

Title	通信損失が存在する場合の群ロボットの位置推定の効率化に関する研究
Author(s)	神ノ尾, 淳
Citation	
Issue Date	2016-09
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/13829
Rights	
Description	Supervisor: 丁 洛榮, 情報科学研究科, 博士

博士論文

通信損失が存在する場合の群ロボットの位置推定の
効率化に関する研究

神ノ尾 淳

主指導教員 丁 洛榮

北陸先端科学技術大学院大学

情報科学研究科

平成28年9月

要旨

本研究では、複数のロボットが協調して動作することで所望の機能を実現する群ロボットにおいて、任意に配置された初期位置から目標形状に収束させる formation 制御に注目し、ロボット間の通信に損失が生じた場合の効率的かつ高精度な推定手法の開発を試みた。

本論文では formation 制御の中でも特に正三角形格子を目標形状とする制御方法を対象とする。まずは任意の初期位置に配置されたロボットが目標となる正三角形格子を形成する際に、ある時点においてどの程度収束しているかを測るための評価指標を提案し、その有効性について検証した。

続いて、今回対象とする制御方法においては、ロボットが目標形状に向かって指数関数的に収束しているという事実に着目し、ロボット間の通信が一定確率で損失した場合に、失われた位置情報を過去の状態から推定するための手法として指数関数を用いた挿手法を新たに提案し、その数学的特性について議論すると共に、その有効性についてシミュレーションによる検証を行った。更にこの指数関数を用いた外挿式に対して改良を行い、その推定精度を向上させた。

最後に、正三角形格子を形成するアルゴリズムにおいては、ロボットが1階の微分方程式に従うことを仮定していたが、今回提案した指数関数を用いた外挿手法が一般にも適用できることを示すため、ロボットの挙動が2階、 \dots 、 n 階の微分方程式に従う場合についても、その数学的な安定性を示した。

Abstract

In this research, in swarm robotics which many robots realize the desired function by cooperating their motions each other, we focus on the formation control which makes many robots to converge to desirable formation from randomly deployed initial position, and address the problem “how can we estimate robots’ positions efficiently and accurately when communication loss among robots has occurred?”

In this thesis, we firstly target the control method especially for equilateral triangle lattice as target configuration in formation control. At first, we proposed the evaluation indices to measure the degree of convergence of robots from randomly deployed initial positions to the target equilateral triangle formation, and validate their efficiency.

Next, we focus on the fact that robots converge to target configuration exponentially intended control method in this time, we proposed novel extrapolating method using exponential function to estimate positional information from past states when communications among robots are lost, discussed its mathematical property and estimation accuracy, and validated its efficiency by simulations. Moreover, we improved this extrapolating method using exponential function, and succeeded to improve its estimation accuracy.

Subsequently, at first although we assumed that robots obey first order differential equation in the algorithm which forms equilateral triangle lattice, but we showed that the extrapolating method using exponential function proposed in this time is applicable to the case robots obey second order, ..., n-th order differential equation.

key words: swarm robot, formation control, triangle formation, communication loss recovery, exponential extrapolation

目次

1	緒言	1
1.1	研究の背景	1
1.2	既存の研究	2
1.3	研究の目的	3
1.4	本論文の構成	4
2	関連研究	5
2.1	群ロボットにおける基礎的研究	5
2.2	群ロボットの formation 制御に関する研究	9
2.3	既存の研究における問題点	10
3	問題設定	14
3.1	群ロボットにおける正三角形格子への収束アルゴリズム	14
3.1.1	表記法及び定義	14
3.1.2	収束アルゴリズム	16
3.1.3	アルゴリズムの収束性	19
3.2	本論文で取り組む問題の定義	19
4	群ロボットの目標形状への収束度の評価指標	22
4.1	Delaunay 三角形を用いた群ロボットの目標形状への収束度の評価指標	23
4.1.1	Delaunay 三角形	23
4.1.2	群ロボットの目標形状への収束度の評価指標	24
4.2	提案した評価指標の定量的評価	26
4.3	提案した評価指標の問題点及びその改良	29
4.4	提案した評価指標の問題点及びその改良	31
4.5	改良した評価指標の定量的評価	32

5	単純な関数を用いたロボット位置の外挿	34
5.1	0次ホールド・線形外挿・2次外挿の概要	34
5.1.1	0次ホールド	34
5.1.2	線形外挿	35
5.1.3	2次外挿	35
5.2	EKF(Extended Kalman Filter) 及びPF(Particle Filter) の概要	36
5.2.1	EKF の原理	37
5.2.2	PF の原理	38
5.3	0次ホールド・線形外挿・2次外挿を用いたロボットの位置推定のシミュレーション結果	40
5.4	速度外挿の概要	44
5.4.1	速度0次ホールド	46
5.4.2	速度線形外挿	47
5.5	速度外挿を用いたシミュレーション結果	47
6	指数関数を用いたロボット位置の外挿	50
6.1	指数関数を用いた外挿式 (EE)	50
6.2	指数関数を用いた外挿式 (EE) の数学的特性	53
6.3	指数関数を用いた外挿式 (EE) の推定精度	55
6.4	Cramér-Rao の下界式	56
6.5	EE によるロボット位置推定のシミュレーション結果	60
7	指数関数を用いたロボット位置の外挿手法の改良	64
7.1	指数関数を用いた外挿式の改良	64
7.2	EEE を用いたロボット位置推定のシミュレーション結果	66
7.3	線形関数を加えた指数関数を用いた外挿式	69
7.4	EEF 及びEEEF によるロボット位置推定のシミュレーション結果	72
8	高階の微分方程式で記述されるシステムへの指数外挿の応用	77
8.1	指数外挿式 (EE) の2階の線形微分方程式に対する適用	77
8.2	指数外挿式 (EE) の n 階の線形微分方程式に対する適用	81

9	まとめ	86
10	結言	89
10.1	まとめ	89
10.2	今後の課題	90
	謝辞	92
	参考文献	94
	本研究に関する発表論文	104

第 1 章

緒言

1.1 研究の背景

近年、ロボット技術の進歩に伴い、ロボットの活躍する場はますます広がっている。当初産業用として活躍していたロボットは最近では家庭にまで進出し、従来人間が行ってきた作業の多くを担うようになってきている。

従来のロボット研究においては、ロボットに要求される様々な役割を実現するために、単一台のロボットに多様な機能を持たせるという方向で開発が行われてきた。しかし、このような多機能な単一台のロボットにおいては、その機構や構成部品が複雑となることから、何らかの原因で不具合が生じた場合に、その修理に手間が掛かるという問題がある。特に近年ロボットの活躍の場として注目を集めている、原子力発電所のような人間の立ち入ることのできない危険場所で作業を行うロボットにとって、こうした故障は致命的であるとも言える。

こうした中、従来の単一台のロボットではなく、多数のロボットを協調して動作させることにより所望の機能を実現させるという群ロボットが注目を集めている [1]-[3]。群ロボットとは比較的単純な機能を持ったロボットを多数台用い、これらのロボットを既定のアルゴリズムに則って動作させることにより、操作者の意図する動きを実現させようとするものである。この群ロボットの特徴としては、多数台のロボットを同時に使用することから、一部のロボットが何らかの原因により動作できなくなった場合においても、全体としての動きに大きな影響を与えることなく作業を行うことができるという点や、各ロボットには機能を制限したあ

まり性能の高くないロボットを利用することから、比較的安価であるといった利点が挙げられる。

こうした群ロボットにおいては、各ロボットが協調して1つの動作を達成するために、ロボット同士が互いに情報を交換し合う必要がある。ロボットが互いの情報を認識するための手法としては無線通信を用いる方法、赤外線通信を用いる方法、センサを用いる方法など様々な種類が存在するが、現場での使用を考えた場合にこれらの方法はいずれも完全な情報交換は保証されておらず、例えば障害物の存在によりセンサが感知できない、あるいは通信が遮断されるといった問題点が存在する。

次節以降においては、群ロボットにおける既存の研究について概観すると共に、本論文において取り組むべき問題について述べる。

1.2 既存の研究

群ロボットとは蟻や魚、鳥の群れなどに代表される、特定の指導者を持たずに群れ全体として1つの目標となる行動を実現する生物の動きに触発されて提案されたロボットの研究領域である。単純な知能を持つ多数のロボット群として特徴付けられる [4]。群ロボットは単体としては簡素な機能しか持たないことからコストパフォーマンスに優れており、また1台のロボットの故障が全体に与える影響が小さいことから、耐故障性が高いことでも知られている。群ロボットを制御する手法は多数考案されてきているが、最近では制御理論における Glocal 制御 [5] を応用した手法も注目されている。Glocal 制御とはロボット群の制御をいくつかの層 (layer) に分け、各層における構成要素 (ロボット群) の動きを制御することにより群れ全体の振る舞いをも制御するものである [6]-[10]。

群ロボットの制御における研究分野の1つに、formation 制御が挙げられる。formation 制御とはランダムな初期位置に配置されたロボットが、互いの位置情報を交換しながら既定の制御手法に則って動作することにより、所望の形状を形成させるというものである。ロボットが目標とする形状としては、正三角形格子を生成するもの [11]-[13] や円を生成するもの [14][15] などが既に研究されている。更に、現実的な場面での動作を考慮して、ロボット間に障害物が存在する場合を想

定し、これらの障害物を避けながら formation 制御を実現する研究も行われている [16]-[20]。

しかし、従来の研究においてはロボット間の通信は常に保障されていることを仮定しており、ロボット間の通信に何らかの理由で障害が発生し、情報の伝達が行われなかった場合についての研究はあまり行われていない。よって、こうした問題を解決するために、次節において本論文において取り組むべき問題について述べる。

1.3 研究の目的

本研究においては、群ロボットの formation 制御においてロボット間の通信に損失がある場合に、失われた情報を外挿するための新たな手法を提案する。ある時点で得られた状態を得ることができなかった場合、失われた情報を外挿するための単純な手法としては、1ステップ前の状態をそのまま保持する0次ホールド、過去の状態から1次関数を用いて現在の状態を求める線形外挿、2次関数を用いて求める2次外挿、…、 n 次関数を用いて求める n 次外挿などの手法が古くから存在するが、これらの手法は計算量をあまり必要とせず簡単に計算ができるという反面、その推定精度は保証されていない。一方、制御理論の分野においては、状態推定のためにカルマンフィルタ(KF)をはじめとする種々のフィルタが考案されている。これらのフィルタはその推定精度について理論的な保証がされている一方で、多くの計算量を必要とすることから、一般に限られた計算能力しか持たず複雑な計算を短時間で行うことのできない群ロボットに実装することは難しいと考えられる。こうしたことから、本研究においては、群ロボットの formation 制御において、既存の外挿手法のようにあまり計算量を必要とせず、かつ理論的なフィルタのように高い精度で通信損失時に現在の状態を推定できる新たな外挿手法を提案するために、群ロボットの formation 制御におけるロボットの運動学的な特性に着目した指数関数を用いた新たな外挿法を考案し、その有効性について検証を行うことを目的とする。

1.4 本論文の構成

本論文は以下の通り構成される。まずは2章においてこれまでに行われてきた群ロボットについての研究について概観すると共に、その問題点について述べる。3章においては、群ロボットの formation 制御においてロボットが正三角形格子を形成するためのアルゴリズムについて記述すると共に、本論文で取り組む問題について設定する。4章においては本論文で取り扱う、formation 制御において目標形状として正三角形格子を考えた場合に、ある時点における各ロボットの位置が目標とする形状にどの程度近づいているか（または離れているか）を定量的に評価するための評価指標を導入し、その有効性について検証を行う。5章においては、制御理論の分野における状態推定手法として知られている EKF(拡張カルマンフィルタ) 及び PF(粒子フィルタ) について概説すると共に、単純な外挿手法として一般的に知られている 0 次ホールド、線形外挿、2 次外挿の 3 種類の手法を用いて通信損失時にロボット位置を外挿した場合、及びこれらを改良した速度外挿を用いて外挿した場合について、その推定精度をこれらの理論的なフィルタと比較し議論する。6章においては、本論文において提案する指数関数を用いた外挿式について導入し、その数学的特性を解析すると共に、推定誤差に関する理論的な下限値として知られている Cramér-Rao の下界式について導入し、この外挿式を用いて通信損失時のロボットの動きをシミュレートし、Cramér-Rao の下界式より得られた下限値と比較することによりその効果について検証を行う。7章においては、6章で導入した指数関数を用いた外挿式に改良を加えた新たな外挿式を提案し、その有効性についてシミュレーションによる検証を行う。8章においては、指数外挿式の適用範囲をこれまでの 1 階の微分方程式から 2 階、 \dots 、 n 階の微分方程式に拡大し、その安定性について数学的な検証を行う。9章では、本論文において提案した新たな外挿手法についての考察を行う。最後に、10章において本論文の結論を示し、今後の展望について述べる。

第 2 章

関連研究

本章においては、群ロボットの分野においてこれまでに研究されてきた内容について概観すると共に、その問題点について記述する。

2.1 群ロボットにおける基礎的研究

自然界における生物の社会的な振る舞いとしては、蟻の集団、鳥の群れ、魚の一群などの例が見られる。こうした生物は一個体の持つ能力は低く、単体で行動することは困難な場合が多いが、集団となって1つのタスクを実行することにより、単体では成し得ないような振る舞いを行うことができる。群ロボットとは、こうした蟻や蜂のような高い知能を持たない昆虫などが、特定の指導者なしに群れ行動により全体として1つの共通の目標となる行動を達成する、あるいは鳥が決まった隊形を維持しながら空を飛行するといった様子から発想を得て研究が開始された。当初、群行動を行う生物の振る舞いは、各個体が協調と通信のための固有の識別子 (ID) を持つ集中型のネットワークにより成されていると考えられていた。しかし、その後の研究によりこうした生物における群行動は分散型であり、各個体は固有の識別子 (ID) を持たず、一定の規則に基づいて局所的な情報を元に行動していることが明らかにされた。文献 [21] においては「社会的な昆虫」と呼ばれる蟻や蜂などが、特定の指導者なしに群れ行動を実現するメカニズムについて述べられている。こうした動物の群れ行動には、ロバスト性、柔軟性、拡張性の3つの特性があるとされている。

1．ロバスト性

群れ行動におけるロバスト性とは、個々の個体の持つ能力が低いのに関わらず、群れが一部の個体の故障や外乱といった要因にも影響を受けず全体としての行動を維持し続けることができることを示す。例えば、家の中に侵入してきた蟻の一部を取り除いたとしても、蟻の群れ全体の行動には大きな影響を及ぼさないということである。このロバスト性はいくつかの要因によって生成される。1つはそのシステムの持つ冗長性である。これは一部の個体が故障又は取り除かれても、その個体の役割を他の個体が代替できるということである。2つめは、分散型の協調システムということである。これは、システムの一部が破壊されてもそれがシステム全体の挙動には大きな影響を与えないということである。3つ目は、各個体が持つ単純さである。これは各個体が比較的単純な機能しか有しないため、故障する可能性が低いということである。

2．柔軟性

柔軟性とは、その群れが置かれた状況に応じて様々な仕事に適應できる能力のことである。例えば蟻の群れ行動を考えた場合、それぞれの蟻は獲物を探す、獲物を捕まえる、捕まえた獲物を巣まで運ぶといった異なる仕事を分担して行っている。このように、その環境に応じて適切な仕事を選択して実行できることが群れ行動における柔軟性であると言える。

3．拡張性

拡張性とは、その群れの大きさを自由に変化させることのできる能力のことである。群れを構成する構成要素の数は、個体の故障によりその数が減少したり、また新たな個体を追加することによりその数が増加したりする。こうした群れの構成要素の数の増減に関わらず、群れ全体として同じ動きを実現できるのが拡張性である。

こうした群知能に関する研究は、文献[22]において初めて行われた。群ロボティクスに関する定義は以下のようなされる。

定義 1

「群ロボティクスとは、比較的単純な機能を持った多数のロボットがロボット間やロボットと環境との間の局所的な相互作用によってどのようにして望ましい動きを実現できるかという研究分野である」

複数台のロボットが行う動作についての研究は、協調ロボティクス [23]、分散型ロボティクス [24]、ロボット共同体 [25] 等様々な研究が行われているが、ここでは群ロボットにおける研究をこれらの研究と区別するために、以下のような基準を設ける。

1．群ロボットの自律性

自律型のロボットとは、周囲の世界と物理的に相互作用を行いながら行動できるロボットのことである。例えば、文献 [26] で示されるようなセンサネットワークは分散型のセンサ要素から成り立っているものの、物理的な行動能力を持たないため、群ロボットシステムと考えることはできない。逆に、文献 [27]、[28] にあるような変形ロボットシステムは群ロボットシステムであると言える。

2．群ロボットの多数性

群ロボティクスにおける研究では、「ロボット群」の協調に関連がある必要がある。このため、少数台のロボットにのみ適用でき拡張性を持たない研究は群ロボティクスの分野に入れることはできない。最低でも 10 台～20 台のロボットで構成される群れを「群ロボット」として扱う必要がある。

3．ロボット群の同質性

群ロボットにおいて、群れを構成するロボットは比較的「同質」であり、その台数は多数である必要がある。近年では性質の異なるロボット同士を協調して動作させる研究も見られるが、その場合にはロボットが同質である場合に比べてタスクを実現することが非常に難しくなる。

あるシステムにおける各ロボットの同質性について論じることは難しいが、文献 [29] においてはその測定基準が与えられている。

4．単純な機能しか有しないロボット

群ロボットにおいては、各ロボットはそれ自身では単純な機能しか有しない比較的能力の低いこととして扱われるのが一般的である。つまり、各ロボット単体ではその仕事を実行することが難しく、またロボット群の配置によりその仕事を扱う性能やロバスト性が改良されることを前提としている。文献 [30] においては、単純な移動式ロボットを用いて非一様に分布する信号を発する目標物の発見という仕事において、群ロボットの使用によりその性能が劇的に向上することが報告されている。

即ち、複数台のロボットが協調して作業を行う場合でも、1. ロボットの台数が少ない、2. 集中型の制御方式である、といった条件が満たされる場合には、そのシステムのことを群ロボットとは呼ばない。

このような利点を持つ群ロボットは、生産性という観点からも利点を持っている。近年の産業技術の進歩に伴い、比較的単純な機構やセンサを持つ機器の生産コストは増々低下してきている。群ロボットに要求される機能はそれほど高いものではないため、こうした技術の進歩に伴い群ロボット1台当たりの生産コストはどんどん下がっていき、安価で利用することが可能となってきた。例えばK-Team社のKilobotは搭載するプロセッサやメモリ、センサなどの機能を最低限のものに抑えているため、1台当たりのコストが非常に安価となっている。群ロボットの現実的な応用として想定される場面をいくつか紹介する。

1. 領域被覆のタスク

群ロボットが得意とする仕事の一つに、領域被覆と呼ばれる、ある領域内を群ロボットで均一に被覆するというタスクが挙げられる。この領域被覆はある領域内における監視や異常の発見といった問題に対処できる。従来の固定式のセンサでも領域内の異常を感知することはできるが、群ロボットの場合は更に異常を発見した場合に、その場所にロボットを集中して再配置することで問題を効率的に解決することができるという利点が挙げられる。

2. 危険場所でのタスク

人間が立ち入ることのできないような危険場所におけるタスクも、群ロボットの利用場所として挙げられる。危険場所においては人間が立ち入ることができない場合が多いため、その場の状況に応じて柔軟に適應できる群ロボットはその利点を発揮することができる。また、こうした危険場所においては故障時の修理も困難であることが多いため、高い耐故障性を持った群ロボットの利用は有効である。

3. 状況に応じてロボット台数の変化を伴うタスク

群ロボットにおいて応用が期待される領域の1つに、置かれた状況によって必要とされるロボット台数が変化するようなタスクが挙げられる。例えばオイルタンクの監視においては、通常時には監視のための少数台のロボットしか必要とされないが、一度オイル漏れが発生するとその修復のために多数のロボットが必要となる。このように状況によって必要とされるロボットの台数が変化するタスク

に対応できるのが、柔軟性を持つ群ロボットの利点である。

2.2 群ロボットの formation 制御に関する研究

近年、群ロボットの領域においてロボット配置を行うための分散型の制御が提案されている。こうした研究においては、全てのロボットが大域的な状態情報を利用することができるかどうか、つまりセンサや通信がその範囲に制約を持っているかどうかという観点から、大域的なクラスか局所的なクラスかで幅広く分類をすることができる。大域的なクラスの問題 [31]-[35] はロボットに高速で正確な、そして効率的な配置を与えることができる反面、技術的に実現不可能な場合が多く、またロボット台数の増加に対する拡張性に乏しいという問題がある。一方、局所的なクラスの問題とは、蟻の共同体や魚の一群のような社会的な生命体、又は結晶化のような物理現象から観察される個々のロボット間の相互作用に基づくものである。局所的なクラスは更に生物学的な創発 [36]-[40]、行動に基くもの [41,42]、仮想物理に基くもの [43]-[56] に分類される。行動や仮想物理に基くものの多くは電荷 [43]、重力 [44],[45]、バネ [46]-[48]、ポテンシャル場 [49],[50]、ファン・デル・ワールス力 [51]、分子の平衡 [52]、その他の仮想力のモデル [53]-[55] のような物理現象を用いている。これらの研究のほとんどは、ある範囲の中にある全てのロボットに引付力や反発力を与える相互作用の力の均衡を用いている。その理由としては、力に基づく相互作用則は単純であるが効果的であり、個々のロボットの振る舞いに対する直観的な理解を与えるからである。これらの中で、典型的な 3 つの局所的な相互作用についての特徴を表 2.1 にまとめる。

こうした群ロボットにおける配置問題の 1 つに、領域被覆問題と呼ばれる問題が存在する。この領域被覆問題とは、一般に地理的な制約のある場所に群ロボットをいかにして効率的に配置するかという問題である。これは美術館のように壁などの制約条件のある中で、どのようにして最小人数の警備員で美術館全体を監視できるよう人員を配置するのかというガードマン配置問題に代表される。このガードマン配置問題はガードマンが指定した場所から移動しないという静的な問題として考えられるが、これを応用していかに少ない台数のロボットで効率良く建物内をくまなく清掃できるかといった清掃ロボット問題なども存在する。こう

した問題に対しては数多くの研究がなされており、様々なアルゴリズムが開発されている (例えば [56]-[59])

また、群ロボットの配置問題における別の問題として、formation 制御が挙げられる。群ロボットにおける formation 制御とは、ランダムな初期位置に置かれた群ロボットを予め既定された形状に配置する、または既定の形状を保ったまま移動を行うための制御則のことである。

ロボットが目標とする形状としては、正三角形のもの [11]-[13]、円形のもの [14][15] を始めとして、様々な形状に対するアルゴリズムが提案されている。また、最近ではロボットの存在する環境に障害物のような地理的な制約が存在する場合に適用できるアルゴリズム、更には単にロボットをランダムな初期位置から目標形状に収束させるだけでなく、目標形状を保ちながら、時には障害物を避けながら移動をするためのアルゴリズムも開発されている [16]-[20]。

前述した局所的な相互作用は、創発的な手法で格子型のロボット配置を与える。格子型のロボット配置は高いレベルでの冗長な接続を与え、そのことによって接続形態の観点から高い信頼性や柔軟性を保証する。こうした接続形態の特徴は、群ロボットが環境の監視などのためにデータを収集し伝達する際に非常に重要である。ロボットが形成する格子形状としては三角形や四角形、六角形など様々な形状が考えられるが、ここでは三角形、四角形、六角形の3種類の格子を例にとりその被覆範囲、被覆密度 [63]、接続性 [64] の観点から比較したものを表 2.2 に示す。なお、各多角形における1辺の長さは d_u とする。

表より明らかな通り、六角形格子が被覆領域の面では三角形格子や四角形格子と比べてよりよい結果を示している。一方で、三角形格子は被覆密度と接続性の点で他の2つよりもよりよい結果を示している。

2.3 既存の研究における問題点

群ロボットにおいては、各ロボットが協調して1つの動作を達成するために、ロボット同士が互いに情報を交換し合う必要がある。ロボットが互いの情報を認識するための手法としては無線通信を用いる方法、赤外線通信を用いる方法、センサを用いる方法など様々な種類が存在するが、現場での使用を考えた場合にこれら

の方法はいずれも完全な情報交換は保証されておらず、例えば障害物の存在によりセンサが感知できない、あるいは通信が遮断されるといった問題点が存在する。

こうした問題を克服するために、ロボット間の通信に損失が発生した場合に、失われた情報を何らかの形で推定する手法が必要となってくる。あるシステムにおける状態量を推定する手法としては、制御理論の分野におけるカルマンフィルタに代表されるフィルタがある。こうしたフィルタは堅固な理論に裏打ちされているため高い精度で状態を推定することが可能であるが、一方で多くの計算量を必要とする場合が多く、一般に限られた計算能力しか持たない群ロボットに直接適用することは難しいと考えられる。また、一方で単純な外挿手法としては1ステップ前の値をそのまま現在の値として用いる0次ホールド、過去の状態値に1次関数を fitting させることで現在値を求める線形外挿、2次関数を用いて求める2次外挿、 \dots 、 n 次関数を用いて求める n 次外挿が存在するが、こうした外挿手法は多くの計算量を必要としない反面、その推定精度については保証されていない。よって、本論文においては群ロボットの formation 制御においてロボット間の通信に損失が発生した場合に、既存の外挿法のように計算量をあまり必要とせず、かつ理論的なフィルタのように高い精度で現在の状態を推定することのできる新たな手法を提案することを研究の目的とする。

表 2.1: 典型的な物理学に基づく局所的な相互作用

	バネ力 [46]-[48]	人工的な重力 [44][45]	ポテンシャル場 [49][50]
相互作用力	$\ddot{x}_i = k_s(d_{ji} - l_o)u_{ji}^{\vec{}}$ <p> $d_{ji} > l_o$: 引き付け $d_{ji} < l_o$: 反発 \ddot{x}_i: ロボットの加速度 k_s: 剛性 d_{ji}: ロボット間の距離 l_o: バネの自然長 $u_{ji}^{\vec{}}$: ロボット間の単位ベクトル </p>	$F_i = \pm \frac{g}{d_{ji}^2}$ <p> $d_{ji} > R$: 引き付け $d_{ji} < R$: 反発 F_i: ロボット間力 g: 重力定数 R: 理想とする距離 </p>	$F_i = \left(-\frac{c_1}{d_{ji}^\alpha} + \frac{c_2}{d_{ji}^\gamma}\right)u_{ji}^{\vec{}}$ <p> 引き付けと反発の結合 c_1, c_2: 制御変数 α, γ: 制御変数 $(c_1, c_2 \geq 0, \alpha > \gamma > 0)$ </p>
特徴	<ul style="list-style-type: none"> • d_{ji} と l_o 間の距離に比例 • 個々の l_o に関して内部ロボット距離を調整可能 • 内部ロボットの通信リンク • formation が隣人口ボットの数の変化に伴い変わる 	<ul style="list-style-type: none"> • d_{ji} の 2 乗に反比例 • R の値により引き付け/反発 • 計算が容易 • 規則的な格子形状 	<ul style="list-style-type: none"> • d_{ji} の冪乗に反比例 • 平衡点: $d_e = (c_1/c_2)^{1/(\alpha-\gamma)}$ • 収束速度と一様分布の間にトレードオフがある
制約	<ul style="list-style-type: none"> • 固定された隣人口ロボットが必要 • 摂動の可能性 	<ul style="list-style-type: none"> • 収束速度が初期分布に依存する • デッドロックの可能性 	<ul style="list-style-type: none"> • 計算コストがかかる • パラメータの調整が必要

