z_2 - z_2^d) &= F_2 \\
 m_d s^2 z_3 + d_d s(z_3 - z_3^d) + k_d(z_3 - z_3^d) &= F_3 \\
 (m_d s^2 z_i + d_d s(z_i - z_i^d) + k_d(z_i - z_i^d) &= F_i)
 \end{aligned} \tag{4.1}$$

である。ここで、 m_d, d_d, k_d はそれぞれ実現したい疑似慣性、疑似粘性、疑似弾性で、単位面積あたりでは、 $\frac{m_d}{\Delta x \Delta y}, \frac{d_d}{\Delta x \Delta y}, \frac{k_d}{\Delta x \Delta y}$ となる。また、 z^d はホームポジションであり、外力が加わっていない場合、 z が安定する点である。ここで、

$$\begin{aligned}
 G_d &= m_d s^2 + d_d s + k_d, \quad G_n = d_d s + k_d \\
 z &= \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}, \quad z^d = \begin{bmatrix} z_1^d \\ z_2^d \\ z_3^d \end{bmatrix}, \quad F = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix}
 \end{aligned} \tag{4.2}$$

とおくと、もとのインピーダンス制御の方程式は、

インピーダンス制御基本式

$$G_d z = F + G_n z^d \tag{4.3}$$

となる。この (4.3) が面インピーダンス制御を決める式で、つかむ動作の最も基礎となる方程式である。ブロック線図を Fig.4.1 に示す。このブロック線図の入力は、負荷により生成される負荷力 F_{ij} と負荷力がないときの平衡位置を決定するホームポジション z_{ij}^d である。前もって決められたインピーダンス m_d, d_d, k_d の堅さで、ホームポジション z_{ij}^d を基準に、外力に応じ位置指令が生成され、アクチュエータの変位 (位置) が決まる。さてところで、この基本式は、

- ホームポジション z_d の調節
- インピーダンス m_d, d_d, k_d の可変

の可能性を含んでいる。ホームポジションの調節は荷重分散へとつながり、インピーダンスの可変は衝撃吸収効果へとつながる。

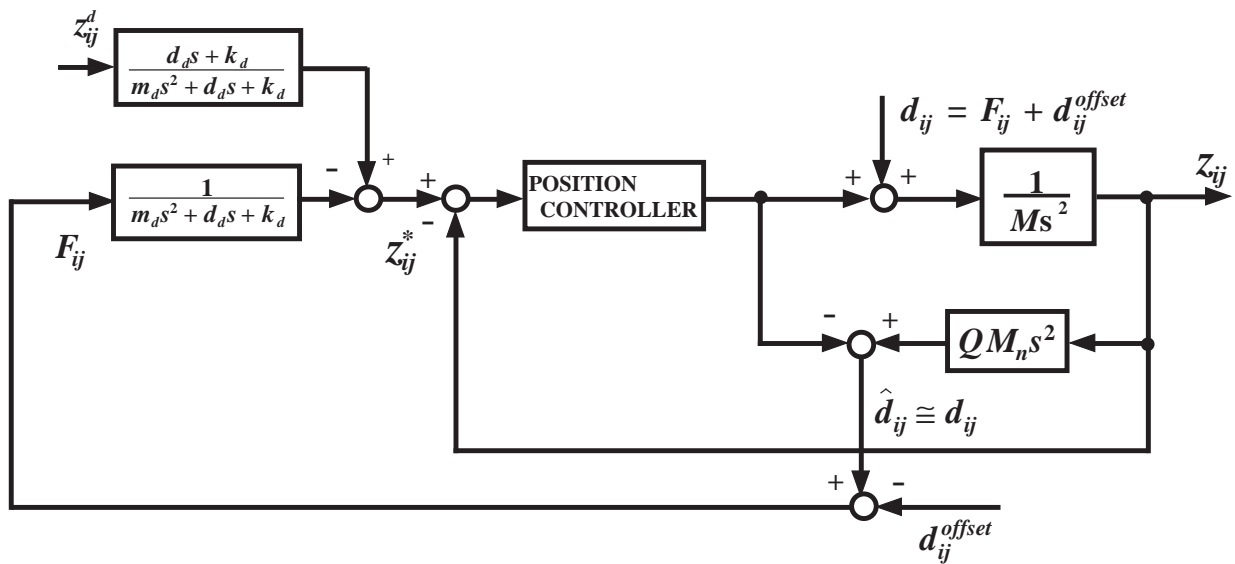


Fig.4.1: インピーダンス制御のブロック線図

4.3 荷重分散

マニピュレーション対象物の形状はまちまちである。デコボコしているものあれば、平らなものもある。『つかむ』動作において、対象物の形状に合わせ、荷重分散を行い全てのアクチュエータで同じくらいの力で把持できれば、対象物を傷つけることがなく、安定した把持に基づく『つかむ』ができる。

4.3.1 可変ホームポジション決定式の導出

対象物の形状に合わせホームポジションを構築し直す。ホームポジションの再構築の方法としては、全てのアクチュエータで出す力がほとんど同じになるように、ホームポジションを変化させる。上手く荷重分散ができている状態では、全てのアクチュエータがほとんど同じ力を発生している。各アクチュエータで発生している力の差はほとんどゼロに等しいはずなので、それぞれのアクチュエータの発生力の差を評価関数 e に反映させる。評価関数を、

$$e = (F_1 - F_2)^2 + (F_2 - F_3)^2 + (F_3 - F_1)^2 \quad (4.4)$$

$$\left(\begin{array}{l} e = (F_1 - F_2)^2 + (F_2 - F_3)^2 + \dots + (F_{imax-1} - F_{imax})^2 \\ = \sum_1^{imax-1} (F_i - F_{i+1})^2 \end{array} \right)$$

とすると、荷重分散ができて、全てのアクチュエータでほとんど同じ力が発生できていると、この評価関数は最小になり、

$$e(z^d) \rightarrow \min_{z^d} \quad (4.5)$$

となっている。このときの停留条件は、

$$\begin{aligned} \frac{\partial e}{\partial z_1^d} = 0, \quad \frac{\partial e}{\partial z_2^d} = 0, \quad \frac{\partial e}{\partial z_3^d} = 0 \\ \left(\frac{\partial e}{\partial z^d} = \mathbf{0} \right) \end{aligned} \quad (4.6)$$

である。

$$\frac{\partial e}{\partial z_1^d} = 2(F_1 - F_2) \frac{\partial F_1}{\partial z_1^d} - 2(F_3 - F_1) \frac{\partial F_1}{\partial z_1^d} = (4F_1 - 2F_2 - 2F_3) \frac{\partial F_1}{\partial z_1^d} = 0 \quad (4.7)$$

$$\frac{\partial e}{\partial z_2^d} = -2(F_1 - F_2) \frac{\partial F_2}{\partial z_1^d} + 2(F_2 - F_3) \frac{\partial F_2}{\partial z_2^d} = (-2F_1 + 4F_2 - 2F_3) \frac{\partial F_2}{\partial z_2^d} = 0 \quad (4.8)$$

$$\frac{\partial e}{\partial z_3^d} = -2(F_2 - F_3) \frac{\partial F_3}{\partial z_1^d} + 2(F_3 - F_1) \frac{\partial F_3}{\partial z_3^d} = (-2F_1 - 2F_2 + 4F_3) \frac{\partial F_3}{\partial z_3^d} = 0 \quad (4.9)$$

$$\left(\frac{\partial e}{\partial z_i^d} = \begin{cases} 2(F_i - F_{i+1}) \frac{\partial F_i}{\partial z_i^d} = 0 & (i = 1) \\ -2(F_{i-1} - F_i) \frac{\partial F_i}{\partial z_i^d} = 0 & (i = i_{\max}) \\ \{2(F_i - F_{i+1}) - 2(F_{i-1} - F_i)\} \frac{\partial F_i}{\partial z_i^d} = 0 & (\text{otherwise}) \end{cases} \right)$$

ちなみに、

$$\frac{\partial F_1}{\partial z_1^d} = \frac{\partial F_2}{\partial z_2^d} = \frac{\partial F_3}{\partial z_3^d} = -k_d \neq 0 \quad \left(\frac{\partial F_i}{\partial z_i^d} = -k_d \neq 0 \right) \quad (4.10)$$

であるから、各停留条件は、

$$\begin{aligned} F_1 - F_2 &= (m_d^2 + d_d + k_d)(z_1 - z_2) - (d_d s + k_d)(z_1^d - z_2^d) \\ &= G_d(z_1 - z_2) - G_n(z_1^d - z_2^d) = 0 \\ -F_1 + 2F_2 - F_3 &= (m_d^2 + d_d + k_d)(-z_1 + 2z_2 - z_3) - (d_d s + k_d)(-z_1^d + 2z_2^d - z_3^d) \\ &= G_d(-z_1 + 2z_2 - z_3) - G_n(-z_1^d + 2z_2^d - z_3^d) = 0 \\ -F_2 + F_3 &= (m_d^2 + d_d + k_d)(-z_2 + 2z_3) - (d_d s + k_d)(-z_2^d + 2z_3^d) \\ &= G_d(-z_2 + z_3) - G_n(-z_2^d + z_3^d) = 0 \end{aligned}$$

となり、

可変ポジション決定式(その1)

$$G_d A z^d = G_n A z \quad (4.11)$$

を得る。ただし、

$$A = [a_{ij}] = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \quad (4.12)$$

$$\left(a_{ij} = \begin{cases} 2(i=j) & -1(i=j-1, i=j+1) & 0(\text{otherwise}) & (2 \leq i \leq i_{\max}) \\ 1(i=j) & -1(i=i+1) & 0(\text{otherwise}) & (i=1) \\ 1(i=j) & -1(i=i-1) & 0(\text{otherwise}) & (i=i_{\max}) \end{cases} \right)$$

である。ここで、 A^{-1} を求め、その A^{-1} を用いて、 z^d を求めることができれば、容易に荷重分散をしたホームポジション z^d を求めることができる。しかし、 $\det A = 0$ であり、逆行列を求めることができない。 A はランク落ちし、逆行列が求まらない訳である。これは、求める解 z^d が一意に定まらないためである。例えば、 z^d と z が解の組であるとき、全ての要素が b の定数ベクトル b を用いて、 $z^d + b$ と $z + b$ も解となるからである。

この冗長性を除去するために、例えば平均ストロークが一定とか、一つのホームポジションは固定とか、といった方程式(4.13)を一本追加する。

$$\mathbf{H}^T z^d = \mathbf{S} \quad (4.13)$$

$$\begin{bmatrix} A \\ \mathbf{H}^T \end{bmatrix} z^d = G_d^{-1} G_n \begin{bmatrix} A \\ \mathbf{0} \end{bmatrix} z + \begin{bmatrix} \mathbf{0} \\ \mathbf{S} \end{bmatrix} \quad (4.14)$$

$$A_h z^d = G_d^{-1} G_n A_0 z + \mathbf{S} \quad (4.15)$$

$$A_h = \begin{bmatrix} A \\ \mathbf{H}^T \end{bmatrix}, \quad A_0 = \begin{bmatrix} A \\ \mathbf{0} \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \mathbf{0} \\ \mathbf{S} \end{bmatrix} \quad (4.16)$$

可変ホームポジション決定式(その2)

$$z^d = A_h^\dagger A_0 G_d G_n^{-1} Q z + A_h^\dagger \mathbf{S} \quad (4.17)$$

を得る。ただし、 A_h^\dagger は A_h が非正方行列なのでMP疑似逆行列[23]である。 Q は、 G_n^{-1} が一次遅れであるのに対して、 G_d は二次で、実現の上で問題があるので、ローパスフィルタである。これにより、評価関数 e を最小にする、すなわち荷重分散をしたホームポジション z^d を求めることができる。ブロック線図で書くと、Fig. 4.2のようになる。

可変ホームポジション構造による荷重分散効果を期待しないブロック線図は前節で示したとおり、Fig.4.1のようになる。Fig.4.1では、ホームポジションの入力はいつも一定の前もって決めた値を入力し、外力に応じた位置指令値によりアクチュエータの変位 z が決定される。

それに対して、Fig.4.2は可変ホームポジション構造をもっており、ホームポジション入力に、全てのアクチュエータの変位 z から全てのアクチュエータで同じくらいの負荷力で

支える荷重分散を実現するための項、 $A_h^\dagger A_0 G_d G_n^{-1} Q z$ と解を一意に決めるための冗長性除去入力 $A_h^\dagger S$ の項がある。

この二つの項により、荷重分散を実現する可変ホームポジションが行われる。

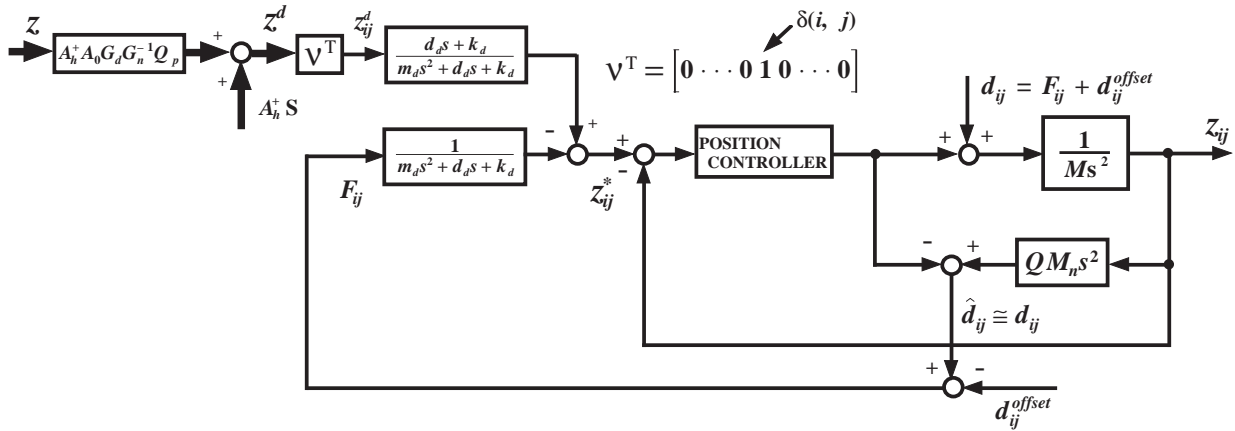


Fig.4.2: 可変ホームポジション構造を含むインピーダンス制御のブロック線図

4.3.2 可変ホームポジション（2次元への拡張）

本節では、アクチュエータが多数、2次元状に並んでいる場合の荷重分散を目的とした可変ホームポジション構造の定式化を目指す。前節では、アクチュエータが1次元状に3個並んでいる場合にのみ定式化を行ったが、本節ではより一般化を目指し、2次元状に多数並んだ場合の定式化を行う。

アクチュエータが2次元状に多数並んでいる場合、各アクチュエータの変位 z_{ij} や各アクチュエータのホームポジション z_{ij}^d のサブインデックスは i, j という風に2次元の広がりを持っている。そのため、例えば、 z, z^d といったベクトルでの取り扱いが非常に不便である。2次元的な広がりのある変数 z_{ij} や z_{ij}^d などをベクトルの列に並べ直さないといけない。

そこで、その並べ直しをスムーズに行うために $\sigma(i, j)$ という関数を導入する。この関数は、2次元的な広がりをもつサブインデックス i, j を1次元の並びに並び直す関数である。ある i, j に対して、1から $i_{\max} \times j_{\max}$ までのある番号が割り当てられる。ただし、他の i, j に対して、割り当てられる番号が重ならないようにする。

まずはじめに(4.3)のようなインピーダンス基本式を2次元的にアクチュエータが並べられている場合について導出する。 i, j 番目のアクチュエータのインピーダンス制御の式は下記の(4.18)のように書くことができる。 $z_{\sigma(i,j)}, z_{\sigma(i,j)}^d, F_{\sigma(i,j)}$ は、 z_{ij}, z_{ij}^d, F_{ij} を後々ベクトルにするときのことを考え、一次元的に並べ直しておく。

$$m_d s^2 z_{\sigma(i,j)} + d_d s [z_{\sigma(i,j)} - z_{\sigma(i,j)}^d] + k_d [z_{\sigma(i,j)} - z_{\sigma(i,j)}^d] = F_{\sigma(i,j)} \quad (4.18)$$

1 次元の場合と同じく、

$$G_d = m_d s^2 + d_d s + k_d, \quad G_n = d_d s + k_d \quad (4.19)$$

とおき、

$$\mathbf{z} = [z_{\sigma(i,j)}], \quad \mathbf{z}^d = [z_{\sigma(i,j)}^d], \quad \mathbf{F} = [F_{\sigma(i,j)}] \quad (4.20)$$

と定める。結局、1 次元の場合と同じく、インピーダンス制御の式は、

$$G_d \mathbf{z} = \mathbf{F} + G_n \mathbf{z}^d \quad (4.21)$$

と書ける。評価関数 e は、

$$\begin{aligned} e = & \sum_{i=1}^{i_{\max}-1} \sum_{j=1}^{j_{\max}-1} \left[\{F_{\sigma(i,j)} - F_{\sigma(i,j+1)}\}^2 + \{F_{\sigma(i,j)} - F_{\sigma(i+1,j)}\}^2 \right] \\ & + \sum_{i=1}^{i_{\max}-1} \left[\{F_{\sigma(i,j_{\max})} - F_{\sigma(i+1,j_{\max})}\}^2 \right] + \sum_{j=1}^{j_{\max}-1} \left[\{F_{\sigma(i_{\max},j)} - F_{\sigma(i_{\max},j+1)}\}^2 \right] \end{aligned} \quad (4.22)$$

となる。(4.22) は煩雑だが、内部のアクチュエータならば、隣接する 4 個のアクチュエータが受けている負荷外乱力が同じようになるように、また端のアクチュエータは 3 個で、角のアクチュエータは 2 個のアクチュエータでそうなるように、エラー関数を定める。

荷重分散ができているときは

$$e(\mathbf{z}^d) \rightarrow \min_{\mathbf{z}^d} \quad (4.23)$$

$$\frac{\partial e}{\partial \mathbf{z}^d} = \mathbf{0} \quad (4.24)$$

で、停留条件はそれぞれ、

$$\frac{\partial e}{\partial z_{\sigma(i,j)}^d} = \{-2F_{\sigma(i-1,j)} - 2F_{\sigma(i,j-1)} + 8F_{\sigma(i,j)} - 2F_{\sigma(i+1,j)} - 2F_{\sigma(i,j+1)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0$$

$$(i = 2, \dots, i_{\max} - 1, j = 2, \dots, j_{\max} - 1)$$

$$\frac{\partial e}{\partial z_{\sigma(i,j)}^d} = \{-2F_{\sigma(i-1,j)} + 6F_{\sigma(i,j)} - 2F_{\sigma(i+1,j)} - 2F_{\sigma(i,j+1)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0 \quad (j = 1)$$

$$\frac{\partial e}{\partial z_{\sigma(i,j)}^d} = \{-2F_{\sigma(i,j-1)} + 6F_{\sigma(i,j)} - 2F_{\sigma(i+1,j)} - 2F_{\sigma(i,j+1)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0 \quad (i = 1)$$

$$\frac{\partial e}{\partial z_{\sigma(i,j)}^d} = \{-2F_{\sigma(i-1,j)} - 2F_{\sigma(i,j-1)} + 6F_{\sigma(i,j)} - 2F_{\sigma(i+1,j)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0 \quad (j = j_{\max})$$

$$\frac{\partial e}{\partial z_{\sigma(i,j)}^d} = \{-2F_{\sigma(i-1,j)} - 2F_{\sigma(i,j-1)} + 6F_{\sigma(i,j)} - 2F_{\sigma(i,j+1)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0 \quad (i = i_{\max})$$

$$\frac{\partial e}{\partial z_{\sigma(i,j)}^d} = \{4F_{\sigma(i,j)} - 2F_{\sigma(i+1,j)} - 2F_{\sigma(i,j+1)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0 \quad (i = 1, j = 1)$$

$$\frac{\partial e}{\partial z_{\sigma(i,j)}^d} = \{-2F_{\sigma(i-1,j)} + 4F_{\sigma(i,j)} - 2F_{\sigma(i,j+1)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0 \quad (i = i_{\max}, j = 1)$$

$$\begin{aligned}\frac{\partial e}{\partial z_{\sigma(i,j)}^d} &= \{-2F_{\sigma(i,j-1)} + 4F_{\sigma(i,j)} - 2F_{\sigma(i+1,j)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0 \quad (i = i, j = j_{\max}) \\ \frac{\partial e}{\partial z_{\sigma(i,j)}^d} &= \{-2F_{\sigma(i-1,j)} - 2F_{\sigma(i,j-1)} + 4F_{\sigma(i,j)}\} \frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = 0 \quad (i = i_{\max}, j = j_{\max})\end{aligned}$$

となる。ここで、

$$\frac{\partial F_{\sigma(i,j)}}{\partial z_{\sigma(i,j)}^d} = -k_d \neq 0 \quad (4.25)$$

だから、結局、停留条件は、

$$\begin{aligned}G_d \{4z_{\sigma(i,j)} - z_{\sigma(i-1,j)} - z_{\sigma(i+1,j)} - z_{\sigma(i,j-1)} - z_{\sigma(i,j+1)}\} \\ -G_n \{4z_{\sigma(i,j)}^d - z_{\sigma(i-1,j)}^d - z_{\sigma(i+1,j)}^d - z_{\sigma(i,j-1)}^d - z_{\sigma(i,j+1)}^d\} = 0 \\ (i = 2, \dots, i_{\max} - 1, j = 2, \dots, j_{\max} - 1) \\ G_d \{3z_{\sigma(i,j)} - z_{\sigma(i-1,j)} - z_{\sigma(i+1,j)} - z_{\sigma(i,j+1)}\} \\ -G_n \{3z_{\sigma(i,j)}^d - z_{\sigma(i-1,j)}^d - z_{\sigma(i+1,j)}^d - z_{\sigma(i,j+1)}^d\} = 0 \quad (j = 1) \\ G_d \{3z_{\sigma(i,j)} - z_{\sigma(i+1,j)} - z_{\sigma(i,j-1)} - z_{\sigma(i,j+1)}\} \\ -G_n \{3z_{\sigma(i,j)}^d - z_{\sigma(i+1,j)}^d - z_{\sigma(i,j-1)}^d - z_{\sigma(i,j+1)}^d\} = 0 \quad (i = 1) \\ G_d \{3z_{\sigma(i,j)} - z_{\sigma(i-1,j)} - z_{\sigma(i+1,j)} - z_{\sigma(i,j-1)}\} \\ -G_n \{3z_{\sigma(i,j)}^d - z_{\sigma(i-1,j)}^d - z_{\sigma(i+1,j)}^d - z_{\sigma(i,j-1)}^d\} = 0 \quad (j = j_{\max}) \\ G_d \{3z_{\sigma(i,j)} - z_{\sigma(i-1,j)} - z_{\sigma(i,j-1)} - z_{\sigma(i,j+1)}\} \\ -G_n \{3z_{\sigma(i,j)}^d - z_{\sigma(i-1,j)}^d - z_{\sigma(i,j-1)}^d - z_{\sigma(i,j+1)}^d\} = 0 \quad (i = i_{\max}) \\ G_d \{2z_{\sigma(i,j)} - z_{\sigma(i+1,j)} - z_{\sigma(i,j+1)}\} \\ -G_n \{2z_{\sigma(i,j)}^d - z_{\sigma(i+1,j)}^d - z_{\sigma(i,j+1)}^d\} = 0 \quad (i = 1, j = 1) \\ G_d \{2z_{\sigma(i,j)} - z_{\sigma(i-1,j)} - z_{\sigma(i,j+1)}\} \\ -G_n \{2z_{\sigma(i,j)}^d - z_{\sigma(i-1,j)}^d - z_{\sigma(i,j+1)}^d\} = 0 \quad (i = i_{\max}, j = 1) \\ G_d \{2z_{\sigma(i,j)} - z_{\sigma(i+1,j)} - z_{\sigma(i,j-1)}\} \\ -G_n \{2z_{\sigma(i,j)}^d - z_{\sigma(i+1,j)}^d - z_{\sigma(i,j-1)}^d\} = 0 \quad (i = 1, j = j_{\max}) \\ G_d \{2z_{\sigma(i,j)} - z_{\sigma(i-1,j)} - z_{\sigma(i,j-1)}\} \\ -G_n \{2z_{\sigma(i,j)}^d - z_{\sigma(i-1,j)}^d - z_{\sigma(i,j-1)}^d\} = 0 \quad (i = i_{\max}, j = j_{\max})\end{aligned}$$

となる。

$$\begin{aligned}
 A &= [a_{\sigma(i,j)\sigma(l,m)}] = \\
 &\left\{ \begin{array}{l}
 4 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i-1, j), (i+1, j), (i, j-1), (i, j+1) \\
 \quad (i = 2, \dots, i_{\max}, j = 2, \dots, j_{\max}) \\
 \\
 3 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i-1, j), (i+1, j), (i, j+1) \\
 \quad (j = 1) \\
 \\
 3 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i+1, j), (i, j-1), (i, j+1) \\
 \quad (i = 1) \\
 \\
 3 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i-1, j), (i+1, j), (i, j-1) \\
 \quad (j = j_{\max}) \\
 \\
 3 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i-1, j), (i, j-1), (i, j+1) \\
 \quad (i = i_{\max}) \\
 \\
 2 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i+1, j), (i, j+1) \\
 \quad (i = 1, j = 1) \\
 \\
 2 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i-1, j), (i+1, j), (i, j-1), (i, j+1) \\
 \quad (i = i_{\max}, j = 1) \\
 \\
 2 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i+1, j), (i, j-1) \\
 \quad (i = 1, j = j_{\max}) \\
 \\
 2 \quad (l, m) = (i, j), \quad -1 \quad (l, m) = (i-1, j), (i, j-1) \\
 \quad (i = i_{\max}, j = j_{\max})
 \end{array} \right. \tag{4.26}
 \end{aligned}$$

とおくと、

$$Az^d = G_n^{-1} G_d Az \tag{4.27}$$

1次元の場合と同じく、 $\det A = 0$ で A^{-1} が求まらないので、その冗長性を除去するために、

$$H^T z^d = S \tag{4.28}$$

$$A_h = \begin{bmatrix} A \\ \mathbf{H}^T \end{bmatrix}, \quad A_0 = \begin{bmatrix} A \\ 0 \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \mathbf{0} \\ S \end{bmatrix} \quad (4.29)$$

という方程式を一本追加する。そうすると、

$$A_h z^d = G_n^{-1} G_d A_0 z + \mathbf{S} \quad (4.30)$$

と書ける。 G_n^{-1} は1次遅れであるのに対して、 G_d は2次なので、 $G_n^{-1} G_d$ は実現の上で問題となる。そこでローパスフィルタ Q を導入して、

$$z^d = A_h^\dagger G_n^{-1} G_d Q A_0 z + A_h^\dagger \mathbf{S} \quad (4.31)$$

を得る。ただしここで A_h^\dagger は A_h のMP疑似逆行列である。結局、1次元の場合と同じように、

インピーダンス制御式

$$G_d z = \mathbf{F} + G_n z^d \quad (4.32)$$

可変ホームポジション決定式

$$z^d = A_h^\dagger G_n^{-1} G_d A_0 z + A_h^\dagger \mathbf{S} \quad (4.33)$$

を得る。

4.3.3 シミュレーションと実験

少数のアクチュエータによる簡単なシミュレーション

Fig.4.3 にアクチュエータ3個の場合のシミュレーション結果を示す。 $\mathbf{H}^T = [1 \ 0 \ 0]$ で $S = 9 \times 10^{-3}[\text{m}]$ である、すなわち、 $z_1^d = 9 \times 10^{-3}$ を与えて固定し、他の z_2^d, z_3^d を調整することにより、荷重分散 $F_1 = F_2 = F_3$ を実現する。また、インピーダンスパラメータと負荷をそれぞれ $m_d = 0, d_d = 200, k_d = 800, 800[\text{g}]$ とした。

初期状態として、 $z_1^d = 9 \times 10^{-3}[\text{m}], z_2^d = 6 \times 10^{-3}[\text{m}], z_3^d = 3 \times 10^{-3}[\text{m}]$ として、外力が加わっていない状態では、アクチュエータ側はデコボコしている状態とする。対象物側はフラットであるが、これはどちらを基準に座標系を作るかの問題である。たまたまアクチュエータ側をデコボコに設定したが、逆に対象物側をデコボコにしても、更に両方がデコボコであっても同様である。

1[s] 後に対象物を分散マニピュレータ上に置く。アクチュエータ側は負荷を受け、その重さに応じ、前もって決められたインピーダンスから位置指令値が計算され、各々のアクチュエータの変位は決まる。1[s] から 5[s] まではただのインピーダンス制御のみだから、デ

コボコのアクチュエータは各々違う負荷を受ける。その様子は Fig.4.3 の下の図で、1[s] から 5[s] までの、各アクチュエータの負荷を比べればよく分かる。

5[s] から可変ホームポジション構造をスタートさせる。Fig.4.3 の下の図から分かるように、5[s] 以降では、見る見るうちに負荷力が等しくなる。これは、ホームポジションが変化しているからであって、ホームポジションは前もって与えられた荷重分散を行う z^d を求めるために、 A のランク落ちを防止するために導入した冗長性除去用入力 $z_1^d = 9 \times 10^{-3}[\text{m}]$ へ変化している。これは Fig.4.3 の上図から確認できる。

荷重分散を目的とした可変ホームポジション構造は、デコボコなアクチュエータ表面上に対象物がのった場合、非常に上手く荷重分散ができるようにホームポジションが変化していることがわかる。

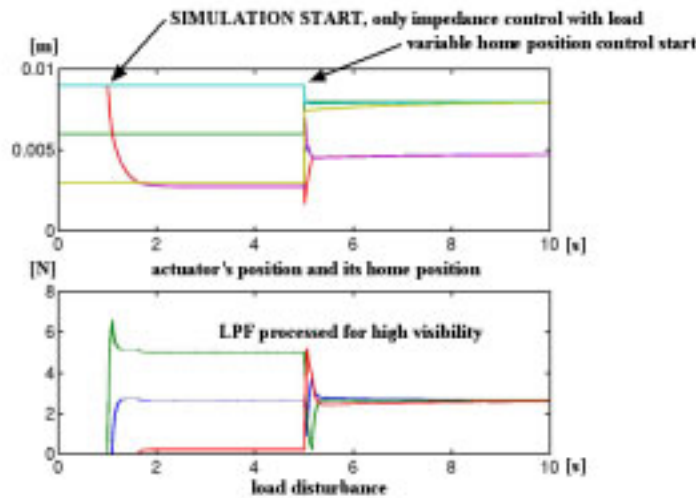


Fig.4.3: アクチュエータ 3 個の場合のシミュレーション結果

2次元で多数のアクチュエータによるシミュレーションと実験

前節のシミュレーションでは、アルゴリズムの基礎的な効果の検証を行うために、アクチュエータが1次元的に3個だけ並んでいるという極めて簡単なシチュエーションだった。もともと、荷重分散は面インピーダンス制御の可変ホームポジション構造によるアルゴリズムである。やはり2次元的にアクチュエータが多数配置されている状態で、対象物を上手く荷重分散しながら把持できるか興味がある。

本節では、前々節で展開した2次元の場合の荷重分散のアルゴリズムに基づき、アクチュエータを2次元平面に 8×8 で64個並べて、何も行わない場合、インピーダンス制御のみ行った場合、更にインピーダンス制御と可変ホームポジションを施した場合の、シミュレー

シミュレーション結果と実験結果を示す。用いたパラメータは『計る』のときと同じように Table 3.1 の通りである。

Figs.4.4, 4.5, 4.6 には制御なし、インピーダンス制御のみ、インピーダンス制御と可変ホームポジションを行った場合の、負荷力の分布のシミュレーション結果を示した。

シミュレーションでは、初期ホームポジションとして乱数を発生することにより得られた数列を用い、デコボコを実現した。Fig.4.4 からわかるように、インピーダンス制御も可変ホームポジション構造も含まない、何もしない位置制御だけの場合は、一点だけで対象物が支えられている。これは、たまたま発生させられた乱数によるホームポジションがその点が一番大きかったためである。この場合、荷重分散効果はゼロである。

Fig.4.5 はただインピーダンス制御のみを施した場合である。デコボコな柔らかい面という形容が正しいと思う。この場合、対象物とインピーダンス制御のパラメータに応じまちまちではあるが、多数のアクチュエータで支えられるため、荷重分散効果は何もしない場合に比べ大きい。デコの部分に当たる、初期状態で与えられたホームポジションが大きいところほど大きな負荷を受けている。

Fig.4.6 は可変ホームポジション構造を含むインピーダンス制御を施した場合である。Fig.4.6 からわかるように、理論上は完全な荷重分散が可能である。支えている全てのアクチュエータで同じ負荷力がかかるようにホームポジションが変化している。

