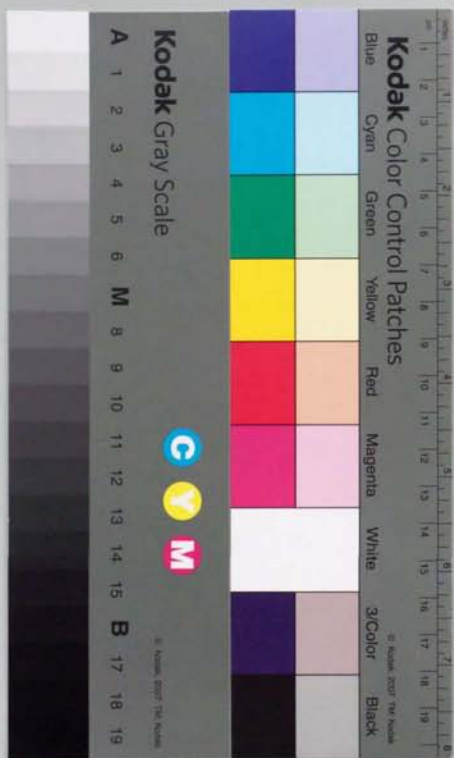


仮想的なインピーダンスを用いた
複数移動ロボット系の動作計画

太田 順



①

博士論文

仮想的なインピーダンスを用いた 複数移動ロボット系の動作計画

太田 順

論文の概要

本論文では、複数移動ロボット系の動作計画手法を提案する。

系を構成するロボットの台数が増加した場合の対応と、上位に存在する作業計画器との統合の観点から、各ロボットが自律的に意志決定でき、ロボットの群行動の動作計画が実現可能である複数移動ロボット系の動作計画手法を提案する。

最初に、分散管理型大域計画器 (Global Motion Planner, GMP) と分散管理型局所計画器 (Local Avoidance Planner, LAP) の階層構造を有する動作計画システムを構築する。ここで、GMPでは、既知動的障害物を回避する軌道を生成し、LAPでは未知動的障害物を回避する軌道を生成する。ロボットが多数である時には、GMPと比べてLAPの設計が重要となる。本論文では、LAPの手法として、人工ポテンシャル法を集中定数系、分散管理型に改良した『仮想インピーダンス法』を提案する。

次に『仮想インピーダンス法』の詳細設計を行う。目標地点以外で停止をする「停留」問題と、目標地点への移動が困難になる「妨害状況」問題について考察し、停留解決器と妨害解決器の設計を行う。また、最適化法とニューラルネットワークによる学習を組合せた手法により、各ロボットが環境の状況を実時間で把握し、それに適応したパラメータを設計する方法を提案する。

複数移動ロボット系における単体ロボットの動作計画をする際に、ロボットの密度が非常に高い環境下において、単純に『仮想インピーダンス法』を利用することは複雑した状況を招くことになり、好ましくない。本論文では、各ロボットに「自分に前方に存在して、自分と同じ方向に進んでいるロボットの後をついていく『追従戦略』」を『仮想インピーダンス法』に導入することにより、環境内のロボットの行動に自律的に秩序をもたらしことを示す。

複数移動ロボット系におけるロボット群の動作計画においては、群が実現すべき作業の性質に基づいた計画がなされるべきである。ここでは、動作計画時における二つの要求仕様である「危険の回避」、「群れの保持」の重要度の比により作業の性質が表現されるとして、この比を表現する指標として『結合度』という概念を提案する。この『結合度』の設計により、二つの要求仕様の重みづけがなされていることを示す。

以上提案した動作計画手法の有効性を実験による実験で検証する。ロボット同士の相対位置・姿勢計画システムとして、複数個の点光源から構成される看板を用いたシステムを新たに提案する。実験により2台のロボットの相互回避実験を行い、提案した計画手法の有効性を示す。

最後に、『仮想インピーダンス法』を定量的に評価する。具体的には、登尾の提案した手法 ([NOBORIO 92]) との比較を行う。計画軌道の性能、計算量、センサ能力の要請等について比較を行い、仮想インピーダンス法の特徴を明らかにする。

上記の議論により、本論文で提案する動作計画器、とりわけその中のLAPに相当する『仮想インピーダンス法』が、「自律的な意志決定機能・群行動の計画機能を有する複数移動ロボット系の動作計画手法の設計」という本論文の目的を満たす動作計画システムであることを示す。

『仮想的なインピーダンスを用いた複数移動ロボット系の動作計画』目次

1 章 序論	1
1. 1 研究の背景	2
1. 2 複数移動ロボット系の動作計画手法に関する従来の研究	5
1. 3 ロボット群の動作計画手法に関する従来の研究	14
1. 4 研究の目的	15
1. 5 本論文の構成	16
1. 6 1章のまとめ	18
2 章 動作計画システムの概念設計	19
2. 1 分散管理型階層構造動作計画システムの提案	22
2. 2 分散管理型大域計画器の概要	26
2. 3 分散管理型局所計画器の概要	30
2. 4 2章のまとめ	34
3 章 仮想的なインピーダンスを用いた分散管理型局所計画器の設計	35
3. 1 『仮想インピーダンス法』の定式化	38
3. 2 停留・妨害状況の定義とその対策	44
3. 3 環境に応じたパラメータ設計論	50
3. 4 動作計画シミュレーション	66
3. 5 3章のまとめ	85
4 章 複数移動ロボット系における単体ロボットの動作計画	87
4. 1 秩序生成の問題設定および『追従戦略』の提案	90
4. 2 追従戦略の『仮想インピーダンス法』への導入	92
4. 3 シミュレーションによる追従戦略の有効性の検証	96
4. 4 異なる系における追従戦略の拡張	102
4. 5 4章のまとめ	107
5 章 複数移動ロボット系におけるロボット群の動作計画	109
5. 1 ロボット群の概念の重要性と従来の研究	112
5. 2 動作計画手法の概念設計	113
5. 3 仮想インピーダンス法を用いたロボット群の動作計画の実現	117
5. 4 ロボット群の動作計画シミュレーション	122
5. 5 5章のまとめ	131
6 章 実験を用いた実験	133
6. 1 実験装置の概略	136
6. 2 実験各要素の構成	138
6. 3 看板による位置・姿勢計画システム	140
6. 4 コントローラの設計	147
6. 5 実験と考察	153
6. 6 6章のまとめ	162

7 章 評価	163
7. 1 従来研究との定量的比較	166
7. 2 本システム全体の評価	178
7. 3 7章のまとめ	180
8 章 結論と今後の展望	181
8. 1 結論	184
8. 2 今後の展望	186
8. 3 8章のまとめ	190
謝辞	191
参考文献	195
研究業績書	209

第1章 序論

1. 1 研究の背景
1. 2 複数移動ロボット系の動作計画手法に関する従来の研究
1. 3 ロボット群の動作計画手法に関する従来の研究
1. 4 研究の目的
1. 5 本論文の構成
1. 6 1章のまとめ

1. 1 研究の背景

移動ロボットに関する従来の研究の多くは、単体の移動ロボットを研究対象としたものであった。しかしながら、移動ロボットの使用される環境がオフィス、家庭へと広がり、移動ロボットに要求される作業の性能が高度になるにつれて、一つの環境内に存在する複数台のロボットが衝突回避や能動的な協調作業を実現する複数移動ロボット系の解析が重要となっている。一方、ハードウェア技術の進歩により、非常に微小なロボットの研究、マイクロ・ロボティクスの実現が可能となりつつある。この分野においては、ロボット単体の能力の限界を考慮すると、複数台ロボットの協調作業という方向に研究が進むものと考えられる。このため、複数ロボット系としての解析が不可欠である。

これらの背景の下、複数移動ロボット系の研究が盛んになりつつあり、群知能ロボット、マルチエージェントロボットシステムの研究が盛んに行われている（〔浅間 92 a〕〔佐藤 92〕〔長田 92〕〔浅間 92b〕〔油田 92〕〔坂根 93〕）。この系の重要な研究課題の一つに計画問題があるが、この問題は〔CALOUD 90〕によれば、以下の3つの段階に分割される。

- (a) 作業計画 (task planning) : 外部から与えられた目的をロボットが実現できる作業に変換する計画
- (b) 作業割当て (task allocation) : 生成された作業を実現するために、その作業を単体または複数台のロボットに配分する計画
- (c) 動作計画 (motion planning) : 作業実現のために、現在位置から目標位置まで移動するための動作を生成する計画。

計画問題は最終的には上記の三者を統合して解決されるべきである。しかしながら、現状では、それぞれの分野で独立して研究がなされているか、あるいは統合を目指した研究でも甚だ不十分な結果しか得られていないという状況となっている。本論文では、三者のうち最も低レベルに存在し、他の二つの計画の基本となる(c)の動作計画問題を研究の対象とする。これは、ボトムアップ的なアプローチにより統合計画システムの構築を目指すことを意味しているが、本論文では、上位の二つの計画は直接的には扱わない。なお、『動作計画』とはほぼ同様の意味の用語として『経路計画』という言葉が存在するが、本論文では、「対象とする環境が動的であり、計画時にはロボットが通過すべき経路のみならず、通過時における時間関数としての動作をも計画する必要がある」という観点から、以降『動作計画』という言葉を用いることとする。

本論文における問題意識を明らかにするために、複数移動ロボット環境として以下のものを想定する。

- ・数10台から数100台のロボットが混在している環境
- ・ロボット以外の障害物はあまり存在しない環境

上記の想定は、『個別のロボットの振舞が環境全体の挙動にどのような影響を及ぼ

すか、更に、そのように生成された環境状態を境界条件として個別のロボットの行動がどのような影響を受けるか』を明らかにするためのものである。これは、昨今盛んに行われている人工生命研究（〔星野 93〕〔米澤 93〕）の方法論、認識論の基本の運動である「創発 (emergence)」に対応している。

上記の想定は、多数の自律体が相互に干渉する環境を意味しているため、適用範囲は十分広いと考えられる。具体的な環境としては、複数台のマイクロロボット群による作業環境、駅のコンコース・廊下を走行する複数台のロボット、自律分散型生産システム等が挙げられる。本論文の2章以降で動作計画手法を提案するが、その具体的な適用の展望に関しては、8章で再度触れることとする。

上記で想定した環境における『複数移動ロボット系の動作計画器』に必要な機能として、本論文では、以下の二つが重要であると考えられる。

- (1) 各ロボットが自律的に意志決定できること。
- (2) ロボットの群行動の動作計画が実現できること（ロボット群の動作計画が実現できること）

(1)は、複数移動ロボット系における、分散管理型計画の必要性に関する議論である。本論文では、分散管理型計画(decentralized planning)を「複数移動ロボット系を構成する個々のロボットが独立して動作計画をとりおこなう方式」と定義し、それに対比する概念としての集中管理型計画(centralized planning)を「システム内すべての移動ロボットを統括する管理者が存在して、動作計画を推進する方式」と定義する。ここで、後者の計画には、

集中管理機構の計画時の負荷

集中管理機構の故障時の系全体の機能の劣化

の二つの問題点がある。これらの問題は、系内のロボットの台数が増加するほど顕著となる。また、人間社会においても、例えば、都会の雑踏中の人間の動作が、個々の自律的な意志により決定されている事実からも自律分散系の優位性が確かめられる。これより、複数移動ロボット系においては、各ロボットの自律的な意志決定による分散管理型決定が不可欠である。

集中管理機構を階層的に実現する方法も考慮可能である。この機構が適切に導入できれば、上記の2つの短所はある程度解消され、また実現すべき作業の性質が高度になればなるほど、完全な分散管理機構的な構造と比較した作業性能は向上すると考えられる。しかしながら、本論文では、

- ・適切な階層構造を実現するためには、階層の決定方法が大きな問題となる。例えば、環境が変動する場合に、階層構造も動的にならざるを得ない。これは解決が非常に難しい。
- ・ロボットによる作業実現の全体を枠組みを考えた場合、現在位置から目的地に向かう動作計画問題は、比較的下位のレベルに位置すると考えられる。ロボット間の通信機能は、通信によるトラフィックの影響を考えると、下位レベルでは、できるだけ利用しないことが望ましい。

という理由により、完全分散管理型計画の立場を採る。

(2)は、先の計画問題に関する議論の延長上にあるものである。前述したように、計画問題においては、作業の割当てが単体あるいは複数台のロボットに対して行われる。作業には様々なものが存在するが、その中で移動を伴うものは数多い。移動を伴う作業を割当てられた単体または複数台のロボットは、現在位置から作業の内容により決定される目標位置に移動する必要があるが、これは動作計画器において計画されるべきものである。上記の議論により、動作計画器においては、「単体のロボットが目標地点まで移動する動作計画を実現する機能」のみならず、

「複数台のロボットが群れを形成しつつ、目標地点まで移動する動作計画を実現する機能」

を有さなければならないことが明らかである。このような群行動の重要性は、

- ・危険を避けたり、餌を捕獲したり、体温を保持するための魚や鳥の集団、
- ・単独のエージェントでは実現困難あるいは不可能な作業の実現、例えば複数のエージェントによる大型対象物の協調搬送作業、分散センシング

等の作業において群れ行動が不可欠である（[日高 92][今福 92]）ことから正しいことがわかる。本論文における「ロボット群」と「ロボット群の動作計画」の定義を以下に示す。

「ロボット群」とは、共通の目的を有していることを相互に認識しているロボットの集りである。

「ロボット群の動作計画」とは、ロボット群が、相互に近接した群れの状態を保持したまま現在位置から目標位置に移動できるように、ロボット群を構成している個々のロボットが行う動作計画を意味する。

本論文におけるロボット群の動作計画は、群を統轄する集中管理機構が存在して、それが群を構成する各ロボットに動作指令を送るものではないことに注意されたい。あくまで個別のロボットがロボット群の存在を意識して計画を行った結果として、群れた状態が生じるものであり、(1)の条件に矛盾しない。また、本論文では、ロボット群は作業の内容に応じて動作計画の前段階で決定されるものとするため、ロボット群の形成過程に関しては論じない。

上記の議論により、複数移動ロボット系の動作計画には、二つの重要な機能が存在することがわかった。この機能実現の観点から従来の研究を概観する。複数移動ロボットの動作計画手法に関する従来の研究の概要を1.2節で、ロボット群の動作計画に関する従来の研究の概要を1.3節で述べる。

1.2 複数移動ロボット系の動作計画手法に関する従来の研究

1.2.1 従来の研究の分類

複数体の動作計画問題は当初ソファロボティクスとかビームボットといわれる物体のすりぬけ問題であったが（Fig. 1.1）、その後マニピュレータの動作計画を中心として、多数のアルゴリズムが提案された。ロボットが単体の場合の動作計画は衝突回避と最適パス問題との組合せであり、アルゴリズムは空間表現方法（作業空間、形態空間、グラフ）、時間軸の取扱、回避戦略などで分類された。複数ロボットの場合も基本的には単体ロボットの計画の積み重ねであるから、この分野での研究はそのまま複数ロボットの動作計画に適用可能である。しかし、複数のロボットが相互に動作するとすると最適化の要求は弱まるが、計画立案の分担、情報の共有が必要になる。よって、既存の動作計画の手法を論ずるには、アルゴリズムの性質を論ずる前に、計画の“枠組み”すなわちアルゴリズムの適用条件の違いを評価しなければならない。そこで、本論では“枠組み”の分類を行った上でアルゴリズムを列挙する。

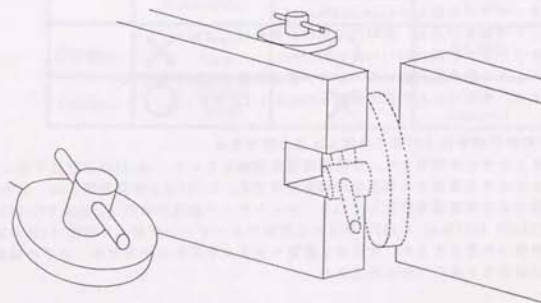


Fig. 1.1 An example of motion planning([SCHWARTZ 83a])

アルゴリズムの適用条件に関して、本論文では、以下の3つの観点から“枠組”を分類する。

- (a) Who? (誰が行うのか):
 (a-1) スーパーバイザー (集中管理型 [Centralized]),
 (a-2) 各ロボット (分散管理型 [Decentralized])
 (b) When? (いつ行うのか):
 (b-1) オフライン [Off-line],
 (b-2) オンライン [On-line]
 (c) Where? (どこで行うのか):
 (c-1) 道路網環境
 (c-2) 2次元平面
 (c-3) 3次元空間

(a)は計算機の構成方法を意味するのではなく、計画の主体の分類であり、情報の流れの違いである。1.1節で述べたように、

(a-1)集中管理型計画 (centralized planning):

「システム内にすべての移動ロボットを統轄する管理者が存在して、動作計画を推進する方式」

(a-2)分散管理型計画 (decentralized planning):

「複数移動ロボット系を構成する個々のロボットが独立して動作計画をとりこなす方式」

である。それぞれの手法の特徴をTable 1.1にまとめる。

(b)は動作計画の立案の時期の分類である。ここで、

(b-1)オンライン計画 (on-line planning):

「センサ情報を利用し、実時間で動作計画を行う方法」

(b-2)オフライン計画 (off-line planning):

「ロボットの動作実行前にあらかじめ動作計画を実施する方法」

と定義する。それぞれの手法の特徴をTable 1.2に示す。

(c)は動作計画を行う対象の環境による分類である。

(c-1)ネットワーク環境とは、道路網環境を意味している。(c-2)の2次元平面とは、工場やオフィスを構成する部屋の環境を意味する。(c-3)の3次元空間とは、水中や宇宙空間のような環境を想定している。ネットワーク環境の研究 ([CHEN 87][GROSSMAN 88][SAITO 91][WANG-J 91]) や3次元環境にターゲットを絞った研究 ([FUJIMURA 93b]) が幾つか存在するが、本論では移動ロボットを対象とするため、以下の議論においては環境を2次元平面に限定する。

Table 1.1 Comparison of centralized / decentralized planning

	Optimality	Comp. Load with Increase in Robot Number	Fault-tolerance
Centralized	○ Possible	✗ Comp. load ↑	✗ Centralized
Decentralized	✗ Difficult	○ Comp. load →	○ Decentralized

Table 1.2 Comparison of on-line / off-line planning

	Robustness	Computational Power	View
Off-line	✗ Open loop	○	Global
On-line	○ Closed loop	✗	Self-centered

(a), (b)は独立であるため、その組合せは4通り存在し、その混合形態も付加すると以下の様になる。

- (S1) 集中管理オフラインシステム
- (S2) 集中管理オンラインシステム
- (S3) 分散管理オフラインシステム
- (S4) 分散管理オンラインシステム
- (S5) 混合型システム

ここで、(S2)のシステムは多数のロボットの動作を実時間で計画することを意味し、現状の計算機の能力では難しいことと、中心となる計算機の信頼性に依存しすぎであることから、この範囲に入る研究は見られない。また(S3)のシステムは、各ロボットが独立して動作計画をするために、作業領域内の環境情報をすべて同等に保持するという冗長な方式となることから、研究がほとんどなされていない。(S5)の混合型とは(S1)~(S4)の組合せである。そこで、実際には3種類の“枠組み”でのアルゴリズムの比較検討を行った。

1. 2. 2 複数移動ロボット系の動作計画手法の具体的アルゴリズム

上述の分類に従い、アルゴリズムの比較検討を行う。

1. 2. 2. 1 集中管理オフラインシステム

この枠組みでの動作計画は移動ロボット系の動作計画研究の基本形であり、時間的制約もなく環境情報も十分存在すると仮定できるので、決定論的に計画するものが多い。よって、ここでの手法は他の枠組みにおいても適用可能であるから、分散管理処理や実時間性を重要視していない手法と考えることもできる。アルゴリズムを概観すれば、以下の3種の手法が多い。

- (1) cell decomposition method ([SCHWARTZ 83a], [SCHWARTZ 83b], [PARSONS 90])
- (2) retraction method ([O'DUNLAING 83], [BARRAQUAND 89])
- (3) ロボット間のpriorityを設定する方法 ([ERDMANN 86], [BUCKLEY 89], [WARREN 90])

空間をcellで分割する(1)の方法は問題を有限の空間に圧縮し、アルゴリズムを単純化できる長所を持つ。しかし、精度良く計画を行うには、非常に計算時間がかかるという欠点がある。そこで、(1)になんらかの改良を加えた方法や、volonoi graphを用いて次元を縮退させる(2)の方法や、優先度の設定により単体ロボットの計画問題に縮退させる(3)の方法が提案されている (Fig. 1.2)。[WARREN 90]では、プライオリティ設定と後述する人工ポテンシャル法を組合せて解を導出している (Fig. 1.3)。この分野の研究の調査は[SHARIR 90]に詳しい。また、[RIU 89]はクアドツリーとベトリネットを組合せることで、[SHIBATA 93] [柴田 93]ではグラフサーチと遺伝的アルゴリズムを組合せることで問題を解いている。[LEE 90]は、ニューラルネットワークのHopfieldモデルを用い、停留(deadlock)を起こした場合にその地点に仮想的な

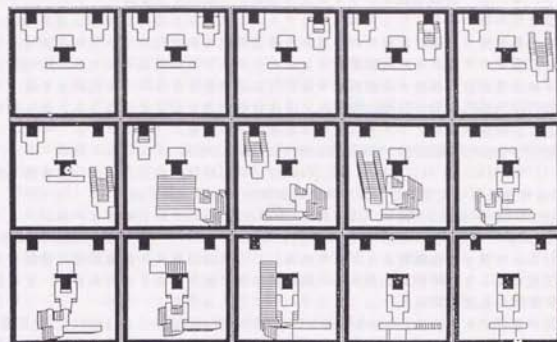


Fig. 1.2 An example of methods with priority ([ERDMANN 86])

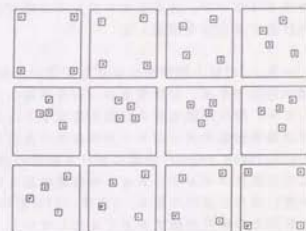
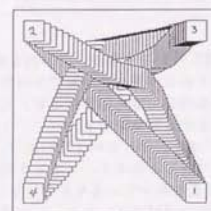


Fig. 1.3 An example of methods with priority and artificial potential field ([WARREN 90])

障害物を付加することにより、回避動作を実現している。

1. 2. 2. 2 分散管理オンラインシステム

分散管理処理の場合、たとえ採用したアルゴリズムが大域的な演算に適用可能だとしても、各ロボットの計画はそのロボット周辺の局所的動作についてのみ意味を持つことに注意すべきである。分散管理オンラインシステムの計画手法は多くの場合、相手ロボットを含む環境との衝突回避を局所的に効率良く行うことを目的とする。その結果生ずる大域的な動作計画の最適性や安定性は別途チェックされることが一般的である。この枠組みでのアルゴリズムの分類を以下に示す。

- (1) ポテンシャル場を用いた方法及びそれに類した方法
 ([TOURNASSOUD 86][WANG-P 88][WANG-P 89a][SAITO 89][斉藤 90a][斉藤 90b])
 - (2) ルールを用いた方法([TUIJNMAN 86][NOBORIO 92][KATO 92])
 - (3) ロボット間通信を用いた方法([PREMVUTI 90][ASAMA 91][NOREILS 91])
- (1)の方法は目標地点からの引力と障害物からの斥力がロボット自身に仮想的にかかるとして、ロボットを駆動する方法である。この手法の最大の問題点は、停留が本質的に存在することと解がポテンシャル場の張り方に依存することであるが、それらの欠点を考慮した改良型は多い。

人工ポテンシャル法は、マニピュレータの障害物回避法として[KHATIB 86]において提案されたものである。これは、作業空間内において、目標地点からの仮想引力と、障害物からの仮想斥力をロボット自身にかけることによりロボットの動作計画を実現するものである。[TOURNASSOUD 86]は、ロボット近傍にある障害物とロボットの両者と接する2直線の法線ベクトルから他のロボットと干渉しない領域を求めることにより、計画を実現している。[WANG-P 88][WANG-P 89a]は、軌道追従のためのスライディングモード制御と障害物回避のための障害物からの仮想斥力とを組合わせた簡単な制御則を用いて、動作計画を行っている。その際、パラメータ変動に対する詳細な解析を位相平面を用いて行っている。[SAITO 89][斉藤 90a][斉藤 90b]は、ポテンシャル法ではないが、近接したロボットの相対位置・速度から衝突危険性を判断して、動作計画する方法を提案した。

一方、(2)は人間社会における交通ルールを適用した方法であり、効率的な衝突回避が可能である。いかなるルールを準備し、かつそれらのルールが遵守されたと仮定しても、解決不能となる状況を生成することは理論的に可能と思われるが、この手法では衝突回避の大半がルールの存在で実現できるとする立場をとる。

[TUIJNMAN 86]は、海洋における船舶の回避ルールを適用している(Fig. 1.4)。この方法はロボットの大きさが作業空間と比較して著しく小さく、ロボットの操作性が悪い場合に適用できる。[喜多 76]は動作計画を直接扱っているものではないが、船舶同士の衝突危険度をモデル化している。[NOBORIO 92]はルールベースの動作計画を提案しているが、その際、停留を生じさせないための条件について論じている。[KATO 92]では、交通ルールを各ロボットに付加することにより、円滑な動作計画を実現している。また、複数移動ロボット系を対象としたものではないが、ファジィ制御

により障害物回避行動を実現する方法もいくつか提案されている([武野 86][前田 88][小山 91][石川 91])。これらの方法は、複数ロボット系にも容易に適用可能であるが、パラメータ調整法が問題となる。

(1)や(2)の手法の欠点として、衝突回避時に各ロボットは相互に相手の行動が予測可能と仮定していることが挙げられる。つまり、自分以外のロボットや環境が予測通りの動きを取らない場合を想定すれば、環境の測定頻度を高め、実時間処理を要求することとなる。これを解決する一手段として(3)は近接したロボット同士が通信を介して交渉することにより計画を行う方法であるが、ロボット間の相互通信の信頼性とチャネル容量に依存する。また、通信を実現する段階にも優先順位関係を確立する必要があることも注意しなければならない。

[PREMVUTI 90]は交差点等を複数ロボットの共有資源と考え、その資源を他のロボットになるべく譲ろうとする“遠慮的協調”の考えと、停留のときにはある1台のロボットが管理者となる集中管理的管理手法のもとで道路網中の三叉路における相互通り抜け動作を実現している。[NOREILS 91]は、ロボットに優先順位を持たせた計画を実現している。2台のロボットが干渉した際に、優先順位の低いロボットが優先順位の低いロボットへ指令(ある地点で待て等)を出すことにより衝突回避を実現している。[渡間 91][ASAMA 91]は移動ロボットの障害物回避手法をTable 1.3の階層で表現し、階層の低い方から順番に回避動作を実現すべきであると主張している。なお、通信すべき情報についての研究はMMS([ISO 90])のような汎用的な定義を設けず、場当たり的に提案されているのが現状である。

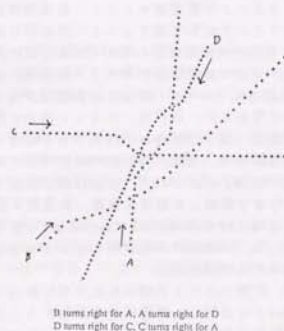


Fig. 1.4 Rule-based planning method([TUIJNMAN 86])

Table 1.3 Levels for collision avoidance problem ([渡間 91])

Difficulty of Situation	Method for Path Planning	Path Planner for Collision Avoidance	Means
High ↑ ↓ Low	Planning by A Human Operator	Human	Communication
	Planning by A High-Level Deadlock Solver	Deadlock solver	
	Planning by A Low-level Deadlock Solver	Deadlock solver	
	Path Planning by A Robot Leader	Mobile Robot	
	Giving Way Based on Priority of Tasks	Mobile Robot	
	Collision Avoidance Based on Mobile Rules	Mobile Robot	Sensing
	Static Path Planning	Mobile Robot	

1. 2. 2. 3 混合型システム

Table 1.1, Table 1.2に示したそれぞれの短所を補うように、上述の2手法を混合させた計画手法が提案されている。混合形態としては、

- (M1)集中管理オフライン+分散管理オンライン混合型計画([KROGH 86][KANT 88])
 - (M2)集中管理オンライン+分散管理オンライン混合型計画([劉 92][NOBORIO 93])
 - (M3)分散管理オフライン+分散管理オンライン混合型計画([YEUNG 87][ARAI 89])
 - (M4)分散管理オンライン+分散管理オンライン混合型計画([BROOKS 86][ARKIN 93])
- の4通りが考えられている。ロボットの数が多い場合、高い作業効率を要求する際には(M1)の方法が望ましい。(M2)は、ロボット同士に問題が生じた時にスーパーバイザが介入する方式で、生じた問題が小規模である場合には、有効な方法である。(M3)の研究は、分散問題解決、分散人工知能の研究分野と関連し、興味深い分野である。ロボットの数が増加するにつれて、完全自律分散型であるこの方式が有利となる。(M4)の方法は、今まで解説した方法とは違った概念に基づいた手法である。この方法は、ロボットの状態に対するふるまい(behavior)を階層的に構成することで、そのふるまいの集積として、有効な作業を実現しようとする立場であり、作業計画と動作計画とを統合的に扱える利点がある。

この分類の手法は、計画システムの構成を大域的計画(global planning)と局所的計画(local planning)とに分割し、階層構成を採用することが一般的である。

[KROGH 86]は、大域的計画では経路緩和法(path relaxation)を用いて経路を生成し、局所的計画では一般化ポテンシャル場(generalized potential field)法を用い

て、実時間の動作を実現している。一般化ポテンシャル場とはポテンシャルが位置のみの関数でなく速度の関数でもある場である。[KANT 88]は2ステップからなる大域的計画を提案し、第1ステップはV-graphを用いてロボット相互の干渉を考慮しない軌道計画を行い、第2ステップでは相互に衝突を起こさないような速度の設計を行い、局所的計画ではポテンシャル法を用いたオンライン制御を行っている。

[劉 92]、[NOBORIO 93]では、通常は各ロボットは分散的に動作し、ロボット同士が干渉しそうになった時に、スーパーバイザが各ロボットに指令を出すことにより問題を解決を図っている。この方法は、環境が複雑になったりロボットの台数が増えた場合などの実現可能性に問題があると考えられる。

[YEUNG 87]も二層構造を採用し、大域的計画ではロボット相互の衝突を考慮しないロボット単体の経路計画をV-graph法に似たより簡単な方法で行い、局所的計画はその時のロボットの状況に応じて実時間で優先順位を設定し、優先順位の低いロボットが回避行動を行うことにより、経路生成を可能にしている。[ARAI 89]は未知の障害物をも環境内に含んだ複数のロボットの動作計画を二層構造で実行することを提案している。この研究に関しては、以降の節で詳しく述べる。

[BROOKS 86]では、包摂アーキテクチャを提唱して、環境に関するモデルを必要としないロボットアーキテクチャを提案している。また、[ARKIN 93]では、スキーマ理論を用いて反射行動を組合せるアーキテクチャを提案している。これらの手法は、動作計画能力そのものはそれほど高くはないものの、比較的低レベルの作業を非常にエレガントに設計できる可能性がある。しかしながら実現すべき作業が高度になるにつれて、その作業の内容を明示的に表現することが困難になり計画器全体をこの形式で表現することには無理があると考えられる。ただし、この概念を計画器の一部にとり入れることは非常に有意義であるため、多数の研究者が、この概念に影響を受け、様々なアーキテクチャを提案している。

本節では複数移動ロボット系の動作計画に関する研究を概観し、いくつかの研究を紹介した。本節での分類は、複数ロボット間の主従関係と処理の実時間性の反映であり、妥当な分類基準を示したと考えられる。また、以下に示す知見が得られた。

- ・現状の研究はきわめて単純なモデルやルールを単独で適用したシミュレーションにとどまっているものが多い。また、動作計画そのものにターゲットが絞られているものがほとんどで、より上位に存在する作業計画との統合性についてはBrooksやArkinの研究を除けば、ほとんど論じられていない。
- ・実際の移動ロボット動作計画としては階層構造が好ましい。それぞれの階層の実現方法には、様々なものが考えられるが、これと言った決定打となるような方式は存在せず、系や環境の特性、作業の種類に依存している。

1. 3 ロボット群の動作計画手法に関する従来の研究

ロボット群の動作計画に関する研究の数はそれほど多くない。[WANG-P 89b]はロボット群の動作計画について論じており、基本的にはリーダーとフォロウの制御となるが、その選び方として、(1)リーダーが1台、他がフォロウ、(2)リーダーがn台、他がフォロウ、(3)リーダーが仮想ロボット、の3通りについて検討を行っている。[NO REILS 92]は、ロボット間の協調を実現できる、ベトリネットを用いたアーキテクチャを提案し、簡単な群行動を実現した。[TSUMURA 92]は、群行動をリーダー・フォロウ型制御で実現する際にロボット間通信を用いる方法と用いない方法を提案した。[PARKER 93]は、群を構成するロボットが有する情報の種類と群行動との関係について論じた。[KUROSU 93]は、群行動をリーダー・フォロウ型制御で実現する際に、フェーザー制御を導入した方法を提案した。

上記の概観から従来の研究は、以下に示す特徴を有することがわかった。

- ・前節の動作計画手法とは分けて論じられているものがほとんどである。
- ・群内部におけるロボット間の関係を取扱ったもの、すなわちリーダーの選択方法、群内のロボット間の通信方式に関して論じたものがほとんどである。
- ・群と群が実現する作業との関連、群と環境との関係については、ほとんど論じられていない。

1. 4 研究の目的

1. 1節で、複数移動ロボット系の動作計画器の要求仕様について以下の様に言及した。

- (1)各ロボットが自律的に意志決定できること
- (2)ロボットの群行動の動作計画が実現できること

このような観点から従来の研究をとらえると、それらはすべて

- (1)すべての移動ロボットの統括管理者が存在する『集中管理型計画』
- (2)単一ロボットが環境のなかを移動し、自己以外のすべての物体を未知の移動障害物とみなす『未知動的環境の動作計画』

のいずれかの特徴を有しており、以下の問題点が指摘できる。

(1)『集中管理型計画』は、ロボットの台数が少ない場合や計画の時間的制約が緩い場合には有効であるが、数十～数百台のロボットからなる大規模システムやリアルタイム性を重視するシステムでは実現が困難となる。

(2)複数移動ロボット系においては、他物体と衝突回避するのみならず、ロボット同士が能動的に協調して群れを構成しつつ作業を実行する必要があるが、従来の枠組みでは協調の概念を表現する事が難しい。

そこで、本論文では、

- (1)各ロボットが自律的に意志決定できること
 - (2)ロボットの群行動の動作計画が実現できることの要求仕様を満たす複数移動ロボット系の動作計画手法の設計

を、研究の目的とする。

具体的には、以下の方針で研究を進める。

(1)の対策として、個々のロボットが独立して動作計画をとりおこなう『分散管理型計画』を採用する。

(2)の対策として、群形態をも統一的に表現できる手法である『仮想インピーダンス法』を導入する。

上記の対策による本論文の動作計画器の概念設計を2章で行う。

1. 5 本論文の構成

本論文の構成を以下に示す。

まず、第1章で、研究の背景、従来研究の概観を述べ、『自律分散型であり、群行動を表現できる複数移動ロボット系の動作計画手法の構築』という研究の目的を設定した。

第2章で、本論文で提案する動作計画器の概要を説明する。提案する計画器は、上位階層と下位階層から構成されるが、両階層間の関係、上位階層、下位階層の設計論に関する議論を行う。

第3章では、本論文で提案する動作計画器の下位階層の説明を行う。具体的には、仮想的なインピーダンスを用いた方法を適用するが、その際に生じる停留問題とその対策について述べる。また、パラメータの設計法について考察する。

第4章では、複数移動ロボット系の単体ロボットの動作計画手法を提案する。この時の問題として、環境内のロボットの台数が増した時に秩序が自律的に生成される『追従戦略』を提案し、その有効性をシミュレーションにより明らかにする。

第5章では、複数移動ロボット系のロボット群の動作計画手法を提案する。この際、群の作業の特性を表現する『結合度』という概念を提案して、それに応じた動作計画手法を提案する。シミュレーションにより提案した手法の有効性を明らかにする。

第6章では、試作装置を用いた実験を行う。その際、ロボット間の相対位置・姿勢を計測することが問題となるため、看板を用いた計測システムを提案し、実ロボットの動作計画を実現する。

第7章では、評価を行う。ルールベースの他の手法との定量的な比較を行うことにより、本論文で提案した動作計画器の有効性を明らかにする。その他の評価項目について議論を行う。

第8章では、結論と展望を述べる。展望では、動作計画と作業計画との統合について考察を加える。

Fig. 1.5に論文全体の構成図を示す。

Chapter
number

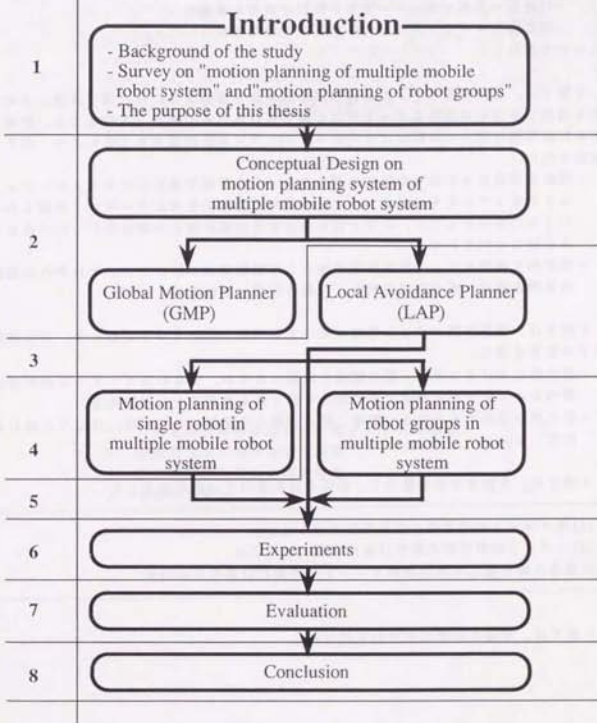


Fig. 1.5 Schematic view of this thesis

1. 6 1章のまとめ

本章では、序論を述べた。

1節で、研究の背景として、

- ・複数移動ロボット系の概念の重要性
- ・計画器の基本モジュールとなる動作計画器の重要性
- ・複数移動ロボット系の動作計画器に必要な機能

について言及した。

2節では、複数移動ロボット系の動作計画手法を概観した。各計画手法は、その手法を適用するための条件によって分けて論じられるべきである。本論文では、計画主体と計画時期に関して分類し、その中でアルゴリズムを評価することにより、以下の知見を得た。

- ・現状の研究はきわめて単純なモデルやルールを単独で適用したシミュレーションにとどまっているものが多い。また、動作計画そのものにターゲットが絞られているものがほとんどで、より上位に存在する作業計画との統合性についてはほとんど論じられていない。
- ・実際の移動ロボット動作計画手法として階層構造が好ましい。それぞれの階層の実現方法は、系や環境の特性、作業の種類に依存する。

3節では、複数移動ロボット系のロボット群の動作計画手法を概観した。その結果以下の知見を得た。

- ・群内部におけるロボット間の関係を取扱ったもの、すなわちリーダーの選択方法、群内のロボット間の通信方式に関して論じたものがほとんどである。
- ・群と群が実現する作業との関連、群と環境との関係については、ほとんど論じられていない。

4節では、3節までの議論の下、研究の目的を以下の様に設定した。

- (1)各ロボットが自律的に意志決定できること。
 - (2)ロボットの群行動の動作計画が実現できることの要求仕様を満たす複数移動ロボット系の動作計画手法の設計

5節では、本論文の構成について述べた。

第2章 動作計画システムの概念設計

- 2. 1 分散管理型階層構造動作計画システムの提案
- 2. 2 分散管理型大域計画器の概要
- 2. 3 分散管理型局所計画器の概要
- 2. 4 2章のまとめ

本章では、複数移動ロボット系の動作計画器を提案し、概要を述べる。

2. 1 節で、従来手法の問題点に基づいて、分散型で階層型の複数移動ロボット系動作計画システムを提案する。提案するシステムは、分散型大域計画器と分散型局所計画器の二つの階層から構成される。それぞれの計画器の機能分担、利用する情報について考察する。

2. 2 節で、上位階層の分散型大域計画器について論じる。まず、大域計画器に必要な機能に関して考察する。それらの機能を満たす従来の動作計画手法のいくつかを紹介する。

2. 3 節で、下位階層の分散型局所計画器を提案する。具体的には、互に近傍に存在するロボット、ロボットの目標位置、障害物間に仮想的なバネ・ダンパを張ることにより動作計画を実現する『仮想インピーダンス法』を提案する。

2. 4 節で本章のまとめを行う。

Fig. 2.1 に論文全体における本章の位置付けを示す。

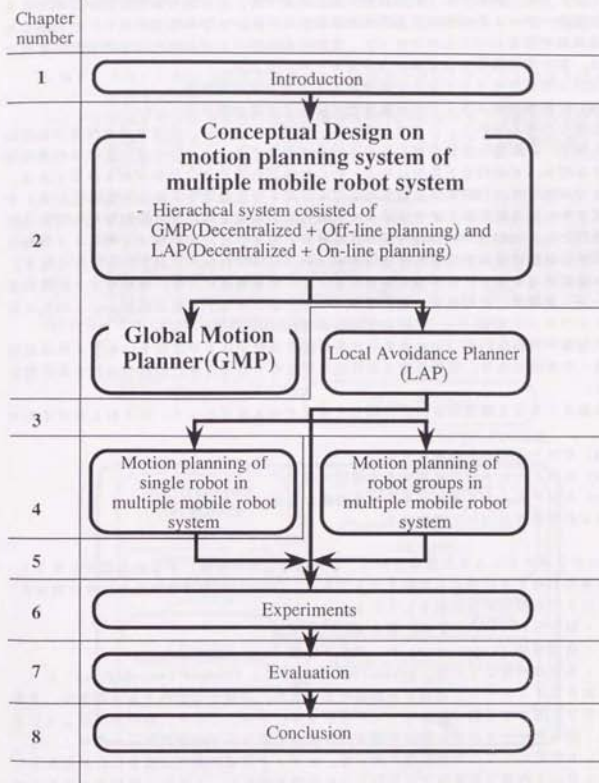


Fig. 2.1 The structure of chapter 2

2. 1 分散管理型階層構造動作計画システムの提案

本論文では、各ロボットが自律的に意思決定でき、群行動の動作計画が実現できる複数移動ロボット系の動作計画手法を設計する。1. 2節の知見により、計画器には階層構造が望ましいことがわかった。また、各ロボットが自律的に意思決定するためには、集中管理機構を含めない方が望ましい。そこで、