表 2.2: 典型的な 3 種類の格子形状の比較

	三角形格子	四角形格子	六角形格子
被覆領域	$\frac{\sqrt{3}}{2}nd_u^2$	nd_u^2	$\frac{3\sqrt{3}}{4}nd_u^2$
被覆密度	$\frac{1.2}{d_u^2}$	$\frac{1}{d_u^2}$	$\frac{0.77}{d_u^2}$
接続性	6	4	3

第 3 章

問題設定

本章においては、本論文において取り扱う群ロボットの formation 制御における正三角形格子への収束アルゴリズム [11] について述べると共に、本論文において取り組む問題について定義する。

3.1 群ロボットにおける正三角形格子への収束アルゴリズム

3.1.1 表記法及び定義

本項においては、本論文において取り扱う群ロボットの正三角形格子への収束アルゴリズムについて述べる上で必要な表記法及び定義について記述する。まず、2次元平面上に n 台の自律型のロボット r_1, r_2, \dots, r_n が存在することを想定する。各ロボットは大きさを持たない1個の点として表されるものとし、2次元平面内を自由に移動できるものとする。ロボットは最初平面内の任意の位置にランダムに配置される。初期状態において、各ロボットは他のロボットの位置情報を持っていないものとする。図 3.1 のように各ロボットはそれぞれ進行方向を表す \vec{y}_i 及びそれを時計回りに 90° 回転させた \vec{x}_i を軸とする固有の座標系を持つものとする。各ロボット r_i の位置を p_i で表すものとし、ロボット r_i が他のロボット r_j の位置と結ぶ最短の線分を $\overline{p_i p_j}$ 、その長さを $dist(p_i, p_j)$ で表すものとする。また、2つのベクトル \vec{s}, \vec{t} のなす角を $ang(\vec{s}, \vec{t})$ で表すものとする。

ロボットが目標とする正三角形格子の1辺の長さを d_u とする。各ロボット r_i は周囲のロボットと電波や赤外線などの無線通信を用いて互いの位置情報を交換することにより、自分とのユークリッド距離が最も近いロボット r_{n1} 、及び r_i 、 r_{n1} 、 r_{n2} で形成される三角形の外周の長さが最小となるようなロボット r_{n2} を選択する。 r_{n1} 及び r_{n2} をロボット r_i の隣人と呼び、その位置の集合 $\{p_{n1}, p_{n2}\}$ を N_i で表すものとする。

定義1 (三角形配置)

与えられた p_i 及び N_i に対し、三角形配置を以下で示される3つのロボット位置の集合 $\{p_i, p_{n1}, p_{n2}\}$ で定義する。

$$\mathbb{T}_i = \{p_i, p_{n1}, p_{n2}\} \quad (3.1)$$

ここで、 r_i の内角 α_i を $\angle p_{n1}p_i p_{n2}$ で定義する。

ロボットが \mathbb{T}_i を形成する時、次を定義する。

定義2 (正三角形配置)

\mathbb{T}_i において、各点間の距離が全て d_u に等しい時、その配置を正三角形配置 \mathbb{E}_i と定義する (\mathbb{E}_i は \mathbb{T}_i の内角が全て等しい場合とも定義できる)

上記の定義に基づき、以下の問題が定義される。

問題3 (局所的な相互作用)

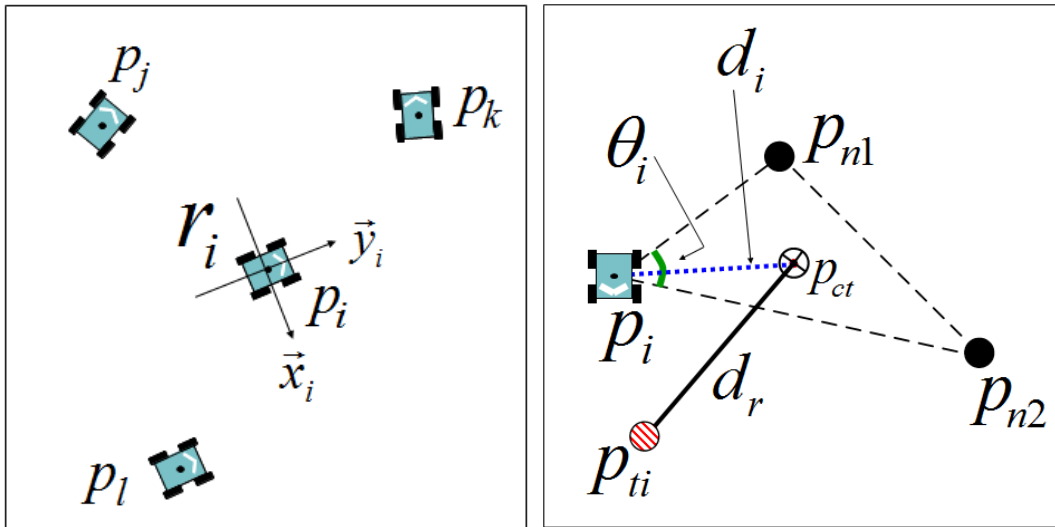
与えられた \mathbb{T}_i に対し、各ロボットが \mathbb{E}_i を形成するために各時間ステップにおいて、 r_i が N_i の中で他のロボットとの距離を d_u に維持するための作用を局所的な相互作用と定義する。

上記で定義した局所的な相互作用に基づき、3台のロボットは \mathbb{E}_i に収束する。これを元にして、以下の問題が定義される。

問題4 (自己配置問題)

2次元平面内で任意にその位置を区別されたロボット群 r_1, \dots, r_n に対し、全てのロボットが局所的な相互作用を有限回実行することで穴のような不規則性なしに \mathbb{E}_i に配置する問題を、自己配置問題と呼ぶ。

各ロボットは隣人口ロボットとの相互作用を繰り返しながら \mathbb{E}_i を形成するように行動する。各ロボットは隣人口ロボットの位置に基づき (観察) 自らが動くべき場所を計算し (計算) 計算された位置に向かって移動する (動作)。



(a) 局所座標系: \vec{x}_i 及び \vec{y}_i

(b) 動作制御: d_i 及び θ_i

図 3.1: ロボット r_i に関する定義及び表記法

3.1.2 収束アルゴリズム

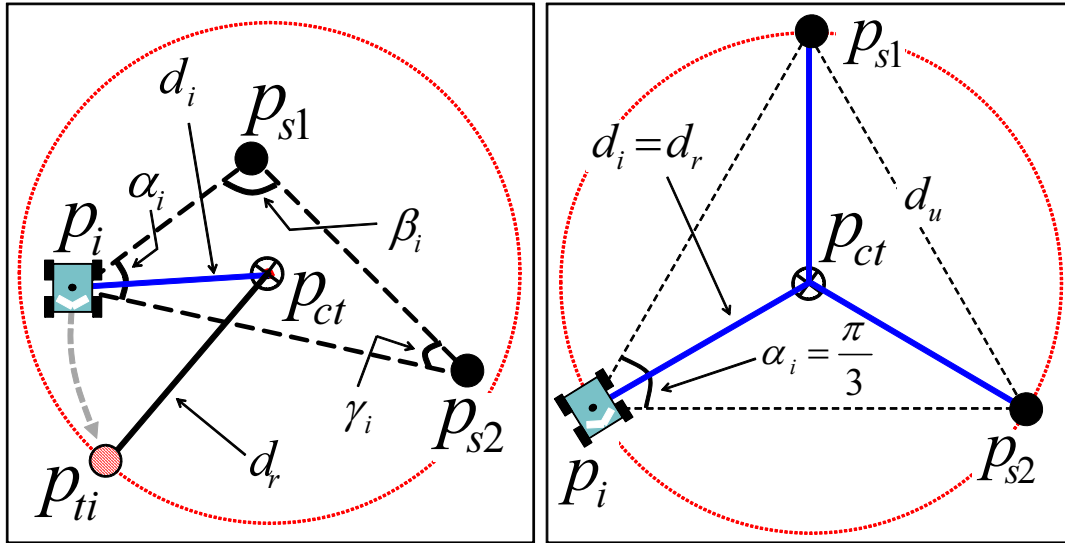
ロボットが正三角形格子を形成するために実行するアルゴリズムを以下の Algorithm1 に示す。各ロボットは周囲のロボットと通信を行い、互いの位置情報を交換することにより 2 台の隣人口ボットを決定し、この 2 台のロボットと自らの位置情報を元に以下の Algorithm1 を実行する。このアルゴリズムは各時間ステップにおける独立変数 p_i 及び N_i を持つ関数 $\varphi_{interaction}$ から成る。3 台のロボットは各頂点が p_i 、 p_{n1} 、 p_{n2} である \mathbb{T}_i に配置されている。 r_i は三角形 $\triangle p_i p_{n1} p_{n2}$ の重心 $p_{ct} = (x_g, y_g)$ を見つけ、その水平軸と線分 $\overline{p_{n1} p_{n2}}$ との成す角 ϕ を求める。 p_{ct} 及び ϕ を用いて、以下の Algorithm1 によりロボット r_i は自らの移動すべき目標位置 p_{ti} を計算する。全てのロボットがその隣人口ボットの位置から以下の Algorithm1 により目標位置を計算し、その場所に向かって移動する。

Algorithm 1(正三角形格子の形成)

- 1) 目標とする正三角形の一辺の長さを d_u と定義する
- 2) 以下の関数 $\varphi_{interaction}(p_{s1}, p_{s2}, p_i)$ を実行する
 - a) $(p_{ct,x}, p_{ct,y})$ を $\triangle p_{s1}, p_{s2}, p_i$ の重心とする
 - b) ϕ を $\overline{p_{s1} p_{s2}}$ と r_i の水平軸の間の成す角とする
 - c) $p_{ti,x} = p_{ct,x} + d_u \cos(\phi + \pi/2)/\sqrt{3}$ とする

d) $p_{ti,y} = p_{ct,y} + d_u \sin(\phi + \pi/2)/\sqrt{3}$ とする

e) ロボット r_i は目標位置 $p_{ti} = (p_{ti,x}, p_{ti,y})$ に移動する



(a) 表記法

(b) 正三角形配置

図 3.2: 局所的な相互作用のための 2 つの制御パラメータ

ここでは、3 台のロボットがどのようにして \mathbb{E}_i に収束するかについて説明する。まず、3 台のロボット A 、 B 、 C の位置を表す三角形の 3 頂点 p_a 、 p_b 、 p_c を考える。 α 、 β 、 γ をそれぞれ三角形の内角とする。 $\Delta p_a p_b p_c$ の頂点に位置する各ロボットは、Algorithm1 に従って計算された新しい目標位置 p_{ta} 、 p_{tb} 、 p_{tc} に移動する。 $\Delta p_{ta} p_{tb} p_{tc}$ の内角をそれぞれ α' 、 β' 、 γ' で表す。 $\Delta p_a p_b p_c$ の重心を p_{ct} で表すものとする。すると、幾何学的な関係により、それぞれの角度の間には、 $\alpha' = (\beta + \gamma)/2$ 、 $\beta' = (\alpha + \gamma)/2$ 、 $\gamma' = (\alpha + \beta)/2$ という関係が得られる。これらの関係式は以下のように書き直すことができる。

$$\begin{aligned}\alpha(t+1) &= (\beta(t) + \gamma(t))/2 \\ \beta(t+1) &= (\alpha(t) + \gamma(t))/2 \\ \gamma(t+1) &= (\alpha(t) + \beta(t))/2\end{aligned}\tag{3.2}$$

これらの式より、次の方程式が成立する。

$$\begin{aligned}
\alpha(t+2) &= \frac{\beta(t+1) + \gamma(t+1)}{2} = \frac{\alpha(t)}{2} + \frac{\beta(t) + \gamma(t)}{4} \\
\beta(t+2) &= \frac{\alpha(t+1) + \gamma(t+1)}{2} = \frac{\beta(t)}{2} + \frac{\gamma(t) + \alpha(t)}{4} \\
\gamma(t+2) &= \frac{\alpha(t+1) + \beta(t+1)}{2} = \frac{\gamma(t)}{2} + \frac{\alpha(t) + \beta(t)}{4}
\end{aligned} \tag{3.3}$$

上式より、以下の関係が得られる。

$$\begin{aligned}
\alpha(t+2) &= \alpha(t)/2 + \alpha(t+1)/2 \\
\beta(t+2) &= \beta(t)/2 + \beta(t+1)/2 \\
\gamma(t+2) &= \gamma(t)/2 + \gamma(t+1)/2
\end{aligned} \tag{3.4}$$

これより、以下の式が得られる。

$$\begin{aligned}
\alpha(t+2) &= \frac{\alpha(t)}{2^2} + \frac{3}{2^2} \cdot \frac{\alpha(t) + \beta(t) + \gamma(t)}{3} \\
\beta(t+2) &= \frac{\beta(t)}{2^2} + \frac{3}{2^2} \cdot \frac{\alpha(t) + \beta(t) + \gamma(t)}{3} \\
\gamma(t+2) &= \frac{\gamma(t)}{2^2} + \frac{3}{2^2} \cdot \frac{\alpha(t) + \beta(t) + \gamma(t)}{3}
\end{aligned} \tag{3.5}$$

上式を一般化すると次のようになる。

$$\begin{aligned}
\alpha(t+n) &= \frac{\alpha(t)}{2^n} + \frac{\sum_{k=1}^n 3 \cdot 2^{k-2}}{2^n} \cdot \frac{\alpha(t) + \beta(t) + \gamma(t)}{3} \\
\beta(t+n) &= \frac{\beta(t)}{2^n} + \frac{\sum_{k=1}^n 3 \cdot 2^{k-2}}{2^n} \cdot \frac{\alpha(t) + \beta(t) + \gamma(t)}{3} \\
\gamma(t+n) &= \frac{\gamma(t)}{2^n} + \frac{\sum_{k=1}^n 3 \cdot 2^{k-2}}{2^n} \cdot \frac{\alpha(t) + \beta(t) + \gamma(t)}{3}
\end{aligned} \tag{3.6}$$

上式において $n \rightarrow \infty$ の極限を取ると、

$$\begin{aligned}
\lim_{n \rightarrow \infty} \alpha(t+n) &= \frac{\alpha(t) + \beta(t) + \gamma(t)}{3} = \pi/3 \\
\lim_{n \rightarrow \infty} \beta(t+n) &= \frac{\alpha(t) + \beta(t) + \gamma(t)}{3} = \pi/3 \\
\lim_{n \rightarrow \infty} \gamma(t+n) &= \frac{\alpha(t) + \beta(t) + \gamma(t)}{3} = \pi/3
\end{aligned} \tag{3.7}$$

これより、3台のロボットが正三角形の形状に収束することが分かる。

3.1.3 アルゴリズムの収束性

前節に示したアルゴリズムに則ってロボットが動作を行う場合、ロボット r_i の動きは p_{ct} からの距離 d_i 及び内角 θ_i を用いて以下の運動方程式で表される [11]。

$$\dot{d}_i(t_c) = -a(d_i(t_c) - d_r) \tag{3.8}$$

$$\dot{\theta}_i(t_c) = k\left(\frac{\pi}{3} - \theta_i(t_c)\right) \tag{3.9}$$

ここで a 及び k は正の定数であり、また $d_r = d_u/\sqrt{3}$ である。運動方程式 (3.8) 及び (3.9) の解は、それぞれ以下の式で表される。

$$d_i(t_c) = (d_i(0) - d_r)e^{-at_c} + d_r \tag{3.10}$$

$$\theta_i(t_c) = \left(\theta_i(0) - \frac{\pi}{3}\right)e^{-kt_c} + \frac{\pi}{3} \tag{3.11}$$

(3.10) 及び (3.11) より、 $t_c \rightarrow \infty$ の時 $d_i \rightarrow d_r$ 、 $\theta_i \rightarrow \pi/3$ となることから、各ロボットは1辺の長さが d_u の正三角形の格子を形成することが分かる。

図 3.3 に台数 $n = 10$ のロボットに前述の Algorithm1 を適用した場合に各ロボットが目標形状である正三角形格子に向けて収束していく様子を示す。

3.2 本論文で取り組む問題の定義

本節において、前節で述べたロボットモデルに基づいた群ロボットにおける位置推定問題を、以下のように定義する。

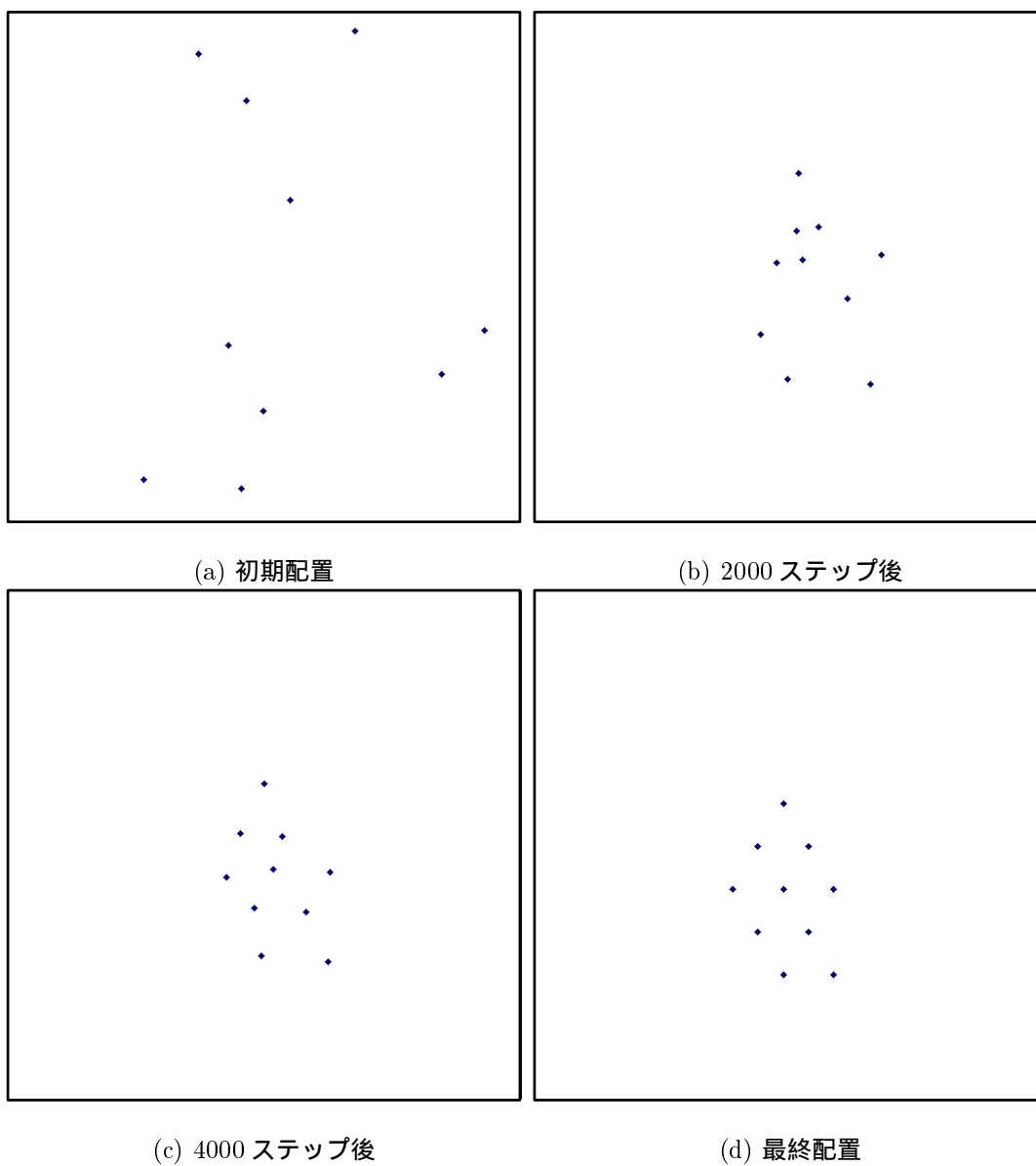


図 3.3: 正三角形配置アルゴリズムの実行結果

「任意の位置に配置されたロボットが (3.10) 及び (3.11) の制御手法に従って動作する場合、ロボット間の通信損失の存在に関わらず、どのようにしてロボットを有限の時間ステップで目標とする形状に収束させることができるか？」

通信損失が発生した場合、ロボット r_i は他のロボット r_j の位置情報を得られないため、次のステップで行う動作を決定することができない。このため、失われた情報を何らかの方法により推定する必要がある。ここで、本論文で想定しているロボットは限られた範囲内では他のロボットと通信ができない、比較的単純な

計算能力しか持たない、過去の他のロボット位置に関する記憶も限定的であるというように限られた能力しか持たないため、他のロボットの位置推定のために多くの計算量を必要とする方法を用いることはできない。このため、ロボットが適切に動作するために必要な精度を保ちつつ、計算量を抑えた外挿手法として、本論文では指数関数を用いた外挿式を提案する。

第 4 章

群ロボットの目標形状への収束度の評価指標

本章においては、群ロボットの formation 制御におけるロボットの目標形状への収束度を測るための評価指標を導入する。群ロボットの formation 制御においては、最初ランダムな初期位置に配置されたロボットが互いの位置情報を交換しながら、それぞれ既定のアルゴリズムを実行することにより、目標形状に向かって収束していく。その際、各ロボットが移動を行っているある時点において、その時点における各ロボットの位置が目標形状に対してどの程度近づいているのか（あるいは離れているのか）を定量的に測ることは有益である。これまで、群ロボットの formation 制御については数多くの研究がなされ、その収束性についても議論が行われてきたが、ロボット位置の目標形状への収束度という観点からの研究はされてこなかったのが現状である。このため、本章においては formation 制御における目標形状として、本論文において用いる正三角形格子を例に取り、目標形状である正三角形格子に対してある時点におけるロボット位置がどの程度近づいているか（離れているか）を定量的に示す評価指標を提案する。具体的には、ロボットの配置が目標とする正三角形格子に完全に一致しているときにはその値が 0 となり、ロボット配置が目標形状である正三角形から離れていくにつれて徐々にその値が大きくなっていくような評価指標を導入することを目標とする。そして、この評価指標が我々人間の持つ直感的な「目標形状に近い」または「ばらついていない」という感覚に一致するかについて検証を行う。

本章では、以下の仮定の元に議論を行う。初期状態として台数 n のロボットを 2 次元平面内の任意の位置に配置するものとする。各ロボットは 3.1.2 節で示した Algorithm1 に従い、1 辺の長さが D の正三角形格子を形成するものとする。各ロボットは大きさを持たず、その位置は点で表されるものとする。また、各ロボットは区別されるものとする。

4.1 Delaunay 三角形を用いた群ロボットの目標形状への収束度の評価指標

4.1.1 Delaunay 三角形

Delaunay 三角形とは、計算幾何学における Voronoi 図の双対であると考えることができる。ここで、Voronoi 図とは以下のように定義することができる。

”平面上に何個かの点が配置されているものとする。このとき、その平面内における任意の位置が、配置されたどの点に最も近いかによって分割された領域のことを Voronoi 領域と呼び、Voronoi 領域によって生成される図を Voronoi 図と呼ぶ”。Voronoi 図の概要を図 4.1 に示す。この時、配置された点のことを母点と呼び、Voronoi 領域の境界のことを Voronoi 境界、Voronoi 境界の交点を Voronoi 点と呼ぶ。

ここで、隣接する Voronoi 領域の母点どうしを線分で結んでできる図形のことを Delaunay 図と呼ぶ。一般に、Delaunay 図によって生成される図形は三角形となるため、この Delaunay 図のことを Delaunay 三角形分割と呼ぶ。図 4.2 にロボット台数 (母点) $n = 9$ に対する Delaunay 三角形を示す。

Delaunay 三角形の特徴としては、各三角形の角の最小値が最大かつ最適な分割であるという特徴を持っている。次項以降においては、この Delaunay 三角形を用いて正三角形格子を目標形状とする群ロボットの収束度を表す指標を定義し、その有効性について議論を行う。

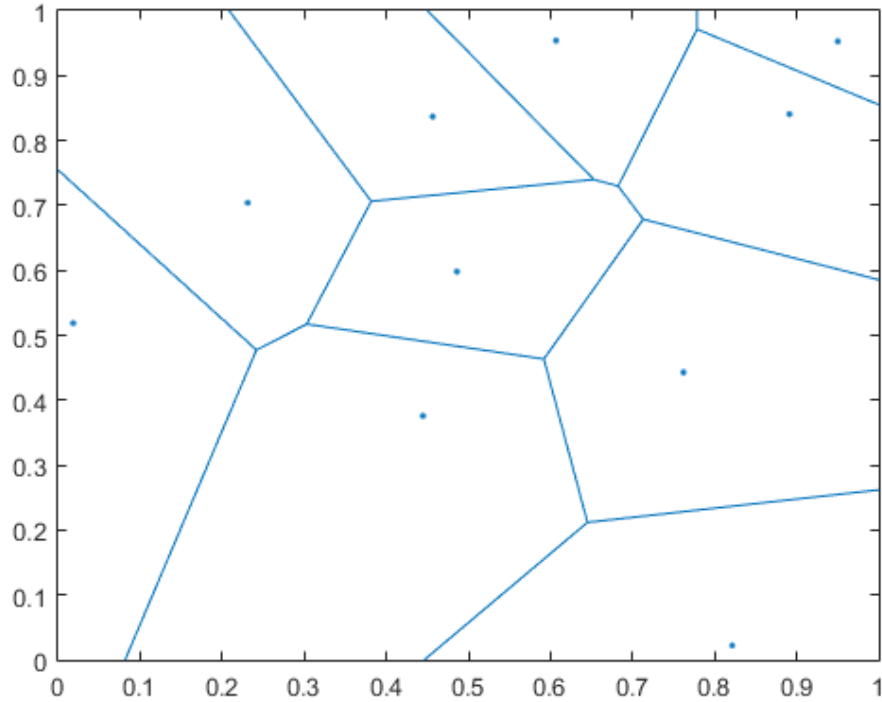


図 4.1: Voronoi 図

4.1.2 群ロボットの目標形状への収束度の評価指標

本項においてはまず、ある時点でのロボット配置において、各ロボットの位置を表す点を頂点とする Delaunay 三角形 [65] を構成するものとする。計算された Delaunay 三角形の各辺の長さと同様とする正三角形格子の 1 辺の長さ D との関係から、次に示す 3 つの評価指標 (EI) を導入する。