また、Figs.4.7, 4.8, 4.9 には制御なし、インピーダンス制御のみ、インピーダンス制御と可変ホームポジションを行った場合の、負荷力の分布の実験結果を示した。

実験結果では、シミュレーションほど顕著にその効果は現れないものの、Fig.4.7 のインピーダンス制御も可変ホームポジション構造も含まない、何もしない位置制御だけの場合に比べ、Fig.4.8 のインピーダンス制御のみを施した場合の方が荷重分散効果はあり、また更に、Fig.4.9 の可変ホームポジション構造を含むインピーダンス制御の方がさらに効果は高い。

荷重分散の効果を定量的に評価するために、

$$\sum_{all\ i\ j\ with\ load} (d_{ij} - d_{ij}^{ave})^2 \quad (4.34)$$

という、平均値からのばらつきを調べてみる。その結果を Table 4.1 に示す。ただしここで、 d_{ij}^{ave} は負荷外乱の平均値である。

シミュレーション結果、実験結果ともに、インピーダンス制御だけでも荷重分散効果はあるものの、やはり、可変ホームポジションまで施すと可能な限り最大の荷重分散ができている。これは、外乱分布からの定性的な理解からも、平均値からのばらつきの定量的な評価からのどちらかも認めることができる。

また、併せて、Figs.4.10, 4.11, 4.12, 4.13 にはそれぞれ、負荷がかかっていないアクチュエータの応答 (シミュレーション)、負荷がかかっているアクチュエータの応答 (シミュ

Table4.1: 荷重分散効果の定量的評価、平均値からのばらつき

	シミュレーション	実験
制御なし	160.5(Fig.4.4)	3.926(Fig.4.7)
インピーダンス制御のみ	0.5016(Fig.4.5)	2.357(Fig.4.8)
可変ホームポジション	0 (Fig.4.6)	1.816(Fig.4.9)

レーション)、負荷がかかっていないアクチュエータの応答(実験)、負荷がかかっているアクチュエータの応答(実験)を示した。最初から 5.5[s] まではインピーダンス制御も可変ホームポジション構造も含まない、何もしない位置制御だけの場合で、Figs.4.4, 4.7 に対応する。次に、5.5[s] から 10.5[s] まではただのインピーダンス制御のみを施した場合で、Figs.4.5, 4.8 に対応する。最後の 10.5[s] 以降は可変ホームポジション構造を含んだインピーダンス制御で Figs.4.6, 4.9 に対応する。

Fig.4.10 は負荷を受けないアクチュエータの時間応答である。そのため、5.5[s] 以降でインピーダンス制御が始まっても、負荷がないためホームポジションのままである。さらに 10.5[s] 以降の可変ホームポジション構造を含むインピーダンス制御が始まっても変化なしである。

Fig.4.12 は、Fig.4.10 の実験バージョンである。負荷はないので本来なら Fig.4.10 と同じようにアクチュエータの変位はホームポジションの位置のままのはずである。しかし、実験ではノイズ、位置制御の位置のばらつきから等価的に負荷があるように推定されてしまう。そのため、若干アクチュエータの変位がゆらぐ。

Fig.4.11 は対象物をのせるアクチュエータで、5.5[s] 以降で負荷を受ける。シミュレーション結果であるので、非常に綺麗に 10.5[s] 以降も可変ホームポジションによりホームポジションが変化しているのがわかる。

Fig.4.13 は Fig.4.11 の実験バージョンで、対象物がのるアクチュエータの時間応答である。アクチュエータの位置制御の性能があまり良くない。指令値から少しオフセットをもって追従している。簡単な構造の実験装置のため仕方ないが、それでも、10.5[s] 以降ではホームポジションが変化し、荷重分散の効果がそれなりに出ているのは評価できる。

4.4 衝撃吸収効果 [24], [25]

4.4.1 Jerk の 2 乗積分最小化による衝撃吸収

基本三動作の一つ『つかむ』の実現を目指し、面インピーダンス制御を提案した。その面インピーダンス制御のインピーダンス設計問題は重要である。例えば、衝撃吸収効果の



Fig.4.4: 何も制御を施していない場合（位置制御のみ）の負荷の時間平均値を示した等高線（シミュレーション）



Fig.4.5: ただのインピーダンス制御のみを施した場合の負荷の時間平均値を示した等高線（シミュレーション）



Fig.4.6: 可変ホームポジション構造とインピーダンス制御を施した場合の時間平均値を示した等高線（シミュレーション）

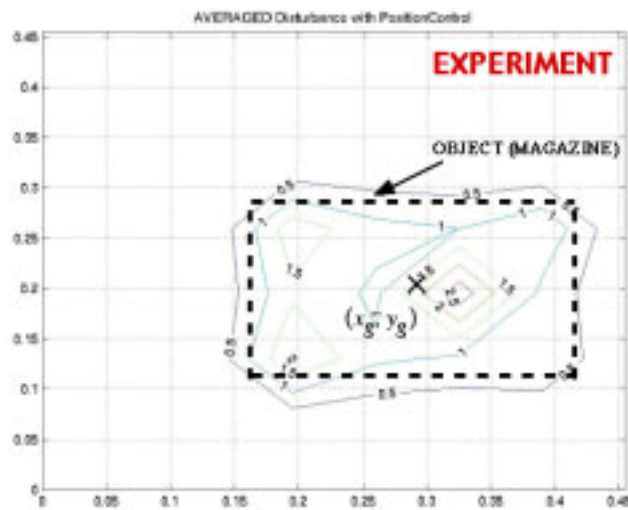


Fig.4.7: 何も制御を施していない場合（位置制御のみ）の負荷の時間平均値を示した等高線（実験）

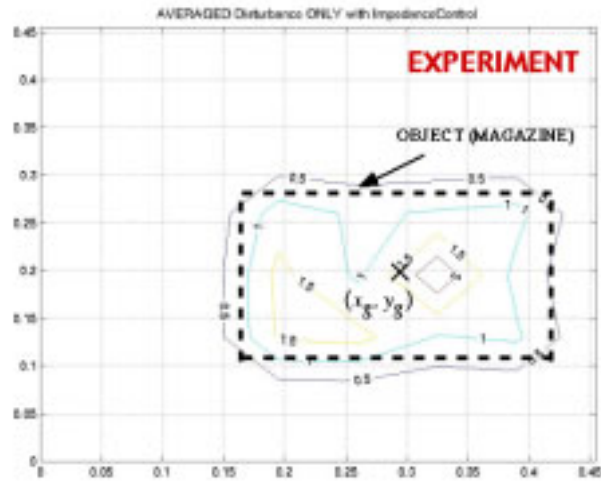


Fig.4.8: ただのインピーダンス制御のみを施した場合の負荷の時間平均値を示した等高線（実験）



Fig.4.9: 可変ホームポジション構造とインピーダンス制御を施した場合の時間平均値を示した等高線（実験）

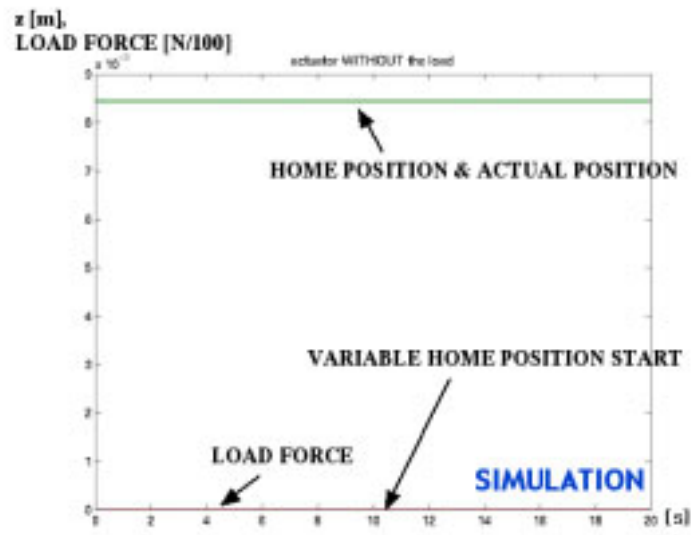


Fig.4.10: 負荷がかかっていないアクチュエータの時間応答 (シミュレーション)

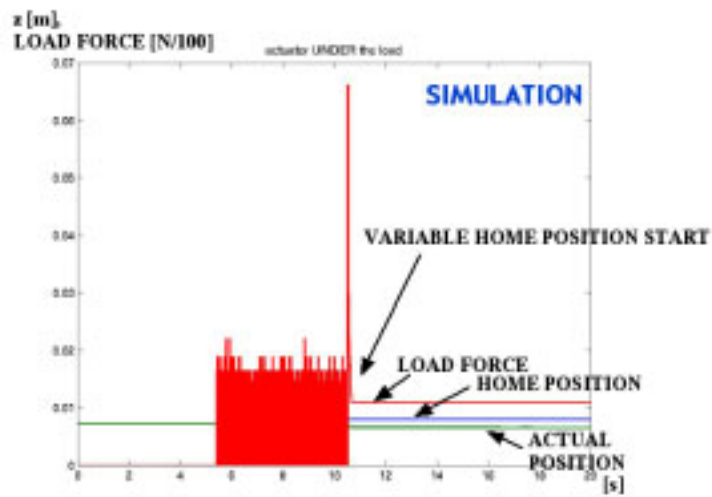


Fig.4.11: 負荷がかかっているアクチュエータの時間応答 (シミュレーション)

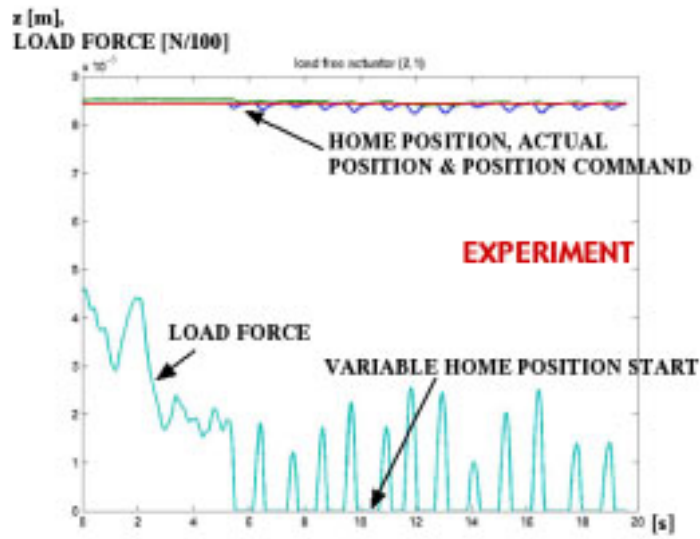


Fig.4.12: 負荷がかかっていないアクチュエータの時間応答（実験）

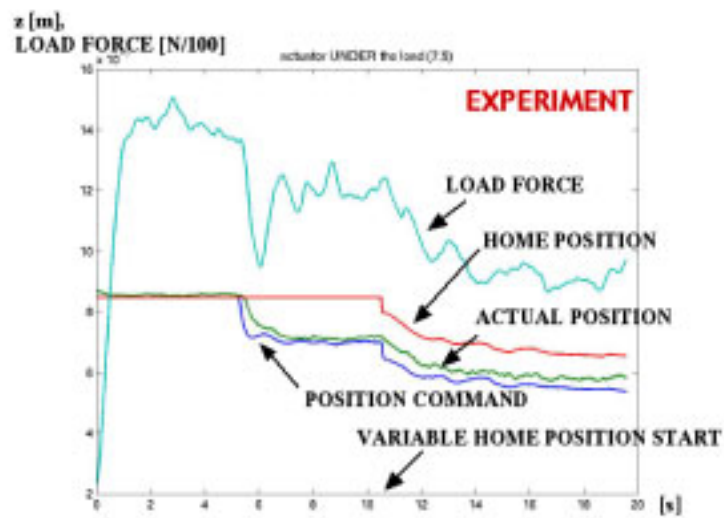


Fig.4.13: 負荷がかかっているアクチュエータの時間応答（実験）

あるスポンジやアルファゲルのような材料と同じインピーダンスパラメータを設定することはもちろん可能で、それによりスポンジやアルファゲルのような材料と同程度の効果は期待できる。ただそれだけでは、現在ある衝撃吸収効果のある材料を越えることができない。せっかくインピーダンスは自由に設定できるのだから、現在ある衝撃吸収材を越えるようなインピーダンスパラメータを設計することを考えたい。

本節では、最も衝撃吸収効果の高い、インピーダンスパラメータを決定する設計法について考察する。なめらかに z の軌道を生成し、衝撃を吸収しつつ対象物を受け止める。そ

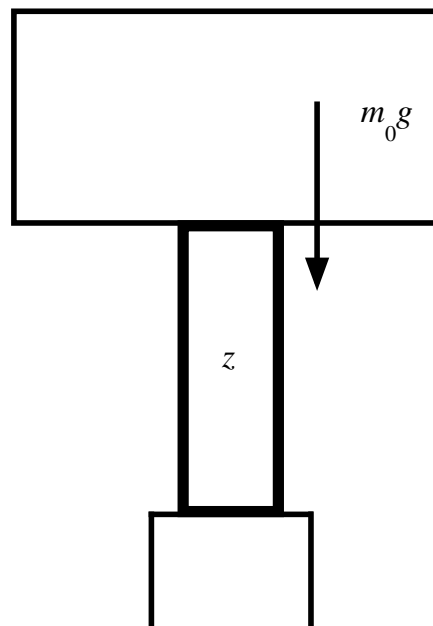


Fig.4.14: 物体を受け止めるとき

んなインピーダンスパラメータを探し出す。

Fig.4.14のようなシチュエーションで、対象物はインピーダンス制御されたアクチュエータの上にかかる。アクチュエータは縮み、どこか平衡位置で落ち着く。

ところで、電車に人が乗っているとき、人に衝撃（ショック）を与えないように加減速したりするにはどうすればいいのか。また、エレベータに人が乗っているとき、人に衝撃（ショック）を与えないように加減速したりするにはどうすればいいのだろうか。

電車の加減速やエレベータの加減速の速度パターンの設計に Jerk（ジャーク）と呼ばれる量がしばしば用いられる。この Jerk は、加速度の時間変化率、加速度の時間微分であり、いかにこの Jerk が発生しないようにするかが、電車の加減速やエレベータの加減速の速度パターンの設計指標となる。

本節の衝撃吸収効果を期待したインピーダンスパラメータの設計にもこの Jerk を設計の指標としてみることにする。Jerk の 2 乗の積分値が最小になるようなインピーダンスパラ

メータを探し出す。

さて、各アクチュエータには、

$$m_d \ddot{z} + d_d (\dot{z} - \dot{z}^d) + k_d (z - z^d) = F \quad (4.35)$$

のインピーダンス制御が施されている。 $t = 0$ で対象物を受け止め始めたとして、Jerkの2乗の積分値を最小にするインピーダンスパラメータというのは、

$$J = \int_0^{T_f} \left(\frac{d\ddot{z}}{dt} \right)^2 dt \Rightarrow \min_{m_d, d_d, k_d} \quad (4.36)$$

ということになる。次節以降では(4.36)を $m_d = 0$ と $m_d \neq 0$ の場合に分けて解く方法について考える。

4.4.2 インピーダンスパラメータの設計 ($m_d = 0$ の場合)

まずはじめは話を簡単にするために、 $m_d = 0$ として、ステップ状の外力入力の大きさを F_0 とすると、

$$Z(s) = \frac{1}{d_d s + k_d} \frac{F_0}{s} \Rightarrow z(t) = \frac{F_0}{k_d} \left(1 - \frac{d_d}{k_d} e^{-\frac{k_d}{d_d} t} \right) \quad (4.37)$$

を得る。ただしここで F_0 は対象物の質量を M とすると、 $F_0 = Mg$ である。

$$J = \int_0^{T_f} \left(\frac{d\ddot{z}}{dt} \right)^2 dt = \frac{F_0 k_d}{2d_d^2} \left(1 - e^{-\frac{2k_d}{d_d} T_f} \right) \quad (4.38)$$

$\min J$ の停留条件は、

$$\frac{\partial J}{\partial d_d} = 0, \quad \frac{\partial J}{\partial k_d} = 0 \quad (4.39)$$

で、計算していくと、結局得られる方程式は、

$$f_1(d_d, k_d) = 1 - \frac{2k_d T_f}{3d_d} - e^{-\frac{2k_d}{d_d} T_f} = 0 \quad (4.40)$$

$$f_2(d_d, k_d) = 1 - \frac{2k_d T_f}{d_d} - e^{-\frac{2k_d}{d_d} T_f} = 0 \quad (4.41)$$

が $\min J$ の条件である。この方程式(4.40), (4.41)は見ての通り、ほとんど似たような形をしている。この両方の方程式を満たす解は、 $k_d \rightarrow 0$ または $d_d \rightarrow \infty$ である。これは当然の結果である。そのまま伸び続けるのが、Jerk最小化の意味ではよいということである。これでは衝撃吸収効果もなければ、物体を停止させることもできない。意味のない解である。次に拘束条件として、

$$z(T_f) = z_{\max} \quad (4.42)$$

というアクチュエータの最大ストローク z_{\max} をつけてみる。すなわち、対象物を受け止めるとき、使えるストロークを最大限使い、その範囲内できっちり止める。ラグランジュア L を用いて、ラグランジュの未定乗数法を用いると、 $\min J$ の条件は

$$L = J + \lambda \{z(T_f) - z_{\max}\} \quad (4.43)$$

$$\frac{\partial L}{\partial d_d} = 0, \quad \frac{\partial L}{\partial k_d} = 0, \quad \frac{\partial L}{\partial \lambda} = 0 \quad (4.44)$$

となり、

$$\begin{aligned} f_1(d_d, k_d) &= -\frac{k_d}{2d_d^2} + \frac{k_d}{2d_d^2}\alpha^2 - \frac{k_d^2}{d_d^3}T_f\alpha^2 - \frac{\lambda}{k_d^2}\alpha + \frac{\lambda}{k_d d_d}T_f\alpha \\ &= 0 \end{aligned} \quad (4.45)$$

$$f_2(d_d, k_d) = \frac{F_0}{k_d} \left(1 - \frac{d_d}{k_d}\alpha\right) - z_{\max} = 0 \quad (4.46)$$

$$\alpha = e^{-\frac{k_d}{d_d}T_f}, \quad \lambda = \frac{\frac{1}{2d_d} - \frac{1}{2d_d}\alpha^2 + \frac{k_d T_f}{d_d^2}\alpha^2}{-\frac{1}{k_d^2} + \frac{2d_d}{k_d^3}\alpha + \frac{T_f}{k_d^2}\alpha} \quad (4.47)$$

を得る。 $\mathbf{x} = [d_d \quad k_d]^T$ 、 $\mathbf{f} = [f_1 \quad f_2]^T$ とすると $\mathbf{f}(\mathbf{x}) = 0$ の求解問題となる。 \mathbf{f} が解析的な関数として与えられているので、 \mathbf{f} のヤコビアンを求めることは簡単である。 $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ が容易に得られるので、 $\mathbf{f}(\mathbf{x}) = 0$ の求解問題は $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ が得られれば、非常に高速に解が得られるニュートン法を用いることにする。

$\mathbf{f}(\mathbf{x}) = 0$ の求解問題、ニュートン法で解く !!

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{x}_n} \right]^{-1} \mathbf{f}(\mathbf{x}_n) \quad (4.48)$$

となる。

ところで、(4.48) で解かれ、得られるインピーダンスパラメータにはある特徴がある。 T_f が十分大きければ、まずはじめに、拘束条件 (4.42) を満たすように、

$$k_d^{opt} = \frac{F_0}{z_{\max}} \quad (4.49)$$

が決まる。この条件で、(4.38) が最小となるように d_d は従属的に決まる。

4.4.3 インピーダンスパラメータの設計 ($m_d \neq 0$ の場合)

$m_d \neq 0$ の場合も、計算は大変なもの、同じような方法で Jerk を最小化するインピーダンスパラメータを求めることができる。2次系の応答は、

$$Z(s) = \frac{1}{m_d s^2 + d_d s + k_d} \frac{F_0}{s} \quad (4.50)$$

のラプラス逆変換より、

$$z(t) = \begin{cases} K \left\{ 1 - e^{-\alpha_1 t} \cos(\alpha_2 t - \alpha_3) \right\} & (0 \leq \xi < 1) \\ K \left\{ 1 - \frac{1}{\beta_1 - \beta_2} (\beta_1 e^{-\beta_2 \omega_n t} - \beta_2 e^{\beta_1 \omega_n t}) \right\} & (\xi > 1) \\ K (1 - e^{\omega_n t} - \omega_n t e^{-\omega_n t}) & (\xi = 1) \end{cases} \quad (4.51)$$

を得る。ただしここで、

$$\begin{aligned} \omega_n &= \sqrt{\frac{k_d}{m_d}}, \quad \xi = \frac{d_d}{2\sqrt{m_d k_d}}, \quad K = \frac{F_0}{k_d} \\ \alpha_1 &= \frac{\xi \omega_n}{\sqrt{1 - \xi^2}}, \quad \alpha_2 = \sqrt{1 - \xi^2} \omega_n, \quad \alpha_3 = \tan^{-1} \frac{\xi}{\sqrt{1 - \xi^2}} \\ \beta_1 &= \xi + \sqrt{\xi^2 - 1}, \quad \beta_2 = \xi - \sqrt{\xi^2 - 1} \end{aligned}$$

である。

$$J = \int_0^{T_f} \left(\frac{d\ddot{z}}{dt} \right)^2 dt \quad (4.52)$$

を計算すると、これが大変ではあるが、

$$J = \begin{cases} \frac{K^2}{4} \left[\frac{\alpha_4 + \alpha_6}{\alpha_1} (1 - e^{-2\alpha_1 T_f}) \right. \\ \left. + \frac{\alpha_4 - \alpha_6}{\alpha_1^2 + \alpha_2^2} \left\{ -\alpha_1 e^{-2\alpha_1 T_f} \cos(2\alpha_2 T_f - 2\alpha_3) + \alpha_2 e^{-2\alpha_1 T_f} \sin(2\alpha_2 T_f - 2\alpha_3) \right. \right. \\ \left. \left. + \alpha_1 \cos(2\alpha_3) + \alpha_2 \sin(2\alpha_3) \right\} \right. \\ \left. + \frac{\alpha_5}{\alpha_1^2 + \alpha_2^2} \left\{ -\alpha_1 e^{-2\alpha_1 T_f} \sin(2\alpha_2 T_f - 2\alpha_3) \right. \right. \\ \left. \left. - \alpha_2 e^{-2\alpha_1 T_f} \cos(2\alpha_2 T_f - 2\alpha_3) - \alpha_1 \sin(2\alpha_3) + \alpha_2 \cos(2\alpha_3) \right\} \right] \quad (0 \leq \xi < 1) \\ \frac{K^2 \beta_1^2 \beta_2^2 \omega_n^5}{(\beta_1 - \beta_2)^2} \left\{ -\frac{\beta_2^3}{2} e^{-2\beta_2 \omega_n T_f} + \frac{2\beta_1^2 \beta_2^2}{\beta_1 + \beta_2} e^{-(\beta_1 + \beta_2) \omega_n T_f} - \frac{\beta_1^3}{2} e^{-2\beta_1 \omega_n T_f} + \frac{\beta_2^3}{2} - \frac{2\beta_1^2 \beta_2^2}{\beta_1 + \beta_2} + \frac{\beta_1^3}{2} \right\} \quad (\xi > 1) \\ K^2 \omega_n^3 \left\{ 2 - \frac{3}{4} \omega_n^2 + \left(\frac{3}{2} \omega_n^3 + \frac{3}{4} \omega_n^2 - 2 \right) e^{2\omega_n T_f} \right\} \quad (\xi = 1) \end{cases}$$

を得る。拘束条件を $C_0 = z(T_f) - z_{\max} = 0$ とし、 $L = J + \lambda C_0$ で、 $\min J$ のための停留条件は、ラクランジュの未定乗数決定法より、

$$\begin{aligned} f_1(\mathbf{x}) &= \frac{\partial L}{\partial m_d} = \frac{\partial}{\partial m_d} (J + \lambda C_0) = \frac{\partial J}{\partial m_d} + \lambda \frac{\partial C_0}{\partial m_d} = 0 \\ f_2(\mathbf{x}) &= \frac{\partial L}{\partial d_d} = \frac{\partial}{\partial d_d} (J + \lambda C_0) = \frac{\partial J}{\partial d_d} + \lambda \frac{\partial C_0}{\partial d_d} = 0 \\ f_3(\mathbf{x}) &= \frac{\partial L}{\partial k_d} = \frac{\partial}{\partial k_d} (J + \lambda C_0) = \frac{\partial J}{\partial k_d} + \lambda \frac{\partial C_0}{\partial k_d} = 0 \\ f_4(\mathbf{x}) &= \frac{\partial L}{\partial \lambda} = \frac{\partial}{\partial \lambda} (J + \lambda C_0) = C_0 = 0 \end{aligned}$$

↓

$$\mathbf{f}(\mathbf{x}) = \mathbf{0}$$

$$(\mathbf{f} = [f_1 \ f_2 \ f_3 \ f_4]^T, \quad \mathbf{x} = [m_d \ d_d \ k_d \ \lambda]^T)$$

$f(\mathbf{x}) = \mathbf{0}$ の求解問題、ニュートン法で解く !!

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{x}_n} \right]^{-1} \mathbf{f}(\mathbf{x}_n) \quad (4.53)$$

可変インピーダンス

拘束条件 (4.42) から規定される (4.49) は満たさないといけない。しかし、可能な限り小さいバネ定数 k_d を設定するのが好ましい。そこで、最終的には (4.49) を満たすような形で、インピーダンスパラメータを時変にすることを考える。時変にした場合の、評価関数 J は

$$\begin{aligned} J &= \int_0^{T_f} \left(\frac{d\ddot{z}}{dt} \right)^2 dz = \int_0^{T_f} \frac{F_0^2 k_d^2}{d_d^4} e^{\frac{2k_d}{d_d} t} dt \\ &= \int_0^{T_1} + \int_{T_1}^{T_2} + \dots + \int_{T_{n-1}}^{T_f} \\ &\simeq -\frac{F_0 k_d(0)}{2d_d^3} \left[e^{-\frac{2k_d(0)}{d_d} t} \right]_0^{T_1} - \frac{F_0 k_d(T_1)}{2d_d^3} \left[e^{-\frac{2k_d(T_1)}{d_d} t} \right]_{T_1}^{T_2} - \dots \\ &\dots - \frac{F_0 k_d(T_{n-1})}{2d_d^3} \left[e^{-\frac{2k_d(T_{n-1})}{d_d} t} \right]_{T_{n-1}}^{T_n} \end{aligned} \quad (4.54)$$

と書き直すことができる。これより、最初、小さいバネ定数を用意し、少しずつ大きくし、最終値として拘束条件 (4.42) から決められる (4.49) の値とすれば、評価関数 J を (4.54) で計算でき、(4.49) の場合と比べ小さくできる。

シミュレーションによる検証

可変インピーダンスの効果をシミュレーションにより検証する。下記、一定値 (4.55)、線形 (4.56)、2 次関数 (4.57)、指数関数 (4.58) の 4 パターンでシミュレーションを行い、そのときの評価関数 J の値と物体の挙動を示す。ただし、このシミュレーションにおいては、 $k_d^{\min} = 1[\text{N/m}]$, $z_{\max} = 5[\text{mm}]$, $F_0 = 0.5[\text{N}]$, $T_f = 10[\text{s}]$ とし、 k_d^{opt} , d_d は (4.48) で解かれた結果で、 $k_d^{\text{opt}} = 99.473[\text{N/m}]$, $d_d = 42.120[\text{Ns/m}]$ である。

Figs.4.15, 4.16 はそれぞれ、可変インピーダンスの時間変化とその可変インピーダンスパラメータ時の対象物の応答である。

Table 4.2 にはそのときの評価関数 J の値を示した。

全ての場合において、可変インピーダンスは一定の k_d と比べ良い結果を示しているが、指数関数的に k_d を増加させる方法がもっとも良い結果を示している。

この結果は直感的にも理解しやすい。インパクトの瞬間に大きな衝撃が生ずるから、そのときはできるだけ柔らかいインピーダンスパラメータを設定し、その後は、堅くし、目標とするストローク内で止める。

$$k_d = k_d^{opt} \tag{4.55}$$

$$k_d = \frac{k_d^{opt} - k_d^{\min}}{T_f} t + k_d^{\min} \tag{4.56}$$

$$k_d = \frac{k_d^{opt} - k_d^{\min}}{T_f^2} t^2 + k_d^{\min} \tag{4.57}$$

$$k_d = (k_d^{opt} - k_d^{\min})(1 - e^{-\frac{t}{10T_f}}) + k_d^{\min} \tag{4.58}$$

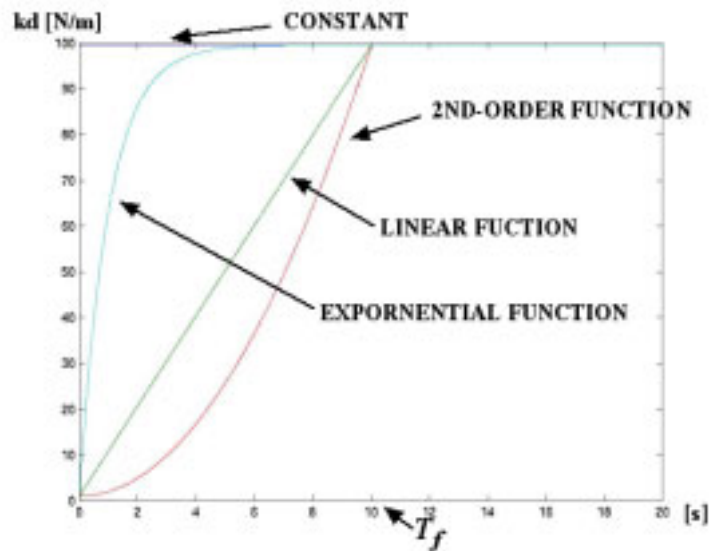


Fig.4.15: 時変でバネ定数を変化させる場合

Table4.2: 時変インピーダンス時の評価関数の値

	評価関数 J
固定	3.33×10^{-4}
線形	6.92×10^{-5}
2次	3.66×10^{-5}
指数	2.31×10^{-5}

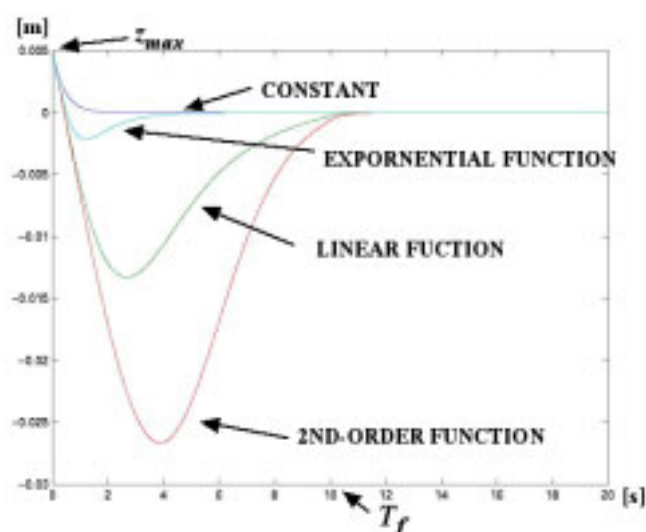


Fig.4.16: バネ定数が時変の場合のアクチュエータの挙動

4.5 あとがき

本章では、分散マニピュレーションの基本動作の二つ目である『つかむ』の実現を目指した。分散マニピュレーションの特長であるきめ細かなマニピュレーションを生かし、高密度の点での作用はほとんど面として作用するから、各アクチュエータでのインピーダンス制御から面インピーダンス制御を提案した。

面インピーダンス制御は、

- 平衡位置を決めるホームポジション
- インピーダンスパラメータ

という二つの設計パラメータがあった。平衡位置を決めるホームポジションを可変にすることにより荷重分散を実現した。また、インピーダンスパラメータは、そのパラメータの設計に Jerk と呼ばれる量を導入することにより、衝撃吸収効果を持つよう設計を行った。面インピーダンス制御による荷重分散、衝撃吸収効果の設計は本章で示したとおりである。

さて、本節あとがきでは、荷重分散や衝撃吸収効果といった事が、第2章の分散マニピュレーションの基本要件の各項目に対してどのような影響を受けるかを検証する。

荷重分散では、無駄時間による制限が問題になると思われるのでその点について以下で考察する。また、衝撃吸収効果では、実現可能インピーダンスパラメータという視点から力密度による制約が支配的となるのでその点について考察する。