(M3)分散管理オフライン+分散管理オンライン混合型計画

(M4)分散管理オンライン+分散管理オンライン混合型計画

の2通りが考えられる。ここで、(M4)は、前述したように、実現すべき作業が高度になるほど、計画器の設計が難しくなる箇所が存在する。この方法は、高度な作業を実現する際に、その内容を隔ではなく、陰に計画器に埋め込む必要があるからである。そこで本論では、(M3)の『分散管理オフライン+分散管理オンライン混合型計画』を採用する。具体的には、オフラインで大域的地図情報を利用する分散管理大域移動計画器 Global Motion Planner (以下、GMPと略す)と、オンラインでセンサ情報を利用する分散管理局所回避計画器 Local Avoidance Planner (以下、LAPと略す)との階層構造をなす計画手法を採用する。この階層構造により、オフライン計画の長所——最適性、地図情報の利用等——と、オンライン計画の長所——外乱に対するロバストネス等——の両者を兼ね備えた計画が可能となる。

両階層の関係を図2.2に示す。なお、個々のロボットが遂行すべきタスクは目標位置への移動であり、目標位置は本研究の対象外である作業計画器が決定すると仮定する。

本論文における階層構造設計に関して考慮すべき項目として、以下の3点が存在する。

- GMPとLAPの機能分担
- GMPとLAPで利用できる情報の種類
- GMPからLAPに伝えるべき情報の種類

それぞれの項目について考察する。

(a)のGMPとLAPの機能分担は、自分自身以外の物体、すなわち他のロボットや障害物が有する性質から考察すべきである。[ARAI 89]では自分自身以外の物体を以下の3つの項目から分類をしている。

- ・動かない (static) か、動く (mobile) か
- ・通信可能 (negotiable) か、通信不可能 (non-negotiable) か
- ・動作が決定している (scheduled) か、決定していない (unscheduled) か

GMPはオフラインで駆動する前提があるため、取扱うことのできる物体は、3番目の項目に関して『動作が決定している』ものである。ここで、動作が決定しているとは、他のロボットがその物体の将来的な動作をきちんと把握できるか否かという、ロボット外部から見た判断基準であって、ロボット自身が動作計画をしたか否かという、ロボット内部での基準ではないことに注意されたい。これは、既知であるか未知であるか、ロボットの保持する地図に記載されているか否かという分類と等価である。

最初の項目の『動かない』か『動く』かに関しては、本論文では、より実際的な環境を想定するため、『動く』物体を対象とする。2番目の『通信可能/不可能』は、通信機能自体が通常オンラインで用いられるものであるため、その可否に関しては、2.3節のLAPの設計論の所で議論をする。すなわち、GMPにおいては『通信不可能』であるものを想定する。以上の議論の結果、

GMP: 『動く』かつ『通信不可能』かつ『動作が決定している』物体を回避する軌道を生成する。

LAP: GMPで対応できない問題の解決を図る。すなわち、『動作の決定していない』物体あるいは、『動作が決定している』が予定した動作通りに動いていない物体を回避する。

という役割分担が適切であるという結論が得られる。分散管理下においては、自分以外のロボットの動作は未知であるため、ロボット同士の相互回避は、すべてLAPで行う。このため、環境内のロボットの密度が増加するにつれて、LAPの負担が大きくなる方法であると言える。換言すれば、未知動的環境下では、実時間の計画、すなわちLAPが非常に重要になるということになる。

(b)のGMPとLAPで利用できる情報の種類について以下に述べる。

GMPでは、オフラインでの情報を取扱う。具体的には、以下のようなものが考えられる。

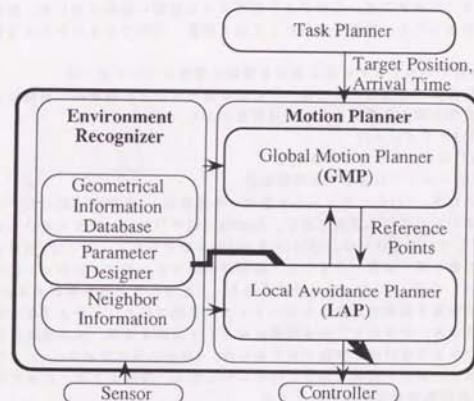


Fig. 2.2 System architecture of the proposed method

- ・大域的地図情報（既知動的障害物の移動軌道を含む）
- ・既知障害物の性質
- ・動作に関する要求仕様

1 番目の地図情報は、ある特定の時間における障害物の位置を記載したものである。
2 番目の既知障害物の性質とは、(1)その障害物にどの程度近づけるかの知識、(2)障害物の性質に関する知識等を意味する。(1)は、例えば、非常に高温の障害物へ近づくことのできる限界の距離に関する知識、非常に軽量の障害物を移動させられるかどうかの知識を意味する。(2)は、自動搬送車のように軌道が時間関数として完全に記述できる障害物か、それとも、自動ドアのようにある状態が生じた時に動作が開始する障害物かの知識である。また、交通ルールのようなもの、例えば「道の右側を歩くべきである」のような知識もGMPに反映させる必要がある。

3 番目の動作に関する要求仕様とは、できるだけ安全な軌道を通りたい、できるだけ早く目的地に到着したい、できるだけエネルギーを少ない軌道を生成したい等の要求を意味する。生成される軌道は要求に応じて異なったものになるはずである。また、この要求は、局所計画器へも反映されるべきである。

LAPでは、オフラインでの情報に加えて、オンラインで得られる情報を取扱う。オンラインで得られる情報とは

- ・ロボット近傍のセンサ情報
- ・他のロボットと通信することにより得られる情報

を意味する。LAPでは、GMPより利用できる情報の種類は多いが、計画を迅速にする必要があるため、情報を如何にして速く解釈、利用できるかが大きな問題となる。

(c)のGMPからLAPに伝えるべき情報の種類について述べる。

第一に伝えるべき情報としては、「目標位置または、目標軌跡、目標軌道」が考えられる。従来研究から考察すると、情報量の少ない順に、

- (1)サブゴール列
- (2)ロボットが辿るべき軌跡
- (3)ロボットが辿るべき時間軌道

等が考えられる。(1)のサブゴールを選択する方法は、[久保田 89]において静的環境に対して用いられている方法であり、[KROGH 86]や[YEUNG 87]でも用いられている。(2)を採用している方法には、[NOREILS 92]等が挙げられる。ここで、(1)と(2)はサブゴールの数の差と解釈できるため、厳密に分離できるものではない。(3)を用いている研究としては、[KANT 86]が挙げられる。上記のうちどれを選択するかは、ロボットの記憶容量と軌道の特性のトレードオフの問題であり、どのような作業を想定するかに依存する。すなわち、ある場所はゆっくり通過するが、ある場所は早々に過ぎ去る、というような計画を表現するためには、(3)が不可欠である。(3)は(1)や(2)を包含しており、より一般的であると考えられるため、本論文では、GMPからLAPへは(3)の時間軌道を与えるものとする。

すなわち、本論文では、GMPが「既知障害物を回避する時間関数としての軌道」

を生成するものとして議論を進める。

以下に計画の手順を述べる。

- (1)作業計画器からロボットの目標位置ならびに目標到達時間がGMPに送られる。
- (2)GMPは個々のロボットが保有する環境情報に基づき、既知障害物を回避する軌道計画をオフラインで立案する。
- (3)GMPが立案した計画軌道は、ロボットが走行中、次の時刻における目標位置として、随時LAPに渡される。
- (4)LAPは、この目標値とセンサからの情報を用いて、ロボットや障害物を回避しつつ、特定のロボットとは群れを構成する制御出力をオンラインで算出し、コントローラに出力する。

先にも議論したように、本計画器はLAPが中心であると考えて差し支えない。そこで、本論文では、GMPで何等かの方法を用いて、『既知障害物を回避する時間関数としての軌道』が生成された前提のもとでLAPの設計論を中心に議論を進めることとする。GMPについては、次節において、要求される仕様とその仕様に沿って適用可能ないくつかの従来の研究の概要を述べるにとどめ、それ以上の議論を行わない。

2. 2 分散管理型大域計画器の概要

前期での議論により、分散管理大域計画器では、

- ・大域的地図情報（既知動的障害物の移動軌道を含む）
- ・既知障害物の性質
- ・動作に関する要求仕様

の3種類の情報を有効に利用して計画を立案する必要があることがわかった。従来の研究において、それらを満たすものは少ない。本節では、いくつかの研究例を紹介し、考察する。

(a)[鈴木 90]では、時間毎の障害物の位置が完全に既知である環境下で、2次元平面を時間毎に重ね合わせた3次元空間における探索問題を解いている。この際探索領域を、再帰的に分割して解く方法を用いている。分割は、探索領域内の経路探索が陽に実現可能となるまで行われる。また、ある領域における移動経路が存在するか否かをベイント手続きを用いて調べている。中間点の選びかたを変えることにより、安全性、直線性、最短距離等の要求仕様に応じた経路計画が実現できる (Fig. 2.3)。

(b)[FUJIMURA 93a]においても、時間毎の障害物の位置が完全に既知である環境を対象としている。ある時間においてある領域を専有しているかを示すTransient Pixelを用いて環境を表現し、問題を定式化している。この論文では、要求仕様として、最小時間と最短経路の2つを考慮し、それぞれの要求仕様を満たす経路をfast-to-short path listというリスト (Fig. 2.4) に並べたことを提案し、そのリストの作成方法を述べている。

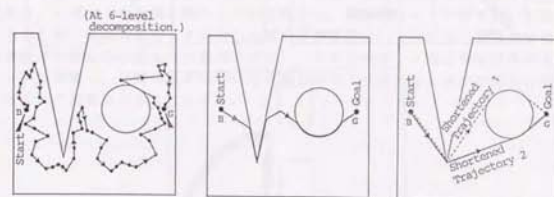
(c)[小方 93]では、既知動的障害物の概念を、位置が完全に既知ではないがある状態をトリガーとして、ボタン化された動作を開始する障害物にまで拡張した。これは、自動ドアのように、その挙動が時間関数として完全には記述できないが、目の前を踏むことにより動作が開始する障害物を意味する。この研究では、

(1) グラフサーチ等で、静止障害物を回避するサブゴールの点列を生成する。

(2) サブゴール間の経路をスプライン関数で補間し、関数のパラメータを、障害物との距離、経路長、到達時間等の線形和で表現された評価関数を最適化するようによりチューニングする。

という方法を採用している。これにより、自動ドアのような障害物を回避でき (Fig. 2.5)、多様な要求仕様に対応した経路 (Fig. 2.6) が生成できる。

以上、3種類の研究を概観した。(a)、(b)、(c)それぞれの方法に長所と短所が存在し、それぞれが取り扱っている内容も微妙に差があることがわかる。すなわち、(a)はより広い環境に、(c)は、より狭い領域に、それぞれ適している。(b)の研究は、特定の2つの要求仕様の最適性に関するトレードオフの表現形式に力点が置かれている。著者の意見では、これらを統合した計画が実務的である。例えば、(a)の方法を用いて問題をいくつかに細分化し、それぞれを(c)の方法を用いて解く。(b)のリストを



(a)Center line (b)Straight line (c)Minimum distance line
Fig. 2.3 Some cases in selection of middle points([鈴木 90])

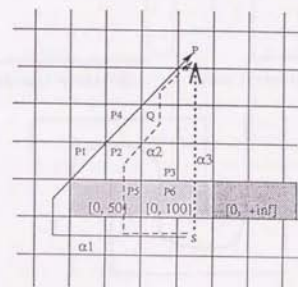


Fig. 2.4 Path through transient pixels([FUJIMURA 93a])

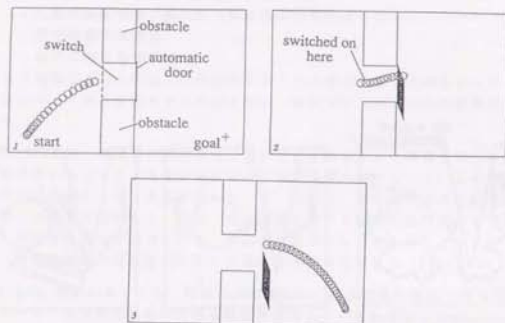


Fig. 2.5 Trajectory avoiding an automatic door([小方 93])

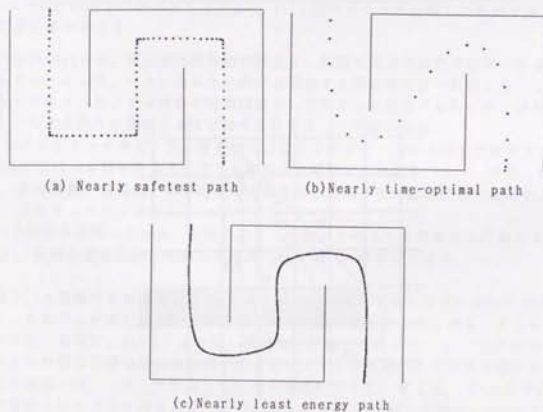


Fig. 2.6 Trajectory with various demands([小方 93])

用いることにより、(c)で得られた最適指標の全体としての整合をとる』等が考えられる。

動作計画において、GMPとLAPのどちらの比重が高いかは、与えられた環境に大きく依存する。例えば、環境内の障害物が非常に複雑な形状をしており、かつロボット同士の相互干渉が比較的少ない系に対しては、このGMPがより重要となると考えられる。本論文では、1章でも述べたように非常に多数のロボットの計画をターゲットとしている。前節でも述べたように、このような系におけるロボット同士の相互作用は、ロボットが実際に動作している最中に、局所的なセンサ等を用いて知るものであるため、必然的にLAPが中心と成らざるを得ない。そこで、本論文では、既知障害物が少数しか存在しない環境下での、LAPとロボット同士の相互作用との関係について考察し、以降GMPについては論じない。GMPとLAPの相互作用に関しては3章で再度触れることとする。

2. 3 分散管理型局所計画器の概要

2. 3. 1 分散管理型局所計画器の概念設計

先に述べたようにLAPは分散管理のオンライン動作計画である。1. 2節で述べたように、従来からこの分野で行われてきた研究は次の3種類の基本論理を用いたものが多い。

- (a) ルールを利用した方法 (TUIJMAN 86他)
- (b) ロボット間通信による方法 (PREMVTI 90他)
- (c) 人工的なポテンシャル場を用いた方法 (KHATIB 86他)

上記のうち、(b)の分類は、厳密に言えば(a)、(c)と同列ではなく、一段上位のレベルに位置するものである。しかしながら、(b)を利用した従来の研究は、(a)や(c)とは独立して取扱われているものがほとんどであるため分離して考えることが可能である。そのため、ここでは便宜的に(a)、(c)と同列として扱う。

上記の方法を、多数台の移動ロボットシステムへの適用可能性の観点から評価する。(a)は交通ルールを適用した方法であるが、環境が複雑になるにつれて、ルールの数が増加し、構造も複雑化する危険性が伴う。

(b)はロボット同士が通信を介して交渉して計画を立てる方法であるが、ロボット台数の増加につれてロボット間通信の信頼性とチャネル容量との問題点が顕著になる短所が存在する。動作計画は、実時間性が非常に要求されるため、通信に要する負荷は無視できない。また、人間社会において、人間同士が互にすれちがう場合を想定しても、会話を交わしながら衝突回避行動をすることが非常に稀であると考えられる。通信は、より上位レベルの作業計画等で充分に用いられるべきものである。

(c)は目標地点からの引力と障害物からの斥力がロボット自身に仮想的に掛かるとしてロボットを駆動する方法である。この手法はロボットの台数が増加しても、ロボット周囲の局所的な力に限定すれば計算時間の爆発的増加は起こらない。

以上の議論より、(c)の方法を基本的枠組みとする戦略をLAPに採用する。

要求仕様として以下の3点が挙げられる。

(7) 群れの概念の表現可能性: 1章で述べたように、複数移動ロボット系においては、衝突回避だけでなく群行動が表現可能なことが不可欠である。

(8) 単純なアルゴリズム: 従来のポテンシャル法を動的環境下のオンライン計画に適用することは計算時間の観点から不可能である。そこで、ポテンシャル場のモデルおよび制御則をできるだけ単純化する必要がある。

(9) 制御パラメータの自律的なチューニング: 分散管理で自律的な計画をするには、各ロボットは自らの制御パラメータを自らの手で決定できる必要がある。

(7)と(9)の仕様は、本論文の目的と同一のものである。(8)は、LAPに対する基本的要求仕様である。

2. 3. 2 『仮想インピーダンス法』の概要

リアル・タイム性に優れ、衝突回避と群形成の統一的表现が可能手法として、Fig. 2.7の概念図に示す『仮想インピーダンス法』を提案する (ARAI 89 [新井 93])。この手法の基本原理解は、

仮想的なマスを表現されたロボット(robot)は、ある時刻におけるロボットの目標位置(reference point)、近傍の障害物(obstacle)、近傍の他のロボット(other robots)から、仮想バネ・ダンパで表現された『仮想的なインピーダンス』により仮想外力を受けて運動をする。その運動を実現するべく、実際のロボットに動作指令を与える。

により計画を実現するものである。ここで、ロボットの目標位置はそのロボットの最終的なゴールに固定しているのではなく、スタートからゴールに向かって時々刻々移動していることに注意されたい。本論文のインピーダンスの定義については、3章で考察する。また、仮想ダンパが含まれている系は厳密な意味でのポテンシャル場ではない。しかしながら、例えば[KROGH 86]で提案されている手法は、相対位置のみならず相対速度に依存したポテンシャルを構成する方法であるが、ポテンシャルの概念を拡張して一般化ポテンシャル法 (generalized potential field method) と命名されている。そこで、本論文においても、[KROGH 86]の立場を踏襲して、仮想ダンパを含む本手法もポテンシャル法の一環であるとみなすこととする。

前項で提示した3つの要求仕様の観点から、『仮想インピーダンス法』を考察する。(7)群れの概念の表現可能性: 『仮想インピーダンス法』においては、群れの関係にあるロボット間では仮想バネの自然長以上になった時に引力が付加されるモデル (固定端のバネ、Fig. 2.8) を採用することにより、ロボットは相対位置を保ち、群形成の概念を表現する事が可能となる。例えば、共通の対象物を協調搬送する2台のロボット間に固定端のバネモデルを採用すると、2台のロボットは互に相対位置を保持しつつ移動するため、搬送作業を実現しやすくなる。具体的な適用例については、4章と5章で述べる。

(8)単純なアルゴリズム: 仮想インピーダンス法は人工ポテンシャル場が発生する力を集中定数系のバネに凝集させ、加えてダンパを付加した方法である。また、仮想バネ・ダンパは、近傍の障害物、ロボットと張られるモデルである。この特徴は、分布定数系の電荷モデルで大域的な領域に対して表現される一般的なポテンシャル法と比較して、計算量が低減される長所を有する。これにより、本手法は、オンライン計画に適用することが可能であると考えられる。次の3章で計算量に関する評価を行う。

(9)制御パラメータの自律的なチューニング: 『仮想インピーダンス法』は、各ロボットと他者との間に構成される仮想的なバネ・ダンパのみから実現できる。ここで、仮想バネ定数、ダンパ定数等の設計パラメータは、必ずしも他のロボットのパラメータと一致させる必要がないため、自律的なパラメータ設計が可能である。自律的なパラメータチューニングの方法についても、3章で述べる。

上記の議論より、『仮想インピーダンス法』が先の要求仕様を満たす手法であることがわかった。最後に、動作計画システム全体における仮想インピーダンス法の位置付けを考える。以下に本論文におけるアプローチを示す。

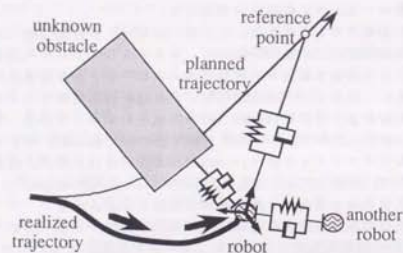


Fig. 2.7 Schematic view of Virtual Impedance Method

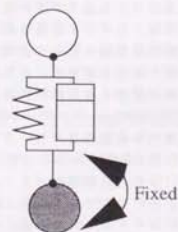


Fig. 2.8 Virtual impedance model between two robots of the same group

- ・計画問題と制御問題を分離するために、車輪機構は動作計画器ではなく、コントローラで考慮する。
- ・計画器における一番下位レベルの動作を、仮想バネ・マス・ダンパモデルで統一的に表現する。計画時における各ロボットの戦略を仮想バネ・ダンパモデルの操作により実現する。

上記の項目の実現形態をFig. 2.9に示す。『仮想インピーダンス法』の具体的な定式化、停留問題の回避、パラメータ設計法については3章で詳説する。具体的な戦略の付与方法を4、5章で論じ、コントローラの設計を6章で述べる。

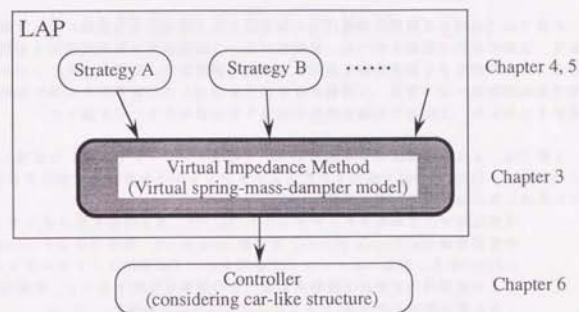


Fig. 2.9 Motion planning system and Virtual Impedance Method

2. 4 2章のまとめ

本章では、複数移動ロボット系の動作計画器を提案し、概要を述べた。

1節で、前章での議論に基づいて、分散管理オフライン+分散管理オンライン混合型計画の採用を決定した。具体的には、オフラインで大域的地図情報を利用する分散管理大域移動計画器(GMP)と、オンラインでセンサ情報を利用する分散管理局所回避計画器(LAP)との階層構造をなす計画手法を構築した。GMPとLAPの分担を、以下の様に決定した。

GMP:『動く』かつ『通信不可能』かつ『動作が決定している』物体を回避する軌道を生成する。

LAP: GMPで対応できない問題の解決を図る。すなわち、『動作の決定していない』物体を回避する。

2節では、GMPに必要な機能について考察した。更に、その機能に沿って適用可能な、従来の研究の概要を述べた。具体的には、2段階探索と双方向探索を利用した方法、2つの相反する要求仕様を実現する軌道の表現方法、動作のパターンのみが既知である障害物への対策等、3種類の従来研究を紹介した。実際的な大域計画器を実現するためには、上記の方法論を有効に統合する必要があることを論じた。

3節では、LAPの具体的な手法として、『仮想インピーダンス法』を提案した。この手法は、[KHATIB 86]で提案された人工ポテンシャル法を実時間で利用できるように改良した方法であり、基本原理は、

仮想的なマスで表現されたロボット(robot)は、ある時刻におけるロボットの目標位置(reference point)、障害物(obstacle)、他のロボット(other robots)から、仮想バネ・ダンパで表現された『仮想的なインピーダンス』により仮想外力を受けて運動をする。その運動を実現するべく、実際のロボットに動作指令を与える。

により計画を実現するものである。また、この手法が、(1)群れの概念の表現可能性、(2)単純なアルゴリズム、(3)制御パラメータの自律的なチューニング、という3つの仕様を十分に満たすものであることを議論した。

第3章

仮想的なインピーダンスを用いた 分散管理型局所計画器の設計

- 3. 1 『仮想インピーダンス法』の定式化
- 3. 2 停留・妨害状況の定義とその対策
- 3. 3 環境に応じたパラメータ設計論
- 3. 4 動作計画シミュレーション
- 3. 5 3章のまとめ

本章では、2、3節で概要を述べた分散管理型局所計画器『仮想インピーダンス法』の詳細について述べる。

3.1節で、仮想的なインピーダンスの定式化を行い、仮想マス、仮想バネ定数、仮想ダンパ定数等の設計パラメータの定性的な性質について述べる。

3.2節では、『仮想的インピーダンス法』において問題となる停留の定義とその対策、すなわち停留解決器のアルゴリズムについて述べる。また、複数移動ロボット系に固有な問題点として『妨害状況』を定義し、それに対処する妨害解決器の設計法についても述べる。シミュレーションにより、妨害解決器の妥当性を検証する。

3.3節では、3.1節で示した設計パラメータの具体的な設計方法について述べる。『仮想インピーダンス法』は、実時間で計画を行うため、環境の変動に対する適切な対応が必要となる。そのため、最適化手法を用いて、センサ情報ボタンと準最適パラメータの関係をニューラルネットワークに学習させ、そのニューラルネットワークを実時間で利用する方式を提案する。シミュレーションにより、提案した手法の妥当性を確かめる。

3.4節で、『仮想インピーダンス法』の有効性を確かめるシミュレーションを行う。まず、3.2節の『妨害状況』と『妨害解決器』の有効性を検証する。次に、3.3節で提案したパラメータ設計法の妥当性について検証する。

3.5節で、本章のまとめを行う。

Fig. 3.1に論文全体における本章の位置付けを示す。

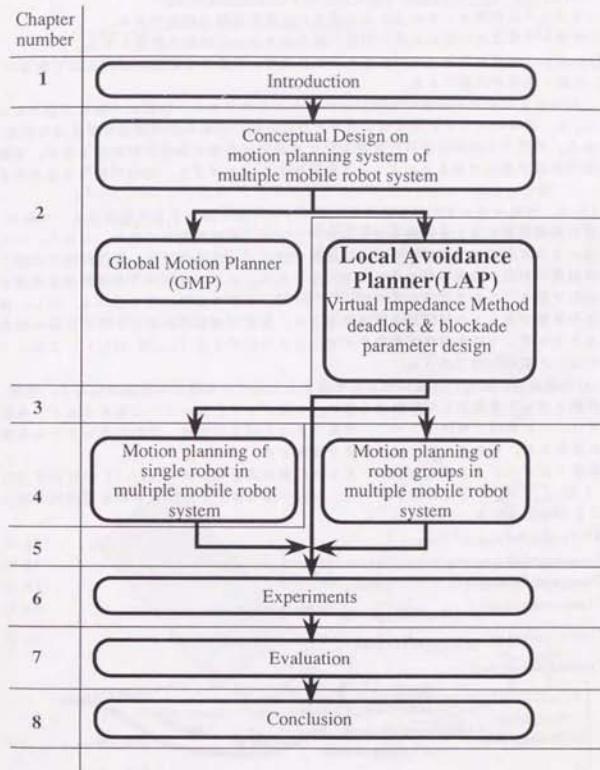


Fig. 3.1 The structure of chapter 3

3.1 『仮想インピーダンス法』の定式化

本論文では、以下に示す3つの仮定のもとで定式化を行う。

- (1) ロボットは内界センサにより自分自身の位置を正確に同定できる。
- (2) ロボットをどの方向にも進行可能な質点であるとして動作計画をする。
- (3) ロボットは自分自身の持つ外界センサにより、自身から距離R以内にある物体の位置・速度が計測できる。

上記の仮定は従来の研究でも用いられているものであり、問題の一般性を損なわない。(1)は、通常のオフィスや工場など、滑りの少ない通常の室内環境では妥当な仮定である。内界センサによる自己位置同定には誤差が累積する短所が存在するが、実際は絶対位置が既知であるランドマークを計測することにより、対処可能であると考えられる。

(2)は、移動ロボットの車輪機構をLAPでは考慮しないことを意味する。一般に通常の車輪機構を有する移動ロボットに対しては、車輪機構をGMP、LAP、コントローラのどの箇所で考慮するかの問題が存在する。本論文では、動作計画の問題と車輪機構の制御の問題を切り分けて考えるために、コントローラで車輪機構を考慮する立場を採る。この具体的な手法論については、6章の実験の所で述べる。但し、複数台の移動ロボットの協調搬送制御のように、高度な制御性が要求される作業の場合、LAPでロボットの車輪機構を考慮することも可能である〔太田 93b〕、これについては、7章の評価で述べる。

(3)の仮定は、従来の技術レベルを考慮すると必ずしも適当な仮定ではなく、実際に移動ロボットを動かして実験する際の一つのネックとなっていたものである。本論文では、この問題を解決するべく、移動ロボット相互の位置・姿勢計測システムを新たに構築した。これについては、6章で提案する。

仮想インピーダンス法におけるロボットの運動方程式を式(3.1)～(3.7)に示す(Fig. 3.2)。なお、以下の表記に於いて、 ϵ_j 内の記号は、そのどちらかを選択的に用いることを示している。

$$M_1 \ddot{x} = F_{tra,1} + F_{obs,1} + F_{rob,1} \quad (3.1)$$

$$F_{tra,1,x} = K_{tra,x}(\dot{x}_{1,T} - \dot{x}_1) + D_{tra,x}(\dot{x}_{1,T} - \dot{x}_1) \quad (3.2)$$

$$F_{tra,1,y} = K_{tra,y}(\dot{y}_{1,T} - \dot{y}_1) + D_{tra,y}(\dot{y}_{1,T} - \dot{y}_1) \quad (3.3)$$

$$F_{obs,rob,1,x} = \sum_{\{O_j, I_j\} \in Neighbor_1} f_{obs,rob,1,x} \quad (3.4)$$

$$F_{obs,rob,1,y} = \sum_{\{O_j, I_j\} \in Neighbor_1} f_{obs,rob,1,y} \quad (3.5)$$

$$f_{obs,rob,1,x} = \begin{cases} K_{obs,rob} \left(\frac{1}{od_{1j} - L_{[min,rmin]}} - \frac{1}{L_{[obs,rob]} - L_{[min,rmin]}} \right) \frac{ox_j - x_1}{od_{1j}} + \\ D_{obs,rob} \left(\frac{1}{od_{1j} - L_{[min,rmin]}} - \frac{1}{L_{[obs,rob]} - L_{[min,rmin]}} \right) (\dot{ox}_j - \dot{x}_1) & (od_{1j} \leq L_{[obs,rob]}) \\ 0 & (od_{1j} > L_{[obs,rob]}) \end{cases} \quad (3.6)$$

$$f_{obs,rob,1,y} = \begin{cases} K_{obs,rob} \left(\frac{1}{od_{1j} - L_{[min,rmin]}} - \frac{1}{L_{[obs,rob]} - L_{[min,rmin]}} \right) \frac{oy_j - y_1}{od_{1j}} + \\ D_{obs,rob} \left(\frac{1}{od_{1j} - L_{[min,rmin]}} - \frac{1}{L_{[obs,rob]} - L_{[min,rmin]}} \right) (\dot{oy}_j - \dot{y}_1) & (od_{1j} \leq L_{[obs,rob]}) \\ 0 & (od_{1j} > L_{[obs,rob]}) \end{cases} \quad (3.7)$$

$F_{tra,1} = (F_{tra,1,x}, F_{tra,1,y})^T$: 目標位置からの力

$F_{obs,1} = (F_{obs,1,x}, F_{obs,1,y})^T$: 障害物からの力

$F_{rob,1} = (F_{rob,1,x}, F_{rob,1,y})^T$: 他のロボットからの力

$x_1 = (x_1, y_1)^T$: ロボットiの現在位置

$u_{1,T} = (u_{x1,T}, u_{y1,T})^T$: ある時刻Tにおけるロボットiの目標位置

$o_1 = (ox_1, oy_1)^T$: 障害物iの現在位置 (Fobsの時)

i : ロボットiの現在位置 (Frobの時)

$od_{1j} = |o_j - x_1|$: ロボットi・障害物j間の距離 (Fobsの時)

i, j : ロボットi,j間の距離 (Frobの時)

Neighbor_i: ロボットiのセンシングエリア (半径R_i)

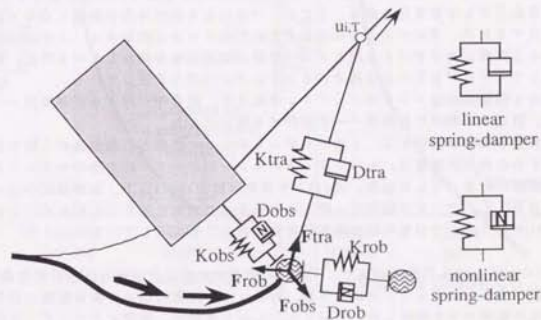


Fig. 3.2 Impedance models for Virtual Impedance Method

2章で述べたように、仮想インピーダンス法は人工ポテンシャル場が発生する力をバネに凝集させ、加えてダンパを付加した方法である。設計パラメータとしては仮想質量 M 、バネ定数 K 、ダンパ定数 D 、バネ長 L の4種類が存在する (Table 3.1)。これらパラメータの意味と仕様とを説明する。

(1)バネとバネ長: バネ定数 K は仮想外力の位置に関する感度の大きさを表現する。衝突の回避性能と解析の簡潔さの観点から、ロボット・目標位置間には線形バネを、ロボット・ロボット間、ロボット・障害物間には相対距離に反比例する力が働く非線形バネを設計した。Fig. 3.2において、線形バネ・ダンパ、非線形バネ・ダンパの表記が異なることに注意されたい。通常のポテンシャル法においては、相対距離の2乗に反比例する力が働くモデルが一般的であるが、本論文では、計算の高速性の観点から距離そのものに反比例するモデルを用いた。バネ自然長 L_{obs} 、 L_{rob} は、仮想外力が付加される領域の大きさを規定する。一般のロボット間には相互距離が L_{rob} 以上の時には外力が加わらないモデル (非固定端のバネ) により衝突回避をする。群れを構成している状態にあるロボット間では L_{rob} 以上の時に引力が付加されるモデル (固定端のバネ) を採用すれば群れの概念が表現可能である。バネ最小長 L_{omin} 、 L_{rain} は、ロボット・障害物間、ロボット間の最近接距離を表す。上記の定式化では、最近接距離となった時の仮想斥力が無限大となるように設計されている。最終的に、静止状態における相対距離と仮想外力との関係はFig. 3.3のように表現できる。Fig. 3.3(a)が非固定端のバネ、Fig. 3.3(b)が固定端のバネにおける関係である。

(2)質量: 仮想質量 M はロボットの動きにくさの度合いを示すため、各ロボットの優先度を意味する。すなわち、優先度が高いと自認するロボットは標準より重い仮想質量を設定する事により、仮想外力の影響をより小さくすることができる。

(3)ダンパ: ダンパ定数 D は仮想外力の速度に関する感度の大きさを表現すると共に、散逸系として安定化を図る。ここで、バネによる仮想外力が距離に応じて非線形に変化するため、ダンパによる外力もそれに対応させる必要がある。上記の定式化では、ある距離におけるバネ定数とダンパ定数が臨界減衰を満たしている時に、距離が変化してもそれがきちんと保持されるようにダンパ項を決定している。

この4種類のパラメータのチューニングにより、障害物に対する回避性能——回避距離、回避する軌道の曲率等——を設計する事ができる。

計算量の概略を評価する。1台のロボットがセンサ情報と目標位置から動作指令を算出するための計算量は、そのロボットのセンシングエリア内の他のロボットと障害物の個数の和を n とした場合、式(1)より乗算回数が $16n+4$ 以下、加算回数が $16n+6$ 以下となる。ここで、 n の値は高々数十のオーダーであると考えられるため、この計算回数はオンラインでの動作計画が充分可能であることを示している。

(1)のバネとバネ自然長の所で、「衝突の回避のために、ロボット・目標位置間には線形バネを、ロボット・ロボット間、ロボット・障害物間には相対距離の逆数に比例する非線形バネを設計した」と記した。しかしながら、仮想インピーダンス法は離散時間系で動作するため、ロボット・ロボット、ロボット・障害物間に相対距離の逆数に比例する非線形バネを設計しても、ロボット同士、ロボットと障害物が衝突する

Table 3.1 Design parameters in Virtual Impedance Method

Kinds	Parameters
Virtual Mass	M
Virtual Spring Coefficient	K_{tra} , K_{rob} , K_{obs}
Virtual Damper Coefficient	D_{tra} , D_{rob} , D_{obs}
Normal Length of Virtual Spring	L_{rob} , L_{obs} , L_{rmin} , L_{omin}

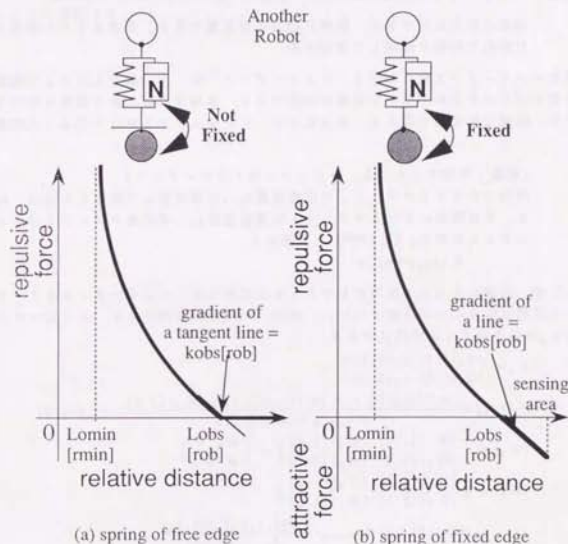


Fig. 3.3 Relationship between relative distance and virtual force

危険性が存在する。そのため、本論文では、ロボットの最大速度を、そのロボットの一番近傍に存在する他の物体との距離に応じて、制限する方策を採用した。具体的には以下のように設定する。

$$V_{\text{limit}} = \frac{d - \text{radius}}{2 \cdot T} \quad (3.8)$$

d : 一番近傍との物体との距離
 radius : ロボットの半径
 T : ロボットのサンプリング時間

すべてのロボットが上記の式に沿って最大速度を設定すれば、1サンプリング時間あたり一番近傍のロボットとの距離の半分は絶対に移動しないため、結果的に衝突を回避できる。

仮想インピーダンス法における“インピーダンス”について考察する。[JIS 87]によると、インピーダンス (impedance) の定義は、以下のようにになっている。

励振の出力に対する比、励振と出力は複素量であり、両者ともその偏角は同じ割合で時間に比例して増加する。

仮想インピーダンス法における“インピーダンス”は、どの値を入力として励振し、どの値を出力とするかで様々な定義が可能である。本論文で取り扱う環境は動的であるため、時間を固定して考える。本論文では、インピーダンスを以下のように定義する。

[定義：時刻 T における、ロボット i のインピーダンス]

時刻 T におけるロボット i の目標位置 $u_{i,T}$ を周波数 ω で励振した時の、 $u_{i,T}$ と、その時刻におけるロボット i の現在位置 $x_{i,T}$ の比をロボット i のインピーダンス行列 $Z_i \in \mathbb{C}^{2 \times 2}$ と定義する。

$$Z_i x_{i,T} = u_{i,T}$$

ここで、励振する入力、出力ともベクトルとなるため、インピーダンスはインピーダンス行列となることに注意されたい。式(3.1)~(3.7)を用いると、インピーダンス行列 Z_i は以下のように定式化できる。

$$Z_i = \begin{bmatrix} z_{i,1,1} & z_{i,1,2} \\ z_{i,2,1} & z_{i,2,2} \end{bmatrix}$$

$$z_{i,j,k} = \frac{-\omega^2 [M_i][j,k] + j\omega [D_i][j,k] + [K_i][j,k]}{K_{\text{tra},x} + j\omega D_{\text{tra},x}} \quad (j,k=1,2)$$

$$[M_i] = \begin{bmatrix} [M_i][1,1] & [M_i][1,2] \\ [M_i][2,1] & [M_i][2,2] \end{bmatrix} = \begin{bmatrix} M_i & 0 \\ 0 & M_i \end{bmatrix}$$

$$[D_i] = \begin{bmatrix} [D_i][1,1] & [D_i][1,2] \\ [D_i][2,1] & [D_i][2,2] \end{bmatrix}$$

$$[D_i][1,1] = D_{\text{tra},x} \frac{F_{\text{obs},i,x} F_{\text{rob},i,x}}{x_i^2}$$

$$[D_i][1,2] = [D_i][2,1] = 0$$

$$[D_i][2,2] = D_{\text{tra},y} \frac{F_{\text{obs},i,y} F_{\text{rob},i,y}}{y_i^2}$$

$$[K_i] = \begin{bmatrix} [K_i][1,1] & [K_i][1,2] \\ [K_i][2,1] & [K_i][2,2] \end{bmatrix}$$

$$[K_i][1,1] = K_{\text{tra},x} \frac{F_{\text{obs},i,x} F_{\text{rob},i,x}}{x_i^2}$$

$$[K_i][1,2] = [K_i][2,1] = -\frac{F_{\text{obs},i,x} F_{\text{rob},i,y}}{y_i}$$

$$[K_i][2,2] = K_{\text{tra},y} \frac{F_{\text{obs},i,y} F_{\text{rob},i,y}}{y_i^2}$$

ここで、 $[M_i]$ 、 $[D_i]$ の非対角成分は0の値を、 $[K_i]$ の非対角成分は非0の値を有している。

ある時刻 T における系全体のインピーダンスも同様にして導出することができるが、ここでは省略する。

3. 2 停留・妨害状況の定義とその対策

3. 2. 1 停留と『妨害状況』の定義

本論の方法を含めてポテンシャル場を用いた計画法では、ロボットがポテンシャル場の局所ミニマムに落ち込む停留問題を解決する必要がある。本論での停留とは、Fig. 3.4に示すように仮想外力の釣り合いにより式(3.1)が

$$M_1 \ddot{x}_1 = 0 \quad (3.9)$$

となり、ゴール以外の地点で停止する状態を意味する。

さらに、本論のように分散管理型計画にポテンシャル法を用いた場合の新たな問題点として、複数台のロボットがロボット相互の干渉力により、お互いが停止することなく、式(3.1)も

$$M_1 \ddot{x}_1 \neq 0 \quad (3.10)$$

であるが、ゴールになかなか到達できない状況が発生する (Fig. 3.5)。これは $M_1 \ddot{x}_1$ のある方向成分が0となり、かつ、目標への移動にはその方向成分が必要である場合を意味する。この状況を解決するために、本論では『妨害状況 (blockade)』を次のように定義する。