EI1. Delaunay 三角形の各辺の長さと同様とする正三角形の 1 辺の長さの差の総和

EI2. Delaunay 三角形の各辺の長さの 2 乗と同様とする正三角形の 1 辺の長さの 2 乗の差の総和

EI3. Delaunay 三角形の各辺の長さと同様とする正三角形の 1 辺の長さの比の絶対値の対数の和

具体的な式は以下の通りである。

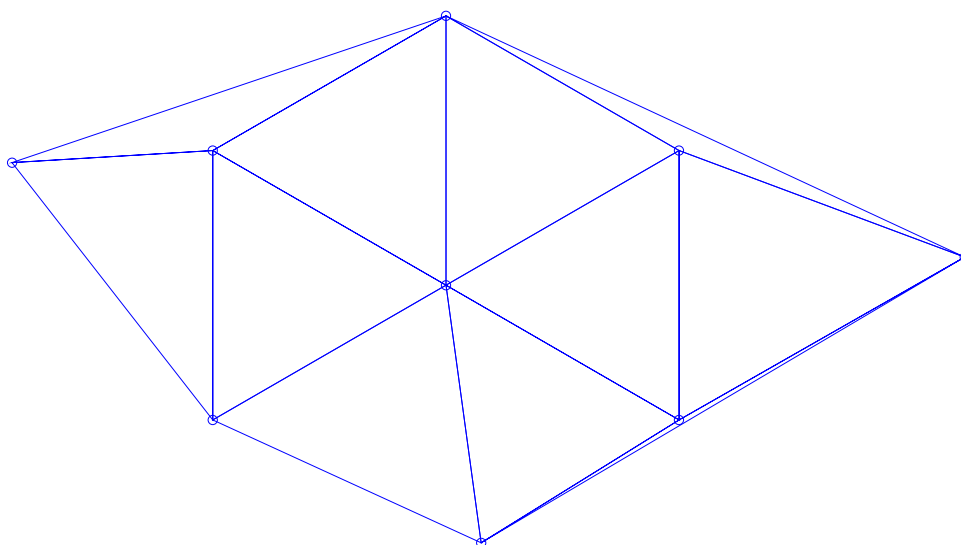


図 4.2: ロボット台数 $n = 9$ に対する Delaunay 三角形

$$EI1 = \sum_{i,j}^N \left| \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - D \right| \quad (4.1)$$

$$EI2 = \sum_{i,j}^N \left| (x_i - x_j)^2 + (y_i - y_j)^2 - D^2 \right| \quad (4.2)$$

$$EI3 = \sum_{i,j}^N \left| \log \left(\frac{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{D} \right) \right| \quad (4.3)$$

ここで、 (x_i, y_i) はロボット i の位置、 D は目標とする正三角形格子の 1 辺の長さを表している。

上記の 3 つの評価指標は、各ロボット位置を頂点とする Delaunay 三角形が既定の正三角形に近い時には 0 に近い値を示し、その形状が正三角形から離れていくにつれて大きな値を取るよう設計されている。

4.2 提案した評価指標の定量的評価

本節においては、4.1.2項で定義した3つの評価指標のうちどれが我々人間の持つ「直感的な」ばらつきに最も近い表現をしているかを、シミュレーションにより検証する。

シミュレーション環境としては2次元平面内に台数 $n = 10, 50$ のロボットが1辺の長さ10の正三角形の格子を形成する場合を想定し、完全な正三角形格子から徐々に大きなばらつきを与えていくにつれて、評価指標の値がどのように変化していくかを計算により求めた。

まずはロボット台数 $n = 10$ の場合について述べる。図4.3にロボットが完全な正三角形格子を形成している場合の配置を示す。この時の各評価指標の値は、 $EI1=7.3156$ 、 $EI2=199.89$ 、 $EI3=0.5497$ と計算された。

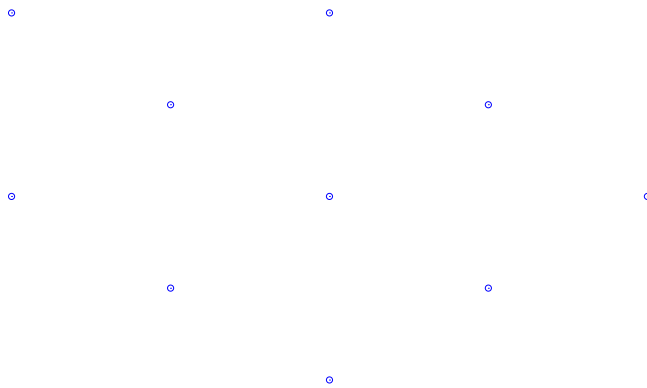


図 4.3: ロボット台数 $n = 10$ の時の完全な正三角形配置

次に、図4.4にロボットの各点の座標に対して、正三角形格子の1辺の長さ D に対して $\pm 50\%$ 以内のランダムなバラつきを与えた場合の配置を示す。この時の各評価指標の値は、 $EI1=8.5406$ 、 $EI2=273.17$ 、 $EI3=4.8558$ と計算された。

続いて、図4.5にロボットの各点の座標に対して図4.4の場合よりも更に大きい、正三角形格子の1辺の長さ D に対して $\pm 100\%$ 以内のバラつきを与えた場合の配置を示す。この時の各評価指標の値は、 $EI1=111.26$ 、 $EI2=3179.7$ 、 $EI3=12.22$ と計算された。

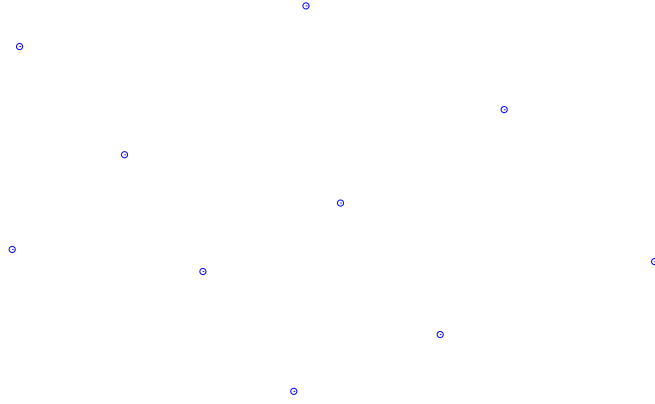


図 4.4: 図 4.3 の配置にランダムなバラつきを与えた場合

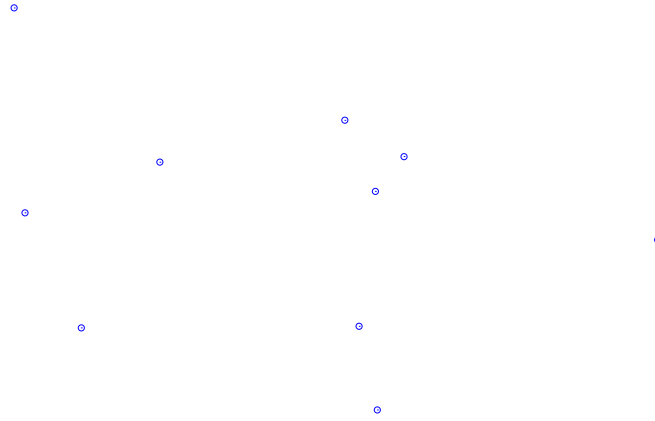


図 4.5: 図 4.4 よりも更に大きなバラつきを与えた場合

以上の結果をまとめた表を表 4.1 に示す。

表 4.1: 各配置における EI の値 (ロボット台数 $n = 10$)

	EI1	EI2	EI3
図 4.3	7.3156	199.89	0.5497
図 4.4	8.5406	273.17	4.8558
図 4.5	111.26	3179.7	12.22

次に、ロボット台数 $n = 50$ の場合についても示す。まずは図 4.6 にロボットが完全な正三角形格子を形成している場合の配置を示す。この時の各評価指標の値は、 $EI1=142.78$ 、 $EI2=6199.1$ 、 $EI3=7.6843$ と計算された。

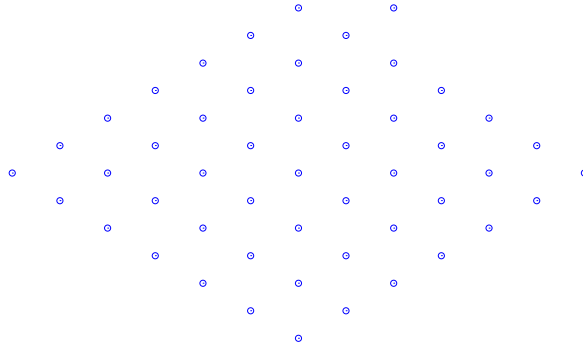


図 4.6: ロボット台数 $n = 50$ の時の完全な正三角形配置

次に、図 4.7 にロボットの各点の座標に対して正三角形格子の 1 辺の長さ D に対して $\pm 50\%$ 以内のランダムなバラつきを与えた場合の配置を示す。この時の各評価指標の値は、 $EI1=342.36$ 、 $EI2=13611$ 、 $EI3=57.595$ と計算された。

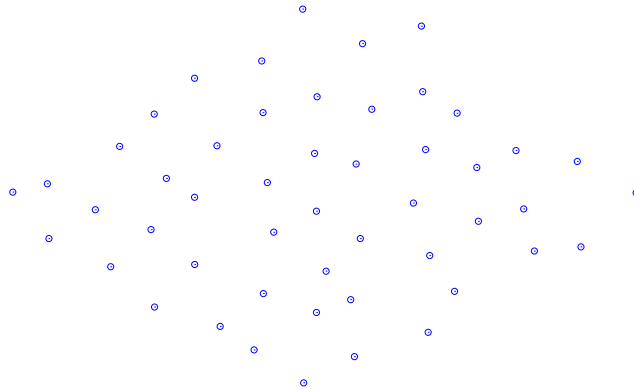


図 4.7: 図 4.6 の配置にランダムなバラつきを与えた場合

続いて、図 4.8 にロボットの各点の座標に対して図 4.7 の場合よりも更に大きい、正三角形格子の 1 辺の長さ D に対して $\pm 100\%$ 以内のバラつきを与えた場合の配

置を示す。この時の各評価指標の値は、 $EI1=449.97$ 、 $EI2=18901$ 、 $EI3=87.498$ と計算された。

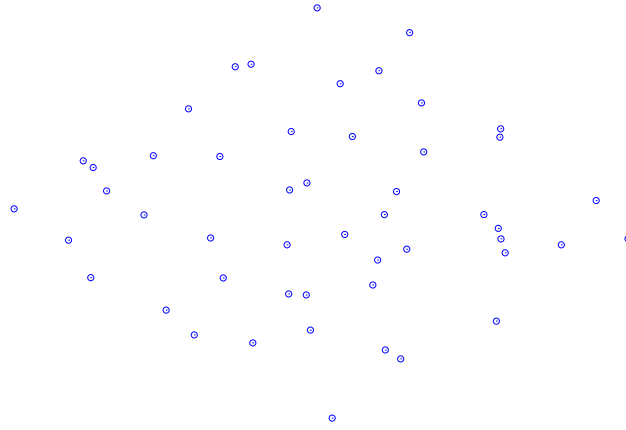


図 4.8: 図 4.7 よりも更に大きなバラつきを与えた場合

以上の結果をまとめた表を表 4.2 に示す。

表 4.2: 各配置における EI の値 (ロボット台数 $n = 50$)

	EI1	EI2	EI3
図 4.6	142.78	6199.1	7.6843
図 4.7	342.36	13611	57.595
図 4.8	499.97	18901	87.498

4.3 提案した評価指標の問題点及びその改良

本節においては、前節で行った各評価指標の計算結果について議論する。

前節で行ったシミュレーション結果より、ロボット台数 $n = 10, 50$ いずれの場合も、ロボット配置におけるバラつきが大きくなるにつれて各評価指標の値は大きくなっていった。これは、各ロボットの位置を頂点とする Delaunay 三角形が目標とする正三角形から離れていくにつれて、各辺の長さとの正三角形の 1 辺の長さとの間の差が大きくなっていくことを考えれば当然のことと言える。また、その

定義より、ばらつきが大きくなっていくにつれてその値は EI2 が最も顕著な変化を示した。

なお、 $n = 10, 50$ いずれの場合もロボットが完全な正三角形格子を形成しているにも関わらず、各評価指標の値は 0 にはならなかった。その理由を以下に述べる。

図 4.9 は、 $n = 10$ 台のロボットが完全な正三角形格子を形成している場合に計算された Delaunay 三角形を表している。

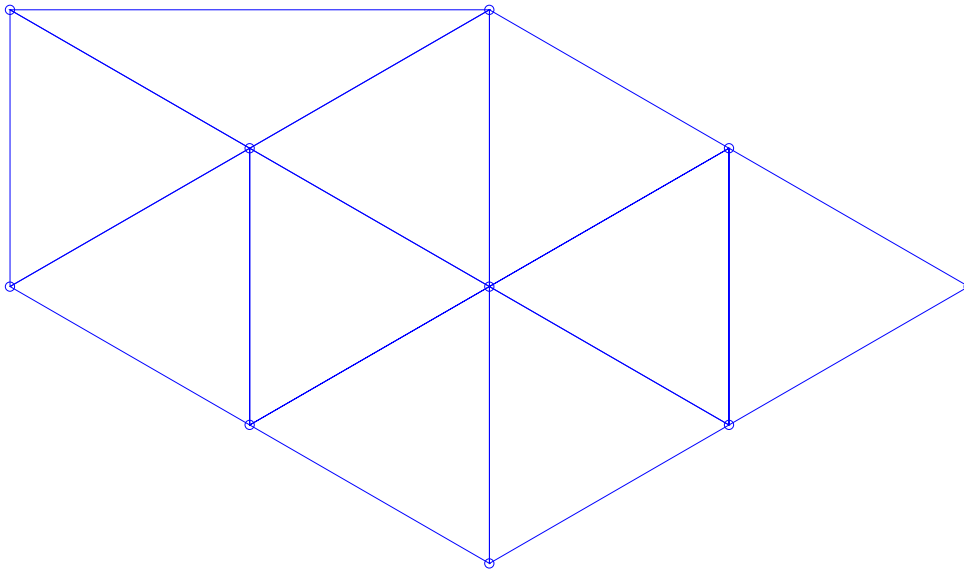


図 4.9: 正三角形でない Delaunay 三角形を持つ正三角形配置

図 4.9 から明らかなように、ロボット台数 n の値によっては、完全な正三角形格子を形成している場合でも正三角形でない Delaunay 三角形が作られてしまい、その結果として評価指標の値が 0 にならないものと考えられる。

また、今回導入した 3 つの評価指標には、以下の問題点も考えられる。

図 4.10 及び図 4.11 はロボット台数 $n = 200$ に対し、ロボットが非常に密に分布している場合と完全な正三角形格子を形成している場合を表している。この場合の各評価指標の値は、表 4.3 のように計算された。

本来ならば評価指標の値は、正三角形格子を構成している図 4.11 の方が図 4.10 よりも小さな値を示すはずであるが、実際には EI1 と EI2 で図 4.11 の方が大きな

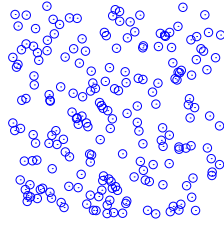


図 4.10: 「密な」配置をしたロボット (台数 $n = 200$)

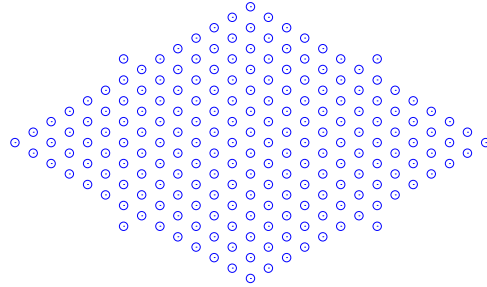


図 4.11: 完全な正三角形配置をしたロボット (台数 $n = 200$)

表 4.3: 各配置における EI の値 (ロボット台数 $n = 200$)

	EI1	EI2	EI3
図 4.10	1488.8	19512	655.12
図 4.11	2545.7	145584	111.82

値を示している。これは、評価指標の定義式より、ロボットの分布が密な場合には評価指標の値が小さく算出される傾向にあるためと考えられるが、これは我々人間が持つ直感的な「ばらつき」とは異なる結果であり、評価指標を導入した目的を考えると望ましいものではなく、何らかの改良を加える必要があると考えられる。

4.4 提案した評価指標の問題点及びその改良

4.1.2 項において導入された 3 種類の評価指標においては、計算された評価指標の値が我々人間の持つロボット位置についての直感的な「ばらつき」と必ずしも一致しないという問題点が明らかとなった。よって、本節においてはこの問題点を克服するため、新たな評価指標を導入する。

ここでもロボットが正三角形格子を形成する場合を想定する。各ロボット位置を頂点とする Delaunay 三角形を構成し、その各辺の長さ及び面積から文献 [66] に示された以下の評価指標 (EI4) を導入する。

$$EI4 = \sum_{k=1}^M \left(\frac{(a_k^2 - b_k^2)^2 + (b_k^2 - c_k^2)^2 + (c_k^2 - a_k^2)^2}{4S_k^2} \right) \quad (4.4)$$

ここで、 a_k 、 b_k 、 c_k は各ロボットの位置を頂点とする Delaunay 三角形を構成したときの三角形の各辺の長さ、 S_k はその面積を表すものとする。

上式の () 内の評価関数も 4.2 節の 3 つの評価指標と同様に、正三角形の場合には 0 の値を、三角形の形状が正三角形から離れていくにつれて大きな値を取るよう設計されている。

4.5 改良した評価指標の定量的評価

本節においては、新たに導入した (4.4) 式で表される評価指標が、4.1.2 項において導入した 3 つの評価指標の特性を維持しつつも、その問題点を解消していることを示す。

シミュレーション環境としては前章と同様に $n = 10, 50$ 台のロボットが 1 辺の長さ 10 の正三角形格子を形成する場合を想定し、ロボット配置に対して与えるばらつきを徐々に増やしていくにつれて EI の値がどのように変化していくかを計算により求めた。

まずはロボット台数 $n = 10$ の場合について示す。ロボットを 4.2 節の図 4.3 ~ 図 4.5 のような 3 種類の形状に配置し、その時の EI4 の値を計算した。計算結果は表 4.4 に示す通りである。

表 4.4: 図 4.3 ~ 図 4.5 に対する EI4 の値 (ロボット台数 $n = 10$)

	図 4.3	図 4.4	図 4.5
EI4	10.666	50.762	66.773

続いて、ロボット台数 $n = 50$ の場合についても示す。この場合もロボットを図 4.6 ~ 図 4.8 の 3 種類の形状に配置し、その時の EI4 の値を計算した。計算結果は

表 4.5 に示す通りである。

表 4.5: 図 4.6 ~ 図 4.8 に対する EI4 の値 (ロボット台数 $n = 50$)

	図 4.6	図 4.7	図 4.8
EI4	1194.66	12939	14256

以上の計算結果より、ロボット台数 $n = 10, 50$ いずれの場合も、EI4 の値は正三角形からのばらつきが大きくなるにつれて大きな値を取ることが分かる。

次に、4.3 節で明らかとなった問題点を EI4 が克服しているかどうかについて検証を行う。

4.3 節と同様に、図 4.10 及び図 4.11 のように $n = 200$ 台のロボットを 2 次元平面内に配置した時の EI4 の値の計算結果を示す。

この時評価指標 EI4 の値はそれぞれ図 4.10 で EI4=114713、図 4.11 で EI4=53952 と計算された。これより、ロボットが密に配置された場合においても EI4 は正三角形の格子を形成している場合に比べて大きな値を示しており、4.3 節で起こった問題が解消されたことを示している。

第 5 章

単純な関数を用いたロボット位置の外挿

本章においては、群ロボットにおいてロボット間の通信に損失が発生した場合、失われた情報を外挿するための手法として既存の単純な外挿式を用いて外挿を行った場合の結果について検証する。過去の状態から現在の状態を外挿するための外挿式としては様々な関数を用いた結果が得られているが、今回はその中でも最も単純で複雑な計算を必要としない外挿式として知られている 0 次ホールド、線形外挿、2 次外挿の 3 種類の外挿式を用いてロボット間の通信に損失が発生した場合に失われた情報を外挿し、その推定精度についてシミュレーションによる考察を行う。

5.1 0 次ホールド・線形外挿・2 次外挿の概要

5.1.1 0 次ホールド

0 次ホールドによる外挿とは、以下の通りである。ある時刻 kt において得られるべき情報が失われた場合、時刻 kt における状態量 $N(kt)$ をその 1 ステップ前の時刻 $k(t-1)$ における状態量 $N(k(t-1))$ と同じものとして考える、つまり

$$N(kt) = N(k(t-1)) \quad (5.1)$$

として現在の状態量を外挿する方法を、0次ホールドによる外挿と呼ぶ。この方法の利点は外挿したい状態の時刻の1ステップ前の状態量のみを利用するため、極めて少ないメモリのみを必要とし、計算を全く必要としないという点である。一方で、1ステップ前の情報のみを用いて状態を外挿するため、その系のダイナミクスといった特性を全く考慮しておらず、正確に外挿を行えない可能性が高いという問題がある。

5.1.2 線形外挿

線形外挿とは、 $y = ax + b$ で表される1次関数を用いて外挿を行う手法のことである。ある時刻 kt において得られるべき情報が失われた場合、その時刻 kt における状態量 $N(kt)$ を、その1ステップ前の時刻 $k(t-1)$ における状態量 $N(k(t-1))$ 及び2ステップ前の時刻 $k(t-2)$ における状態量 $N(k(t-2))$ を1次関数に fitting させた式を用いて求めるという手法である。具体的には、2つの1次方程式

$$N(k(t-1)) = a(k(t-1)) + b \quad (5.2)$$

$$N(k(t-2)) = a(k(t-2)) + b \quad (5.3)$$

より a 及び b の値を求め、この値を用いて

$$N(kt) = a(kt) + b \quad (5.4)$$

により現在の状態量を推定する。

5.1.3 2次外挿

2次外挿とは $y = ax^2 + bx + c$ で表される2次関数を用いて外挿を行う手法のことである。ある時刻 kt において得られるべき情報が失われた場合、その時刻 kt における状態量 $N(kt)$ を、その1ステップ前の時刻 $k(t-1)$ における状態量 $N(k(t-1))$ 、2ステップ前の時刻 $k(t-2)$ における状態量 $N(k(t-2))$ 、3ステップ前の時刻 $k(t-3)$ における状態量 $N(k(t-3))$ を2次関数に fitting させた式を用いて求めるという手法である。具体的には、3つの1次方程式

$$N(k(t-1)) = a(k(t-1))^2 + b(k(t-1)) + c \quad (5.5)$$

$$N(k(t-2)) = a(k(t-2))^2 + b(k(t-2)) + c \quad (5.6)$$

$$N(k(t-3)) = a(k(t-3))^2 + b(k(t-3)) + c \quad (5.7)$$

より a 、 b 、 c の値を求め、この値を用いて

$$N(k(t-1)) = a(kt)^2 + b(kt) + c \quad (5.8)$$

により現在の状態量を推定する。

5.2 EKF(Extended Kalman Filter) 及びPF(Particle Filter) の概要

前節で述べた 0 次ホールド、線形外挿、2 次外挿はいずれも過去の状態から現在の状態を外挿する単純な手法として知られている。しかし、これらの手法はその原理が単純な一方で、その推定精度については保証されていない。一方で、制御理論の分野では過去の状態から現在の状態を理論的に推定する状態推定に関する研究が広く行われており、その中の 1 つにフィルタを用いる手法が存在する。理論的なフィルタとしては KF(Kalman Filter)[67] が有名であるが、この KF はその適用対象が線形システムに限定されているという弱点がある。この問題を克服するため、非線形システムに対してもそのシステムを部分的に線形化することで KF の適用を可能とした EKF(Extended Kalman Filter) や UKF(Unscented Kalman Filter)[68]-[70]、EnKF(Ensemble Kalman Filter)、更には統計学におけるベイズ推定を利用した粒子フィルタ PF(Particle Filter)[71][72] などが開発されている。本節においてはこれらの中で EKF 及び PF について取り上げ、その原理について説明する。

5.2.1 EKF の原理

本項においては、KF の適用対象を非線形に拡張した EKF についてその原理を説明する。

最初に、次の離散時間非線形システムを考える

$$x_{t+1} = f_t(x_t) + w_t \quad (5.9)$$

$$y_t = h_t(x_t) + v_t \quad (5.10)$$

ここで、 $x_t \in \mathbb{R}^n$ は状態ベクトル、 $y_t \in \mathbb{R}^p$ は観測ベクトル、 $w_t \in \mathbb{R}^n$ は共分散行列 Q_t かつ零平均の多変数正規分布に従うシステム雑音、 $v_t \in \mathbb{R}^p$ は共分散行列 R_t かつ零平均の多変数正規分布に従う観測雑音を表すものとする。また、 $f_t(x_t)$ 、 $h_t(x_t)$ は微分可能であるとする。

$\hat{x}_{t|t}$ をシステムの状態推定値、 $P_{k|k}$ を誤差の共分散行列とする時、EKF は以下のアルゴリズムに従い現在の状態を推定するものとする [73]。

Algorithm2(EKF)

1) 初期値を $\hat{x}_{0|-1} = \bar{x}_0$ 、 $P_{0|-1} = P_0$ とおき、 $t = 0$ とする。

2) 観測更新ステップ Input: $[\hat{x}_{t|t-1}, P_{t|t-1}, y_t] \rightarrow$ Output: $[\hat{x}_{t|t}, P_{t|t}]$

a) 観測ヤコビアン

$$\hat{H}_t = \left[\frac{\partial h_t}{\partial x_t} \right]_{x_t = \hat{x}_{t|t-1}}$$

b) 拡張カルマンゲイン

$$K_t = P_{t|t-1} \hat{H}_t^T [\hat{H}_t P_{t|t-1} \hat{H}_t^T + R_t]^{-1}$$

c) 濾波推定値

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t [y_t - h_t(\hat{x}_{t|t-1})]$$

d) 濾波推定誤差共分散行列

$$P_{t|t} = P_{t|t-1} - K_t \hat{H}_t P_{t|t-1}$$

3) 時間更新ステップ Input: $[\hat{x}_{t|t}, P_{t|t}] \rightarrow$ Output: $[\hat{x}_{t+1|t}, P_{t+1|t}]$

a) 1 段予測推定値

$$\hat{x}_{t+1|t} = f_t(\hat{x}_{t|t})$$

b) 状態遷移ヤコビアン

$$\hat{F}_t = \left[\frac{\partial f_t}{\partial x_t} \right]_{x_t = \hat{x}_{t|t}}$$

c) 予測誤差共分散行列

$$P_{t+1|t} = \hat{F}_t P_{t|t} \hat{F}_t^T + Q_t$$

4) $t \leftarrow t + 1$ としてステップ 2) に戻る。

上記のステップを繰り返すことにより、濾波推定値 $\hat{x}_{t|t}$ 、1 段予測推定値 $\hat{x}_{t+1|t}$ を逐次的に計算することができる。