4.5.1 荷重分散

インピーダンス制御のホームポジションを可変にすることにより、荷重分散が実現できた。この荷重分散実現のプロセスは、

1. 各アクチュエータの変位を集める。
2. 変位からアクチュエータが受けている力を逆算
3. その力が等しくなるようにホームポジションを再構築

であった。これは、ホームポジションを入力として、変位を出力とするような閉ループ系を作っている。

ところで、第 2 章の分散マニピュレーションの基本要件で示したとおり、分散マニピュレーションはそもそも、無駄時間による制限を受けた。

この無駄時間を含んだ可変ホームポジション系のブロック線図を Fig.4.17 に示す。アクチュエータの数は、場合、場合のよりまちまちであるが、可変ホームポジション構造を含む閉ループ系はこの無駄時間によらず、安定でなくてはならない。

無駄時間に対してロバストであるかどうかを検証したシミュレーション結果を Fig.4.18 に示す。無駄時間として $\tau_d = 500[\text{ms}], 1[\text{s}], 2[\text{s}]$ をそれぞれ設定し、前節同様のシチュエーションで行った。そのときのアクチュエータ 2 の挙動を Fig.4.18 に示した。

無駄時間があったとしても、安定性には全く関係ない結果を得た。これはこの可変ホームポジション構造が仮にどんなに多数のアクチュエータになって、無駄時間が増えたとしても適用できる優れたアルゴリズムであることを示している。

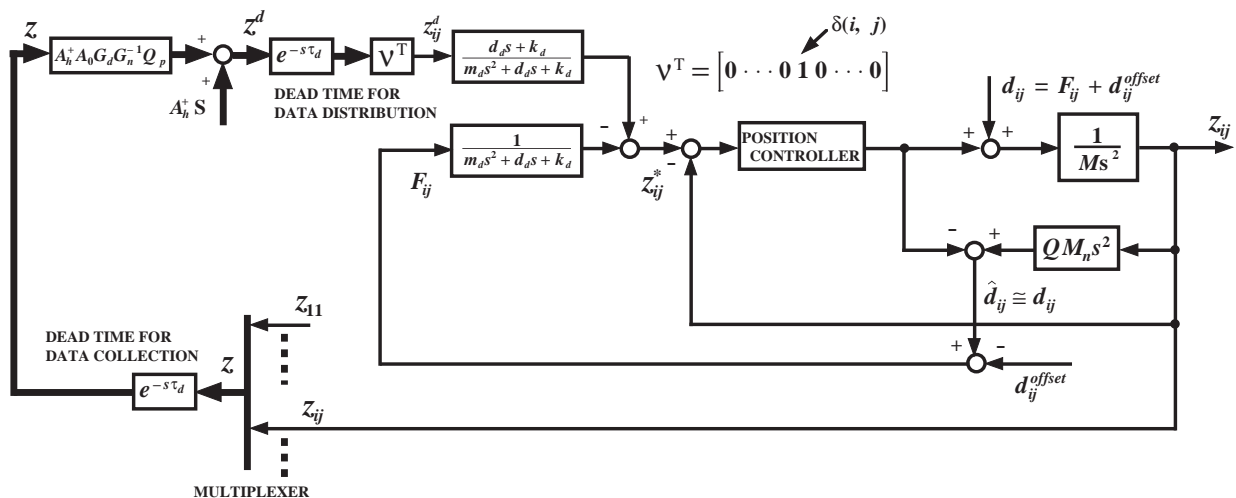


Fig.4.17: 無駄時間を含む場合の可変ホームポジション構造のブロック線図

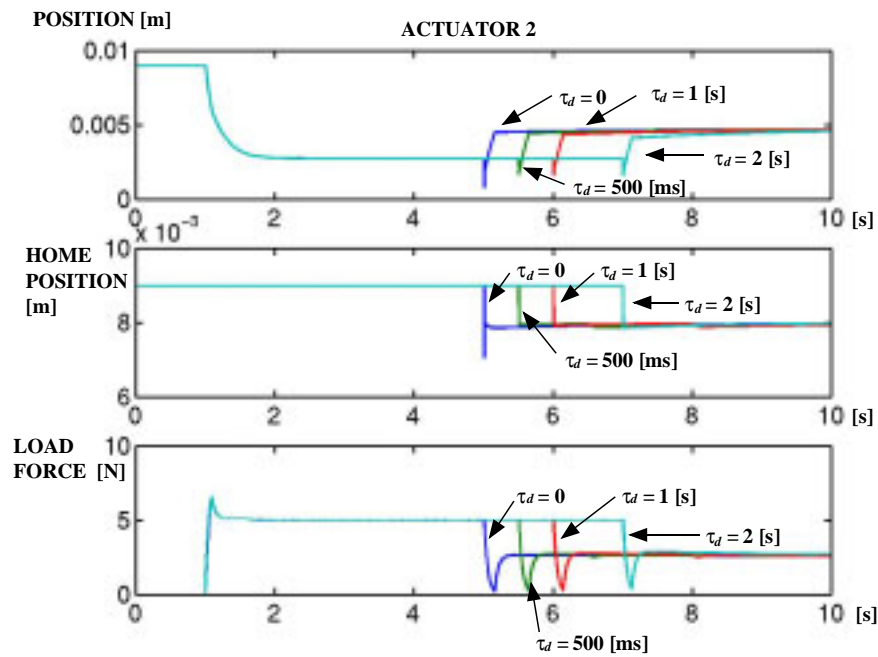


Fig.4.18: 無駄時間を変化させた場合のアクチュエータ 2 の挙動

4.5.2 衝撃吸収効果

(4.38) からわかるように、インピーダンスパラメータの設計で Jerk を最小化するためには、可能な限り小さい k_d を設定することが良い。ただ、拘束条件 (4.42) から、 k_d は規定されてしまう。ところで、分散マニピュレーションはそもそもマニピュレーションの可能性から力密度に関する制限を受けた。この力で、アクチュエータは自分を動かし、インピーダンスを実現する振る舞いをする。だから、この力密度から少なくとも実現できる、インピーダンスパラメータが決まる。また逆に全てのインピーダンスパラメータが実現できるのでなく、出せる力からある範囲内のインピーダンスパラメータのみが実現できるわけである。この実現できるインピーダンスパラメータの最小のバネ定数を k_d^{\min} とおく。これにより可能な限りの衝撃吸収効果を期待することになる。

第 5 章

運ぶ

5.1 まえがき

本章では基本動作の最後である『運ぶ』を実現する。この『運ぶ』では自律分散システム [26], [27], [28], [29] の視点を導入し、各アクチュエータの自律性による局所的な制御とアクチュエータ群全体を制御する集中系の二つを組み合わせた自律分散集中複合系という、マクロ制御とミクロ制御の双方向アルゴリズムの提案を行う。集中系でのアクチュエータ群制御に、フィードフォワードによる方法と、フィードバックによる方法の二つの手法を提案した。それらの手法を適用することにより、搬送物体の位置制御が可能となる。

本章最後では、第 2 章で指摘したように分散マニピュレーション法は本質的に無駄時間を含む。フィードバックによる方法ではその無駄時間により不安定化してしまうことがある。その不安定化を抑えるために、外乱補償器を含んだスミス法を適用する。

5.2 物体搬送システムモデル

運ぶの実現を目指すに当たり、まずはじめに問題を設定する。Fig.5.1 のような分散マニピュレータシステムを想定し、これは Fig.2.7 と構成は同じである、そのシステムによる物体搬送 (『運ぶ』) を実現する。

搬送物体 (ボール) を運ぶ。

$$(x_{ini}, y_{ini}) \Rightarrow (x_c, y_c)$$

分散マニピュレーションでは、対象物の大きさにより、慣性力が支配的になったり、表面力が支配的になったりする。本章の『運ぶ』においては、対象物は比較的大きく、慣性力が支配的となると仮定し、最も簡単なシナリオで重力による物体の転がりによる搬送を目指す。

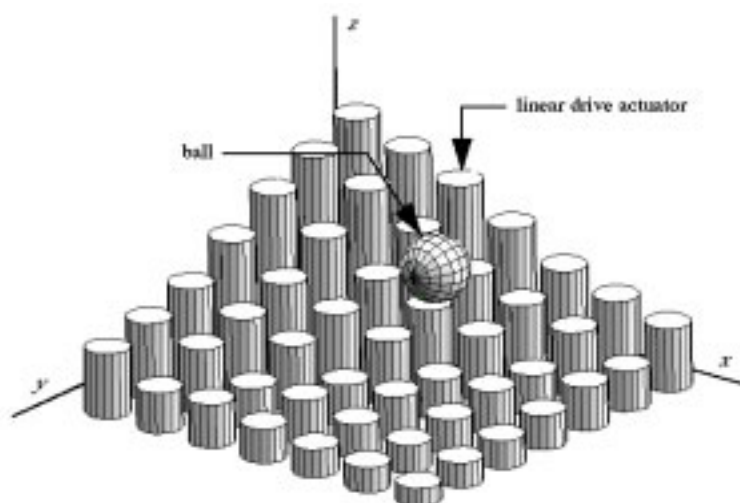


Fig.5.1: 分散マニピュレータによる物体搬送

5.3 場の概念と自律分散集中複合系

分散マニピュレーションは多数のアクチュエータでの操作する。第1章での定義の『多数のアクチュエータにより多数の支点を通じて対象物に作用し操作する方法』からも多数のアクチュエータでの操作が前提であることがわかる。

多数のアクチュエータ（モジュール）によるシステムの設計を考える上で指標となる概念がある。場の概念と呼ばれる概念である。

場の概念 [30], [31] とは、

『自律分散系は一般にたくさんの自律モジュールにより構成される複雑なシステムである。したがって、各モジュールの振る舞い（ミクロ的振る舞い）とたくさんのモジュールで構成される全体の振る舞い（マクロ的振る舞い）の二つを考えなくてはならない。例えば、社会システムを自律分散系と見るならば、各個人の行動や考えがミクロ的振る舞いに対応し、社会の流行や世論などがマクロ的振る舞いに対応する。社会の流行や世論は“場”の一種である。すなわち場とは全体の秩序性のことである。また、ミクロな振る舞いが比較的短い時間スケールであるのに対して、マクロな振る舞いが長いタイムスケールで進行するのも場の特徴である。』

である。

この概念は集中制御系（マクロスケールの制御）と各モジュールの振る舞い（ミクロスケールの制御）との関係を示唆するものであり、各モジュールの振る舞いは比較的速い周期で行うべきであり、また集中制御系での制御周期は比較的ゆっくりであり全体の流れを組んでいくような形であるべきであるということについていっている。すなわち、高速な

ローカルマイナーループと、その外側の集中制御系による制御ループの多重ループの構成を意味する。

この多重ループによる概念を分散マニピュレーションシステムに適用しやすいように、Fig.5.2 のような自律分散集中複合系と名付けるシステムで適用する。

By applying a "field" , we can freely design the interference between neighboring modules.

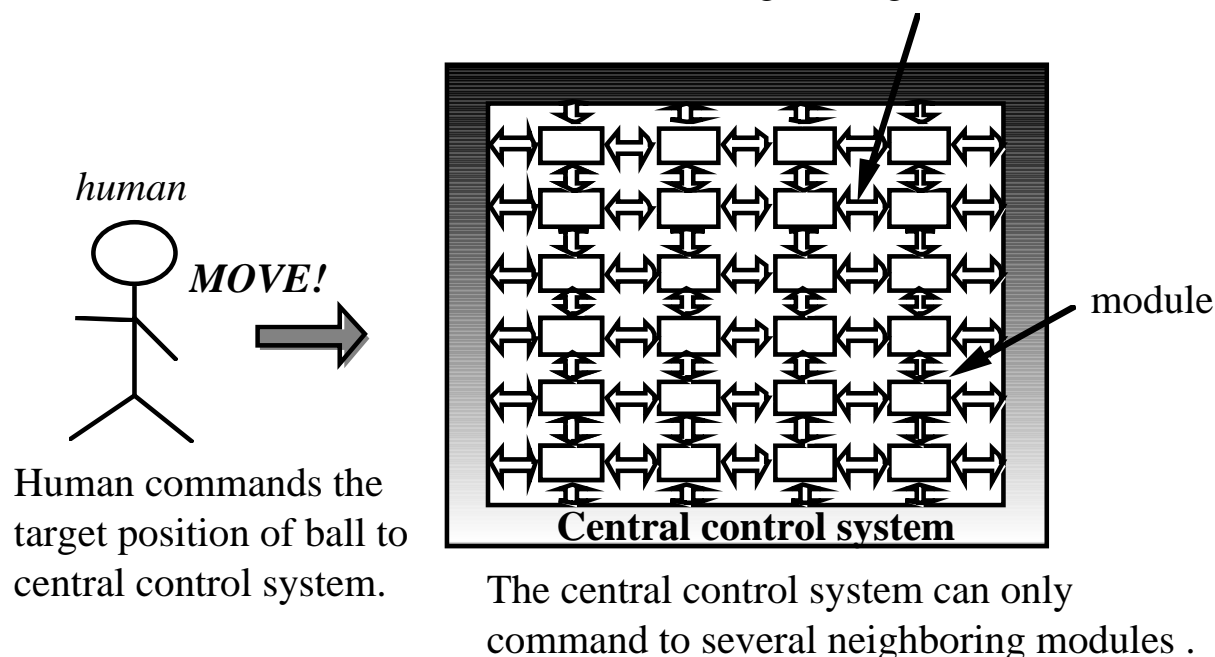


Fig.5.2: 自律分散集中複合系

分散マニピュレーションの設計を考えた場合、大きく分けて2つのことをベースにして考えることができる。

1. 集中系での制御（マクロな制御）

集中系での制御は、ずばり物体搬送すなわち『運ぶ』の制御ループである。対象物の位置制御である。

2. 各アクチュエータでの制御（ミクロな制御）

各アクチュエータでの制御である。アクチュエータ自身の制御をするローカルマイナーループである。アクチュエータの変位の位置制御である。

次節以降では、集中系の制御手法として、フィードフォワードによる方法と、フィードバックによる方法を提案する。また、各アクチュエータでの制御はアクチュエータの構造などにも依存し、一般的な議論が難しい。そこで、製作した実験装置に限りアクチュエー

タの位置制御系の設計と構成を付録 B に示すことにし、ミクロの制御はできているものとしてマクロな制御の議論をする。

5.4 フィードフォワードによる方法

5.4.1 モジュール群の振る舞いと境界条件の設定

物体搬送という目的に対して、目標位置の力学的なポテンシャルを下げてやることにより、物体を搬送することを考える。目標位置の力学的なポテンシャルを下げてやるためには、各アクチュエータは、

1. 自分の座標
2. 目標位置の座標

の情報が必要である。自分の座標と目標位置の座標がわかれば、各モジュールは目標位置を原点とする相対座標系を生成することができる。したがって、自分の働きに何らかの重みづけを行うことができる。では、自分の座標と目標位置を知らせるための場と境界条件（集中制御系の働き）はどのようになるのだろうか。自分の位置を知らせる方法と目標位置を知らせる方法を順を追って説明する。はじめに各モジュールに自分の座標を設定するこ

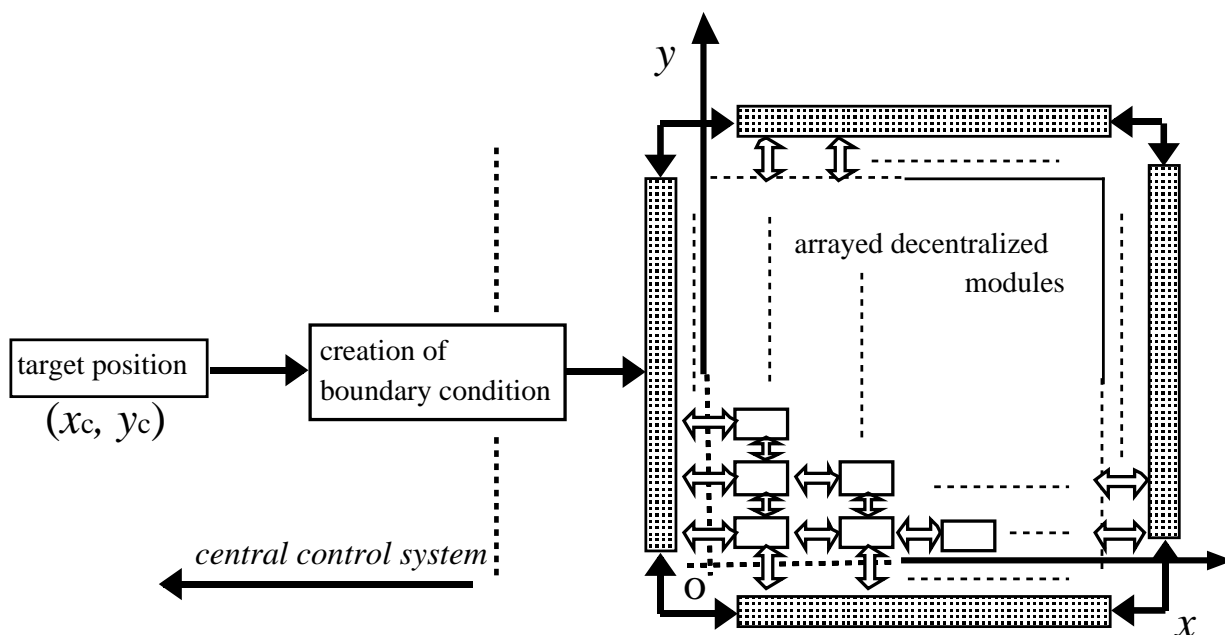


Fig.5.3: フィードフォワードによる方法のブロック線図

とを考えよう。各モジュールの x 座標を設定するための場を $u_x^{cd}(x, y)$ としよう。このとき

$u_x^{cd}(x, y)$ が解、

$$u^{xcd} = x \quad (5.1)$$

$$(x_{\min} \leq x \leq x_{\max}, y_{\min} \leq y \leq y_{\max}) \quad (5.2)$$

を持てば、各モジュールに自分の座標を設定することができる。局所的な通信のみで自分の座標を設定するために通信の局所性を満たすような形で (5.2) のような解を持たせたい。幾つかあるだろうが代表的なものとして、

$$\frac{\partial^2 u^{xcd}}{\partial x^2} + \frac{\partial^2 u^{xcd}}{\partial y^2} = 0 \quad (5.3)$$

$$u^{xcd}(x_{\min}, y) = x_{\min}, u^{xcd}(x_{\max}, y) = x_{\max}, \quad (5.4)$$

$$u^{xcd}(x, y_{\min}) = u^{xcd}(x, y_{\max}) = x$$

を用いよう。(5.3) は各モジュール間に適用される相互干渉を示す式であり、(5.5) は境界条件として集中制御系で設定される値である。このような方法を適用すれば、通信の局所性を満たしながら各モジュールは自分の座標を知ることができる。また、 y 方向も同様に考え u^{ycd} とした。

つぎに各モジュールに搬送物体の目標位置を設定することを考えよう。各モジュールに搬送物体の目標位置の x 座標を設定するための場を $u^{xc}(x, y)$ としよう。このとき $u^{xc}(x, y)$ が解、

$$u^{xc} = x_c$$

$$(x_{\min} \leq x \leq x_{\max}, y_{\min} \leq y \leq y_{\max}) \quad (5.5)$$

を持てば、各モジュールに搬送物体の目標位置を設定することができる。各モジュールに自分の座標を設定したときと同じ様な方法を用いれば、

$$\frac{\partial^2 u^{xc}}{\partial x^2} + \frac{\partial^2 u^{xc}}{\partial y^2} = 0 \quad (5.6)$$

$$u^{xc}(x_{\min}, y) = u^{xc}(x_{\max}, y) = u^{xc}(x, y_{\min}) \quad (5.7)$$

$$= u^{xc}(x, y_{\max}) = x_c$$

となる。これにより各モジュールは搬送物体の目標位置を知ることができる。また、 y 方向も同様に考え u^{yc} とした。

つぎに各モジュールの座標と搬送物体の目標位置の座標から実際に曲面を作ることを考える。各モジュールの座標と搬送物体の目標位置がわかれば搬送物体の目標位置を基準と

した相対座標系を作ることができる。したがって何らかの重みづけをその相対座標系に基づき行うことができる。

最初に示したように搬送物体を目標位置まで運びたいのなら目標位置に穴を空ければ自然と搬送物体は目標位置に転がるだろうというアナロジーのもとに、具体的に相対座標系を用いて考える。穴を空けたければ穴が空くような下記の関数を用いればよい。以上から搬送物体の目標位置 (x_c, y_c) に簡単に穴を空けることができる。

$$\begin{aligned} z(x, y) &= f(\mathbf{u}) = f(u^{xc}, u^{xcd}, u^{yc}, u^{ycd}) \\ &= \frac{\alpha \left\{ (u^{xcd} - u^{xc})^2 + (u^{ycd} - u^{yc})^2 \right\}}{\alpha \left\{ (u^{xcd} - u^{xc})^2 + (u^{ycd} - u^{yc})^2 \right\} + 1} \end{aligned} \quad (5.8)$$

5.4.2 シミュレーションと実験結果

ここでは、フィードフォワードによる方法で行ったシミュレーションと実験結果を示す。Table 5.1 はシミュレーションと実験に用いたパラメータである。実験は実際に実験装置を製作し、その実験装置を用いて行っている。実験装置は 64 個のアクチュエータモジュールにより構成され、その 64 個のアクチュエータモジュールの上にゴムシートを覆い被せ、その上にピンポンボールをおき、その挙動を観察するシステムとなっている。実験装置の構成など詳しくは本論文最後の付録に示す。Figs.5.4, 5.9 は搬送物体（ピンポンボール）の

Table5.1: シミュレーション、実験に用いたパラメータ

Δx	0.065 [m]
Δy	0.065 [m]
n_x	8
n_y	8
(x_{\min}, y_{\min})	(0, 0)
(x_{\max}, y_{\max})	(0.455, 0.455) [m]
(x_{ini}, y_{ini})	$(5.5\Delta x, 5.5\Delta y)=(0.3575, 0.3575)$ [m]
(x_c, y_c)	$(1.5\Delta x, 2.5\Delta y)=(0.0975, 0.1625)$ [m]
α_1	0.5
α_2	50

位置の時間応答でそれぞれシミュレーションと実験結果である。Figs.5.4, 5.9 は時間応答で、初期位置 $(5.5\Delta x, 5.5\Delta y)=(0.3575, 0.3575)$ [m] から定常誤差を伴って目標位置 $(1.5\Delta x, 2.5\Delta y)=(0.0975, 0.1625)$ [m] に収束している。この定常誤差はアクチュエータ間隔 ($\Delta x,$

Δy) に原因がある。アクチュエータ間隔が小さければ小さいほど、この定常誤差も小さくなる。アクチュエータ間隔が搬送物体の位置制御の解像度となることは直感的にもイメージが付きやすい。また、Figs.5.5, 5.10 は搬送物体の軌跡である (は 500[ms] ごとで付けてある)。搬送物体の軌跡の実験結果 Fig.5.10 の 3-5[s] 付近で搬送物体が不自然な挙動を示している。これは実験装置の制作上の問題に起因している。実験装置は自作のため、高精度に作ることは技術的に不可能である。そのため、分散マニピュレータ表面に多少の凸凹がある。搬送物体の不自然な挙動はこの表面の凸凹な部分を搬送物体がそのとき移動したため、その結果として不自然な挙動を示している。Figs.5.6, 5.7, 5.8 はそれぞれ 0.00[s], 3.79[s], 14.19[s] のときの分散マニピュレータの挙動のシミュレーションである。また、Figs.5.11, 5.12, 5.13 はそれぞれ、実験をはじめてすぐ、中間、終わりのときのカーペットの挙動を撮影したものである。

シミュレーション結果 Figs.5.4, 5.5 や実験結果 Figs.5.9, 5.10 に共通で見られるオーバーシュートや振動は搬送物体が本来持つ固有の振動でここで提案しているフィードフォワード手法では避けることができない。そこで次節では搬送物体の位置を推定にもとづく搬送物体のフィードバック位置制御を実現し、フィードバックコントローラによりオーバーシュートや振動を押さえることを考える。

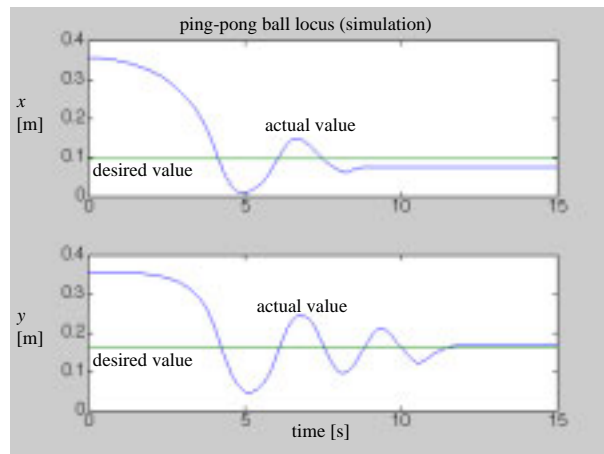


Fig.5.4: 対象物の位置に関する時間応答 (シミュレーション) 上図: x 座標の応答、下図: y 座標の応答

5.5 フィードバックによる方法

前節では、フィードフォード的手法により、搬送物体の目標位置の力学的なポテンシャルを下げてやることにした。搬送物体が自然と低いポテンシャルに落ち込むことにより搬送を

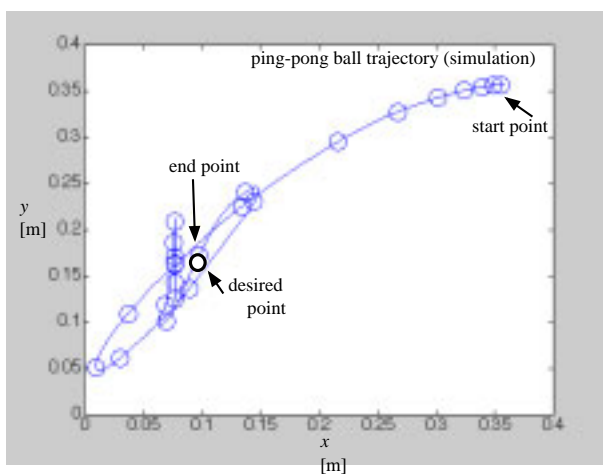


Fig.5.5: 対象物の軌跡 (シミュレーション)

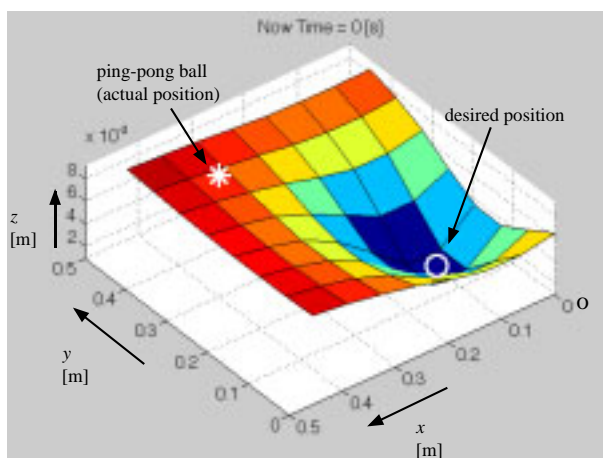


Fig.5.6: 0.00 [s] での表面曲面 (シミュレーション)

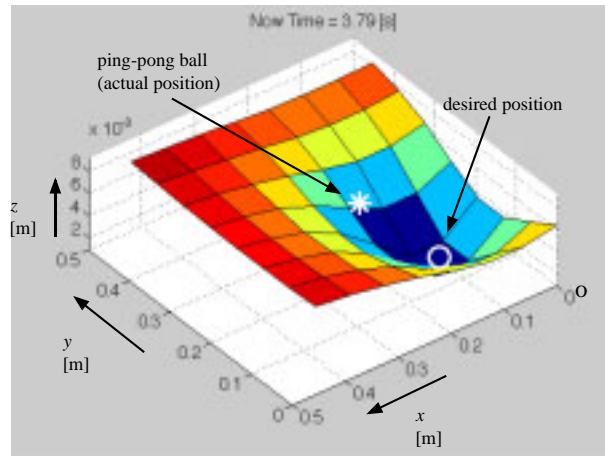


Fig.5.7: 3.79 [s] での表面曲面 (シミュレーション)

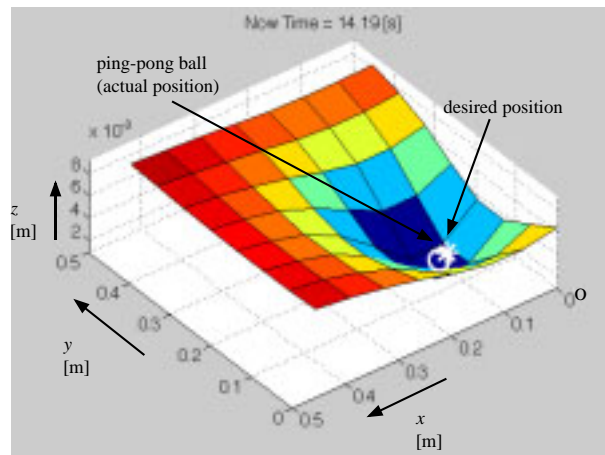


Fig.5.8: 14.19 [s] での表面曲面 (シミュレーション)

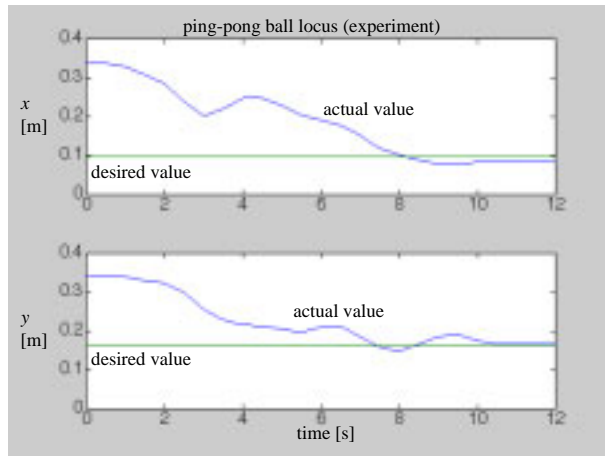


Fig.5.9: 対象物の位置に関する時間応答（実験）上図： x 座標の応答、下図： y 座標の応答

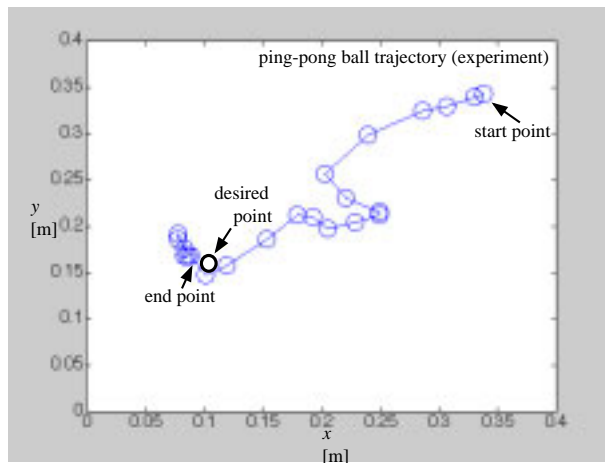


Fig.5.10: 対象物の軌跡（実験）

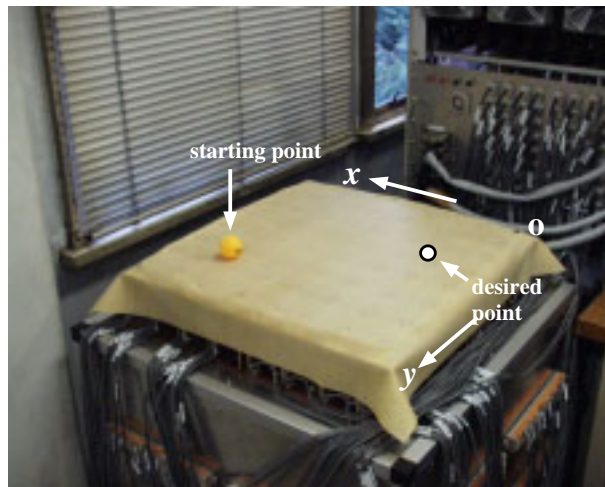


Fig.5.11: 実験開始時の実験装置の状態（対象物は初期位置）

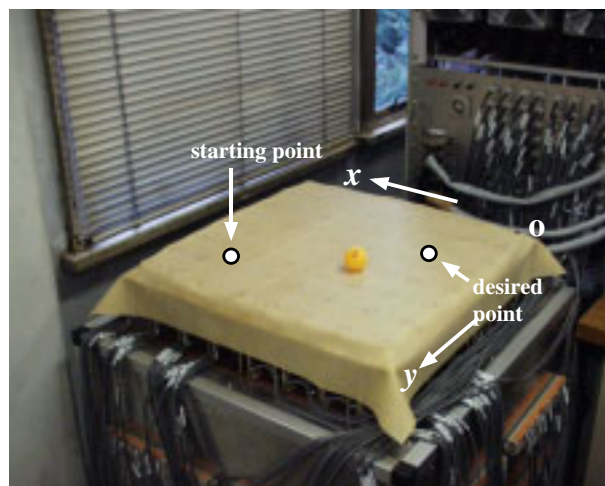


Fig.5.12: 実験途中での実験装置の状態（対象物は目標位置へ搬送中）

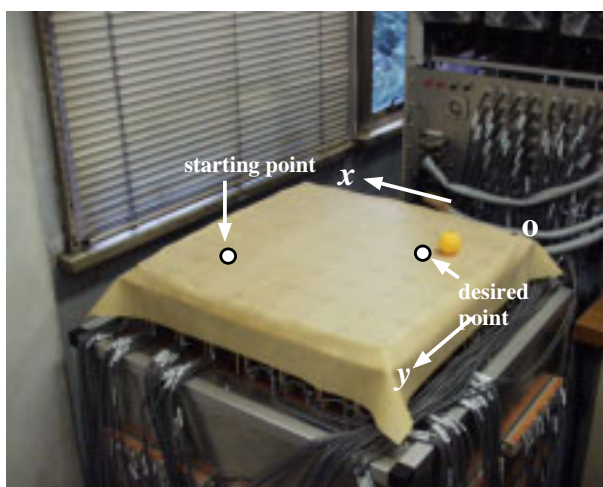


Fig.5.13: 実験終了時の実験装置の状態（対象物は目標位置）

実現した。本節では個々のアクチュエータの外乱オブザーバで推定された負荷外乱を積極的に利用することにより、搬送物体の位置を推定し、その推定結果をフィードバックすることにより、搬送物体の位置制御を実現する。搬送物体の位置応答特性は位置制御フィードバックコントローラによって設計ができる。本節で構成するシステムのブロック線図は Fig.5.15 に示すが、主に