『妨害状況』とは、あるロボットの、ある時刻における目標位置と実際の位置との距離がしきい値 l_{block} を越えた状況である。

上記の定義は、他のロボットや障害物の存在により、あるロボットのある時刻における目標位置からの“ずれ”が過大である状況を意味し、 l_{block} はその“ずれ”量が過大かどうかを判断するしきい値である。この定義により Fig. 3.5 の状況を妨害状況として検知することができる。妨害状況は、静的な停留とは異なり、動的状況下で発生する状態であり、動的な停留と解釈可能である。停留と妨害状況は共に回避すべきであると考える。

3. 2. 2 停留と妨害状況への対策

停留と妨害状況を解決する LAP の詳細な構造を Fig. 3.6 に示す。ここで、番犬 (watch dog) とはセンサ情報と目標位置からロボットの状況を判断して、適切な戦略を選択するモジュールである。停留は $M_1 \ddot{x}_1 = 0$ で、妨害状況は目標位置と現在位置の距離で状況を決定する。番犬は停留を検出した時には停留解決器 (deadlock solver) を、妨害状況を検出した時には妨害解決器 (blockade solver) をそれぞれ起動する。それ以外の時には操縦器 (pilot) を働かせ、3. 2. 3 項に示すアルゴリズムでロボットの動作計画を行う。

3. 2. 2. 1 停留解決器の戦略

停留解決器 (deadlock solver) では、停留しているロボットのポテンシャルエネルギーから停留点近傍のバネ定数を求め、剛性の最も低い方向へ動作指令を出す戦略を採用した (Fig. 3.7)。

(1) 停留しているロボットのポテンシャル関数のヘシアン行列 A の固有値 λ_1, λ_2

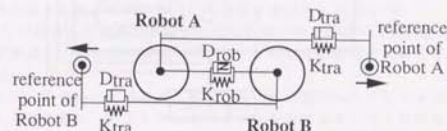


Fig. 3.4 An example of deadlock

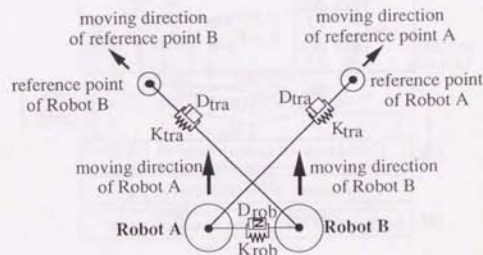


Fig. 3.5 An example of "Blockade"

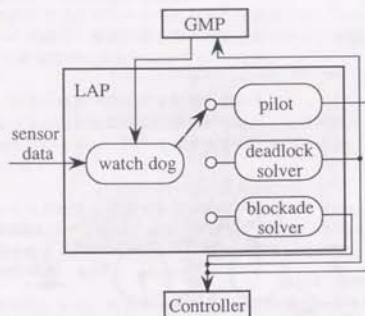


Fig. 3.6 System architecture for LAP

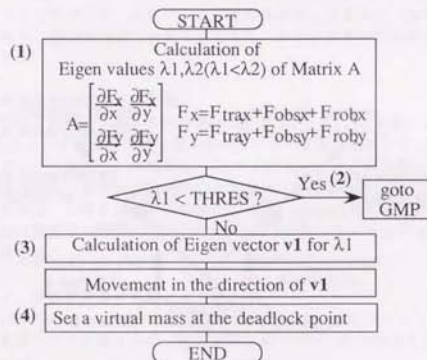


Fig. 3.7 Flow chart of deadlock solver

($\lambda_1 \leq \lambda_2$)を求める。

(2)固有値 λ_1 があるしきい値 THRES 以上となった場合には、停留した地点を新たな出発地点として、現在検出されている周辺の障害物を既知障害物として GMP を駆動する (Fig. 3.8(a)) ([ARA1 89])。これは、障害物や他のロボットに異常に接近した結果、ロボットがどの方向に回避しても危険であり、局所的情報のみを利用した対策では解決不可能であると判断したことを意味する。なお、GMP も LAP と同様に分散管理的に起動され、他のロボットの計画と同期させる必要のないことに注意されたい。

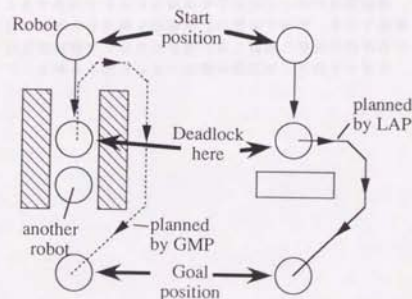
(3)上記の条件を満たさなかった場合、固有値 λ_1 に対応する固有ベクトル v_1 の方向に微小速度を加える。

これは、物理的には、バネ定数の最も弱い方向にロボットを微小変位させることを意味する。得られる方向は2つ存在するが、その選択は例えば、ゴールに向かって時計周りか反時計周りかのルール等を用いて決定する (Fig. 3.8(b))。

(4)ロボットが動作した後に、停留点に仮想的なマスを設置した動作計画をすることにより、停留点の不安定化を図り、ロボットが再びその停留点に戻らないようにする ([LEE 90])。

この戦略は、停留時には力学的に最もエネルギー消費が小さくなる方向への移動計画を実現したものであり、力学モデルの観点から妥当であると考えられる。

ここで THRES は、どの程度の局所停留点まで LAP で解決を図るかの指標である。局所停留点は、環境が複雑になるほど発生やすくなるため、THRES はどの程度複雑な環境まで LAP で対処するかを決定するしきい値と解釈できる。THRES を大きく設定



(a) Solver with GMP (b) Solver without GMP

Fig. 3.8 Examples of deadlock solver

すると、できるだけLAPのみを用いて計画する形態となり、逆に小さく設定すると、環境の複雑さに応じてGMPを頻繁に起動する形態となる。

3. 2. 2. 2 妨害解決者の戦略

妨害解決者(blockade solver)は「ロボットの目標位置を停止させる」なる戦略を採る。換言すれば、妨害状況を検知した際に、本来時間と共にゴールへ進んでいくはずのロボットの目標位置を停止させることを意味する。この方法は妨害状況に非対称性を発生させて解消させると共に、万一解消できない場合でも停留へ着かせる狙いを有している。LAPは自律的な分散計画であるので、目標位置の停止には時間差が生れ、不安定な釣合い状態を崩すこととなる。その効果については、4章のシミュレーションで述べることにする。また、妨害解決者の戦略は、目標地点からロボットにかかる仮想外力に限界を与えることを意味するため、ロボット間、ロボット・障害物間の干渉時に停留に陥りやすく、迅速な回避行動を実現できる。

ロボットの目標位置の運動方程式を、式(3.11)に示す。

$$u_1 = \begin{cases} p_{next} \\ u_1 \end{cases} \quad \begin{cases} (|x_1 - u_1| \leq l_{block}) \\ (|x_1 - u_1| > l_{block}) \end{cases} \quad (3.11)$$

p_{next} : 次のステップの目標位置

妨害解決者の有効性を示すシミュレーションは3.4節で示す。

本論の扱う動的な場においては、より複雑な問題を考える事もできる。例えばFig. 3.9に示すように、目標位置が小円の円周を回転運動をし、その周囲に大円の障害物がある場合を想定する。現在位置がその大円の外側の場合、いつになっても大円の内側に入る事ができなくても、ロボットは大円の外側を回り続け、妨害にも停留にもならない。この問題は本研究の手法で解決することができないが、このような特殊な問題を除けば、停留問題のほとんど全てを本論のLAPで解決することができ、かつ、その構造は単純である。すべての動作計画問題を解決するためには、ある一定時間後にロボットが最終目的位置に到達しない場合にGMPで再計画を行う戦略を採用することにより、ロボットのゴール到達を保証できると考えられる。

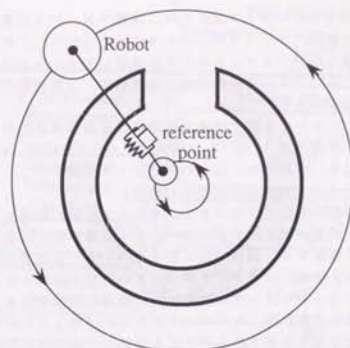


Fig. 3.9 One of the problems which cannot be solved with the proposed method

3.3 環境に応じたパラメータ設計論

前節で、『仮想インビデンス法』には、仮想マス、バネ定数、ダンパ定数、バネ自然長の4種類の設計パラメータが存在することを述べた。これらのパラメータを設計する際に、多種の設計パラメータを以下に適切にチューニングするか、環境の変動をどう取扱うか、各ロボットの動作計画の実時間性、自律性をどのように保つか、等の問題を考える必要がある。

前章のFig. 2.2において本論で提案する動作計画器の概念図を示したが、Fig. 3.10に、再度示す。本節では、Fig. 3.10内のパラメータ設計器 (parameter designer) の設計方法を論じる。本節では、以下のアプローチを採る。

[STEP 1: 設計パラメータの変換と安定性の議論]

前節までの4種類の設計パラメータを、そのまま設計に利用しようとすると複雑である。そのため、過渡特性に関するパラメータ・定常特性に関するパラメータという、設計しやすい形に変換する。過渡特性に関するパラメータとして、つりあい状態における極を用いる。これは、パラメータがつりあい状態において安定か否かが判断できる利点を有するため、パラメータの変換を安定領域に限定できる。また、設計されたパラメータは環境の変動に対してロバストでなければならないので、ロボットの周囲の障害物が最も疎である状態と最も密である状態のそれぞれに対する極を新たな設計パラメータとする。定常特性に関するパラメータとしては、ロボット間のつりあい距離とロボット相互が仮想力を及ぼしあう最大距離 L_{rob} を用いる。

[STEP 2: 特定のサンプルデータに対するセンサ情報ボタンと準最適設計パラメータの導出]

STEP 1で、設計パラメータが変換され、安定なパラメータの領域が導出できた。次に、ある与えられた環境と、それに対する適切な設計パラメータを導出する必要がある。しかしながら、複数移動ロボット系における各ロボットの状態を一般的に表現し、それに対する最適パラメータを導出することは難しい。本論では、サンプルデータ (ある固定された環境下で、ある設計パラメータによって動作するロボットの走行状況) をいくつか用意し、それぞれのデータに対する設計パラメータの組合せ最適化問題を解く。サンプルデータとしては、同一の環境においてロボットの密度、速度を変化させたものをいくつか用意すれば良い。その時の各ロボット近傍のセンサ情報ボタンも記憶する。このセンサ情報ボタンとは、各ロボットに時々刻々と入力されるセンサ情報をマクロにとらえ、あるロボット近傍の環境の状況を表現したものである。具体的には、周囲のロボットの密度、速度、方向等の情報を含むものである。

[STEP 3: センサ情報ボタンと準最適パラメータセットとの関係の導出および、その実時間での利用]

STEP 3で得られたセンサ情報ボタンを入力とし、準最適設計パラメータを出力とするニューラルネットワークを構成し学習させることにより、センサ情報ボタンと準最適パラメータとのfittingを実現する。実際に各ロボットを動作させる時には、上記の手続きで学習を完了させたニューラルネットワークを各ロボットに持たせれば良い。各ロボットは、自分自身のセンサ情報ボタンを検知し、ニューラルネットワークを用

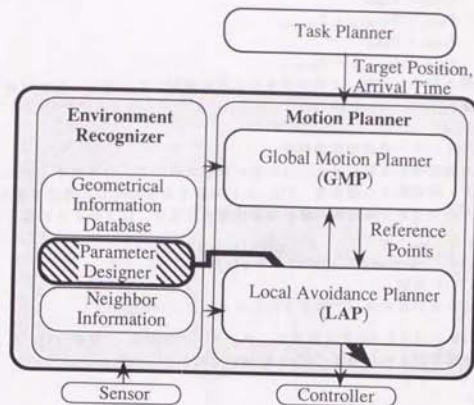


Fig. 3.10 Architecture of the proposed method

いて準最適パラメータセットを実時間で自律的に求め、動作することができる。このネットワークによりパラメータ設計器を構成する。STEP 1~3を項別に分けて論じる。

3.3.1 設計パラメータの変換と安定性の議論

環境内に多数のロボットが動作している動的環境下において、適切な未知パラメータを決定することは大変難しい。本節では、ロボット同士が停留してつりあった状態、すなわち静的状態に着目して、その状況における過渡特性、定常特性を解析し、パラメータ設計を実現するというアプローチを採る。

まず、設計時に以下の前提を設ける。

- ・仮想質量は、ロボット間の優先度を表すものとし、ここでは、設計対象としない。
- ・バネ最小長 ($L_{\text{min}}, L_{\text{min}}$) は、実際のロボットの外接円の直径と等しく設計する。これより、ロボット間の干渉は存在しなくなる。
- ・簡単のため、障害物とロボットに関するパラメータはすべて同一とする。また、特に断りがない限り、目標位置からの仮想バネ定数・仮想ダンパ定数を x, y 方向に関して同一とする。すなわち、独立した設計パラメータは、

$$\begin{aligned} L_{\text{rob}} &= L_{\text{obs}} \\ L_{\text{block}} \end{aligned}$$

$$\begin{aligned}K_{rob} &= K_{obs} \\K_{tra} &= K_{tra,x} = K_{tra,y} \\D_{rob} &= D_{obs} \\D_{tra} &= D_{tra,x} = D_{tra,y}\end{aligned}$$

の6種類となる。以降、上記の等号の左辺に表記したパラメータを代表して用いることとする。

3.3.1.1 過渡特性の解析

最初に過渡特性を解析する。つりあい状態における、あるロボットと、近傍の他のロボット・障害物との関係は、Fig. 3.11のように単純化して表記できる。この時のロボットのつりあい点近傍の離散時間状態方程式は、以下ようになる。

$$\begin{bmatrix} x[k] \\ x[k+1] \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ M-DT+KT^2 & -2M+DT \end{bmatrix} \begin{bmatrix} x[k-1] \\ x[k] \end{bmatrix} \quad (3.12)$$

T: 1サンプリング時間,

K, D: つりあい点において換算されたバネ・ダンパ定数

式(3.12)におけるK, Dの安定領域は、ジュリーの判別法〔美多 84〕により以下の3つの不等式で囲まれた領域 (Fig. 3.12) であることがわかる。

$$\begin{cases} D-KT > 0 \\ \frac{KT^2}{M} > 0 \end{cases} \quad (3.13)$$

$$2DT-KT^2 < 4M \quad (3.14)$$

$$2DT-KT^2 < 4M \quad (3.15)$$

上記の領域内で適当なK, Dを設計すればよい。しかしながら、式(3.12)のK, Dは、近傍の他の物体の疎密に応じて変化するため、K, Dの代わりに、式(3.12)における極を設計パラメータとした方が妥当である。安定であるための極の条件は、極の絶対値が1より小さいことであるため、その範囲内で極の設計をすればよい。極を p_1, p_2 とすると、 p_1, p_2 とK, Dの関係は、簡単な計算により、式(3.16), (3.17)で表されることがわかる。

$$K = \frac{M}{T^2}(-1+p_1p_2) + \frac{1}{MT^2}(2-p_1-p_2) \quad (3.16)$$

$$D = \frac{1}{MT}(2-p_1p_2) \quad (3.17)$$

ここで、ロボットのつりあい点近傍の挙動は、環境がどのように変動しても安定でなければならない。そこで、本論文では、

(1) 一番疎である状態 (Fig. 3.13(a))

(2) 一番密である状態 (Fig. 3.13(b))

のそれぞれにおいて、安定領域内におさまる極配置——(1)に対して2つ、(2)に対して2つ、の計4つ——を行う。但し、つりあい状態において(2)に示す状況が最も密となるためには、 l_{rob} に関する拘束式が存在するが、それに関しては後で述べる。ここで、最も疎な状態では、目標位置からの力しか存在しないため、

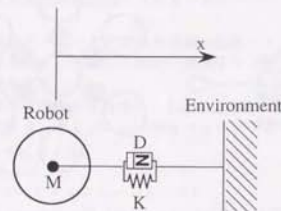


Fig. 3.11 Virtual impedance model between a robot and an environment

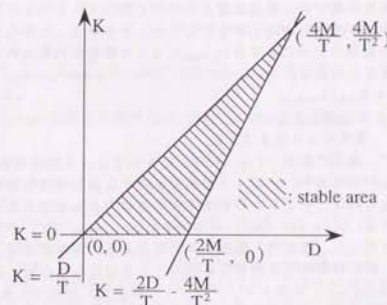
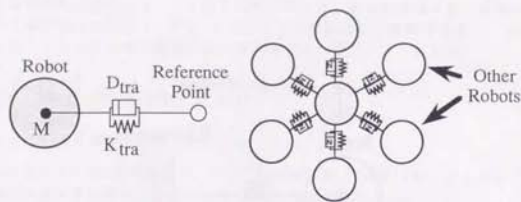


Fig. 3.12 A stable area about K and D



(a) The sparsest situation (b) The densest situation

Fig. 3.13 The impedance model near a robot

$$K = K_{tra} \quad (3.18)$$

$$D = D_{tra} \quad (3.19)$$

となる。最も密な状態では、目標位置からの力に加えて、6つの近傍の物体（ロボットまたは障害物）からの力が存在する。ここで、あるロボットが他のロボットに加えることのできる仮想外力の大きさは l_{block} によって限定されるため、Fig. 3.13の状態より密になることはない。

$$K = K_{tra} + N_{rob} \cdot K_{rob-e} \quad (3.20)$$

$$D = D_{tra} + N_{rob} \cdot D_{rob-e} \quad (3.21)$$

N_{rob} : 環境により決まる定数

となる。ここで、通常の環境 (Fig. 3.13) においては、2次元的に仮想外力が付与されるため $N_{rob}=4$ となるが、ロボットが相互通行する廊下環境の様にロボットの流れが1次元である時は、ロボットの前後方向のみに力がかかると考えられるため、 $N_{rob}=2$ が適当である。式(3.20)、(3.21)における K_{rob-e} 、 D_{rob-e} は、つりあい状態におけるロボット（ロボット・障害物）間のバネ定数である。本論文では、ロボット（ロボット・障害物）間に距離に反比例する非線形バネを設計したため、 K_{rob} 、 D_{rob} は、それぞれ K_{rob-e} 、 D_{rob-e} と、ロボット（ロボット・障害物）間のつりあい距離の関数となる。 K_{rob-e} 、 D_{rob-e} から K_{rob} 、 D_{rob} を導出する方法は後で述べる。

上記の解析より、最も疎な状態における極 (P_{d1} 、 P_{d2} とする) と最も密な状態にお

ける極 (P_{d1} 、 P_{d2} とする) を『新しい設計パラメータ』とすると、式(3.15)～(3.20)を用いて、設計パラメータの K_{tra} 、 D_{tra} 、 K_{rob-e} 、 D_{rob-e} が導出できることがわかった。

3.3.1.2 定常特性の解析

次に定常特性を決定する。定常特性とは、ここでは、つりあい状態におけるロボット（ロボット・障害物）間の相対位置関係を表す。この特性に関する仮想インピーダンス法の設計パラメータは、 l_{rob} と l_{block} の2変数である。ここで、 l_{block} は、設計パラメータとして扱いにくいため、代りにロボット間のつりあい距離を用いる。すなわち、最も近接した際のロボット（ロボット・障害物）間のつりあい距離（以下 $dist$ と呼ぶ）と力のかかる最大距離 l_{rob} を『新しい設計パラメータ』とする。

3.3.1.3 設計パラメータの変換の手順

以上の議論を整理すると、『新しい設計パラメータ』から設計パラメータを求める変換手順は以下の様になる。

- (1) P_{s1} 、 P_{s2} (最も疎な状態における極)、 P_{d1} 、 P_{d2} (最も密な状態における極)、 $dist$ 、 l_{rob} を設計する。但し、6つの設計パラメータには以下の制約がつく。

$$\begin{cases} P_{s1} \geq P_{s2} \geq P_{d1} \geq P_{d2} \\ \cdot |P_{s1}| < 1, |P_{d1}| < 1 \quad (i=1,2) \\ \cdot l_{rob} \leq (\text{ロボットのセンシングエリア}) \\ \cdot dist < l_{rob} \\ \cdot l_{rob} < 2 \cdot dist \\ \cdot dist < l_{rmin}(omin) \end{cases} \quad (3.22)$$

5番目の式は、Fig. 3.13(b)が最も密な状態であるための条件である。ここで、 $l_{rob} > 2 \cdot dist$ とすると、Fig. 3.13(b)において、中心にいるロボットを取り囲んでいる6台のロボットの外側に存在するロボットも、中心のロボットに仮想外力を与えることになってしまう。すなわち、上記の条件は、Fig. 3.13(b)の状態が最も密であることを保証するものである。

- (2) K_{tra} 、 D_{tra} 、 K_{rob-e} 、 D_{rob-e} を、式(3.16)～(3.21)を用いることにより P_{s1} 、 P_{s2} 、 P_{d1} 、 P_{d2} から導出する。

- (3) K_{rob} 、 D_{rob} 、 l_{block} を以下に示す式(3.23)～(3.25)により導出する。導出手順の説明は省略する。

$$K_{rob} = \frac{(dist - l_{rmin})^2}{l_{rob} \cdot l_{rmin}} \quad (3.23)$$

$$D_{rob} = \frac{dist - l_{rmin} - D_{obs-e}}{l_{rob} \cdot l_{rmin}} \quad (3.24)$$

$$l_{block} = \frac{K_{rob-e}}{K_{tra}} (dist - l_{rmin})^2 \left(\frac{1}{dist - l_{rmin}} - \frac{1}{l_{rob} \cdot l_{rmin}} \right) \quad (3.25)$$

上記の設計により、環境がどのように変化しても、つりあい点近傍において安定が保証されるパラメータ設計が可能となった。しかしながら、設計パラメータの数は6つであり、ある環境が与えられた時に、適したパラメータを試行錯誤的に決定するこ

とが大変困難である。次節以降では、与えられた環境に対する適切なパラメータを学習・記憶させる方法を述べる。

3. 3. 2 特定のサンプルデータに対する準最適設計パラメータとセンサ情報パタンの導出

複数移動ロボット系の環境の状態を一般的に表現し、それに対する最適パラメータを導出することは難しい。本論文では、(1)適当なサンプルデータに対する準最適設計パラメータを導出する。(2)各ロボットが準最適パラメータを用いた場合のセンサ情報パターンを導出する。(3)準最適設計パラメータとセンサ情報パターンとの関係をニューラルネットワークで学習させる。という3段階の戦略を採る (Fig. 3.14)。実時間で動作する時に、センサ情報パターンをニューラルネットワークへの入力とすることにより、そのセンサ情報パターンに対する設計パラメータを導出し、利用することができる。本項では、上記の前半部分にあたる、準最適設計パラメータとセンサ情報パタンの導出方法について述べる。

3. 3. 2. 1 準最適設計パラメータの導出

適当なサンプルデータに対して、前項で得られた6つの設計パラメータ $p_{s1}, p_{s2}, p_{d1}, p_{d2}, dist, l_{rob}$ とある評価関数に関する組合せ最適化問題を解くことにより、準最適設計パラメータを導出する。解法として、最急降下法、シミュレーテッド・ア

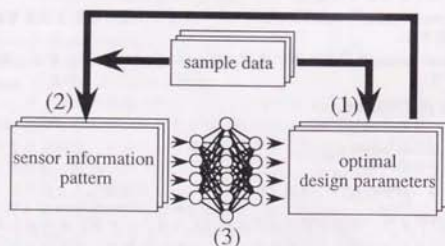


Fig. 3.14 The method for designing parameters

ーリング, genetic algorithm等、多種存在するが、本論では、評価値の勾配を必要とせず、局所最小点が多い場合にも適用でき、探索の際のパラメータチューニングが難しい理由によりランダム多スタート局所探索法 ([茂木 93]) を用いる。この手法は、ランダムに発生された多数の初期解それぞれに局所探索法を適用し、得られた準最適解のうちで最良のものを出力する方法である。以下にその概略を示す。

[ランダム多スタート局所探索法]

ステップ1 (初期設定) : ITE (最大反復回数) を設定する。 $k = 0$ とする。

ステップ k (初期解, 改良) : 以下の (i), (ii) を行う。

(i) 初期可能解 x を求める。 $k := k + 1$ とする。

(ii) x の近傍内で x より良い評価関数値を持つ可能解 y を探索する (近傍探索)。そのような y が見つかれば、その中で最小の評価関数値を有する y を選択し、 $x := y$ 。 $k := k + 1$ として、(ii) の先頭に戻る。存在しない場合には、 $k < ITE$ の時は、 x を局所解として記憶し、(i) に戻る。 $k \geq ITE$ の時には、処理を終了させる。上記の手続きにより得られた局所解のうち最良のものが準最適解である。

評価関数としては、各ロボットの平均ゴール到達時間や得られた軌道の曲率の平均等が考えられる。また、6つの『新しい設計パラメータ』は変域内で離散化して適用する。近傍は、「6つのパラメータのうちのどれか1つを1ステップだけ増加あるいは減少させることにより得られる領域」とする。具体的には、以下の手順で準最適解を導出する。

[ランダム多スタート局所探索法を用いた準最適設計パラメータの導出]

ステップ1 (初期設定) : ITE を設定する。 $k = 0$ とする。

ステップ k : 以下の (i), (ii) を行う。

(i) 初期設計パラメータをランダムに設定する。 $k := k + 1$ とする。初期設計パラメータに対する評価関数値を求める。具体的には、あるサンプルデータ (各ロボットのスタート位置、ゴール位置、目標到達時間) において、環境内のすべてのロボットの設計パラメータとして同一の初期設計パラメータを与えて、各ロボットを実際にシミュレーション上で動作させ、その際的评价関数値を求める。評価関数としては、

$$PI = \frac{1}{\text{Number_of_robot}} \sum_{i=1}^{\text{Number_of_robot}} \text{Time}_i$$

Number_of_robot : 環境内のロボットの台数
Time_i : ロボット i のゴール到達時間

を用いる。これは、環境内のすべてのロボットのゴール到達時間の平均値を意味している。

(ii) 初期設計パラメータの近傍のパラメータに対して、上記と同様なシミュレーションを行い、評価関数値を求める。初期設計パラメータの評価関数値より小さい評価関数値を有する近傍パラメータが存在した時、その中で最小の評価関数値を有する近傍パラメータを選択し、それを初期設計パラメータに置き換える。 $k := k + 1$ として、

(ii)の先頭に戻る。存在しなかった場合、 $k < \text{ITE}$ の時には、初期設計パラメータを局所解として記憶し、(i)に戻る。 $k \geq \text{ITE}$ の時には、処理を終了させる。
上記の手続きにより得られた局所解のうち最小のものを準最適解とする。

上記の手続きを、Fig. 3.15を例に採って説明する。Fig. 3.15(a)に示すようにサンプリングデータ、すなわち環境内の各ロボットのスタート位置・ゴール位置・目標到達時間が与えられている。

(i)ある適当な初期設計パラメータ（例えば、 $p_{s1} = p_{s2} = 0.5$, $p_{d1} = p_{d2} = -0.5$, $\text{dist} = 1.0$, $l_{\text{rob}} = 1.5$ ）を各ロボットに付与して、実際にロボットの動作計画を実現する。各ロボットが移動を完了するのに要した時間により、評価関数値を得ることができる（Fig. 3.15(b)）。

(ii)初期設計パラメータの近傍のパラメータ、すなわち、

$$p_{s1} = p_{s2} = 0.4, p_{d1} = p_{d2} = -0.5, \text{dist} = 1.0, l_{\text{rob}} = 1.5$$

$$p_{s1} = p_{s2} = 0.5, p_{d1} = p_{d2} = -0.5, \text{dist} = 1.0, l_{\text{rob}} = 1.5$$

$$p_{s1} = p_{s2} = 0.5, p_{d1} = p_{d2} = -0.4, \text{dist} = 1.0, l_{\text{rob}} = 1.5$$

$$p_{s1} = p_{s2} = 0.5, p_{d1} = p_{d2} = -0.6, \text{dist} = 1.0, l_{\text{rob}} = 1.5$$

$$p_{s1} = p_{s2} = 0.5, p_{d1} = p_{d2} = -0.5, \text{dist} = 0.9, l_{\text{rob}} = 1.5$$

$$p_{s1} = p_{s2} = 0.5, p_{d1} = p_{d2} = -0.5, \text{dist} = 1.1, l_{\text{rob}} = 1.5$$

$$p_{s1} = p_{s2} = 0.5, p_{d1} = p_{d2} = -0.5, \text{dist} = 1.0, l_{\text{rob}} = 1.4$$

$$p_{s1} = p_{s2} = 0.5, p_{d1} = p_{d2} = -0.5, \text{dist} = 1.0, l_{\text{rob}} = 1.6$$

の8種類の近傍設計パラメータを用いて、各ロボットの動作計画を行い、それぞれのパラメータに対する評価関数値を求める。評価関数値が最小である近傍パラメータの評価関数値が初期設計パラメータの評価関数値（Fig. 3.15(b)では、49.7sec）より小さい時には、その近傍パラメータを初期値に置き換えて、(ii)の最初に戻る。そうでない場合は、現在の設計パラメータが局所最小点であるため、そのパラメータを記憶して、(i)に戻る。評価関数の計算がITE回を超えたら、処理を終了する。

上記の手続きで得られた局所最小点のうち、評価関数値が最小である設計パラメータを、その環境における準最適設計パラメータとする。これにより、あるサンプリングデータに対する準最適設計パラメータを導出することができた。

3.3.2.2 センサ情報パタンの導出

センサ情報パターンは、ロボットを越える近傍の環境の状況、すなわち、近傍の他のロボットの密度、速度、方向等の情報を正確に反映したものでなければならない。本論文では、以下の手順で導出する。

(1)サンプリングデータに対して、各ロボットに、先に求めた準最適設計パラメータを用いて、ロボットの動作計画を行う。

(2)各サンプリング時間における、各ロボットのセンシングエリア内で仮想力が働いていない領域、すなわち各ロボットからの距離がロボットのセンシングエリアより小さく l_{rob} より大きい領域（Fig. 3.16の灰色の部分）に存在する他のロボットの速度ベクトルの方向をセンサ情報パターンとして用いる（Fig. 3.16）。これは、

- ・各ロボットが局所的に獲得できる情報でなければならない
- ・各ロボット間距離が近くなると相互に斥力が働いて回避行動を起こすため、ロボットの挙動が環境を反映したものではなくなる可能性があるという理由による。

例えば、Fig. 3.16で中心に存在するロボットRのセンサ情報パターンとしては、ロボットRのセンシングエリア内に存在する近傍のロボット1〜ロボット12のうち、ロボットRとの距離が l_{rob} 以上であるロボット、すなわち、ロボット1〜ロボット9の速度ベクトルの方向を用いることを意味する。

(3)近傍のロボットを速度の方向別に分類して、方向毎にロボットの台数ヒストグラムを導出する（Fig. 3.17）。Fig. 3.17では、方向をロボットRと等しい方向を0rad、速度ベクトルの方向の定義域を $-\pi \text{ rad} \sim \pi \text{ rad}$ として、

$$\text{方向0: } -\pi/4\text{rad} \sim \pi/4\text{rad},$$

$$\text{方向1: } \pi/4\text{rad} \sim 3\pi/4\text{rad},$$

$$\text{方向2: } 3\pi/4\text{rad} \sim \pi \text{ rad, または, } -\pi \text{ rad} \sim -3/4\text{rad}$$

$$\text{方向3: } -3\pi/4\text{rad} \sim -\pi/4\text{rad},$$

の4つに分割している。ロボット1〜ロボット9を速度ベクトルの方向別に分類し、それぞれの方向に含まれるロボットの台数を求めると、

$$\text{方向0に向かっているロボットの台数 (vec[0]) = 3台}$$

$$\text{方向1に向かっているロボットの台数 (vec[1]) = 2台}$$

$$\text{方向2に向かっているロボットの台数 (vec[2]) = 3台}$$

$$\text{方向3に向かっているロボットの台数 (vec[3]) = 1台}$$

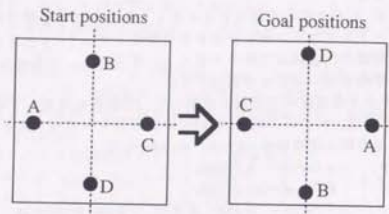
なるヒストグラムが得られる。

(4)上記のヒストグラムをすべてのロボット・サンプリング時間について導出し、平均をとる。得られたヒストグラムが、あるサンプリングデータの準最適解に対するセンサ情報パターンである。

以上提案したセンサ情報パターンは、すべてのロボットがほぼ等しいパラメータ、ほぼ等しい目標速度で動作している環境下においては、有効であると考えられる。相互に速度の異なったロボットが存在する環境では、上記の手法をそのまま用いると、環境識別が不十分となり、パラメータ設計器が適切に働かない可能性がある。その場合には、近傍のロボットの速度の方向のみではなく、速度の分散等の情報特徴量として組入れることにより適用可能であると考えられる。

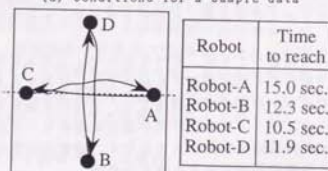
3.3.3 準最適パラメータセットの学習・記憶

前項の解析により、準最適設計パラメータセットとセンサ情報パタンの組が得られた。ある環境に対する準最適設計パラメータセットを実時間で利用するためには、センサ情報パターンを入力とし、準最適設計パラメータセットを出力とするfittingをする必要がある。fittingの有力な手法として重回帰分析があるが、この方式は基本的には回帰関数が既知である場合には有効であるが、本節の問題のように関係が未知である場合には適さない。本項では、センサ情報パターンを入力とし、準最適設計パラメータセットを出力とするニューラルネットワークを構成し、教師付き学習を施すこと



Robot	Start Position	Goal Position	Demanded time to reach
Robot-A	(-6, 0)	(7, 0)	10 sec.
Robot-B	(0, -6)	(0, -7)	10 sec.
Robot-C	(6, 0)	(-7, 0)	10 sec.
Robot-D	(0, 6)	(0, 7)	10 sec.

(a) Conditions for a sample data



$$\begin{aligned}
 PI &= 15.0 + \\
 &12.3 + \\
 &10.5 + \\
 &11.9 \\
 &= 49.7
 \end{aligned}$$

(b) Calculation of penalty index

Fig. 3.15 An example for getting penalty index to some sample data

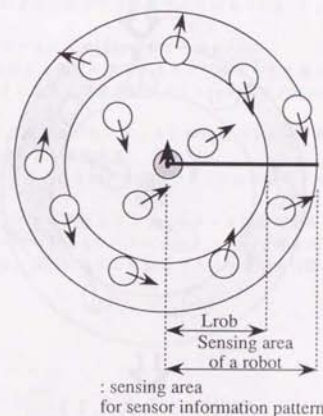


Fig. 3.16 Sensing area for getting sensor information pattern

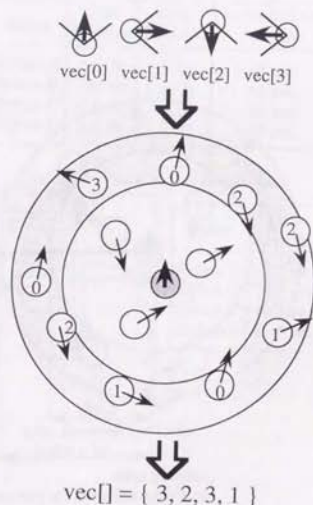


Fig. 3.17 Sensor information pattern with histogram

により、準最適設計パラメータセットの学習・記憶を実現する。ニューラルネットワークとしては、通常の階層型ニューラルネットワークを用い (Fig. 3.18)、バックプロパゲーション法を用いて学習を実現する。

本節の冒頭で、「サンプルデータとしては、同一の環境においてロボットの密度、速度を変化させたものをいくつか用意すれば良い」と述べた。ここでは、ニューラルネットワークによる学習機能を用いているため、ある固定した環境内を走行するロボットの密度、速度に関してばらつきを持ったサンプルデータの選択が望ましいことがわかる。これは、サンプルデータのロボットの台数、目標ゴール到達時間を変化させることにより実現可能である。サンプルデータの具体的な選定については、3.4節のシミュレーションの所で述べる。

ロボットが動作する環境が複雑である場合、ニューラルネットワークを階層構造にする方法も存在する。例えば、ロボットが廊下環境と広場環境の両者を走行する場合、それぞれの準最適パラメータを共通のニューラルネットワークに学習させることは、ネットワークの大規模化を招き、学習の収束性の観点などから好ましくない。そこで、Fig. 3.19に示すように、廊下環境に対して設計パラメータを学習するネットワーク (NN for corridor) と、広場環境に対して設計パラメータを学習するネットワーク (NN for plaza) を分離して、その上位階層に、廊下環境と広場環境を識別するニューラルネットワーク (NN for corridor/plaza selection) を用意することにより、複雑な問題を細分化することができると考えられる (Fig. 3.19)。

本論文で提案したパラメータ設計法をまとめる。

(1) ロボットが動作する環境を表現するサンプルデータを幾つか用意する。それぞれのサンプルデータに対する準最適設計パラメータをランダム多スタート局所探索法を用いて導出する。

(2) サンプルデータに対して各ロボットに準最適パラメータを用いた場合の各ロボットのセンサ情報パターンを導出する。

(3) 準最適設計パラメータとセンサ情報パターンとの関係をニューラルネットワークで学習させる。

(1)~(3)のオフラインの手続きによりパラメータ設計が実現される。(1)~(3)の手続きを踏んだニューラルネットワークに、オンラインで得られるセンサ情報パターンを入力することにより、環境に適応した設計パラメータを実時間で導出できる。

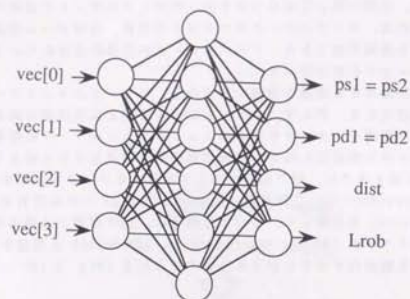


Fig. 3.18 Neural network in parameter designer

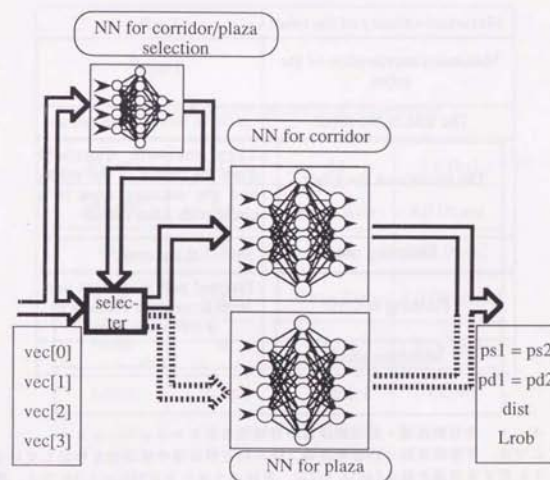


Fig. 3.19 Hierarchical neural network system in parameter designer

3. 4 動作計画シミュレーション

本章で提案した手法の有効性を検証するシミュレーションを行う。最初に、シミュレーションにおけるロボットのスペックをTable 3.2に示す。ここでのパラメータは、現在一般に研究されている実際の移動ロボットの仕様と将来的に必要な要求仕様を参考にして決定した。本節では、以下の3種類のシミュレーションを行う。

(a) 3. 2節で示した停留・妨害状況の定義と停留解決器・妨害解決器の有効性を示すシミュレーション

(b) 3. 3節で示したパラメータ設計法の有効性を示すシミュレーション

(c) ロボット群の動作計画シミュレーション

上記のうち、(a)と(b)は、提案した手法の検証のために行う。(c)は、群を構成するロボット間に固定端の仮想バネモデルを構成することにより群行動が実現できることを示すシミュレーションである。

Table 3.2 Specification of robots for simulation

Maximum velocity of the robot	1 m/s
Maximum acceleration of the robot	1 m/s ²
The size of the robot	a circle with 0.25m radius
The sensors on the robot	They measure distances from the center of the robot, and the sensing area is a circle with 3.0m radius
Sampling time	0.1 second
Planning at GMP	Planned path is straight line with a constant velocity of a reference speed
reference speed	1 m/s

3. 4. 1 停留解決器・妨害解決器の有効性を示すシミュレーション

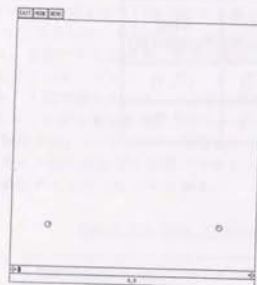
ここでは、停留解決器と妨害解決器、特に妨害解決器の有効性を検証している。ロボットに対する目標仕様をTable 3.3に、設計パラメータをTable 3.4に示す。設計パラメータは、安定な極を適当に選択して実験的に決定した。シミュレーション結果をFig. 3.20 (妨害解決器を組込んだ場合)、Fig. 3.21 (妨害解決器を組込まない場合)

Table 3.3 Conditions for the environment for verifying effectiveness of "Blockade Solver"

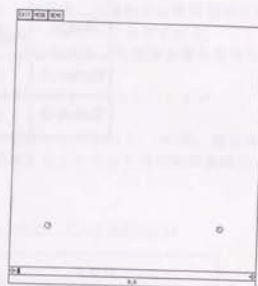
Robot	Start Position(m)	Goal Position(m)
Robot-A	(-7,-7)	(7,7)
Robot-B	(7,-7)	(-7,7)

Table 3.4 Values of designed parameters

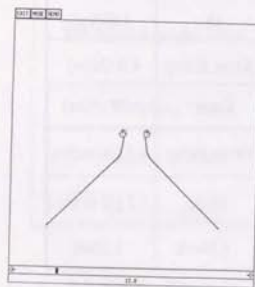
ps1,ps2	0.8	M	1.0 [kg]
pd1,pd2	0.4	Ktrax,Ktray	4.0 [N/m]
dist	2.0[m]	Krob	2.88 [N/m]
Lrob	3.0[m]	Dtrax,Dtray	4.0 [N sec/m]
Nrob	4	Drob	1.2 [N sec/m]
Lrmin	0.5[m]	Lblock	1.2[m]



(a) 0 sec.

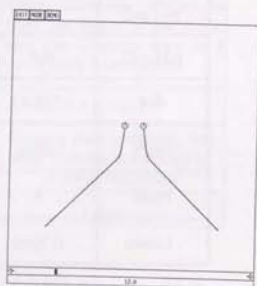


(a) 0 sec.



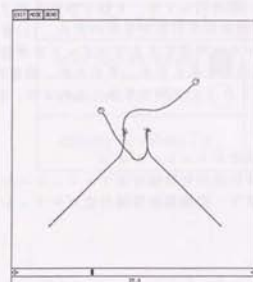
(b) 1.2 sec.