5.2.2 PF の原理

本項においては、統計学におけるベイズ推定に基づいて設計されたフィルタである PF についてその原理を説明する。

最初に、次の離散時間非線形システムを考える

$$x_{t+1} = f_t(x_t, w_t) \quad (5.11)$$

$$y_t = h_t(x_t, v_t) \quad (5.12)$$

ここで、 $x_t \in \mathbb{R}^n$ は状態ベクトル、 $y_t \in \mathbb{R}^p$ は観測ベクトル、 $w_t \in \mathbb{R}^m$ はシステム雑音、 $v_t \in \mathbb{R}^p$ は観測雑音を表すものとする。また、 $f_t(x_t)$ 、 $h_t(x_t)$ は微分可能であるとする。 w_t 、 v_t は平均値 0 の互いに無関係な白色雑音であり、かつ雑音は初期状態ベクトル x_0 とは無相関であるとする。また、雑音の分布は一般に非ガウス分布であるとする。

PF の基本的な考え方は、事後確率分布をその分布から独立にサンプルされた多数の粒子によって近似的に表現することである。ここで、分布を近似する粒子の

集まりをアンサンブルと呼ぶ。PF においては、隠れたパラメータ列 x_t を観測値 y_t のみから推定する。具体的なアルゴリズムは以下の通りである [73]。

Algorithm3(PF)

1) 事前分布 $p(x_0)$ に従って初期アンサンブル $X_{0|-1} = [x_{0|-1}^{(1)}, \dots, x_{0|-1}^{(M)}]$ を生成して、 $t = 0$ とおく。

2) 観測更新ステップ Input: $[X_{t|t-1}, y_t] \rightarrow$ Output: $[X_{t|t}]$

a) 尤度の計算

$$\alpha_t^{(i)} = p(y_t | x_t = x_{t|t-1}^{(i)}), i = 1, \dots, M$$

b) リサンプリングにより濾波アンサンブル行列を計算

$$X_{t|t} = [x_{t|t}^{(1)}, x_{t|t}^{(2)}, \dots, x_{t|t}^{(M)}]$$

c) 濾波推定値

$$\hat{x}_{t|t} = \frac{1}{M} \sum_{i=1}^M x_{t|t}^{(i)}$$

3) 時間更新ステップ Input: $[X_{t|t}] \rightarrow$ Output: $[X_{t+1|t}]$

a) システム雑音のサンプルを生成

$$w_t^{(i)} \sim p(w_t), i = 1, \dots, M$$

b) 予測アンサンブル行列

$$x_{t+1|t}^{(i)} = f_t(x_{t|t}^{(i)}, w_t^{(i)}), i = 1, \dots, M$$

$$X_{t+1|t} = [x_{t+1|t}^{(1)}, x_{t+1|t}^{(2)}, \dots, x_{t+1|t}^{(M)}]$$

4) $t \leftarrow t + 1$ として、ステップ 2) に戻る。

5.3 0次ホールド・線形外挿・2次外挿を用いたロボットの位置推定のシミュレーション結果

本節においては、0次ホールド、線形外挿、2次外挿の各手法を用いた場合の通信損失時のロボット位置推定に対する有効性を、実際にロボットの動きをコンピュータ上でシミュレートすることにより考察する。シミュレーション条件としては以下の通りである：初期状態として台数 $n = 10$ のロボットを2次元平面内の 100×100 ユニットの正方形内のランダムな位置に配置する。ロボットは(3.8)式及び(3.9)式に従い、互いに位置情報を交換しながら1辺の長さが10ユニットの正三角形格子を形成するものとする。ロボット間の通信は一定の損失確率 α で失われるものとし、通信損失により得られなかった情報は5.1節で示した3種類の手法(0次ホールド、線形外挿、2次外挿)を用いて外挿するものとする。ここで、通信損失確率 α とは、ロボット間で通信が m 回行われた場合に、そのうちの $m\alpha$ 回の通信が損失することを表す値である。また、推定精度を比較するために、5.2節で述べた2種類のフィルタ(EKF及びPF)を用いてロボット位置を推定した結果についても同時に計算を行った。以上のシミュレーションを同条件で10回行い、その時のロボット位置の相対誤差を平均したものをグラフに示す。なお、相対誤差についての評価指標としては、ロボット間の通信損失が存在しない場合のロボット位置の変化も同時に計算することにより、通信損失時と通信損失が無い場合とのロボット位置の相対誤差の平均値である以下の式を用いた。

$$l_\alpha = \frac{\sum_{j=1}^{n-1} \sqrt{(x_j - x'_j)^2 + (y_j - y'_j)^2}}{n-1} \quad (5.13)$$

ここで、 (x_i, y_i) はロボット間に通信損失が無い場合のロボット i の位置を、 (x'_i, y'_i) はロボット間に通信損失が起こった場合のロボット i の位置を表している。即ち、この式の値が小さいほど各ロボットの位置が通信損失がない場合の本来あるべき位置に近い位置に存在する、つまり推定精度が高いことを意味している。

まずは図5.1に通信損失確率 $\alpha = 0.2$ の場合についてのステップ数 $N=210$ までの l_α の時間変化を示す。図中で赤線が0次ホールド、青線が線形外挿、緑線が2次外挿、茶線がEKF、紫線がPFの結果を表している。グラフより、0次ホールド及

び線形外挿の推定誤差は比較的小さい値に収まっているのに対し、2次外挿の推定誤差は他の2つに比べて大きくなっていることが分かる。このような結果が得られた理由としては、2次外挿においては直前の値の変化率を更に大きくする方向に外挿を行うため、一度誤差が大きくなるとその誤差を更に拡大してしまうためではないかと考えられる。また、既存の3種類の外挿手法による推定結果は、理論的なフィルタと比べると、0次ホールドと線形外挿についてはEKFとは近い結果が得られたものの、PFと比べるとその推定精度は明らかに悪く、改善の余地があるものと考えられる。

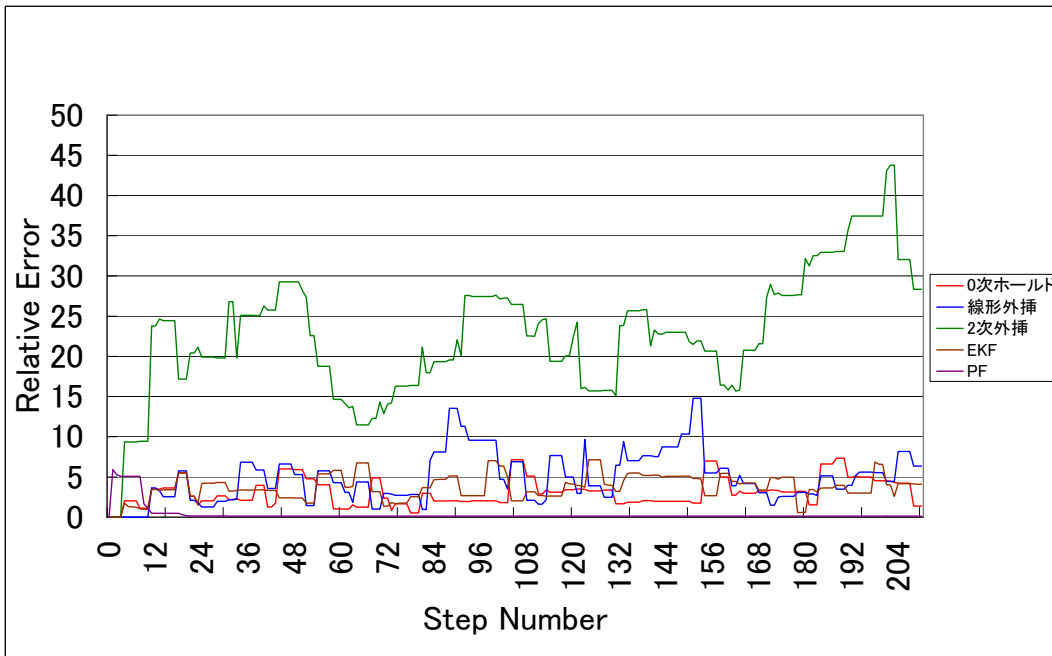


図 5.1: 通信損失確率 $\alpha=0.2$ の時のロボット位置の相対誤差の変化

また、図 5.2 ~ 図 5.4 に 5.1 節で述べた 3 種類の外挿法を用いて推定を行った場合に、4.4 節で導入した目標とする正三角形への収束度を表す評価指標 (EI4) の推移を示すグラフを示す。ここで、EI4 の値が一定値に収束し変化しなくなった時点でロボットが目標形状である正三角形格子を形成したと判断し、その時の時間を収束時間と定義する。3 つのグラフの比較から分かるように、0 次ホールドや線形外挿を用いた場合の結果については、ロボットが一定時間で目標とする形状に収束しているのに対して、2 次外挿においては一定値に収束することなく、その値が発散していることが分かる。

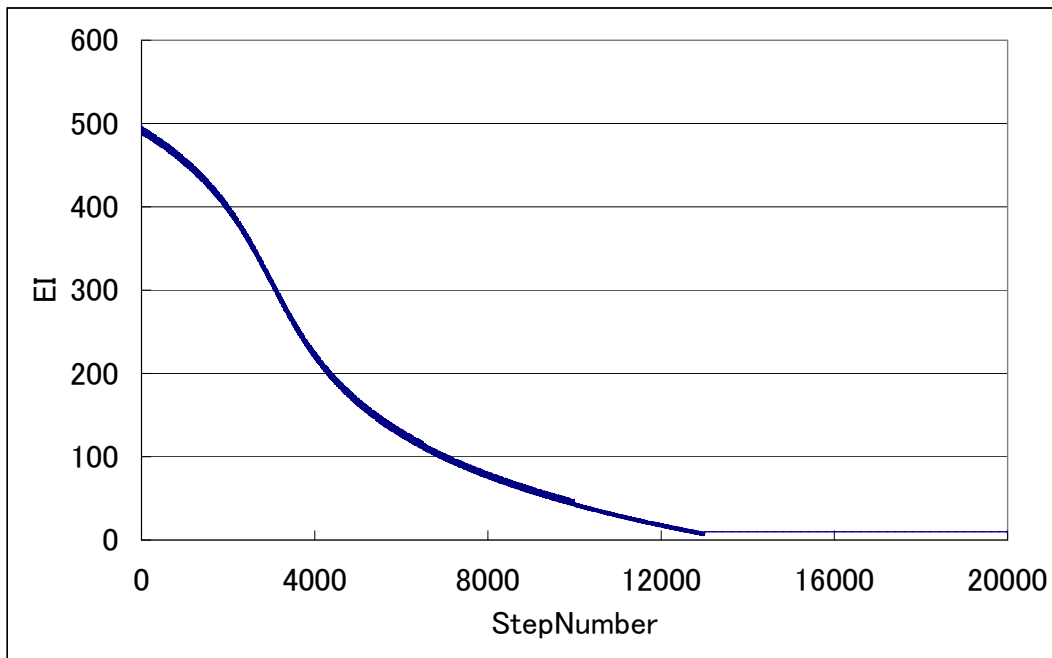


図 5.2: 0 次ホールドに対する EI4 値の変化 ($\alpha=0.2$)

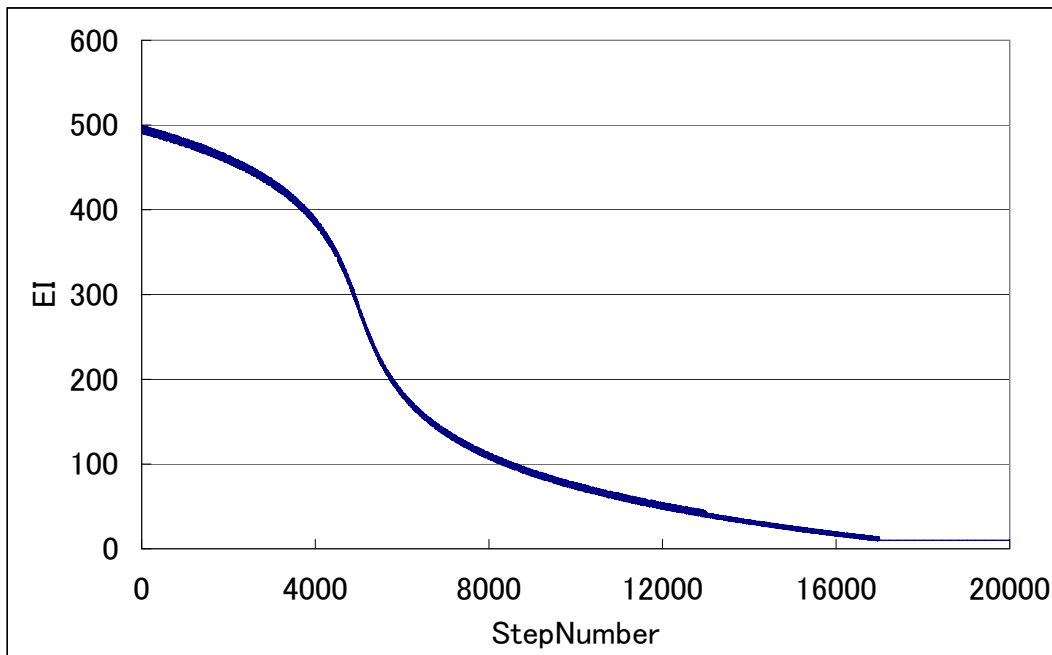


図 5.3: 線形外挿に対する EI4 値の変化 ($\alpha=0.2$)

続いて、図 5.5 に通信損失確率 $\alpha = 0.5$ の場合についてのステップ数 $N=210$ までの l_α の時間変化を示す。なお、グラフ中の凡例は図 5.1 と同様である。この場合

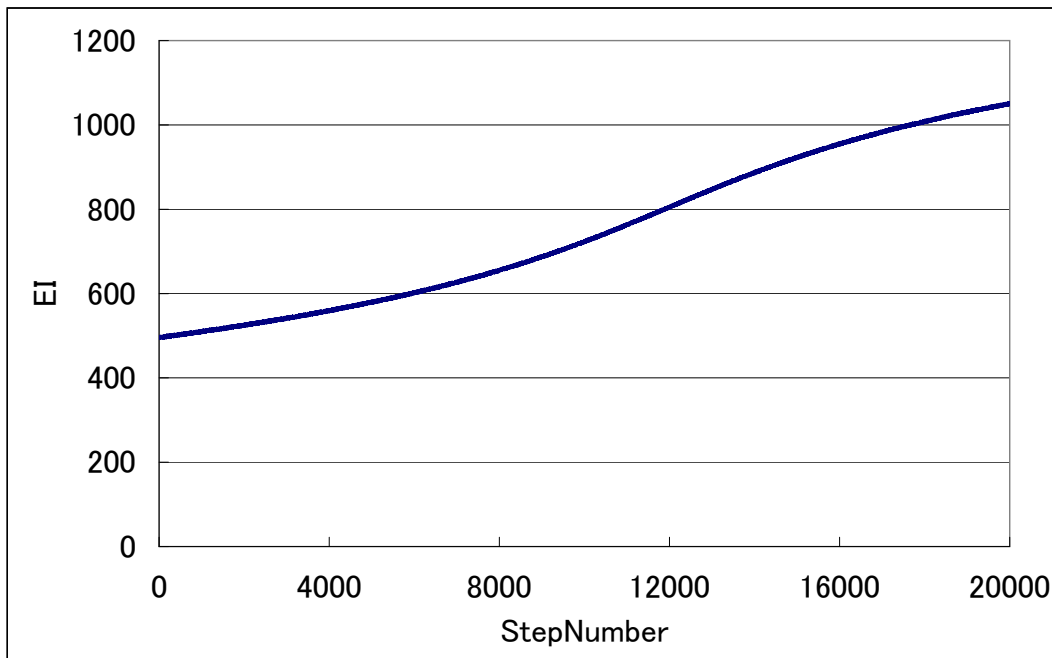


図 5.4: 2次外挿に対する EI4 値の変化 ($\alpha=0.2$)

も図 5.1 の場合と同様、0 次ホールドと線形外挿の推定誤差はそれほど大きくなっていないのに対して、2 次外挿の推定誤差の値は更に増加されているなっていることが分かる。通信損失確率の値を大きくすると推定誤差が大きくなる理由としては、損失確率が大きくなることで実際に得られるデータの割合が減少するため、外挿式を用いて値を外挿する割合が大きくなり、誤差が蓄積されやすくなり結果的に大きな推定誤差を生む結果となったと考えられる。また、 $\alpha = 0.2$ の場合と比べて、0 次ホールドと線形外挿については EKF とは近い推定誤差を得られたものの、PF による計算結果と比較した場合の相対精度の差が更に拡大されていることが分かる。これは、通信損失確率 α の値が大きくなったことに伴い、外挿式を用いて計算を行う確率が増大したため、高い推定精度を持たない既存の 3 種類の外挿法においては推定誤差が大きくなったためだと考えられる。

また、図 5.6 ~ 図 5.8 に 5.1 節で述べた 3 種類の外挿法を用いて推定を行った場合の収束度の評価指標 (EI4) の推移を示すグラフを示す。3 つのグラフの比較から分かるように、0 次ホールドと線形外挿を用いた場合については $\alpha = 0.2$ の場合と比べて完全な正三角形格子に収束するまでの時間が長くなっており、また 2 次外挿の場合においては、ロボットは目標となる正三角形格子に収束することなく発

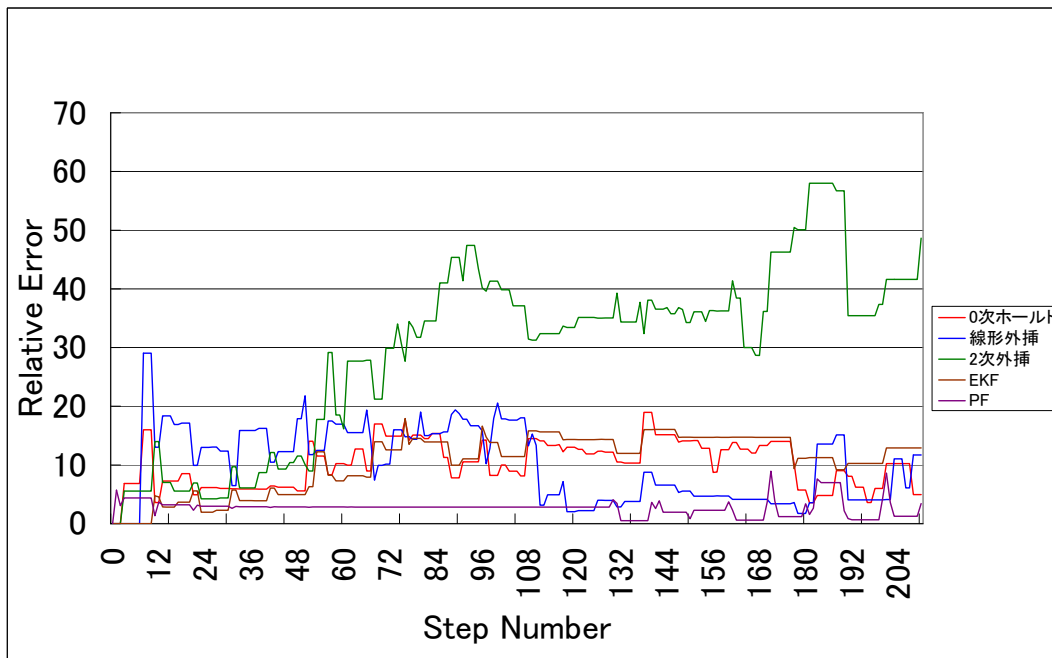


図 5.5: 通信損失確率 $\alpha=0.5$ の時のロボット位置の相対誤差の変化

散していった。これは、通信損失確率 α の値が大きくなったことにより、ロボットが正しい動きをするために必要な十分な情報を得ることができなくなったためと考えられる。

5.4 速度外挿の概要

前節で行ったシミュレーション結果より、既存の0次ホールド、線形外挿、2次外挿の3つの外挿手法においては群ロボットの formation 制御において通信損失時に十分な精度でロボット位置を推定することができなかった。このため、本節においてはこれらの単純な外挿式に改良を加えることにより、推定精度を向上させることを目指す。

本節では、速度0次ホールド及び速度線形外挿と呼ばれる新たな外挿法を提案する。速度0次ホールドとは、通信損失が発生した場合に、その直前のロボットの速度をそのまま用いてロボットを移動させるというものである。また、速度線形外挿とは、通信損失時にロボットの1ステップ前及び2ステップ前の速度を1次関数に fitting させることにより現在のロボット速度を計算し、この速度でロボット

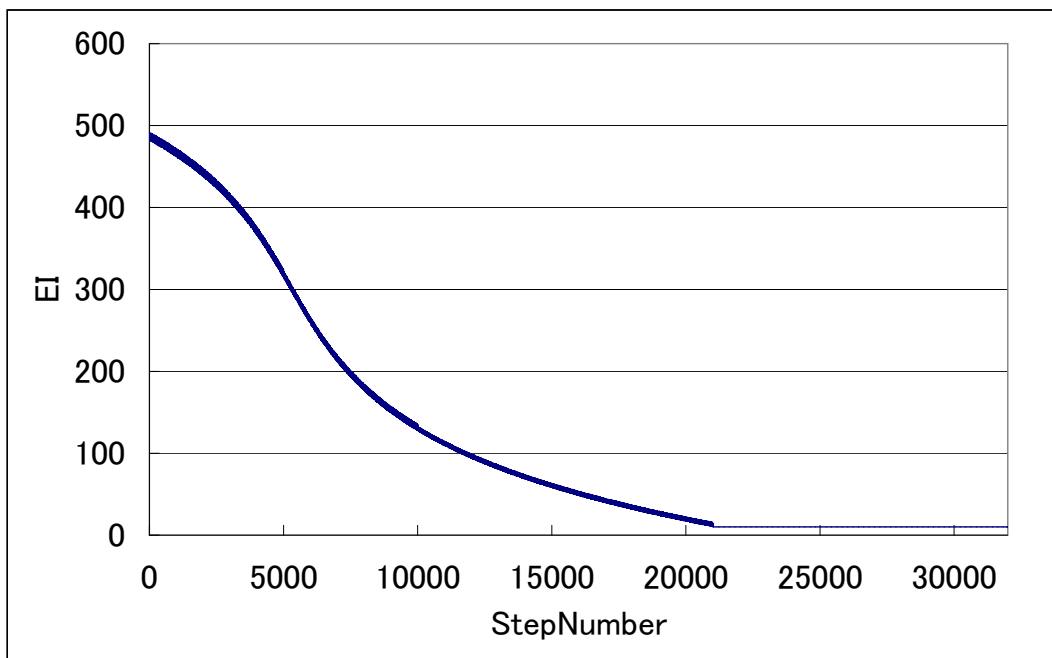


図 5.6: 0 次ホールドに対する EI4 値の変化 ($\alpha=0.5$)

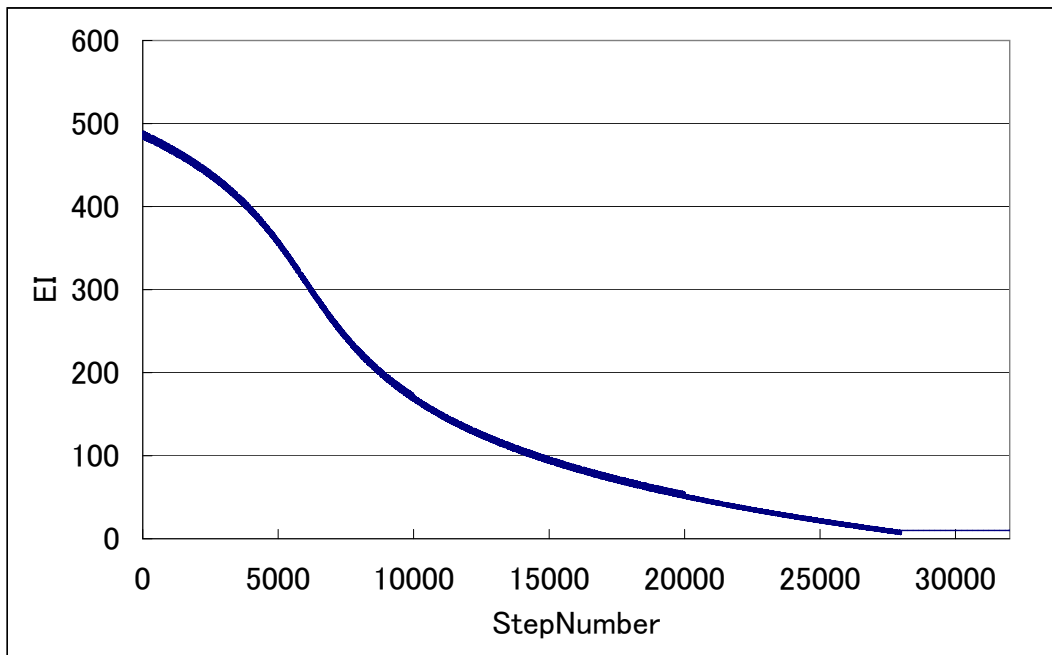


図 5.7: 線形外挿に対する EI4 値の変化 ($\alpha=0.5$)

を次の位置に移動させるというものである。以上を式で表すと、次の通りとなる。

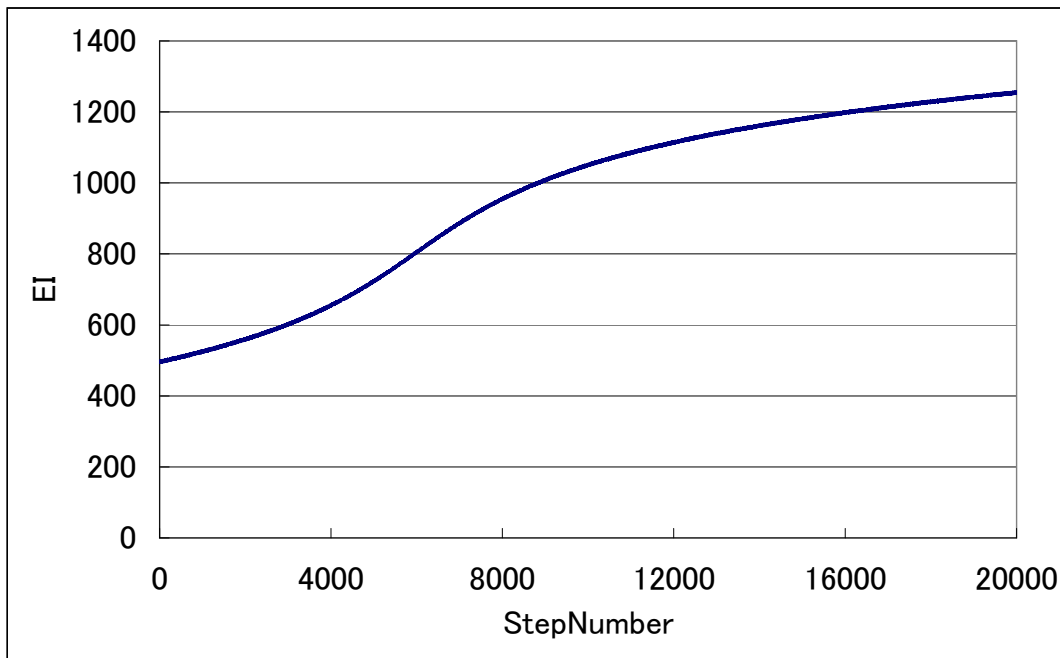


図 5.8: 2次外挿に対する EI4 値の変化 ($\alpha=0.5$)