1. アクチュエータモジュール群の振る舞いと境界条件の設定
2. 推定された負荷外乱を用いた位置の推定
3. フォードバックコントローラの設計

について設計を行う。

5.5.1 モジュール群の振る舞いと境界条件の設定

各モジュール群は各々勝手に振る舞っているわけであるが、何らかの形で協調をしていなくては物体搬送という目的を達成することはできない。そのため協調のアルゴリズムは集中系とのやりとりが簡単であり、強度に機能を限定化した自律モジュールを協調させられるものでなくてはならない。

自律分散モジュールはまた通信の局所性をみたくなくてはならない。そこで、集中系とのやりとりが簡単であり、なおかつ自律分散モジュールの通信の局所性が両方とも満たせる物理的場の適用を考える。Laplace 方程式を用いて分散モジュールの秩序性を考えてみる。

$$\frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2} = 0 \quad (5.9)$$

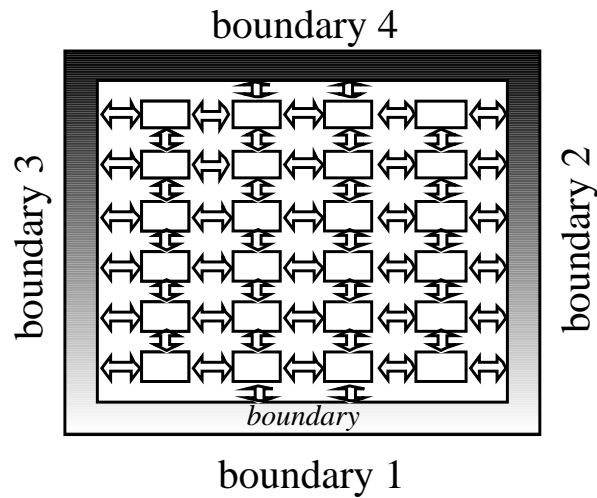


Fig.5.14: 境界条件の生成法

この方程式はよく知られているように周りの値の平均を取っている。物理的には非常に簡単ではあるが、これを上手く利用すると自律モジュールの制御が可能になり、これが場の生成ルールである。

次に境界条件の生成ルールを考える。(5.9)の解は調和関数であり、内部に極大、極小を持たないことが一般に知られているから、以下のロジックにより境界条件を生成する。

- case1 $x_{inp} > 0$ & $y_{inp} > 0$
 $z(\text{onBoundary}3) = kx_{inp}, z(\text{onBoundary}1) = ky_{inp}$
- case2 $x_{inp} < 0$ & $y_{inp} > 0$
 $z(\text{onBoundary}2) = kx_{inp}, z(\text{onBoundary}1) = ky_{inp}$
- case3 $x_{inp} > 0$ & $y_{inp} < 0$
 $z(\text{onBoundary}3) = kx_{inp}, z(\text{onBoundary}4) = ky_{inp}$
- case4 $x_{inp} < 0$ & $y_{inp} < 0$
 $z(\text{onBoundary}2) = kx_{inp}, z(\text{onBoundary}4) = ky_{inp}$

ただし、ここで用いた k は正の定数とする。

5.5.2 負荷外乱を用いた搬送物体の位置の推定

ここでは、搬送物体の位置推定を分散的に行う方法を考える。実際に何らかの方法で物体を目標位置まで運ぶためには目標位置と現在位置についての情報が必要となる。物体

の位置推定には分散モジュールの場を生成することにより行う。生成する場は秩序構造と非常に類似性のある Poisson 場である。

$$\frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2} = f(x, y) \quad (5.10)$$

同次項 $f(x, y)$ の設定を工夫することにより、境界上のみで物体の現在位置を推定することができる。同次項の設定は各アクチュエータの外乱オブザーバで推定された負荷外乱出力を用いる。

5.5.3 搬送物体のフィードバック位置制御コントローラの設計

境界条件生成への入力から実際の搬送物体の位置までの伝達関数は 2 重積分の形になる (境界条件を生成し入力するところは力の次元と同じになる)。

$$\frac{1}{m_x^{eqv} s^2}, \quad m_x^{eqv} = \frac{2(x_{\max} - x_{\min})^2}{\{1 + 2(x_{\max} - x_{\min})\}g} \quad (5.11)$$

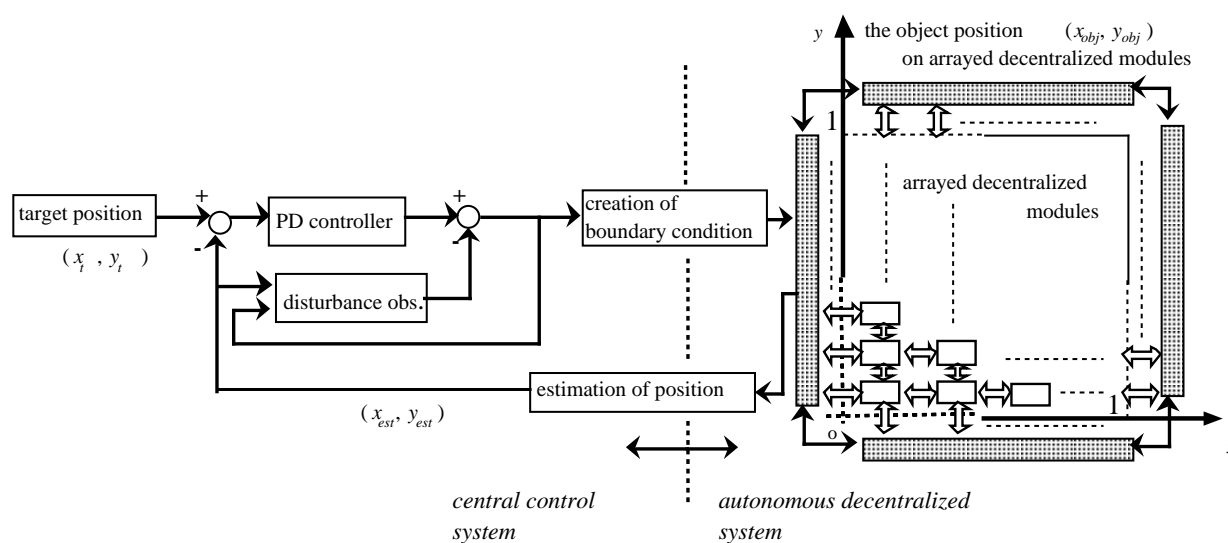


Fig.5.15: フィードバックによる方法のブロック線図

このプラントモデル (対象物の運動モデル) に対して、位置制御コントローラを設計を試みる。位置制御コントローラとして PD コントローラと外乱オブザーバを用いることとする。PD コントローラを

$$K(s) = k_0 + sk_1 \quad (5.12)$$

と置けば、閉ループ系伝達関数は、

$$\frac{k_0 + k_1 s}{m_x^{eqv} s^2 + k_1 s + k_0} \quad (5.13)$$

となる。2次遅れ系となる。2次系の固有応答時間 τ_r と減衰係数 ξ を用いれば、ゲイン k_0, k_1 は

$$\tau_r^2 = \frac{m_x^{eqv}}{k_0}, \quad 2\xi\tau_r = \frac{k_1}{k_2} \quad (5.14)$$

で表すことができる。この固有応答時間と減衰係数を適当な選ぶことにより、搬送物体の位置応答特性を自由に設計できる。

5.5.4 シミュレーション結果

ここでは以上で設計したフィードバック手法のシミュレーション結果を示す。シミュレーションパラメータはフィードフォワード手法のときに用いたものと同じで、Table 5.1である。Fig.5.16は搬送物体の位置応答特性である（ステップ応答）。また、Fig.5.17は搬送物体の軌跡である（はフィードフォワード手法のときと同じように500[ms]ごと）。Figs.5.18, 5.19, 5.20はそれぞれ0.59[s], 1.79[s], 4.14[s]のカーペットの挙動である。

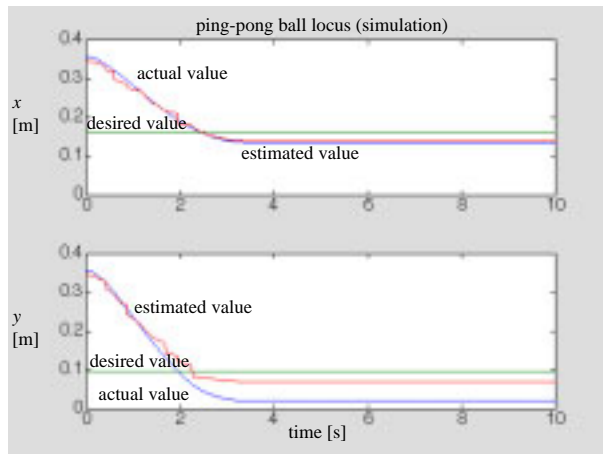


Fig.5.16: 対象物の位置に関する時間応答（シミュレーション）

5.6 分散マニピュレーション基本要件の無駄時間制限

5.6.1 無駄時間による不安定化

マクロ側からのアプローチの、フィードフォワードとフィードバックの二つの手法による設計を行った。

ところで、分散マニピュレーションには多数のアクチュエータによる実現から、第2章の分散マニピュレーション基本要件で示したとおり『無駄時間による制限』を受けた。本

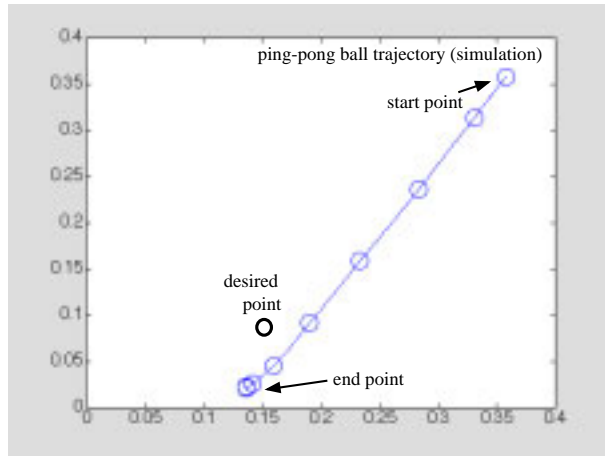


Fig.5.17: 対象物の軌跡

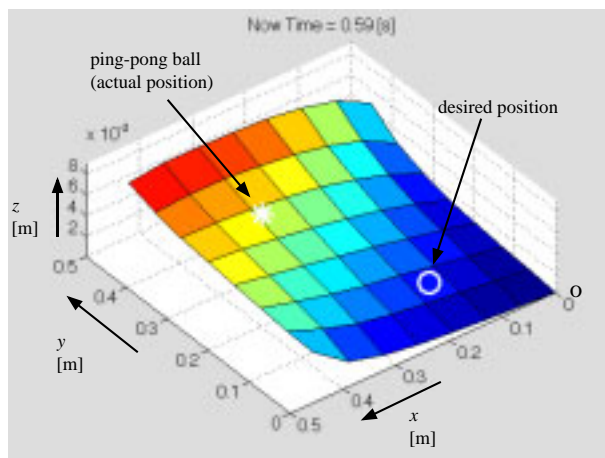


Fig.5.18: 0.59[s] での表面曲面

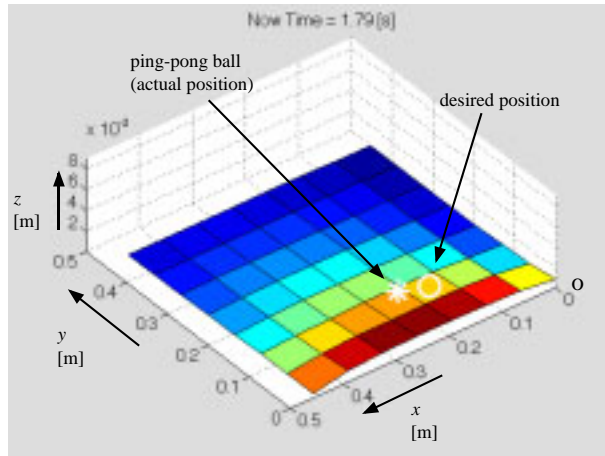


Fig.5.19: 1.79[s] での表面曲面

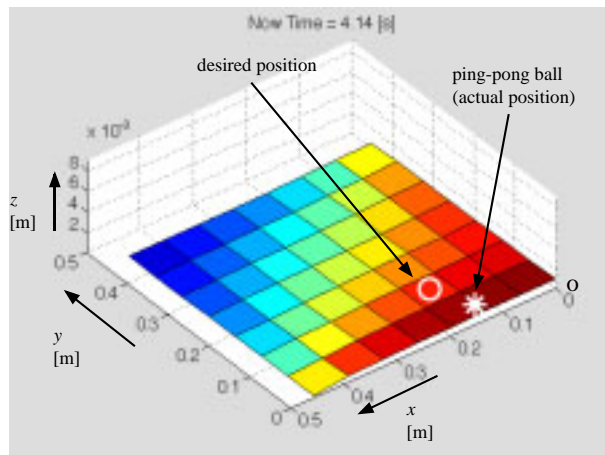


Fig.5.20: 4.14[s] での表面曲面

節ではこの無駄時間による制限が、物体搬送（『運ぶ』）にどれほど影響するか、提案したフィードフォワードとフィードバックのそれぞれについて考える。

無駄時間を含む物体搬送のブロック線図はフィードフォワードの場合、フィードバックの場合それぞれ、Figs.5.21, 5.22 のように書き直すことができる。

Fig.5.21 のフィードフォワードの場合は、無駄時間 $\tau_d = \tau_{d1}$ を含むものの、無駄時間を含んだフィードバック系を作っていないので、安定性には問題ない。応答が純粋に無駄時間分遅れるだけである。

Fig.5.22 のフィードバックの場合は、 $\tau_d = 2\tau_{d1} + \tau_{d2}$ の無駄時間を含むフィードバック系を構成している。これは対象物の挙動（安定性）に大きな影響を及ぼすと考えられる。

Fig.5.23 にフィードバックで無駄時間を $\tau_d = 0, 60, 80[\text{ms}]$ と変化させた場合の、対象物の x 方向の位置応答と対象物の位置推定値を示す。値がなめらかでなく、多少ギザギザしている応答が位置推定値である。

この結果 Fig.5.23 が示すように、無駄時間増加に伴い対象物の応答は不安定となる。Fig.5.23 の場合は、 $\tau_d = 80[\text{ms}]$ で対象物がはじき飛ばされて、制御不能となった。

これからわかるように、無駄時間があってもそれらを補償し、運ぶ動作は無駄時間があっても安定性を保証する形で実現しなくてはならない。

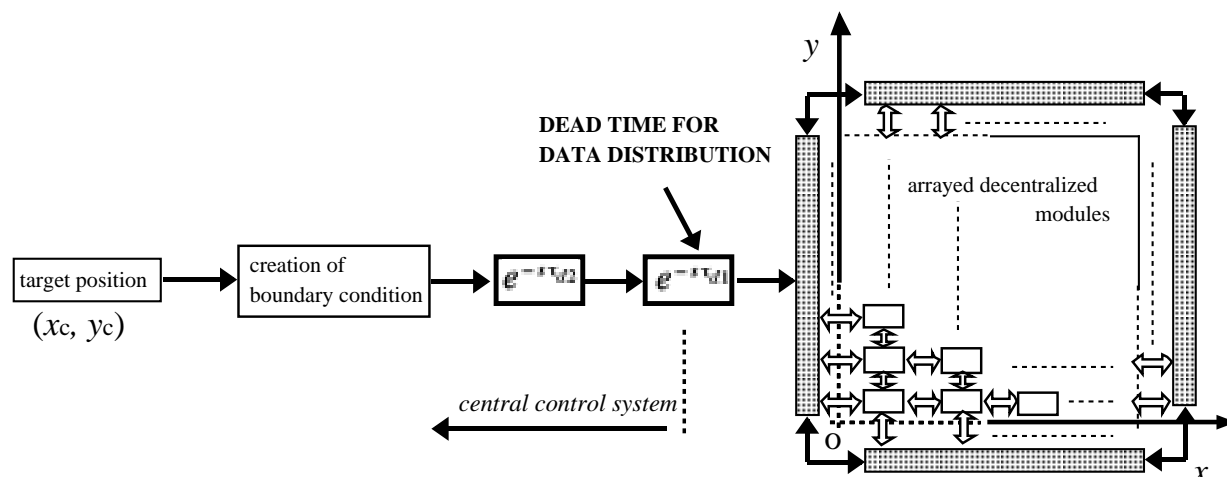


Fig.5.21: 無駄時間を考慮した場合のフィードフォワードのブロック線図

5.6.2 スミス法を代表とする無駄時間補償器

フィードバック制御系が無駄時間を内部に含む場合、そのフィードバック制御系の安定性が脅かされることは良く知られている。そのため、プラントが無駄時間を含んだ場合でも、安定性を補償するコントローラ的设计法が提案されている。

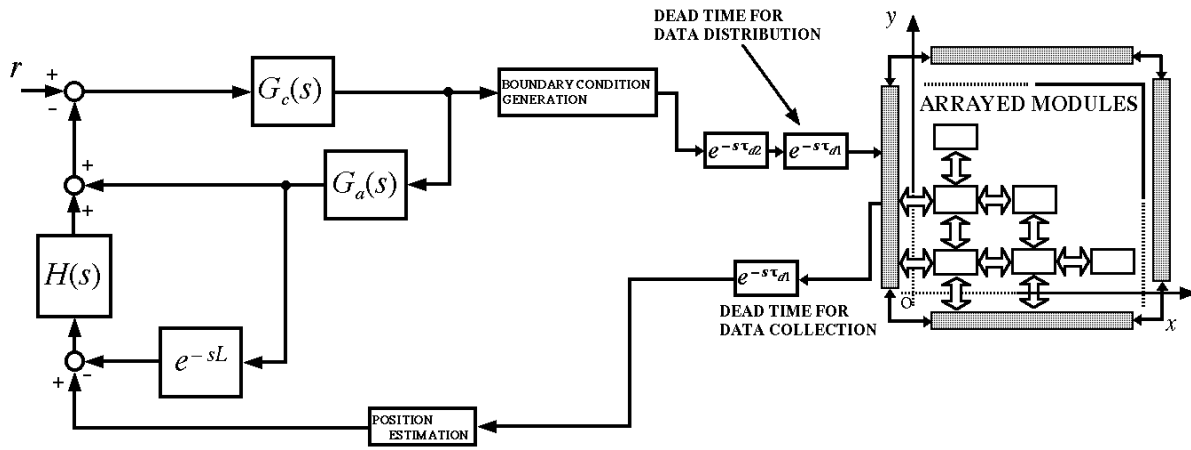


Fig.5.22: 無駄時間を考慮した場合のフィードバックのブロック線図

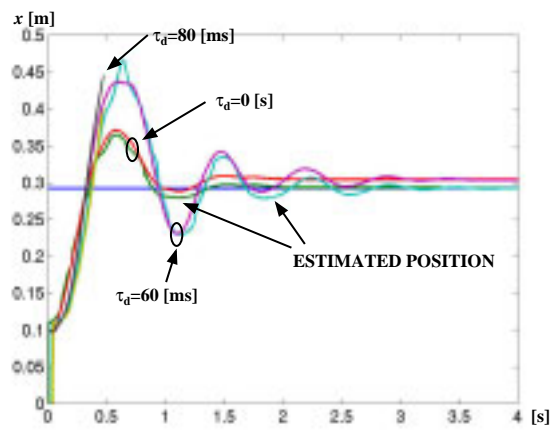


Fig.5.23: 無駄時間を考慮した設計を行わない場合

その中で、最も古くから良く知られているスミス法 [33], [34], [35] を用いて、分散マニピュレーションの基本三動作の一つ『運ぶ』が持つ無駄時間の補償を行う。

プラントが無駄時間を含む場合、Fig.5.24 のように単純にフィードバック制御を施すと、入力 r から出力 y までの伝達関数は、

$$\frac{y(s)}{r(s)} = \frac{G_c(s)G(s)e^{-sL}}{1 + G_c(s)G(s)e^{-sL}} \quad (5.15)$$

となり、分母に e^{-sL} を含み無限個の極を持つ。これに対して、 $G_c(s)$ のコントローラでそれら全ての極を指定することは難しい。それ故、無駄時間系の制御は難しい。

もっとも基本的なスミス法のブロック線図は Fig.5.25 で表すことができる。この入力から出力までの伝達関数は、

$$\frac{y(s)}{r(s)} = \frac{G_c(s)G(s)}{1 + G_c(s)G(s)}e^{-sL} \quad (5.16)$$

となる。(5.16) からわかるように、無駄時間要素 e^{-sL} をフィードバック系の中に含まない Fig.5.26 のブロック線図のように等価変換でき、 $G_c(s)$ でフィードバック系の極を指定でき安定化が可能である。

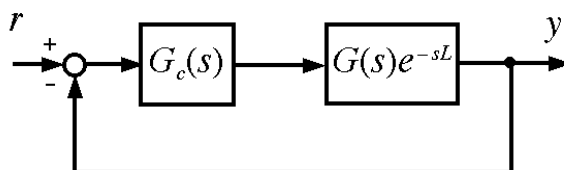


Fig.5.24: プラントが無駄時間を含む場合のフィードバック制御のブロック線図

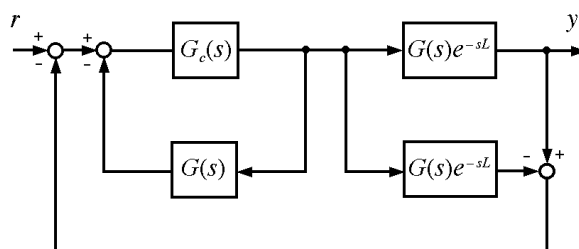


Fig.5.25: スミス法のブロック線図

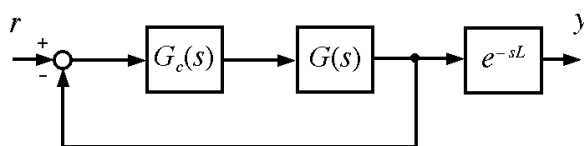


Fig.5.26: Fig.5.25 を等価変換したブロック線図

5.6.3 スミス法の外乱補償

スミス法は無駄時間を含む系を制御する手法として極めて有効である。しかし、外乱抑圧特性が悪かったり、定常誤差が残ったりすることが良く知られている [36], [37], [38]。スミス法のブロック線図は Fig.5.27 のように書き直すことができる。スミス法により付加された補償用のフィードバックが $G(s) - G(s)e^{-sL}$ である。

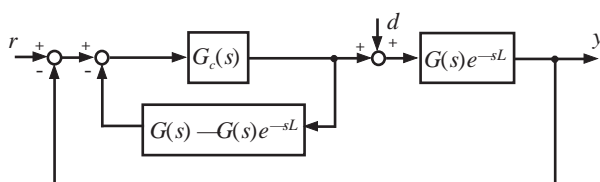


Fig.5.27: スミス法のブロック線図書きかえ版

サーボ系を構成する場合、ステップ状の目標値に対する定常誤差をなくすため、コントローラに積分器を入れる。内部モデル原理からもステップ状の目標値に定常誤差なく追従するためには、積分器が必要である。

しかしながら、スミス法では、 $G(s) - G(s)e^{-sL}$ の局所フィードバックがあるため、 $G_c(s)$ に積分器を含ませ、 $s = 0$ に極を持たせてもそのフィードバックで極が別のところに移され

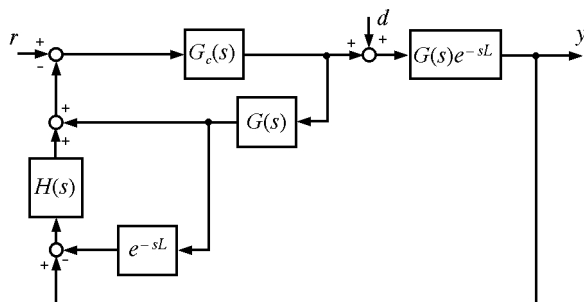


Fig.5.28: 外乱補償器を含んだスミス法のブロック線図

てしまう。そのため外乱抑圧特性が悪かったり、定常誤差が残ったりする。

この欠点を改善するために、Fig.5.28 のような外乱補償器 $H(s)$ を含んだブロック線図が考えられている。入力 $r(s)$ から出力 $y(s)$ までの伝達関数と外乱 $d(s)$ から出力 $y(s)$ までの伝達関数はそれぞれ (5.17), (5.18) で与えられる。

$$\frac{y(s)}{r(s)} = \frac{G_c(s)G(s)e^{-sL}}{1 + G_c(s)G(s)} \quad (5.17)$$

$$\frac{y(s)}{d(s)} = \frac{G(s)e^{-sL}}{1 + G_c(s)G(s)} + \frac{G_c(s)G(s)e^{-sL}}{1 + G_c(s)G(s)} \{G(s) - H(s)G(s)e^{-sL}\} \quad (5.18)$$

(5.17), (5.18) からわかるように、入力から出力までの伝達関数 $\frac{y(s)}{r(s)}$ は外乱補償器 $H(s)$ を含まない。これは、外乱補償器は目標値応答特性には影響を及ぼさないことを意味する。それに対して、外乱から出力までの伝達関数 $\frac{y(s)}{d(s)}$ は $H(s)$ を含む。これは、外乱抑圧特性は $H(s)$ で調整できることを意味している。実際、

1. $H(s)$ の極を任意に指定でき、
2. $1 - H(s)e^{-sL}$ の零点で $G(s)$ の極を相殺できる。
3. $H(0) = 1$

を満たすように、 $H(s)$ を設計すれば、外乱から出力までの過渡特性は $1 + G_c(s)G(s) = 0$ の根と $H(s)$ の極で調整が可能である。

上記の条件を満たす $H(s)$ 設計法として、伝達関数による方法を用いることにする。

$$G(s) = \sum_{i=1}^n \frac{b_i}{s + a_i}, \quad H(s) = \beta_0 + \sum_{i=1}^{n+1} \frac{\beta_i}{s + \alpha_i} \quad (5.19)$$

プラントの無駄時間を除く部分 $G(s)$ と外乱補償器 $H(s)$ を (5.19) のようにおく。ただし、 a_i は相異なる正の実数で、 $b_i \neq 0$ である。各条件から、下記の (5.20) を得る。これから、外乱補償器 $H(s)$ の各係数は (5.21) となる。

$$\begin{aligned} \beta_0 + \sum_{i=1}^n \frac{\beta_i}{\alpha_i - a_j} &= e^{-a_j L} \quad (j = 1, \dots, n) \\ \beta_0 + \sum_{i=1}^n \frac{\beta_i}{\alpha_i} &= 1 \end{aligned} \quad (5.20)$$

$$\begin{bmatrix} \beta_0 \\ \vdots \\ \beta_n \\ \beta_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{\alpha_1 - a_1} & \cdots & \frac{1}{\alpha_n - a_1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \frac{1}{\alpha_1 - a_n} & \cdots & \frac{1}{\alpha_n - a_n} \\ 1 & \frac{1}{\alpha_1} & \cdots & \frac{1}{\alpha_n} \end{bmatrix}^{-1} \begin{bmatrix} e^{-a_1 L} \\ \vdots \\ e^{-a_n L} \\ 1 \end{bmatrix} \quad (5.21)$$

5.6.4 無駄時間補償器による安定化

Fig.5.29のように、運ぶ動作に前節で説明した外乱補償器を含んだスミス法による無駄時間補償構造を含ませる。これにより無駄時間補償を行う。ちなみに外乱補償を行うスミス法の研究は他には [39], [40], [41] などがあり、より高性能のものもあるが、本節では、構造がシンプルでそこそこ性能もよい前節で説明した外乱補償器を含んだスミス法を適用する。

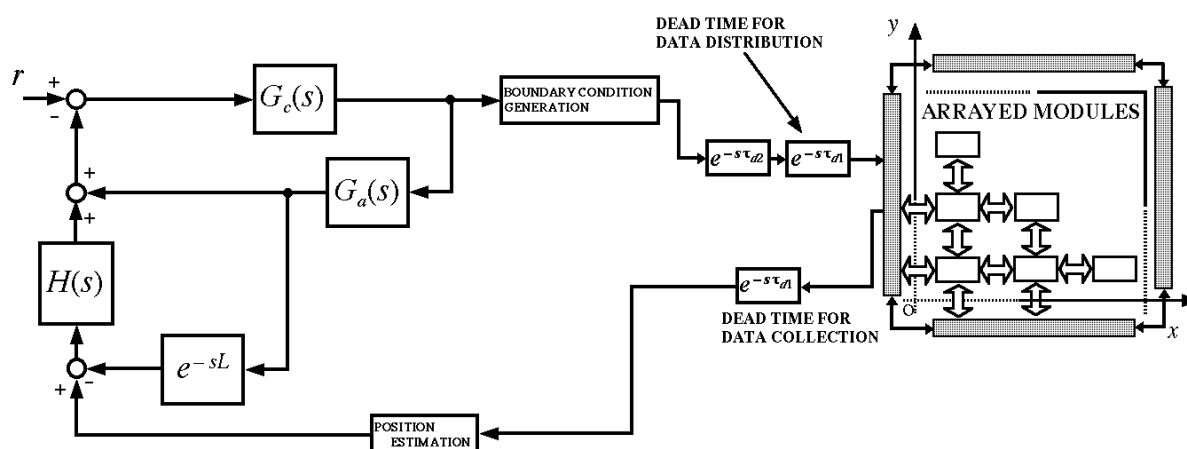


Fig.5.29: 外乱補償器を含むスミス法を運ぶ動作に適用したときのブロック線図

$$G(s) = \frac{1}{m_x^{\text{eqv}} s^2} \simeq G_a(s) = \frac{1}{m_x^{\text{eqv}} (s + a_1)(s + a_2)} \quad (5.22)$$

$G(s)$ に積分器が含まれているので、 $G_a(s)$ の $a_1, a_2 (a_1 \neq a_2)$ に極めて小さい値を設定することにより、運ぶ動作時のプラントモデル $G(s)$ の近似モデルとして、制御系設計に用いることにする。外乱補償器 $H(s)$ を (5.23) とすると、外乱補償器の各パラメータは、(5.24) で得る。

$$H(s) = \beta_0 + \frac{\beta_1}{s + \alpha_1} + \frac{\beta_2}{s + \alpha_2} \quad (5.23)$$

$$\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{\alpha_1 - a_1} & \frac{1}{\alpha_2 - a_1} \\ 1 & \frac{1}{\alpha_1 - a_2} & \frac{1}{\alpha_2 - a_2} \\ 1 & \frac{1}{\alpha_1} & \frac{1}{\alpha_2} \end{bmatrix}^{-1} \begin{bmatrix} e^{-a_1 L} \\ e^{-a_2 L} \\ 1 \end{bmatrix} \quad (5.24)$$

以上の方法で設計を行い、シミュレーションした結果を、Figs.5.30, 5.31 に示す。シミュレーションに用いたパラメータは、 $a_1 = 0.001, a_2 = 0.002$ で、外乱補償器の極は、 $\alpha_1 = 1.2, \alpha_2 = 0.8$ とした。

Fig.5.30 は外乱補償器を含まない、ただのスミス法で Fig.5.29 のブロック線図で $H(s) = 1$ とした場合である。無駄時間を 0, 40, 100[ms] と変化させ、位置応答出力をプロットした。Fig.5.30 の結果からわかる通り、無駄時間に関係なく安定化はされているものの、無駄時間が増加すればするほど、定常誤差が大きくなり、位置制御性能は落ちる。