Fig. 3.20 Results WITH
blockade solver
(continue to next page)

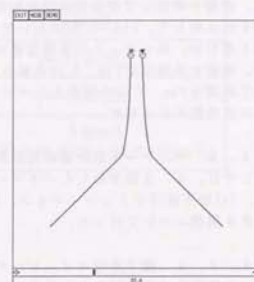


(b) 1.2 sec.

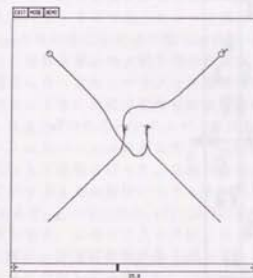
Fig. 3.21 Results WITHOUT
blockade solver
(continue to next page)



(c) 2.2 sec.

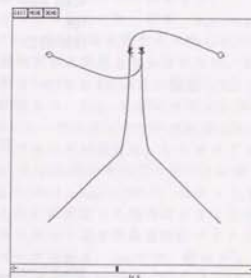


(c) 2.2 sec.



(d) 29.8 sec.

Fig. 3.20 Results WITH
blockade solver



(d) 34.5 sec.

Fig. 3.21 Results WITHOUT
blockade solver

に示す。ここで、円はロボットの外周を、円に付属の線分は、その瞬間におけるロボットの速度ベクトルの向きを示している。Fig. 3.20の流れを見ると、(b)で両者のロボットが妨害状況を検知して妨害解決器を起動し、その結果、各ロボットの目標位置(reference point)が停止し、ロボットの動きもFig. 3.21より遅くなっている。(c)では、停留を検知して停留解決器を起動し、最終的に29.8秒で移動を完了している。それに対して、Fig. 3.21においては妨害状況を判断できないため、(c)までは一見速く進むが、各ロボットの目標位置がゴールに到達するまでロボットが停留しないので、停留する時間がFig. 3.20の場合と比較して長くなる。そのため、最終的な移動完了時間はFig. 3.20の場合と比べて長く、34.5秒である。これより、妨害解決器の有効性が示された。

3.4.2 パラメータ設計法の有効性を示すシミュレーション

ここでは、3.3節で示したパラメータ設計法の有効性を示すシミュレーションを行う。(a)廊下走行シミュレーション、(b)廊下・広場混合環境の走行シミュレーション、の2種類について行った。

3.4.2.1 廊下走行シミュレーション

一つの廊下を相互走行する複数台のロボットのシミュレーションを行う(Fig. 3.22)。ロボットの走行環境とロボットのスタート・ゴールラインの座標値をTable 3.5に示す。各ロボットのスタート・ゴール位置を以下の様に決定した。

・各ロボットの初期y座標は、一方のスタートラインからもう一方のスタートライン

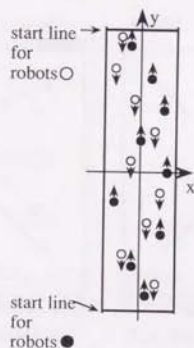


Fig. 3.22 A corridor environment for simulations

Table 3.5 Conditions for the corridor environment

width of the road(WIDTH)	5.0[m]
length of the road	22.0[m]
start line for robots ●	y=-11.0 [m]
start line for robots ○	y=-10.5 [m]
density of robots (ρ)	5 kinds (0.02,0.04,0.06,0.08,0.10/[m ²])

に向かって($\rho \cdot \text{WIDTH}$)mおきに配置する。その後、ロボットを $1/(\rho \cdot \text{WIDTH} \cdot V_{\text{ref}})$ 秒おきに両スタートラインに発生させる。

・各ロボットの初期x座標は廊下の横幅の範囲内でランダムに決定する。

また、3.3節で示した4つの極を2つずつの重根とした。即ち、 $p_{s1}=p_{s2}$, $p_{d1}=p_{d2}$ とみなすことにより新しい設計パラメータの独立した変数を6つから4つに減らした。目標位置とのx軸方向の位置偏差は動作表現上問題とならないため、 $x_{\text{tra},x}=0$ に固定した。また、パラメータ設計時の定数をTable 3.6のように設定した。ニューラルネットワークの中間層の数は実験的に決定した。Fig. 3.23にロボットの密度を0.08/m²のあるサンプルデータに対するニューラルネットワークを組み込んだ場合のシミュレーション結果を示す。この際、パラメータの変更は、10サンプリング、すなわち1秒毎に行なった。比較のため、Fig. 3.24に同じサンプルデータに対してすべてのロボットの設計パラメータの値を $p_{s1}=-0.4$, $p_{d1}=-0.7$, $\text{dist}=1.0$, $l_{\text{rob}}=1.4$, $N_{\text{rob}}=2$ (式(3.20),(3.21)参照のこと)に固定した場合のシミュレーション結果を示す。このパラメータは、ランダム多スタート法で準最適設計パラメータを求めた際に、評価関数値が最も悪かったパラメータである。(a)では、各ロボットが初期位置に存在している。(b),(c),(d)において、Fig. 3.24では、Fig. 3.23と比較して、各ロボットがゴールに到達しにくい状況となっている。これは、Fig. 3.23では、環境に応じて適切なパラメータ設計がなされているのに対して、Fig. 3.24では、つまりあい状態において安定が保持されているもののロボット同士の相互回避に時間がか

Table 3.6 Constants for parameter design in the corridor environment

iteration for random multi-start method(ITE)	1000
dimension of sensor information pattern	$4 (\frac{\pi}{4} \text{ rad for each})$
the number of sample data	40 (10 for each p 7: data for learning NN 3: data for testing)
iteration for learning for NN	100000
the number of units at input layer	4
the number of units at hidden layer	6
the number of units at output layer	4

かっていることが原因である。式(3.24)に示した評価指標を比較すると、Fig. 3.23の場合が31.6秒、Fig. 3.24の場合が44.6秒、ランダム多スタート法で得られた準最適設計パラメータに対する評価関数値が23.5秒(設計パラメータは、 $p_{\text{dist}} = -0.1, p_{\text{dist}} = -0.8, \text{dist} = 0.9, l_{\text{rob}} = 1.6, N_{\text{rob}} = 2$)となった。他のサンプルデータに対しても同様な結果が得られた(Table 3.7, Fig. 3.25)。Fig. 3.25においてグラフのほぼ下端に存在する横方向の線分はロボットが他の物体と一切干渉せずにゴールに到達するために要する時間である。このシミュレーションでは、廊下長が2.2mで速度が1m/sであるから、2.2秒となる。ニューラルネットワークによる学習に用いていないサンプルデータに対しても、学習に用いたサンプルデータとはほとんど変わらない性能が得られていることから、ニューラルネットワークによる学習機能が有効に動作していることがわかる。Table 3.7から、本論文で提案したニューラルネットワークを用いた場合の評価関数値が準最適解の場合の15~35%増程度であることがわかる。これは、準最適解の評価関数値が準最適解の79~99%増であることを考慮すると、ここで提案した手法は、十分適切なパラメータ設計法であると言える。

ここで、設計パラメータの l_{rob} を非常に大きく設計すると、ロボット間の相互斥力と廊下の壁からの力によりロボットが一切相互通行できなくなる状況が生じることがある。このときには評価関数値が無限大となるため、Table 3.7で示した最適解よりも更に悪い例であると言える。すなわち、Table 3.7における最適解は、真の最適解ではなく、あくまで目安であることに注意されたい。

本論文で提案した手法は、オフラインでの学習に多少の時間がかかるものの、動作計画時には実時間で各ロボットが自律的に実現できるため、適用範囲は大変広いと考えられる。

3. 4. 2. 2 廊下・広場混合環境の走行シミュレーション

ここでは、本論文で提案した手法が、環境の変化に対応して適用できることを示す。広場と廊下が接続している環境においてシミュレーションを行う(Fig. 3.26)。広場環境における学習は、廊下環境とは独立させて行った。広場環境におけるロボットの走行条件をTable 3.8に示す。パラメータ設計時の定数をTable 3.9に示す。評価関数は、廊下環境の際と同じ式(3.24)とする。廊下環境の条件は、先程のシミュレーションと同一とする。パラメータの変更も先程のシミュレーションと同様に10サンプリング、すなわち1秒毎に行った。Fig. 3.27に学習によるパラメータ設計を行った場合のロボットの挙動を示す。広場領域から廊下領域を横断しているロボットが、ある時間において、どちらの環境とみなしているかをFig. 3.28に示す。Fig. 3.27(c)から、横断しているロボットは、約1.8秒で広場領域から廊下領域へ移っていることがわかる。その時間を境に、ロボットは自分自身を取り巻く環境を広場領域から廊下領域とみなすようになっていくことが、Fig. 3.28からわかる。ここで、実際に存在している環境と、センサ情報を用いてみなしている環境が異なる場合が存在する(例えば、2.2秒のあたり、Fig. 3.28(d))が、この場合は、そのロボットを取り巻く環境が広場環境と類似した状況となっているためであると考えられ、この時のパラメータは、広場環境用のパラメータを用いることが望ましい。上記の考察により、本論文で提案した手法により環境の認識が良好になされていることがわかる。すなわち、環境の変化に迅速に対応したパラメータ設計が実現できていることが示された。

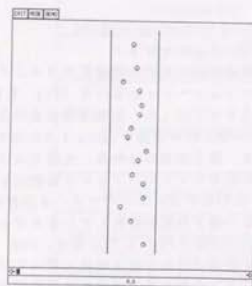
本設計手法の問題点としては、環境の変動により設計パラメータが大きく変化して、ロボットの挙動が滑らかでなくなる危険性があることが挙げられる。これについては、パラメータを変更する周期を緩やかにする、パラメータを変更する際に瞬間的にはなく徐々に変更する、という対策が考えられる。

上記のシミュレーションにおいて、パラメータを適切に設計することにより、ロボットの、ある程度望ましい挙動を得ることができた。しかしながら、環境内のロボットの密度が増加するにつれて、環境の混雑の度合いが高まっていることも確かであると考えられる。この混雑状況を回避するためには、環境内のロボットの行動に何等かの秩序を付与する必要がある。

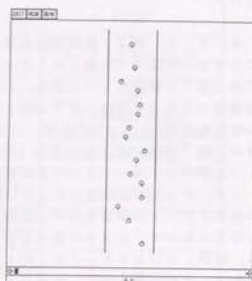
人間が雑踏の中で動作するときには、なんらかの戦略を用いていることにより、動作性能を改善させていることが多い。本論文では、4章で、各ロボットに行動戦略を付与することにより、環境内に自律的に秩序を付与する方法を提案する。

3. 4. 3 ロボット群の動作計画シミュレーション

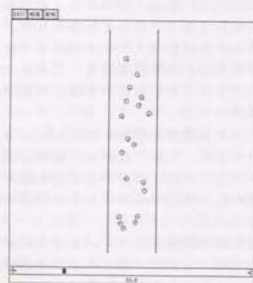
本項では、ロボット間のバネ・ダンパモデルを修正することにより、ロボット群の動作計画が容易に実現できることを示す。まず、ロボット群の表現方法について考察する。相互に群を形成しているロボット間に、Fig. 2.8に示したように相互に拘束さ



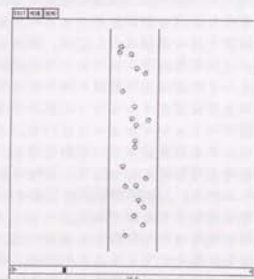
(a) 0 sec.



(a) 0 sec.



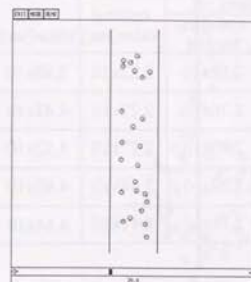
(b) 10 sec.



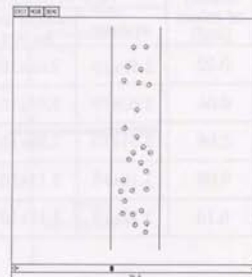
(b) 10 sec.

Fig. 3.23 Results WITH
parameter designer
in the corridor environment
(continue to next page)

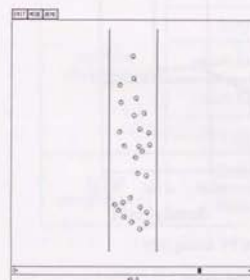
Fig. 3.24 Results WITHOUT
parameter designer
in the corridor environment
(continue to next page)



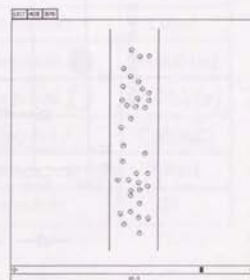
(c) 20 sec.



(c) 20 sec.



(d) 40 sec.



(d) 40 sec.

Fig. 3.23 Results WITH
parameter designer
in the corridor environment

Fig. 3.24 Results WITHOUT
parameter designer
in the corridor environment

Table 3.7 Comparison between the method WITH parameter designer and the one WITHOUT it

density of robots [m ²]	value by using NN [sec]			optimal value[sec]	worst value[sec]
	average	sample data for NN	non-data for NN		
0.02	2.59x10	2.60x10	2.58x10	2.22x10	3.98x10
0.04	2.63x10	2.58x10	2.76x10	2.27x10	4.42x10
0.06	2.91x10	2.88x10	2.99x10	2.27x10	4.52x10
0.08	3.16x10	3.11x10	3.29x10	2.34x10	4.60x10
0.10	3.12x10	3.12x10	3.14x10	2.41x10	4.64x10

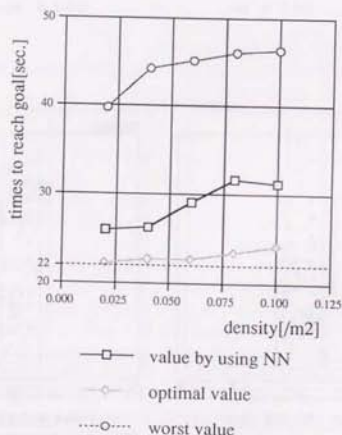


Fig. 3.25 Efficiency of the parameter designer with variation of density of robots

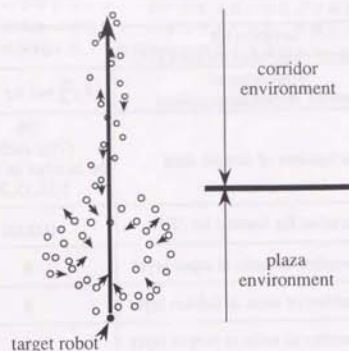


Fig. 3.26 A corridor/plaza fused environment for simulations
Table 3.8 Conditions for the corridor/plaza fused environment

corridor environment	width of the road(WIDTH)	5.0[m]
	length of the road	22.0[m]
	start line for robots ●	y=-11.0 [m]
	start line for robots ○	y= -10.5 [m]
plaza environment	density of robots (ρ)	0.08[m ²]
	Radius of the circle	7.0[m]
	Number of robots	20

Table 3.9 Constants for parameter design in the plaza environment

iteration for random multi-start method(ITE)	1000
dimension of sensor information pattern	$4 (\frac{\pi}{4} \text{ rad for each})$
the number of sample data	28 (7for each p the number of robots : 5,10,15,20)
iteration for learning for NN	100000
the number of units at input layer	4
the number of units at hidden layer	6
the number of units at output layer	4

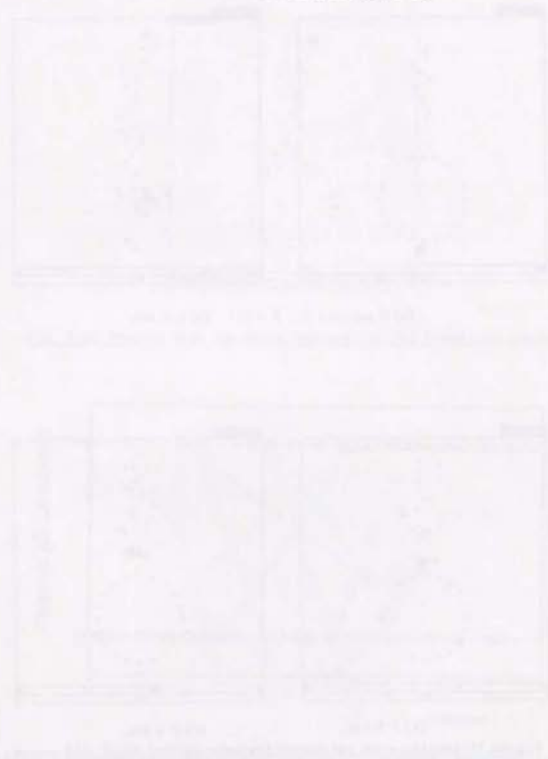
れたバネ・ダンパモデルを採用することにより、群を構成しているロボット同士が近接した位置を保つことができる (Fig. 3.29(a))。それ以外のロボット間や、ロボット・障害物間では Fig. 3.29(b) に示すモデルを採用して相互に独立した回避行動を実現する。

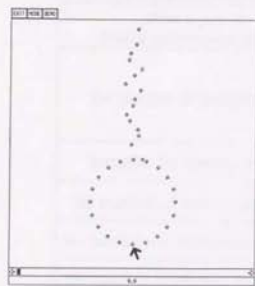
シミュレーション環境として、円周上に配置された複数のロボット群が、円領域を横切って同じ円周上のランダムに設定されたゴールに向かう広場領域を想定する (Fig. 3.30)。シミュレーションに用いる定数を Table 3.10 に、ロボットの設計パラメータを Table 3.11 に示す。ロボットの性能は、3.4.1 項のシミュレーションと同一とする。本シミュレーションでは、群行動の実現形態に関して考察するため、パラメータ設計法として 3.3 節で示した方法は用いずに、適当な設計パラメータにすべてのロボットに対して固定して与えた。また、同一の群を構成するロボット間に働く引力が過大なものにならないように、ロボット間距離がバネ自然長の 2 倍以上の距離となったときには、引力が働かないこととした。Fig. 3.31 にシミュレーション結果を示す。複数のロボット群が、群の形態を保持しつつ、相互に回避行動を実現できていることがわかる。

上記のシミュレーション結果により、ロボット間のバネ・ダンパモデルの若干の変更によりロボット群の動作計画が実現できていることが分かった。

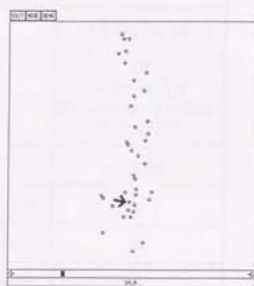
群行動の本来の目的を考えた場合、それは、群が実現すべき作業に依存するもので

ある。すなわち、協調して一つの大きな荷物を搬送しているロボット群の挙動と、群れて広い領域をセンシングするロボット群の挙動は異なるはずである。Fig. 3.31 の結果は、そのような意味からは不十分であると考えられる。本論文では、5 章において、作業の性質を考慮したロボット群の動作計画手法を提案する。

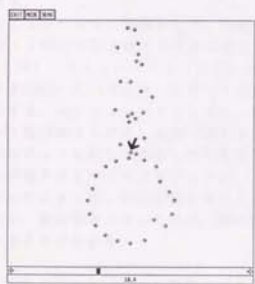




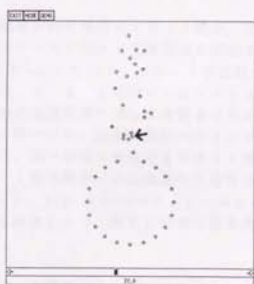
(a) 0 sec.



(b) 1.0 sec.

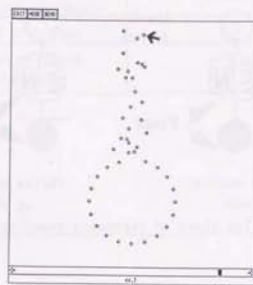


(c) 1.8 sec.



(d) 2.2 sec.

Fig. 3.27 Results with parameter designer in the fused environment
(continue to next page)



(e) 4.47 sec.

Fig. 3.27 Results with parameter designer in the fused environment

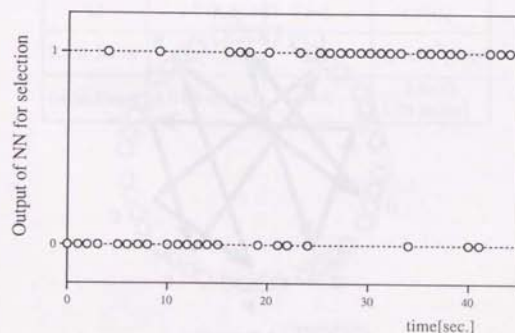


Fig. 3.28 Recognized environment by the robot which goes
from the plaza environment to the corridor environment

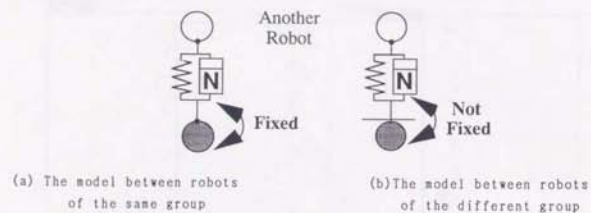


Fig. 3.29 Two kinds of impedance model between robots

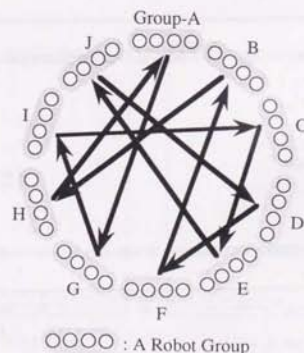


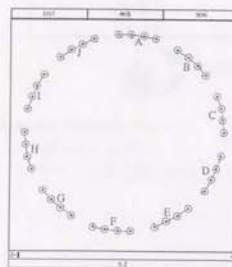
Fig. 3.30 A plaza environment for simulation of robot groups

Table 3.10 Conditions for the plaza environment

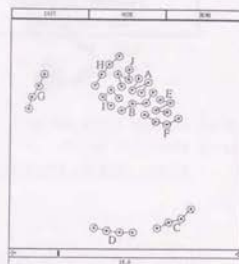
Radius of the circle for start/goal positions	7.5[m]
Number of groups	10
Number of robots for one group	4

Table 3.11 Design parameters for the robots in groups

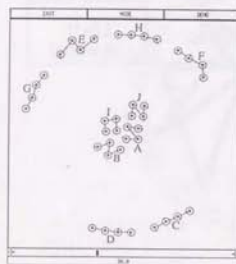
M	1.0 [kg]	Lrob	0.9[m]
Ktrax, Ktray	4.0 [N/m]	Krob	3.24×10^2 [N/m]
Dtrax, Dtray	4.0 [N sec/m]	Drob	3.6×10 [N sec/m]



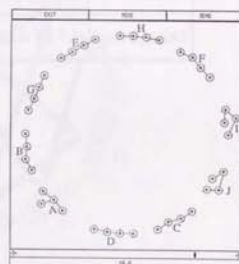
(a) 0 sec.



(b) 1.5 sec.



(c) 3.0 sec.



(d) 6.6 sec.

Fig. 3.31 Simulation results for robot groups

3.5 3章のまとめ

本章では、前章で提案した『仮想インピーダンス法』の詳細について説明した。

1節で、『仮想インピーダンス法』の定式化を行った。設計パラメータとしては、仮想質量、仮想バネ定数、仮想ダンパ定数、仮想バネ自然長の4種類が存在することを述べ、それぞれのパラメータの定性的な性質について述べた。

2節では、ゴール以外の場所に停止する『停留』を定義した。さらに、本論文のように分散管理型計画にポテンシャル法を用いた場合の新たな問題点として、複数台のロボットがロボット相互の干渉力により、相互に停止することなく、ゴールになかなか到達できない状況を『妨害状況』と定義した。停留、妨害状況それぞれに対する解決法の構造について述べた。『停留』と『妨害状況』の定義により、複数移動ロボット系の動作計画問題のほとんど全てをLAPで解決することができ、かつその構造は単純であることを示した。

3節では、仮想インピーダンス法における設計パラメータのチューニング方法について論じた。具体的には、以下の3つのステップから構成される。

STEP 1: 設計パラメータの変換と安定性の議論

1節で示した4種類の設計パラメータを直接的に設計に用いることは難しい。そこで、過渡特性に関する設計パラメータ（つりあい状態における極）と定常特性に関する設計パラメータ（つりあい距離、仮想外力がかかる最長ロボット間距離）を新しい設計パラメータとした。つりあい状態における安定性の議論により、適切な設計パラメータの変換を導出した。

STEP 2: 特定のサンプルデータに対する最適パラメータとセンサ情報パタンの導出

一般的な環境に対する最適パラメータを導出することは大変困難である。ここでは、あるサンプルデータ（ある環境とロボットの初期位置、ゴール、目標到達時間のセット）に対して、ある評価関数を設定して、それに対する組み合わせ最適化問題を解くことにより、最適パラメータとセンサ情報パタンの組を求めた。

STEP 3: センサ情報パターンと最適パラメータセットとの関係の導出および、その実時間での利用

STEP 2において得られたセンサ情報パターンと最適パラメータの組を階層型ニューラルネットワークに学習させる。学習されたネットワークを各ロボットが保持することにより、実時間で獲得されたセンサ情報パターンに適した設計パラメータが導出される。

4節では、シミュレーションにより提案した手法の有効性を検証した。具体的には、妨害解決の有効性を示した。さらに、廊下領域走行シミュレーションにより、3節で示したパラメータ設計法を用いた場合の評価関数値が、最適解と最大35%程度の差で収めることが確認された。また、廊下・広場混合領域走行シミュレーションにより、異なった環境を識別するニューラルネットワークの有効性が確認された。これにより、提案したパラメータ設計法の有効性が示された。最後に、ロボット群の動作計画シミュレーションを行った。しかしながら、ここで行われたシミュレーションには、混雑状況の回避、群の行動における作業の性質の表現、という観点から不十分な点が存在することを明らかにした。その問題点への対策を4章、5章で述べる。

第4章 複数移動ロボット系における 単体ロボットの動作計画

- 4.1 秩序生成の問題設定および『追従戦略』の提案
- 4.2 追従戦略の『仮想インピーダンス法』への導入
- 4.3 シミュレーションによる追従戦略の有効性の検証
- 4.4 異質な系における追従戦略の拡張
- 4.5 4章のまとめ

本章では、複数移動ロボット系の動作計画時の混雑状況を回避するために各ロボットに付与する戦略として、『追従戦略』を導入する。

4. 1節では、複数移動ロボット系において混雑状況を回避する、すなわち秩序を付与する方法について論じ、「自分自身前方に存在し、自分自身と同じ方向に向かっているロボットの後を追従する『追従戦略』」を提案する。

4. 2節では、4. 1節で示した『追従戦略』を『仮想インピーダンス法』に適用する。具体的には、最初に、追従関係にあるロボット間のバネ・ダンパモデルとそうでない関係にあるロボット間のバネ・ダンパモデルの2種類を用意する。次に、各ロボットが環境の状態からルールベースで追従する相手を決め、そのロボットとのモデルを追従するモデルに変更するという方法を提案する。

4. 3節では、『追従戦略』の有効性を廊下走行シミュレーションにより明らかにする。ロボットの密度を変化させた際の各ロボットの平均角速度を、追従戦略を用いた場合と用いていない場合とで比較することにより環境内のロボットの行動に秩序が付与されている度合いを調べる。

4. 4節では、環境内に互に動作性能の異なるロボットが存在するheterogeneousな系に対する追従戦略の拡張方法について述べる。この系においては、他のロボットに追従するのみでは、環境内にロボットの行動に秩序をもたらすことはできない。ここでは、状況に応じて前方のロボットを超越するという戦略を付加することにより、秩序の生成を目指す。

4. 5節では、本節のまとめを述べる。

Fig. 4.1に論文全体における本章の位置付けを示す。

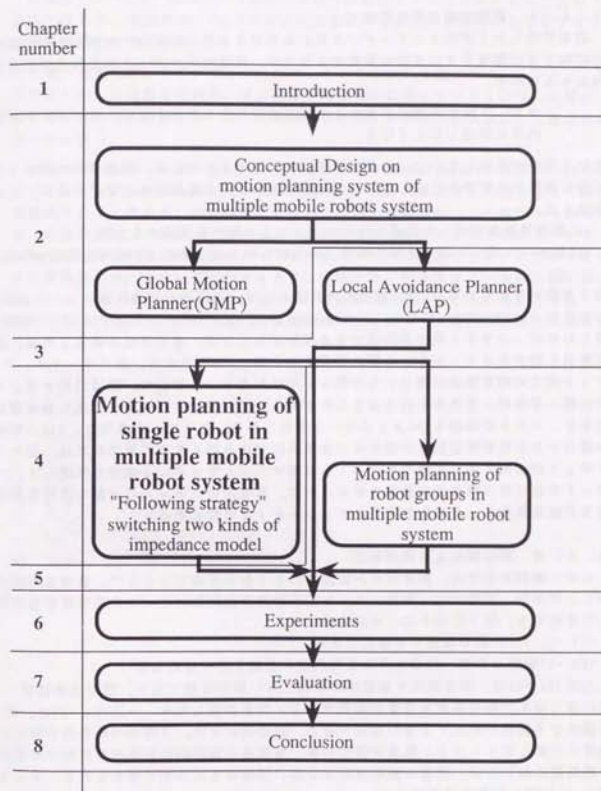


Fig. 4.1 The structure of chapter 4

4. 1 秩序生成の問題設定および『追従戦略』の提案

4. 1. 1 秩序生成の問題設定

前章で示した『仮想インピーダンス法』をそのまま用いた場合、ロボットの密度が比較的小さい環境下では有効に機能するものの、環境内のロボットの台数及び密度が増加するにつれ、

ロボット同士の相互干渉により混雑状況が生じやすくなり、各ロボットが目的地に到達しにくくなる、

という短所が存在していた。この混雑状況を回避するためには、環境内のロボットの行動に何等かの秩序を生成して、ロボット同士の干渉を軽減する必要があるが、その方法には、

(a)環境情報を利用した規範を与えることにより秩序を生成する方法、

(b)個々のロボット間の情報を利用した規範を与えることにより秩序を生成する方法、

の2種類が考えられる。(a)の方法の代表例として、道路の右側通行等のルールを用いた方法([KATO 92])がある。この方法は、一旦環境とそれに付随するルールが整備されれば、たやすく秩序が形成できる長所がある反面、適用環境が限定される、環境整備に要するコストがかかる等の短所を有する。(b)の方法は、各ロボットに、ロボット同士の相互関係に基づいた行動ルールを与えることにより、環境内のロボットの行動に自律的に秩序を形成させるものである。この方法は、多様な環境に適用可能であり、コストの問題も(a)より小さいと考えられる。そこで、本章では、(b)の戦略に基づいた分散管理型動作計画手法の構築を研究の目的とする。具体的には、個々のロボットが状況に応じて他のロボットを追従することでロボットの群を形成して、ロボットの走行時の角速度を低減させることで、環境内のロボットの行動に秩序を付与する手法を提案し、その有効性をシミュレーションで検証する。

4. 1. 2 群行動による秩序形成

人間・動物社会では、個体同士が状況に応じて群を形成することで、秩序を形成することが多い。本節では、複数ロボット系の動作計画問題にとって有効な群形態に関して考察する。以下に示す二つの群を例にとる。

(Ex.-1)一つの箱を搬送する複数の人々

(Ex.-2)雑踏の中で、同じ方向に進む人達が形成する一時的な群

上記(Ex.-1)は、群を構成する個体に変化しない静的な群であり、群形成の結果、『単体では不可能な高度な作業が実現できる』効果が得られる。一方(Ex.-2)は、群を構成する個体に変化する動的な群であり、群形成により、『環境内の各個体同士が他者に干渉しにくくなる』効果が得られる。各個体が環境内を自律的に移動する動作計画問題においては、後者の群形成がより良い効果をもたらすと考えられる。そこで、(Ex.-2)に関して次節で考察する。

4. 1. 3 追従戦略

前節の(Ex.-2)は、環境内の各個体の単純な群形成アルゴリズムによって実現できるものである。本論では、(Ex.-2)の状況を実現する各個体の戦略として以下に示す『追従戦略』を提案する。

[定義：追従戦略]あるロボット(ロボットFとする)は他のロボット(ロボットLとする)が、自分自身の前を、自分自身と同じ方向に進んでいるとみなした時に、そのロボットの後を追従する。そうでないときには、ロボットFは独立して自分自身のゴールに向う。

本章では、Lをリーダー(Leader)、Fをフォロワー(Follower)と呼ぶ。一組のリーダーとフォロワーにより局所的な群が形成され、LとFの組が1つ以上集って群全体が構成されると考える。追従戦略の効果は巨視的と微視的の二つの側面から説明できる。巨視的には、「雑踏中で各ロボットが追従戦略を採用することにより、ロボットの流れが生成されるため、各ロボットは容易にゴールに到達できる」と解釈できる。また微視的には、「リーダーがフォロワーに対する一種のナビゲータの役割を果たすため、フォロワーはリーダーを追従するのみで他者を回避することが可能となる。」と説明できる。そこで、本章では、追従戦略を移動ロボットの動作計画問題に適用する。

4. 2 追従戦略の『仮想インピーダンス法』への導入

追従戦略を動作計画に導入するためには、戦略を組入れる枠組みとなる基本アルゴリズムが必要である。そのアルゴリズムは、以下の条件を満たさなければならない、

- ・分散管理型計画を行う、
- ・追従戦略が組込み可能である、

本章では、上記の要求仕様を満たすものとして『仮想インピーダンス法』を採用する。追従戦略を組入れるために、

(a) 仮想インピーダンス法における追従動作のモデル化

(b) 追従するロボットの選択による追従戦略の実現

の2つのステップが必要である。(a)、(b)のそれぞれを以降で論じる。

4. 2. 1 追従動作のモデル化

追従戦略を実現するために、ロボット間のバネ・ダンパモデルとして、回避モデル (avoiding model) と追従モデル (following model) の2種類を用意する。これは、Fig. 3.27の2つのモデルと等価である。そして、個々のロボットがセンシングエリア内の他のロボットとのモデルを実時間で自律的に選択する方式を採用する。リーダーを有さないロボットFに対して、4. 2. 2項に示すルールからリーダーLが選ばれた時には、ロボットFはリーダーLに対して追従モデルを選択し、その結果ロボットFはリーダーLを追従すると考える。逆にロボットFがリーダーLを追従しては目的の行動を達成できないと判断する場合、すなわち、リーダーLが4. 2. 2項に示すルールを満たす場合には、ロボットFはリーダーLとのモデルを追従モデルから回避モデルに変更し、リーダーを追従することをやめる。

仮想インピーダンス法において、回避モデルはFig. 4.2(a)に示す形態を採る。ロボットは他のロボットに拘束されず、斥力のみを受ける。追従モデルはFig. 4.2(b)に示す形態を採る。ロボットFはロボットLにバネ・ダンパで拘束されるため、ロボットFとLが等距離を保つような引力と斥力がロボットFにかかる。その結果、ロボットFがロボットLを追従する動作が実現できる。ロボットの動力学方程式は、3. 1節で示した式とほとんど同じとなる。式(3.6)、(3.7)を、式(4.1)~(4.4)に再度示す。

$$f_{\text{rob}, i, j, x} = \begin{cases} k_{\text{rob}} \left(\frac{1}{od_{ij} - l_{rmin}} - \frac{1}{l_{\text{rob}} - l_{rmin}} \right) \frac{ox_j - x_i}{od_{ij}} + \\ D_{\text{rob}} \frac{l_{\text{rob}} - l_{rmin}}{od_{ij} - l_{rmin}} (\dot{ox}_j - \dot{x}_i) & (od_{ij} \leq l_{\text{rob}}) \\ 0 & (od_{ij} > l_{\text{rob}}) \end{cases} \quad (4.1)$$

$$(4.2)$$

$$f_{\text{rob}, i, j, y} = \begin{cases} k_{\text{rob}} \left(\frac{1}{od_{ij} - l_{rmin}} - \frac{1}{l_{\text{rob}} - l_{rmin}} \right) \frac{oy_j - y_i}{od_{ij}} + \\ D_{\text{rob}} \frac{l_{\text{rob}} - l_{rmin}}{od_{ij} - l_{rmin}} (\dot{oy}_j - \dot{y}_i) & (od_{ij} \leq l_{\text{rob}}) \\ 0 & (od_{ij} > l_{\text{rob}}) \end{cases} \quad (4.3)$$

$$(4.4)$$

ロボットiがロボットjに対して追従モデルを選択している場合、 od_{ij} と l_{rob} の大小関係によらず式(4.1)、(4.3)のみを用いることとする。回避モデルのときには、式(4.1)~(4.4)をそのまま用いる。

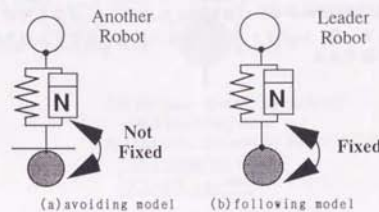


Fig. 4.2 impedance models between robots

4. 2. 2 追従戦略の実現

モデルを遷移するためのルールは、個別のロボットがモデル選択を実時間で実行するように、できるだけ単純であることが望ましい。

モデル遷移のための条件を以下に示す。

(1) 回避モデルから追従モデルへの遷移条件 (Fig. 4.3)

(A) ロボットFの姿勢ベクトルと、ロボットFからロボットLの方向ベクトルのなす角(α)が θ_0 より小さい、

(B) ロボットFから目標位置への方向ベクトルとロボットLへの方向ベクトルのなす角(β)とロボットFの姿勢ベクトルとロボットLの姿勢ベクトルのなす角(γ)との和が θ_0 より小さい、

(C) ロボットFとロボットLを避る他のロボットが存在しない、

の全ての条件が満たされた時に、ロボットFはロボットLをリーダーとみなし、追従モデルへ遷移する。(A)~(C)の条件を満たすロボットが複数台存在した場合には、ロボットFとの距離が最も近いロボットをリーダーとみなす。

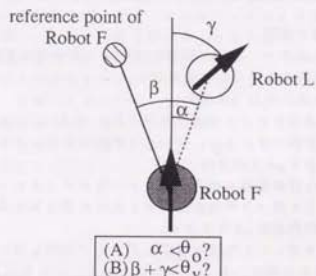
(2) 追従モデルから回避モデルへの遷移条件 (Fig. 4.4)

・ロボットFの目標位置とロボットFの現在位置の距離が l_d より大きい、

の条件が満たされた時に、ロボットFはリーダーロボットへの追従をやめ、回避モデルに戻る。

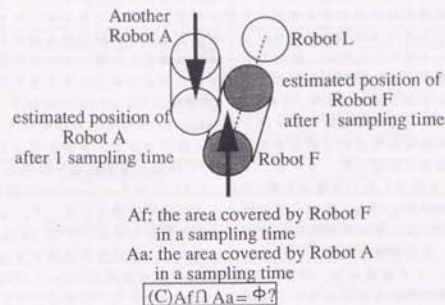
条件(1)は、自分自身の前方に存在し、自分自身の現在進行方向と同一の方向に向っており、自分自身の目標進行方向と同一の方向に向かっており、そのロボットを追従することで他のロボットと干渉しないようなロボットをリーダーロボットの選択基準としている。しきい値 θ_v と比較する値として、 β と γ の和を用いたのは、追従する際の速度ベクトルの方向のずれ γ に対するしきい値を、本来進むべき方向とのずれ β の関数、すなわち $(\theta_v - \beta)$ とすることを意味している。すなわち、本来進むべき方向とのずれ量が大きいほど、追従するための γ に関するしきい値が小さくなり、条件が厳しくなる。

とを意味している。条件(2)は、現在の経路が計画された軌道より離れているかどうかをチェックしている。しきい値の θ_o 、 θ_v 、 L_d はモードを遷移するための設計パラメータである。 θ_o はロボットの視野角に対応し、 θ_v は同じ方向に向かって行くと判断するためのしきい値、 L_d はリーダーLが自分自身が目指す方向と違う方向に進むと判断するしきい値である。



(a) Condition(A), Condition(B)

Fig. 4.3 Conditions for changing from avoiding model to following model
(continue to next page)



(b) Condition(C)

Fig. 4.3 Conditions for changing from avoiding model to following model

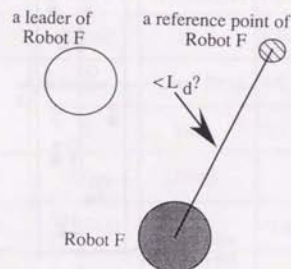


Fig. 4.4 Condition for changing from following model to avoiding model

4. 3 シミュレーションによる追従戦略の有効性の検証

本節では、追従戦略の有効性を検証する。シミュレーション環境として、3章で示した廊下走行シミュレーション (Fig. 4.5) を扱う。前章のシミュレーションでは、廊下のゴールラインに到達したロボットは動作を終了し、スタートラインから新たなロボットが周期的に発生するモデルを用いたが、本章のシミュレーションでは、ある瞬間にゴールラインに到達したロボットが、常に次の瞬間にはスタートラインに戻って再びゴールラインに向い、新たなロボットは発生させないモデルを採用した。これにより、ロボットの行動の秩序生成過程をより明らかに示せると考えられる。ロボットの性能は3章と等価であり、Table 4.1に示す。

各ロボットに付与する設計パラメータをTable 4.2に示す。本章のシミュレーションでは、追従戦略の有効性を検証するために、設計パラメータをロボット毎に共通で固定として適当に設定した。 $p = 0$ 、1台/秒の時の、追従戦略を採用している計画 (Fig. 4.6) と採用しない通常の計画 (Fig. 4.7) のシミュレーション結果の比較を示す。Fig. 4.6(a), Fig. 4.7(a)で各ロボットは初期位置にいる。7.5秒後において、通常の計画では、少々の混雑状況が廊下の約3箇所で生じている (Fig. 4.7(b)) のに対して、追従戦略を採用している計画はロボットの流れが生成できつつある (Fig. 4.6(b))。1.5秒後には、通常の計画が混雑状況が緩和されない (Fig. 4.7(c)) のに対し、追従戦略を採用している計画はロボットの流れが形成されている (Fig. 4.6

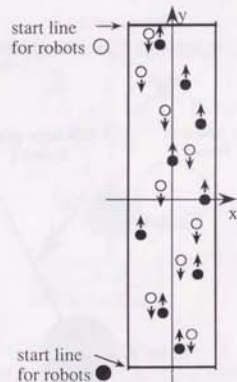


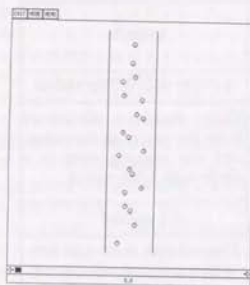
Fig. 4.5 The corridor environment for simulation

Table 4.1 Specification of robots for simulation

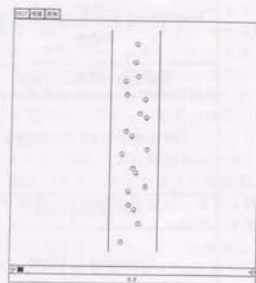
Maximum velocity of the robot	1 m/s
Maximum acceleration of the robot	1 m/s ²
The size of the robot	a circle with 0.25m radius
The sensors on the robot	They measure distances from the center of the robot, and the sensing area is a circle with 3.0m radius
Sampling time	0.1 second
Planning at GMP	Planned path is straight line with a constant velocity of a reference speed
reference speed	1 m/s

Table 4.2 Values of design parameters

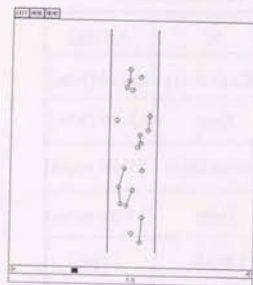
ps1,ps2	0.6	M	1.0 [kg]
pd1,pd2	-0.9	Ktrax,Ktray	1.6x10 [N/m]
dist	0.7[m]	Krob	4.3x10 [N/m]
Lrob	0.9[m]	Dtrax,Dtray	8.0 [N sec/m]
Nrob	2	Drob	7.5 [N sec/m]
Lrmin	0.5[m]	Lblock	1.1[m]
θ_0	$\frac{2}{25} \pi$ [rad]	Ld	3.0x10[m]
θ_v	$\frac{1}{5} \pi$ [rad]		



(a) 0 sec.