5.4.1 速度 0 次ホールド

本項においては、速度 0 次ホールドについて説明する。時刻 kt において通信損失が発生したものとする。ロボット i の時刻 $k(t-1)$ における速度を

$$\vec{v}_i(k(t-1)) = (x_i(k(t-1)) - x_i(k(t-2)), y_i(k(t-1)) - y_i(k(t-2))) \quad (5.14)$$

とする時、現在時刻 kt におけるロボット i の速度を

$$\vec{v}_i(kt) = \vec{v}_i(k(t-1)) \quad (5.15)$$

として、現在位置を

$$(x_i(kt), y_i(kt)) = (x_i(k(t-1)), y_i(k(t-1))) + \vec{v}_i(kt) \quad (5.16)$$

として推定するというものである。

5.4.2 速度線形外挿

本項においては、速度線形外挿について説明する。時刻 kt において通信損失が発生したものとする。ロボット i の時刻 $k(t-1)$ 及び時刻 $k(t-2)$ における速度をそれぞれ

$$\vec{v}_i(k(t-1)) = (x_i(k(t-1)) - x_i(k(t-2)), y_i(k(t-1)) - y_i(k(t-2))) \quad (5.17)$$

$$\vec{v}_i(k(t-2)) = (x_i(k(t-2)) - x_i(k(t-3)), y_i(k(t-2)) - y_i(k(t-3))) \quad (5.18)$$

とする時、現在時刻 kt におけるロボット i の速度を

$$v_{xi}(k(t-1)) = a_x(k(t-1)) + b_x \quad (5.19)$$

$$v_{yi}(k(t-1)) = a_y(k(t-1)) + b_y \quad (5.20)$$

$$v_{xi}(k(t-2)) = a_x(k(t-2)) + b_x \quad (5.21)$$

$$v_{yi}(k(t-2)) = a_y(k(t-2)) + b_y \quad (5.22)$$

(但し、 v_{xi} 、 v_{yi} はそれぞれ \vec{v}_i の x 成分、 y 成分を表す)

の4本の連立方程式を解いて a_x 、 a_y 、 b_x 、 b_y を求め、

$$\vec{v}_i(kt) = (v_{xi}(kt) = a_x(kt) + b_x, v_{yi}(kt) = a_y(kt) + b_y) \quad (5.23)$$

として、現在位置

$$(x_i(kt), y_i(kt)) = (x_i(k(t-1)), y_i(k(t-1))) + v_i(\vec{k}t) \quad (5.24)$$

として推定するというものである。

5.5 速度外挿を用いたシミュレーション結果

本節においては、前節で提案した速度外挿式を用いて通信損失時にロボット位置を推定したシミュレーション結果について示す。シミュレーション条件につい

では、5.3 節と同様である。同条件でのシミュレーションを 10 回行い、その結果を平均したものをグラフとして示した。

なお、相対誤差についての評価指標としては、5.3 節で示した式 (5.13) を用いるものとする。

まずは図 5.9 に通信損失確率 $\alpha = 0.2$ の場合についてのステップ数 $N=210$ までの l_α の時間変化を示す。図中で赤線が 0 次ホールド、青線が線形外挿、緑線が 2 次外挿、茶線が速度 0 次ホールド、紫線が速度線形外挿を表している。グラフより、速度 0 次ホールド、速度線形外挿のいずれも単純な 0 次ホールドや線形外挿の場合よりも大きな相対誤差の値を示し、推定精度を改善することはできなかった。

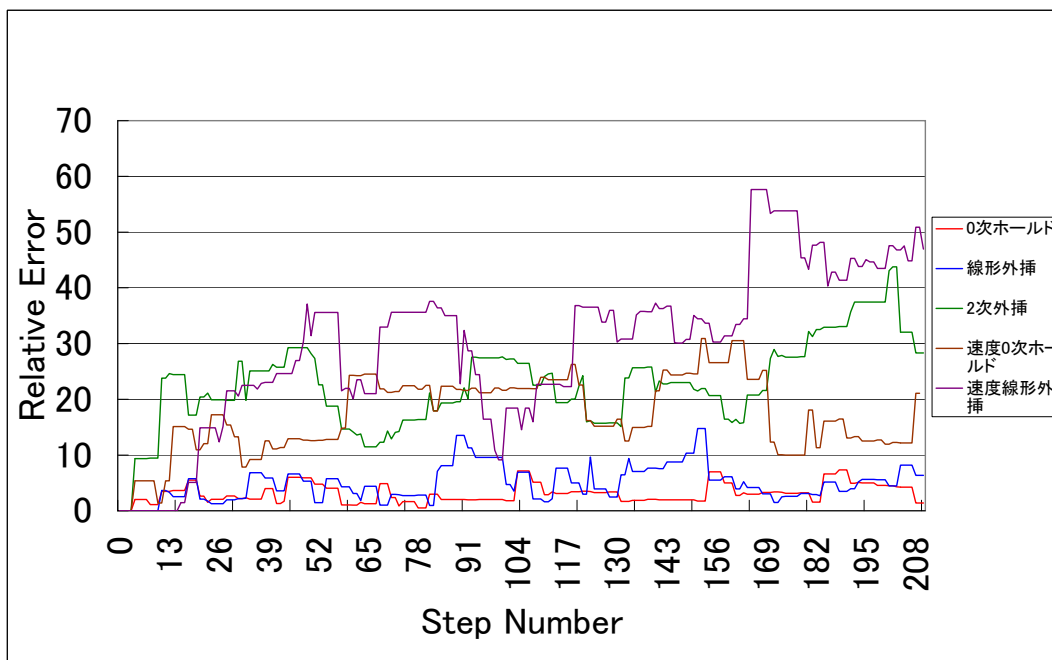


図 5.9: 通信損失確率 $\alpha=0.2$ の時のロボット位置の相対誤差の変化 (速度外挿)

続いて、図 5.10 に通信損失確率 $\alpha = 0.5$ の場合についてのステップ数 $N=210$ までの l_α の時間変化を示す。図中の凡例は図 5.9 と同様である。この場合も図 5.9 の場合と同様、速度 0 次ホールド、速度線形外挿のいずれも単純な 0 次ホールドや線形外挿と比べて大きな相対誤差を示すことが確認された。

このように、本節で提案した速度 0 次ホールド及び速度線形外挿によって推定誤差を小さくすることができないばかりか、かえって誤差を大きくする結果となってしまう。この原因としては、次のような理由が考えられる。一般に、群ロボッ

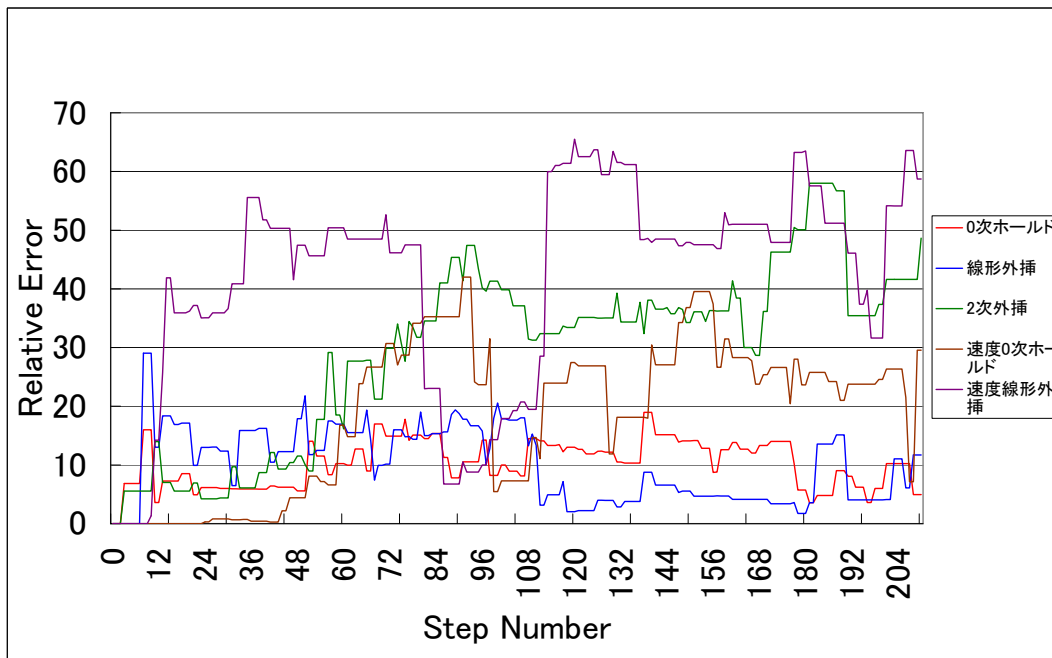


図 5.10: 通信損失確率 $\alpha=0.5$ の時のロボット位置の相対誤差の変化 (速度外挿)

トの formation 制御において、各ロボットは目標形状に向けて徐々にその移動距離を小さくしていく傾向があると考えられる。このため、過去の速度をそのまま、または 1 次関数で外挿する今回の手法においては、本来ロボットが動くべき速度よりも大きな速度でロボットを動かしてしまう可能性があり、その結果として相対誤差の大きさが大きくなってしまったのではないかと考えられる。

第 6 章

指数関数を用いたロボット位置の外挿

前章で述べた 3 種類の単純な外挿手法は、計算量をほとんど必要としないという点からは群ロボットに適していると言えるものの、推定精度の面から見るとその精度は決してよいものではなく、改善する余地があるものと思われる。一方で、5.2 節で紹介した通り、観測できない現在の状態を過去の状態から推定する手法としては、制御理論の分野で一般的に知られているカルマンフィルタに代表されるフィルタが存在するが、こうしたフィルタは一般に多くの計算量を必要とするため、単純な計算能力しか持たない場合が多い群ロボットが直接利用することは難しいと考えられる。このため、計算量を抑えた上で高い精度で推定を行うための手法を開発する必要がある。そのため、本章において群ロボットの従う運動方程式に着目した新たな外挿手法を提案する。本論文で取り扱う群ロボットの正三角形格子を目標形状とする formation 制御においては、各ロボットが目標形状に向かって指数関数的に収束していくことが文献 [11] により示されている。この特性に注目し、指数関数的に収束していくロボットの動きを指数関数式を用いて外挿すれば、低い計算量でかつ高い精度で推定を行うことができるのではないかと考えるの元に、指数関数を用いた新たな外挿式を本章において提案する。

6.1 指数関数を用いた外挿式 (EE)

本節では、外挿式として単純な指数関数を用いた手法 (指数外挿 : Exponential Extrapolation (EE)) を導入する。まず最初に、時刻 t におけるロボット r_i の隣人

のロボット r_{n1} 又は r_{n2} の位置ベクトルに対応する状態量 $N_i(t)$ を定義する。時刻 t における状態量 $N_i(t)$ が通信損失等の理由により観測できなかった場合、ロボット r_i は時刻 $t - 2$ 及び $t - 1$ において観測された状態量 $N_i(t - 2)$ 及び $N_i(t - 1)$ を以下で導入する指数関数を用いた外挿式に代入することにより、 $N_i(t)$ の値を推定するものとする。ここで、 $x_i(t)$ ($y_i(t)$) を $N_i(t)$ の x (y) 要素とする。通信損失時に $x_i(t)$ 及び $y_i(t)$ は以下の手法で外挿される。ここでは $x_i(t)$ についてのみ述べるが、 $y_i(t)$ についても同様である。

- Case-1: $x_i(t - 2) > x_i(t - 1)$

この時、以下の指数関数式を外挿式として用いる。

$$x_i(t) = A_i e^{-B_i t} \quad (6.1)$$

但し、 A_i 及び B_i は共に正の定数である。(6.1) 式を時刻 $t - 1$ 及び $t - 2$ に適用すると、以下の通りとなる。

$$x_i(t - 1) = A_i e^{-B_i(t-1)} \quad (6.2)$$

$$x_i(t - 2) = A_i e^{-B_i(t-2)} \quad (6.3)$$

(6.2) 式及び (6.3) 式より、 e^{-B_i} は以下のように求められる。

$$e^{-B_i} = \frac{x_i(t - 1)}{x_i(t - 2)} \quad (6.4)$$

(6.4) 式を (6.1) 式に代入することにより、現在の位置 $x_i(t)$ は以下のように導出される。

$$x_i(t) = A_i e^{-B_i t} = A_i e^{-B_i(t-1)} \cdot e^{-B_i} = \frac{x_i(t - 1)}{x_i(t - 2)} \cdot x_i(t - 1) \quad (6.5)$$

- Case-2: $x_i(t - 2) < x_i(t - 1)$

この時、以下の指数関数式を外挿式として用いる。

$$x_i(t) = A_i(1 - e^{-B_i t}) \quad (6.6)$$

但し、 A_i 及び B_i は共に正の定数である。(6.6) 式を、時刻 $t - 1$ 及び $t - 2$ に適用すると、以下の通りとなる。

$$x_i(t - 1) = A_i(1 - e^{-B_i(t-1)}) \quad (6.7)$$

$$x_i(t - 2) = A_i(1 - e^{-B_i(t-2)}) \quad (6.8)$$

ここで、一般性を失うことなく時刻 $(t - 1)$ 及び $(t - 2)$ をそれぞれ 2 及び 1 で置き換えると、 $x_i(1)$ 及び $x_i(2)$ の値は以下の通りとなる。

$$x_i(2) = A_i(1 - e^{-2B_i}) \quad (6.9)$$

$$x_i(1) = A_i(1 - e^{-B_i}) \quad (6.10)$$

(6.9) 式及び (6.10) 式より、 A_i 及び e^{-B_i} は以下のように求められる。

$$A_i = \frac{[x_i(1)]^2}{x_i(2)} \quad (6.11)$$

$$e^{-B_i} = \frac{x_i(1) - x_i(2)}{x_i(1)} = 1 - \frac{x_i(2)}{x_i(1)} \quad (6.12)$$

(6.11) 式及び (6.12) 式より、 $x_i(3)$ の値は以下のように求められる。

$$x_i(3) = A_i(1 - e^{-3B_i}) = \frac{[x_i(1)]^2}{x_i(2)} \cdot \left[1 - \left[1 - \frac{x_i(2)}{x_i(1)}\right]^3\right] \quad (6.13)$$

(6.13) 式を時刻 t に一般化することにより、現在の位置 $x_i(t)$ は以下のように導出される。

$$x_i(t) = \frac{[x_i(t - 2)]^2}{x_i(t - 1)} \cdot \left[1 - \left[1 - \frac{x_i(t - 1)}{x_i(t - 2)}\right]^3\right] \quad (6.14)$$

なお、ロボット r_i が時刻 T 毎にサンプリングを行っている場合には、上述の式の t を Ts (ここで、 s はその時点までのサンプリングの回数) で置換することにより、同様の議論が成り立つ。

6.2 指数関数を用いた外挿式 (EE) の数学的特性

本節では、前節において提案した指数外挿式 ((6.5) 式及び (6.14) 式) の数学的特性について考察する。まず最初に、(6.5) 式及び (6.14) 式を用いて外挿を行った場合の収束性について議論する。

ここで、 $x_i(t-2) > x_i(t-1)$ かつ $y_i(t-2) > y_i(t-1)$ の場合を想定する。この場合、 $x_i(t)$ 及び $y_i(t)$ は共に (6.5) 式を用いて外挿される。(6.3) 式より、 $x_i(t-2)$ 及び $y_i(t-2)$ の値はそれぞれ以下の通りとなる。

$$\begin{aligned}x_i(t-2) &= A_{x_i} e^{-B_{x_i}(t-2)} \\y_i(t-2) &= A_{y_i} e^{-B_{y_i}(t-2)}\end{aligned}\tag{6.15}$$

同様に、 $x_i(t-1)$ 及び $y_i(t-1)$ についても (6.2) 式より以下のようなになる。

$$\begin{aligned}x_i(t-1) &= A_{x_i} e^{-B_{x_i}(t-1)} \\y_i(t-1) &= A_{y_i} e^{-B_{y_i}(t-1)}\end{aligned}\tag{6.16}$$

この時、(6.1) 式を用いて $x_i(t)$ 及び $y_i(t)$ は以下の式で外挿される。

$$\begin{aligned}x_i(t) &= A_{x_i} e^{-B_{x_i}t} \\y_i(t) &= A_{y_i} e^{-B_{y_i}t}\end{aligned}\tag{6.17}$$

ここで、(6.15) 式及び (6.16) 式を (3.10) 式に代入することにより、 $d_i(t-2)$ 及び $d_i(t-1)$ はそれぞれ以下のように計算される。

$$\begin{aligned}d_i(t-2) &= (d_i(0) - d_r) e^{-a(t-2)} + d_r \\&= \sqrt{(x_i(t-2) - x_g)^2 + (y_i(t-2) - y_g)^2} \\&= \sqrt{(A_{x_i} e^{-B_{x_i}(t-2)} - x_g)^2 + (A_{y_i} e^{-B_{y_i}(t-2)} - y_g)^2}\end{aligned}\tag{6.18}$$

$$\begin{aligned}
d_i(t-1) &= (d_i(0) - d_r)e^{-a(t-1)} + d_r \\
&= \sqrt{(x_i(t-1) - x_g)^2 + (y_i(t-1) - y_g)^2} \\
&= \sqrt{(A_{x_i}e^{-B_{x_i}(t-1)} - x_g)^2 + (A_{y_i}e^{-B_{y_i}(t-1)} - y_g)^2} \quad (6.19)
\end{aligned}$$

なお、 $p_{ct} = (x_g, y_g)$ は3台のロボットによって構成される三角形の重心の座標を表すものとする。続いて、(6.17)式より $d_i(t)$ の値は以下のように計算される。

$$\begin{aligned}
d_i(t) &= \sqrt{(x_i(t) - x_g)^2 + (y_i(t) - y_g)^2} \\
&= \sqrt{(A_{x_i}e^{-B_{x_i}t} - x_g)^2 + (A_{y_i}e^{-B_{y_i}t} - y_g)^2} \quad (6.20)
\end{aligned}$$

((6.20)式)² - ((6.19)式)² と ((6.19)式)² - ((6.18)式)² とを比較することにより、以下の関係が成り立つ。

$$\begin{aligned}
&((d_i(t))^2 - (d_i(t-1))^2)^{\frac{1}{t-1}} \\
&= ((d_i(t-1))^2 - (d_i(t-2))^2)^{\frac{1}{t-2}} \quad (6.21)
\end{aligned}$$

(6.21)式を式変形することにより、(6.20)式は以下のように表すことができる。

$$d_i(t) = \sqrt{(d_i(t-1))^2 - \left((d_i(t-2))^2 - (d_i(t-1))^2 \right)^{\frac{t-1}{t-2}}} \quad (6.22)$$

ここで、 $d_i(t)$ の値が d_r にへの収束することを示すため、リアプノフの安定性理論を用いる [74]。最初に、リアプノフ関数の候補として以下の関数を定義する。

$$f_i = \frac{1}{2}(d_i(t) - d_r)^2 \quad (6.23)$$

(6.23)式で表される関数は、 $d_i \neq d_r$ の時を除いて正定である。(6.23)式を微分することにより、

$$\dot{f}_i = \dot{d}_i(t)(d_i(t) - d_r) \quad (6.24)$$

となるが、この式は負定であることは明らかである。よって、平衡点 d_r は安定であることから、リアプノフの安定性理論より、 d_i は $d_r = \frac{d_y}{\sqrt{3}}$ に収束する。

なお、 $x_i(t-2) < x_i(t-1)$ 又は $y_i(t-2) < y_i(t-1)$ の場合も、外挿式として (6.5) 式の代わりに (6.14) 式を用いることにより、同様の議論を行うことでその安定性を証明できる。

6.3 指数関数を用いた外挿式 (EE) の推定精度

本節においては、6.1 節で提案した指数外挿式を用いて状態を推定した場合に得られる推定値の推定精度について考察する。

本節では、各ロボットは以下の微分方程式に従うことを仮定する。

$$\dot{x}_i(t) = ax_i(t) + b \quad (6.25)$$

上記の微分方程式の解は、以下のように与えられる。

$$x_i(t) = Ae^{-\lambda t} + B \quad (6.26)$$

但し、ロボットは目標位置に収束することから、 A 、 B 、 λ はそれぞれ正の定数である。

上式を時刻 $t-1$ 及び $t-2$ の場合に当てはめると、以下の通りとなる。

$$x_i(t-1) = Ae^{-\lambda(t-1)} + B \quad (6.27)$$

$$x_i(t-2) = Ae^{-\lambda(t-2)} + B \quad (6.28)$$

ここで、時刻 t において通信損失により情報が失われたと仮定し、ロボットが (6.1) 式を用いて失われた情報を外挿するものとする。

この時、(6.5) 式より、現在の時刻 t における状態は、以下の通り外挿される。

$$x_i(t) = \frac{\{x_i(t-1)\}^2}{x_i(t-2)} = \frac{A^2e^{-2\lambda(t-1)} + 2ABe^{-\lambda(t-1)} + B^2}{Ae^{-\lambda(t-2)} + B} \quad (6.29)$$

よって、式 (6.26) により与えられる実際の値との誤差 $e(t)$ は、

$$\begin{aligned}
e(t) &= \frac{A^2 e^{-2\lambda(t-1)} + 2ABe^{-\lambda(t-1)} + B^2}{Ae^{-\lambda(t-2)} + B} - (Ae^{-\lambda t} + B) \\
&= \frac{A^2 e^{-2\lambda(t-1)} + 2ABe^{-\lambda(t-1)} + B^2 - (Ae^{-\lambda t} + B)(Ae^{-\lambda(t-2)} + B)}{Ae^{-\lambda(t-2)} + B} \\
&= \frac{ABe^{-\lambda(t-2)}(e^{-\lambda} + 1)^2}{Ae^{-\lambda(t-2)} + B} \\
&= \frac{B(x(t-2) - B)}{x_i(t-2)}(e^{-\lambda} + 1)^2 \\
&= B\left(1 - \frac{B}{x_i(t-2)}\right)(e^{-\lambda} + 1)^2 \tag{6.30}
\end{aligned}$$

ここで、 $x_i(t)$ は減少関数であることを仮定しているので、 $B < x_i(t-2)$ であることから、 $0 < \left(1 - \frac{B}{x_i(t-2)}\right) < 1$ となる。

よって、指数関数を用いた外挿式 (EE) を使って外挿を行った場合の推定誤差は、 $B(e^{-\lambda} + 1)^2$ 以下となる。

6.4 Cramér-Rao の下界式

ある推定式を用いて何らかの状態量を推定する際に、その推定式によって得られた値がどの程度正確なのか、どの程度の誤差を含んでいるのかを評価する必要がある。ある状態を推定した際に、その推定精度の理論的な限界値を示す値として知られているのが、以下に示す推定誤差の分散に関する Cramér-Rao の下界式と呼ばれる不等式である。

Cramér-Rao の下界式 [75] とは、あるシステムの状態を推定した際に生じる推定誤差についての共分散行列 P の値が、以下の関係式を満たすというものである。この Cramér-Rao の下界式を用いて、位置推定における推定精度の理論的な限界を求める研究は、文献 [76] などに見られる。

Cramér-Rao の下界式は、以下の不等式で与えられる。

$$P \geq P^* = J^{-1} \tag{6.31}$$

ここで、 J はそのシステムにおけるフィッシャー情報行列を表しており、その各成分は以下の式で与えられる。

$$(J)_{ij} = E \left[\frac{\partial \ln p(x, y)}{\partial x_i} \frac{\partial \ln p(x, y)}{\partial x_j} \right] \quad (6.32)$$

$$= -E \left[\frac{\partial^2}{\partial x_i \partial x_j} \ln p(x, y) \right] \quad (6.33)$$

ここで、 $p(x, y)$ は x と y の結合確率密度を表し、また $E(\cdot)$ は期待値を表す。

すなわち、(6.31) 式及び (6.33) 式より、 x の i 要素 x_i の 2 乗平均推定誤差 $\bar{\epsilon}_i^2 = E[x_i - \hat{x}_i(y)]^2$ は以下の不等式を満たす。

$$\bar{\epsilon}_i^2 \geq J_{ii}^{-1} \quad (6.34)$$

但し、 \hat{x}_i は x_i の推定値を表す。

一般に、非線形なシステムにおいてフィッシャー情報行列 J の値を直接求めることは困難とされている [77]-[80] が、本論文においては、文献 [81] に示された以下の定理を用いて J の値を求めている。

定理：

”ある動的システムのフィッシャー情報行列の逆行列の値は、そのシステムを拡張カルマンフィルタ (EKF) を用いて推定した時のフィルタ共分散行列の値に等しい ”

証明：

次の非線形連続-離散推定問題を考える。連続時間システムのダイナミクスは以下の式に従うものとする。

$$\dot{x} = f(x, u, t) \quad (6.35)$$

ここで、 x は状態変数を表す n 次元ベクトルであり、 u は r 次元の入力ベクトルである。このダイナミクスモデルの未知の軌道は次のように推定される。

$$x(t) = x(t; x_0, u(\tau) : 0 \leq \tau \leq t) \quad (6.36)$$

ここで、 x_0 も未知であるとする。各サンプリング時間 $t = kT_s$ における x の値は以下のように表される。

$$x_k = x(kT_s; x_0, u(\tau) : 0 \leq \tau \leq kT_s), k = 1, 2, \dots \quad (6.37)$$

そして、離散観測は次の形式を取る。

$$z_k = h(x_k, k) + v_k = h_k + v_k, v_k \sim N(0, R_k) \quad (6.38)$$

ここで、 z_k は m 次元ベクトルである。上式は各サンプリング時間において利用できる。最後に、 x_0 についての先験的な不偏統計情報は利用可能であり、以下の形式を取るものとする。

$$\hat{x}_0 \sim N(x_0, S_0) \quad (6.39)$$

ここで、文献 [87] より、時刻 $t = KT_s$ における未知の状態ベクトル x の条件付き情報行列は以下の式で与えられる。

$$J_K = -E \left[\frac{\partial \ln p_K}{\partial x_K^2} \middle| X_K \right] \quad (6.40)$$

ここで、 p_K は Z_K の条件付き密度であり、以下の式で表される。

$$p_k = p_{Z_k|X_k} \quad (6.41)$$

そして、 X_K は次の集合である。

$$X_k = \{x_0, x_1, \dots, x_k\} \quad (6.42)$$

J_K が与えられると、不偏推定器に対する Cramér-Rao の下界は (6.25) 式で与えられる。特に、 x_K の各要素について、

$$\sigma_{iK}^2 = E[(x_{Ki} - \hat{x}_{Ki}(Z_k))^2] > (J_K^{-1})_{ii} = P_{Ki}^* \quad (6.43)$$

が成り立つ。ここで、 P_{Ki}^* は (6.31) 式中の P_K^* の (i, i) 要素を表す。仮定より、(6.38) 式及び (6.39) 式より、

$$p_{Z_K|X_K} = \frac{1}{(2\pi)^{n/2} |S_0|^{1/2}} e^{-1/2(x_0 - \hat{x}_0)^T S_0^{-1} (x_0 - \hat{x}_0)} \cdot \sum_{k=1}^K \frac{1}{(2\pi)^{m/2} |R_k|^{1/2}} e^{-1/2(x_k - h(x_k))^T R_k^{-1} (x_k - h(x_k))} \quad (6.44)$$