Fig.5.31 には (5.24) で設計した外乱補償器 $H(s)$ を含むスミス法の位置応答シミュレーションを示す。無駄時間を 0, 100, 150[ms] と変化させ、そのときの位置応答出力を示した。無駄時間に関係なく、応答は安定で、定常誤差もなく所望の特性が得られていることがわかる。

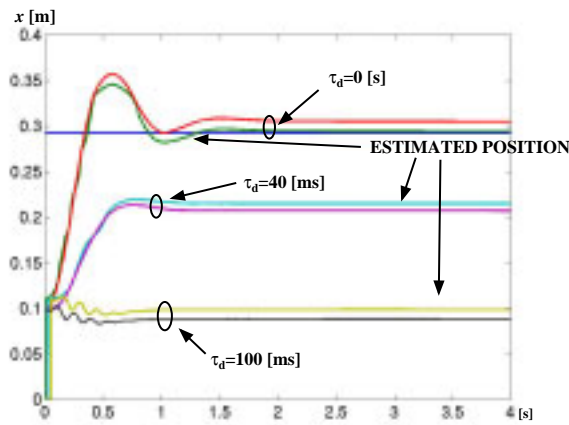


Fig.5.30: ただのスミス法でコントローラを設計した場合

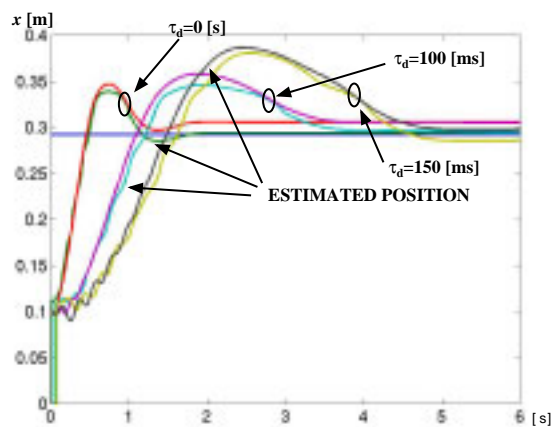


Fig.5.31: 外乱補償器を含んだスミス法で設計した場合

5.7 あとがき

本章では、基本三動作の最後の一つである『運ぶ』の実現を目指した。実現に当たり、多数のモジュールによる制御を考える場合、キーとなる場の概念を適用し、その適用に適した自律分散集中系というシステム形態での設計を試みた。

この自律分散集中系では、二つの制御存在した。

1. アクチュエータ全体のマクロな制御
2. アクチュエータ単体のミクロな制御

ミクロな制御を高速マイナーループとして、アクチュエータ全体のマクロな制御は外側の制御ループの多重ループの構造となっていた。ミクロな制御はアクチュエータ自身の制御で、具体的にはアクチュエータの変位の位置制御に当たる。こちらはアクチュエータ構造に依存するため、製作した実験装置に限り、付録 B で設計した。

マクロな制御はまさに運ぶの制御で、対象物の位置制御に当たる。こちらの設計は、場の概念により各制御レベルを分解した形で、ミクロな制御（マイナーループ）ができているという前提で、フィードフォワードによる方法とフィードバックによる方法を提案した。

フィードフォワードによる方法は、対象物の位置を推定する必要がなく容易な方法で目標位置まで搬送ができる。しかし、対象物が自然に持つダンピングを利用して、目標位置へ収束するため、収束特性が設計できない。また、目標位置に対して定常誤差を生じる場合がある。

この問題を解決するために、フィードバックによる方法では、対象物の位置制御のコントローラに PD コントローラと外乱オブザーバを導入し、閉ループ系の極を設計することにより、対象物の位置の収束特性を調整した。また、定常誤差も改善された。

分散マニピュレーション基本要件から無駄時間による制約を受けた。

フィードフォワードによる方法は、無駄時間により不安定化などの制約は受けなかった。単純に出力が無駄時間分だけ遅れるだけである。それに対して、フィードバックによる方法は、フィードバックにより構成される閉ループ系の中に無駄時間を含むため、その無駄時間により閉ループ系が不安定化してしまう。そこで、その補償を無駄時間の制御で古くから知られるスミス法により行うこととした。ただ、単純なスミス法は外乱抑圧特性が悪く、外乱がある場合、出力に定常誤差を残し目標値追従特性が悪いので、外乱補償器を含んだスミス法という改良されたスミス法を適用することとした。

これにより、無駄時間があったとしても安定なフィードバックによる方法を提供することができた。

第 6 章

各動作の有機的な結合

6.1 まえがき

本章では、第 3、4、5 章でそれぞれ単独で実現した、基本三動作『計る』、『つかむ』、『運ぶ』を有機的に結合させ、更なる高度な分散マニピュレーションの実現を目指す。

そもそも基本三動作は、その三動作があれば全てのマニピュレーションが実現できる分散マニピュレーション法の基本の動作として定義した。確かに、全てのマニピュレーションは実現できる。しかし、計った情報をもとにして、つかんだり、運んだりできたらよりよいマニピュレーションができると予想される。

例えば、分散マニピュレーション法ではなく、普通のマニピュレーション法であるが、自分の手で対象物がある点から別のある点へ、移動させることを考えてみる。まず手に取り、つかみながら、対象物の重さや堅さ形状などを手で無意識のうちに感じつつ、最適な持ち方をする。この動作は、はかりながらつかんでいるといえるだろう。そして、しっかりつかみつつ、目標位置まで目標軌道を描きつつ決して対象物をつかみ損ねることなく運ぶ。こちらの動作は、つかみながら運ぶといえるだろう。

計った情報をもとにして、つかんだり、そして、しっかりつかみながら運んだりといった複合動作により、マニピュレーションはより高度なものとなる。

本章では、はかりながらつかむと題し、対象物の運動量や角運動量といった力学的な量を保存し、対象物を安定な状態で荷重分散を実現しつつ、最適なつかむ方法をまず考える。対象物の力学的な量を保存し、つかむために対象物の重心位置や質量が必要となる。これらは計る動作で得られるので、それを利用する。これにより、対象物を安定につかむだけでなく、質量分布をしている対象物も荷重分散しつつ、つかむことができるようになる。

次に、つかみながら運ぶと題し、搬送に対して有効な荷重分散モードのみ有効にする方法を考える。バランス良くつかみながら運ぶ方法である。対象物に力学的な力を加え運ぶ場合、対象物が有限な大きさのため、運ぶ動作による力で不必要なモーメントが発生し、静止時には有効に作用していた荷重分散（つかむ）が実現できなくなってしまう。そこで、

運ぶ動作でどの程度の不必要なモーメントが発生するか、そしてその方向はどの向きか、予測しそれらを考慮した荷重分散（つかむ）を実現する。これがつかみながら運ぶである。

本章最後に、全ての動作の結合として、計りながらつかみ、そして運ぶというマニピュレーションの完成系を示す。

6.2 計りながらつかむ - 局所的な荷重分散から大域的な荷重分散 -

6.2.1 つかむ動作単独の場合の問題点

質量が分布している対象物の把持を考えてみる。Fig.6.1 は質量分布負荷のモデルである。以下では、この Fig.6.1 のモデルの把持を例とし、つかむ単独動作時の問題点について考察する。

Fig.6.1 は、質量の面密度が左半分を 1 とすると、右半分が 2 の質量分布をしている。そのため重心位置 (x_g, y_g) は、 x 軸方向の対象物の長さを a 、 y 軸方向の対象物の長さを b とすると、 $(x_g, y_g) = (x_c + \frac{a}{3}, y_c)$ となり、対象物の中心 (x_c, y_c) と異なる。

Fig.6.2 に、Fig.6.1 の質量分布モデルで、第 4 章のつかむで提案した荷重分散を行った結果を示す。Fig.6.2 は対象物上部から見た図で、各アクチュエータの負荷を時間平均した負荷分布の等高線である。

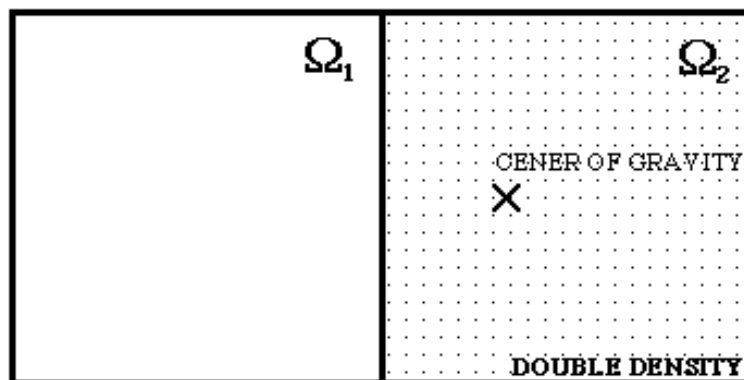


Fig.6.1: 質量が分布している負荷のモデル

この結果は、一見上手くいっているように見えるが二つの点で問題がある。

一つはアクチュエータの負荷の時間平均値は荷重分散ができていているということである。しかし、ある瞬間瞬間で見れば、アクチュエータは対象物に触れたり、離れたりを繰り返し、非常に高周波で振動していて安定な応答をしていない(Figs.6.3, 6.4)。負荷の平均値は

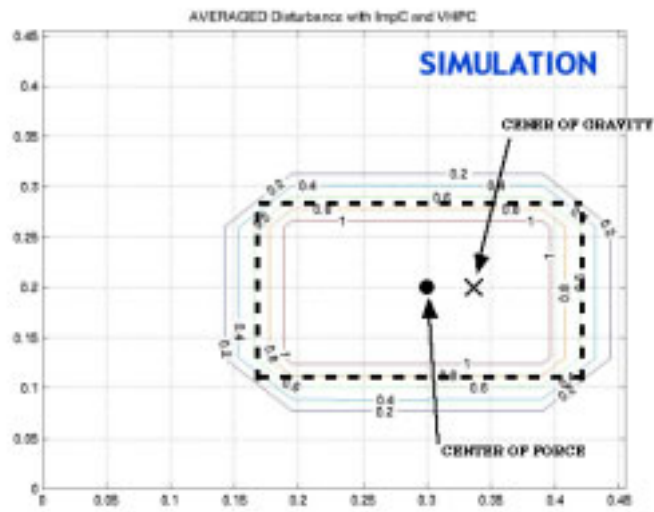


Fig.6.2: 分布負荷を荷重分散で支えた場合の等高線



Fig.6.3: Fig.6.1 の質量分布負荷モデルでの Ω_1 (軽い領域) のアクチュエータの時間応答



Fig.6.4: Fig.6.1 の質量分布負荷モデルでの Ω_2 (重い領域) のアクチュエータの時間応答

荷重分散できているがそれは平均値が等しくなっているだけであって、振動し不安定になってしまっは意味がない。

二つ目は、そもそも質量が分布しているような対象物の場合、その質量分布に合わせた形でアクチュエータは力を出さないと安定に把持ができない。Fig.6.1 の場合は、左半分比べて、右半分を支えるアクチュエータは大きな力を発生しないとイケないはずである。Fig.6.5 は、対象物に働いている力を示した図である。対象物の重さに起因する力は重心を中心に鉛直方向に作用している。それに対して、支える力はトータルでは対象物の中心に対象物の重さに等しい量が発生している。上方向と下方向に作用する力の位置が異なるためモーメントが発生しており、対象物は回転方向に力を受けてしまう。

本来、こういった場合は、全てのアクチュエータで同じ力で支える荷重分散はできず、左半分の領域で荷重分散ができ、左半分の領域でまた荷重分散ができる。二つの領域で異なった力で荷重分散をし、モーメントが発生しないようする。それが最適な荷重分散である。

6.2.2 対象物の力学的な安定と荷重分散

前節で指摘したように、対象物の質量が分布している場合は、完全な荷重分散はできない。Fig.6.1 のような質量分布モデルの場合は、左半分比べ、右半分は大きな力で支持するような二つの荷重分散領域を作るのが最適な荷重分散といえる。

本節では、このように質量が分布している対象物に対して、対象物の力学的な状態を保持し安定に把持しつつ、複数の荷重分散領域を組み合わせ最適な荷重分散を行うことを考える。

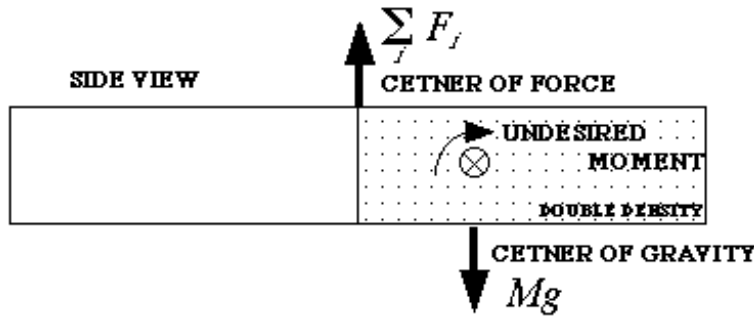


Fig.6.5: 重量が分布している負荷を横から見た場合

まずはじめに二つの仮定を行う。

1. 対象物は一つである。
2. 質量分布の変化は連続で急激には変化しない。

次に、対象物の力学的な安定状態を定義する。

1. 対象物に働いている力は釣り合っている。すなわち、各アクチュエータで発生している力の総和は、対象物の質量に等しい。

$$M\dot{v} = F_{ext} = 0 \Leftrightarrow Mg = \sum_i F_i \quad (6.1)$$

2. 重心を中心とする回転運動はなく、モーメントは釣り合っている。

$$\dot{L} = N_{ext} = 0 \Leftrightarrow (r_i - R) \times F_i = 0 \quad (6.2)$$

ただし、ここで、 $v, F_{ext}, L, N_{ext}, F_i, r_i, R$ はそれぞれ対象物の速度、対象物に作用している力、対象物の角運動量、外力のモーメント、外力 (i 番目のアクチュエータの力)、 i 番目のアクチュエータの位置、重心である。二つの仮定を用いて、(6.1), (6.2) を満たすように荷重分散力を決めるのが目標である。

二つ目の仮定から、小さい領域 Ω_k 内 (局所的荷重分散領域と呼ぶことにする) では、対象物の質量分布のばらつきは小さく、荷重分散はできるとする (局所的荷重分散という)。すなわち、

$$\begin{aligned} G_d z &= F + G_n z^d \quad \text{in } \Omega_k \\ z^d &= A_h^\dagger A_0 G_d G_n^{-1} Q z + A_h^\dagger S \quad \text{in } \Omega_k \end{aligned}$$

を満たすようにホームポジション z^d を決定できる。複数ある局所的荷重分散領域で発生する力が、(6.1), (6.2) を満たすようにバランスを取ることができればよい。

(6.1), (6.2) を局所的荷重分散領域に書き直すと、

$$\begin{cases} Mg = \sum_i F_i \\ \sum_i (x_i - x_g) F_i = 0 \\ \sum_j (y_j - y_g) F_j = 0 \end{cases} \quad (6.3)$$

を得る。 $\sigma_k(i)$ は局所的荷重分散領域 Ω_k 内の i 番目のアクチュエータが全体のアクチュエータ番号の何番に対応するかを決める関数で、これを用いて書き直すと、

$$\begin{cases} Mg = \sum_k \sum_i F_{\sigma_k(i)} \\ \sum_k \sum_i (x_{\sigma_k(i)} - x_g) F_{\sigma_k(i)} = 0 \\ \sum_k \sum_i (y_{\sigma_k(i)} - y_g) F_{\sigma_k(i)} = 0 \end{cases} \quad (6.4)$$

領域 Ω_k 内では荷重分散が出来ていて、

$$F_{\sigma_k(1)} = F_{\sigma_k(2)} = \cdots = F_{\sigma_k(i_{\max}^k)} \equiv F_{\sigma_k} \quad (6.5)$$

とかけるはずだから、

$$\begin{cases} \sum_k \sum_i F_{\sigma_k(i)} = \sum_k i_{\max}^k F_{\sigma_k} = Mg \\ \sum_k \sum_i (x_{\sigma_k(i)} - x_g) F_{\sigma_k(i)} = \sum_k \left\{ \left(\sum_i x_{\sigma_k(i)} \right) - i_{\max}^k x_g \right\} F_{\sigma_k} \\ \sum_k \sum_i (y_{\sigma_k(i)} - y_g) F_{\sigma_k(i)} = \sum_k \left\{ \left(\sum_i y_{\sigma_k(i)} \right) - i_{\max}^k y_g \right\} F_{\sigma_k} \end{cases} \quad (6.6)$$

と書き直すことができる。結局、(6.6) は、

$$DF_{\sigma} = M_r \quad (6.7)$$

とかける。ただし、

$$D = \begin{bmatrix} i_{\max}^1 & \cdots & i_{\max}^k \\ \sum_i x_{\sigma_1(i)} & \cdots & \sum_i x_{\sigma_k(i)} \\ \sum_i y_{\sigma_1(i)} & \cdots & \sum_i y_{\sigma_k(i)} \end{bmatrix}, \quad \mathbf{F}_{\sigma} = \begin{bmatrix} F_{\sigma_1} \\ \vdots \\ F_{\sigma_k} \end{bmatrix}, \quad \mathbf{M}_r = \begin{bmatrix} Mg \\ x_g Mg \\ y_g Mg \end{bmatrix} \quad (6.8)$$

\mathbf{F}_{σ_k} は領域の数だけあるのに対して、方程式は3本しかない。この方程式を解くために、疑似逆行列 (MP 逆行列) を用いて、

$$\mathbf{F}_{\sigma_k} = D^{\dagger} \mathbf{M}_r \quad (6.9)$$

で解くことができる。次節で説明するが、領域を合成して徐々に領域の数は少なくなり、行列 D は正方行列に近づいていき、正確に冗長性なく解けるようになる。ここで得られた局所的な荷重分散力の実現を次で目指す。ちなみに、複数の局所的荷重分散を用いて、バランスを取りながら荷重分散を行うこの方法を、局所的と対比させ、大域的荷重分散と呼ぶことにする。

物体の状態を保存しながら力の分配を決定する式（大域的荷重分散決定式）

$$\mathbf{F}_{\sigma_k} = D^\dagger \mathbf{M}_r \quad (6.10)$$

$$\begin{cases} G_d \mathbf{z} = \mathbf{F} + G_n \mathbf{z}^d \\ \mathbf{z}^d = A_h^\dagger A_0 G_d G_n^{-1} Q \mathbf{z} + A_h^\dagger \mathbf{S} \end{cases} \quad (6.11)$$

もともと荷重分散を行うための、インピーダンス制御による可変ホームポジション構造は上式の (6.11) であった。この可変ホームポジション決定の方程式は冗長性を持っていた。その冗長性を殺すために、

$$\mathbf{H}^T \mathbf{z}^d = \mathbf{S}$$

という方程式を追加していた。この冗長性を Ω_k 内の目標荷重分散力 F_{σ_k} の実現に利用する。 Ω_k 内の局所的なアクチュエータ番号が $\xi(k)$ のアクチュエータを目標荷重分散力 F_{σ_k} とする。すなわち、

$$z_{\sigma_k(\xi(k))}^d = G_n^{-1} G_d z_{\sigma_k(\xi(k))} - G_n^{-1} F_{\sigma_k} \quad (6.12)$$

である。各ベクトルを、 Ω_k 内用書き直すと、

$$\mathbf{z}_k = \begin{bmatrix} z_{\sigma_k(1)} \\ \vdots \\ z_{\sigma_k(i_{\max}^k)} \end{bmatrix}, \quad \mathbf{z}_k^d = \begin{bmatrix} z_{\sigma_k(1)}^d \\ \vdots \\ z_{\sigma_k(i_{\max}^k)}^d \end{bmatrix}, \quad \mathbf{H}_k = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (6.13)$$

となる。ただしここで、 \mathbf{H}_k は $\xi(k)$ 行のみ 1 であとはゼロである。

$$\begin{bmatrix} A^k \\ \mathbf{H}_k^T \end{bmatrix} \mathbf{z}_k^d = G_n^{-1} G_d \begin{bmatrix} A^k \\ \mathbf{H}_k^T \end{bmatrix} \mathbf{z}_k - G_n \begin{bmatrix} \mathbf{0} \\ F_{\sigma_k} \end{bmatrix} \quad (6.14)$$

$$\mathbf{z}_k^d = G_n^{-1} G_d A_h^{k\dagger} A_h^k \mathbf{z}_k - G_n A_h^{k\dagger} \mathbf{F}_{\sigma_k}^0 \quad (6.15)$$

$$\simeq G_n^{-1} G_d \mathbf{z}_k - G_n A_h^{k\dagger} \mathbf{F}_{\sigma_k}^0 \quad (6.16)$$

$$A_h^k = [A^k | \mathbf{H}_k^T]^T, \quad \mathbf{F}_{\sigma_k}^0 = [0 | F_{\sigma_k}]^T \quad (6.17)$$

力の分配を考慮した局所的荷重分散決定式

$$\mathbf{z}_k^d = G_n^{-1} G_d \mathbf{z}_k - G_n A_h^{k\dagger} \mathbf{F}_{\sigma_k}^0 \quad (6.18)$$

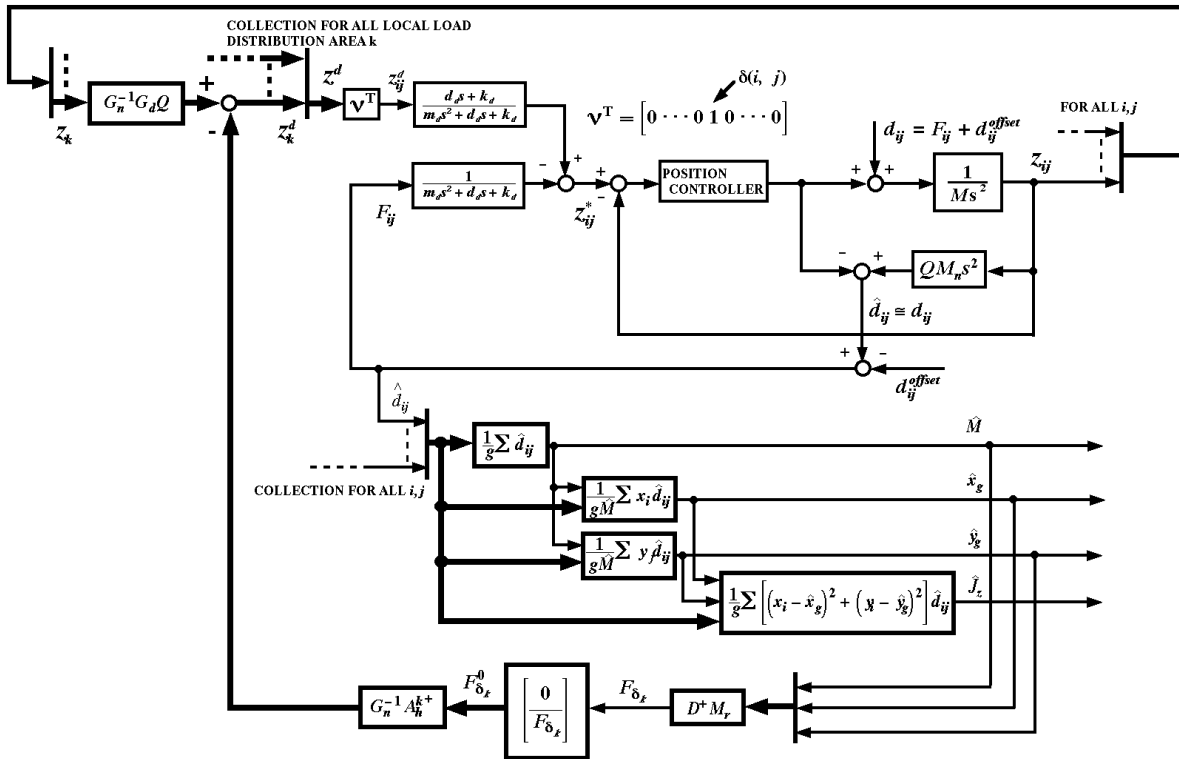


Fig.6.6: 大域的な荷重分散構造を含む場合のブロック線図

局所的な荷重分散 Ω_k に対して、(6.10) から荷重分散力を決定し、(6.18) の方法でその荷重分散力を実現すればいい。ブロック線図で示すと、Fig.6.6 となる。

6.2.3 局所的荷重分散領域作成法

大域的な荷重分散は、(6.10) と (6.18) で実現できる。しかし、これは局所的荷重分散領域が与えられたときである。

局所的荷重分散領域を作り方を考えるに当たり、注意すべき点は下記の通りである。

1. できるだけ大きな領域で荷重分散するのが望ましい。

2. 対象物の密度が分布している場合は、一つの領域だけで荷重分散することは不可能。
3. 最終的な目的は、対象物を出来るだけ広く少ない領域で荷重分散し支えること。
4. あるタイミングでは、重心周りにロール、ピッチといったモーメントが最もゼロに近くなるように、各々の局所的領域の荷重分散力を決定する。
5. 各々の局所的領域での荷重分散力がモーメントを発生しない状態が出来た後、各々の局所的領域を合成し、新しい領域を作る。
6. 新しい領域でまた同じように各々の局所的領域で荷重分散力を計算し、領域更新に伴い広がった領域で、荷重分散を実現。

以上を踏まえて、最適な方法ではないと思われるが、少なくとも局所的な荷重分散が作れる方法は、

1. 初期ステップ (Fig.6.7) $n = 1 \quad \Omega_k^{n=1}$
 - (a) 負荷外乱量 $\neq 0$ のアクチュエータの検出。
 - (b) 負荷外乱量 $\neq 0$ のアクチュエータを一つ選び、 Ω_k の代表とする。
 - (c) 負荷外乱量 $\neq 0$ で、 Ω_k の代表アクチュエータに隣接するアクチュエータを Ω_k に属させる。
 - (d) 負荷外乱量 $\neq 0$ で、 Ω_k に属するアクチュエータに隣接するアクチュエータはまた Ω_k のアクチュエータとする。
2. 次のステップ (Fig.6.8) $n \rightarrow n + 1 \quad \Omega_k^n \rightarrow \Omega_k^{n+1}$
 - (a) 近い領域同士を合成し、一つの領域とする。
 - (b) Fig.6.8 のように近い領域同士を合成し、同じ局所的な荷重分散領域として、荷重分散を行う。
 - (c) Ω_1 と Ω_2 との境界領域 Ω_{12} の作成方法としては、小さい領域 Ω_2 を大きな領域 Ω_1 へ延長し、延長された領域を Ω_{12} として、 $\Omega_1 + \Omega_{12} + \Omega_2 \rightarrow \Omega_1^{new}$ とする。
3. 閉領域の合成 (Fig.6.9) $\Omega_1 + \bar{\Omega}_1 \rightarrow \Omega_1^{new}$
 - (a) ある領域 Ω_1 に囲まれた閉領域 $\bar{\Omega}_1$ がある場合は、 $\Omega_1 + \bar{\Omega}_1 \rightarrow \Omega_1^{new}$ とする。
4. 領域合成終了
 - (a) ある荷重分散の領域に加わっていて、荷重分散を行っているアクチュエータでも負荷外乱がゼロならば、そのアクチュエータは荷重分散領域から外す。

(b) 荷重分散領域の分布に変化がなくなり、 $\Omega_k^n = \Omega_k^{n+1}$ (for all k) ならば、領域合成のプロセスを終了する。

となる。

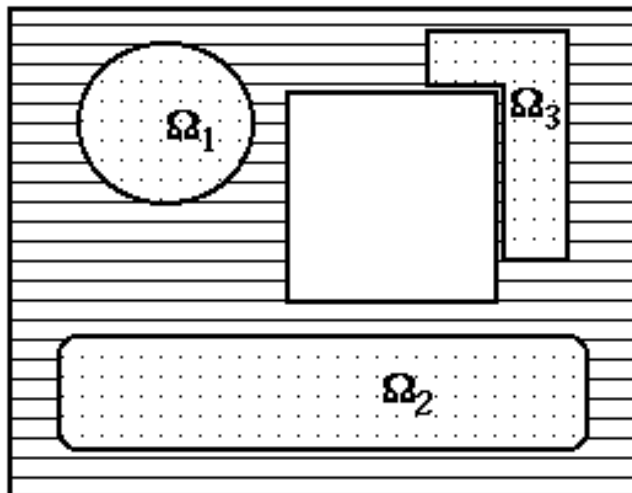


Fig.6.7: 初期ステップ

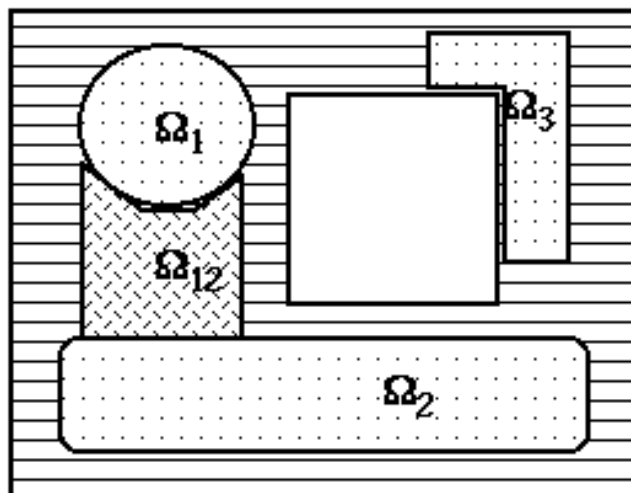


Fig.6.8: 次のステップへの移行

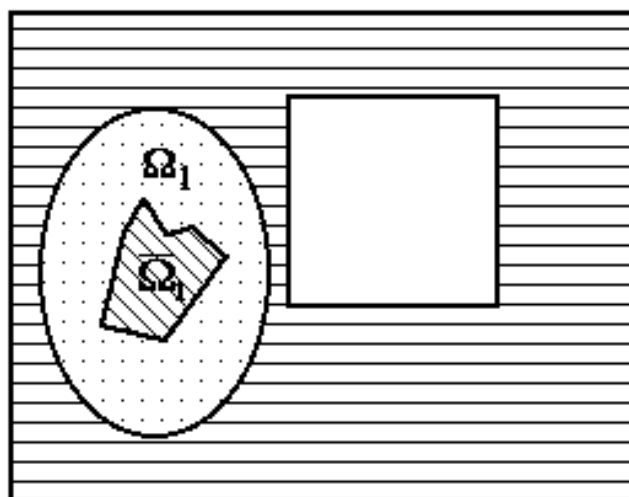


Fig.6.9: 閉領域の合成

6.2.4 シミュレーション結果

以上で示した (6.10), (6.18) と局所的な重分散領域の作成法を用いてシミュレーションした結果を以下に示す。Fig.6.1 で示した質量分布負荷モデルを用いてシミュレーションを行った。Figs.6.10, 6.11, 6.12 はそれぞれ、アクチュエータが発生している力の等高線、軽い領

域 Ω_1 にあるアクチュエータの時間応答、重い領域 Ω_2 にあるアクチュエータの時間応答である。

Fig.6.10 から、右半分の重い領域 Ω_2 には、等高線が過密に分布しているのがわかる。これは、右半分の Ω_2 では大きな力で支えられていることを示している。また、逆に左半分の軽い領域 Ω_1 の領域では、小さい力で支えられていることがわかる。Fig.6.10 には、この等高線から計算した力の中心位置を示した。力の中心位置は、質量分布負荷モデルの重心位置と同じところにあり、不用意なモーメントが発生することなく、対象物は力学的に安定な状態にあるといえる。

Figs.6.11,6.12 はアクチュエータの時間応答で、局所的な荷重分散だけで場合の Figs.6.3, 6.4 に対応する。Figs.6.3, 6.4 では、アクチュエータが支える力が非常に高周波でばたつき、不安定となる問題があった。しかし、大域的な荷重分散手法を用いた Figs.6.11,6.12 では、アクチュエータが支える力が不安定になることなく、安定な応答を示していることがわかる。