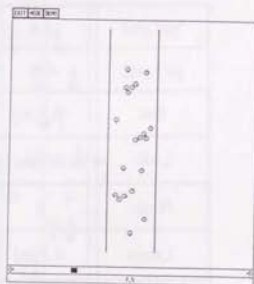


(a) 0 sec.



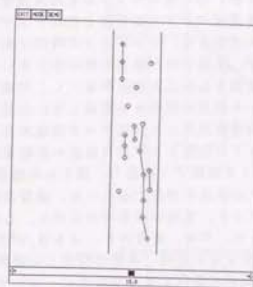
(b) 7.5 sec.

Fig. 4.6 Simulation WITH
"A strategy of
following another robot"
(continue to next page)



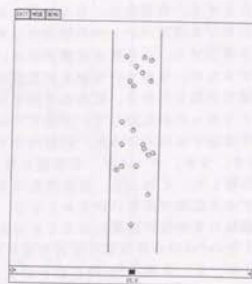
(b) 7.5 sec.

Fig. 4.7 Simulation WITHOUT
"A strategy of
following another robot"
(continue to next page)



(c) 15 sec.

Fig. 4.6 Simulation WITH
"A strategy of
following another robot"



(c) 15 sec.

Fig. 4.7 Simulation WITHOUT
"A strategy of
following another robot"

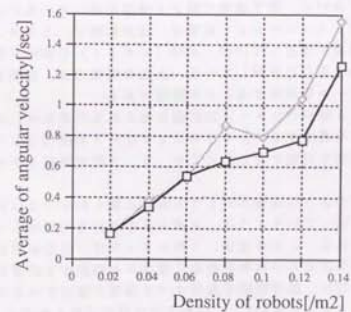


Fig. 4.8 Comparison of angular velocity with variety of robot's density

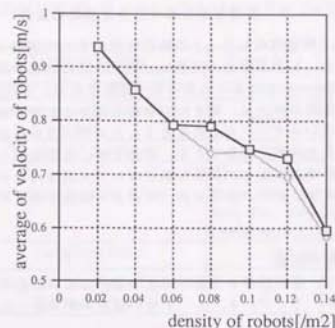
(c) ことがわかる。追従戦略の定量的な評価をするために、本節では3.3節で用いたゴール到達時間とは別に一定時間(30秒)内の各ロボットの平均角速度を用いることとする。角速度は、ある瞬間における速度ベクトルの方向と次のサンプリング時間における速度ベクトルの方向のなす角の大きさを、サンプリング時間で割ることにより導出する。この角速度値が小さい方が、生成された軌道の曲率が小さいことを意味するため、なめらかな動きが生成できて望ましいことがわかる。ここで新たな評価指標を提案したのは、従来の研究ではゴール到達時間のみを評価していた場合がほとんどであったのに対して、本論文では移動速度以外にも評価すべき指標が存在することを主張するためである。環境内のロボットの密度 ρ と平均角速度の比較をFig. 4.8に示す。なお、データは、各密度に対して10種類ずつ用意し、得られた結果の平均を比較した。これより、環境内のロボットの密度が増加するにつれ、通常の戦略を採用すると混雑の度合いが大きくなり、最大25.9%の改善がみられた。これより、追従戦略の有効性が顕著になることがわかった。なお、密度が0.08台/m²の時と0.1台/m²の時の評価値に差異がほとんど生じないのは、本節で固定した設計パラメータが、0.1台/m²の時に各ロボットが適切に動作するようにチューニングしたためと考えられる。

Fig. 4.9に、環境内のロボットの密度 ρ と移動速度のゴール方向成分との比較を示す。ここで平均移動速度は、各ロボットのゴール到達時間に対応するものである。追従戦略を採用した場合、最大6.8%の改善がみられた。この結果からも、追従戦略の有効性が実証された。

以上のシミュレーションの結果、提案した手法の有効性が示された。なお、このシミュレーションにおいて、廊下環境に関する情報を用いていないことに注意されたい。すなわち、上記のシミュレーション結果は、追従戦略が、スクランブル交差点の様に「障害物による拘束が存在しないが、人間(ロボット)の流れが交錯する環境」に対しても有効に機能することを示している。上記の環境では、右側通行などの環境情報を利用した交通ルールを適用することが困難である。

追従戦略を3.3節のパラメータ設計器に組入れるためには、センサ情報ボタンから設計パラメータを導出するニューラルネットワークの出力として、追従戦略の設計パラメータの3種類を付加することにより、3.3節の枠組みをそのまま適用できると考えられる。

追従戦略は、どのような環境に対しても有効に働くわけではない。特に、ロボットがランダムに動き回っているような、ロボットの進行方向のばらつきが大きな環境に対しては不向きである。この事実、「各ロボットは、自分が追従戦略をとるべきかとらざるべきかを現在の自分自身を取巻く環境から判断する必要がある」ことを意味している。この問題は、追従戦略を採用すべき環境と採用すべきでない環境を識別するニューラルネットワークをFig. 3.18のような構造で組入れることにより解決可能であると考えられる。



—□— WITH a strategy of following another robot
—◇— WITHOUT a strategy of following another robot

Fig. 4.8 Comparison of translational velocity with variety of robot's density

4. 4 異質な系における追従戦略の拡張

前節までの議論は、環境内のロボットの動作性能がすべて同質 (homogeneous) である系を想定していた。しかしながら実際は、相互に動作性能の異なるロボットが存在する異質 (heterogeneous) な系の方がより一般的である。この系においては、前節までに示した追従戦略のみでは、望ましい挙動を得ることができない。すなわち、自分自身より速度の遅いロボットを追従することにより、自分自身の目標速度で動作することが不可能となり大変不都合である。本節では、追従戦略と他のロボットを追越す戦略を組合せた戦略を提案し、異質な系においても環境内のロボットの動作に秩序をもたらすことを示す。また、本節では、ロボットの最大速度に関して動作性能の異なる系を想定する。

4. 4. 1 追従戦略の拡張

先に議論したように、自分自身より遅い速度で走行している他のロボットを追従することは好ましくない。そこで、4. 2. 2 項の回避モデルから追従モデルへの遷移条件 (A)~(C) に条件 (D) を付加する。

・回避モデルから追従モデルへの遷移条件

- (A) ロボット F の速度ベクトルと、ロボット F からロボット L の方向ベクトルのなす角 (α) が θ_0 より小さい。
- (B) ロボット F から現時刻の目標位置へ方向ベクトルとロボット L へ方向ベクトルのなす角 (β) の絶対値とロボット F の速度ベクトルとロボット L の速度ベクトルのなす角 (γ) の絶対値との和が θ_0 より小さい。
- (C) ロボット F とロボット L を通る他のロボットが存在しない。
- (D) ロボット F の速度ベクトルの絶対値とロボット L の速度ベクトルの絶対値の比が、 R_{min} 以上 R_{max} 以下である。

上記の条件設定は、条件 (D) を満たさない速度の遅いロボットとは追従モデルをとらず、結果的に追越す動作をとることを意味している。すなわち、追従戦略と追越し戦略を融合したものであり、異質な系において有効に機能することが期待される。

4. 4. 2 シミュレーションによる融合戦略の有効性の検証

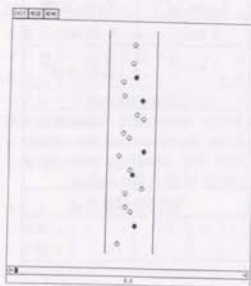
異質な系の動作シミュレーションを行い、追従戦略と追越し戦略を融合した戦略が有効であることを示す。シミュレーション環境として、廊下環境を想定する。ロボットの性能を2種類用意し、Table 4.3 に示す。Type-A が速いロボット、Type-B が遅いロボットである。各ロボットの設計パラメータは、前節のシミュレーションにおける値と同一とし、Table 4.4 に再度示す。ロボットの密度は、0.1 台/ m^2 とした。シミュレーション結果を Fig. 4.10, Fig. 4.11 に示す。Fig. 4.10 は一切の戦略を用いない場合、Fig. 4.11 は、本節で提案した戦略を用いた場合を示している。Fig. 4.10 では、時間が経過してもロボットの混雑状況がほとんど変化していないのに対して、Fig. 4.11 では、各ロボットが追従する相手を適切に選択している場合が多いため、Fig. 4.11(d) に示すように、遅いロボットが走行する列、速いロボットが走行する列

Table 4.3 Specification of robots for simulation of heterogeneous system

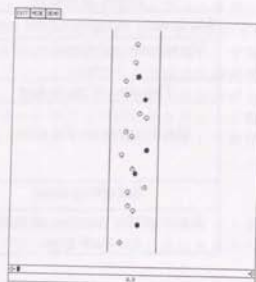
Types of robots	Type-A○	Type-B●
Maximum velocity of the robot	1 m/s	0.25 m/s
Maximum acceleration of the robot	1 m/s ²	0.25 m/s ²
The size of the robot	0.25m radius	
The sensors on the robot	They measure distances from the center of the robot, and the sensing area is a circle with 3.0m radius	
Sampling time	0.1 second	
Ratio of the number of robots for each type	4	1

Table 4.4 Values of design parameters

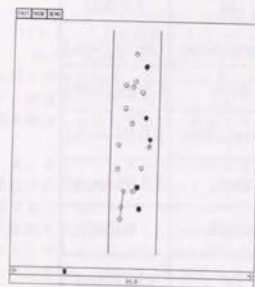
ps1,ps2	0.6	M	1.0 [kg]
pd1,pd2	-0.9	Ktrax,Ktray	1.6x10 [N/m]
dist	0.7[m]	Krob	4.3x10 [N/m]
Lrob	0.9[m]	Dtrax,Dtray	8.0 [N sec/m]
Nrob	2	Drob	7.5 [N sec/m]
Lrmin	0.5[m]	Lblock	1.1[m]
θ_0	$\frac{2}{25}\pi$ [rad]	Ld	1.4x10[m]
θ_v	$\frac{1}{5}\pi$ [rad]	RMIN,RMAX	0.4, 1.6



(a) 0 sec.

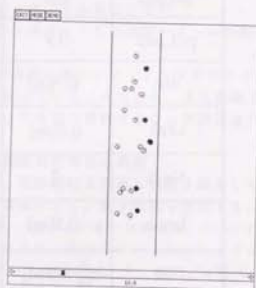


(a) 0 sec.



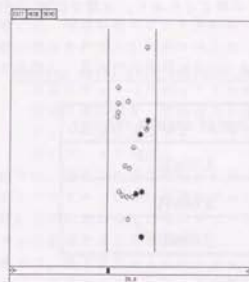
(b) 1.0 sec.

Fig. 4.10 Simulation with
no strategies
(continue to next page)

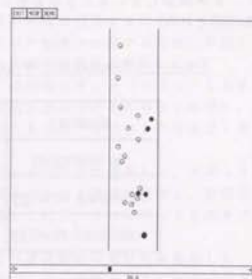


(b) 1.0 sec.

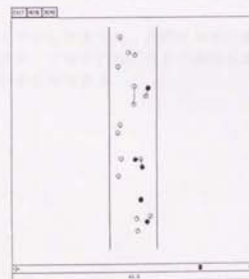
Fig. 4.11 Simulation with
combined strategy
(continue to next page)



(c) 2.0 sec.

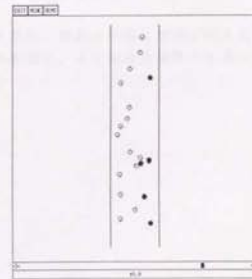


(c) 2.0 sec.



(d) 4.0 sec.

Fig. 4.10 Simulation with
no strategies



(d) 4.0 sec.

Fig. 4.11 Simulation with
combined strategy

が分けられ、ロボットの行動に秩序が生成されていることがわかる。10個のデータに対してシミュレーションを行った結果、平均角速度の大きさの比較をTable 4.5に示す。本節で提案した戦略を用いた場合は、戦略を用いていない場合と比較して、約11.8%減少していることがわかった。このことにより、本節で提案した戦略の有効性が示された。

Table 4.5 Comparison of angular velocity with three methods

strategy	avelage of angular velocity
no strategies	8.96×10^{-2}
following strategy	8.64×10^{-2}
combined strategy	7.90×10^{-2}

4.5 4章のまとめ

本章では、複数移動ロボット系の動作計画問題における秩序生成問題として追従戦略を提案した。

1節では、複数移動ロボット系において混雑状態を回避するために、「個々のロボット間の情報を利用した規範を与えることにより秩序を生成する方法」が適していることを主張し、具体的な規範として、以下に示す追従戦略を提案した。

あるロボット（ロボットFとする）は他のロボット（ロボットLとする）が、自分自身の前を、自分自身と同じ方向に進んでいるとみなした時に、そのロボットの後を追従する。そうでないときには、ロボットFは独立して自分自身のゴールに向う。

2節では、提案した追従戦略を仮想インピーダンス法に適用した。ロボット間のバネ・ダンパモデルとして、回避モデルと追従モデルの2種類を用意し、自分自身の前方に同じ方向に向かっているロボットが存在した時に、そのロボットとのモデルを追従モデルに変更する方式を採用した。

3節では、廊下走行シミュレーションにより追従戦略の有効性を検証した。シミュレーション結果により、ロボットの密度が増大して、混雑しやすい状況になるにつれて追従戦略の有効性が明らかになった。なお評価指標としては、ロボットの平均角速度を用い、最大25.9%の改善がみられた。

4節では、最大速度が相互に異なるロボットが存在する、異質な系においても、追従戦略を拡張することでロボットの行動に秩序を生成することができることを示した。具体的には、追従するための条件に、速度の絶対値が一致するという条件を組入れることで、速度の等しいロボットのみを追従する動作を生成することができた。廊下環境におけるシミュレーションにより提案した戦略の有効性を示した。

ここで示したように、仮想インピーダンス法は、複数の戦略を容易に組入れることができる。このことは、より一般的な環境への適応、より高度な動作の生成への可能性を示すものである。

第5章 複数移動ロボット系における ロボット群の動作計画

- 5.1 ロボット群の概念の重要性和従来の研究
- 5.2 動作計画手法の概念設計
- 5.3 仮想インピーダンス法を用いたロボット群の動作計画の実現
- 5.4 ロボット群の動作計画シミュレーション
- 5.5 5章のまとめ

本章では、作業の性質を考慮したロボット群の動作計画手法を提案する。ロボット群とロボット群の動作計画の定義を第1章で行ったが、ここに再記する。

「ロボット群」とは、共通の目的を有していることを相互に認識しているロボットの集りである。

「ロボット群の動作計画」とは、ロボット群が、相互に近接した群れの状態を保持したまま現在位置から目標位置に移動できるように、ロボット群を構成している個々のロボットが行う動作計画を意味する。

5. 1節で、ロボット群の動作計画においては、群が実現すべき作業が重要であることを述べる。

5. 2節では、作業の性質を明示的に表現できるロボット群の動作計画手法の問題設定を行う。まず、作業の性質を「危険の回避」と「群れの保持」という2つの要求仕様の重要度の比の形で表現できるという立場を採る。そして、作業の性質、すなわち2つの要求仕様の重要度の比を表現する指標として、同じ群を構成するロボットとの解体されにくさを示す「結合度」という概念を提案する。

5. 3節では、5. 2節で示した「結合度」を用いたロボット群の動作計画手法を仮想インピーダンス法のもとに適用することで、具体的な動作計画手法を提案する。

5. 4節では、5. 3節で示した計画手法の有効性をシミュレーションで検証する。「結合度」を設定することにより、「危険の回避」と「群の保持」という2つの要求仕様の重み付けが実現できていることを示す。

Fig. 5.1に論文全体における本章の位置付けを示す。

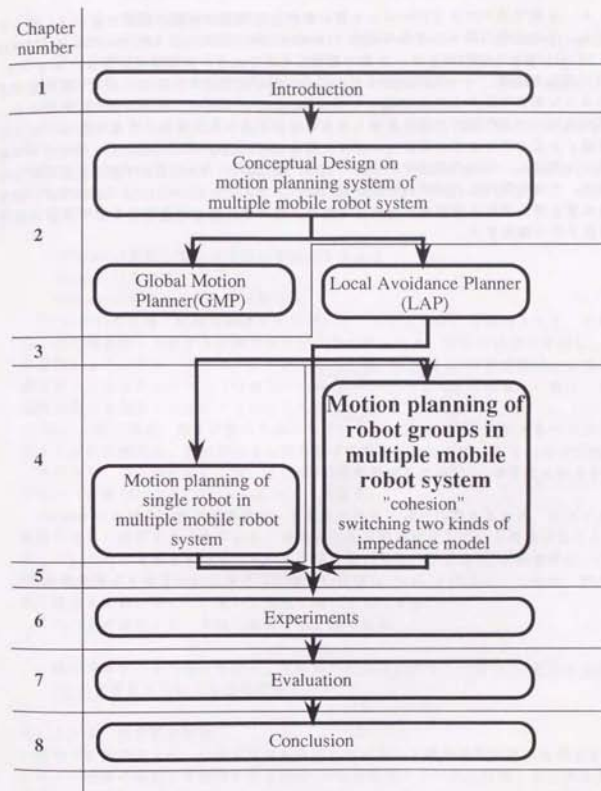


Fig. 5.1 The structure of chapter 5

5. 1 ロボット群の概念の重要性和従来の研究

1. 1節で述べたようにロボット群の動作計画問題は非常に重要である。しかしながら、その分野に関する従来研究〔WANG-P 89b〕〔NOREILS 92〕〔TSUMURA 92〕〔PARKE R 93〕〔KUROSU 93〕等〕は、1. 3節で概観したようにリーダーの選択方法や、ロボット間の通信方法等、一つの群におけるロボット間の相互関係の変化によって群形態がどのような影響を受けるかを取扱ったものがほとんどであり、群の本来の目的である、群が実現すべき作業の内容を反映したものではなかった。また、3章で示したように、仮想インピーダンス法においてロボット群を相互に仮想バネで拘束されたモデルで表現する方法も、同様の理由で不十分である。すなわち、群による作業の実現を考えた場合、その性質に応じて群の挙動が変化するべきであると考えられる。本章では、作業の性質と群の挙動との関係について考察し、作業の性質を考慮したロボット群の動作計画手法を提案する。

5. 2 動作計画手法の概念設計

5. 2. 1 作業と群との関係

本項では、作業と群の挙動の関係について考察する。ロボット群の動作計画において、群が目的地に到達するためには、以下に示す2つの要求仕様が存在する。

(仕様-A)他のロボットや障害物との接近による危険を避ける〔危険の回避〕。

(仕様-B)互に近接した「群れ」の状態を保つ〔群れの保持〕。

仕様(A)は、動作計画問題の基本仕様であり、仕様(B)は、群行動のために必要な仕様である。群の動作計画時には仕様(A)、(B)が互いに競合する場合が存在するが、その際に、どちらを重視するかが問題となる。これは、群が実現すべき作業の性質により決定されるものである。そこで、「実現すべき作業の性質が異なる群と、仕様(A)、(B)の重要度との関係」について、以下に示す自然界の3種類の群を例にとり考察する。

(Group-1)協調して一つの物体を搬送する人達

(Group-2)魚の群

(Group-3)群衆の中を歩く仲間たち

(Group-1)の場合、物体を搬送する作業が完了するまで群れは保持される。すなわち、他の障害物と干渉する危険性が多少大きくなっても、群れの状態を保持し、作業を実現しようとする。よって、この群においては、「仕様(A)の重要度は、仕様(B)の重要度よりはるかに小さい(仕様(B)/仕様(A)→∞)」と考えられる。これは、群の危険に耐える限界が非常に大きいことを意味する。

(Group-2)の場合、通常は群れを保持しているが、察知した危険がある一定以上になると群れを解体し、危険が去ると再構築する戦略が採られている〔PARTRIDGE 82〕。すなわち、この群においては、「仕様(A)の重要度と仕様(B)の重要度に大きな違いがない(仕様(B)/仕様(A)=const.>0)」となる。

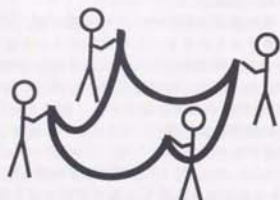
(Group-3)の場合、群の構成員は、群衆内の他の人達を回避するため、群れている状態を容易に解体する。すなわち、構成員同士で群を構成している意識がほとんど存在しないためであると考えられる。この群においては、「仕様(A)の重要度は、仕様(B)の重要度よりはるかに小さい(仕様(B)/仕様(A)→0)」となる。これは、群の危険に耐える限界が著しく小さいと表現することもできる。

これらの考察により、本論では以下の立場を採る。

群が実現すべき作業の性質は、その動作計画における、仕様(A)の重要度と仕様(B)の重要度の比により表現可能である。

5. 2. 2 結合度の提案

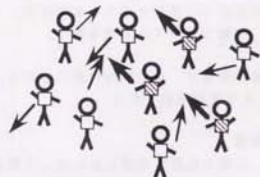
前項での議論により、作業の性質を考慮したロボット群の動作計画を実現するためには、「危険の回避」を目的とする仕様(A)の重要度と「群れの保持」を目的とする仕様(B)の重要度の比が必要であることがわかった。本節では、この比を表現するための指標「結合度」を新たに提案する。



(a) Group-1: People carrying a certain object together



(b) Group-2: A school of fish



(c) Group-3: Several persons who go to the same direction in a crowd

Fig. 5.2 Examples of groups in the natural world

【定義：結合度】同一の群に属するロボットは、群外のロボットやその他の障害物による危険が大きくなった場合に、それを回避するために群れを解体する必要がある。『結合度』とは、そのような外界からの危険に耐える限界を示す定量的な評価指標であるとする。

上記の定義によると、群を構成するロボットは、『結合度』を用いることにより群れを保持するか解体するかを決定する。この結果として、群全体としての挙動が得られる。ある群に対する結合度が大きくなるほど、群れが解体されにくくなる。そのため、『結合度』の小さな群は仕様(A)を重視し、『結合度』の大きな群は仕様(B)を重視することが期待できる。このことより、『結合度』は、(仕様(B)の重要度)/(仕様(A)の重要度)を表現できていると考えられる。

5. 2. 3 本章の動作計画問題の設定

5. 2. 2項で提案した『結合度』を有するロボット群の動作計画問題を設定する。ここで、以下に示す問題設定を行う。

- ・環境内に複数のロボット群が存在しており、それぞれの目的地に向う。
- ・動作開始時には、各ロボット群は、単一の群れを形成している。
- ・各ロボット群には、作業に応じて群の結合度があらかじめ設定されており、その構成員であるロボットは、自分自身が属する群の結合度を認識している。
- ・各ロボットは同じ群に属している他のロボットを識別可能である。
- ・動作計画は、ロボット単位に行う。

上記の仮定は、複数移動ロボット系に複数の作業が与えられた状況を想定したものであり、妥当である。

本論では、Fig. 5.3に示す方法で動作計画を実現する。すなわち、各ロボットは自分自身を群外の物体による危険の大きさ(index of danger, 以下『危険度』と呼ぶ)と、群の有する『結合度』を用いて群れを解体するかどうかを実時間で判断する。個別のロボットの動作の集積として、結合度に応じて群形態が変化するロボット群の動作計画が実現される。

5. 2. 4 期待される効果

5. 2. 2項で述べたように、ある群に対する『結合度』を小さくすると、動作計画時に危険を回避する要求が重視されるため、仕様(A)の重要度が相対的に大きくなることを期待される。逆に『結合度』を大きくすると、群れが保持する要求が重視されるため、仕様(B)の重要度が増すことが期待される。すなわち、『結合度』の設計により仕様(A)と仕様(B)の重みづけが実現できれば、作業の性質を結合度というパラメータで表現した動作計画手法が構築できたとと言える。また、このとき、5. 2. 2項の例で示した群における結合度を、

$$(\text{群-1の結合度}) > (\text{群-2の結合度}) > (\text{群-3の結合度})$$

に設計することにより、(群-1)～(群-3)の挙動を統一した枠組みの中で扱うことができると考えられる。

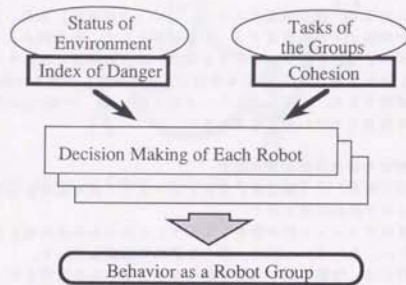


Fig. 5.3 Concept of the proposed method

5. 3 『仮想インピーダンス法』を用いたロボット群の動作計画の実現

前章で示した動作計画手法の概念を実現する枠組みの手法として、仮想インピーダンス法を導入する。この際、群れのモデル化と、結合度に応じた動作計画の実現方法が問題となるため、以下で述べる。

5. 3. 1 ロボット群の仮想バネ・ダンパモデル化

本章では、ロボット間の仮想バネ・ダンパモデルとしてFig. 5.4に示す2種類を用意する。このモデルはFig. 3.27のモデルと等価である。Fig. 5.4(a)のモデルは、2台のロボットが互いに仮想バネ自然長に拘束されたモデルであり、『結合モデル (connection model)』と呼ぶ。このモデルでは、ロボット同士がバネ自然長内に近づいた場合には斥力がかかり、バネ自然長より遠ざかった場合には引力がかかる。複数台のロボットが相互に結合モデルを採ることにより、群れの状態を表現することができる。この場合の動力学方程式は、3. 1節で示した式とほとんど同じになる。式(3.6), (3.7)を式(5.1)~(5.4)に再度示す。

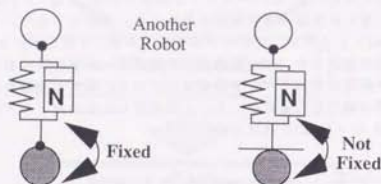
$$f_{\text{rob}, i, j, x} = \begin{cases} k_{\text{rob}} \left(\frac{1}{od_{ij} - l_{\text{rmin}}} - \frac{1}{l_{\text{rob}} - l_{\text{rmin}}} \right) \frac{ox_j - x_i}{od_{ij}} + \\ D_{\text{rob}} \frac{l_{\text{rob}} - l_{\text{rmin}}}{od_{ij} - l_{\text{rmin}}} (\dot{ox}_j - \dot{x}_i) & (od_{ij} \leq l_{\text{rob}}) \\ 0 & (od_{ij} > l_{\text{rob}}) \end{cases} \quad (5.1)$$

$$f_{\text{rob}, i, j, y} = \begin{cases} k_{\text{rob}} \left(\frac{1}{od_{ij} - l_{\text{rmin}}} - \frac{1}{l_{\text{rob}} - l_{\text{rmin}}} \right) \frac{oy_j - y_i}{od_{ij}} + \\ D_{\text{rob}} \frac{l_{\text{rob}} - l_{\text{rmin}}}{od_{ij} - l_{\text{rmin}}} (\dot{oy}_j - \dot{y}_i) & (od_{ij} \leq l_{\text{rob}}) \\ 0 & (od_{ij} > l_{\text{rob}}) \end{cases} \quad (5.2)$$

ロボット*i*がロボット*j*に対して結合モデルを選択する場合、 od_{ij} と l_{rob} の大小関係によらず、式(5.1), (5.3)を用いることとする。一方、Fig. 5.4(b)のモデルはロボット同士がバネ自然長内に近づいた場合だけに反力を及ぼすモデルであり、相互に独立した動作を実現するため『独立モデル (independent model)』と呼ぶ。この場合の動力学方程式は、式(5.1)~(5.4)をそのまま用いる。通常、同一の群に属するロボット同士が結合モデルを採ることにより群れの状態を保持し、異なった群のロボットとは独立モデルを採る。危険が迫った時に、その大きさに応じて、各ロボットが同一の群を構成する近傍の他のロボットとの仮想バネ・ダンパモデルを実時間で選択することにより群れの形態が変化する。次節で具体的な選択方法について述べる。

5. 3. 3 ロボット間モデルの選択方法

各ロボットが他のロボットとの仮想バネ・ダンパモデルを選択するために、他の物体からの危険の大きさを表す危険度を定量的に評価する必要がある。危険度は、各ロ



(a) Connection Model (b) Independent Model

Fig. 5.4 Impedance Models between Robots

ロボットが自律的にモデルを選択できるために、各ロボット近傍のローカルな情報のみを用いて評価できるものが望ましい。

そこで、本章では、危険度を以下の様に設計する。すなわち、ロボット i から見たロボット j との間の危険度を、ロボット i が同じ群に属さない他の物体——障害物、他のロボット——から受ける仮想外力のうち、ロボット i とロボット j を分離する方向に作用する外力の和として表現する (Fig. 5.5)。他の物体がロボットの近くに存在しても、そこからの仮想力がロボット i と j を引き離す方向に働いていない場合 (Fig. 5.6) は、危険度は 0 となる。これは、この状況下においてロボット i と j のモデルを結合モデルから独立モデルに遷移させても、ロボット i は他の物体との近接状態を回避できないと考えられるからである。式 (5.5) に危険度の計算方法を示す。

$$\begin{aligned} \text{Index of danger} = & \sum_k \begin{cases} F_{\text{rob},k} \cos \theta_{\text{rob},k} & (\cos \theta_{\text{rob},k} \geq 0) \\ 0 & (\cos \theta_{\text{rob},k} < 0) \end{cases} \\ & + \sum_h \begin{cases} F_{\text{obs},h} \cos \theta_{\text{obs},h} & (\cos \theta_{\text{obs},h} \geq 0) \\ 0 & (\cos \theta_{\text{obs},h} < 0) \end{cases} \quad (5.5) \end{aligned}$$

各ロボットの仮想バネ・ダンパモデルの選択方法を Fig. 5.7 に示す。このとき、『結合度』は、外部環境から計算される危険度に対して結合モデルを維持するためのしきい値であると解釈できる。

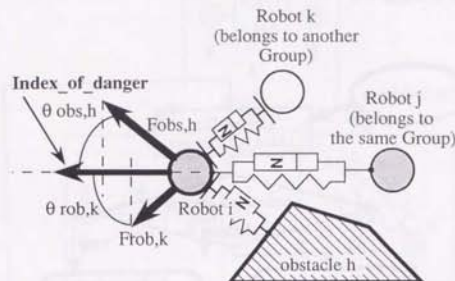


Fig. 5.5 Expression of "index of danger"

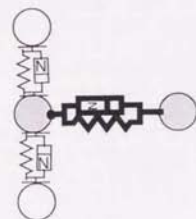


Fig. 5.6 The Case that index of danger is equal to zero

以上の設計により、2章で示した「危険を避けたい」要求仕様と、「群れて行動したい」要求仕様の二者の重みづけが可能となる。すなわち、結合度が大きく設定されている場合、多少の危険を受けても群れを保ったまま行動する (Fig. 5.8(a)) のに対して、結合度が小さく設定されている場合、群れが解体されやすくなる代りに、被る危険が小さくなる (Fig. 5.8(b)) といった行動が期待される。この事を次節のシミュレーションで検証する。

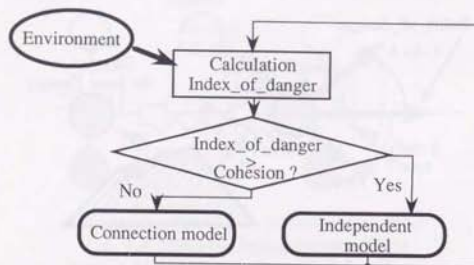
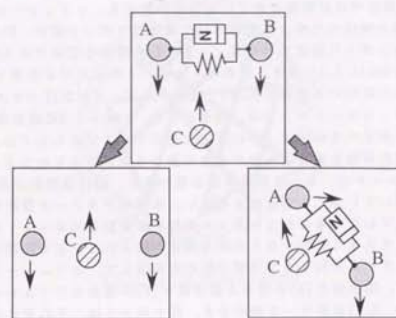


Fig. 5.7 Method for selecting impedance models



A,B : the same group
C : another group

(a) Small Cohesion

(b) Large Cohesion

Fig. 5.8 Motion depending on Cohesion

5. 4. ロボット群の動作計画シミュレーション

5. 4. 1 シミュレーションの条件

前節で提案した手法の有効性を検証する。具体的には、結合度に対応して仕様(A)と仕様(B)の重みづけが実現できているかを検証する。シミュレーション環境として、3. 4. 3項と同様の円周上に配置された複数のロボット群が、円領域を横切って同じ円周上のランダムに設定されたゴールに向かう環境を想定する(Fig. 5.9)。ロボットの性能をTable 5.1に示す。これは、3章、4章における性能と同じである。シミュレーションに用いる定数をTable 5.2に、ロボットの設計パラメータをTable 5.3に示す。本シミュレーションでは、簡単のため、Table 5.2の結合度、Table 5.3の設計パラメータがすべてのロボットに共通に適用されているものとした。このシミュレーションの目的が結合度による群の挙動の違いを検証するものであるため、Table 5.3の設計パラメータは、3. 3節の手法を使わずに、試行錯誤的に得られた適当な値に固定して用いていることに注意されたい。本章のパラメータ設計に関する定量的な知見は得られていないものの、ロボット間のバネ定数をロボット・目標位置間のバネ定数と比較して大きく設計すると良好な特性が得られることを確認している。

10種類のデータに対して7種類の結合度を用いて、シミュレーションを行った。代表例として、結合度0[N]の場合と結合度 ∞ [N]の場合のシミュレーション結果をFig. 5.10, Fig. 5.11に示す。0秒のとき、各ロボットは、それぞれの初期位置に存在する(Fig. 5.10(a), Fig. 5.11(a))。1.5秒後、結合度0の方は、ロボット間の結合モデルの多くが独立モデルに遷移しており、群れが解体されつつある(Fig. 5.10(b))のに対し、結合度 ∞ の方は、群れを保持している(Fig. 5.11(b))。4.5秒後には、結合度0の方は13台のロボットが中央に残っている(Fig. 5.10(c))のに対して、結合度 ∞ の方は、16台のロボットが中央に残っている(Fig. 5.11(c))。6.6秒後には、すべてのロボットが目的地に到達している(Fig. 5.10(d), Fig. 5.11(d))。これより、群の結合度が大きい場合、群同士の相互干渉が大きくなることが定性的にわかる。

群形態に関する評価指標は、単純で一般的なのものが望ましい。本章では、群外の物体との近接による危険の大きさの評価指標として、「群以外のロボットからの仮想外力の大きさの絶対値の和」を用いる。この指標は、近接している他の物体が多いロボットほど高い値を示すため、危険の評価指標として妥当である。また、群の形態の保持状況の評価指標として、「クラスタ数」を用いることとする。ここで、クラスタとは、同一の群内で結合モデルで拘束されているロボットの集りを意味する。クラスタとは、結合モデルにより「群れ」ているロボットを意味しているため、評価指標として適当であると考えられる。Fig. 5.12に全ての群のクラスタ数の平均の時間毎の推移の比較を、Fig. 5.13にすべてのロボットにかかる仮想外力の和の時間毎の推移の比較を示す。この2つの図からの以下の事がわかる。

- ・仮想外力の大きさの時間毎の和の最大値は、結合度0の場合が、結合度 ∞ の場合の約70.1%となっている。すなわち、結合度0の方が、危険を回避できている。
- ・クラスタ数の時間平均は、結合度 ∞ の場合が、結合度0の場合の約62.1%とな

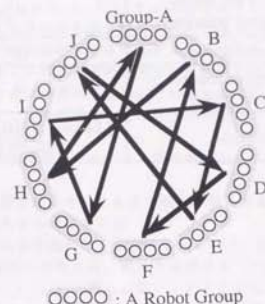


Fig. 5.9 Simulation environment

Table 5.1 Capacity of robots

Maximum velocity of the robot	1 m/s
Maximum acceleration of the robot	1 m/s ²
The size of the robot	a circle with 0.25m radius
The sensors on the robot	They measure distances from the center of the robot, and the sensing area is a circle with 3.0m radius
Sampling time	0.1 second
Planning at GMP	Planned path is straight line with a constant velocity of a reference speed
reference speed	1 m/s

Table 5.2 Condition for the simulation

Radius of the circle for start/goal positions	7.5 [m]
Number of groups	10
Number of robots for one group	4
cohesion	7 kinds (0, 10, 20, 30, 40, 50, ∞)[N]

Table 5.3 Parameters for robots

M	1.0 [kg]	Lrob	0.9[m]
Ktrax, Ktray	4.0 [N/m]	Krob	3.24×10^2 [N/m]
Dtrax, Dtray	4.0 [N sec/m]	Drob	3.6×10 [N sec/m]

っている。すなわち、結合度 ∞ の方が群れを保持できている。

結合度とクラスタ数の関係をFig. 5.14に、群の結合度と仮想外力との関係をFig. 5.15に示す。Fig. 5.15において、仮想外力の大きさは結合度が0のときの値を1として正規化している。Fig. 5.14, Fig. 5.15の結果より、群の結合度が増加させると、危険が増大する代わりに群れの形態を保持できていることがわかる。ここで、結合度が50 [N]の場合が ∞ [N]の場合に大きな差がみられないのは、仮想バネ定数 k_{pos} 等に応じて他の物体から与えられる外力の大きさには限界が存在するためであると考えられる。仮想外力の大きさに作業の性質に応じた結合度を設定することにより、危険の回避と群れの保持の2つの要求仕様の重み付けが可能であることが確認され、本章の手法の有効性が実証された。

上記で定義した結合度を用いたロボット群の例として、以下のようなものが考えられる。

[例1]宝探し作業、地図生成等をするために隊列を組んで動作するロボット群

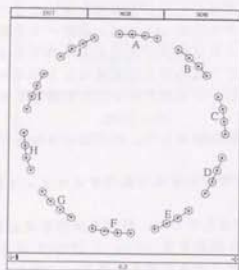
[例2]センサ負荷を分担するロボット群

[例1]では、未知の領域を洩れなく探索するためには、隊列を保持する必要がある。探索途中で障害物が存在した時には、群れを保持する(Fig. 5.16(a))よりも群れを分けて探索した(Fig. 5.16(b))方が効率が上がると考えられる。このとき、結合度を適切にチューニングすることにより、作業性の良好な群行動を実現できる。

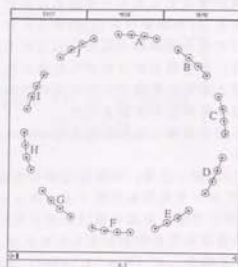
[例2]では、一つの群を構成するロボットの台数が多いほど、各ロボットのセンシングの負荷を減少させることができる。危険が迫ってきて、単一の群れを保持すると危険が回避できないと判断した時には、群れを分割する戦略をとることが妥当である。群れを分割してセンシング負荷が増大することによる各ロボットの損失と、危険を回避することによる各ロボットの利得を比較することで、結合度が決定される。

上記の例から考察できるように、群れを保持することによる利得と危険を回避することによる利得が均衡するように結合度を設計し、それに基づいてロボット群を制御することで、最適な群行動が実現できるものと考えられる。

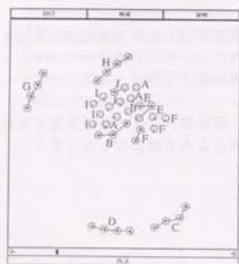
ロボット群による作業実現を考えた場合、結合度ではない他の指標も必要となる可能性がある。これに関しては、7章の評価のところで論じることとする。



(a) 0 sec.

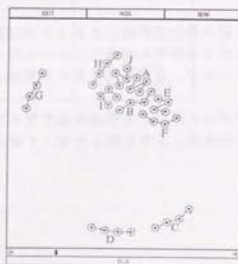


(a) 0 sec.



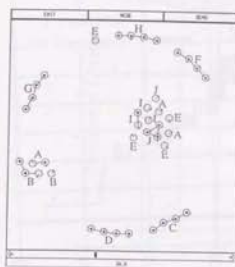
(b) 1.5 sec.

Fig. 5.10 Simulation with
Cohesion = 0 [N]
(continue to next page)

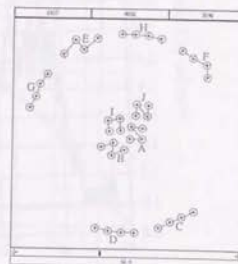


(b) 1.5 sec.

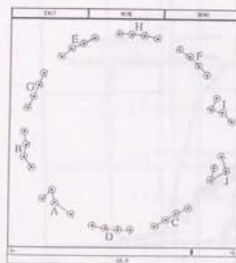
Fig. 5.11 Simulation with
Cohesion = ∞ [N]
(continue to next page)



(c) 3.0 sec.

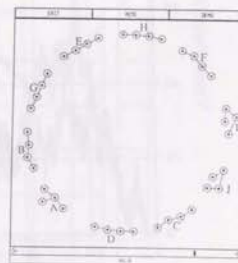


(c) 3.0 sec.