よって、

$$-\ln p_k = \text{constant} + \frac{1}{2}(x_0 - \hat{x}_0)^T S_0^{-1} (x_0 - \hat{x}_0) + \frac{1}{2} \sum_{k=1}^K (z_k - h(x_k))^T R_k^{-1} (z_k - h(x_k)) \quad (6.45)$$

(6.40) 式に従って $-\ln p_K$ の 2 階偏微分の期待値を取ると、以下ようになる。

$$J_K = \left(\frac{\partial x_0}{\partial x_K} \right)^T S_0^{-1} \left(\frac{\partial x_0}{\partial x_K} \right) + \sum_{k=1}^K \left(\frac{\partial h_k}{\partial x_K} \right) R_k^{-1} \left(\frac{\partial h_k}{\partial x_K} \right) \quad (6.46)$$

ここで、(6.46) 式は次のように定義することで再公式化される。

$$H_k = \frac{\partial h(x_k, k)}{\partial x_k} \quad (6.47)$$

よって、偏微分の連鎖律により、

$$J_K = \left(\frac{\partial x_0}{\partial x_K} \right)^T S_0^{-1} \left(\frac{\partial x_0}{\partial x_K} \right) + \sum_{k=1}^K \left(\frac{\partial x_k}{\partial x_K} \right)^T H_k^T R_k^{-1} H_k \left(\frac{\partial x_k}{\partial x_K} \right) \quad (6.48)$$

ここで、次の K 個の補助行列が J_K を評価するのに必要となる。

$$M_k = \frac{x_k}{x_K} = \left(\frac{\partial x_K}{\partial x_k} \right)^{-1} = \Phi_{K,k}^{-1}, k = 0, 1, 2, 3, \dots, K-1 \quad (6.49)$$

次のヤコビアン行列

$$F(t) = \frac{\partial f(x, u, t)}{\partial x} \quad (6.50)$$

が x 及び t について連続であるという一般条件の元で、文献 [88] より $\Phi_{K,k}$ は以下の式を初期条件 $\Phi(kT_s, kT_s) = I$ の元で $t = kT_s$ から KT_s まで積分することにより正確に求められる。

$$\dot{\Phi} = F(t)\Phi \quad (6.51)$$

加えて、連続特性

$$\Phi_{K,k} = \Phi_{K,K-1}\Phi_{K-1,k} \quad (6.52)$$

により (6.48) 式は以下の再帰的な形式で書くことができる。

$$J_K = (\Phi_{K-1}^{-1})^T J_{K-1} \Phi_{K-1}^{-1} + H_K^T R_K^{-1} H_K \quad (6.53)$$

ここで、 $\Phi_{K-1} = \Phi_{K,K-1}$ である。 P_K^* に関して、

$$(P_K^*)^{-1} = (\Phi_{K-1} P_{K-1}^* \Phi_{K-1}^T)^{-1} + H_K^T R_K^{-1} H_K \quad (6.54)$$

となるが、この式は EKF の共分散行列の逆行列に等しい。よって、定理は証明された。□

6.5 EE によるロボット位置推定のシミュレーション結果

本節においては、EE の有効性について実際にロボットの動きをコンピュータ上でシミュレートすることにより考察する。シミュレーション条件としては以下の通りである：初期状態として台数 $n = 10$ のロボットを 2 次元平面内の 100×100 ユニットの正方形内のランダムな位置に配置する。ロボットは (3.8) 式及び (3.9) 式に従い、互いに位置情報を交換しながら 1 辺の長さが 10 ユニットの正三角形格子を形成するものとする。ロボット間の通信は一定の損失確率 α で失われるものとし、通信損失によってロボットが得ることのできなかつた情報は 6.1 節で示した指数外挿式を用いて外挿するものとする。このシミュレーションを異なる初期位置から同条件で 10 回行い、その時のロボット位置の相対誤差を平均したものをグラフに示す。なお、相対誤差についての評価指標としては、5.3 節で示した式 (5.13) を用いるものとする。

また、ロボットの目標形状への収束度を測るための評価指標としては、4.4 節で提案した評価指標 (EI4) を用いて計算を用いる。4.5 節での計算結果より、ロボット台数 $n = 10$ の場合には評価指標 EI4 の値が 10.666 となった時に各ロボットによって完全な正三角形格子が形成されたとすることができる。EI4 の値が一定の値となった時の時刻を収束時間と定義する。

図 6.1 及び図 6.2 に、通信損失確率 α の値が 0.2 及び 0.5 の時の時間ステップ $N = 210$ までのシミュレーション結果を示す。各グラフでは横軸が時間ステップ、縦軸が各外挿手法を適用した場合の相対誤差の評価指標 l_α の値を表している。なお、比較のため単純な外挿手法として代表的な 0 次ホールド (ZOH) と計算時間が指数外挿と同程度になるように粒子数を減らした PF による推定を行った結果、及び理論的な推定限界として 6.4 節に示した Cramér-Rao の下界式を用いて求めた限界値についても同時に掲載した。

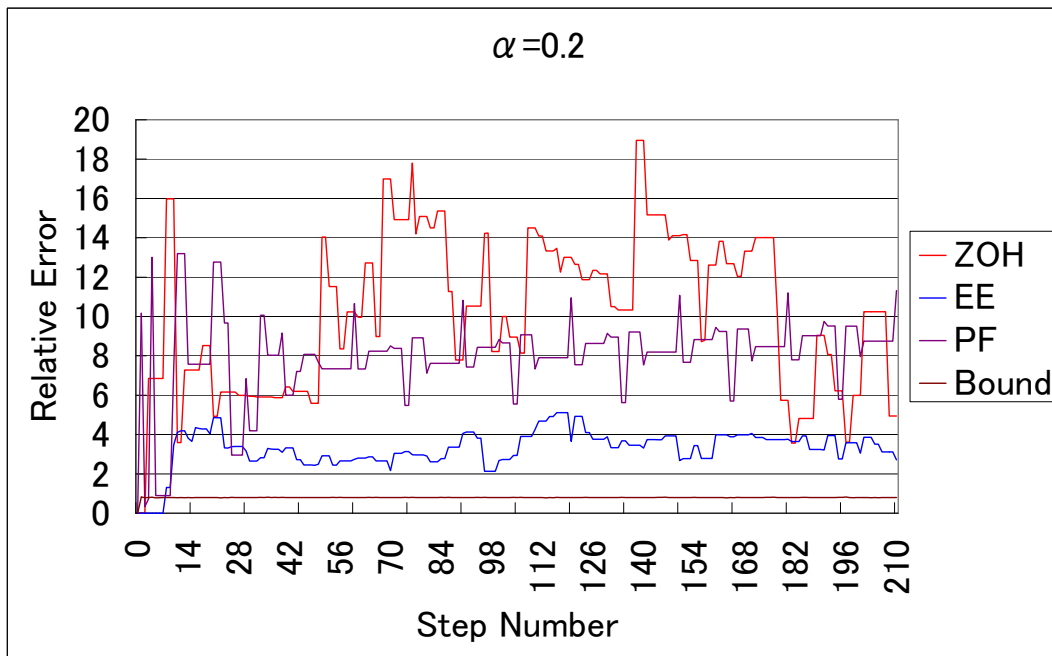


図 6.1: 通信損失確率 $\alpha=0.2$ の時のロボット位置の相対誤差の変化 (EE)

図 6.1 は通信損失確率 $\alpha = 0.2$ の場合の l_α の変化を表したグラフである。図中で赤線が 0 次ホールド、青線が指数外挿、紫線が PF、茶線が Cramér-Rao の下界式を用いて求めた推定誤差の理論的な限界値 (Bound) を表している。グラフより、6.1 節で提案した指数関数を用いた外挿手法は従来の単純な外挿手法や PF と比べ

て良好な推定結果を示していることが分かる。一方で、Cramér-Rao の下界式より求めた推定精度の限界と比較するとまだ誤差が残っていることが分かる。

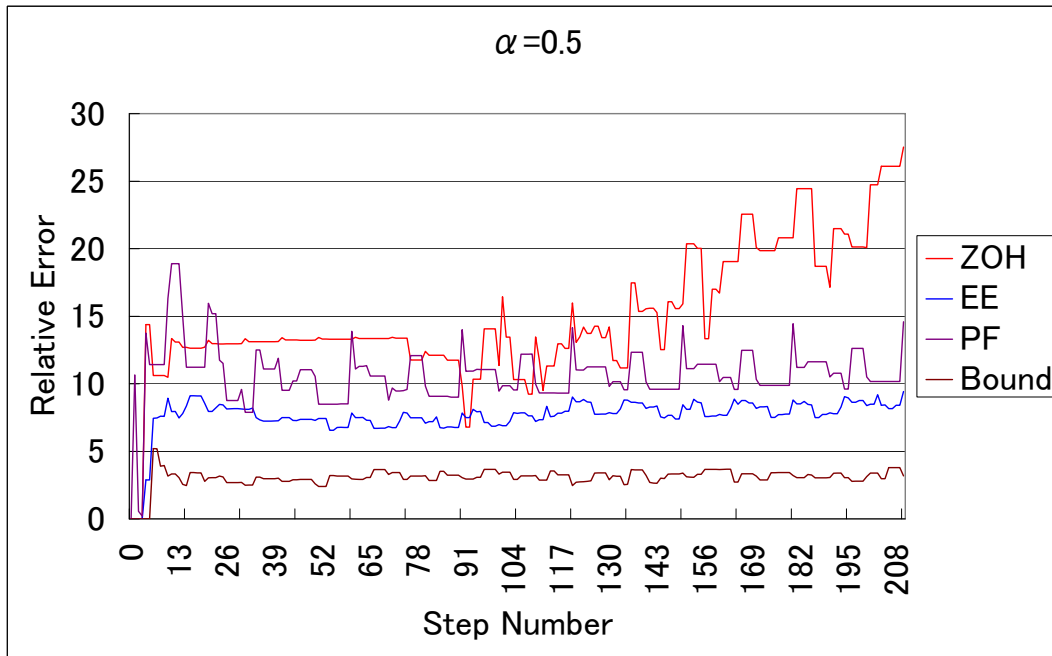


図 6.2: 通信損失確率 $\alpha=0.5$ の時のロボット位置の相対誤差の変化 (EE)

続いて、図 6.2 に通信損失確率 $\alpha = 0.5$ の場合の l_α の変化を表したグラフを示す。グラフ中の凡例は図 6.1 と同様である。グラフより、6.1 節で提案した指数関数を用いた外挿手法は従来の単純な外挿手法や PF と比べると優れた推定精度を示しているものの、 $\alpha = 0.2$ の場合と比較するとその精度は相対的に悪くなっており、また Cramér-Rao の下界式より求めた推定精度の限界値と比較すると、その誤差が更に大きくなっていることが分かる。以上の結果より、EE は従来の 0 次ホールドや線形外挿などの外挿式と比べると優れた推定精度を示すものの、Cramér-Rao の下界式より求めた理論的な限界値との間にはまだ隔たりがあり、推定誤差を更に改善する余地があると思われる。

次に、図 6.3 及び図 6.4 に通信損失確率 $\alpha = 0.2$ 及び $\alpha = 0.5$ の時の時間ステップに対する EI4 値 ((4.4) 式) の変化の様子を示す。2 つのグラフより、EI4 の値は時間ステップの増加に伴いその値が 10.666 に近付いており、このことからロボットが目標とする正三角形格子に収束していくことが分かる。

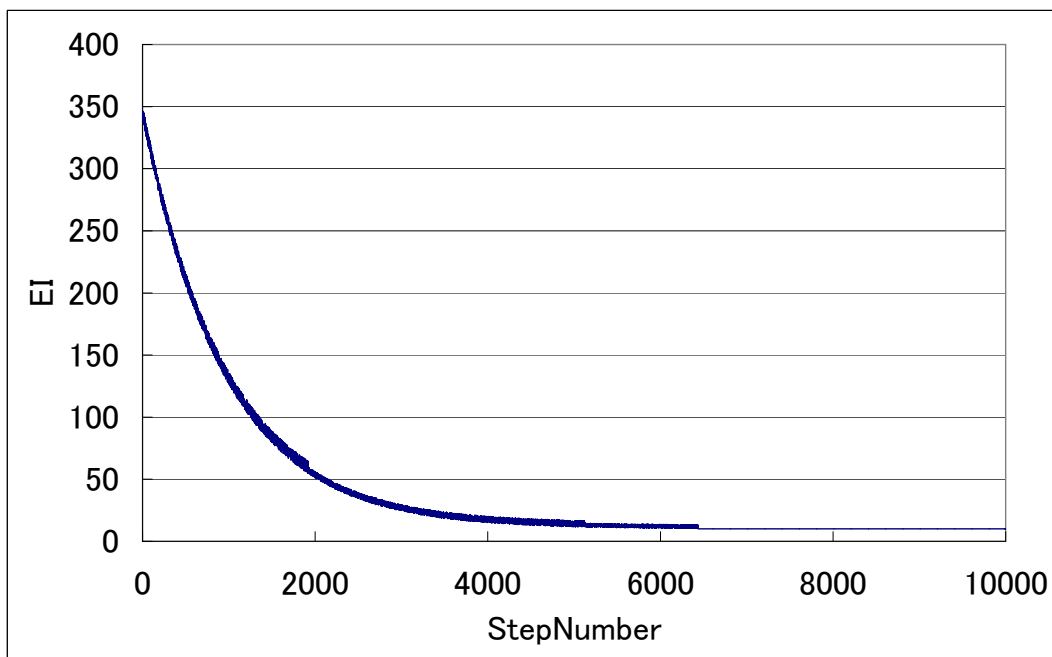


図 6.3: EE に対する EI4 値の変化 ($\alpha=0.2$)

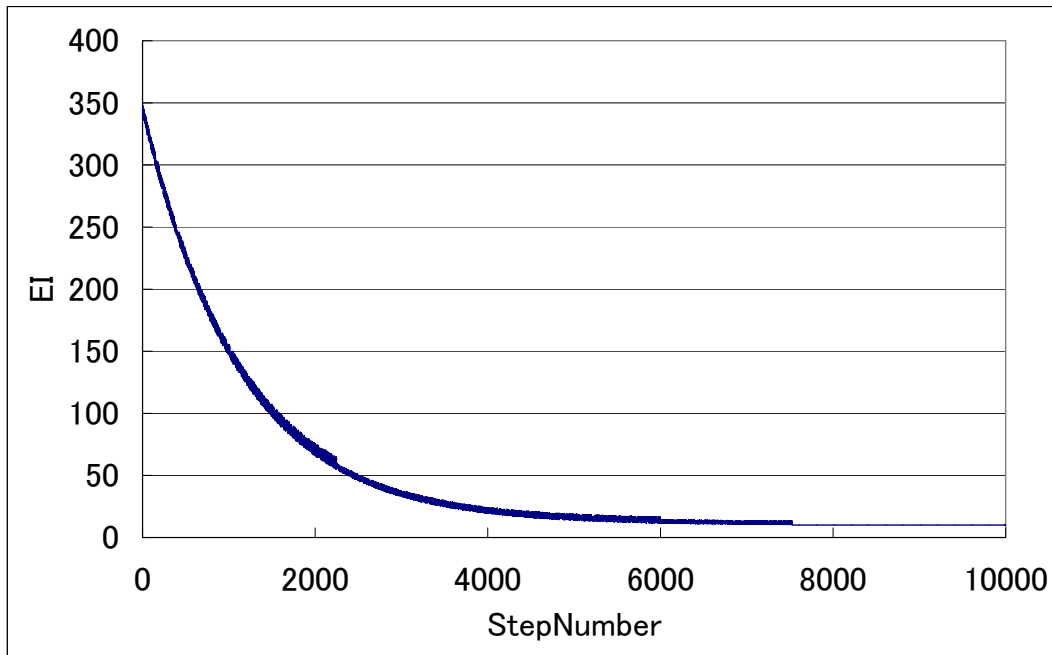


図 6.4: EE に対する EI4 値の変化 ($\alpha=0.5$)

第 7 章

指数関数を用いたロボット位置の外挿 手法の改良

前章で導入した指数関数を用いた新たな外挿式を使用することにより、従来の単純な外挿手法と変わらない低い計算量でその推定精度を向上させることに成功した。しかし、一方でその推定精度は Cramér-Rao の下界式により求められた推定精度の理論的な限界との間にはまだ開きがあり、その精度を更に改善する余地があるものと思われる。よって、本章においては前章で提案した指数関数を用いた外挿式に更に改良を加えることにより、その推定精度をより改善することを目指す。

7.1 指数関数を用いた外挿式の改良

前章で述べた指数関数を用いた外挿式を使用することにより、群ロボットにおける通信損失時のロボット位置推定精度を従来の手法よりも改善することができた。本節ではこの指数関数外挿式 (EE) に更に改良を加えることにより、推定精度を更に向上させることを目指す。前章で提案された指数関数を用いた外挿式においては、式の特性上外挿値が直前の変化率と同じ方向に推定されることになる。しかし、現実のロボットの動きを考えた場合、ロボットの座標は常に一方向に減少（または増加）しているのではなく、増減を繰り返しながら目標位置に向かって収束しているものと考えられる。この点を考慮して、本節においては前章で提案

した指数関数を用いた外挿式を改良した拡張型指数外挿 (Extended Exponential Extrapolation:EEE) を導入し、推定精度を向上させることを目指す。

EEE においては、前章で導入した EE における外挿式 ((6.5) 式及び (6.14) 式) を用いて計算した値を、その直前の変化率と逆方向に与えることによりその推定精度を高めることを試みる。具体的には、直前の値の変化率が正の場合には EE により求めた外挿値をその変化量分負の方向に、直前の値の変化率が負の場合には EE で求めた外挿値をその変化量分正の方向に与えることで現在の値を外挿する。

EEE における具体的な外挿手順は、以下に示す通りである。なお、6.1 節と同様、 $N_i(t)$ の x 要素である $x_i(t)$ についてのみ議論を行う。 y 要素 $y_i(t)$ についても同様の議論を行うことで外挿できる。

- Case-1: $x_i(t-2) > x_i(t-1)$

EEE の概念より、 $x_i(t)$ は以下の式を用いて外挿される。

$$\begin{aligned} x_i(t) &= x_i(t-1) + (x_i(t-1) - x'_i(t)) \\ &= \left[2 - \frac{x_i(t-1)}{x_i(t-2)} \cdot x_i(t-1) \right] \end{aligned} \quad (7.1)$$

但し、(6.5) 式より $x'_i(t)$ は以下の式で表される。

$$x'_i(t) = \frac{x_i(t-1)}{x_i(t-2)} \cdot x_i(t-1) \quad (7.2)$$

- Case-2: $x_i(t-2) < x_i(t-1)$

この場合は Case-1 の場合とは逆に、 $x_i(t)$ を以下の式を用いて外挿する。

$$\begin{aligned} x_i(t) &= x_i(t-1) - (x'_i(t) - x_i(t-1)) \\ &= 2x_i(t-1) + \left[\left[1 - \frac{x_i(t-1)}{x_i(t-2)} \right]^3 - 1 \right] \cdot \frac{(x_i(t-1))^2}{x_i(t-2)} \end{aligned} \quad (7.3)$$

但し、(6.14) 式より、 $x'_i(t)$ は以下の式で与えられる。

$$x'_i(t) = \frac{[x_i(t-2)]^2}{x_i(t-1)} \cdot \left[1 - \left[1 - \frac{x_i(t-1)}{x_i(t-2)} \right]^3 \right] \quad (7.4)$$

なお、ロボット r_i が時刻 T でサンプリングを行っている場合は、6.1 節と同様に扱うものとする。

7.2 EEE を用いたロボット位置推定のシミュレーション結果

本節においては、EEE の有効性について実際にロボットの動きをコンピュータ上でシミュレートすることにより考察する。シミュレーション条件は6.5 節と同様であり、通信損失によりロボットが得られなかった情報は7.1 節で示した手法を用いて外挿するものとする。6.5 節と同条件でのシミュレーションを10 回行い、その時のロボット位置の相対誤差を平均したものをグラフに示す。なお、相対誤差についての評価指標としては、5.3 節で示した式 (5.13) を用いるものとする。

また、ロボットの目標形状への収束度を測るための評価指標としては6.5 節と同様、4.4 節で提案した評価指標 (EI4) を用いて計算を行った。4.5 節での計算結果より、ロボット台数 $n = 10$ の場合には評価指標 EI4 の値が10.666 となった時に各ロボットにより完全な正三角形格子が形成されたと言える。

図 7.1 及び図 7.2 に、通信損失確率 α の値が0.2 及び0.5 の時の時間ステップ $N = 210$ までのシミュレーション結果を示す。各グラフでは横軸が時間ステップ、縦軸が各外挿手法を適用した場合の相対誤差の評価指標 l_α の値を表している。なお、比較のために単純な外挿手法として代表的な0 次ホールド (ZOH) と計算時間が指数外挿と同程度になるように粒子数を減らした PF による推定を行った結果、6.1 節で提案した EE による推定結果、及び理論的な推定限界として6.4 節で示した Cramér-Rao の下界式を用いて求めた限界値についても同時に掲載した。

図 7.1 は通信損失確率 $\alpha = 0.2$ の場合の l_α の変化を表したグラフである。図中で赤線が0 次ホールド、青線が EE、緑線が EEE、紫線が PF、茶線が Cramér-Rao の下界式を用いて求めた推定誤差の理論的な限界値 (Bound) を表している。グラフから分かる通り、7.1 節で提案した指数関数を用いた外挿手法は6.1 節で提案した手法と比べてその推定精度の点で大きな変化はなく、従来の単純な外挿手法や PF と比べると良好な推定結果を示している一方で、Cramér-Rao の下界式より求

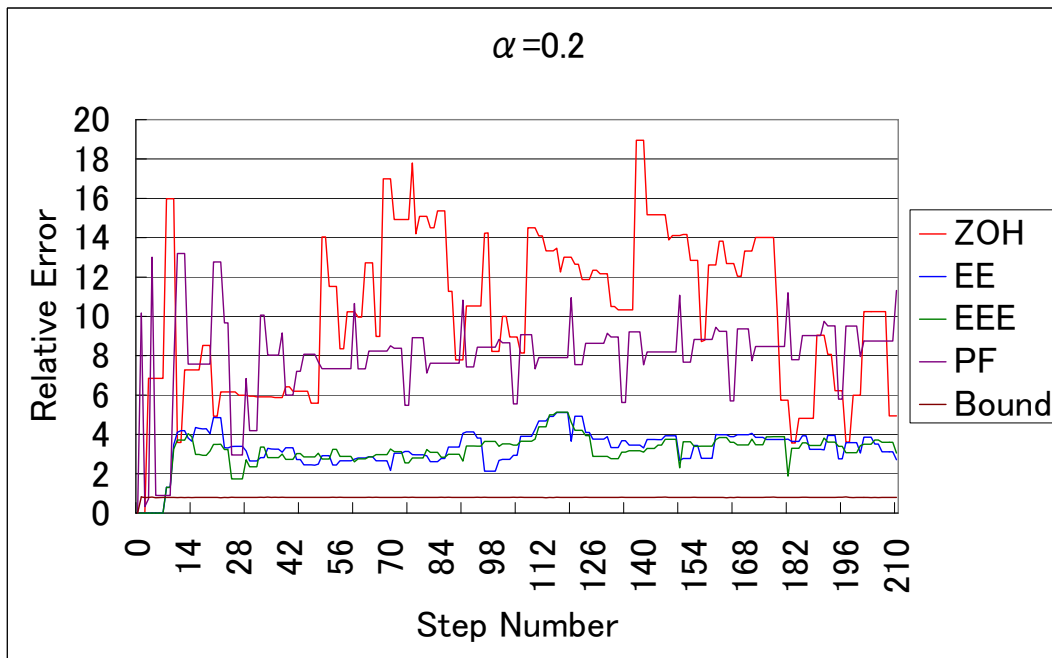


図 7.1: 通信損失確率 $\alpha=0.2$ の時のロボット位置の相対誤差の変化 (EEE)

めた推定精度の限界値と比べると、まだその間に開きがあることが分かる。

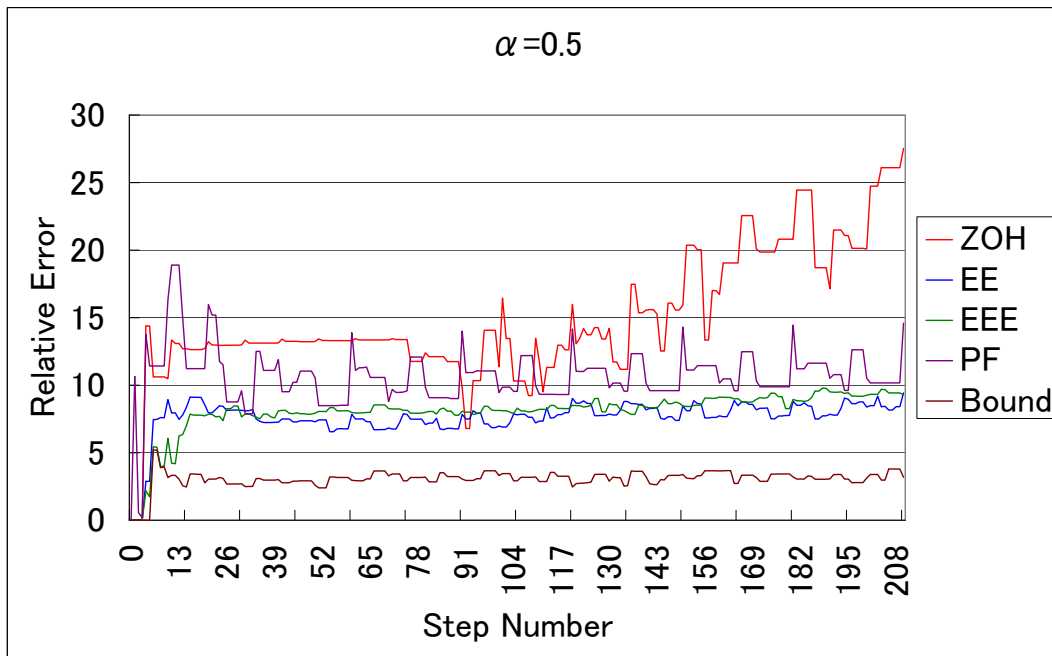


図 7.2: 通信損失確率 $\alpha=0.5$ の時のロボット位置の相対誤差の変化 (EEE)