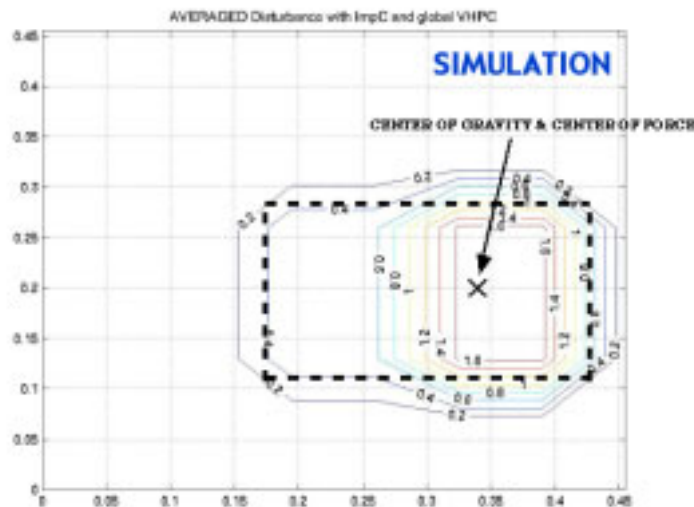


Fig.6.10: 大域的荷重分散による質量分布モデルの支持を行ったときのアクチュエータの力の等高線

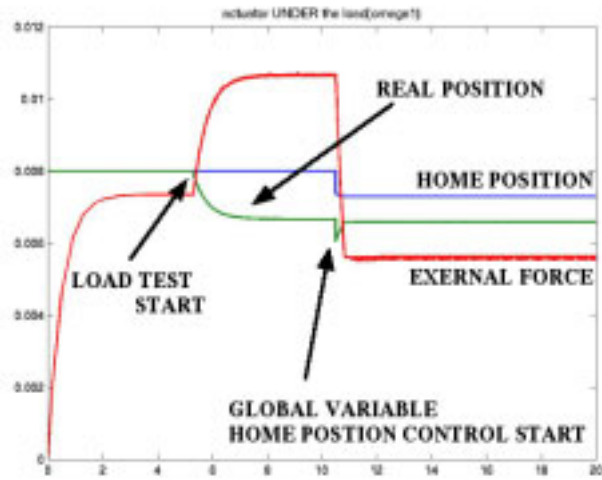


Fig.6.11: 大域的荷重分散で Fig.6.1 の質量分布モデルの軽い領域 Ω_1 のアクチュエータの時間応答

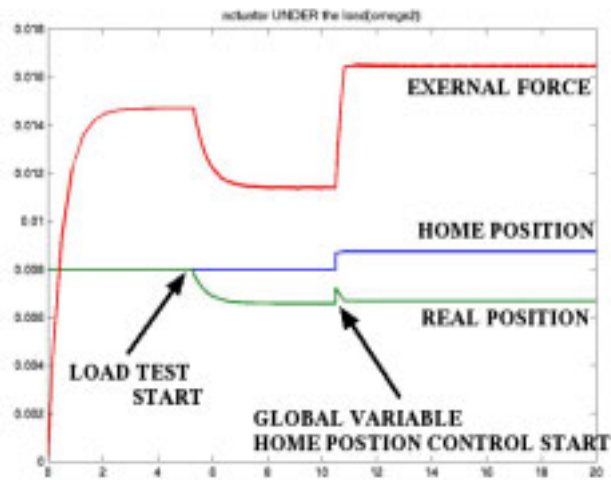


Fig.6.12: 大域的荷重分散で Fig.6.1 の質量分布モデルの軽い領域 Ω_2 のアクチュエータの時間応答

6.3 つかみながら運ぶ - 搬送方向だけ大域的荷重分散モードの解放 -

6.3.1 大域的荷重分散の制御目標の拡張

前節で提案した大域的荷重分散の制御目標は、対象物の平衡状態のために、

$$\dot{\mathbf{P}} = \mathbf{F}_{ext} = \begin{bmatrix} 0 \\ 0 \\ \sum_i F_i - Mg \end{bmatrix} = \mathbf{0} \Leftrightarrow \sum_i F_i = Mg \quad (6.19)$$

$$\dot{\mathbf{L}} = \mathbf{N}_{ext} = (\mathbf{r}_i - \mathbf{x}_g) \times F \mathbf{e}_z = \begin{bmatrix} \sum_i (y_i - y_g) F_i \\ - \sum_i (x_i - x_g) F_i \\ 0 \end{bmatrix} = \mathbf{0} \Leftrightarrow \begin{cases} \sum_i (x_i - x_g) F_i = 0 \\ \sum_i (y_i - y_g) F_i = 0 \end{cases} \quad (6.20)$$

であった。(6.19), (6.20) が満たされるように F_i (各アクチュエータ出力) は調整されていた。

実際は物体を搬送するために対象物に力が加えられる。この搬送のための力が加わったとしても、対象物に不用意なモーメントを発生させることなく、大域的な荷重分散が行われることが望ましい。

搬送物体に搬送力が加わっているとき、前節までの大域的な荷重分散がどのような効果を持つか検証する。搬送物体に加えられている力の x 方向成分、 y 方向成分をそれぞれ F_x, F_y とし、その力の作用点を (x_g, y_g, z_p) とする (6.13 参照)。以下の計算を簡単にするために、作用点を重心直下とする。これより、(6.19) は、

$$\dot{\mathbf{P}} = \mathbf{F}_{ext} = \begin{bmatrix} F_x \\ F_y \\ \sum_i F_i - Mg \end{bmatrix} \quad (6.21)$$

となる。(6.20) は、

$$\dot{\mathbf{L}} = \mathbf{N}_{ext} = (\mathbf{r}_i - \mathbf{x}_g) \times \mathbf{F}_i = (\mathbf{x}_p - \mathbf{x}_g) \times \begin{bmatrix} F_x \\ F_y \\ 0 \end{bmatrix} + (\mathbf{r}_i - \mathbf{x}_g) \times \begin{bmatrix} 0 \\ 0 \\ F_i \end{bmatrix} \quad (6.22)$$

$$= \begin{bmatrix} -(z_p - z_g) F_y + \sum_i (y_i - y_g) F_i \\ (z_p - z_g) F_x - \sum_i (x_i - x_g) F_i \\ 0 \end{bmatrix} \quad (6.23)$$

となる。前節で提案した大域的な荷重分散を行っている場合は、

$$\begin{cases} \sum_i F_i = Mg \\ \sum_i (x_i - x_g) F_i = 0 \\ \sum_i (y_i - y_g) F_i = 0 \end{cases} \quad (6.24)$$

を満たすように F_i が決定される。したがって、

$$\dot{\mathbf{L}} = \begin{bmatrix} -(z_p - z_g) F_y \\ (z_p - z_g) F_x \\ 0 \end{bmatrix} \neq \mathbf{0} \quad (6.25)$$

となる。(6.25) からわかるように、対象物に作用しているモーメントはゼロではなく有限の値となり、不必要なモーメントが発生している。この不必要なモーメントは大域的な荷重分散手法で除去したい。

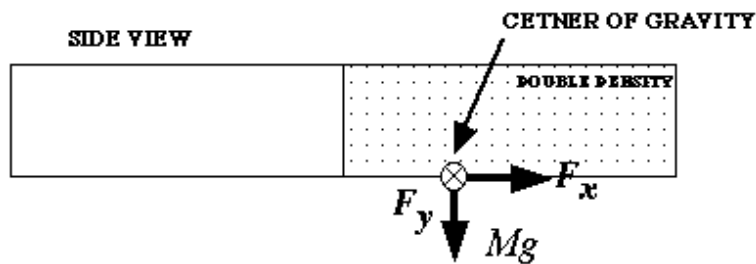


Fig.6.13: 搬送力をはじめ対象物に作用している力

ここで、Fig. 6.14 にあるように、対象物に作用している力に関する定義をする。 $F_x + F_y$ 方向の単位ベクトルを e_{f1} とする。その e_{f1} を反時計方向に $\pi/2$ だけ回転させ、 e_{f1} に直交するベクトルを e_{f2} と定義する。すなわち、

$$e_{f1} = \begin{bmatrix} \frac{F_x}{\sqrt{F_x^2 + F_y^2}} \\ \frac{F_y}{\sqrt{F_x^2 + F_y^2}} \end{bmatrix} \quad (6.26)$$

$$e_{f2} = \begin{bmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{bmatrix} e_{f1} = \begin{bmatrix} -\frac{F_y}{\sqrt{F_x^2 + F_y^2}} \\ \frac{F_x}{\sqrt{F_x^2 + F_y^2}} \end{bmatrix} \quad (6.27)$$

である。

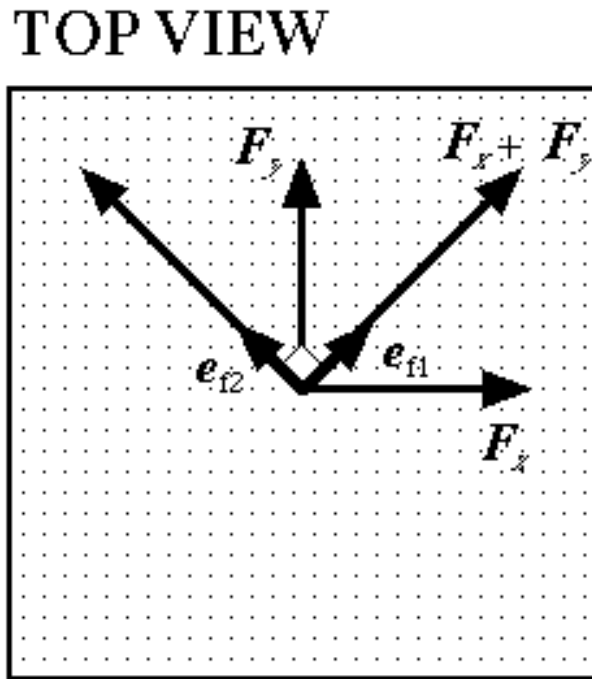


Fig.6.14: 対象物に作用している力ベクトルの定義

e_{f1} 方向に力を受け進んでいるとき、対象物は前後に揺れ e_{f2} 方向のモーメントを持つ。これは車に乗っているときを例にとると分かり易い。アクセルを踏んで、進行方向に力を受けているとき車体は後ろのめりになる。逆にブレーキを踏んでいるときは車体は前のめりになる。進行方向に直交する方向のモーメントを持つことになる。

例からわかるように、対象物に力を加え搬送するとき、対象物に作用している力に直交する方向 e_{f2} のモーメントの発生は仕方がない（これは分散マニピュレータの自由度の問題で詳しくはあとがきで考察する）。

$F_x + F_y$ によって生じる $F_x + F_y$ 方向の前後の振れは荷重分散で補償しない。すなわち、 e_{f2} まわりのモーメントはそのままにする。対して、 $F_x + F_y$ 方向に対して、左右の振れは大域的な荷重分散で補償する。 e_{f1} まわりのモーメントを補償するように、前節で提案した大域的荷重分散の目標を拡張する。

搬送方向の大域的な荷重分散モードを解放した方程式は (6.31), (6.32) となる。

$$\dot{\mathbf{L}} \cdot \mathbf{e}_{f2} = \mathbf{N}_{ext} \cdot \mathbf{e}_{f2} \tag{6.28}$$

$$= \frac{1}{\sqrt{F_x^2 + F_y^2}} \left\{ F_x \sum_i (y_i - y_g) F_i - F_y \sum_i (x_i - x_g) F_i \right\} = 0 \tag{6.29}$$

$$\Leftrightarrow F_x \sum_i (y_i - y_g) F_i - F_y \sum_i (x_i - x_g) F_i = 0 \tag{6.30}$$

搬送方向の荷重分散モードを解放した大域的荷重分散方程式

$$\begin{aligned} \dot{\mathbf{P}} \cdot \mathbf{e}_z &= \mathbf{F}_{ext} \cdot \mathbf{e}_z = 0 \\ \Leftrightarrow \sum_i F_i &= Mg \end{aligned} \quad (6.31)$$

$$\begin{aligned} \dot{\mathbf{L}} \cdot \mathbf{e}_{f1} &= \mathbf{N}_{ext} \cdot \mathbf{e}_{f1} = 0 \\ \Leftrightarrow F_x \sum_i y_i F_i - F_y \sum_i x_i F_i &= Mg(F_x y_g - F_y x_g) \end{aligned} \quad (6.32)$$

次に、(6.31), (6.32) が局所的な荷重分散ができていると仮定すると、

$$\sum_i F_i = \sum_{k=1}^{k_{\max}} i_{\max}^k F_{\sigma_k} = Mg \quad (6.33)$$

$$\begin{aligned} F_x \sum_i y_i F_i - F_y \sum_i x_i F_i &= \sum_{k=1}^{k_{\max}} F_{\sigma_k} \left\{ F_x \sum_{i=1}^{i_{\max}^k} y_{\sigma_k(i)} - F_y \sum_{i=1}^{i_{\max}^k} x_{\sigma_k(i)} \right\} \\ &= Mg(F_x y_g - F_y x_g) \end{aligned} \quad (6.34)$$

$$D = \begin{bmatrix} i_{\max}^1 & \cdots & i_{\max}^k \\ F_x \sum_{i=1}^{i_{\max}^1} y_{\sigma_k(i)} - F_y \sum_{i=1}^{i_{\max}^1} x_{\sigma_1(i)} & \cdots & F_x \sum_{i=1}^{i_{\max}^k} y_{\sigma_k(i)} - F_y \sum_{i=1}^{i_{\max}^k} x_{\sigma_k(i)} \end{bmatrix} \quad (6.35)$$

$$\mathbf{F} = \begin{bmatrix} F_{\sigma_1} \\ \vdots \\ F_{\sigma_1} \end{bmatrix}, \quad \mathbf{M}_r = \begin{bmatrix} Mg \\ Mg(F_x y_g - F_y x_g) \end{bmatrix} \quad (6.36)$$

とおくと、(6.31), (6.32) は、

$$D\mathbf{F}_\sigma = \mathbf{M}_r \quad (6.37)$$

$$\mathbf{F}_\sigma = D^\dagger \mathbf{M}_r \quad (6.38)$$

と書き直すことができ、搬送方向のみ荷重分散モードを解放した大域的な荷重分散ができるようになる。この搬送方向のみ荷重分散モードを解放した大域的な荷重分散のブロック線図は Fig.6.15 となる。

6.3.2 シミュレーション結果

Fig.6.16 のモデルでシミュレーションした結果を以下に示す。シミュレーション結果は大きく分けて二つある。一つは、 F_x, F_y の比で決まる F のなす角をそれぞれ変えた、 $\angle F = \frac{\pi}{4}$ と $\angle F = -\frac{\pi}{4}$ である。もう一つは、Fig.6.16 の領域 Ω_2 に割り当てられる局所的な荷重分散力 F_{σ_2} を 1, 1.2, 1.6, 1.8, 2.0[N] を変化させた場合を示した。

Figs.6.17, 6.18, 6.19 はそれぞれ、力の角度 $\angle F = \frac{\pi}{4}$ でその力の方向の荷重分散モードを解放した場合のアクチュエータの負荷分布の等高線、力の角度 $\angle F = \frac{\pi}{4}$ で搬送方向の荷重

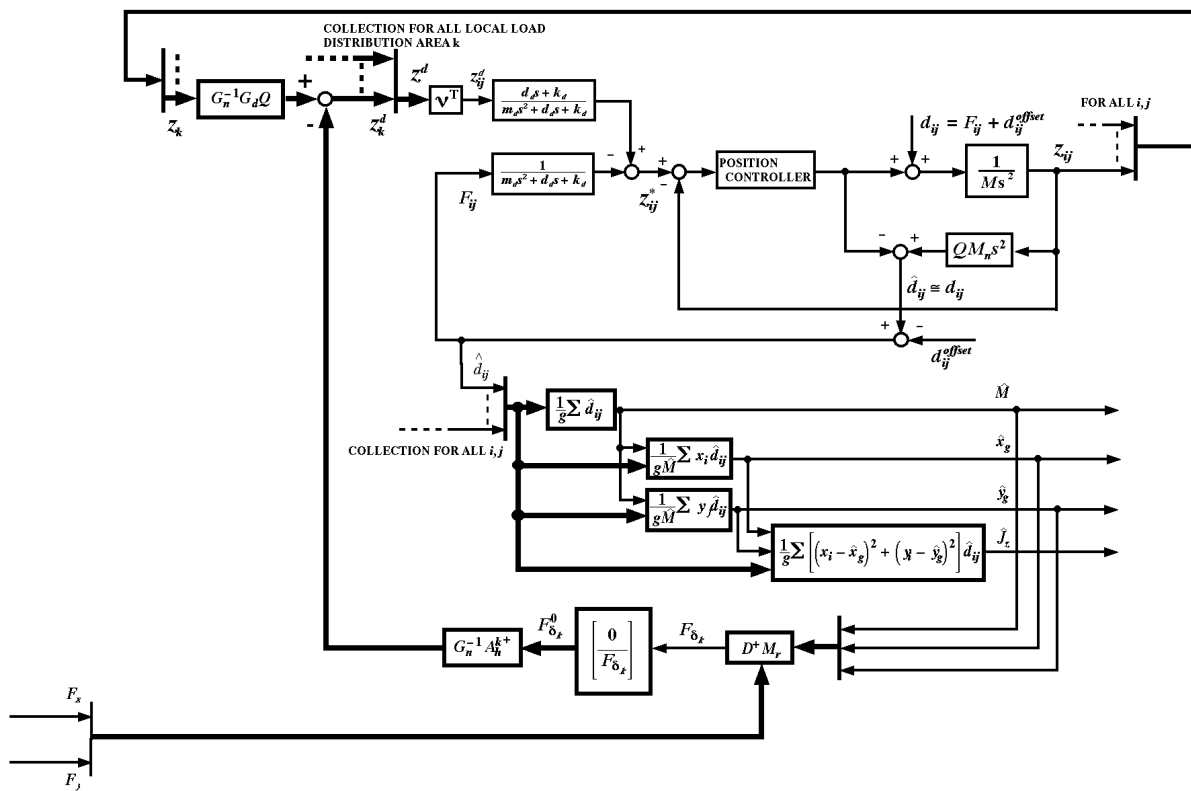


Fig.6.15: 搬送方向のみ荷重分散モードを解放した大域的な荷重分散のブロック線図

分散モードは解放せず単に面インピーダンス制御だけを施したアクチュエータの負荷分布の等高線、力の角度 $\angle F = \frac{\pi}{4}$ で搬送方向の荷重分散モードは解放せず位置制御だけを施した場合である。

Figs.6.20, 6.21, 6.22, 6.23 はそれぞれ、領域 $\Omega_2, \Omega_4, \Omega_1, \Omega_3$ に属するアクチュエータの時間応答である。力の角度 $\angle F = \frac{\pi}{4}$ である。ちなみに、Fig.6.16 のシミュレーションモデルと領域の位置を揃えてあるため、領域の順序がバラバラである。

Fig.6.17 から、Figs.6.18, 6.19 の比較により、提案手法が搬送力の方向の荷重分散モードが解放されていることがわかる。搬送力の方向には非対称の負荷分布が得られ、搬送力の方向と直交する方向には対称の負荷分布が得られている。すなわち、不必要な搬送力の方向 e_{f1} のモーメントは除去され、搬送力の方向に直交する e_{f2} 周りのモーメントはフリーである。

また、Figs.6.20, 6.21, 6.22, 6.23 から、搬送力の方向の荷重分散モードを解放するように決められた荷重分散力 F_σ が各領域のアクチュエータの時間応答から安定に実現されていることがわかる。

Figs.6.24, 6.25, 6.26 に荷重分散モードの解放を x 軸方向逆向きの $\angle F = -\frac{\pi}{4}$ とした場合の、荷重分散モードを解放した場合のアクチュエータの負荷分布の等高線、搬送方向の荷重分散モードは解放せず単に面インピーダンス制御だけを施したアクチュエータの負荷分布の等高線、搬送方向の荷重分散モードも解放せず位置制御だけを施した場合の等高線を示した。 $\angle F = \frac{\pi}{4}$ の場合と比較することにより、任意の搬送力の方向の荷重分散モードの解放ができることがわかる。

Figs.6.27, 6.28, 6.29, 6.30 は $\angle F = -\frac{\pi}{4}$ の方向に荷重分散モードを解放したときの各領域 $\Omega_2, \Omega_4, \Omega_1, \Omega_3$ に属するアクチュエータの時間応答である。Figs.6.20, 6.21, 6.22, 6.23 の $\angle F = \frac{\pi}{4}$ と比較すると分かるように、領域 Ω_4, Ω_3 に属するアクチュエータの応答が逆になっており、荷重分散モードが $\angle F = \frac{\pi}{4}$ から x 軸方向逆の $\angle F = -\frac{\pi}{4}$ になっていることが確認できる。

Figs.6.31, 6.32, 6.33, 6.34, 6.35 は、領域 Ω_2 の荷重分散力 $F_{\sigma_2} = 1, 1.2, 1.5, 1.8, 2.0$ [N] と変えたときのアクチュエータ負荷の等高線である。荷重分散力によらず、荷重分散モードを解放した搬送力の方向には非対称で、それと直交する方向には対称となっているのがわかる。

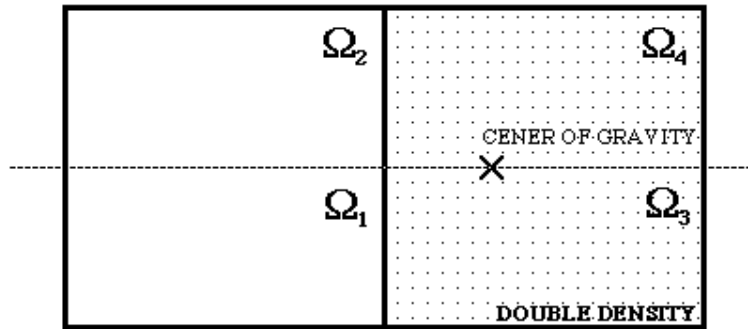


Fig.6.16: シミュレーションモデル

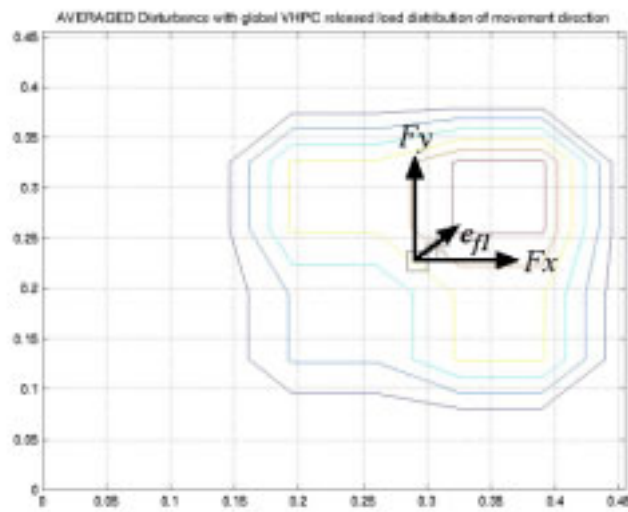


Fig.6.17: 搬送力の方角の荷重分散モードを解放した場合のアクチュエータの負荷分布の等高線 (力の角度 $\frac{\pi}{4}$)

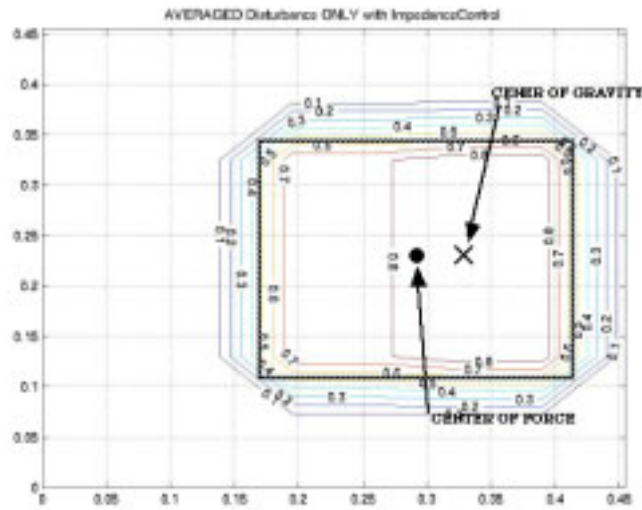


Fig.6.18: 搬送力の方角の荷重分散モードを解放せず、単なる面インピーダンス制御を施した場合のアクチュエータの負荷分布の等高線（力の角度 $\frac{\pi}{4}$ ）

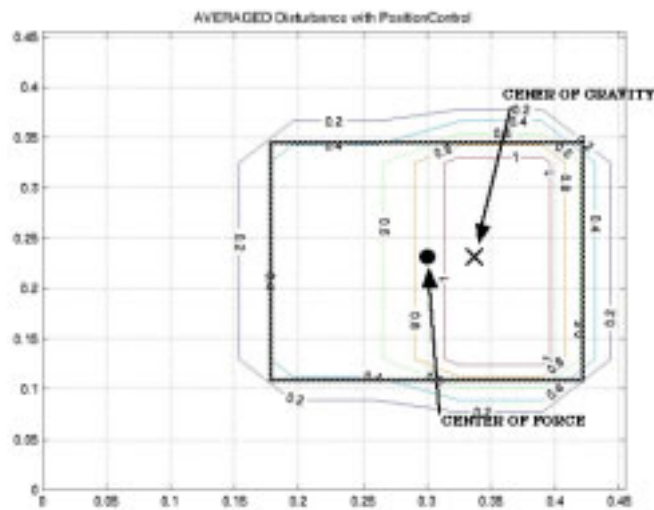


Fig.6.19: 搬送力の方角の荷重分散モードも解放せず、単なる位置制御だけを施した場合のアクチュエータの負荷分布の等高線（力の角度 $\frac{\pi}{4}$ ）



Fig.6.20: Fig.6.16 の領域 Ω_2 におけるアクチュエータの時間応答 (力の角度 $\frac{\pi}{4}$)



Fig.6.21: Fig.6.16 の領域 Ω_4 におけるアクチュエータの時間応答 (力の角度 $\frac{\pi}{4}$)



Fig.6.22: Fig.6.16 の領域 Ω_1 におけるアクチュエータの時間応答 (力の角度 $\frac{\pi}{4}$)

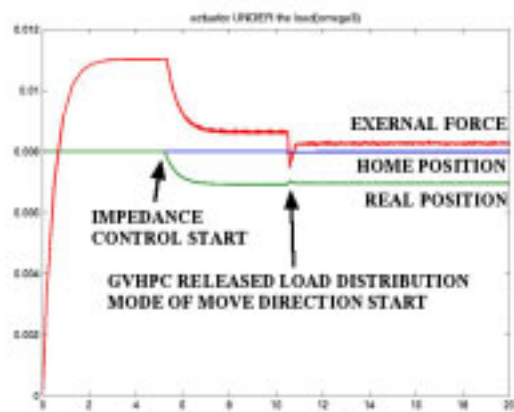


Fig.6.23: Fig.6.16 の領域 Ω_3 におけるアクチュエータの時間応答 (力の角度 $\frac{\pi}{4}$)

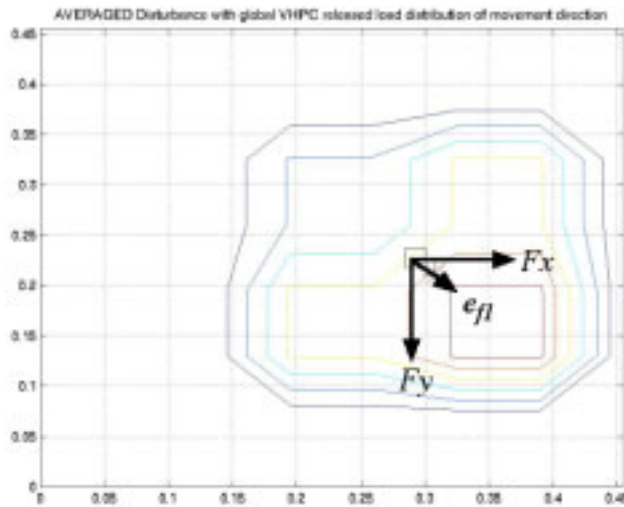


Fig.6.24: 搬送力方向の荷重分散モードを解放した場合のアクチュエータの負荷分布の等高線（力の角度 $\frac{\pi}{4}$ ）

6.4 計りながらつかみ、そして運ぶ

計る動作とつかむ動作の結合を大域的荷重分散法で行った。つかむ動作と運ぶ動作の結合を搬送力方向の荷重分散モードを解放した大域的な荷重分散で行った。これで全ての動作が結合できるようになる。ブロック線図で表すと Fig.6.36 のようになる。

いつも対象物の状態を計り、 $\hat{M}, \hat{x}_g, \hat{y}_g, \hat{J}_z$ を得る。それらの情報をもとにして、重心位置につくむ動作で発生する力の中心がくるようにする。また、搬送目標位置から生成される搬送力方向を元にして、その方向の荷重分散モードの解放を行い搬送もし、荷重分散もする。Fig.6.36 はそういった構造を含んでいるわけである。

第 2 章で対象物は非圧縮性の物質と仮定したが、いつも対象物の状態を計りながらマニピュレーションを行うため、瞬間瞬間では非圧縮性の物質である必要があるが、その静的な状態のつなぎとして、対象物は柔らかいものなどの密度分布が時変のもの、ただしそれほど変化は激しくないものにも対応できるだろう。

6.5 あとがき

本章、第 6 章では、前章までで実現された基本三動作の有機的結合を目指し、計る動作とつかむ動作の結合を大域的荷重分散法で、つかむ動作と運ぶ動作の結合を搬送力方向の

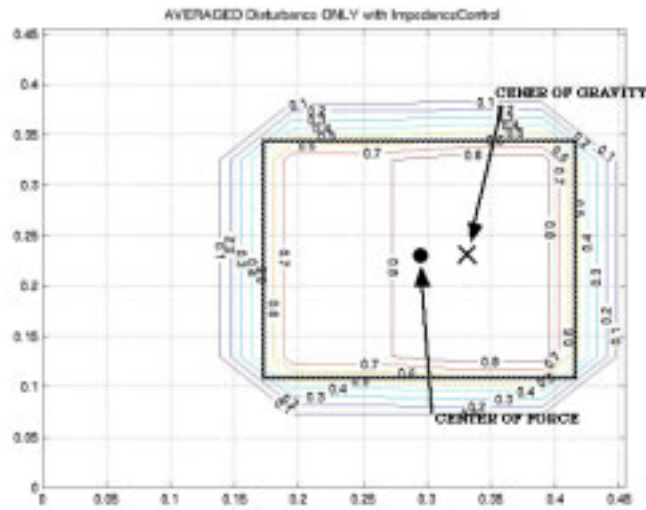


Fig.6.25: 搬送力の方角の荷重分散モードを解放せず、単なる面インピーダンス制御を施した場合のアクチュエータの負荷分布の等高線（力の角度 $-\frac{\pi}{4}$ ）

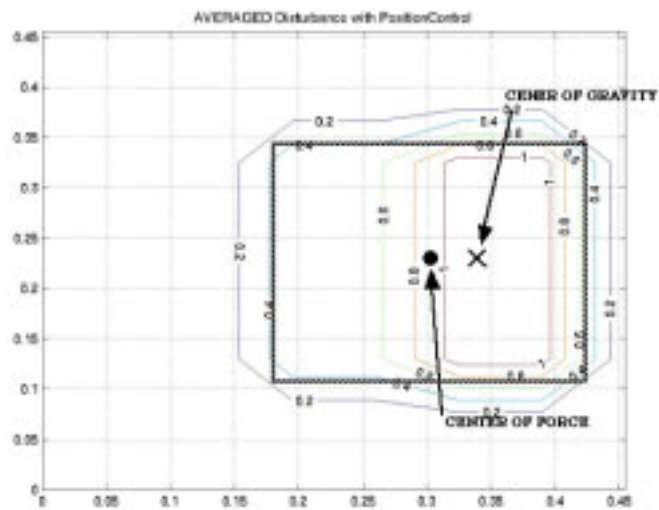


Fig.6.26: 搬送力の方角の荷重分散モードも解放せず、単なる位置制御だけを施した場合のアクチュエータの負荷分布の等高線（力の角度 $-\frac{\pi}{4}$ ）



Fig.6.27: Fig.6.16 の領域 Ω_2 におけるアクチュエータの時間応答 (力の角度 $-\frac{\pi}{4}$)

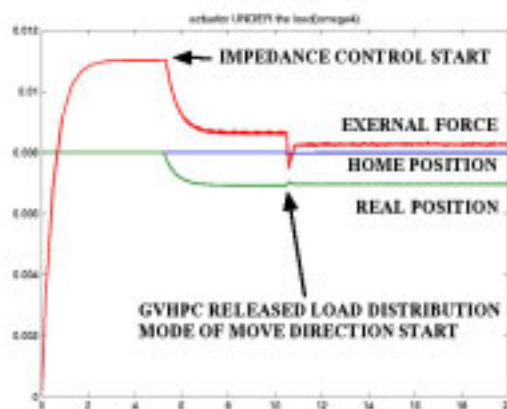


Fig.6.28: Fig.6.16 の領域 Ω_4 におけるアクチュエータの時間応答 (力の角度 $-\frac{\pi}{4}$)



Fig.6.29: Fig.6.16 の領域 Ω_1 におけるアクチュエータの時間応答 (力の角度 $-\frac{\pi}{4}$)

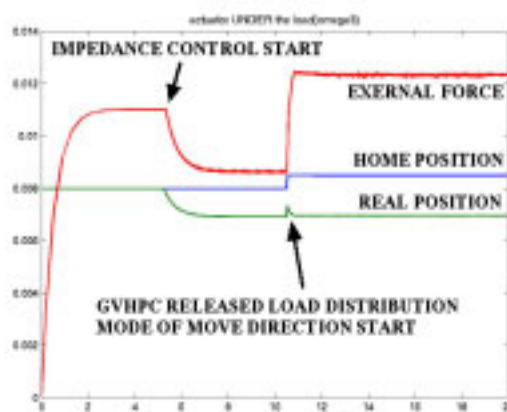


Fig.6.30: Fig.6.16 の領域 Ω_3 におけるアクチュエータの時間応答 (力の角度 $-\frac{\pi}{4}$)