(d) 6.6 sec.

Fig. 5.10 Simulation with
Cohesion = 0 [N]



(d) 6.6 sec.

Fig. 5.11 Simulation with
Cohesion = ∞ [N]

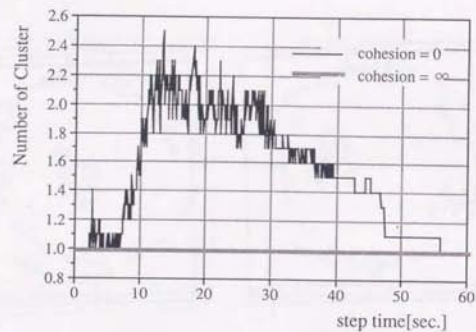


Fig. 5.12 Comparison of number of cluster between Cohesion = 0 and ∞

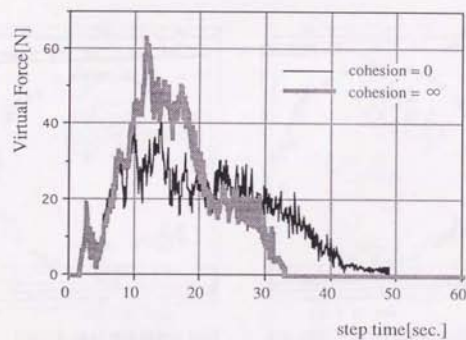


Fig. 5.13 Comparison of virtual force between Cohesion = 0 and ∞

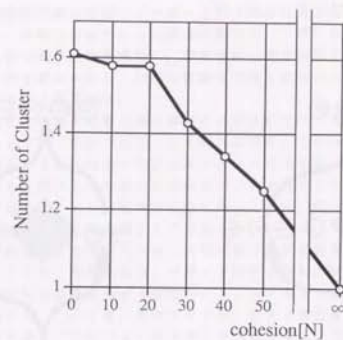


Fig. 5.14 Relationship between cohesion and the number of cluster

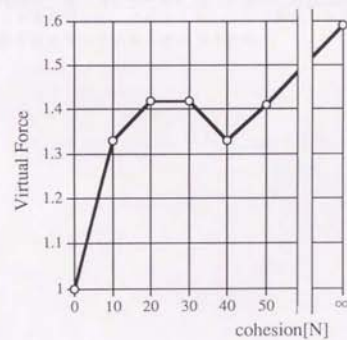
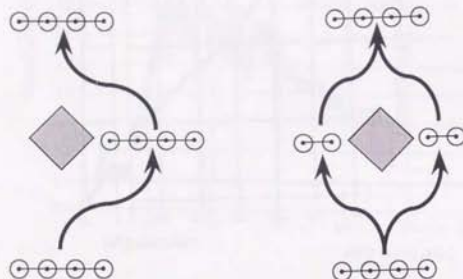


Fig. 5.15 Relationship between cohesion and virtual force



(a) Behavior by keeping groups (b) Behavior by dividing groups

Fig. 5.16 Behavior of map making robot groups

5.5 5章のまとめ

本章では、作業の性質を考慮したロボット群の動作計画手法を提案した。

1節では、1.3節で概観を示した従来の研究と、3章における仮想インピーダンス法のロボット群の動作計画手法から、群の挙動に群が実現すべき作業の性質が反映されていないことを明らかにし、作業の性質を考慮したロボット群の動作計画手法の確立が必要であることを述べた。

2節では、まず、作業の性質と群の挙動との関係について考察した。群が動作する際の要求仕様として、「危険の回避」と「群れの保持」という2つが存在するが、そのどちらを重視するかが作業の性質を表すという立場を採った。次に、作業の性質を表す、すなわち上記の2つの要求仕様のどちらを重視するかの重み付けをするパラメータとして『結合度』という概念を提案した。これは、同じ群を構成しているロボット同士の群の解体しにくさを表すものである。

3節では、『結合度』を用いたロボット群の動作計画手法を仮想インピーダンス法にインプリメントした。具体的には、ロボット同士が群れた状態を表す「結合モデル」と群れを解体している状態を表す「独立モデル」をロボット間の仮想バネ・ダンパモデルとして用意する。このとき、各ロボットが、群の『結合度』と他のロボットからの危険の大きさをを用いて上記の2つのモデルを選択することにより、ロボット群の動作計画を実現する方法を採用した。

4節では、3節で示した動作計画手法の有効性をシミュレーションで検証した。円形領域をロボット群が互いに交錯するシミュレーション環境において、『結合度』に応じて、「危険の回避」と「群れの保持」という2つの要求仕様の重み付けがきちんとなされていることを実証した。これにより、ここで提案した手法が、作業の性質を考慮した動作計画手法となっていることが示された。

1. 実験装置の概要

2. 実験装置の構成

3. 実験装置の動作

4. 実験装置の性能

5. 実験装置の応用

第6章 実機を用いた検証実験

- 6.1 実験装置の概略
- 6.2 実験各要素の構成
- 6.3 看板による位置・姿勢計測システム
- 6.4 コントローラの設計
- 6.5 実験と考察
- 6.6 6章のまとめ

本章では、前章までで提案した手法の有効性を実験を用いた実験により検証する。具体的には、『車輪機構を有するロボットの動作計画を行う際の、パラメータ設定のための評価関数の設計方法』を考察することに焦点を絞り、4、5章で提案した手法の直接的な検証実験は行わない。

6.1節では、実験装置全体の概要について述べる。

6.2節では、実験各要素の構成について述べる。

6.3節では、複数移動ロボット系の実験において一番の問題となる移動ロボット相互の位置、姿勢計測システムとして点光源から構成される看板システムを提案する。

6.4節では、動作計画器で計算された出力を実際のロボットの車輪駆動指令と結びつけるコントローラの設計方法と設計論について述べる。

6.5節では、実験を行う。具体的には、2台のロボットの相互回避実験を行う。これより、車輪機構を有するロボットの動作計画に『仮想インピーダンス法』を適用する際に、パラメータ設計時の評価関数をどのように設定するべきかを考察する。前章までで提案した動作計画手法の有効性を検証する。

6.6節では、6章のまとめを述べる。

Fig. 6.1に、論文全体における本章の位置付けを示す。

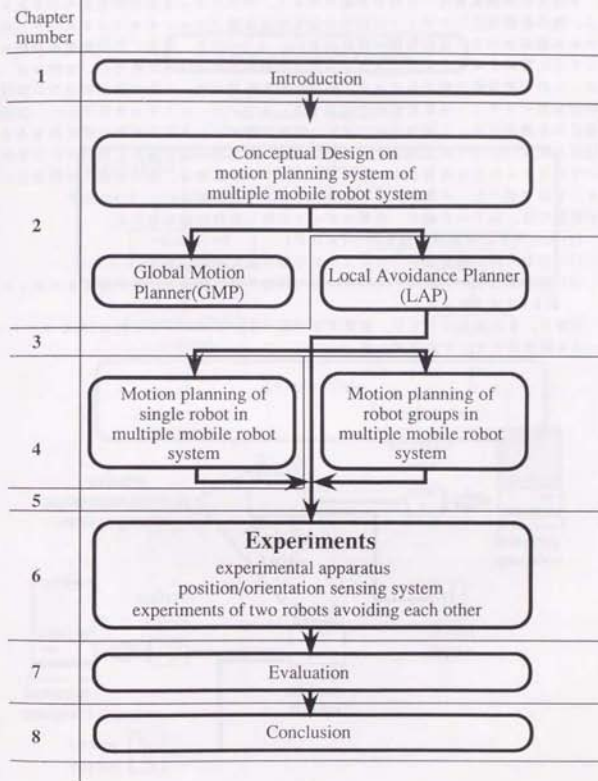


Fig. 6.1 The structure of chapter 6

6.1 実験装置の概略

本論文の実験装置は、2台の移動ロボット、他のロボットを測定するセンサシステム、動作計画を行いロボットに指令を送る処理装置（パーソナルコンピュータ）の3つから構成される。実験装置の構成図をFig. 6.2に示す。本来、処理装置を移動ロボット上に搭載することによりロボットの自律性を高めるべきであるが、本論では、ロボットの可搬重量の観点などから、処理装置を移動ロボットの作業領域の外に設置し、処理装置・ロボット相互を有線で結合することにより、システムを実現した。実験装置各部の概要を6.2節で示す。また、複数移動ロボット系の実験の際に最も大きな問題となるロボット相互の位置・速度計測は、複数個の点光源から構成された看板センサシステムにより実現した。この看板システムの詳細は、他の装置とは節を分けて6.3節で述べる。本実験におけるソフトウェア構成図をFig. 6.3に示す。

本実験では、以下の手順で、実験ロボットに対し動作計画を与える。

- (1) オフラインで大域計画を行う(GMP)
- (2) (1)の大域計画を基に、オンラインで動作指令を生成する(LAP)
- (3) (2)の出力をオンラインでロボットへの指令値に変換し、その指令をロボットに送る(コントローラ)

GMP, LAPについては、前章までで述べているので、ここでは(3)のコントローラの設計法を6.4節で述べる。

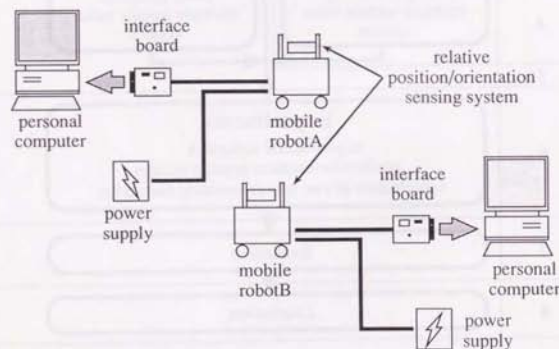


Fig. 6.2 Experimental system

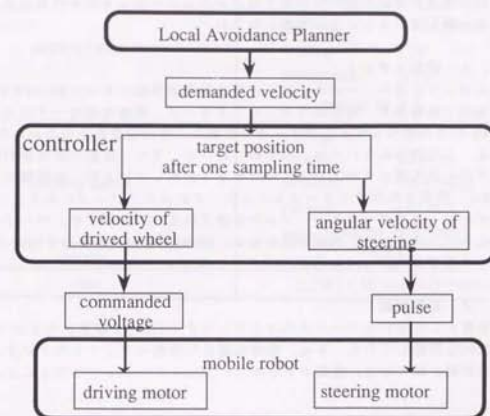


Fig. 6.3 Software architecture

6. 2 実験要素の構成

本節では、実験装置を構成する要素を説明する。なお、本章の実験ではパラメータ設計方法の確立に主眼を置いているため、シミュレーションにおけるロボットのスペックとは一致していないことに注意されたい。

6. 2. 1 移動ロボット

移動ロボットにはニューテクノロジー振興財団製移動ロボットMM-03を用いる。MM-03は1操向1駆動型で、操向はステッピングモータ、駆動はDCモータにより行われる。また、MM-03は外界センサとしてカメラを搭載し、自己位置表示のために看板を搭載している。この詳細については、次の節で述べる。また、正面に赤外線対物センサ、先端部下面に赤外線ラインレセセンサを3組有しているが、本実験では用いない。

MM-03は、搭載されたインターフェイスボードにより、パーソナルコンピュータ側のインターフェイスボードとケーブルで接続される。これにより、パーソナルコンピュータにより、ロボットの制御が行われる。MM-03ロボットの諸元をTable 6.1に、ロボットの概要をFig. 6.4に示す。

6. 2. 2 処理装置

処理装置として用いたパーソナルコンピュータは25MHzで駆動されるインテル社製486SX型CPUを搭載している。また、処理装置には移動ロボットとのインターフェイスボードが搭載されており、移動ロボットとコミュニケーションを行うことが可能である。

6. 2. 3 駆動電源

移動ロボットの操向、駆動用各モータ、コントロールボード等に電源を供給するため、外部電源を用意した。この外部電源は、移動体とケーブルにより接続され、移動ロボットに電源を供給する。

Table 6.1 Specification of MM-03 type mobile robot

steering motor	step angle	0.9°
	current	0.4A/phase
	voltage	12V
	resistance	30Ω
geared by 2:1		
driving motor	voltage	12V
	torque	30 g-cm
	velocity	3,600 rpm
	current	185 mA
	max. velocity	0.05[m/s]
weight	3.0 kg	
size	L 280×W 165×H 190[mm]	

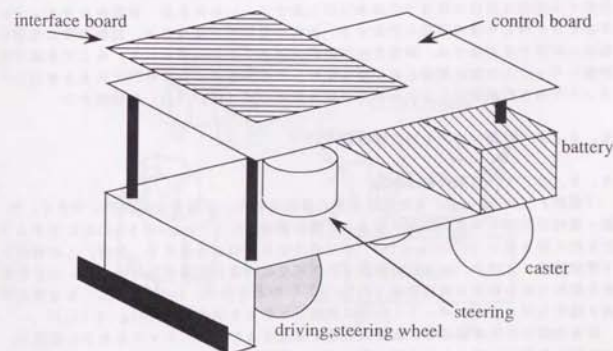


Fig. 6.4 Schematic view of MM-03 type mobile robot

6. 3 看板による位置・姿勢計測システム

複数移動ロボット系の位置・姿勢計測に関する従来の研究は、環境内に固定されたランドマーク〔斉 91〕〔藤原 90〕や、レンジファインダ〔津村 88〕を用いることによりロボットの自己位置・姿勢を同定する研究が大半であった。複数台のロボットの相対位置・姿勢を相互に計測に関する研究は少なく〔津村 92〕、ロボット間通信を用いた方法〔YUTA 92〕が見られる程度であった。本節では、複数移動ロボット系における相互位置・姿勢を光源とカメラを用いて実時間計測するシステムを提案する〔ARAI 93b〕〔新井 94〕。実験を用いた測定実験によりシステムの有効性を検証する。

6. 3. 1 システムの基本構成

本研究で提案するシステムの適用環境として、

- ①環境内には、複数台のロボットのみが存在する。
- ②通常の明るさの室内である。
- ③床面は平坦で、ロボットは滑らかに走行する。よって、ロボットの倒れは無視できる。
- ④各移動ロボットは自律分散制御を行う。

を仮定する。これらの仮定は、移動ロボットの走行環境の一般性を損うものではない。なお、一般的な障害物、すなわち光源等を備えていない障害物の検知ならびに回避のためには、別途センサシステムを準備する必要があるが、本論では扱わない。

位置・姿勢計測は、能動的方法と受動的方法とに大別される。計測側が光や音波を投射する前者は複雑な環境での高速計測に適するといわれるが、複数台のロボットが存在すると相互干渉の問題が発生する。後者の受動的方法のうち、被測定物の視覚的形狀に依存する方法では、環境変化に脆弱であることが予想される。そこで本論では、移動ロボット上の被計測側に点光源を配置した看板を掲げ、その形と大きさを別のロボットである計測側の CCD カメラで読み取る方式 (Fig. 6.5) を提案する。

6. 3. 2 計測システムの詳細設計

6. 3. 2. 1 看板の基本設計

「看板」(signboard)とは被計測側に搭載された、計測側から識別しやすく、位置・姿勢を計測しやすい物体である。看板の構造として、Fig. 6.5(a)(b)に示すように 2 枚を開き角 α ($0^\circ < \alpha < 180^\circ$) で組合せたものを採用する。以降、この構造を「看板要素」と呼ぶ。 α は計測範囲と計測精度の両者に影響を及ぼす。単一の看板要素を観察可能な範囲の理論値は $(360 - \alpha)^\circ$ であるので、 $90^\circ < \alpha < 180^\circ$ なる看板要素 2 組を対向させてロボット 1 台用の看板システムを構成する (Fig. 6.5(c))。

画像処理の負荷を軽減し、実時間処理を実現するために、カメラを水平に設置し、点光源 P_1 をカメラのレンズ光軸と同一高さに設置する。この結果、すべての移動ロボットの点光源 P_{11} は必ず水平面上同一高さとなりレンズ光軸上で観察される。この輝

点 P_{11} を取掛かりとし、その上下に他の点光源が存在することで看板を認識する。

1 台のロボットの前後判別は、Fig. 6.6(b)でわかるように点光源 P_1 の上方に点光源 P_0 を追加し、 P_0 が存在すれば看板要素 B で後、なければ看板要素 A で前であるとす。看板要素上には 4 つの光源が Fig. 6.6(a)のように一定距離 L で配置されている。

6. 3. 2. 2 画像処理アルゴリズム

複数個の看板を撮影した画像上での点光源配置の例を Fig. 6.7 に示す。この画像情報から、複数台のロボットの位置・姿勢を計算するアルゴリズムを以下に示す。

(a) レンズ光軸上を走査して輝点を見つける。n 個検出されたら、それらは皆看板要素の第 1 光源 $P_{11} \sim P_{n1}$ と仮定する。

(b) すべての P_{11} に対して以下の処理を行う。

(b-1) P_{11} の下方を画面上で Lf/d_{\min} だけ探索して、 $P_{11}P_{12}$ から計測対象までの距離 d_1 を求める。ここで、 L は輝点 P_1, P_2, P_3, P_4 間の距離、 f はカメラの焦点距離、 d_{\min} は計測範囲の最短距離である。

(b-2) P_{11} の上方を画面上で Lof/d_1 だけ走査して P_{10} を探す。 L_0 は輝点 P_0, P_1 間の距離である。

(b-3) P_{12} の左右を Lf/d_1 だけ走査して、最初に見つけた輝点をそれぞれ P_{13}, P_{14} とする。

なお、探索している輝点を見えなかった場合には、元の輝点はノイズと判断して、続く処理を停止する。

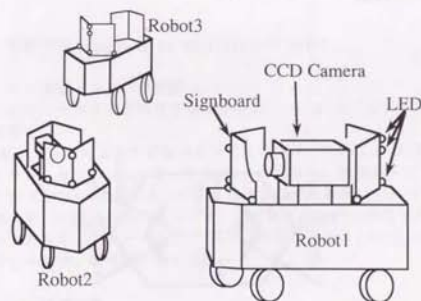
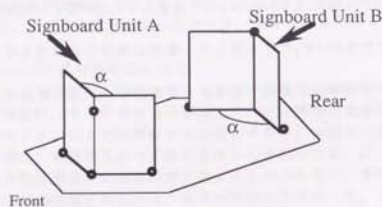
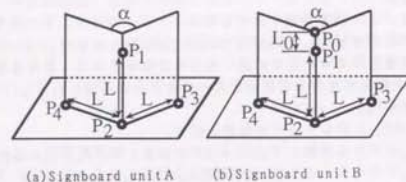
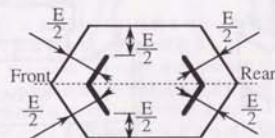


Fig. 6.5 Schematic view of the proposed system



(c) An overall view of signboard



(d) A Top view of signboard

Fig. 6.6 Schematic view of signboard

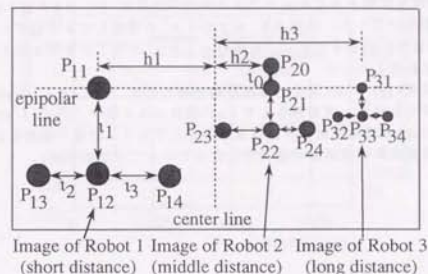


Fig. 6.7 An example of images of CCD camera

上記の処理により求めた点光源の位置情報から、ロボット間の相対位置・姿勢を導出できる。具体的な計算方法については、文献[ARA1 93b][新井 94]を参考にされたい。

6. 3. 4 検証実験

6. 3. 4. 1 実験システムの概要

前項までで設計した手法の有効性を実証するため、Fig. 6.8に示す計測システムを試作し、実験を行った。

MM-03型移動ロボットの上に看板およびCCDカメラ($f=12\text{mm}$)を搭載した。看板の寸法を $L=0.07\text{m}$ 、 $L_0=0.02\text{m}$ とした。また、誤差解析により、 $\alpha=14.0^\circ$ とした([ARA1 93b][新井 94])。点光源には高輝度LED(7-7製GLSLR60)を用いた。これより、一般の照明下において、点光源を安定的に検出できた。看板情報をCCDカメラを介してフレームメモリ(512×256画素、8bit)に取込み、パーソナルコンピュータで、実時間計測を行う。

6. 3. 4. 2 計測実験

試作システムの計測精度と計測時間を評価するために、以下の3種類の計測実験を行った。

(E-1) 精度を評価する1台のロボットの計測実験
 (E-2) 時間を評価する複数台ロボットの計測実験

(a) 1台のロボットの計測実験

実験は、以下の2種類について行った。

- (a) 位置精度を計測するためにカメラとの相対位置を 0.5mより3mまで0.5m刻みで、看板を計測した。姿勢は0°および45°に固定して20回ずつ計測した。
- (b) 姿勢精度を計測するために、姿勢を-180°から180°まで15°回転ずつさせた看板を計測した。

実験結果をFig. 6.9に示す。姿勢計測に関しては、1m、2m、3mの場合とも同様な結果が得られたが、代表例として2mの場合のみを示す。Fig. 6.9の結果より、位置に関して5%以内の誤差で計測結果が得られており、位置・姿勢計測値の誤差の平均値、標準偏差が理論誤差の範囲内に収まっていることがわかる。

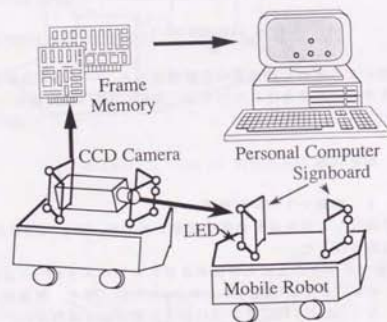
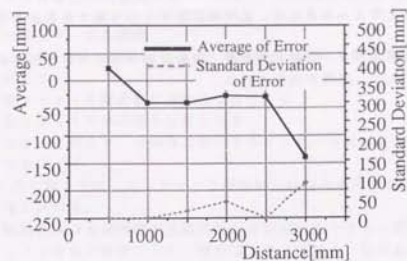
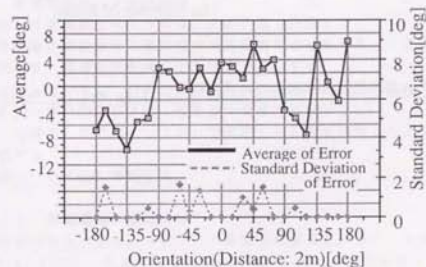


Fig. 6.8 Experimental system



(a) Accuracy of position



(b) Accuracy of posture

Fig. 6.9 Experimental results on (E-1)

(b) 複数台ロボットの計測結果

2台または3台のロボットを任意の位置・姿勢に固定させた時の各ロボット毎の位置・姿勢を計測した。計測時間の結果をFig. 6.10に示す。この結果より、1台のロボットで0.2秒強の計測時間、 n 台のロボットが存在した時には $(0.12 + 0.09n)$ 秒で計測できるが、1台のカメラの視野内に入るロボットはただか5~6台であると考えられるため、実時間処理が十分可能であると考えられる。

これらの結果により、提案した位置・姿勢計測システムの有効性が実証された。この計測システムは、実験に利用される。

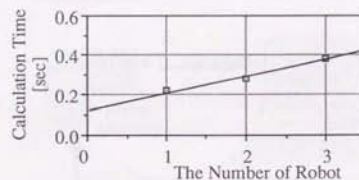


Fig. 6.10 Relationship between calculation time and the number of robot

6.4 コントローラ的设计

本章で述べた実験システムにおいて、LAPにより出力された目標速度指令を実機ロボットにおいて実現するために、コントローラを作成した。本節では、その構成および設計について述べる。

6.4.1 コントローラの構成

本実験においては、MM-03型移動ロボットの機構が1操向1駆動型であるため、LAPの出力に応じて

- 移動ロボットの駆動輪に速度指令を与える
- 移動ロボットの操向輪を回転させる

が必要となる。これより、本実験におけるコントローラはFig. 6.11に示す構成をとるものとする。

Fig. 6.11に示したように、コントローラが車輪への指令速度を導出する際に、以下の問題点が考えられる。

- ・1操向1駆動方式の車輪機構の逆運動学を解析的に解くことが困難である、
- ・移動ロボットの性能の限界により、駆動輪への速度指令、操向輪への回転指令共に離散値としてしか与えることができない、

そこで、本論文では、LAPの出力速度指令の範囲内で、駆動輪への速度指令、操向輪への回転指令を与えるテーブルをオフラインであらかじめ作成し、コントローラはLAPの出力に応じてテーブルを参照し、移動ロボットへの指令を出力するものとした。

以下、逆運動学テーブルの構成と設計について述べる。

6.4.2 逆運動学テーブル的设计

本実験で用いた逆運動学テーブルは、次の手順で作成した。

- (1) ある駆動輪の速度指令と操向輪の角速度指令の組を与え、一定時間走行させた後の位置を、順運動学を解くことにより算出する
- (2) 実機ロボットに対し、いくつかの駆動輪の速度指令と操向輪の角速度指令の組を与え、一定時間走行させ、移動後の座標を測定する
- (3) (1)で算出した位置と、(2)で測定した位置を比較した上で、テーブルを作成する

以下、手順について順に説明する。

6.4.2.1 車輪機構の順運動学

車輪機構の解析をする際に、ロボットの座標系の原点を、ロボットの従動輪の中間点とした(Fig. 6.12)。なお、このロボットを中心とした座標系を「移動ロボット座標系」、移動領域の座標系を「基準座標系」と呼ぶものとする。

移動ロボット座標系において、姿勢角 θ 、操向輪切り角 ϕ の状態から、操向輪角速度 ω 、駆動輪速度 v で t 秒間運動を続けた場合を考える。動作前のロボットの位置を

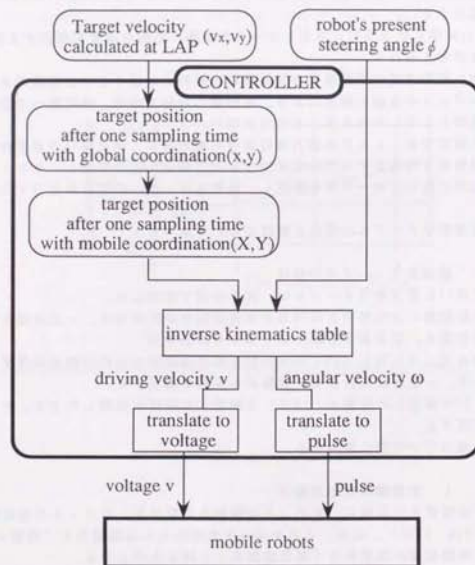


Fig. 8.11 Architecture of controller

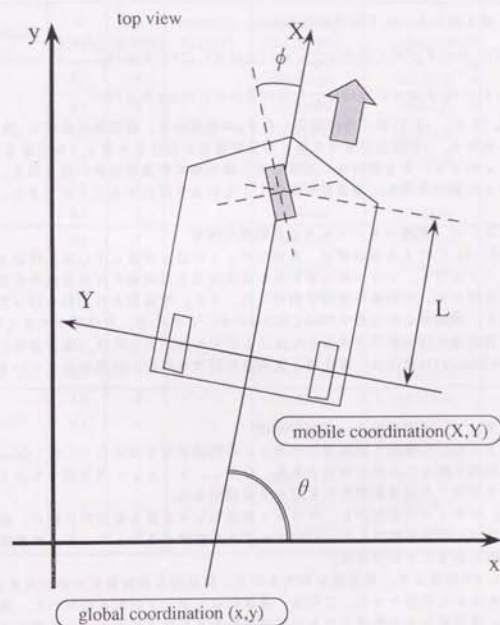


Fig. 6.12 Coordinate system for a mobile robot

(x_0, y_0)、姿勢角を θ_0 、操向輪切り角を ϕ_0 とし、 t 秒後のロボットの位置を($x(t), y(t)$)、姿勢角を $\theta(t)$ 、操向輪切り角を $\phi(t)$ とすると、次の式が成り立つ。

$$x(t) = v \int_0^t \cos \phi \cos \theta dt' \quad (6.1)$$

$$y(t) = v \int_0^t \cos \phi \sin \theta dt' \quad (6.2)$$

$$\theta(t) = \frac{v}{L} \int_0^t \sin \phi dt' \quad (6.3)$$

$$\phi(t) - \phi_0 = \int_0^t \omega dt' = \omega t \quad (6.4)$$

上記の式を解くと、以下の式が得られる。

$$X(t) = v \int_0^t \cos(\phi_0 + \omega t') \cos\left[\frac{v}{L\omega} \{-\cos(\phi_0 + \omega t') + \cos \phi_0\}\right] dt' \quad (6.5)$$

$$Y(t) = v \int_0^t \cos(\phi_0 + \omega t') \sin\left[\frac{v}{L\omega} \{-\cos(\phi_0 + \omega t') + \cos \phi_0\}\right] dt' \quad (6.6)$$

この式(6.5)、(6.6)は、操向輪切り角 ϕ_0 の状態から、駆動輪に速度 v 、操向輪の角速度 ω を与え、 t 秒間走行させた後の作業座標系におけるロボットの位置を意味している。これより、ある駆動輪の速度指令と操向輪の角速度指令の組を与え、一定時間走行させた後の位置を、順運動学を解くことにより算出することができる。

6. 4. 2. 2 実機ロボットによる移動量の測定

式(6.5)、(6.6)による理論値が、実機ロボットの値と合致しているか検証するため、実機ロボットに対し、いくつかの駆動輪の速度指令と操向輪の角速度指令の組を与え、一定時間走行させ、移動後の座標を測定した。また、理論値との比較を行った。

理論値と、実測値との比較をTable 6.2に示す。これより、角速度が大きくなると、算出値と実測値の間の誤差が大きくなるのが分かる。とりわけ、速度指令 $5[\text{cm/s}]$ 、角速度が $9[\text{deg/s}]$ の場合は、算出値と実測値の間で3割近い誤差が生じていることがわかる。

6. 4. 2. 3 逆運動学テーブルの作成

前述のように、1操向1駆動型のロボットの順運動学を求めた。しかし、この逆運動学を解析的に解くことは不可能である。しかし、6. 4. 1項で述べたように、コントローラではこの逆運動学の解を求める必要がある。

そこで、ロボットの諸元から、ロボットの出力しうる値を離散的に集め、順運動学に対応したテーブルを作成した。このテーブルを参照することにより、逆運動学に対する近似解を得ることができる。

Table 6.2の結果より、角速度 ω が大きいと、算出値と実測値との差が大きいう傾向があることが分かった。これは、具体的なロボットの動きにおいて、角速度が大きいと、算出値による軌道よりも回転半径の大きな軌道をとることを意味する。

そこで、出力される角速度指令を低く抑え、理論値と、Table 6.2に示した実測値

がほぼ合致する領域について、目標移動位置に対する駆動輪速度指令、操向輪速度指令、および予想移動位置を配列に格納し、逆運動学テーブルとした。逆運動学テーブルに関するパラメータをTable 6.3に示す。

すなわち、作業座標系における目標移動位置を、移動目標点座標きざみ幅で正規化した値を入力すると、単位速度指令、単位角速度指令の値に正規化された出力が得られるテーブルが作成された。

Table 6.2 Comparison of theoretical value and measured value

initial ϕ	v [cm/s]	ω [deg/s]	running time[s]	calculated value[cm]	measured value[cm]	error [cm]
0	5.0	9	10	(24.6,11.9)	(37.21,5)	(12.4,9.6)
	2.5	9		(14.9,3.65)	(20.0,4.5)	(5.1,0.85)
	5.0	6		(31.6,16.8)	(44.0,14.5)	(12.4,2.3)
	2.5	6		(19.3,5.01)	(21.0,3.5)	(1.7,1.51)
	5.0	3		(43.6,13.6)	(49.5,9.5)	(5.9,4.1)
-22.5	2.5	3	10	(23.4,3.6)	(22.5,2.0)	(0.9,1.6)
	5.0	9		(38.8,2.95)	(42.0,-10.0)	(3.2,12.95)
	2.5	9		(20.4,0.88)	(20.0,-2.0)	(0.4,2.88)
	5.0	6		(46.1,-6.07)	(41.0,-2.9)	(5.1,3.17)
	2.5	6		(23.5,-1.53)	(20.5,-1.5)	(3.0,0.03)
22.5	5.0	3	10	(43.2,-20.7)	(36.5,-25.5)	(6.7,4.8)
	2.5	3		(23.8,-5.39)	(19.5,-5.0)	(4.3,0.39)
	5.0	9		(16.8,8.47)	(13.0,31.5)	(3.8,23.03)
	2.5	9		(9.05,1.42)	(13.0,6.0)	(3.95,4.58)
	5.0	6		(14.6,15.2)	(19.5,22.5)	(4.9,6.7)
	2.5	6	10	(12.2,5.62)	(15.5,7.0)	(3.3,1.38)
	5.0	3		(18.2,23.7)	(22.0,28.5)	(3.8,4.8)
	2.5	3		(16.4,8.31)	(17.5,8.0)	(1.1,0.31)

Table 6.3 Parameters for making inverse kinematics table

maximum velocity	0.05[m/s]
step velocity	0.0025[m/s]
maximum angular velocity	5.85[deg/s]
step angular velocity	0.45[deg/s]
step of target position	0.0025[m]

6.4 実験と考察

前章までで示した動作計画手法の有効性を、前節までで示した実験装置を用いて検証する。具体的には、以下の事を検証する。

前章までの議論では、ロボットをすべて質点モデルとみなして、解析・パラメータ設計を行ってきたが、実際の移動ロボットでは、車輪機構を有するものが多数を占める。前章まで行った質点モデルの解析結果を、車輪機構の移動ロボットの制御に有効に適用するためには、質点モデルの挙動が、実際の車輪モデルの挙動をできるだけ誤差を少なく近似できていることが望まれる。しかしながら、全方向移動を仮定している LAP において計算される「ロボットへの速度入力」を、コントローラを介して「車輪への速度入力」に変換する場合に、車輪で移動が困難な方向への速度指令や車輪のすべりが原因となって、質点モデルの軌道と車輪モデルの軌道間に多大な誤差が生じる危険性がある。このような誤差は、解析・パラメータ設計時に車輪機構を考慮する必要があることを示しており、望ましくない。そこで、本実験では、3.3節で示したランダム多スタート局所探索法を適用する際の評価関数の設定方法と、設定されたパラメータを用いて駆動した場合の軌道誤差——質点モデルにおける軌道と車輪モデルにおける軌道の誤差——との関係を明らかにする。

実験では、2台のロボットの相互回避実験を『仮想インピーダンス法』を用いて実現する。

実験時のロボットのサンプリング時間は、看板センサを用いる関係上 0.5秒とした。相互回避実験の実験条件を Table 6.4 に、ロボットの設計パラメータを Table 6.5 に示す。設計パラメータは、3.3節のランダム多スタート局所探索法を適用することによって得られた最適近似解を用いた。Table 6.5 の (1) のパラメータは、(両方のロボットの到達時間の和) を評価関数とした時の解であり、(2) のパラメータは、(両方のロボットの角速度 $\times 1.0 +$ 両方のロボットの到達時間の和 $\times 0.1$) を評価関数とした時の解である。ここで、角速度は、「あるロボットのあるサンプリング時刻 t における速度ベクトルと $t+1$ における速度ベクトルのなす角」を、すべてのロボットとすべてのサンプリング時刻毎に総和を採ることにより近似的に求めた。(2) のパラメータの評価関数に両方のロボットの到達時間の和の項を入れたのは、その項を導入しないと、まったく移動しないのが最適であるという結果が得られてしまうからである。また、GMP では、初期位置から目標位置までを目標時間で移動する等速直線運動を生成した。これは、前章までのシミュレーションと同様である。

Fig. 6.13, Fig. 6.14 に質点モデルにおけるシミュレーション結果を示す。Fig. 6.13 が (1) のパラメータを用いた場合、Fig. 6.14 が (2) のパラメータを用いた場合である。Fig. 6.13, Fig. 6.14 において、ロボット中心から出ている線分は、その瞬間におけるロボットの速度の方向を示している。ゴール到達まで、Fig. 6.13 では、6.0 秒、Fig. 6.14 では、6.7.5 秒かかった。Fig. 6.13 と Fig. 6.14 の結果を比較すると以下のことがわかる。

Table 6.4 Conditions for the experiment

Robot	Start Position(m)	Goal Position(m)	Demanded Times to Reach(sec)
Robot-A	(0.1,-1.5)	(0.1, 1.5)	60.0
Robot-B	(-0.1, 1.5)	(-0.1,-1.5)	60.0
Maximum velocity of the robot		0.05m/s	
Maximum acceleration of the robot		0.2m/s ²	
The size of the robot		a circle with 0.25m radius	
The sensors on the robot		They measure distances from the center of the robot, and the sensing area is a circle with 3.0m radius	
Sampling time		0.5 second	

Table 6.5 Design parameters for robots

parameters	set (1)	set (2)
ps1,ps2	0.0	0.7
pd1,pd2	-0.9	0.4
dist	2.0	0.9
Lrob	2.8	1.7
Nrob	1	
Ktrax,Ktray	4.00	0.36
Dtrax,Dtray	4.00	1.20
Krob	4.44	0.12
Drob	2.35	0.40
Lrmin	0.50	0.50
Lblock	1.36	0.80

・目標時間を最適にするパラメータを用いた場合は、大回りをした回避行動をとる。一方、角速度を重視するパラメータを用いた場合は、回避経路を小さく行動する。これは、3章で示したように仮想インピーダンス法ではロボットの最大速度をロボット間距離に応じて調整しているため、目標地点に速く到達するためには、ロボット間の距離を大きくとりつつ迂回する必要があるからである。

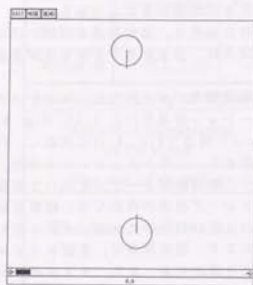
・目標時間を最適にするパラメータを用いた場合は、速度指令の方向のばらつきが大きいのに対して、角速度を重視した場合は、速度指令の方向がほぼ連続的に変化していることがわかる。

次に、ロボットを質点モデルではなく車輪機構モデルとみなし、センシングエリアの限界(±1.5 deg.)を考慮したシミュレーション結果をFig. 6.15, Fig. 6.16に示す。Fig. 6.15は、到達時間を評価関数にとった場合でFig. 6.13に対応し、Fig. 6.16は角速度を重視した場合でFig. 6.14に対応する。このシミュレーションは、コントロールを、ロボットの車輪機構を前節で述べた逆運動学テーブルを用いて駆動するように書き換えたものであり、車輪誤差、センシング誤差が存在しない理想的な状況における軌道を表現している。Fig. 6.15, Fig. 6.16においてロボットから出ている屈型はロボットのセンシングエリアを示す。ここで、速度情報は、看板システムにより得られた相対位置情報の差分をとることにより導出する。また、カメラのセンシングエリアから逸脱した近傍のロボットの挙動は、センシングエリア内で計測した過去5ステップの相対運動の平均をとり、それを持続するという推定をした。これにより、カメラのセンシングエリアの狭さを補償することができる。ゴール到達時間は、Fig. 6.15が11.9秒、Fig. 6.16が15.1秒となっており、角速度を重視したパラメータの方が短いことがわかる。以下のことが考察される。

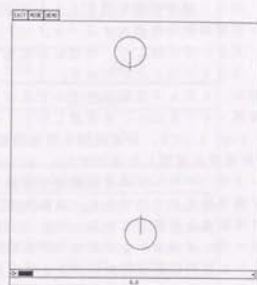
- ・質点モデルと比較して、到達時間が2倍近くかかっているのは、車輪機構では、車輪方向の移動をするために時間がかかるためであると考えられる。
- ・速度ベクトルの方向に関するばらつきが大きい場合、車輪を介して動作させると軌道に大きなうねりを生じやすくなる。このため、時間が余計にかかる。
- ・上記の2つの特性は、逆運動学テーブルを精度良く作成し直す、コントロールのサンプリング時間を短くすることである程度は対処可能であるが、速度方向のばらつきが大きい場合には、車輪がすべりを生じやすくなり、好ましくない。

以上の考察により、質点モデルで設計したパラメータを車輪機構において実現する際には、評価パラメータとして、角速度を考慮した方が望ましいことがわかった。実験では、角速度を重視した設計パラメータを用いることとする。

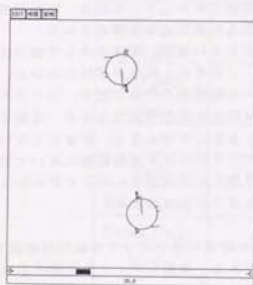
実験結果をFig. 6.17に示す。ここに示した結果は各ロボットの絶対座標値ではなく各ロボットの自己位置推定に基づくものである。移動ロボットの近傍の小円は、一方のロボットが計測・推定したロボットの相対位置を示している。小円の中に×印がない場合は看板システムを用いて計測したことを、×印がある場合はセンシングエリア外にあるためロボットの位置を推定したことを意味する。概ねシミュレーションと類似した軌跡が得られたが、ゴールに到達した時間がシミュレーションの時よりも早いという結果が得られた。これは、相対位置の計測時の横軸方向の相対位置が実際より



(a) 0 sec.

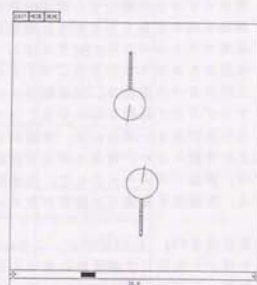


(a) 0 sec.



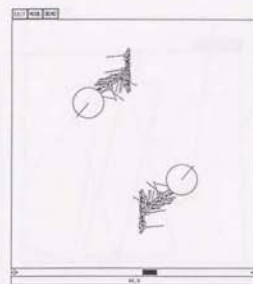
(b) 2.0 sec.

Fig. 6.13 Simulation results for
parameter set (1) with point model
(continue to next page)

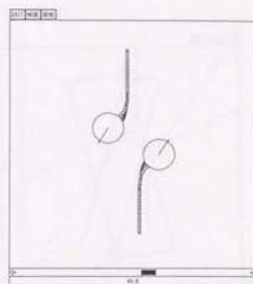


(b) 2.0 sec.

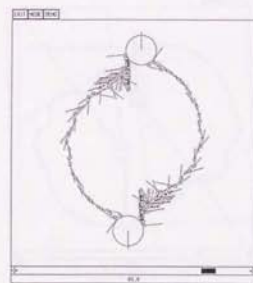
Fig. 6.14 Simulation results for
parameter set (2) with point model
(continue to next page)



(c) 4.0 sec.