続いて、図 7.2 に通信損失確率 $\alpha = 0.5$ の場合の l_α の変化を表したグラフを示

す。グラフ中の凡例は図 7.1 と同様である。グラフから、7.1 節で提案した指数関数を用いた外挿手法は通信損失確率 α の値を大きくしても 6.1 節で提案した手法と推定精度の面で大きな違いはなく、従来の単純な外挿手法や PF と比べると優れた推定精度を示しているものの、 $\alpha = 0.2$ の場合と比べるとその精度は悪化しており、また Cramér-Rao の下界式より求めた推定精度の限界値と比較すると、その差が更に拡大されていることが分かる。以上の結果より、EEE は従来の 0 次ホールドや線形外挿といった外挿式と比較すると良好な推定精度を示すものの、Cramér-Rao の下界式より求めた理論的な限界と比べるとまだ隔たりがあり、改善の余地があると思われる。

次に、図 7.3 及び図 7.4 に通信損失確率 $\alpha = 0.2$ 及び $\alpha = 0.5$ の時の時間ステップに対する EI4 値 ((4.4) 式) の変化の様子を示す。2 つのグラフより、EI4 の値は時間ステップの増加に伴いその値が 10.666 に近付いており、このことからロボットが目標とする正三角形格子に収束していくことが分かる。また、その収束時間については 6.1 節で提案した手法と大きな違いはなかった。

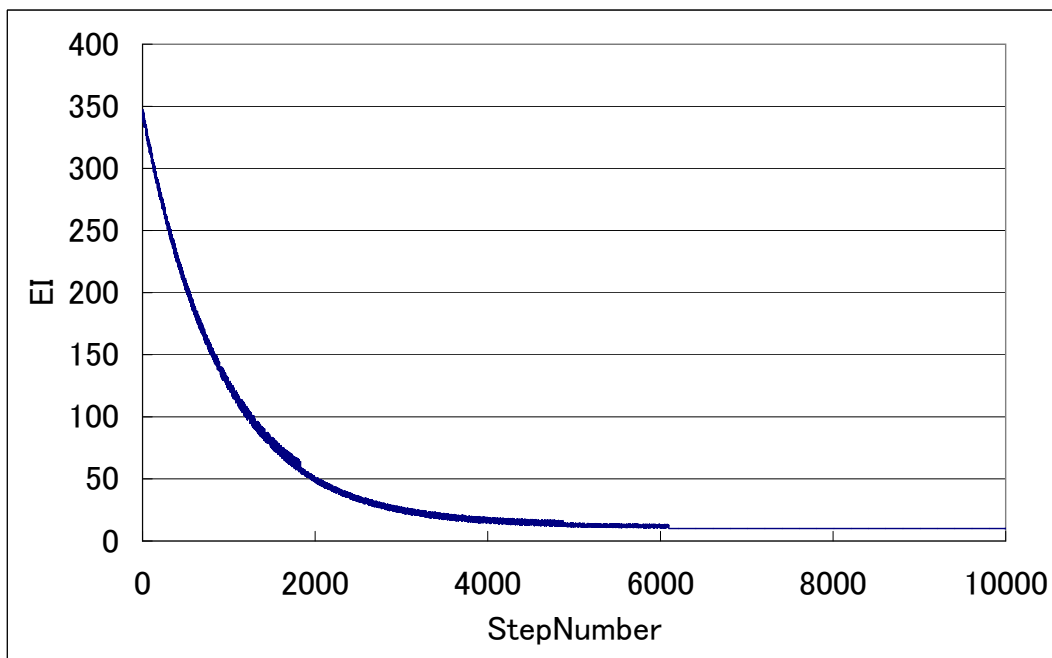


図 7.3: EEE に対する EI4 値の変化 ($\alpha=0.2$)

以上の結果より、EEE によって推定精度が改善されたとは言い難く、推定精度の向上のためには外挿式を更に改良する必要があると考えられる。

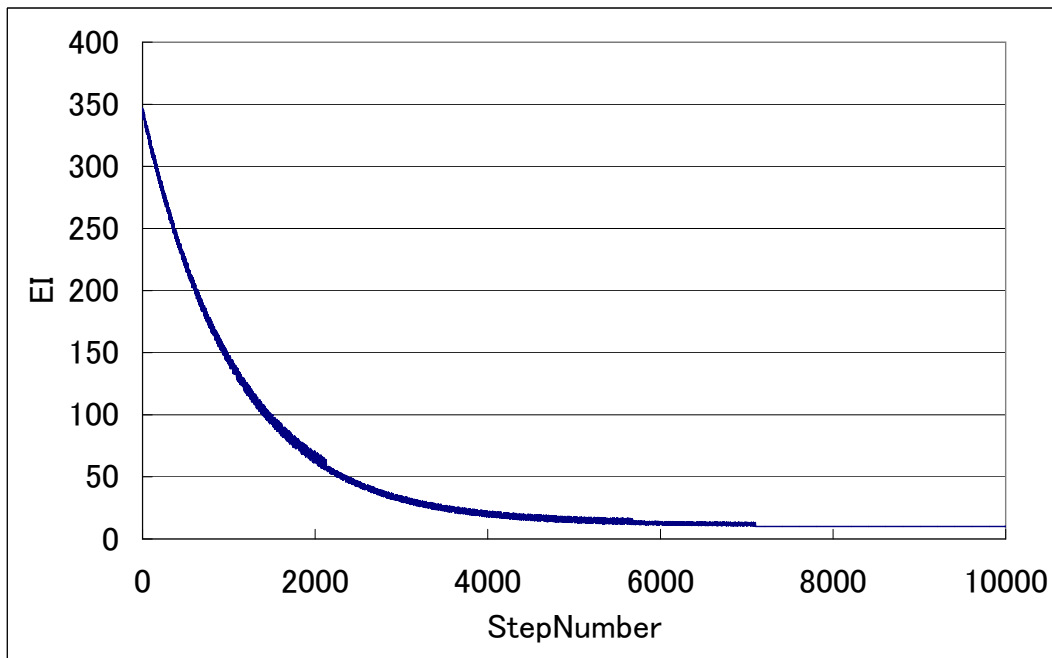


図 7.4: EEE に対する EI4 値の変化 ($\alpha=0.5$)

7.3 線形関数を加えた指数関数を用いた外挿式

これまで、外挿式として指数関数のみを取り扱ってきたが、この手法の根本的な問題点として、直前の値の変化率が大きい場合には値の変化に対して鋭敏に反応することができる一方、変化率が小さい場合にはその反応が鈍くなってしまうという点が挙げられる。これは、指数関数の持つ、微分値が大きい領域においてはその前後で値が大きく変化するのに対して、微分値が小さい領域においては値がほとんど変化しないという特性に起因するものと思われる。よって、直前の値の変化率が小さい場合においても変化に対して鋭敏に反応することを目的として、本節においては、前章で示した指数関数を用いた外挿式に対し、新たに 1 次関数を加えた外挿式（1 次多項式を用いた指数外挿（Exponential Extrapolation with First-degree polynomial function : EEF）及びその拡張型（拡張型 1 次多項式を用いた指数外挿（Extended Exponential Extrapolation with First-degree polynomial function, EEEF））を提案する。具体的には、6.1 節及び 7.1 節で提案した指数関数を用いた外挿式（EE 及び EEE）に対して更に線形項を加えることにより、その推定精度を更に向上させることを目指す。

1次多項式を用いた指数外挿式 (EEF) の概要は以下の通りである。時刻 t における状態量 $N_i(t)$ が通信損失により観測できなかった場合、その直前に観測された4つの状態量 $N_i(t-1)$ 、 $N_i(t-2)$ 、 $N_i(t-3)$ 、 $N_i(t-4)$ を以下の外挿式に代入することにより、現在の状態 $N_i(t)$ を外挿するものとする。またここでも前章と同様、 $N_i(t)$ のうち x 要素 $x_i(t)$ についてのみ議論を行うものとする。

EEF においては、以下の式を外挿式として用いる。

$$x_i(t) = A_i e^{-B_i t} + C_i t + D_i \quad (7.5)$$

ここで、 A_i 、 C_i 、及び D_i は定数であり、また B_i は正であるとする。通信損失が発生した場合、ロボット r_i は $x_i(t)$ を時刻 $t-1$ 、 $t-2$ 、 $t-3$ 、 $t-4$ の4つの情報を用いて外挿するものとする。時刻 $t-1$ 、 $t-2$ 、 $t-3$ 、 $t-4$ における値を (7.5) 式に代入することにより、以下の式を得ることができる。

$$x_i(t-1) = A_i e^{-B_i(t-1)} + C_i(t-1) + D_i \quad (7.6)$$

$$x_i(t-2) = A_i e^{-B_i(t-2)} + C_i(t-2) + D_i \quad (7.7)$$

$$x_i(t-3) = A_i e^{-B_i(t-3)} + C_i(t-3) + D_i \quad (7.8)$$

$$x_i(t-4) = A_i e^{-B_i(t-4)} + C_i(t-4) + D_i \quad (7.9)$$

(7.6) 式から (7.8) 式を用いることにより、以下の値が得られる。

$$b_1 = x_i(t-1) - 2x_i(t-2) + x_i(t-3) = A_i e^{-B_i(t+1)}(e^{-2B_i} - 2e^{-B_i} + 1) = A_i e^{-B_i(t+1)}(e^{-B_i} - 1)^2 \quad (7.10)$$

同様に、(7.7) 式から (7.9) 式を用いることにより、以下の値が得られる。

$$b_2 = x_i(t-2) - 2x_i(t-3) + x_i(t-4) = A_i e^{-B_i t}(e^{-2B_i} - 2e^{-B_i} + 1) = A_i e^{-B_i t}(e^{-B_i} - 1)^2 \quad (7.11)$$

(7.10) 式及び (7.11) 式より、以下の関係式が求められる。

$$b_1 = A_i e^{-B_i t}(e^{-B_i} - 1)^2 = b_2 e^{-B_i} \quad (7.12)$$

$$\therefore e^{-B_i} = \frac{b_1}{b_2}$$

上式を (7.10) 式に代入することにより、以下の値が求まる。

$$A_i = \frac{b_1}{e^{-B_i t}(e^{-B_i} - 1)^2} = \frac{b_2^{(t-1)}}{b_3^2 b_1^{(t-4)}}$$

但し、

$$b_3 = x_i(t-1) - 3x_i(t-2) + 3x_i(t-3) - x_i(t-4) \quad (7.13)$$

である。ここで、(7.8) 式 – (7.9) 式より、

$$x_i(t-3) - x_i(t-4) = A_i e^{-B_i t}(e^{-B_i} - 1) + C_i \quad (7.14)$$

(7.14) 式より、以下の値が求まる。

$$C_i = x_i(t-3) - x_i(t-4) - A_i e^{-B_i t}(e^{-B_i} - 1) = \frac{b_2^2}{b_3}$$

同様に、(7.9) 式より、以下の値が求まる。

$$D_i = x_i(t-4) - A_i e^{-B_i(t-4)} - C_i(t-4)$$

以上をまとめると、次の通りとなる。

$$A_i = \frac{b_2^{t-1}}{(b_3)^2 (b_1)^{t-4}} \quad (7.15)$$

$$e^{-B_i} = \frac{b_1}{b_2} \quad (7.16)$$

$$C_i = x_i(t-3) - x_i(t-4) - \frac{(b_2)^2}{b_3} \quad (7.17)$$

$$D_i = x_i(t-4) - A_i e^{-B_i(t-4)} - C_i(t-4) \quad (7.18)$$

但し、

$$b_1 = x_i(t-1) - 2x_i(t-2) + x_i(t-3)$$

$$b_2 = x_i(t-2) - 2x_i(t-3) + x_i(t-4)$$

$$b_3 = x_i(t-1) - 3x_i(t-2) + 3x_i(t-3) - x_i(t-4)$$

である。よって、(7.15)式～(7.18)式を(7.5)式に代入することにより、 $x_i(t)$ の値を直前の4つの状態より外挿することができる。

次に、7.1節と同様に、拡張型1次多項式を用いた指数外挿(Extended Exponential Extrapolation with First-degree polynomial function、EEEEF)を次のように定義する。7.1節で述べた考え方をEEFに適用することで、EEEEFの外挿式を得ることができる。具体的には、7.1節の(7.1)式及び(7.3)式において代入する $x'_i(t)$ の値として(7.5)式を用いることで、EEEEFによる外挿式が得られる。なお、ロボット r_i のが時刻 T でサンプリングを行っている場合の扱いは、6.1節と同様である。

7.4 EEF及びEEEEFによるロボット位置推定のシミュレーション結果

本節においては、通信損失時にEEF及びEEEEFを用いてロボット位置を外挿した場合の推定精度について、実際にロボットの動きをコンピュータ上でシミュレートすることにより、定量的な評価を行う。シミュレーション条件及び推定誤差と収束度についての評価指標は6.5節と同様である。同条件でのシミュレーションを10回行い、その平均値をグラフに示す。なお、これまでの手法との性能を比較するため、EE及びEEEの結果についても再掲した。

図7.5は通信損失確率 $\alpha = 0.2$ の場合の l_α の変化を表したグラフである。図中で青線がEE、緑線がEEE、赤線がEEF、紫線がEEEEF、茶線がCramér-Raoの下界式を用いて求めた推定誤差の理論的な限界値(Bound)を表している。グラフより、7.3節で提案したEEF及びEEEEFは、Cramér-Rao式から求めた理論的な限界値に比べると推定精度が劣っているものの、良好な推定精度を示していることが分かる。このことと図5.1で示したZOH及びPFによる推定結果を合わせて考えることにより、EEF及びEEEEFが計算量及び推定精度の両面で優れていることを示している。

続いて、図7.6に通信損失確率 $\alpha = 0.5$ の場合の l_α の変化を表したグラフを示す。グラフ中の凡例は図7.5と同様である。グラフから、EEF及びEEEEFの推定

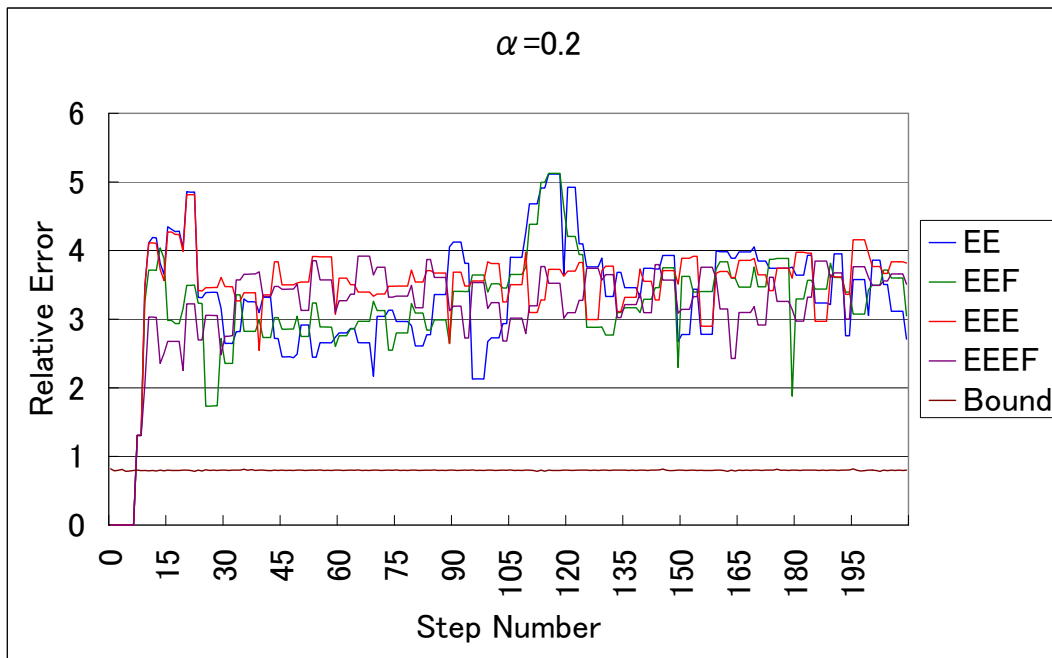


図 7.5: 通信損失確率 $\alpha=0.2$ の時のロボット位置の相対誤差の変化 (EEF 及び EEEF)

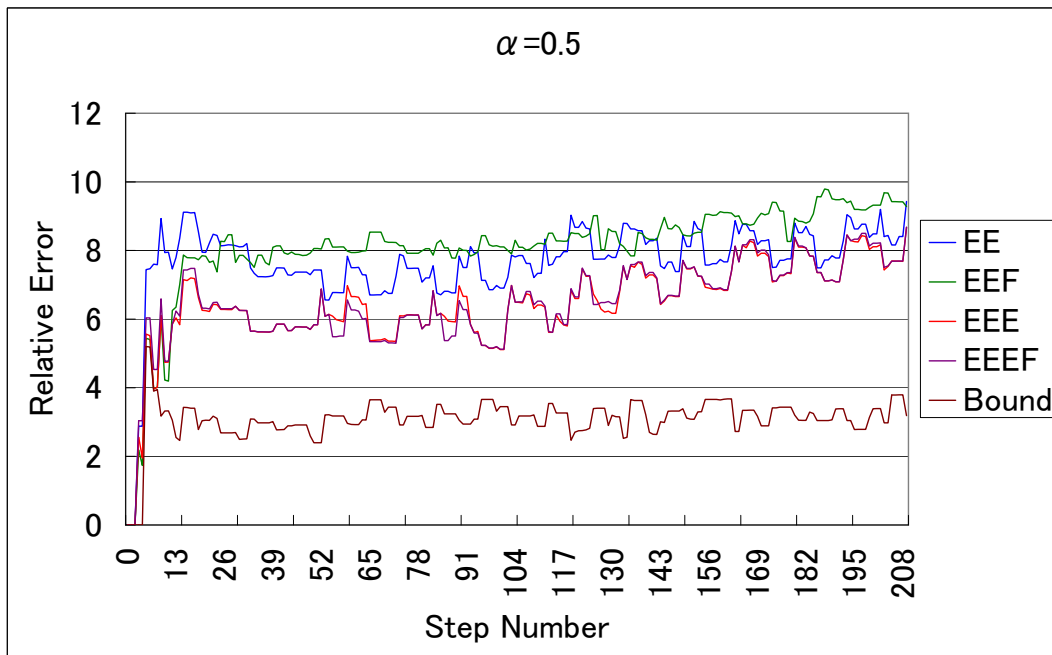


図 7.6: 通信損失確率 $\alpha=0.5$ の時のロボット位置の相対誤差の変化 (EEF 及び EEEF)

精度は Cramér-Rao の下界式より求めた理論的な限界値と比べると依然とした悪いものの、EE や EEE と比べると改善されていることが分かる。以上の結果と図

5.5 で示した ZOH 及び PF による推定結果を合わせて考えることにより、EEF 又は EEEF は通信損失が存在する場合の群ロボットにおける位置推定手法として、計算量の面からも推定精度の面からも優れていると結論付けることができる。

次に、図 7.7～図 7.10 に通信損失確率 $\alpha = 0.2$ 及び $\alpha = 0.5$ の時の EEE 及び EEEF の時間ステップに対する EI4 値 ((4.4) 式) の変化の様子を示す。2つのグラフより、EEF、EEEF のいずれの場合も EI の値は時間ステップの増加に伴いその値が 10.666 に近付いており、このことからロボットが目標とする正三角形格子に収束していくことが分かる。また、EI4 の値が 10.666 に収束するまでの時間ステップは、EE や EEE の場合と比べて短くなっていることが図 6.3、図 6.4、図 7.3、図 7.4 との比較から分かる。このことから、EEF や EEEF は EE や EEE と比べて優れていることが分かる。

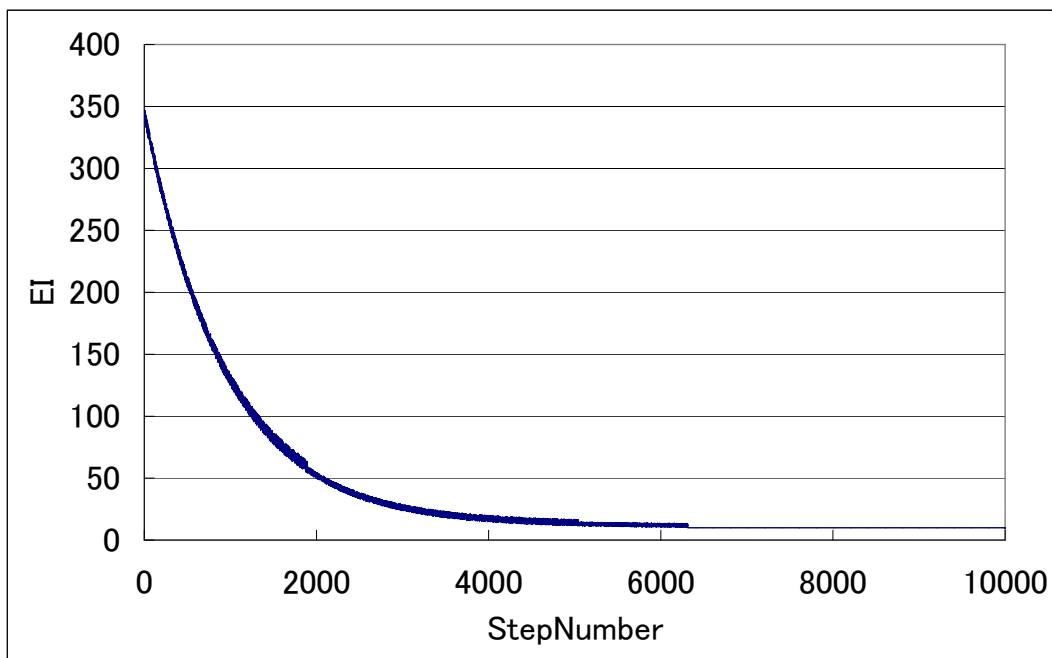


図 7.7: EEF に対する EI4 値の変化 ($\alpha=0.2$)

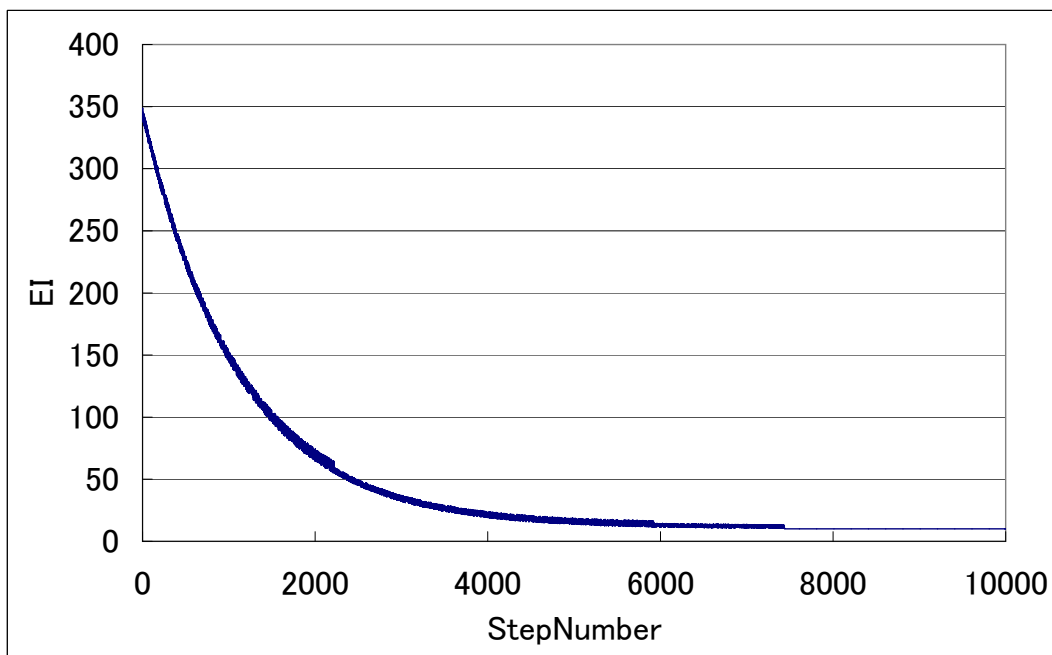


図 7.8: EEF に対する EI4 値の変化 ($\alpha=0.5$)

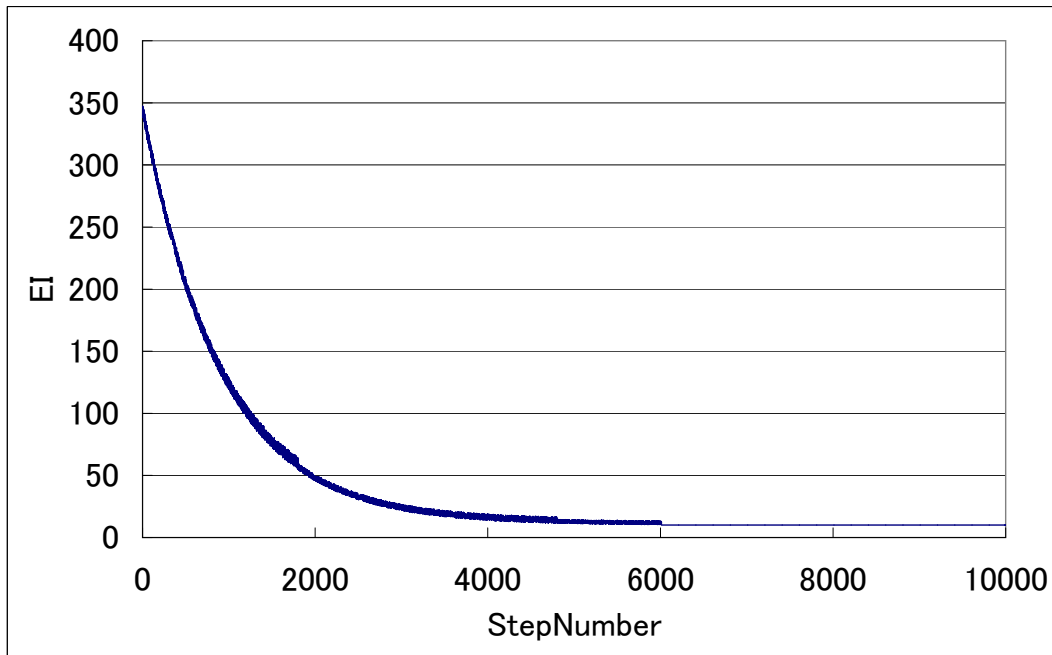


図 7.9: EEEF に対する EI4 値の変化 ($\alpha=0.2$)