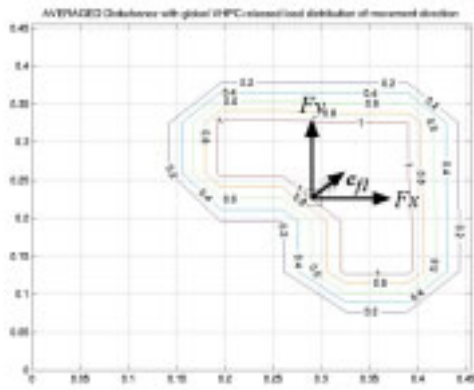


Fig.6.31: Ω_2 の領域の局所的荷重分散力
 $F_{\sigma_2} = 1[\text{N}]$

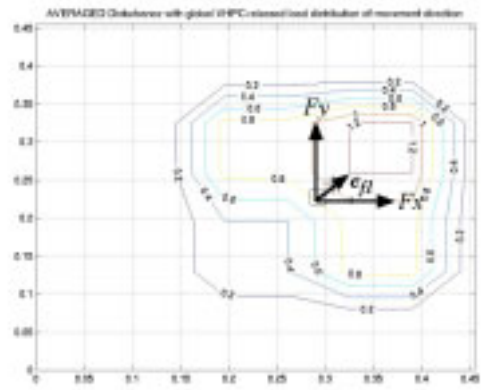


Fig.6.32: Ω_2 の領域の局所的荷重分散力
 $F_{\sigma_2} = 1.2[\text{N}]$

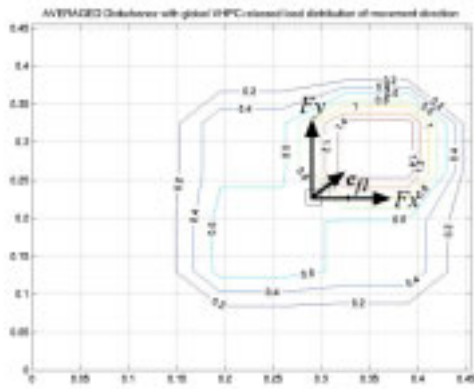


Fig.6.33: Ω_2 の領域の局所的荷重分散力
 $F_{\sigma_2} = 1.5[\text{N}]$

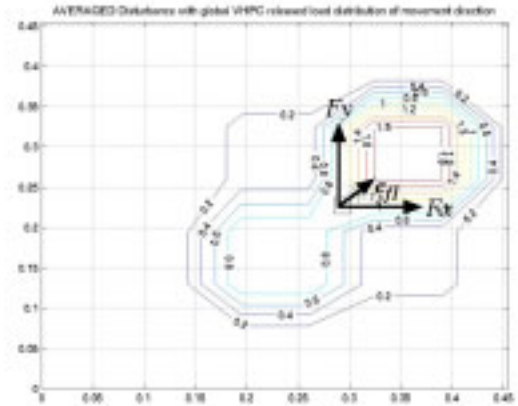


Fig.6.34: Ω_2 の領域の局所的荷重分散力
 $F_{\sigma_2} = 1.8[\text{N}]$

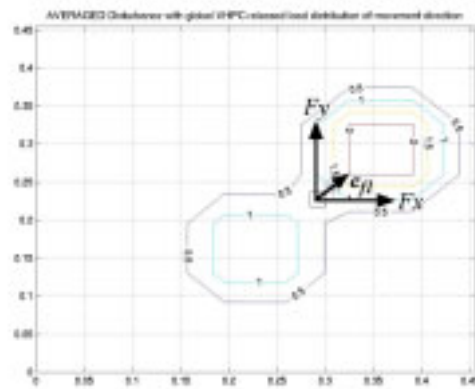


Fig.6.35: Ω_2 の領域の局所的荷重分散力
 $F_{\sigma_k}(2) = 2[\text{N}]$

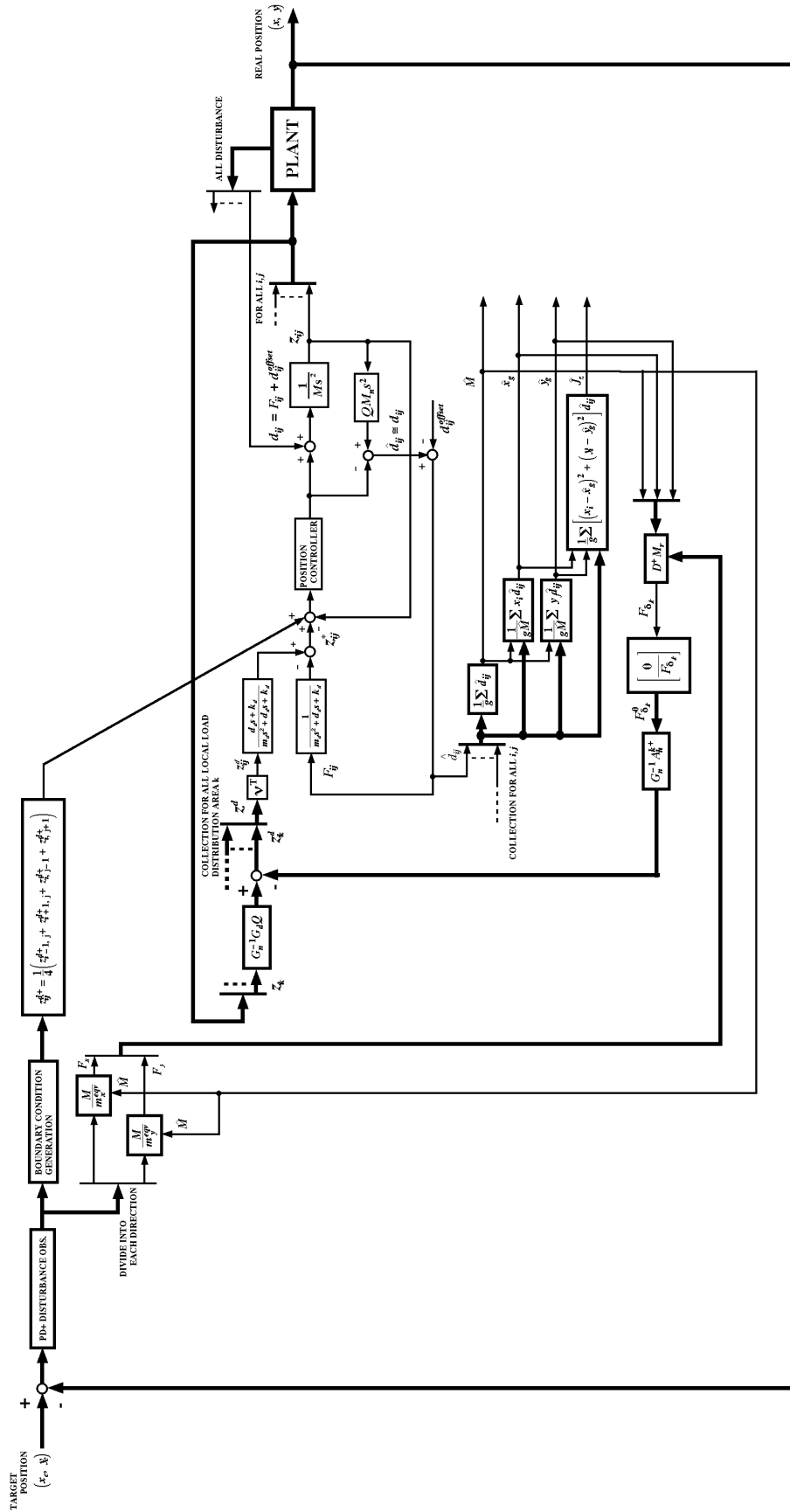


Fig.6.36: 計りながらつかみ、そして運ぶ動作時のブロック線図

荷重分散モードを解放した大域的な荷重分散で行った。それにより、最終的には計りながらつかみ、そして運ぶという全ての基本動作の結合ができるようになった。そのブロック線図は Fig.6.36 であった。

つかむ動作と運ぶ動作の結合は搬送力方向の荷重分散モードを解放した大域的な荷重分散で行ったわけだが、なぜ搬送力方向の荷重分散モードを解放しないと搬送力方向に対象物は運べないのであろうか。

これは分散マニピュレーションを実現するアクチュエータの自由度に起因している。本論文で採用している分散マニピュレータはリニアドライブできるアクチュエータをアレイ化する方式である。そのため、 F_x, F_y といった搬送に寄与する力と、 z 方向の F_z はそれぞれ独立に発生させることができない。

搬送力 F_x, F_y を概算的に求めてみる。対象物全体のポテンシャルエネルギーは微小領域 $dxdydz$ の持つポテンシャルエネルギーの総和と考えると、

$$\begin{aligned} V(x, y) &= Mgz = g \int \rho(x, y, z)z(x, y)dxdydz \\ F_x &= -\frac{\partial V}{\partial x} = -g\frac{\partial}{\partial x} \int \rho(x, y, z)z(x, y)dxdydz \\ &\simeq -g \int \rho \frac{\partial z}{\partial x} dxdydz \end{aligned} \tag{6.39}$$

$$\begin{aligned} F_y &= -\frac{\partial V}{\partial y} = -g\frac{\partial}{\partial y} \int \rho(x, y, z)z(x, y)dxdydz \\ &\simeq -g \int \rho \frac{\partial z}{\partial y} dxdydz \end{aligned} \tag{6.40}$$

となる。ただし、対象物の密度の変化は小さいと考え、 $\frac{\partial \rho}{\partial x}, \frac{\partial \rho}{\partial y} \simeq 0$ とした。これからわかるように、搬送力 F_x, F_y は変位 z の勾配から作られ、変位 z 、それに関係する z 方向の力 F_z は搬送力と独立に生成できない。

そのため、搬送力方向の荷重分散モードの解放を解放するしか、搬送力を発生しながら荷重分散も行う、つかみながら運ぶという複合動作は実現できない。

逆に考えれば、搬送力と荷重分散を行う z 方向の力が独立に出せる構造ならば、荷重分散をやりながら搬送もできる。これは、第 1 章の序論内の分散マニピュレーション研究の現状で紹介した MDMS のような x, y 方向に回転する回転体を有するアクチュエータが z 方向にリニアドライブできる構造を持ち合わせれば実現できる。また本論文で提案しているリニアアクチュエータアレイの先端に回転型のアクチュエータを付けるとも言える。このようなアクチュエータが分散マニピュレータの一つ一つのアクチュエータの最も一般的な構造だと結論付けることができるだろう。

第 7 章

結論

本論文は、近年注目を集めている新しい物体の操作法である分散マニピュレーション法をリアアクチュエータをアレイ状に並べ実現する場合の、制御論的一般的枠組みを構築した。

分散マニピュレーションは多数の支点を通じて対象物に作用し操作する方法で、多数のアクチュエータで実現される。近年のマイクロマシン技術により多数のアクチュエータが簡単で、低コストで実現できるようになり注目されてきた新しく興味深い概念である。このマニピュレーション法は、

1. 多数のアクチュエータで作用するため、きめ細かなマニピュレーションが可能
2. 少数のアクチュエータ故障には無関係、耐故障性の高いマニピュレーションが可能
3. 個々のアクチュエータが低出力でも、多数の支点で支えるため大型で重い物体の操作、パワフルなマニピュレーションが可能

の特長を持つ。

従来、この分散マニピュレーションはマイクロマシン技術の一つとして注目されていたためか、いかにこの分散マニピュレーションを実現するデバイスを製作するかということを中心に研究が行われてきた。また、分散マニピュレーションという概念自体が新しいため、体系的な研究はほとんどなく、体系的な研究が強く必要とされている。

これに対して、本論文では、分散マニピュレーションの基本三動作を、

1. 『計る』
2. 『つかむ』
3. 『運ぶ』

と定義し、いかに作るのではなく、いかに使うかという新しい視点から、制御工学の知見をふんだんに用い、それぞれの動作の実現を目指した。一般的に全てのマニピュレーシ

ンはこの 3 つの基本動作の組み合わせにより実現できるため、この 3 つの動作の実現が分散マニピュレーション法の体系化に大きく寄与する。

さらに分散マニピュレーションの基本三動作 (1) つかむ (2) 計る (3) 運ぶを、有機的に結合し、『計りながらつかむ』、『つかみながら運ぶ』そして『計りながらつかみ、そして運ぶ』という複合動作を実現し、各動作間の連携についても考察した。

すなわち、本論文では、分散マニピュレーション法のマニピュレーション法としての体系化を、

- 各基本動作の実現
- 各動作間の連携

から行った。

第 1 章では、研究の序論として、分散マニピュレーションの定義からはじめ、現在、世界中で繰り広げられている分散マニピュレーション研究について述べ、体系的な研究の必要性を提案した。

第 2 章では、分散マニピュレーションの定義から導かれる基本的性質について言及し、それを分散マニピュレーションの基本要件として述べた。上下方向 (z 軸方向) にドライブするリニアアクチュエータを縦、横の方向 (x, y 軸方向) に多数並べ、そのアクチュエータアレイが対象物に作用することにより、分散マニピュレーションが実現される。高密度でアクチュエータを集積するものの、アクチュエータ間隔は必ず有限の値となる。そのため、空間周波数に関するサンプリング定理から対象物の情報はアクチュエータ間隔に制限された形でしか得られないことを示した。また、対象物の密度から、マニピュレーションに必要な力密度についても言及した。アクチュエータが多数であるが故に、分散マニピュレーション法では、そのマニピュレータの制御に必ず無駄時間が生じる。その無駄時間の量はアクチュエータの並べ方に依存するため、アクチュエータの並べ方によりどの程度になるか考察した。アクチュエータの構造についても考察する必要がある。必要となる力の自由度からアクチュエータの構造は決まった。アルゴリズムのタイプはボトムアップ方式とトップダウン方式の二つの可能性があった。その複合方式の可能性もあることも示した。第 2 章最後では第 3 章以降での論理の展開を助けるため幾つかの現実的な仮説を立てた。

第 3 章では、『計る』の動作では外乱オブザーバアレイの提案を行った。これは従来よく知られていた外乱オブザーバを空間的に分布させ、多重的に配置させるという新しい使い方を提案していた。これにより、分散マニピュレーションにおいて必ず必要な、マニピュレーション対象物の情報、質量、重心位置、慣性モーメントの推定が可能になった。外乱オブザーバは制御の分野では良く知られる技術である。この外乱オブザーバにより、アクチュエータにかかっている負荷量が推定できる。全てのアクチュエータにこの外乱オブザー

バを適用し、それぞれのアクチュエータにかかっている負荷を推定する。それらの情報を収集することにより、対象物の質量、重心位置、慣性モーメントの推定する手法を提案した。これは、外乱オブザーバをアレイ状にした外乱オブザーバアレイにより初めて成し得ることであり、分散マニピュレーション法ならではの手法であった。シミュレーションと実験結果から提案手法により、良好な推定が出来ることが示された。しかしながら、推定値は必ずしも厳密に正しい値ではない。それは外乱オブザーバは必ずしも正確に正確な負荷量が推定できるわけではないことに起因している。しかし、ここで『計られた』情報は『つかむ』『運ぶ』動作で用いるため、ある程度正確な情報であれば、必ずしも厳密に正確な情報である必要はない。すなわち、体重計のようなものである必要はなく、簡単に適用できる方法という意味でこの外乱オブザーバのアレイ化という手法は意味があった。

第4章の『つかむ』の動作では、各アクチュエータにインピーダンス制御を施すことにより、面インピーダンス制御を提案し、堅い面、柔らかい面など自在に実現ができるようになった。多数のアクチュエータが高密度に集積され、多数の支点で作用する場合、その多数の支点はほとんど面として振る舞う。そのため、例えば、スポンジやアルファゲルのような衝撃吸収材のような材料の持つインピーダンスを設定すれば、擬似的にその材料のような振る舞いをさせることができた。

この面インピーダンス制御には二つの設計可能なパラメータがあった。一つはホームポジションであり、もう一つはインピーダンスパラメータであった。

一つ目のパラメータであるホームポジションという、平衡位置を決めるパラメータを可変にすることにより、マニピュレーション対象物の形状に合わせた把持、荷重分散的な把持が可能となった。対象物表面には凹凸があり、その凹凸に応じて、表面形状を変化させ、全てのアクチュエータでやさしく包み込むように把持する。この可変ホームポジション構造なしの場合は、対象物の凹凸のため、一部のアクチュエータに対象物の重さが集中してしまう。それに対して、制御を加え、可変ホームポジション構造を導入すると、多数のアクチュエータで荷重分散的に把持ができるようになった。二つ目のパラメータであるインピーダンスパラメータを時変で変化させることにより、加速度の時間微分値を抑制し、衝撃吸収効果のある面が実現できることを示唆した。インパクトの瞬間は柔らかくして、少しずつ堅い面にインピーダンスパラメータを時変させることによりその衝撃吸収効果が得られた。線形、二次、指数などの方法で時変させ、それぞれ評価した。

第5章の『運ぶ』では自律分散システムの視点を導入し、各アクチュエータの自律性による局所的な制御とアクチュエータ群全体を制御する集中系の二つを組み合わせた自律分散集中複合系という、マクロ制御とミクロ制御の双方向アルゴリズムの提案を行った。

集中系でのアクチュエータ群制御に、フィードフォワードによる方法と、フィードバックによる方法の二つの手法を提案した。それらの手法を適用することにより、搬送物体の

位置制御が可能となった。

フィードフィードによる方法は、制御アルゴリズムが単純で、不安定にならないというメリットを持つ。しかし、目標位置への応答を決める極が設定できないデメリットを持った。

それに対して、フィードバックによる方法は、フィードバックコントローラにより、目標位置への応答を決める極を自由に設定できるようになった。そのため目標値応答特性が設計できるようになった。しかし、第2章で指摘したように分散マニピュレーション法は本質的に無駄時間を含む。フィードバックによる方法ではその無駄時間により不安定化してしまった。その不安定化を抑えるために、外乱補償器を含んだスミス法を適用した。

第6章では、3, 4, 5章で提案し実現した基本三動作『計る』、『つかむ』、『運ぶ』を有機的に結合させることを考察した。『計りながらつかむ』と名付け、計りながらつかむ方法を提案した。『つかむ』で提案した荷重分散的把持を行う場合、対象物の中心と重心位置が異なるもの（質量が分布している）は、上手く把持ができない。そのため、『計る』動作で得られた質量、重心位置、慣性モーメントを活用し、荷重分散把持を行った。それにより、対象物に不用意なモーメントを発生させることなく、さらに荷重分散まで出来た状態で把持ができるようになった。『つかみながら運ぶ』では、対象物に搬送用の力が加わった場合でも、その搬送用の力を考慮した荷重分散を実現した。そこで、搬送方向のみ荷重分散モードを解放し、その方向と直交する方向の荷重分散モードのみを有効にする方法を提案した。第6章最後で、『はかりながらつかみ、そして運ぶ』と名付け、以上で提案した二つの複合動作『計りながらつかむ』、『つかみながら運ぶ』を結合させ、各動作の有機的な結合とした。

上記、各章、各動作の関係は、Fig.7.1の通りである。

また、第2章で考察した分散マニピュレーション基本要件と各動作との関係をまとめ表にすると、Tables 7.1, 7.2, 7.3, 7.4, 7.5のようになる。

Table 7.1 は各動作と空間周波数による制約との関係を、Table 7.2 は力密度の制限との関係、Table 7.3 は無駄時間による制約との関係、Table 7.4 はアルゴリズムのタイプとの関係、Table 7.5 はアクチュエータ構造との関係を示した。

動作の性質上、分散マニピュレーションの基本要件に制限を与えるものもあれば、逆に分散マニピュレーションの基本要件から動作が制限されるものなどがあり、必ずしも何でもできるわけではなく、様々な制約を受けることが各章の議論より明らかとなった。

以上、本論文では、近年注目を集めている新しい概念である分散マニピュレーションの一般的な枠組みの構築のため、基本三動作を定義し、それぞれの動作と各動作間の連携動作の実現を行った。

『計る』動作では、外乱オブザーバレイという新しい手法を提案し、対象物の重量、重心位置、慣性モーメントを推定する手法を提案した。

『つかむ』動作では、面インピーダンス制御という全く新しく分散マニピュレーション法ならではの方法を提案した。ホームポジションを可変にすることにより、対象物の形状に応じた、荷重分散把持が可能となることを示した。また、インピーダンスパラメータを時変させることにより、衝撃の少ない衝撃吸収効果のある面が作れることも示した。

『運ぶ』動作では、自律分散システムの視点を導入し、自律分散集中複合系という各アクチュエータの自律性による局所的な制御とアクチュエータ群全体を制御する集中系の二つを組み合わせ、ボトムアップとトップダウンの双方向アルゴリズムの提案を行った。

『計りながらつかむ』『つかみながら運ぶ』『はかりながらつかみ、そして運ぶ』という各動作の有機的な結合、すなわち連携動作を提案した。

また、各動作の実現に関する議論から、動作の性質上、分散マニピュレーションの基本要件に制限を与えるものもあれば、逆に分散マニピュレーションの基本要件から動作が制限されるものなどがあることがわかった。必ずしも何でもできるわけではなく、様々な制約を受けることが明らかとなった。これは分散マニピュレーション考察上、もっとも礎となる結果といえる。

本論文は分散マニピュレーションの枠組みの構築が主たる目的であるが、それだけでなく、分散マニピュレーションがマニピュレーション法として、他にはない特別な方法で、興味深いことも明らかにしたといえる。

今後、分散マニピュレーション法の研究が盛んに行われ、その土台として本論文が少しでも役に立つことを期待し、本論文の結論とする。

Table7.1: 各動作の分散マニピュレーション基本要件の満足度(その1)

	空間周波数による制約
計る 外乱オブザーバレイ	: $\rho(x, y, z)$ をアクチュエータ間では一定と近似しているため、制限より低いところに仮定
つかむ 面インピーダンス制御	: 無関係
運ぶ 自律分散集中複合系	: フィードバックによる方法で、対象物の位置推定に $\rho(x, y, z)$ がアクチュエータ間では一次関数以下の早さで変化すると仮定
計りながらつかむ 大域的荷重分散	: 計ると同様
つかみながら運ぶ 搬送方向の荷重分散モード解放	: 運ぶと同様

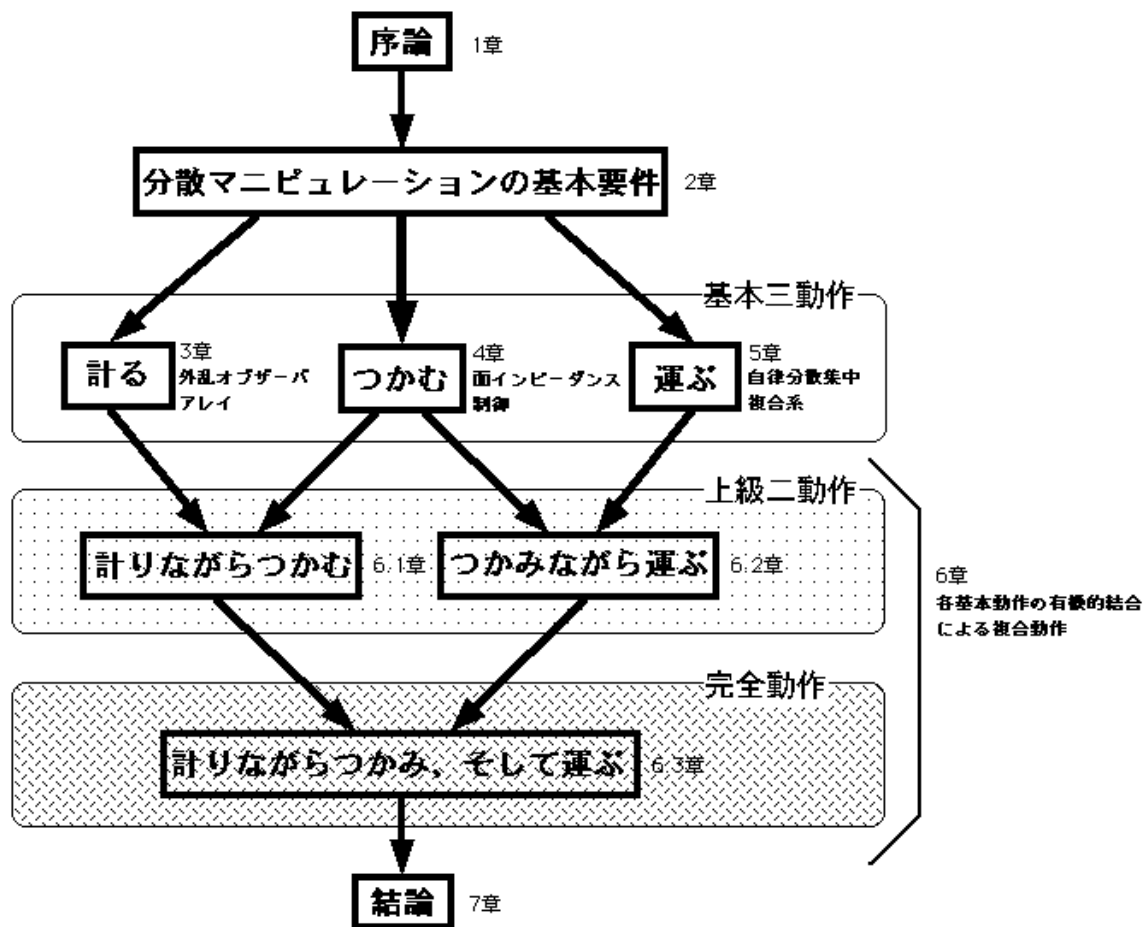


Fig.7.1: 本論文の構成チャート (再掲)

Table7.2: 各動作の分散マニピュレーション基本要件の満足度（その2）

	力密度による制限
計る 外乱オブザーバレイ	: 計る動作は力を発生する必要がないので無関係
つかむ 面インピーダンス制御	: 実現可能なインピーダンスパラメータが力密度により規定される。そのため、衝撃吸収効果は制限を受ける。荷重分散はほとんど無関係
運ぶ 自律分散集中複合系	: ほとんど無関係。パワフルなマニピュレーションが特長だから、衝撃吸収効果と違い、定常的には十分力があると考えられる。
計りながらつかむ 大域的荷重分散	: つかむの荷重分散と同様で、ほとんど無関係
つかみながら運ぶ 搬送方向の荷重分散モード解放	: 運ぶと同様で、ほとんど無関係

Table7.3: 各動作の分散マニピュレーション基本要件の満足度（その3）

	無駄時間による制約
計る 外乱オブザーバレイ	: 出力に無駄時間分だけ遅れが出るが、安定性などには問題ない。よって無関係
つかむ 面インピーダンス制御	: 出力に無駄時間分だけ遅れが出るが、安定性などには問題ない。よって無関係
運ぶ 自律分散集中複合系	: フィードバックによる方法は無駄時間をフィードバックループの中に含むので不安定化してしまう。そこで安定化のために外乱補償器を含んだスミス法を適用
計りながらつかむ 大域的荷重分散	: 計るつかむと同様で無関係
つかみながら運ぶ 搬送方向の荷重分散モード解放	: 運ぶ同様に外乱補償器を含んだスミス法により安定化

Table7.4: 各動作の分散マニピュレーション基本要件の満足度 (その 4)

	アルゴリズムのタイプ
計る 外乱オブザーバアレイ	全てのアクチュエータでの推定外乱量が必要で、その情報を一点に集める必要がある。そのためアルゴリズムはトップダウン方式となる。
つかむ 面インピーダンス制御	各アクチュエータのみのルールで記述できるボトムアップ方式でほとんど記述可能であるが、一部冗長性除去の為に条件がトップダウン方式でないと記述できない。
運ぶ 自律分散集中複合系	提案する自律分散集中複合系はトップダウン方式 & ボトムアップの混合が可能である。提案手法はトップダウン方式に限りフィードフォワードによる方法とフィードバックによる方法を適用。
計りながらつかむ 大域的荷重分散	局所的荷重分散はボトムアップ方式で実現できる。しかし、大域的荷重分散方程式の解の実現はトップダウン方式で実現する必要がある。
つかみながら運ぶ 搬送方向の荷重分散モード解放	搬送方向の大域的荷重分散モード解放はトップダウン方式で実現する必要がある。運ぶは上記同様に混合方式で可能。

Table7.5: 各動作の分散マニピュレーション基本要件の満足度 (その 5)

	アクチュエータ構造
計る 外乱オブザーバアレイ	z 方向に変位可能なアクチュエータとそれに付随する外乱オブザーバが少なくとも必要である。 $F_{ij} = F_{zat}(x_i, y_j)$ を推定する必要がある。
つかむ 面インピーダンス制御	計ると同様に、 z 方向に変位可能なアクチュエータとそれに付随する外乱オブザーバが少なくとも必要である。 $F_{ij} = F_{zat}(x_i, y_j)$ を発生する必要がある。
運ぶ 自律分散集中複合系	F_x, F_y が必要となる。リニアアクチュエータのみの場合は、隣接アクチュエータとの間で生成される勾配を利用し、 F_x, F_y を生成する。
計りながらつかむ 大域的荷重分散	上記、計る、つかむと同様に z 方向のみ必要となる。
つかみながら運ぶ 搬送方向の荷重分散モード解放	F_x, F_y が必要となる。リニアアクチュエータのみの場合は、隣接アクチュエータとの間で生成される勾配を利用し、 F_x, F_y を生成する。 F_x, F_y と F_{ij} が独立ならば、完全な大域的荷重分散をしながら搬送ができる。

第 8 章

今後の展望

本章、本論文最後の第 8 章では、本論文作成により得られた知見をもとに分散マニピュレーションの今後の展望を読む。

分散マニピュレーション発展のキーとなるのは、

1. 分散マニピュレーションの興味深い応用
2. 分散マニピュレーション基礎理論の確立

である。

第 1 章で紹介したすばるのような興味深い応用が分散マニピュレーション発展に大きく寄与するだろう。

分散マニピュレーションは、通常のマニピュレーションと違い特殊なマニピュレーション法である。そのためまず十分に認識されなくてはならない。興味深い応用は、多くの人の目に触れ、認識される機会となる。

分散マニピュレーションは多くの高密度のアクチュエータでの操作であるから、マイクロマシンは実現デバイスとして最有力候補である。そのためマイクロマシンの発展は不可欠である。マイクロマシンデバイスの発展により、そこから生まれる興味深い応用が考えられる。

分散マニピュレーション基礎理論の確立が望まれる。本論文はリニアアクチュエータアレイによる分散マニピュレーション法の一般的な枠組み目指したが、アクチュエータの構造は、リニアドライブすなわち z 方向のみの変位が可能で、 $F_z (= F_i)$ のみ自由に発生できる構造であった。 F_x, F_y といった横力は、隣接するアクチュエータとの勾配を利用し、

$$\begin{aligned} F_x &= -\frac{\partial V}{\partial x} = -g \frac{\partial}{\partial x} \int \rho(x, y, z) z(x, y) dx dy dz \\ &\simeq -g \int \rho \frac{\partial z}{\partial x} dx dy dz \\ F_y &= -\frac{\partial V}{\partial y} = -g \frac{\partial}{\partial y} \int \rho(x, y, z) z(x, y) dx dy dz \end{aligned} \quad (8.1)$$

$$\simeq -g \int \rho \frac{\partial z}{\partial y} dx dy dz \quad (8.2)$$

の形で得ることにした。これはアクチュエータ自身の構造がシンプルになるが、 F_z と F_x, F_y が独立に発生できないという欠点を持つ。これは各々のアクチュエータの自由度が 1 つしかないためである。

これに対して、各々のアクチュエータの自由度が 3 つあり、 $F_x, F_y, F_z (= F_i)$ がそれぞれ独立に発生できる機構が分散マニピュレーションシステムの一般的な機構である。実現の方法としては、第 1 章で紹介した MDMS のような回転体の様な構造を持つアクチュエータで F_x, F_y を実現し、本論文で対象としたリニアドライブで F_z を実現するような、回転リニアタイプの併用したものが考えられる。

本論文の第 6 章のあとがきで考察したように、このような 3 つの自由度を持つアクチュエータが分散マニピュレータの基本アクチュエータならば、荷重分散を行いながらの搬送も可能である。その他にも様々なことが実現可能となるだろう。

この 3 つの自由度を持つアクチュエータが最も一般的な基本アクチュエータ構成である。この 3 自由度のアクチュエータによる分散マニピュレーション法の理論的な方法の確立が必要であると思われる。

また、分散マニピュレーション法は多数のアクチュエータで構成されるため、アルゴリズムの形式も本論文で示したとおり、ボトムアップ方式とトップダウン方式の双方の可能性もある。またそれらを混合した形も大変興味深い。システムの形式の考察も分散マニピュレーション法の確立に大きく必要だと思われる。