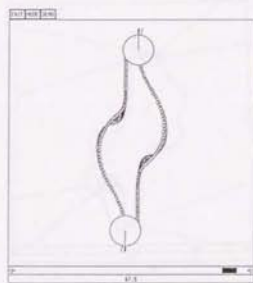


(c) 4.0 sec.



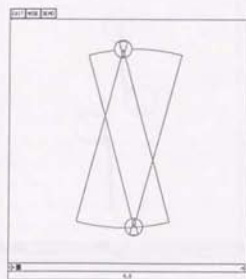
(d) 6.0 sec.

Fig. 6.13 Simulation results for
parameter set (1) with point model

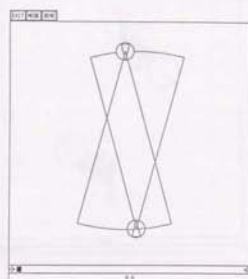


(d) 6.75 sec.

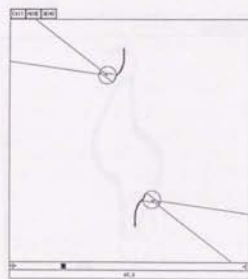
Fig. 6.14 Simulation results for
parameter set (2) with point model



(a) 0 sec.

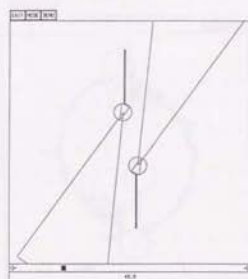


(a) 0 sec.



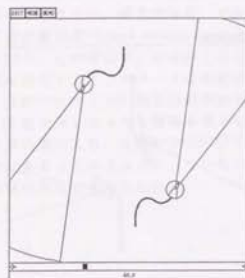
(b) 4.0 sec.

Fig. 6.15 Simulation results for
parameter set (1) with car-like model
(continue to next page)

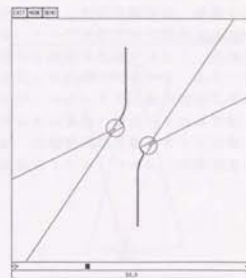


(b) 4.0 sec.

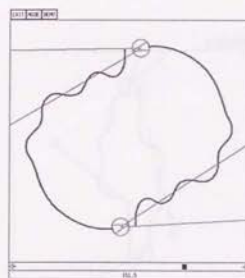
Fig. 6.16 Simulation results for
parameter set (2) with car-like model
(continue to next page)



(c) 6.0 sec.

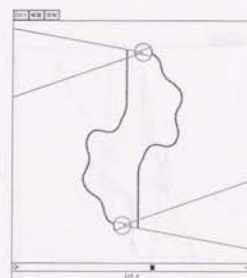


(c) 6.0 sec.



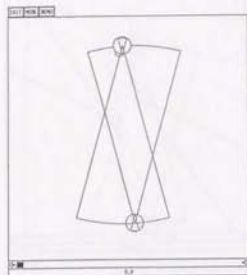
(d) 151.5 sec.

Fig. 6.15 Simulation results for
parameter set (1) with car-like model

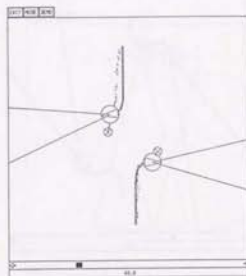


(d) 119 sec.

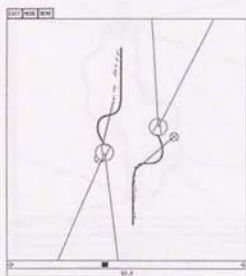
Fig. 6.16 Simulation results for
parameter set (2) with car-like model



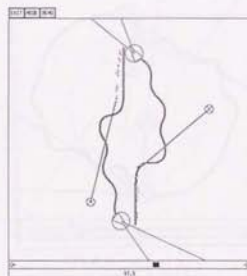
(a) 0 sec.



(b) 4.0 sec.



(c) 6.0 sec.



(d) 9.3. 5 sec.

Fig. 6.17 Experimental results

りも大きく計画されたため、相互の回避行動がシミュレーションの時よりも速く開始されたことが原因であると考えられる。

終端地点における各ロボットの位置決め誤差は、図の下から上に向っているロボットが横方向に0.40 m、縦方向に0.20 m、上から下へ向っているロボットが横方向に0.05 m、縦方向に0.10 mであった。これらの誤差は、車輪の滑りによる自己位置同定 (dead reckoning) のずれと、テーブル作成時の誤差によるものと考えられる。この誤差は、なるべく角速度が小さい起動を生成すること、環境内に絶対位置を保证するランドマークを配置することにより改善可能である。また、本実験では、2台のロボットの相互回避実験を行ったが、ロボットの台数が増加した場合、4章、5章で示したような戦略を導入した場合にも本実験の議論をそのまま拡張できる。

実験結果により、仮想インピーダンス法が、車輪機構型移動ロボットの動作計画に有効であることが示された。この時の評価関数としては、ロボットの角速度を重視する必要があることを示した。

6. 5 6章のまとめ

本章では、前章までで提案した理論の有効性を実機を用いた実験により検証した。

1節では、実験装置の概要について述べた。装置は、移動ロボット、センサシステム、コンピュータの3種類から構成することを述べた。

2節では、実験各部の詳細について述べた。

3節では、複数移動ロボット系の相対位置・姿勢相互計測システムとして、実時間処理の可能なシステムを提案した。具体的には、

- ・各ロボットの計測部として『カメラ』と『看板』からなるシステム構成を提案した。
- ・誤差解析により看板の設計を行い、画像情報からの位置・姿勢計測アルゴリズムを提案した。
- ・検証実験により相対位置に関して手法の有効性を実証した。

4節では、実機を用いた実験を行った。パラメータ設計の際の評価関数の選定方法に関して考察した。2台のロボットの相互回避実験を行い、コントローラの性能、車輪の滑り等を考慮すると、評価関数には、ロボットの角速度を重視する必要があることを示した。

第7章 評価

7. 1 従来研究との定量的比較

7. 2 本システム全体の評価

7. 3 7章のまとめ

本章では、本論文で提案した手法の評価を行う。

7. 1 節では、従来研究との定量的な比較を行う。具体的には、[NOBORIO 92]で提案されている手法と『仮想インピーダンス法』とを、3. 4 節の広場走行シミュレーションにより比較する。これより、『仮想インピーダンス法』の長所・短所を明らかにする。

7. 2 節では、『仮想インピーダンス法』の、その他の項目に関する評価を行う。車輪機構を有するロボットへの適用可能性、作業計画システムとの統合可能性等について評価した。

Fig. 7.1に論文全体における本章の位置付けを示す。

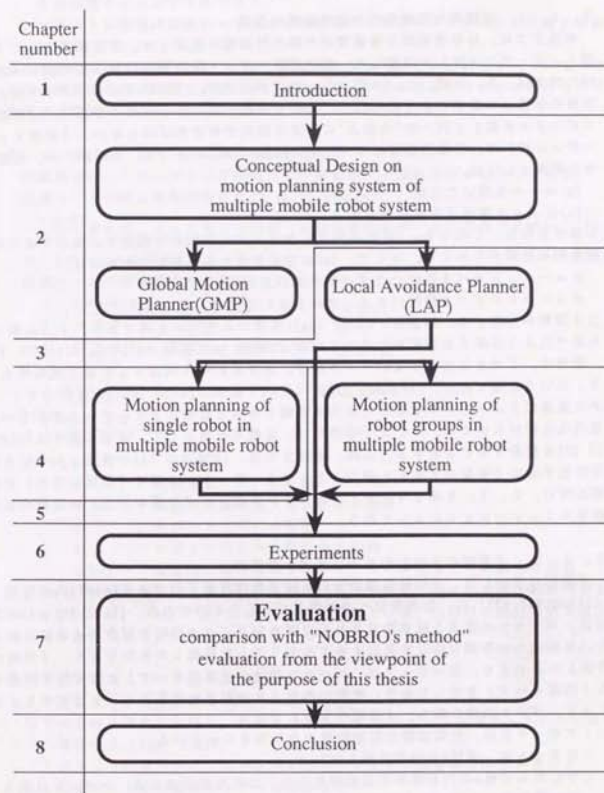


Fig. 7.1 The structure of chapter 7

7. 1 従来研究との定量的比較

7. 1. 1 定量的な比較を行う従来研究の選定

本論文では、分散管理型で階層構造の動作計画器を設計した、下位構造として『仮想インピーダンス法』を提案した、複数移動ロボット系の動作計画手法に関する定量的な評価は、1. 2節、1. 3節で行った。本節では、『仮想インピーダンス法』と従来の手法との定量的な比較を行う。比較することのできる従来の手法は、『仮想インピーダンス法』と同一の“枠組み”に属する研究でなければならない。『仮想インピーダンス法』は、分散管理型のオンライン計画手法である(2. 3節)ため、比較できる研究としては、

(a) ルールを用いた方法

(b) ロボット間通信を用いた方法
が挙げられる。このうち、(b)の手法は、ロボットに付加的な機能を必要とするため、定量的な比較がしにくい。そこで、(a)の方法が考えられるが、それには、

チューニングすべき設計パラメータが多い方法

チューニングすべき設計パラメータが少ない方法
の2種類が存在する。前者は、fuzzy logicを用いた方法が主流であり、1. 2節でも述べたように多くの研究がなされている(【武野 88】【前田 88】【小山 91】【石川 91】)。後者は、「できるだけ少ないルール数で、必ず目的地に到達する計画手法を考えよう」という立場であり、【NOBORIO 92】が代表的である。fuzzy logicの設計パラメータを最適にチューニングすることは大変困難であり『仮想インピーダンス法』との定量的な比較がしにくい。そこで、本節では、比較の対象として、後者に属する【NOBORIO 92】を採用することとする。以降、本論文では、【NOBORIO 92】で提案されている動作計画手法を『登尾の方法』と呼ぶこととする。7. 1. 2項で『登尾の方法』の説明を行い、7. 1. 3項で『仮想インピーダンス法』と『登尾の方法』の定量的な比較をシミュレーションによって行う。

7. 1. 2 『登尾の方法』のアルゴリズムの詳細

『登尾の方法』は、未知静的環境下における動作計画手法であるLumelskyの方法(【LUMELSKY 87】)の、動的環境への拡張を目指したものである。【LUMELSKY 87】の方法は、ゴールへの直進と障害物まわりの迂回という二つの戦略を組み合わせることにより、目的地への単調な接近を保障するアルゴリズムを提案したものである。『登尾の方法』においても、各ロボットは、ゴールへ向かう「直進モード」と障害物を回避する「回避モード」を有しており、状況に応じて2つのモードをスイッチングすることにより、ゴール到達を図る。『登尾の方法』では、

- ・ロボットには、初期位置と目標位置のみが与えられている。
- ・ロボットは、半径Rの円形状をしている。
- ・中心から半径eの円形状の可視領域をもつ。この可視領域では、ロボットは最も近い障害物・ロボット上の点Cが見いだせる。
- ・制御指令は、1 サンプル時間後の現在位置からの移動量である。これは、移

動量をサンプリング時間で割ることにより速度と考えても差し支えない。指令した移動量の分は必ず移動できる。

- ・ロボットは環境内の他のロボット・障害物の1 サンプル時間あたりの最大移動量O_MAXを認識している。

に示す想定をしている(Fig. 7.2)。最初にロボットがそれぞれのモードになっている時の移動指令値の計算方法を示す。次に、モード間を遷移するための条件を示す。最後に『登尾の方法』における設計パラメータを示す。

7. 1. 2. 1 各モードにおける移動指令値

移動指令は、1 サンプル時間当たりの移動量rと移動方向αの2つである。直進モードの時の移動指令を式(7.1)、(7.2)に示す。

$$r = \frac{e-h-R}{2} \quad \text{または、ゴールまでの距離(可視領域内にゴールが存在する時)} \quad (7.1)$$

$$\alpha = (\text{ゴールへの向き}) \quad (7.2)$$

回避モードの際の移動指令を式(7.3)、(7.4)に示す。

$$r = \begin{cases} -\left(\frac{2d-e-h-R}{e-h-R}\right)z \cdot \frac{e-h-R}{2} & \text{(方法 } r_1) \\ -\left(\frac{2d-e-h-R}{e-h-R}\right)z \cdot \left(\frac{e-h-R}{2} - O_{MAX}\right) + \frac{e-h-R}{2} & \text{(方法 } r_2) \end{cases} \quad (7.3)$$

$$\alpha = \begin{cases} \left(\frac{2d-e-h-R}{e-h-R}\right)p \cdot \frac{\pi}{2}, & p=1.0 & \text{(方法 } p_1) \\ \left(\frac{2d-e-h-R}{e-h-R}\right)p \cdot \frac{\pi}{2}, & p=3.0 & \text{(方法 } p_2) \end{cases} \quad (7.4)$$

ただし、

- e : 可視領域(円)の半径
- h : 障害物周囲の侵入禁止領域の距離
- R : ロボット(円)の半径
- d : ロボット中心から最近点の距離
- O_MAX : 環境内の障害物の1 サンプル時間あたりの最大移動量

式(7.3)、(7.4)において、それぞれ方法 r_1 、 r_2 、方法 p_1 、 p_2 をあらかじめ選択するものとする。また、αは、Fig. 7.3に示す表記をしている。回避時に時計回りをか反時計回りにするかも、あらかじめ選択するものとする。

7. 1. 2. 2 モード間の遷移条件

(a) 直進モードから回避モードへの遷移条件

以下の条件を満たした時に、最近点Cを衝突点 H_1 として、直進モードから回避モードへ遷移する(Fig. 7.4)。

- ・ロボットのセンシングエリア内に他のロボット・障害物が存在すること

(b) 回避モードから直進モードへの遷移条件

以下の2つの条件を満たした時に、最近点Cを離脱点 L_1 として、回避モードから

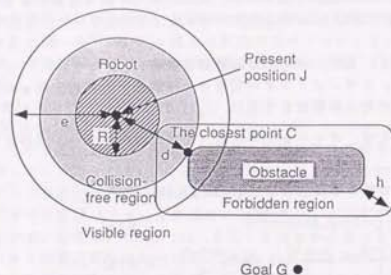


Fig. 7.2 A mobile robot with a visible region and its obstacle with a forbidden region in a workspace([NOBORIO 92])

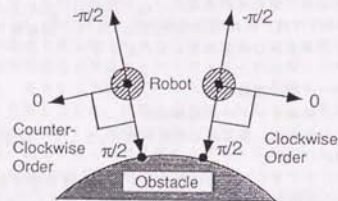


Fig. 7.3 An angle to define a robot orientation([NOBORIO 92])

直進モードへ遷移する (Fig. 7.4) . Fig. 7.4において、スタートSを離脱点 L_0 とする。

- ・最近点 C が、Fig. 7.5において、
 - 弧 β (時計周りを選択している場合)
 - 弧 α (反時計周りを選択している場合)
 に存在している時
- ・ゴールと最近点の距離が、以下に定義される距離 d^* より小さい時
 - (1) これまでで、最もゴールに接近した最近点 C からゴールまでの距離
 - (2) 衝突点 H_1, \dots, H_i とゴール間の距離の最小値
 - (3) 離脱点 L_0, \dots, L_{i-1} とゴール間の距離の最小値
 - (4) 衝突点 H_1, \dots, H_i とゴール間の距離の平均値
 - (5) 離脱点 L_0, \dots, L_{i-1} とゴール間の距離の平均値

上記の(1)~(5)はあらかじめ選択するものとする。[NOBORIO 92]においては、(6)が存在するが、[NOBORIO 92]において好ましい評価がなされていないかったので、ここでは省略した。

『登尾の方法』における設計パラメータを以下に示す。

1. 回避モードの際に時計周りを選択するか反時計周りを選択するか
2. 可視領域 e
3. 侵入禁止領域の距離 h
4. 回避モードの時の r の決定法 (r_1, r_2 の2通り)
5. 回避モードの時の α の決定法 (p_1, p_2 の2通り)
6. 回避モード→直進モードとなる時の、 d^* の計算方法 (5通り)

文献[NOBORIO 92]においては、上記のアルゴリズムを用いた場合のゴールへの収束性の証明を行っているが、ここでは、省略する。この証明は、あるロボットがある1台の他のロボットを回避し終わるまでは、それ以外のロボットとは干渉しないという前提がある。そのため、ロボットの密度が非常に高い環境化においては、上記の仮定が成立たないため、ゴールへの収束性が保証されない。実際、[NOBORIO 92]に示されているアルゴリズムにおいては、「あるロボットAが近傍のロボットBを回避している最中に、別のロボットCがロボットBよりロボットAの近傍になった時には、ゴール到達をきらめる」と記されている。次項のシミュレーションでは、『登尾の方法』を幾分か拡張して、常に一番近傍のロボットを回避する戦略を採ることとする。これは、「あるロボットAが近傍のロボットBを回避している最中に、別のロボットCがロボットBよりロボットAの近傍になった時には、ロボットAはロボットCに対して回避行動を採る」という意味である。もちろん、この際のロボットのゴール到達性は保証されない。

7. 1. 3 シミュレーションによる定量的比較

本項では、『仮想インピーダンス法』と、前項で説明した『登尾の方法』をシミュ

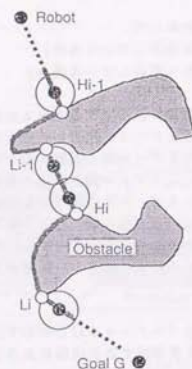


Fig. 7.4 Successive selection of leave and hit points L and H in a workspace

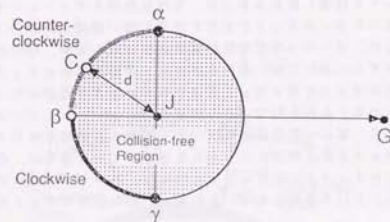
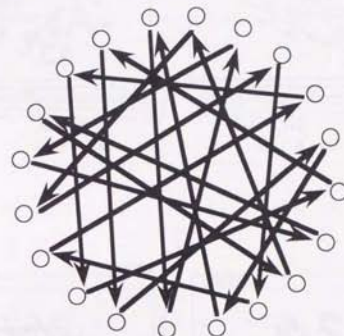


Fig. 7.5 The position of the closest point C on a collision-free region around the robot

レーションにより比較する。シミュレーション環境としては、3.4節で示した広場環境走行シミュレーション (Fig. 7.6) において4種類のロボットの密度、各密度に対して5個ずつのサンプルデータを用いて比較を行う。Table 7.1にシミュレーション条件を示す。『登尾の方法』においては、回避方向は時計周りとし、前項で示した7種類の設計パラメータを総当り的に探索し、ゴール到達時間が最も短いパラメータを各サンプルデータ毎に選択した。『仮想インビデダンス法』においては、3章のランダム多スタート局所探索法を用いて、サンプルデータ毎に最適近似パラメータを選択した。

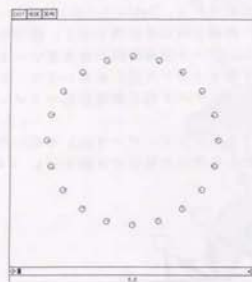
環境内のロボットの台数が20台の時の『仮想インビデダンス法』を用いた場合のロボットの挙動をFig. 7.7に、『登尾の方法』を用いた場合の挙動をFig. 7.8に示す。



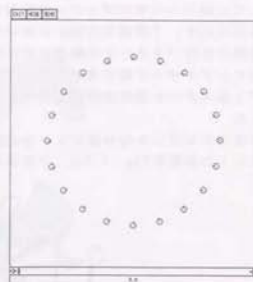
goal positions for each robot is given randomly

Fig. 7.6 The plaza environment for simulations
Table 7.1 Conditions for the plaza environment

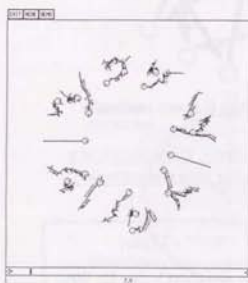
Radius of the circle for start/goal positions	7.0[m]
Number of robots	4 kinds(5, 10, 15, 20)



(a) 0 sec.

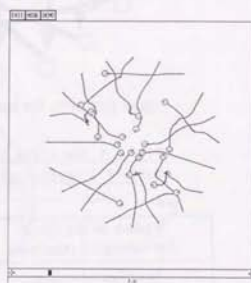


(a) 0 sec.



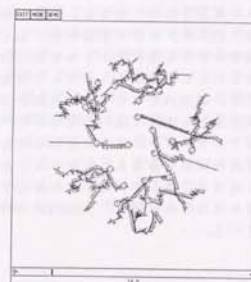
(b) 7 sec.

Fig. 7.7 Trajectory with
Noborio's method
(continue to next page)

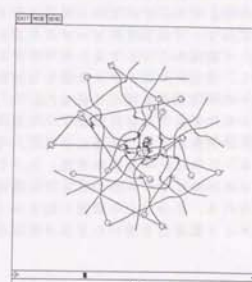


(b) 7 sec.

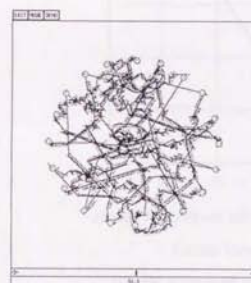
Fig. 7.8 Trajectory with
Virtual Impedance Method
(continue to next page)



(c) 1.4 sec.

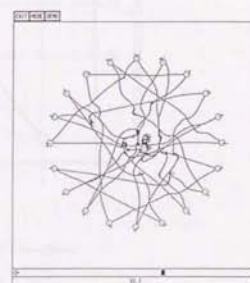


(c) 1.4 sec.



(d) 51.1 sec.

Fig. 7.7 Trajectory with
Noborio's method



(d) 22.7 sec.

Fig. 7.8 Trajectory with
Virtual Impedance Method

『登尾の方法』では、あるロボットがセンシングエリア内に他のロボットを見つけた瞬間に、時計回りに回避する動作を行っている。一方『仮想インピーダンス法』では、状況に応じて回避方向が異なることがわかる。このシミュレーションの例では、『登尾の方法』では、すべてのロボットがゴールに到達するのに約51.1秒かかっているのに対し、『仮想インピーダンス法』では、約23.0秒で移動が完了している。また、『登尾の方法』では、ロボットの移動中における速度ベクトルの方向の変動が大きく、仮想インピーダンス法では変動が小さいことがわかる。ロボットの台数とロボットの平均到達時間の比較をFig. 7.9に示す。環境内のロボットの台数が少ない時、すなわちロボットの密度が小さい時には、両方法とも大きな違いが見られないが、ロボットの密度が大きくなるにつれて、『仮想インピーダンス法』が有利になることがわかる。これは、『登尾の方法』は、2体問題の解決を基本としたモデルであるため、一度に3体や4体が干渉するような複雑した環境には適していないのが原因であると考えられる。3体以上の問題に対する『登尾の方法』の拡張は、[NOBORIO 93]においてロボット間通信を用いた方法が提案されている。

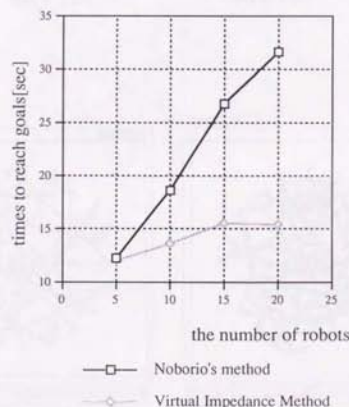


Fig. 7.9 Comparison of average times to reach goals between Noborio's method and Virtual Impedance Method

このような3体以上の物体が干渉する環境に対する限界は、『登尾の方法』以外のルールを用いた方法にも共通する問題点であると考えられる。すなわち、2体問題に適したルールを3体以上の環境に利用すると望ましくない挙動を示し、また3体以上の問題に対応するルールを考慮することはn台の問題に対する対応するルールを考慮することになるためルールの爆発を招き好ましくない。『仮想インピーダンス法』は、多体のロボットが同時に干渉する系に対して有効であることが明らかになった。

『登尾の方法』が『仮想インピーダンス法』より優れている要素も、いくつか存在する。例えば、要求するセンサの性能は、『仮想インピーダンス法』が相対位置・速度の両者を要求するのにに対し、『登尾の方法』は相対位置のみで十分である。更に、ゴール到達性の保証、動作計画アルゴリズムの簡単さ、設計パラメータの種類の少なさは、『登尾の方法』が勝っていると考えられる。すなわち、環境内のロボットの密度が低いことが保証されており、ロボットが単独で行動するような環境では、『登尾の方法』で十分であると言える。

7. 2 本システム全体の評価

本論文の研究の目的は、①各ロボットが自律的に意思決定でき、②群行動の動作計画が実現できる、複数移動ロボット系の動作計画手法の確立であった。本節では、本論文で提案した手法が、上記2つの要求仕様を満たすかどうかの評価を行う。その後、

- ・車輪型移動ロボットの動作計画可能性の評価
- ・上位階層との統合可能性に関する評価

を行う。前者は、動作計画に沿って実際にロボットを駆動する際に重要となる点であり、後者は、本論文で提案した手法の拡張性に関する議論である。

7. 2. 1 自律分散システムの実現可能性に関する評価

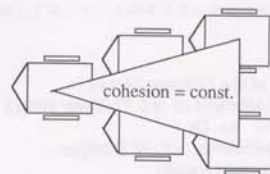
本論文で提案した手法は、パラメータ設計、動作計画ともに自律的に意思決定により実現できるものである。また、本論文で行われたシミュレーションではすべてのロボットを等質と看做するものが殆どであったが、ロボットの機能に応じて設計パラメータを変化させることで、振舞に個性を持たせることも十分可能であると考えられる。更に、本論文で提案した手法は、『登尾の手法』と比較してロボットの台数が多い時に効果を発揮することが7. 1節の議論で明らかになった。自律分散的に動作計画を行う従来の手法の殆どはいくつかのif-then形式ルールをベースに用いたものであるが、ロボットの密度が増大するにつれて考慮すべき状況の数が爆発的に増大するため、好ましくない。それに対して、ポテンシャルを用いた方法は、ロボットの台数にかかわらず統一した手法で計画を実現できるため有効である。

7. 2. 2 群行動の動作計画の実現可能性に関する評価

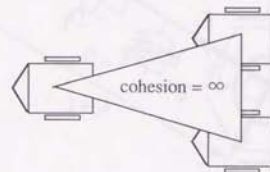
従来から提案されてきた複数移動ロボット系の動作計画手法は、『登尾の方法』がそうであるように、自分以外のロボットを避けるべき障害物とみなしたものであり、複数ロボット系における単体ロボットの計画とロボット群の計画を統合して取扱うことがほとんど不可能であった。それに対して、本論文で提案した手法は、ロボット間の関係を集中定数系の仮想的なバネ・ダンパで表現したものである。この方法は、3. 4節のシミュレーションで示したように、ロボット間の仮想バネ・ダンパによる拘束の仕方を変えるだけで、単体ロボットの動作計画と、ロボット群の動作計画の両者を統合して取扱うことができる。このことは、複数ロボット系内での、単体での行動、群としての行動を動的に変化させるような動作を実現する上で非常に有用であると考えられる。実際、その有効性は、5章で示されたと考えられる。

本論文では、ロボット群の行動の指標として『結合度』を提案した。これは、群れている状態を解体するための指標であると考えられる。作業実現を考えた場合、結合度を定義した状況とは逆に、各ロボットが相互に離れた状態から作業地点に集結するための指標が必要である。すなわち、ある作業を実現するために必要なロボットの台数の範囲を明示できる『集結度』なる指標を定義し導入すれば、作業計画問題の解決に役立つと考えられる。例えば、ある大型対象物を搬送するために、ロボットが3台

以上必要であったとし、ある時、その物体を5台で搬送していたとする。この場合では、2台が搬送作業から離れて他の作業に就くことができるため、5台のロボット間の結合度を低く設定する (Fig. 7.10(a))。もし、5台のうちの2台が他の作業に就くために搬送作業を中止したとすると、それ以上ロボットが離れてしまつては困るので結合度を ∞ に設定する (Fig. 7.10(b))。このことから、結合度および集結度をうまく設計することにより作業の動的な割り付けが可能になることが期待される。



(a) When the number of robots is redundant



(b) When the number of robots is not redundant

Fig. 7.10 Transporting tasks of multiple mobile robots

7. 2. 3 車輪型移動ロボットの動作計画可能性の評価

6章の実験結果より、コントローラの設計により、『仮想インピーダンス法』を車輪型移動ロボットの動作計画に適用可能であることが示された。移動ロボットの車輪機構が変わったときには、コントローラの部分を変更するのみで対応可能である。また、力の干渉を伴うような高度な制御性を目指した作業の場合、すなわち複数台の車輪型移動ロボットが協調して大型対象物を搬送するような場合は、LAPに車輪機構を組込むことにより対応が可能である。Fig. 7.11に車輪機構を考慮した仮想バネ・ダンパモデルの例を示す。これは、[太田 93b]における2台のロボットによる搬送作業の際のモデルである。この論文では、車輪が動かない方向の仮想外力を仮想モーメントに変換する制御モデルを考案している。

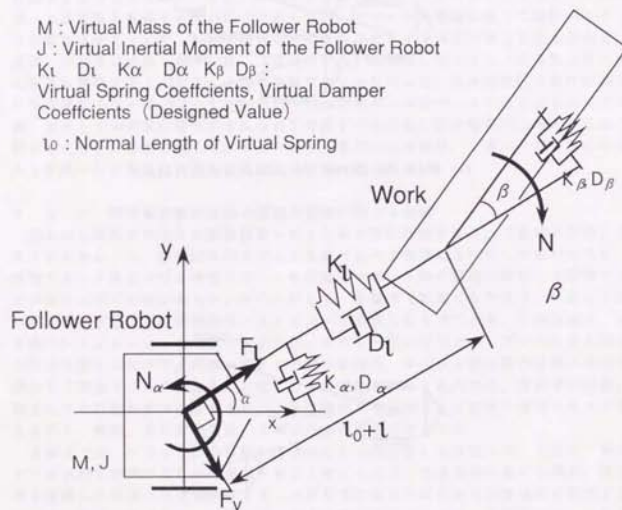


Fig. 7.11 Virtual impedance model of a car-like mobile robot ([太田 93])

7. 2. 4 上位階層との統合可能性に関する評価

『仮想インピーダンス法』は、ロボット・障害物・目標位置間に張られた仮想的なバネ・ダンパにより働く仮想外力に基づいて計画を実現している。これは一番低レベルでの議論であるが、より上位のレベルに位置する、動作をする上での様々なスキルも、仮想バネ・ダンパのパラメータや構造を変化させることにより実現可能である。例えば、『道の右側を歩く』というスキルは、道の左側に仮想的なマスを設置することにより実現できる。このことは、4章の追従戦略、5章の結合度の提案においても実現されており、『仮想インピーダンス法』が有効である点の一つである。すなわち、作業計画やその他の上位の概念を仮想バネ・ダンパのパラメータ変化や構造変化の形で表現することにより、現在の枠組みをそのまま保持しつつ、系を拡張することができる。本論文で提案した動作計画期と上位階層との統合は容易であると考えられる。作業計画器との統合に関しては、8. 2節の今後の展望で触れる。

7. 3 7章のまとめ

本章では、本論文で提案した動作計画手法の評価を行った。

1節では、従来研究との定量的比較を行った。『仮想インピーダンス法』と同じ枠組みの計画手法で再現しやすい手法として『登尾の方法』(NOBORIO 92)を採用した。広場環境走行シミュレーションを行った結果、環境内のロボットの密度が比較的小さい時には両者ともほぼ等しい計画性能が得られるが、密度が高くなるにつれて『仮想インピーダンス法』が有効であることが示された。

2節では、その他の項目について評価を行った。具体的には、群形態の表現可能性、分散管理の実現可能性等について論じ、本論文で提案した手法が本研究の目的を満たしたものであることを議論した。その他、上位階層との統合性についても論じ、本手法が作業計画システムとの親和性が高いことを主張した。

第8章 結論と今後の展望

- 8. 1 結論
- 8. 2 今後の展望
- 8. 3 8章のまとめ

本章では、本論文の結論と展望について述べる。

8. 1 節では、本論文の結論を述べる。

8. 2 節では、今後の展望を述べる。1 項で今後の課題について述べる。また、本論文の主題は、複数移動ロボット系の動作計画手法の設計であったが、この論文の成果は、様々な分野に適用可能である。2 項で適用範囲について述べ、3 項で複数移動ロボット系の作業計画、作業割当ての問題について、今後の展望を述べる。

8. 3 節では、本章のまとめを述べる。

Fig. 8.1 に論文全体における本章の位置付けを示す。

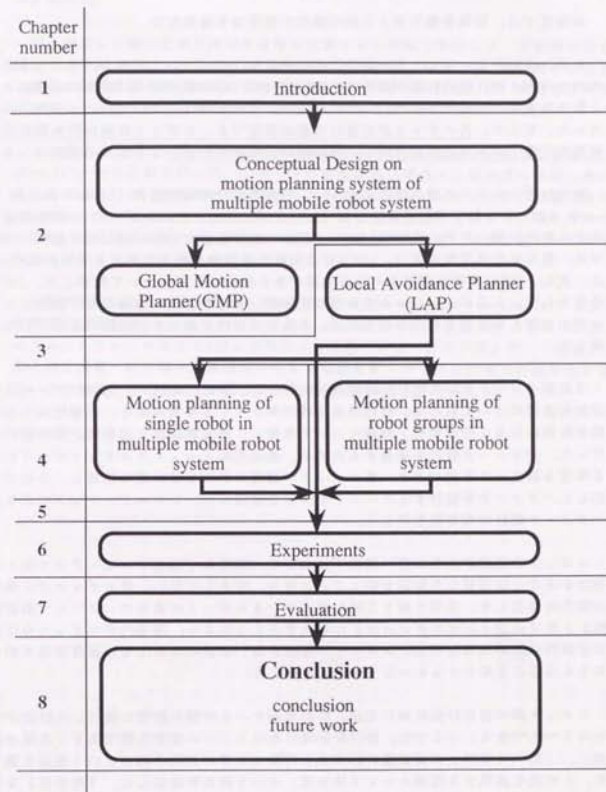


Fig. 8.1 The structure of chapter 8

8. 1 結論

本論文では、複数移動ロボット系の動作計画手法を提案した。

・まず、複数移動ロボット系の動作計画に必要な仕様について考察し、ロボットの自律的な計画と群行動の計画が重要であると述べた。この観点から、従来から提案された動作計画手法を概観すると、いずれも上記の仕様を満たしていないことが明らかになった。そこで、各ロボットが自律的に意志決定でき、ロボットの群行動の動作計画が実現可能である複数移動ロボット系の動作計画手法の設計を本論文の目的とした。

・動作計画システムの概念設計を行った。分散管理型大域計画器 (Global Motion Planner, GMP) と分散管理型局所計画器 (Local Avoidance Planner, LAP) の階層構造を有する動作計画システムを提案した。上記2つの計画器の分担に関して考察し、GMPでは、既知動的障害物を扱い、LAPでは未知動的障害物を扱うべきであると結論づけた。次に、GMPに必要な機能とそれを実現できる従来の研究について概観した。LAPの手法として、人工ポテンシャル法を集中定数系、分散管理型に改良した『仮想インピーダンス法』を提案した。この手法は、本論文の目的を満たす計画器であることを主張した。

・『仮想インピーダンス法』の詳細設計を行った。定式化を行い、設計パラメータの定性的性質について述べた。目標地点以外で停止をする停留問題と、目標地点への移動が困難になる「妨害状況」問題について考察し、停留解決器と妨害解決器の設計を行った。パラメータ設計を実現するために、最適化法とニューラルネットワークによる学習を組合せた手法により、各ロボットが環境の状況を実時間で把握し、それに適応したパラメータを設計するアーキテクチャを提案した。シミュレーションにより、パラメータ設計の有効性を示した。

・ロボットの密度が非常に高い環境下において、単純に『仮想インピーダンス法』を利用することは混雑した状況を招くことになり、好ましくない。各ロボットに「自分が前方に存在して、自分と同じ方向に進んでいるロボットの後をついていく『追従戦略』」を『仮想インピーダンス法』に導入することにより、環境内のロボットの行動に自律的に秩序をもたらし、ロボットの密度が高い状況においても、混雑状況を招きにくくなることをシミュレーションにより示した。

・ロボット群の動作計画においては、群が実現すべき作業の性質に基づいた計画がなされるべきである。ここでは、動作計画時における二つの要求仕様である「危険の回避」、「群れの保持」の重要度の比により作業の性質が表現されるという仮説を建てた。この比を表現する指標として「結合度」という概念を提案した。『結合度』を有するロボット群の動作計画手法を『仮想インピーダンス法』の枠組みの中で実現し、『結合度』の設計により、二つの要求仕様の重みづけがなされていることをシミュレ

ーションにより示した。これにより、作業の性質を考慮した動作計画手法が構築できたとと言える。

・以上提案した動作計画手法の有効性を実機による実験で検証した。実験時に最も問題となるのが、ロボット間の相対位置の測定であるが、本論では、複数個の点光源から構成された看板を用いた相対位置・姿勢計測システムを提案した。実機により2台のロボットの相互回避実験を行い、提案した計画手法の有効性を示した。

・『仮想インピーダンス法』を定量的に評価するために、登尾の提案した手法 (NOB ORI0 92)) との比較を行った。ロボットの密度が低い場合の計画軌道の性能に違いは見られないものの、密度が高くなるにつれて『仮想インピーダンス法』の優位性が明らかになった。更に、両手法の計算量、センサ能力の要請等について比較を行った。その後、『仮想インピーダンス法』における車輪機構の考慮、作業計画システムの統合可能性等についての評価を行った。

上記の議論により、分散管理型大域計画器 (Global Motion Planner, GMP) と分散管理型局所計画器 (Local Avoidance Planner, LAP) の階層構造を有する動作計画システム、とりわけその中のLAPに相当する『仮想インピーダンス法』が、「自律的な意志決定機能・群行動の計画機能を有する複数移動ロボット系の動作計画手法の設計」という本論文の目的を満たす動作計画システムであることを示した。

8. 2 今後の展望

本節では、1項で本論文の課題を述べ、2項で本論文で提案した手法がどのような分野に拡張可能かについて展望する。3項では、2項の内容から1つピックアップして複数移動ロボット系の作業計画器の提案を行う。

8. 2. 1 本論文の課題

本論文で提案した手法において克服すべき課題は多数存在するが、そのいくつかを以下に列挙する。これらの課題はいずれも重要なものであり、今後の研究対象とした。

・多数の戦略を導入した場合の整合性について

Fig. 2.9において、仮想インピーダンス法の位置付けについて述べ、いくつかの戦略を仮想インピーダンス法の上に投影させる方法を提案した。ここで複数の戦略の導入を考えると、戦略同士が整合性を有するかどうかをチェックする機能が必要である。・ロボット間通信機能と本論文の手法の関係について

本論文では、ロボット間の隔った通信を全く想定しなかったが、実際的には、通信機能をうまく取り込むことにより、作業性がより一層向上すると考えられる。具体的には通信による負荷の評価方法が問題になると考えられる。

・仮想インピーダンスと群行動との関係について

ロボットの目標入力と実際の動作との関係が、あるインピーダンス行列という形で表現されることを3章で述べた。このインピーダンス行列を拡張することによりロボット群としての行動がインピーダンス行列の性質を調べることで明らかになる可能性がある。

8. 2. 2 本論文で提案した手法が適用できる分野

本論文で提案した手法は、一つの環境内の多数の自律的なエージェントが相互に干渉しながら何らかの作業を行う系の解析、設計に適していると考えられる。以下、いくつかの具体例を挙げる。

生産計画システムへの適用

昨今、生物指向型生産システムという概念が提唱され〔上田 93〕、生産計画に自律分散的な概念が導入されている。本論文の4章において、廊下を走行するロボットのシミュレーションを行い、追従戦略の有効性を示した。ここで、ロボットとタスク、廊下とプロセスというアナロジーをとると、追従戦略は『同種の仕事を同一の機械でこなす』という戦略と対応することがわかる。この戦略は生産計画に非常に有用であると考えられるため、4章の成果が生産計画に適用できる可能性がある。

航空管制システムへの適用

ある飛行場に多数の飛行機が離発着する状況を考える。この際、各飛行機同士の挙動は、仮想的なバネ・マス・ダンパモデルで近似できる可能性がある。各飛行機のバ

イロットが離発着する際のルールを適当に付与した時に、離発着がスムーズに進行するか否かをシミュレートし、どのルールを与えると妥当かを判断するためのツールとして、仮想インピーダンス法が適用できる可能性がある。

都市計画への適用

魚の行動を仮想的なバネ・マス系でモデル化した研究〔ドウスクリ 92〕や歩行者の歩行パターンを仮想的なバネ・マス系でモデル化した研究〔平井 77〕は既に存在する。仮想インピーダンス法を用いれば、例えば、デパート等、混雑する環境において緊急時の避難を円滑にするために、避難経路をどう作るべきか、表示灯をどのように設置すべきかという環境設計の問題に対処できる。また、緊急時に各個人がどのような振舞いをするか望ましいかがわかれば、普段から個人にはそのような教育をすれば良いことになる。これも仮想インピーダンス法に適用可能であると考えられる。

移動ロボットの作業計画への適用

複数移動ロボット系の作業計画器を設計することを考える。結合度の概念は作業表現をするのに適していると考えられる。追従戦略等の妥当な戦略を付与することで、秩序が生成し、作業の遂行が迅速に行われるに違いない。これに関して、3項で詳しく述べる。

8. 2. 3 複数移動ロボット系の作業計画器の提案

本論文では、複数移動ロボット系の動作計画手法を提案したが、これは、複数のロボットによる作業の実現という大きな枠組みから考えると、一部分に過ぎない。ここでは、今後の展望として、複数移動ロボット系の作業計画システムを提案する。

多数自律ロボット系における作業計画システムを、多数自律ロボット系における局所通信と、行動規範という2つの枠組みを中心に設定する。すなわち、各ロボットが自分自身の近傍のロボットとのみ通信できる局所型通信と各ロボットが環境に対して反射的に行動する行動規範型制御を組合せたアプローチを採る。このシステムでは、計画に要するロボット間通信のオーバーヘッドを抑えつつ、高度な作業性能を得ることができると考えられる。

まず、研究の背景から述べる。ロボット技術の進歩により、数百台から数千台の自律ロボットが協調して高度な作業を行うことが実現可能になりつつある。このような「多数自律ロボット系」では、複数個の作業を多数台のロボットに割り振り、作業を迅速に実行する「作業計画」研究が重要である。従来から提案されている作業計画手法は、以下の2つに大別される。

(a) ロボット同士が陽に通信することにより所与の作業を実現する方法

(b) 個々のロボットが勝手に環境に対して反射行動を行うことにより、その集積として所与の作業を実現する方法

ここで、(a)は、分散人工知能の分野で盛んに行われている研究で、ロボット同士の交渉による契約ネットプロトコルを用いた研究が有名である。これは、人間集団の計

画立案作業等をモデル化していると言える。(b)は、行動規範型計画と呼ばれる研究分野で、ロボット同士の通信を必要としない。これは、軍隊様の求組行動等をモデル化していると考えられる。ここで、(a)の手法は、系のロボットの台数が少ない場合には高度な作業計画が実現できる反面、ロボットの台数が増加した際の通信負荷が大きな問題となる。(b)の手法は、各ロボットの反射行動と実現できる作業との関係が明確でないため、要求する作業を実現する反射行動を設計することが難しく、また高度な作業の実現が難しい。このように作業計画問題には、通信量と作業性能のトレードオフが存在しており、その両者を解決する計画手法は提案されていない。換言すれば、従来研究は、通信負荷と作業性能に関して両極端に位置する手法であると言えるが、実用的なシステム構築のためにはその両者を統合する必要があることは明らかである。

上記の考察により、作業計画問題においては、通信量を最小限に抑え、かつ、作業性能を向上させるという二つの要求仕様の両者を満たすことが重要であることがわかった。そこで、本テーマでは、各ロボットが自分自身の近傍のロボットとのみ局所的に通信する『局所通信』と、ロボット間の交渉なしで環境情報から各ロボットが自律的に作業を選択する『行動規範』を各ロボットに付与した作業計画手法を構築し、上記の問題点の解決を図る。

このテーマは、作業実現の手順に沿って、以下の3つのサブテーマに分けることができる。

- (1) 局所通信による作業内容の告知、
- (2) 作業のロボットへの割り付け、
- (3) 割当てられたロボットによる作業の実現。

上記3つのサブテーマの詳細を以下に示す。

(1) 作業内容の告知：作業の要求は、環境内のある特定の場所に提示される。この情報は、その作業を実行するロボットのみが知ればよく、環境内の全てのロボットが知る必要はない、すなわち、局所的な通信のみで十分である。局所通信における情報の伝播のしくみは、システム全体としてマクロ的に見れば、ロジスティック方程式で表現可能なことが示されている([ARA1 93a][新井 93b])。情報伝播の解析として、局所的な通信による情報の伝播特性の解析を、ロボットが移動する環境を対象として数学的に解析する必要がある。システム全体の特性をとらえるため、ランダムウォークするロボットの空間分布をボアソン分布による確率モデルで表現する。これを用いて、微分方程式で情報伝播モデルを構築し、その近似解を求めることにより、ロボット数や通信範囲等をパラメータとする情報伝播の数学的モデルを構築する。更に進めて、情報伝播の解析と伝達すべき作業情報の解析を行う。具体的には、

- ・ロボットの移動が(ランダムウォークではなく)目標地点への移動を行う場合、
- ・ロボットが作業を融合的に獲得する場合

に関して解析する。伝達すべき作業情報の解析に関しては、先に述べた作業解析結果を基に、伝達する作業情報量と実現された作業性能との関係を明らかにする。最終的には、通信と作業性を考慮した最適作業情報量を導出する。

(2) 作業のロボットへの割り付け：作業の特性として、同一作業を実現している複数

台のロボット(ロボット群と呼ぶ)相互の繋がりや強さとして、『結合度』という概念を既に提案した。この結合度は、作業の性質を表現したものであるため、この概念を用いて作業を表現する。ここでは、本システムにおいて実現すべき幾つかの基本的な作業(協調荷運び作業、未知領域の地図生成、パケツリレー等)に関して、各ロボットが作業情報を認識して、環境の状況(ロボットの密度、他のロボットが作業に参加しそうか否か)を判断すること、他のロボットと通信することなしに自律的な判断のみにより作業に参加するか否かを決定する行動規範を設計する。作業割り付けがきちんとなされる行動規範を各ロボットに埋め込むことを目指す。更に、

- ・環境が変化した際のパラメータチューニングが困難である、
- ・センサ情報の誤差、通信の誤りの影響がある、

のような問題点を解決するために、環境の状況に応じた最適パラメータを各ロボットが学習により獲得する方法を構築する。学習には、作業中に実時間で行う方法と、作業前にあらかじめ行う方法の2種類が存在するが、本テーマでは、後者の学習をターゲットとする。この場合は、環境状況の抽象的記述法と学習されたパラメータの記憶方式が問題となるが、3.3節のパラメータ設計の議論が役に立つと考えられる。

(3) 作業の実現：本環境では、他のロボットを移動障害物としてとらえ、動的に衝突回避を行う必要があるため、仮想インピーダンス法が適用できると考えられる。具体的には、実現すべき作業を、仮想バネ・ダンパの枠組みで表現する。例えば、協調搬送のように群れて動作する作業は、ロボット間を自然長の短い仮想的なバネ・ダンパで拘束したバネ・ダンパモデルを採用することにより、ロボットはロボット相互の位置を保ったまま移動するため、搬送作業が実現できる。また、未知領域の地図生成のように、ロボット同士が分散して作業する必要がある場合、ロボット間を自然長の長い仮想的なバネ・ダンパから構成されるモデルを採用することで、作業が実現できると考えられる。このようにして、基本作業を仮想バネ・ダンパモデルで表現できれば、ロボット群が形成された後の計画は容易に実現できると考えられる。

これまで書かれた解析の検証は、すべてワークステーション上で行うが、最終的には、実際のロボットを用いて実験を行い、提案した作業計画システム全体の有効性を検証する必要がある。一番の問題点は、多数ロボット系における局所通信システムの実現である。本研究では、干渉が少ないという利点を考えて、超音波や無線ではなく、光を用いることとする。6章で示した看板による移動ロボット相互位置・姿勢計測システムに、通信用の看板を加えることにより、局所的に通信できるシステムを構成できる。実際のロボットに看板を搭載し、作業を実現させたい。

本章では、結論と今後の展望について述べた。

2 節では、1 項で本論文の成果の適用範囲についての概観した、2 項で本論文で提案した動作計画システムを拡張した作業計画システムを提案した。作業計画において重要ななるのは、通信量と作業性とのトレードオフの問題であると考えられる。明示的な通信による負荷を軽減するために、局所的通信、行動規範型制御の概念をどう適切に組み込めるか、作業計画システム実現のための一番の課題ではなideうだろうか。

本論文は、筆者が東京大学工学部精密機械工学科助手である約2年半の間、新井・大隅研究室において行った研究の成果をまとめたものです。その間絶えず熱心にご指導頂きました東京大学工学部精密機械工学科教授 新井 民夫先生に心より感謝致します。先生の斬新なアイデア、深遠な探求力のおかげで本論文の完成させることができました。また、研究を離れた一社会人としての振る舞い方等においても、数多くのことを教えて頂きました。

本論文をまとめるにあたり、適切な御指導を賜りました、東京大学工学部精密機械工学科教授 高野 政晴先生、東京大学工学部計数工学科教授 有本 卓先生、東京大学工学部精密機械工学科教授 大園 成夫先生、東京大学先端科学技術センター教授 佐藤 知正先生に心から感謝の意を表します。先生方の御指摘により将来的な研究課題に関する知見を得ることができました。

東京大学工学部精密機械工学科助教授 大隅 久先生には、筆者が在学中からの同研究室の先輩、助手、先生として有益で暖かい助言と鋭いご指摘をいただき、大変お世話になりました。筆者が学部学生から修士1年の頃にかけて同研究室の助手であった松元 明弘先生（現、東洋大学助教授）には、常日頃から適切な助言を頂くことができました。登尾 啓史先生（現、大阪電気通信大学助教授）、坪内 孝司氏（現、東京大学工学部計数工学科助手）には、学会などの折に研究の方向付けに関する有益な指摘を受けました。特に、登尾先生には、7章の評価をするためのコメントを頂くことができました。油田 信一先生（現、筑波大学助教授）には、移動ロボット研究に対する姿勢を教えてくださいました。浅間 一氏（現、理化学研究所）には、研究の概念の重要性を教えてくださいました。石田 慶樹氏（現、東京大学大型計算機センター）には、筆者が学生の頃から様々な助言をして頂きました。陶山 毅氏（現、東京ガス）には、定期的な勉強会におけるディスカッションにより多大な知見を得ることができました。ここに、謹んで感謝の意を表します。

本論文の理論構成、実験に関しましては、石渡 亮伸氏（現、川崎製鉄）、前田 健太郎氏（現、新日本製鉄）、武術 康彦氏（修士課程在学中）、倉林 大輔氏（修士課程在学中）の卒業論文・修士論文が大変大いに参考になりました。また、同研究室の研究員であった木村 久幸氏（現、神奈川県立商工高校）、長谷川 潔氏（現、同研究室研究員、日立製作所）には、実験装置その他の面で大変お世話になりました。

また、梅田 和昇氏（博士課程在学中）、山口 博明氏（博士課程在学中）、沈 于思氏（博士課程在学中）、吉田 英一氏（博士課程在学中）他同研究室の学生の方々には、研究に於けるディスカッションその他の様々な点でお世話になりました。また、吉田氏には、本論文の構成をして頂きました。

筆者が大学の修士課程を修了してから、同研究室の助手に着任するまでの約2年間所属した新日本製鐵株式会社の中野 宣邦氏（現、エレクトロニクス研究部長）、有田秀稔氏（現、エレクトロニクス研究部主任研究員）、筆者が所属した画像処理グ

ループの方々には、大変お世話になりました。本論文は筆者が会社に所属していたころのテーマが基となったものでした。

この他にも、著者が助手として着任してから数多くの方々のご協力、ご援助を頂きました。中でも東京大学工学部精密機械工学科の先生方、職員の方々には、様々な面でご協力頂きました。特に、新井教授秘書の北山 理英さん（現、東京海上火災保険）、山田 真紀さん（現、お茶の水女子大学修士課程在学中）、岡田 直子さん、坂田 奈代子さんには、大変お世話になりました。技官の中山 茂子さんには研究室関係の仕事でお世話になりました。

本論文は、時には激励し、時には冷静にさせてくれた家族の存在なしには完成しませんでした。妻の理世、長男の励に感謝いたします。

平成6年2月15日

太田 順

< 欧文文献 >

[ARA1 89]

Arai, T., Ogata, H., Suzuki, T.:
Collision Avoidance among Multiple Robots Using Virtual Impedance.
Proc. IROS'89, 479/485(1989).

[ARA1 92]

Arai, T., Ota, J.:
Motion Planning of Multiple Mobile Robots.
Proc. IROS'92, 1761/1768(1992).

[ARA1 93a]

Arai, T., Yoshida, E., Ota, J.:
Information Diffusion by Local Communication of Multiple Mobile Robots.
Proc. 1993 IEEE Int. Conf. Syst. Man Cybern., 4, 535/540(1993).

[ARA1 93b]

Arai, T., Kimura, H., Ota, J.:
Real-Time Measuring System of Relative Position on Mobile Robot System.
Proc. Int. Symp. Ind. Robots, 931/938(1993).

[ARKIN 93]

Arkin, R. C., Balch, T., Nitz, E.:
Communication of Behavioral State in Multi-Agent Retrieval Tasks.
Proc. 1993 IEEE Int. Conf. Robotics and Automat., 588/594(1993).

[ASAMA 91]

Asama, H., Ozaki, K., Itakura, H., Matsumoto, A., Ishida, Y., Endo, I.:
Collision Avoidance among Multiple Mobile Robots Based on Rules and
Communication.
Proc. IROS'91, 1215/1220(1991).

[BARRAQUAND 89]

Barraquand, J., Langlois, B., Latombe, J.:
Robot Motion Planning with Many Degrees of Freedom and Dynamic Constraints.
Proc. 5th Int. Symp. Robotics Research, 435/444(1989).

[BROOKS 86]

Brooks, R. A.:
A Robust Layered Control System for a Mobile Robot,
IEEE J. Robotics and Automation, RA-2, 1, 14/23(1986).

[BUCKLEY 89]

Buckley, S. J.:
Fast Motion Planning for Multiple Moving Robots.
Proc. 1989 IEEE Int. Conf. Robotics and Automat., 322/326(1989).

[CALOUD 90]

Caloud, P., Choi, W., Latombe, J., Pape, C. L., Yim, M.:
In-door Automation with Many Mobile Robots.
Proc. IROS'90, 67/72(1990).

[CARRIOLI 91]

Carrioli, L.:
Unsupervised Path-Planning of Many Asynchronously Self-Moving Vehicles.
Proc. IROS'91, 555/559(1991).

[CHEN 87]

Chen, C. L., Lee, S. G., McGillerm, C. D.:
Task Assignment and Load Balancing of Autonomous Vehicles in a Flexible
Manufacturing System.
Proc. 1987 IEEE Int. Conf. Robotics and Automat., 1033/1039(1987).

[ERDMANN 86]

Erdmann, M., Lozano-Perez, T.:
On Multiple Moving Objects.
Proc. 1986 IEEE Int. Conf. Robotics and Automat., 1419/1424(1986).

[FUJIMURA 91]

Fujimura, K.:
A Model of Reactive Planning for Multiple Mobile Agents.
Proc. 1991 IEEE Int. Conf. Robotics and Automat., 1503/1509(1991).

[FUJIMURA 93a]

Fujimura, K.:
Motion Planning using Transient Pixel Representations.
Proc. 1993 IEEE Int. Conf. Robotics and Automat., 2, 34/39(1993).

[FUJIMURA 93b]

Fujimura, K.:
Motion Planning amidst Dynamic Obstacles in Three Dimensions.
Proc. IROS'93, 1387/1392(1993).

[PREUND 86]

Freund, E., Hoyer, H.:
Pathfinding in Multi-Robot Systems: Solution and Applications.
Proc. 1986 IEEE Int. Conf. Robotics and Automat., 103/111(1986).

[GROSSMAN 88]

Grossman, D. D.:
Traffic Control of Multiple Robot Vehicles.
IEEE J. Robotics and Automat., 4, 5, 491/497(1988).

[GUREWITZ 90]

Gurewitz, E., Fox, G., Wong, Y.:
Parallel Algorithm for One and Two-vehicle Navigation.
Proc. IROS'90, 51/58(1990).

[HOPCROFT 84]

Hopcroft, J.E., Schwartz, J.T., Sharir, M.:
On the Complexity of Motion Planning for Multiple Independent Objects;
PSPACE-Hardness of the "Warehouseman's Problem".
Int. J. Robotics Research, 3, 4, 75/88 (1984).

[ISO 90]

ISO/IEC 9506-1. Industrial automation systems - Manufacturing
Message Specification - Part 3: Companion Standard for Robotics.
First edition, 1990-10-15, (1990).

[KANT 88]

Kant, K., Zucker, S.:
Planning Collision-Free Trajectories in Time-Varying Environments:
A Two-Level Hierarchy.
Proc. 1988 IEEE Int. Conf. Robotics and Automat., 1644/1649 (1988).

[KATO 92]

Kato, S., Nishiyama, S., Takeno, J.:
Coordinating Mobile Robots by Applying Traffic Rules.
Proc. IROS'92, 1535/1541 (1992).

[KHATIB 86]

Khatib, O.:
Real-Time Obstacle Avoidance for Manipulators and Mobile Robots.
Int. J. Robotics Research, 5, 1, 90/98 (1986).

[KROGH 86]

Krogh, B.H., Thorpe, C.E.:
Integrated Path Planning and Dynamic Steering Control for Autonomous
Vehicles.
Proc. 1986 IEEE Int. Conf. Robotics and Automat., 1664/1669 (1986).

[KUROSU 93]

Kurosaki, K., Furuya, T., Soeda, M.:
Fuzzy Control of Group with a Leader and their Behaviors.
Proc. IROS'93, 1105/1109 (1993).

[LEE 90]

Lee, J., Bien, Z.:
Collision-Free Trajectory Control for Multiple Robots Based on Neural
Optimization Network.
Robotica, 8, 3, 185/194 (1990).

[LE PAPE 90]

Le Pape, C.:
A Combination of Centralized and Distributed Methods for Multi-Agent
Planning and Scheduling.

Proc. 1990 IEEE Int. Conf. Robotics and Automat., 488/493 (1990).

[LIU-P 89]

Liu, P., Fu, L.:
Planning and Scheduling in a Flexible Manufacturing System Using a Dynamic
Routing.

Proc. 1990 IEEE Int. Conf. Robotics and Automat., 1584/1589 (1990).

[LIU-Y 89]

Liu, Y., Kuroda, S., Naniwa, T., Noborio, H., Arimoto, S.:
A Practical Algorithm for Planning Collision-Free Coordinated Motion of
Multiple Mobile Robots.
Proc. 1989 IEEE Int. Conf. Robotics and Automat., 1427/1432 (1989).

[LUMELSKY 87]

Lumelsky, V.J. et al.:
Path-Planning Strategies for a Point Mobile Automaton Moving amidst
Unknown Obstacles of Arbitrary Shape.
Algorithmica, 2, 403/430 (1987).

[NOBORIO 91]

Noborio, H., Hashime, J.:
A Feasible Collision-Free and Deadlock-Free Path Planning Algorithm in a
Certain Workspace where Multiple Robots Move Flexibly.
Proc. IROS'91, 1074/1079 (1991).

[NOBORIO 92]

Noborio, H.:
A Collision-Free and Deadlock-Free Path-Planning Algorithm for Multiple
Mobile Robots without Mutual Communication.
Proc. IROS'92, 479/486 (1992).

[NOBORIO 93]

Noborio, H., Edashige, M.:
A Cooperative Path-Planning for Multiple Automata by Dynamic/Static
Conversion.
Proc. IROS'93, 1955/1962 (1993).

[NOREILS 90]

Noreils, F.R.:
Integrating Multi-Robot Coordination in a Mobile-Robot Control System.
Proc. IROS'90, 43/49 (1990).

[NOREILS 91]

Noreils, F.R.:
Coordinated Execution of Trajectories by Multiple Mobile Robots.
Proc. IROS'91, 1061/1067 (1991).

[NOREILS 92]

- Norells, F. R.:
Multi-Robot Coordination for Battlefield Strategies.
Proc. IROS' 92, 1777/1784 (1992).
- [O'DUNLAING 83]
O'Dunlaing, C., Sharir, M., Yap, C.:
Retraction: A new Approach to Motion Planning.
Proc. 15th ACM Symp. Theory and Computing, 207/220 (1983).
- [PARKER 93]
Parker, L. E.:
Designing Control Laws for Cooperative Agent Teams.
Proc. 1993 IEEE Int. Conf. Robotics and Automat., 582/587 (1993).
- [PARSONS 90]
Parsons, D., Canny, J.:
A Motion Planner for Multiple Mobile Robots.
Proc. 1990 IEEE Int. Conf. Robotics and Automat., 8/13 (1990).
- [PARTRIDGE 82]
Partridge, B. L.:
The Structure and Function of Fish Schools.
Scientific American, June, 90/99 (1982).
- [PREMVUTI 90]
Premvuti, S., Yuta, S.:
Consideration on the Cooperation of Multiple Autonomous Mobile Robots.
Proc. IROS' 90, 59/63 (1990).
- [SAITO 89]
Saito, M., Tsumura, T.:
Collision Avoidance between Mobile Robots.
Proc. IROS' 89, 473/478 (1989).
- [SAITO 91]
Saito, M., Tsumura, T.:
Collision Free Motion Control of Multiple Mobile Robots on Path Network.
Proc. IROS' 91, 1068/1073 (1991).
- [SCHWARTZ 83a]
Schwartz, J. T., Sharir, M.:
On the Piano Movers' Problem: II. General Techniques for Computing
Topological Properties of Real Algebraic Manifolds.
Advances in Applied Mathematics, 4, 298/351 (1983).
- [SCHWARTZ 83b]
Schwartz, J. T., Sharir, M.:
On the Piano Movers' Problem: III. Coordinating the Motion of Several
Independent Bodies: The Special Case of Circular Bodies Moving Amidst

- Polygonal Barriers.
Int. J. Robotics Research, 2, 3, 46/75 (1983).
- [SHARIR 90]
Sharir, M.:
Algorithmic Motion Planning in Robotics.
Computer, March, 9/19 (1990).
- [SHIBATA 93]
Shibata, T., Fukuda, T.:
Coordinative Behavior in Evolutionary Multi-Agent-Robot System.
Proc. IROS' 93, 448/453 (1993).
- [SUGIHARA 90]
Sugihara, K., Suzuki, I.:
Distributed Motion Coordination of Multiple Mobile Robots.
Proc. 5th IEEE Int. Symp. Intelligent Contr. 1990, 1, 138/143 (1990).
- [TOURNASSOUD 86]
Tournassoud, P.:
A Strategy for Obstacle Avoidance and Its Application to Multi-Robot
Systems.
Proc. 1986 IEEE Int. Conf. Robotics and Automat., 1224/1229 (1986).
- [TSUMURA 92]
Tsumura, T., Okubo, H., Komatsu, N.:
Directly Follow-up and/or Traced Control System of Multiple Ground
Vehicles.
Proc. IROS' 92, 1769/1775 (1992).
- [TUIJNMAN 88]
Tuijnman, F.:
Algorithms for the Avoidance of Collision between Autonomous Mobile Robots.
Proc. IROS' 88, 193/197 (1988).
- [WANG-P 88]
Wang, P. K. C.:
Dynamics of Interacting Multiple Autonomous Mobile Robots with Simple
Navigation Strategies.
Proc. IROS' 88, 179/185 (1988).
- [WANG-P 89a]
Wang, P. K. C.:
Interaction Dynamics of Multiple Mobile Robots with Simple Navigation
Strategies.
J. Robotics Systems, 6, 1, 77/101 (1989).
- [WANG-P 89b]
Wang, P. K. C.:

Navigation Strategies for Multiple Autonomous Mobile Robots,
Proc. IROS'89, 486/493(1989).

[WANG-91]

Wang, J.:
Fully Distributed Traffic Control Strategies for Many-AGV Systems,
Proc. IROS'91, 1199/1204(1991).

[WARREN 90]

Warren, C.W.:
Multiple Robot Path Coordination Using Artificial Potential Fields,
Proc. 1990 IEEE Int. Conf. Robotics and Automat., 500/505(1990).

[YEUNG 87]

Yeung, D., Bekey, G.A.:
A Decentralized Approach to the Motion Planning Problem,
Proc. 1987 IEEE Int. Conf. Robotics and Automat., 1779/1784(1987).

[YUTA 92]

Yuta, S., Pnevuti, S.:
Design and Implementation of Multiple Autonomous Robots' Cooperative
Behavior Based on Modest Cooperation,
Proc. 9th Int. Symp. Automat. Robotics in Contruction, 887/896(1992).

<和文文献>

[浅間 91]

浅間 一, 尾崎 功一, 板倉 広明, 松元 明弘, 石田 慶樹, 遠藤 勲:
通信及びビジュアルに基づく複数の移動ロボットの衝突回避,
第1回日本ロボット学会ロボットシンポジウム予稿集, 73/78(1991).

[浅間 92a]

浅間 一:
マルチエージェントロボットシステム研究の動向と展望,
日本ロボット学会誌, 10, 4, 428/432(1992).

[浅間 92b]

浅間 一:
複数の移動ロボットによる協調行動と群知能,
計測と制御, 31, 11, 1155/1161(1992).

[新井 92]

新井 民夫, 太田 順:
複数移動ロボット系の計画,
日本ロボット学会誌, 10, 4, 444/449(1992).

[新井 93a]

新井 民夫, 太田 順:
仮想的なインピーダンスを用いた複数移動ロボット系の動作計画,
日本ロボット学会誌, 11, 7, 1039/1046, (1993).

[新井 93b]

新井 民夫, 吉田 英一, 太田 順:
複数移動ロボット系の局所的な通信に関する研究,
日本ロボット学会誌, 論文査読中(1993).

[新井 94]

新井 民夫, 木村 久幸, 前田 健太郎, 太田 順, 梅田 和昇:
移動ロボット相互の位置・姿勢実時間計測システムの開発,
日本ロボット学会誌, 12, 3, 掲載予定(1994).

[石川 91]

石川 繁樹:
ファジィ制御を用いた自律型移動ロボットの誘導方式の検討,
日本ロボット学会誌, 9, 2, 149/161(1991).

[茨木 93]

茨木 俊秀:
岩波講座 応用数学1 離散最適化法とアルゴリズム,
岩波書店(1993).

[今福 92]

今福 道夫:

- 生物集団における群知能—動物集団の知能行動、
計測と制御, 31, 11, 1185/1189 (1992).
- [上田 93]
上田 完治:
知能化生産システム,
精密工学会誌, 59, 11, 1755/1760 (1993).
- [太田 93a]
太田 順, 新井 民夫, 倉林 大輔:
作業の性質を考慮したロボット群の動作計画,
日本ロボット学会誌, 論文査読中 (1993).
- [太田 93b]
太田 順, 武衛 康彦, 新井 民夫, 大隅 久, 陶山 毅一:
移動ロボットの協調による搬送制御,
日本ロボット学会誌, 論文査読中 (1993).
- [太田 94]
太田 順, 新井 民夫:
群秩序生成可能な追従戦略を用いた複数移動ロボット系の動作計画,
日本ロボット学会誌, 12, 3, 掲載予定 (1993).
- [小方 93]
小方 博之, 新井 民夫, 太田 順:
時変環境でユーザ仕様を考慮した移動ロボットの軌道計画法,
日本ロボット学会誌, 論文査読中 (1993).
- [喜多 76]
喜多 秀行:
船舶衝突危険度の解析モデル,
航海, 87, 27/35 (1976).
- [久保田 89]
久保田 孝, 橋本 秀紀, 原島 文雄:
ローカルプランニングによる移動ロボットの経路探索,
日本ロボット学会誌, 7, 4, 267/274 (1989).
- [小山 91]
小山 宏, 前田 陽一郎, 深海 悟, 高木 友博:
ファジィプロダクションシステムによる移動ロボットの障害物回避問題の検討,
日本ロボット学会誌, 9, 1, 75/78 (1991).
- [齊 91]
齊 丙辰, 大川 善邦:
移動ロボットの誘導のための立体構識,
日本ロボット学会誌, 9, 2, 129/136 (1991).
- [齊藤 90a]
齊藤 守, 津村 俊弘:

- 移動ロボット間における衝突回避,
システム制御情報学会論文誌, 3, 1, 1/7 (1990).
- [齊藤 90b]
齊藤 守, 津村 俊弘:
移動ロボット群における衝突回避—非線形計画法に基づく局所的解法—,
システム制御情報学会論文誌, 3, 8, 252/260 (1990).
- [坂根 93]
坂根 茂幸:
マルチロボットの行動,
日本ロボット学会誌, 11, 8, 1185/1191 (1993).
- [佐藤 92]
佐藤 知正:
群知能ロボットの地平と将来展望,
計測と制御, 31, 11, 1115/1118 (1992).
- [柴田 93]
柴田 崇徳, 福田 敏男:
Genetic Algorithmを用いた移動ロボットの最適経路計画,
(第2報, 複数移動ロボットののための利己的計画と協調的計画),
日本機械学会論文集 (C編), 59, 560, 1134/1141 (1993).
- [鈴木 90]
鈴木 寿, 有本 卓:
時変環境における点状の移動ロボットのペイント手続きを利用した再帰的軌道計画
法,
日本ロボット学会誌, 8, 5, 513/522 (1990).
- [登尾 92]
登尾 啓史, 橋目 謙:
可視領域をもつ移動ロボットの未知空間における実地的なパスプランニング
アルゴリズム,
日本ロボット学会誌, 10, 3, 378/384 (1992).
- [武野 86]
武野 純一, 柿倉 正義:
移動ロボットの移動障害物に対する衝突回避問題について,
日本ロボット学会誌, 4, 5, 499/503 (1986).
- [津村 88]
津村 俊弘, 橋本 雅文, 藤原 直史:
レーザとコーナーキューブを利用した3次元移動体の能動位置計測法,
日本ロボット学会誌, Vol. 6, No. 1, 26/34 (1987).
- [津村 92]
津村 俊弘:
複数ピークルにおける協調制御,

1992年度機械学会ロボティクス・メカニクス講演会, B, 265/268(1992).

[ドウスクリ 92]

モハメッド アリ ドウスクリ, 三宮 信夫:
シミュレーションによる魚群行動の成群機構に関する考察,
システム制御情報学会論文誌, 5, 12, 521/523(1992).

[長田 92]

長田 正:
群知能ロボット総論,
計測と制御, 31, 11, 1119/1124(1992).

[日高 92]

日高 敏隆:
動物の行動と社会,
放送大学教材, (1992).

[平井 77]

平井 一正, 樽井 賢治:
パニック時の群衆の行動とそのシミュレーション,
システムと制御, 21, 6, 331/338(1977).

[藤原 90]

藤原 直史, 米澤 徹, 津村 俊弘:
画像情報を利用した移動体の位置姿勢計測法,
システム制御情報学会論文誌, 3, 4, 110/117(1990).

[星野 93]

星野 力:
人工生命の現状と将来への期待,
計測と制御, 32, 8, 677/683(1993).

[前田 88]

前田 陽一郎, 竹垣 盛一:
ファジィ推論を用いた移動ロボットの動的障害物回避制御,
日本ロボット学会誌, 6, 6, 518/522(1988).

[美多 84]

美多 勉:
デジタル制御理論,
昭晃堂, (1984).

[油田 92]

油田 信一:
複数の自律移動ロボットの協調行動,
日本ロボット学会誌, 10, 4, 433/438(1992).

[米澤 93]

人工生命 コンピュータに宿る生命,
bit, 25, 7, 15/25(1993).

[劉 92]

劉 雲輝, 有本 卓:
局所センサ情報を用いた未知環境における複数の移動ロボットの運動計画法,
日本ロボット学会誌, 10, 2, 216/225(1992).

[JIS 87]

JIS 用語辞典機械編第2版,
日本規格協会(1987).

1. 研究之動機與目的：本研究旨在探討我國中小學教師之專業發展現況，並分析影響教師專業發展之因素，以期為教育政策之制定提供參考。

2. 研究之範圍與對象：本研究之範圍為我國中小學教師，對象則為不同地區、不同學段之教師。

3. 研究之方法：本研究採用問卷調查法，並輔以深度訪談，以獲取教師之真實想法與感受。

4. 研究之結果與討論：研究結果顯示，我國中小學教師之專業發展現況不容樂觀，存在諸多問題，如培訓機會不足、專業發展動力不足等。討論部分則分析了造成這些問題之原因，並提出了相應之建議。

5. 研究之結論：本研究結論認為，應加強對中小學教師之專業發展支持，包括提供充足之培訓機會、建立專業發展激勵機制等，以促進教師專業素養之提升。

6. 研究之貢獻：本研究之貢獻在於，首次系統性地調查了我國中小學教師之專業發展現況，並為教育政策之制定提供了實證依據。

7. 研究之限制：本研究之限制在於，樣本僅限於我國中小學教師，且採用問卷調查法，可能存在一定之偏差。

8. 研究之建議：本研究建議，教育政策制定者應重視教師專業發展之重要性，加大對教師專業發展之投入，為教師提供良好之專業發展環境。

9. 研究之展望：本研究之展望在於，未來可進一步擴大樣本範圍，並採用更多樣化之研究方法，以更全面地探討教師專業發展之問題。

10. 研究之參考文獻：本研究參考了國內外相關領域之研究文獻，包括教師專業發展理論、調查方法等。

1. 研究之動機與目的：本研究旨在探討我國中小學教師之專業發展現況，並分析影響教師專業發展之因素，以期為教育政策之制定提供參考。

2. 研究之範圍與對象：本研究之範圍為我國中小學教師，對象則為不同地區、不同學段之教師。

3. 研究之方法：本研究採用問卷調查法，並輔以深度訪談，以獲取教師之真實想法與感受。

4. 研究之結果與討論：研究結果顯示，我國中小學教師之專業發展現況不容樂觀，存在諸多問題，如培訓機會不足、專業發展動力不足等。討論部分則分析了造成這些問題之原因，並提出了相應之建議。

5. 研究之結論：本研究結論認為，應加強對中小學教師之專業發展支持，包括提供充足之培訓機會、建立專業發展激勵機制等，以促進教師專業素養之提升。

6. 研究之貢獻：本研究之貢獻在於，首次系統性地調查了我國中小學教師之專業發展現況，並為教育政策之制定提供了實證依據。

7. 研究之限制：本研究之限制在於，樣本僅限於我國中小學教師，且採用問卷調查法，可能存在一定之偏差。

8. 研究之建議：本研究建議，教育政策制定者應重視教師專業發展之重要性，加大對教師專業發展之投入，為教師提供良好之專業發展環境。

9. 研究之展望：本研究之展望在於，未來可進一步擴大樣本範圍，並採用更多樣化之研究方法，以更全面地探討教師專業發展之問題。

10. 研究之參考文獻：本研究參考了國內外相關領域之研究文獻，包括教師專業發展理論、調查方法等。

研究業績書

1. 研究之動機與目的：本研究旨在探討我國中小學教師之專業發展現況，並分析影響教師專業發展之因素，以期為教育政策之制定提供參考。

2. 研究之範圍與對象：本研究之範圍為我國中小學教師，對象則為不同地區、不同學段之教師。

3. 研究之方法：本研究採用問卷調查法，並輔以深度訪談，以獲取教師之真實想法與感受。

4. 研究之結果與討論：研究結果顯示，我國中小學教師之專業發展現況不容樂觀，存在諸多問題，如培訓機會不足、專業發展動力不足等。討論部分則分析了造成這些問題之原因，並提出了相應之建議。

5. 研究之結論：本研究結論認為，應加強對中小學教師之專業發展支持，包括提供充足之培訓機會、建立專業發展激勵機制等，以促進教師專業素養之提升。

6. 研究之貢獻：本研究之貢獻在於，首次系統性地調查了我國中小學教師之專業發展現況，並為教育政策之制定提供了實證依據。

7. 研究之限制：本研究之限制在於，樣本僅限於我國中小學教師，且採用問卷調查法，可能存在一定之偏差。

8. 研究之建議：本研究建議，教育政策制定者應重視教師專業發展之重要性，加大對教師專業發展之投入，為教師提供良好之專業發展環境。

9. 研究之展望：本研究之展望在於，未來可進一步擴大樣本範圍，並採用更多樣化之研究方法，以更全面地探討教師專業發展之問題。

10. 研究之參考文獻：本研究參考了國內外相關領域之研究文獻，包括教師專業發展理論、調查方法等。

【投稿論文】

新井 民夫, 太田 順:

仮想的なインピーダンスを用いた複数移動ロボット系の動作計画,
日本ロボット学会誌, 11, 7, 1039/1046(1993).

新井 民夫, 木村 久幸, 前田 健太郎, 太田 順, 梅田 和昇:

移動ロボット相互の位置・姿勢実時間計測システムの開発,
日本ロボット学会誌, 12, 3, 掲載予定(1994).

太田 順, 新井 民夫:

群秩序形成可能な追従戦略を用いた複数移動ロボット系の動作計画,
日本ロボット学会誌, 12, 3, 掲載予定(1994).

新井 民夫, 吉田 英一, 太田 順:

複数移動ロボット系の局所的な通信に関する研究,
日本ロボット学会誌, 論文査読中.

太田 順, 武衛 康彦, 新井 民夫, 大隅 久, 陶山 毅一:

複数の移動ロボットの協調による搬送制御,
日本ロボット学会誌, 論文査読中.

太田 順, 新井 民夫, 倉林 大輔:

作業の性質を考慮したロボット群の動作計画,
日本ロボット学会誌, 論文査読中.

小方 博之, 新井 民夫, 太田 順:

時変環境でユーザ仕様を考慮した移動ロボットの軌道計画法,
日本ロボット学会誌, 論文査読中.

太田 順:

複数移動ロボット系における環境適応型動作計画器の設計,
日本ロボット学会誌, 論文査読中.

【講演論文】

石渡 亮伸, 新井 民夫, 太田 順:

複数の移動ロボットの運動制御に関する研究(第3報)一制御パラメータの設計一,
第9回日本ロボット学会学術講演会予稿集, 603/604(1991).

石渡 亮伸, 新井 民夫, 太田 順:

車輪式移動ロボットの運動制御,
1992年精密工学会春季大会学術講演会講演論文集, 191/192(1992).

太田 順, 前田 健太郎, 新井 民夫, 大隅 久:

画像を用いた移動ロボット間の通信システム,
1992年精密工学会春季大会学術講演会講演論文集, 193/194(1992).

太田 順, 武衛 康彦, 新井 民夫, 大隅 久:

複数の移動ロボットの協調による搬送制御,
1992年精密工学会春季大会学術講演会講演論文集, 199/200(1992).

Arai, T., Ota, J.:

Motion Planning of Multiple Mobile Robots,
Proc. 1992 IEEE/RSJ Int. Conf. on Intel. Robots and Systems(IROS'92),
1761/1768(1992).

新井 民夫, 太田 順:

複数移動ロボット系の計画,
日本ロボット学会誌, Vol.10, No.4, 444/449(1992).

Arai, T., Ota, J.:

Motion Planning of Multiple Mobile Robots using Robot Groups,
Proc. Int. Symp. Distributed Autonomous Robotics Systems,
103/108(1992).

小橋 隆昭, 大隅 久, 太田 順, 森 素子, 武衛 康彦, 新井 民夫:

マニピュレータを搭載した移動ロボットの軌道生成,
1992年精密工学会秋季大会学術講演会講演論文集, 965/966(1992).

新井 民夫, 田代 健治, 太田 順:

移動ロボット用分散地図の生成と利用,
1992年精密工学会秋季大会学術講演会講演論文集, 967/968(1992).

新井 民夫, 田代 健治, 太田 順:

自立分散型移動ロボット用環境情報システムの構築,
第10回日本ロボット学会学術講演会予稿集, 279/280(1992).

木村 久幸, 前田 健太郎, 太田 順, 新井 民夫:

移動ロボット相互の位置・姿勢計測システムの開発,
第10回日本ロボット学会学術講演会予稿集, 281/282(1992).

吉田 英一, 新井 民夫, 太田 順:

複数ロボット系における協調作業のための群形成,
第10回日本ロボット学会学術講演会予稿集, 301/302(1992).

太田 順, 新井 民夫:

複数の移動ロボットの運動制御に関する研究(第4報)一追従戦略の導入一,
第10回日本ロボット学会学術講演会予稿集, 305/306(1992).

武衛 康彦, 太田 順, 大隅 久, 新井 民夫, 陶山 毅一:

複数の移動ロボットの協調による搬送作業の実現,
第10回日本ロボット学会学術講演会予稿集, 561/562(1992).

陶山 毅一, 新井 民夫, 太田 順:

天井蛍光灯を利用した移動ロボットの位置修正,
第10回日本ロボット学会学術講演会予稿集, 933/934(1992).

太田 順, 倉林 大輔, 新井 民夫:

複数移動ロボット系におけるロボット群の動作計画,
1993年精密工学会秋季大会学術講演会講演論文集, 401/402(1993).

木村 久幸, 太田 順, 新井 民夫:

画像を用いた移動ロボット間の通信システム(第2報)看板による通信の実現,
1993年精密工学会春季大会学術講演会講演論文集, 897/898(1993).

吉田 英一, 新井 民夫, 太田 順, 木村 久幸:

複数移動ロボット間の分散型通信に関する研究,

1993年精密工学会春季大会学術講演会講演論文集, 899/900(1993).

新井 民夫, 田代 健治, 太田 順, 山口 博明:

移動ロボット用分散地図の生成と利用(第2報)一環境の動的変化に対する地図の更新一,

1993年精密工学会春季大会学術講演会講演論文集, 1099/1100(1993).

Ota, J., Arai, T.:

Motion Planning of Multiple Mobile Robots Using Dynamic Groups,
Proc. 1993 IEEE Int. Conf. Robotics and Automat., 2, 28/33(1993).

倉林 大輔, 太田 順, 新井 民夫:

移動ロボット群の動作計画,

日本機械学会ロボティクス・メカトロニクス講演会'93講演論文集, 950/953(1993).

吉田 英一, 新井 民夫, 太田 順:

分散型通信による複数移動ロボットの協調,

日本機械学会ロボティクス・メカトロニクス講演会'93講演論文集, 954/957(1993).

Ota, J., Arai, T., Kurabayashi, D.:

Dynamic Grouping in Multiple Mobile Robots System,
Proc. 1993 IEEE/RSJ Int. Conf. on Intel. Robots and Systems(IROS'93),
1963/1970(1993).

太田 順, 新井 民夫:

追従戦略を用いた複数移動ロボット系の群形成,

1993年精密工学会秋季大会学術講演会講演論文集, 3, 859/860(1993).

Arai, T., Yoshida, E., Ota, J.:

Information Diffusion by Local Communication of Multiple Mobile Robots,
Proc. 1993 IEEE Int. Conf. Syst. Man, Cybern., 5, 535/540(1993).

Arai, T., Kimura, H., Ota, J.:

Real-Time Measuring System of Relative Position on Mobile Robot System,
Proc. Int. Symp. Ind. Robots, 931/938(1993).

太田 順, 新井 民夫:

複数の移動ロボットの運動制御に関する研究(第5報)一設計パラメータの定量的評価一,

第11回日本ロボット学会学術講演会予稿集, 1, 1/2(1993).

武衛 康彦, 太田 順, 大隅 久, 新井 民夫, 陶山 毅一:

複数の移動ロボットの協調による搬送作業の実現(第2報:ローカルな情報のみを用いた協調手法の提案),

第11回日本ロボット学会学術講演会予稿集, 2, 103/104(1993).

倉林 大輔, 太田 順, 新井 民夫:

動的な群形成による移動ロボット群の動作計画,

第11回日本ロボット学会学術講演会予稿集, 2, 411/414(1993).

吉田 英一, 新井 民夫, 太田 順:

局所的な通信を用いた複数移動ロボットの協調,

第11回日本ロボット学会学術講演会予稿集, 2, 895/898(1993).

陶山 毅一, 新井 民夫, 太田 順:

案内ロボットの開発,

第11回日本ロボット学会学術講演会予稿集, 3, 1319/1320(1993).