アクチュエータ間隔 $\Delta x, \Delta y$ を可変にするのはどうだろうか。本論文ではアクチュエータ間隔は一定の設定であったが、構造上の問題がなければ可変にすることも考えられる。可変にすれば、対象物が重い部分にはアクチュエータを集中的に再配置させ分散マニピュレーションをすることが可能になるだろう。

普通の(集中)マニピュレーションと分散マニピュレーションの組み合わせにより高度なマニピュレーションが可能となるだろう。例えば、人間の腕は腕の先まではアクチュエータが直列につながる通常のマニピュレーションであり、その先にある 5 本の指は対象物に対して並列に作用する分散マニピュレーションである。人間の腕の非常にしなやかで自在な対象物の操作は、普通のマニピュレーションと分散マニピュレーションの複合により得られている。

通常のマニピュレーションと分散マニピュレーションを複合したマニピュレーション法の一般的な枠組みの研究が望まれるだろう。

謝辞

本研究をすすめるにあたり、多くの方々からさまざまなご指導・ご助言をいただきました。指導教官である堀洋一教授には、ご多忙にもかかわらずさまざまな視点からいろいろなタイミングで適切なご指導をいただきました。

私は博士課程2年の時に、大病を患ってしまい4ヶ月ほどの入院を余儀なくされました。無事退院はできたものの、研究は上手くいかず、退院後であまり体調も良くなかったため、大学院を中退しようと考えたことがありました。そのとき、堀先生が必死になって励ましてくださいました。そのとき先生の励ましがなかったら、こうやって博士論文を書き上げることはできませんでした。残念ながら私の病気は一生治りません。しかし論文を書き上げることにより、これから病気と折り合いをつけながら、仕事に研究に励んでいける自信ができました。堀先生をお師匠様に持てたこと、堀先生の元で勉強できたこと本当に幸せに思っております。私の一生の誇りです。堀先生、本当にありがとうございます。

本学工学系研究科精密機械工学専攻新井民夫教授には、入院中に開催され出席が不可能となった国際会議の代理発表をしていただきました。新井先生には、先生ご自身の経験から、研究も病気の治療ものんびりやるのが良いとアドバイスを頂きました。そのアドバイスのおかげでずいぶん心救われたのを、今でもよく思い出します。新井先生、ありがとうございました。

山形大学工学部情報科学科神谷淳助教授には、卒業論文の指導はじめ山形大学在学中大変お世話になりました。神谷先生が大学院進学を勧めてくれなければそもそもスタート地点に立つことができませんでした。ありがとうございます。

立命館大学理工学部機械工学科小西聡助教授には、本論文題目を決めるにあたり大変重要なアドバイスを頂きました。論文題目は論文の顔ですので非常に重要でした。ありがとうございます。

私が所属していた電気系専攻の諸先生方には大変お世話になりました。大学院輪講や審査などで研究に関する有意義なアドバイスを多くいただきました。感謝いたします。

内田利之技術官にはお世話になりました。私の博士論文の幾つかあるポイントの一つは実験装置の製作です。非常に多くの時間と情熱をかけて、実験装置の製作を行いました。この実験装置の製作は、内田さんの協力なしには考えられませんでした。また、日々の研究

室の生活においても大変お世話になりました。私の 6 年間の堀研究室生活ではお世話になり続けました。内田さん、ありがとうございます。

電気系工作室の渋谷武夫技術官にも実験装置の製作で大変お世話になりました。旋盤などの工作機の使い方や材料の購入などいろいろとお世話になりました。

研究室では先輩であり、現在では各方面で活躍されている坂井真一郎さん、鈴木文泰さん、藤本博志さんには研究のことはもちろんのこと、年齢が近かったこともあり私にとってお兄さんのような存在であり本当にお世話になりました。ありがとうございます。身近にすばらしい先輩を持てた自分はとても幸せだったと思っております。

修士課程の 2 年間一緒に過ごした小田井正樹君にもとてもお世話になりました。一緒に勉強したり、お酒を飲んだりと研究室での生活全般においてお世話になりました。ありがとう。

呉君、佐渡君、関君、堀君、徳田君、原君、北島君、角嶋君、鹿野君、片岡君、馬君、坂東君、畠君、丸谷君、宮本君、北川君、岡崎君、張さん、岡野君、タイ君、井上さん、谷君、猿田君、呉君、古川君、福井君、長瀬君、大埜君、磯君、高橋君、佐藤君、小安君、青木君といった素晴らしい研究室の後輩たちに恵まれ、充実した研究室生活を送れたことに対して感謝いたします。ありがとうございます。

堀先生の秘書をされていた高木貴代（旧姓 宇佐美）さん、木村麻里子さんにも大変お世話になりました。女性ならではのきめ細やかな気配りでいろいろと助けていただきました。ありがとうございます。

最後になりましたが、母親に感謝したいと思います。早くに父親を亡くしておりますので、母親には並々ならぬ苦勞をかけました。こうやって論文を書き上げることができたのは母親の経済的、精神的なサポートがあったおかげです。

その他、ここには書き切れませんがたくさんの方々にお世話になりました。ここに感謝し謝辞と代えさえていただきたいと思います。

付録 A

可変ホームポジション（荷重分散の数理的な意味）

- 『つかむ』 -

可変ホームポジション決定式は下記の式で与えられた。この式は境界条件の問題さえ除けば、基本的にはある力がアクチュエータに印加されたとき、そのアクチュエータは隣接するアクチュエータの発生する力の平均値を出力するように、ホームポジションを可変させる。

可変ホームポジション決定式

$$z^d = A_h^\dagger A_0 G_d G_n^{-1} Q z + A_h^\dagger S \quad (\text{A.1})$$

すなわち、発生する力を平均値にするという拘束をかけていることになる。アクチュエータが無限の密度で分布していると考え、アクチュエータの発生する力の分布は調和関数となり、Laplace 方程式が拘束条件となる。 (x, y) で発生すべき力 $F = F(x, y, t)$ は、

$$\frac{\partial^2 F}{\partial x^2} + \frac{\partial^2 F}{\partial y^2} = 0 \quad (\text{A.2})$$

という拘束を受け、これが荷重分散をするということである。ちなみに、膜などの変位は振動方程式 (A.3) で解が拘束され振動解が、熱も熱の拡散方程式 (A.4) などで拘束を受けるわけである。

$$\frac{\partial^2 z}{\partial t^2} = c^2 \left(\frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2} \right) \quad (\text{A.3})$$

$$\frac{\partial u}{\partial t} = \kappa \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (\text{A.4})$$

さてところで、各アクチュエータはインピーダンス制御が施されているから、

$$m_d \frac{d^2 z}{dt^2} + d_d \frac{d}{dt} (z - z^d) + k_d (z - z^d) = F \quad (\text{A.5})$$

である、ただし、 $z = z(x, y, t)$, $z^d = z^d(x, y, t)$ である。(A.2) を (A.5) を用いて書き直すと、

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \left(d_d \frac{\partial}{\partial t} + k_d \right) z^d = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \left(m_d \frac{\partial^2}{\partial t^2} + d_d \frac{\partial}{\partial t} + k_d \right) z \quad (\text{A.6})$$

を得る。 $m_d \frac{\partial^2}{\partial t^2} + d_d \frac{\partial}{\partial t} + k_d$ を演算子 I_{mp} とおくと、

荷重分散方程式

$$\nabla^2 \left(I_{mp} - m_d \frac{\partial^2}{\partial t^2} \right) z^d = \nabla^2 I_{mp} z \quad (\text{A.7})$$

を得る。

付録 B

製作した実験装置

研究の主たる目的は多数のアクチュエータ（例えばマイクロマシン）により構成される分散マニピュレーションシステムの構築とそのアルゴリズムの開発である。提案システムの実験装置の制作を行い、その実験装置に適合するアルゴリズムの開発からより一般的な分散マニピュレーション法のアルゴリズムを考える。分散マニピュレーションの最大の実現可能デバイスマイクロマシンだろう。 Fig.B.1 に制作する実験装置の基本構成図を示

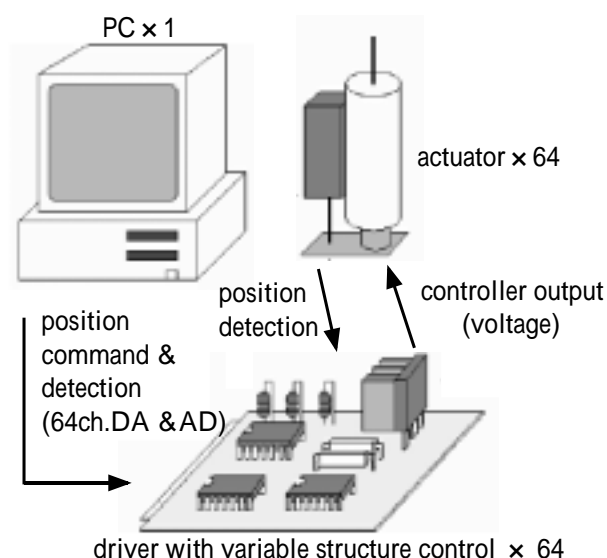


Fig.B.1: 実験装置の基本構成図

す。実験装置は64個のモジュールにより構成される。具体的には64個のアクチュエータ、センサ、可変構造を持つ位置制御系、1台のPCである。64個のプロセッサは1つのPCの中で仮想的に実現することとする。すなわち、実験装置自体は集中システムである。

ところで実験装置をリアルサイズで構成するために発生する問題が幾つか考えられる。前もって整理しておく。

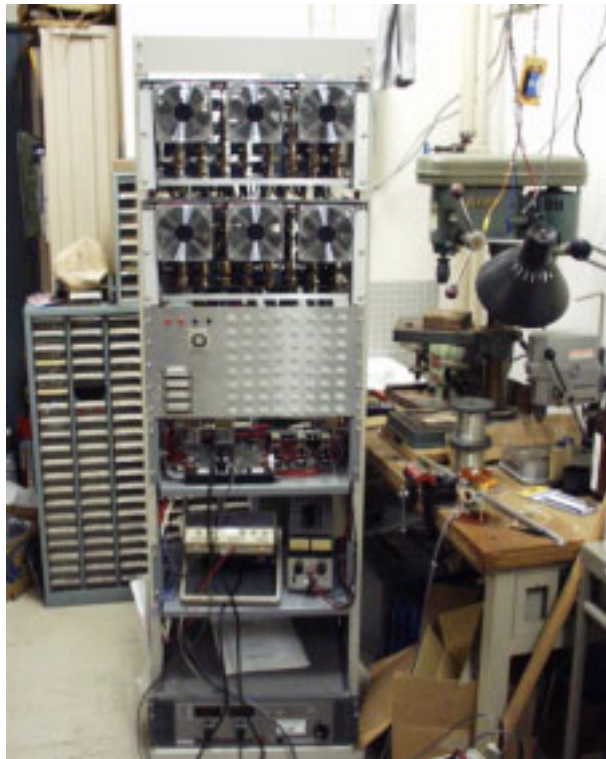


Fig.B.2: 実験装置のアクチュエータ駆動用ドライバと電源

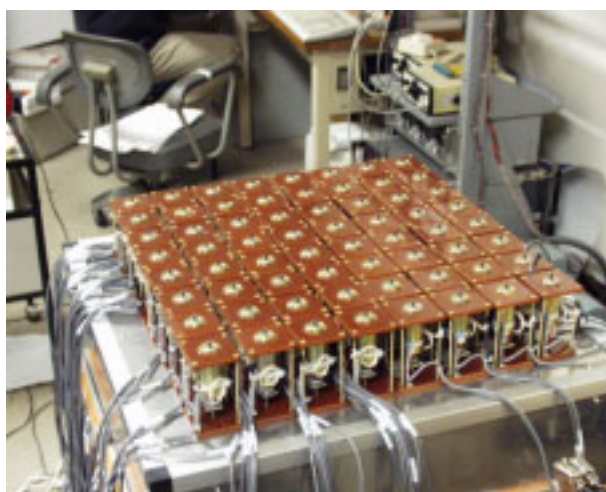


Fig.B.3: アクチュエータ群

1. 生成される場は重力に依存

マイクロな世界で物体を重力を使って運ぶことは極めて非現実的なことである。なぜなら、サイズが小さくなればなるほど、摩擦力などの表面力が支配的となるからである。どんな力が支配的かはマイクロ世界とマクロ世界では全く異なる。マクロ世界で重力を使って物体を転がしながら運ぶことは自然なことであるけど、マイクロ世界ではそうではない。

2. ソレノイドプランジャの短いストローク

ソレノイドプランジャは一般的に極めて強い非線形を持ち、ストロークは短い。しかしながら、ソレノイドプランジャは極めて大きな力が発生でき、高速な動作が可能である。もし、ソレノイドプランジャより長いストロークが必要なならボールスクリュウタイプのアクチュエータを使うなどの選択肢がある。

生成される場が重力に依存することやアクチュエータのストロークが短いことは、どちらも実験装置の制作上の技術的な問題である。制作した実験装置の役割は提案するアルゴリズムの検証とし、これらの問題はあまり深く考えないこととする。

この付録では個々のアクチュエータの位置制御の設計を行う（搬送物体の位置制御ではない）。具体的には製作した実験装置のアクチュエータを取り上げながら、多数のアクチュエータで構成される物体搬送システムを考えたとき、個々のアクチュエータの制御はどのようなものがよいか、実際に 3 タイプの制御アルゴリズムを適用を試み、考察してみるとした。適用した 3 タイプは、

1. P 制御のみの方法

シンプルだが制御性能は低い。

2. PD 制御と外乱オブザーバによる方法

制御性能は高いが複雑。

3. 可変構造制御 [42][†] による方法

そこそこシンプルで、そこそこ制御性能も高いが高速なフィードバックが必要となる。

である。それぞれ、一長一短がある。多数のアクチュエータで構成される物体搬送システムの個々のアクチュエータの持つべき特徴は、

1. 高速なフィードバックが実現できる。

2. 多数あるから個々の制御ロジックまたは回路はシンプルな方がよい。

[†]スライディングモード制御は可変構造制御の代表として、その優れた制御性能がよく知られている。

であるから、可変構造制御が最も適している。以下では可変構造制御を実験装置のアクチュエータであるソレノイドプランジャに適用し設計を行う。Fig.B.4 に採用したソレノイドプランジャの構造図を、Fig.B.5 にソレノイドプランジャの電磁気モデルを示す。位置制御系を実現するためにソレノイドプランジャには位置センサが付加されている。単純な Fig.B.5 の電磁気モデルを用いれば、ソレノイドプランジャに発生する力を求めることができる [43]。ソレノイドプランジャに発生する力は Ampere の周回積分を用い、(B.1) の形で求めるこ

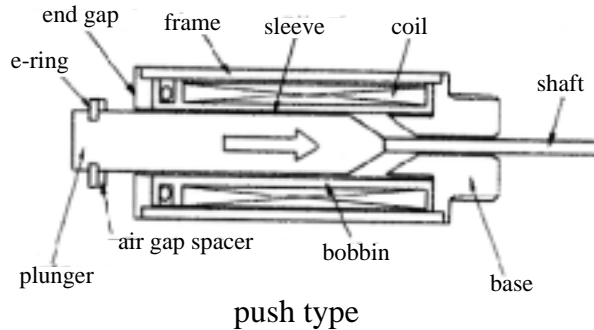


Fig.B.4: ソレノイドプランジャの構造

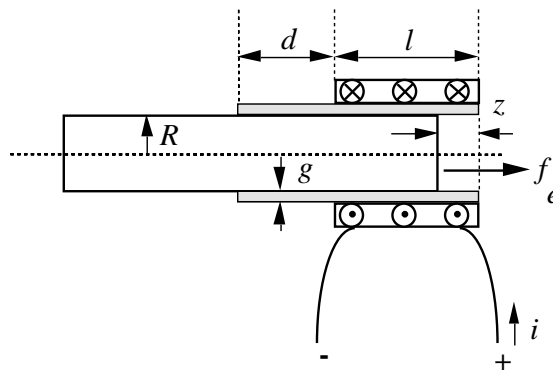


Fig.B.5: ソレノイドプランジャの電磁気モデル

とができる。得られた式から力は電流 i の 2 乗に比例し、位置 z の 2 乗に反比例することがわかる。

$$f_e = f_e(z^{-2}, i^2) = -\frac{2\mu_0 N^2 \pi R^2}{(2dz + gR)^2} i^2 \quad (\text{B.1})$$

ただしここで、 μ_0 , N , R , g , π , i はそれぞれ真空の透磁率、巻数、プランジャの半径、ギャップ長、円周率、ソレノイドに流れる電流である。ソレノイドプランジャは非線形性の強い

ため、ロバストなコントローラが必要となる。我々はいくつかのソレノイドプランジャ位置制御用のコントローラを制作し、その結果として可変構造を持つコントローラを採用することとする（スランディングモード制御は可変構造制御の一つとしてその優れたロバスト性がよく知られている）。

Fig.B.6 に採用した可変構造制御系のブロック線図を示す。また、Fig.B.7 は実際に制作した回路図である。Fig.B.6 では、コンパレータが Fig.B.7 では LM319 が可変構造を生成している。また、Fig.B.6 中にある HPF は目標位置 z^* 付近での実際の位置 z のチャタリングを防止するために入れてある。Fig.B.7 の回路で駆動されたソレノイドプランジャの位置

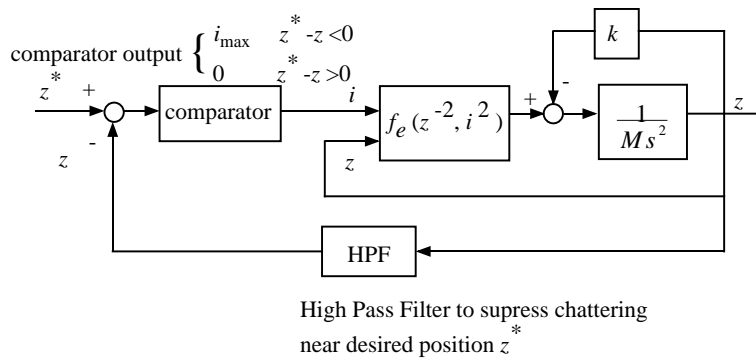


Fig.B.6: 可変構造機構とハイパスフィルタを含んだソレノイドプランジャの位置制御ブロック線図

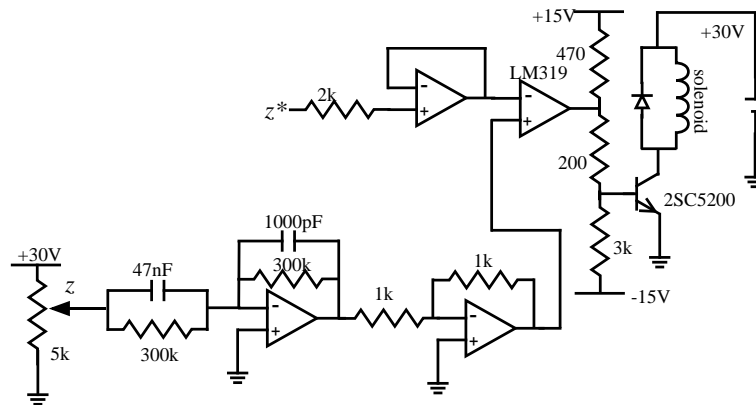


Fig.B.7: 可変構造制御を実現するアナログ電子回路 [44], [45], [46]

応答を Figs.B.8,B.9 に示す。Fig.B.8 はシミュレーション結果であり、上はハイパスフィルタがない場合、下がハイパスフィルタがある場合である。ハイパスフィルタを追加することにより、チャタリングが抑制され、良好な位置応答を示しているのがわかる。また実際

の実験結果である Fig.B.9 からソレノイドプランジャは強い非線形性があるにもかかわらず、実際の位置は指令値に良好の追従していることがわかる。

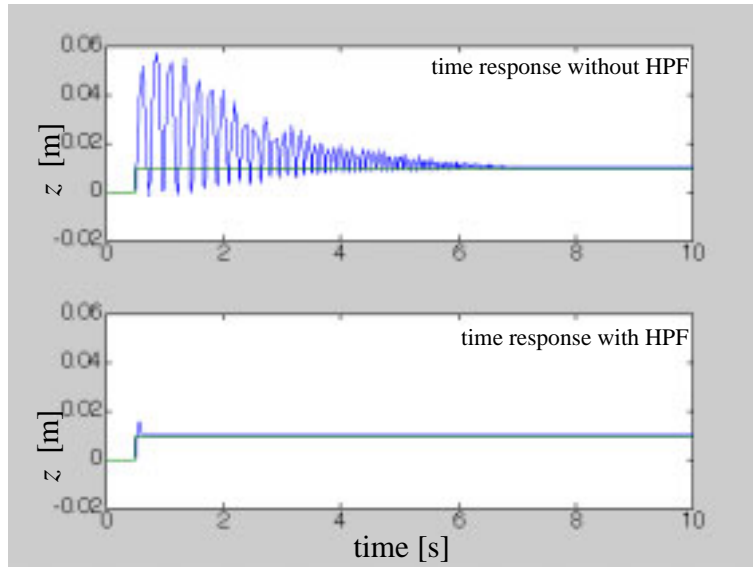


Fig.B.8: アクチュエータの位置応答（シミュレーション）上図：w/o HPF、下図：with HPF

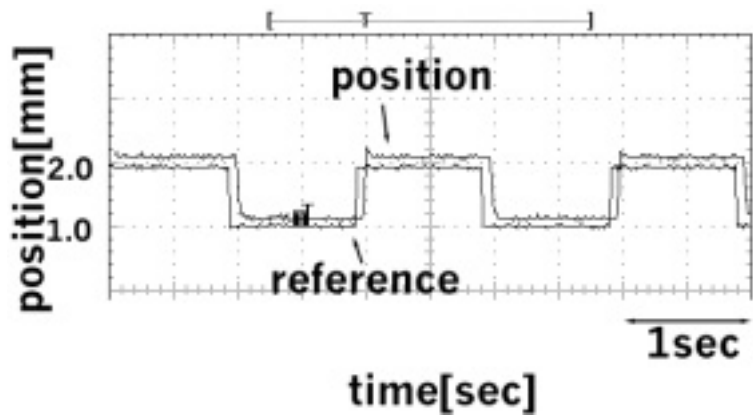


Fig.B.9: アクチュエータの位置応答（実験）上図：w/o HPF、下図：with HPF

研究業績

学会誌論文、紀要等

1. 及部 七郎齋, 堀 洋一, ”自律分散型 magic carpet の提案と設計”, 東京大学工学部附属総合試験所年報, vol.57, pp.221-226,1998
2. 及部 七郎齋, 堀 洋一, ”リニアアクチュエータアレイによる物体搬送の試み - 魔法のじゅうたんを目指して - ”, 電学論 D, vol.121, no.12, pp.1250-1255, 2001 (⇒ 本論文第 5 章に対応)
3. 及部 七郎齋, 堀 洋一, ”外乱オブザーバアレイによる対象物の質量、重心位置、慣性モーメントの推定 - リニアアクチュエータアレイによる分散マニピュレーション法の確立を目指して - ”, (投稿準備中) (⇒ 本論文第 2 章と第 3 章に対応)
4. 及部 七郎齋, 堀 洋一, ”面インピーダンス制御による対象物の荷重分散的把持 - リニアアクチュエータアレイによる分散マニピュレーション法の確立を目指して - ”, (投稿準備中) (⇒ 本論文第 2 章と第 4 章と第 6 章に対応)

国際会議論文

5. H.Oyobe , H. Kitajima, Y.Hori, ”Design and Realization of Autonomous Decentralized Object Transfer System:Magic Carpet”, in Proc. 6th IEEE International Workshop on Advanced Motion Control(AMC2000), pp.25-29, 2000
6. H.Oyobe, R. Marutani, Y.Hori, ”Experimental Manufacturing of Object Transfer System ”Magic Carpet” Consisting of Actuator Array with Autonomous Decentralized Control”, in Distributed Autonomous Robotic System 4(DARS4), Springer-Verlag, pp.437-446, 2000
7. H. Oyobe and Y. Hori, ”Object Conveyance System ”Magic Carpet” Consisting of 64 Linear Actuators - Object Position Feedback Control with Object Position Estimation - ”, in Proc. 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics(AIM2001), pp.1307-1312, 2002

国内会議、研究会等

8. 及部七郎齋, 堀 洋一, ”場の概念を用いた自律分散型 magic carpet の提案と設計”, 平成 10 年電気学会産業応用部門全国大会, pp.379-382, 1998
9. 及部七郎齋, 北島博晃, 堀 洋一, ”自律分散型物体搬送システム Magic Carpet の設計と試作”, 平成 11 年電気学会産業応用部門大会, pp.417-420, 1999
10. 及部七郎齋, 堀 洋一, ”場の概念を用いた自律分散型 magic carpet の提案と設計手法の分類”, 第 11 回自律分散システム・シンポジウム, pp. 267-270, 1999
11. 及部七郎齋, 堀 洋一, ”魔法のじゅうたん - 64 個のリニアアクチュエータによる物体搬送実験 -”, 平成 12 年電気学会産業応用部門大会, pp.577-580, 2000 (電気学会優秀論文発表賞受賞)
12. 及部七郎齋, 堀 洋一, ”外乱オブザーバレイによる魔法のじゅうたん上の物体の重心位置、重量、慣性モーメントの推定”, 平成 14 年電気学会産業計測制御研究会, pp.31-35, 2002 (電気学会優秀論文発表賞受賞)

共著論文

13. 丸谷 良, 及部七郎齋, 堀 洋一, ”超多数のアクチュエータによる自律分散型物体搬送システム Magic Carpet の試作”, 平成 12 年電気学会産業計測制御研究会, pp.25-29, 2000

参考文献

- [1] K. -F. Böhringer, H. Choset, "Distributed Manipulation", Kluwer Academic Publishers
- [2] NHK 科学番組部編, "『すばる』が捉えた驚異の大宇宙", NHK 出版
- [3] http://SubaruTelescope.org/j_index.html
- [4] 三菱電機技報, vol.73, no.12, 1999
- [5] 藤田博之, "自律分散マイクロシステム", 日本機械学会誌, Vol.97, No.905, pp.298-301, 1994
- [6] S. Konishi, Y. Mita, H. Fujita, "Autonomous Distributed System for Cooperative Micromanipulation", Distributed Manipulation, pp.87-102, Kluwer Academic Publishers
- [7] S. Konishi, Y. Mita, H. Fujita, "Two-Dimensional Conveyance System Using Cooperative Motions of Many Fluidic Microactuators", Jour. Advanced Robotics, vol.12, no.2, pp.155-165, 1998
- [8] 三田吉郎, Andreas Kaiser, Patrick Garda, Maurice Milgram, 藤田博之, "形状認識を行なう分散型マイクロ搬送システム, 第11回自律分散システムシンポジウム, pp.259-262, 1999
- [9] Thorbjörn, Johan Ulfstedt Mattsson, Edvard Kälvesten, Göran Stemme, "A Walking Silicon Micro-Robot", Tech. Digest Transducers '99, vol.2, pp.1202-1205, 1999
- [10] J. Luntz, W. Messner, H. Choset, "Stick-Slip Operation of the Modular Distributed Manipulator System", In Proc. American Control Conference, pp.3853-3857, 1998
- [11] J. Luntz, W. Messner, H. Choset, "Parcel Manipulation and Dynamics with a Distributed Actuator Array: The Virtual Vehicle", In Proc. the IEEE International Conference on Robotics and Automation(ICRA), pp.1541-1546, 1999

- [12] J. Luntz, W. Messner, H. Choset, "Discreteness Issues in Actuator Arrays", In Proc. Workshop on Distributed Manipulation at the International Conference on Robotics and Automation, 1999
- [13] K. -F. Böhringer, B. R. Donald, R. Mihailovich, N. C. MacDonald, "A Theory of Manipulation and Control for Microfabrication Actuator Arrays", In Proc. 7th IEEE International Workshop on Micro Electro Mechanical System(MEMS), pp.102-107, 1994
- [14] J. W. Suh, R. B. Darling, K. -F. Böhringer, B. R. Donald, H. Bltes, T. A. Kovacs, "CMOS Integrated Ciliary Actuator Array as General-Purpose Micromanipulation Tool for Small Objects", Journal of Microelectromechanical Systems, vol.8, no.4, pp.483-496, 1999
- [15] K. -F. Böhringer, V. Bhatt, K. Y. Goldberg, "Sensorless Manipulation Using Transverse Vibrations of a Plate", In Proc. the IEEE International Conference on Robotics and Automation(ICRA), pp.1989-1996, 1995
- [16] K. -F. Böhringer, B. R. Donald, L. E. Kavraki, F. Lamiroux, "Part orientation with one or two stable equilibria using programmable vector fields", In Proc. Workshop on Distributed Manipulation at the International Conference on Robotics and Automation, 1999
- [17] T. Fukuda, et al., "Distributed Control of Flexible Transfer system using Learning Automata", In Proc. the IEEE International Conference on Robotics and Automation(ICRA), pp.96-101, 1999
- [18] <http://www.mein.nagoya-u.ac.jp/activity/1999/FTS1-99J.html>
- [19] 斎藤恒雄, "画像処理アルゴリズム", 近代科学社 (1993)
- [20] V.D. バーガー, M.G. オルソン, "力学—新しい視点にたって—, 培風館 (1975)
- [21] 堀洋一, 大西公平, "制御工学の基礎", 丸善 (1997)
- [22] 堀洋一, 大西公平, "応用制御工学", 丸善 (1997)
- [23] 武者利光, 岡本良夫, "逆問題とその解き方", オーム社 (1992)
- [24] 志村康治, 堀 洋一, "ロボットマニピュレータにおける力制御のロバスト化と衝突過程の制御", 日本ロボット学会論文誌, Vol.11, No.2, pp.235-245, 1993

- [25] 志村康治, "ロボットマニピュレータにおける力制御のロバスト化と衝突過程の制御", 東京大学修士論文, 1992
- [26] 新誠一, 池田建司, 湯浅秀男, 藤田博之, "自律分散システム", 朝倉書店 (1995)
- [27] 伊藤正美, "自律分散システム研究の課題と将来", 計測自動制御学会誌, Vol.32, pp.789-796, 1993
- [28] 村田智, "均質ユニットによる機械のかたちづくり", 日本ロボット学会誌, Vol.15, No.5, pp.675-679, 1997
- [29] Dragoslav D.Siljak, "Large-scale dynamic systems:stability and structure", Elsevier North-Holland(1978)
- [30] 石田好輝, "自律分散システムの情報処理の特徴", 計測自動制御学会誌, Vol.32, pp.830-836, 1993
- [31] 深尾毅, "分散システム論", 昭晃堂 (1987)
- [32] 吉田勝久, 佐藤昭, "基礎システム理論", コロナ社 (1978)
- [33] O. J. M. Smith, "A cotroller to Overcome Dead Time", ISA Journal, Vol.6, No.2, p.28-33, 1959
- [34] 示村悦二郎, "むだ時間システムの制御 - 研究の歴史とその将来", システムと制御, Vol.28, No.5, pp.265-268, 1984
- [35] 渡部慶二, "むだ時間システムの制御", コロナ社
- [36] 渡部慶二, 伊藤正美, "入・出力にむだ時間を含むシステムの制御", システムと制御, Vol.28, No.5, pp.269-277, 1984
- [37] 荒木光彦, 桑田龍一, 藤中透, 安田一則, 渡部慶二, "むだ時間システムを料理する - 数値例による検討", システムと制御, Vol.28, No.5, pp.327-340, 1984
- [38] 荒木光彦, 桑田龍一, 藤中透, "むだ時間システムに対する PID/I-PD および離散型最適制御", システムと制御, Vol.28, No.5, pp.278-289, 1984
- [39] M. R. Mastaušek, A. D. Micić, "A Modified Smith Predictor for Controlling a Process with an Integrator and Long Dead-Time", IEEE trans. on Automatic Control, Vol.41, No.8, pp.1199-1203, 1996

- [40] K. J. Astrom, C. C. Hang, B. C. Lim, "A New Smith Predictor for Controlling a Process with an Integrator and Long Dead-Time", IEEE trans. on Automatic Control, Vol.39, No.2, pp.343-345,1994
- [41] M. R. Mastaušek, A. D. Micić, "On the Modified Smith Predictor for Controlling a Process with an Integrator and Long Dead-Time", IEEE trans. on Automatic Control, Vol44, No.8, pp.1603-1606, 1999
- [42] 野波健蔵, 田宏奇, "スライディングモード制御 - 非線形ロバスト制御の基礎理論 - ", コロナ社 (1994)
- [43] 浅野和俊, "電気-力学システムの基礎と応用, 養賢堂 (1994)
- [44] 藤井信生, "アナログ電子回路 - 集積回路化時代の - ", 昭晃堂 (1984)
- [45] 藤井信生, "デジタル電子回路 - 集積回路化時代の - ", 昭晃堂 (1987)
- [46] 伊東規之, "デジタル回路", 日本理工出版会 (1986)