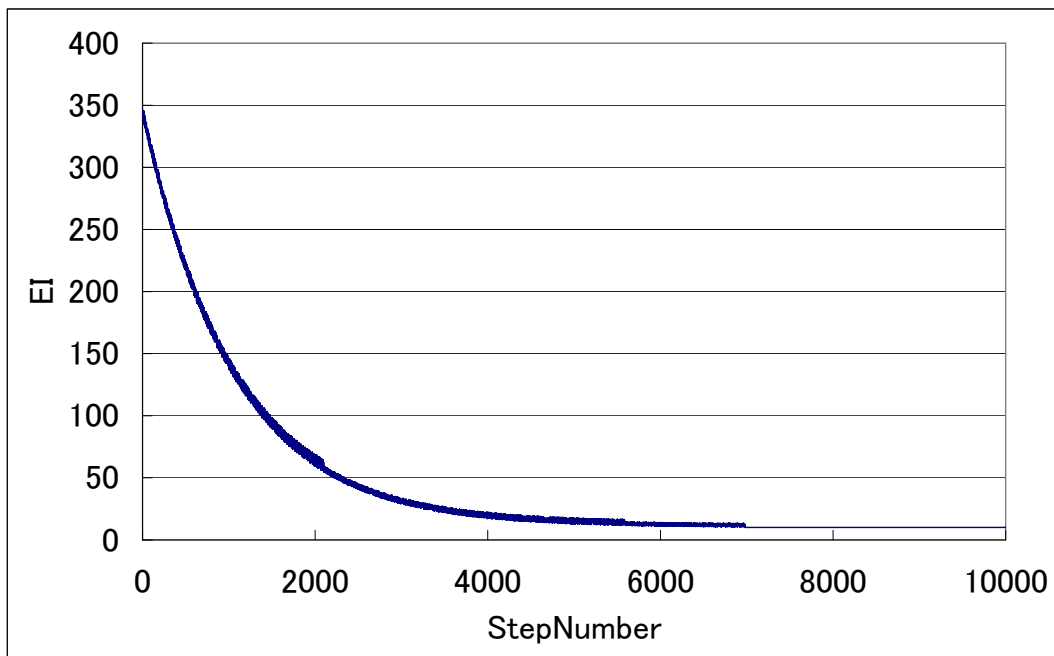


図 7.10: EEEF に対する EI4 値の変化 ($\alpha=0.5$)

第 8 章

高階の微分方程式で記述されるシステムへの指数外挿の応用

前章までにおいては、通信損失時の外挿手法として指数関数を用いた指数外挿式及びその改良式を提案し、その有効性について検証を行ってきた。しかし、これまでの議論においてはロボットが 1 階の微分方程式に従うことを仮定しているため、その適用範囲が限定されていると言える。よって、本章においてはロボットの動きが高階の線形微分方程式で記述される場合について、指数外挿法を適用した場合の有効性について数学的な検証を行う。

8.1 指数外挿式 (EE) の 2 階の線形微分方程式に対する適用

本節においては、6.1 節で導入した指数外挿式を、ロボットの動作が 2 階の線形微分方程式で表される場合に適用した時の安定性について調べる。

本節では、ロボットの動作は以下の 2 階の線形微分方程式に従うものとする。

$$\ddot{x}_i(t) = a\dot{x}_i(t) + bx_i(t) + c \quad (8.1)$$

上記の微分方程式の解は、以下の式で与えられる。

- Case-1: (8.1) 式が 2 つの実解 $-\lambda_1$ 、 $-\lambda_2$ を持つ場合

$$x_i(t) = A_1 e^{-\lambda_1 t} + A_2 e^{-\lambda_2 t} + B \quad (8.2)$$

- Case-2: (8.1) 式が 2 つの複素解 $-\lambda_1 \pm i\lambda_2$ を持つ場合

$$x_i(t) = A_1 e^{-\lambda_1 t} \sin(\lambda_2 t) + A_2 e^{-\lambda_1 t} \cos(\lambda_2 t) + B \quad (8.3)$$

- Case-3: (8.1) 式が 1 つの重解 $-\lambda_1$ を持つ場合

$$x_i(t) = A_1 e^{-\lambda_1 t} + t A_2 e^{-\lambda_1 t} + B \quad (8.4)$$

但し、ロボットは最終的に目標形状に収束するため、 $x_i(t)$ は一定値に収束することから、解が $t \rightarrow \infty$ で無限大に発散する Case-3 については今回の議論からは除外する。同様の理由により、 A_1 、 A_2 、 λ_1 、 λ_2 、 B はそれぞれ正の定数である。

上記の式に従って行動するロボット間の通信に損失が発生した場合に、6.1 節で述べた指数外挿式を用いて現在の状態を外挿することを考える。

- Case-1: (8.1) 式が 2 つの実解を持つ場合

この時、(8.1) 式の解は (8.2) 式で表される。ここで、(8.2) 式は減少関数であることから、通信損失時には以下の (6.1) 式を外挿式として用いて失われた情報を外挿する。

$$x_i(t) = A_i e^{-B_i t}$$

この時、(6.5) 式より $x_i(t)$ は $x_i(t-1)$ 及び $x_i(t-2)$ を用いて以下のように外挿される。

$$x_i(t) = A_i e^{-B_i t} = A_i e^{-B_i(t-1)} \cdot e^{-B_i} = \frac{x_i(t-1)}{x_i(t-2)} \cdot x_i(t-1)$$

ここで、(8.2) 式より $x_i(t-1)$ 及び $x_i(t-2)$ は実際には以下の式で与えられる。

$$x_i(t-1) = A_1 e^{-\lambda_1(t-1)} + A_2 e^{-\lambda_2(t-1)} + B \quad (8.5)$$

$$x_i(t-2) = A_1 e^{-\lambda_1(t-2)} + A_2 e^{-\lambda_2(t-2)} + B \quad (8.6)$$

ここで、安定性を判別するためのリアプノフ関数を以下のように定義する。

$$f_i = \frac{1}{2}(x_i(t) - B)^2 \quad (8.7)$$

この式が正定であることは明らかである。ここで、 f_i の微分は、以下のように与えられる。

$$\dot{f}_i = \dot{x}_i(t)(x_i(t) - B) \quad (8.8)$$

ここで、 $x_i(t) = \frac{x_i(t-1)}{x_i(t-2)} \cdot x_i(t-1)$ であることから、

$$\dot{x}_i(t) = \frac{2\dot{x}_i(t-1)x_i(t-1)x_i(t-2) - \{x_i(t-1)\}^2 \dot{x}_i(t-2)}{\{x_i(t-2)\}^2} \quad (8.9)$$

と計算される。上式に (8.5) 式及び (8.6) 式を代入することにより、 $\dot{x}_i(t) < 0$ が示される。また、 $x_i(t) - B > 0$ であることから、(8.8) 式は負定となる。よって、リアプノフの安定性定理より、 $x_i(t)$ は一定値に収束する。

- Case-2: (8.1) 式が 2 つの複素解を持つ場合

この時、次の 2 つの場合に分けて考察を行う必要がある。

- Case-a: $x_i(t-2) > x_i(t-1)$ の時

この場合、通信損失時には以下の (6.1) 式を外挿式として用いて失われた情報を外挿する。

$$x_i(t) = A_i e^{-B_i t}$$

この時、(6.5) 式より $x_i(t)$ は $x_i(t-1)$ 及び $x_i(t-2)$ を用いて以下のように外挿される。

$$x_i(t) = A_i e^{-B_i t} = A_i e^{-B_i(t-1)} \cdot e^{-B_i} = \frac{x_i(t-1)}{x_i(t-2)} \cdot x_i(t-1)$$

ここで、(8.3) 式より $x_i(t-1)$ 及び $x_i(t-2)$ は実際には以下の式で与えられる。

$$x_i(t-1) = A_1 e^{-\lambda_1(t-1)} \sin(\lambda_2(t-1)) + A_2 e^{-\lambda_1(t-1)} \cos(\lambda_2(t-1)) + B \quad (8.10)$$

$$x_i(t-2) = A_1 e^{-\lambda_1(t-2)} \sin(\lambda_2(t-2)) + A_2 e^{-\lambda_1(t-2)} \cos(\lambda_2(t-2)) + B \quad (8.11)$$

ここで、安定性を判別するためのリアプノフ関数を以下のように定義する。

$$f_i = \frac{1}{2}(x_i(t) - B)^2 \quad (8.12)$$

後は Case-1 の場合と同様の議論を行うことにより、 f_i の正定性及び \dot{f}_i の負定性から、リアプノフの安定性理論より $x_i(t)$ が一定値に収束することが示される。

- Case-b: $x_i(t-2) < x_i(t-1)$ の時

この場合、通信損失時には以下の (6.6) 式を外挿式として用いて失われた情報を外挿する。

$$x_i(t) = A_i(1 - e^{-B_i t})$$

この時、(6.14) 式より $x_i(t)$ は $x_i(t-1)$ 及び $x_i(t-2)$ を用いて以下の式で外挿される。

$$x_i(t) = \frac{[x_i(t-2)]^2}{x_i(t-1)} \cdot \left[1 - \left[1 - \frac{x_i(t-1)}{x_i(t-2)} \right]^3 \right]$$

ここで、 $\alpha = \frac{x_i(t-1)}{x_i(t-2)}$ 、 $\beta = x_i(t-2)$ と置くと、上式は以下のように変形することができる。

$$x_i(t) = \beta(3 - 3\alpha + \alpha^2) \quad (8.13)$$

ここで、(8.3) 式より $x_i(t-1)$ 及び $x_i(t-2)$ は実際には以下の式で与えられる。

$$x_i(t-1) = A_1 e^{-\lambda_1(t-1)} \sin(\lambda_2(t-1)) + A_2 e^{-\lambda_1(t-1)} \cos(\lambda_2(t-1)) + B \quad (8.14)$$

$$x_i(t-2) = A_1 e^{-\lambda_1(t-2)} \sin(\lambda_2(t-2)) + A_2 e^{-\lambda_1(t-2)} \cos(\lambda_2(t-2)) + B \quad (8.15)$$

ここで、安定性を判別するためのリアプノフ関数を以下のように定義する。

$$f_i = \frac{1}{2}(x_i(t) - B)^2 \quad (8.16)$$

この式が正定であることは明らかである。ここで、 f_i の微分は、以下のように与えられる。

$$\dot{f}_i = \dot{x}_i(t)(x_i(t) - B) \quad (8.17)$$

上式の正負は、 $\dot{x}_i(t)$ の符号により決定される。ここで、(8.13) 式より、 $\dot{x}_i(t)$ を計算すると、

$$\dot{x}_i(t) = \dot{\beta}(3 - 3\alpha + \alpha^2) + \beta(2\dot{\alpha}\alpha - 3\dot{\alpha}) \quad (8.18)$$

この式に α 及び β の値を代入することにより、 \dot{f}_i の負定性を示すことができる。よって、リアプノフの安定性理論より、 $x_i(t)$ は一定値に収束する。

8.2 指数外挿式 (EE) の n 階の線形微分方程式に対する適用

本節においては、6.1 節で導入した指数外挿式を、ロボットの動作が n 階の線形微分方程式で表される場合に適用した時の安定性について検証する。

本節では、ロボットの動作は以下の n 階の線形微分方程式に従うものとする。

$$x_i^{(n)}(t) = a_{n-1}x_i^{(n-1)}(t) + a_{n-2}x_i^{(n-2)}(t) + \cdots + a_0 \quad (8.19)$$

上記の微分方程式の解は、以下の式で与えられる。

- Case-1: (8.19) 式が n 個の実解 $-\lambda_1, -\lambda_2, \dots, -\lambda_n$ を持つ場合

$$x_i(t) = A_1 e^{-\lambda_1 t} + A_2 e^{-\lambda_2 t} + \dots + A_n e^{-\lambda_n t} + B \quad (8.20)$$

- Case-2: (8.19) 式が $2k$ 個の複素解 $-\lambda_1 \pm i\lambda_2, \dots, -\lambda_{2k-1} \pm i\lambda_{2k}$ と $n - 2k$ 個の実解 $-\lambda_{2k+1}, \dots, -\lambda_n$ を持つ場合

$$\begin{aligned} x_i(t) = & A_1 e^{-\lambda_1 t} \sin(\lambda_2 t) + A_2 e^{-\lambda_1 t} \cos(\lambda_2 t) + \dots \\ & + A_{2k-1} e^{-\lambda_{2k-1} t} \sin(\lambda_{2k} t) + A_{2k} e^{-\lambda_{2k-1} t} \cos(\lambda_{2k} t) \\ & + A_{2k+1} e^{-\lambda_{2k+1} t} + \dots + A_n e^{-\lambda_n t} + B \end{aligned} \quad (8.21)$$

なお、(8.19) 式が重解を持つ場合については、 $x_i(t)$ が $t \rightarrow \infty$ で無限大に発散することから、ロボットが目標形状に収束するという仮定に反するため、今回は扱わない。同様の理由により、 $A_1, A_2, \dots, A_n > 0$ 、 $\lambda_1, \lambda_2, \dots, \lambda_n > 0$ 、 $B > 0$ である。

- Case-1: (8.19) 式が n 個の実解を持つ場合

この時、(8.19) 式の解は (8.20) 式で表される。ここで、(8.20) 式は減少関数であることから、通信損失時には以下の (6.1) 式を外挿式として用いて失われた情報を外挿する。

$$x_i(t) = A_i e^{-B_i t}$$

この時、(6.5) 式より $x_i(t)$ は $x_i(t-1)$ 及び $x_i(t-2)$ を用いて以下のように外挿される。

$$x_i(t) = A_i e^{-B_i t} = A_i e^{-B_i(t-1)} \cdot e^{-B_i} = \frac{x_i(t-1)}{x_i(t-2)} \cdot x_i(t-1)$$

ここで、(8.20) 式より $x_i(t-1)$ 及び $x_i(t-2)$ は実際には以下の式で与えられる。

$$x_i(t-1) = A_1 e^{-\lambda_1(t-1)} + A_2 e^{-\lambda_2(t-1)} + \dots + A_n e^{-\lambda_n(t-1)} + B \quad (8.22)$$

$$x_i(t-2) = A_1 e^{-\lambda_1(t-2)} + A_2 e^{-\lambda_2(t-2)} + \dots + A_n e^{-\lambda_n(t-2)} + B \quad (8.23)$$

ここで、安定性を判別するためのリアプノフ関数を以下のように定義する。

$$f_i = \frac{1}{2}(x_i(t) - B)^2 \quad (8.24)$$

この式が正定であることは明らかである。ここで、 f_i の微分は、以下のように与えられる。

$$\dot{f}_i = \dot{x}_i(t)(x_i(t) - B) \quad (8.25)$$

ここで、 $x_i(t) = \frac{x_i(t-1)}{x_i(t-2)} \cdot x_i(t-1)$ であることから、

$$\dot{x}_i(t) = \frac{2\dot{x}_i(t-1)x_i(t-1)x_i(t-2) - \{x_i(t-1)\}^2\dot{x}_i(t-2)}{\{x_i(t-2)\}^2} \quad (8.26)$$

と計算される。上式に (8.22) 式及び (8.23) 式を代入することにより、 $\dot{x}_i(t) < 0$ が示される。また、 $x_i(t) - B > 0$ であることから、(8.25) 式は負定となる。よって、リアプノフの安定性定理より、 $x_i(t)$ は一定値に収束する。

- Case-2: (8.19) 式が $2k$ 個の複素解と $n - k$ 個の実解を持つ場合

この時、次の 2 つの場合に分けて考察を行う必要がある。

- Case-a: $x_i(t-2) > x_i(t-1)$ の時

この場合、通信損失時には以下の (6.1) 式を外挿式として用いて失われた情報を外挿する。

$$x_i(t) = A_i e^{-B_i t}$$

この時、(6.5) 式より $x_i(t)$ は $x_i(t-1)$ 及び $x_i(t-2)$ を用いて以下のように外挿される。

$$x_i(t) = A_i e^{-B_i t} = A_i e^{-B_i(t-1)} \cdot e^{-B_i} = \frac{x_i(t-1)}{x_i(t-2)} \cdot x_i(t-1)$$

ここで、(8.21) 式より $x_i(t-1)$ 及び $x_i(t-2)$ は実際には以下の式で与えられる。

$$\begin{aligned}
x_i(t-1) &= A_1 e^{-\lambda_1(t-1)} \sin(\lambda_2(t-1)) + A_2 e^{-\lambda_1(t-1)} \cos(\lambda_2(t-1)) + \dots \\
&+ A_{2k-1} e^{-\lambda_{2k-1}(t-1)} \sin(\lambda_{2k}(t-1)) + A_{2k} e^{-\lambda_{2k-1}(t-1)} \cos(\lambda_{2k}(t-1)) \\
&+ A_{2k+1} e^{-\lambda_{2k+1}(t-1)} + \dots + A_n e^{-\lambda_n(t-1)} + B
\end{aligned} \tag{8.27}$$

$$\begin{aligned}
x_i(t-2) &= A_1 e^{-\lambda_1(t-2)} \sin(\lambda_2(t-2)) + A_2 e^{-\lambda_1(t-2)} \cos(\lambda_2(t-2)) + \dots \\
&+ A_{2k-1} e^{-\lambda_{2k-1}(t-2)} \sin(\lambda_{2k}(t-2)) + A_{2k} e^{-\lambda_{2k-1}(t-2)} \cos(\lambda_{2k}(t-2)) \\
&+ A_{2k+1} e^{-\lambda_{2k+1}(t-2)} + \dots + A_n e^{-\lambda_n(t-2)} + B
\end{aligned} \tag{8.28}$$

ここで、安定性を判別するためのリアプノフ関数を以下のように定義する。

$$f_i = \frac{1}{2}(x_i(t) - B)^2 \tag{8.29}$$

後は Case-1 の場合と同様の議論を行うことにより、リアプノフの安定性理論より $x_i(t)$ が一定値に収束することが示される。

- Case-b: $x_i(t-2) < x_i(t-1)$ の時

この場合、通信損失時には以下の (6.6) 式を外挿式として用いて失われた情報を外挿する。

$$x_i(t) = A_i(1 - e^{-B_i t})$$

この時、(6.14) 式より $x_i(t)$ は $x_i(t-1)$ 及び $x_i(t-2)$ を用いて以下の式で外挿される。

$$x_i(t) = \frac{[x_i(t-2)]^2}{x_i(t-1)} \cdot \left[1 - \left[1 - \frac{x_i(t-1)}{x_i(t-2)} \right]^3 \right]$$

ここで、 $\alpha = \frac{x_i(t-1)}{x_i(t-2)}$ 、 $\beta = x_i(t-2)$ と置くと、上式は以下のように変形することができる。

$$x_i(t) = \beta(3 - 3\alpha + \alpha^2) \tag{8.30}$$

ここで、(8.21) 式より $x_i(t-1)$ 及び $x_i(t-2)$ は実際には以下の式で与えられる。

$$\begin{aligned}
 x_i(t-1) = & A_1 e^{-\lambda_1(t-1)} \sin(\lambda_2(t-1)) + A_2 e^{-\lambda_1(t-1)} \cos(\lambda_2(t-1)) + \dots \\
 & + A_{2k-1} e^{-\lambda_{2k-1}(t-1)} \sin(\lambda_{2k}(t-1)) + A_{2k} e^{-\lambda_{2k-1}(t-1)} \cos(\lambda_{2k}(t-1)) \\
 & + A_{2k+1} e^{-\lambda_{2k+1}(t-1)} + \dots + A_n e^{-\lambda_n(t-1)} + B
 \end{aligned} \tag{8.31}$$

$$\begin{aligned}
 x_i(t-2) = & A_1 e^{-\lambda_1(t-2)} \sin(\lambda_2(t-2)) + A_2 e^{-\lambda_1(t-2)} \cos(\lambda_2(t-2)) + \dots \\
 & + A_{2k-1} e^{-\lambda_{2k-1}(t-2)} \sin(\lambda_{2k}(t-2)) + A_{2k} e^{-\lambda_{2k-1}(t-2)} \cos(\lambda_{2k}(t-2)) \\
 & + A_{2k+1} e^{-\lambda_{2k+1}(t-2)} + \dots + A_n e^{-\lambda_n(t-2)} + B
 \end{aligned} \tag{8.32}$$

ここで、安定性を判別するためのリアプノフ関数を以下のように定義する。

$$f_i = \frac{1}{2}(x_i(t) - B)^2 \tag{8.33}$$

この式が正定であることは明らかである。ここで、 f_i の微分は、以下のように与えられる。

$$\dot{f}_i = \dot{x}_i(t)(x_i(t) - B) \tag{8.34}$$

上式の正負は、 $\dot{x}_i(t)$ の符号により決定される。ここで、(8.30) 式より、 $\dot{x}_i(t)$ を計算すると、

$$\dot{x}_i(t) = \dot{\beta}(3 - 3\alpha + \alpha^2) + \beta(2\dot{\alpha}\alpha - 3\dot{\alpha}) \tag{8.35}$$

この式に α 及び β の値を代入することにより、 \dot{f}_i の負定性を示すことができる。よって、リアプノフの安定性理論より、 $x_i(t)$ は一定値に収束する。

第 9 章

まとめ

本章においては、これまでに提案した手法及びその検証結果についてまとめる。

本論文においては、主に群ロボットの formation 制御の中でも正三角形格子を目標形状とする制御手法において、ロボット間の通信に損失がある場合を想定し、失われた情報をいかにして効率的に推定するかという課題について取り組んだ。まずは前段階として、4章においてランダムな初期位置に配置されたロボットが目標とする正三角形格子を形成していく段階において、ある時点でのロボット配置が目標形状に対してどの程度近付いている/離れているかを定量的に測るための評価指標を提案し、その有効性について検証を行った。

一般に群ロボットにおいては個々のロボットの持つ計算能力に制約があるため、従来の制御理論において提案されている理論的なフィルタは、高い精度で現在の状態を推定できる一方で多くの計算量を必要とすることから、群ロボットに直接適用することが困難である。こうした観点に基づき、5章においては従来より単純で計算量を必要としない外挿手法として提案されている 0 次ホールド、線形外挿、2 次外挿の 3 種類の外挿方法を用いて通信損失時の状態推定を行い、その精度について検証を行った。その結果、0 次ホールドや線形外挿では高い精度で失われた情報を推定することができず、また 2 次外挿においてはこれら 2 つの手法に比べて更に大きな推定誤差を生じる結果となった。この傾向は特に通信損失確率 α の値が大きくなった場合について顕著である。この原因としては、今回想定している群ロボットの formation 制御においては、そのアルゴリズムの特性上、初期状態から目標とする形状を形成していく際のロボットの動きが徐々に小さくなる傾向にあ

るため、直前の値によっては現在値を過去の値から増大させて算出してしまふ2次関数を用いた外挿手法は適していないためであると考えられる。一般に、多項式を用いた外挿法においては多項式の次数 n の値を大きくしていくことにより推定精度を向上させることが期待できる場合もあるが、本論文で扱う問題についてはこの考え方は適用できず、これ以上多項式の次数を挙げても推定精度を向上させることは期待できないため、別の外挿式を考案する必要があると考えられる。このため、5章においてはロボットの位置情報そのものだけでなく、ロボットの移動速度も考慮した速度0次ホールド及び速度線形外挿の2種類の外挿方法についても検証を試みたが、結果的に推定精度を向上することはできなかった。この理由についても、ロボットは目標形状に向けて徐々にその移動距離を小さくしていくため、速度0次ホールドにおいてはこの点を考慮した推定を行うことができず、また速度線形外挿においては速度を減少させることはできても、その移動方向（ベクトル）までは考慮されていないため、推定精度を向上させることができなかったものと考えられる。

こうした点から、本論文においては正三角形格子を目標形状とする群ロボットの formation 制御においては、ロボットが指数関数的に目標形状に向かって収束しているという点に着目し、指数関数を外挿式として用いることにより、計算量を必要としない単純な外挿式で推定精度を向上させることを試みた。6章においてはまず、単純な指数関数を用いた指数外挿式について提案し、その数学的特性について解析を行い、その推定精度について考察を行うと共に、その有効性についてシミュレーションによる検討を行った。その結果、提案した指数外挿式により、5章で検証を行った3種類の手法(0次ホールド、線形外挿、2次外挿)に比べて推定精度を改善することに成功し、また計算量を抑えたPF等の理論的なフィルタと比べても良好な推定結果を得ることができた。しかし、一方で推定誤差に関する理論的な下限値を表す Cramér-Rao の下界式と比べるとその推定誤差にはまだ隔たりがあり、更なる推定誤差の向上を目指す必要があることが判明した。

このため、7章においては6章で提案した指数関数外挿式に対して改良を加えることにより、その推定精度を Cramér-Rao の下界式に近付けることを試みた。ここでは、指数関数の定義域の値が比較的小さい範囲においては値域の値の変化が比較的大きいのに対し、定義域の値が大きくなると値域の値の変化が鈍くなりあま

り変化がなくなるという数学的特性に着目した。このことは、ロボットが目標形状に近付き、1ステップあたりの動きが鈍くなってきた際には、単純な指数関数外挿式では必要な動きを推定することが難しくなることを示している。この問題を解決するため、指数関数の値の変化が乏しい領域におけるロボット位置の変化をサポートするため、1次関数を指数関数を用いた外挿式に加えることによって、値の変化の小さな領域における応答性の改善を行った。その結果、Cramér-Raoの下界式と比べるとまだ隔たりはあるものの、従来の単純な指数関数を用いた外挿式と比べてその推定精度を更に改善することができた。

最後に、今回提案した指数関数を用いた外挿手法が一般にも適用できることを示すため、8章においてロボットの挙動が1階の微分方程式((3.8)式及び(3.9)式)ではなく、2階、 \dots 、 n 階の微分方程式に従う場合についても考察し、その数学的な安定性について示した。このことより、本論文で提案した指数外挿式は群ロボットの従う微分方程式に関係なく、幅広く利用できる手法であると考えることができる

なお、本論文においてはロボットの収束アルゴリズムとして正三角形格子を形成するものを例として扱ったが、このアルゴリズムにおいて用いられているロボットの位置及び角度についての1階の微分方程式((3.8)、(3.9))はこの正三角形格子を構成するアルゴリズムに限られたものではなく、一般的な formation 制御において広く用いられていることが文献[82]に示されている。実際、円を目標形状とするアルゴリズム[14]や直線・弧・四角形・六角形などを目標形状とするアルゴリズム[18]などにおいて各ロボットの挙動が(3.8)、(3.9)式と同様の微分方程式で表されることが示されている。こうしたことから、本研究で提案した指数関数を用いた外挿手法は論文中で用いた正三角形格子を目標形状とするアルゴリズムに限らず、群ロボットの formation 制御に広く用いることのできる汎用性のある手法であると考えることができる。

第 10 章

結言

10.1 まとめ

本論文においては、群ロボットにおける正三角形格子の formation 制御を例に取り、ロボット間の通信に一定確率で通信損失が発生した場合に失われた情報を外挿する新たな手法について提案し、その有効性について検証を行った。また、その検証の際に必要な、ロボットの目標形状（正三角形格子）への収束度を測るための評価指標についても提案し、その妥当性について考察した。本論文における主な成果は以下の通りである。

- ・群ロボットの正三角形格子への配置問題において、ロボットが目標形状である正三角形格子に対してどの程度収束しているかを定量的に示すための評価指標として、その時点における各ロボットの位置を頂点とする Delaunay 三角形を構成し、その各辺の長さ及び面積から計算される評価指標（EI）を提案し、この指標がロボットの目標形状（正三角形）への収束度を示す値として妥当であることを示した。

- ・群ロボットの formation 制御において、ロボット間の通信に一定確率で損失がある場合を想定し、失われた情報を従来から存在する単純な 3 種類の外挿手法（0 次ホールド、線形外挿、2 次外挿）により外挿した場合の推定誤差について検証を行った。その結果、これら既存の手法ではロボットが正しく目標形状に向かって収束していくために十分な推定精度を持っていないことが確認された。

- ・群ロボットの正三角形を目標形状とする formation 制御において、ロボット間

の通信に一定確率で通信損失が発生する場合を想定し、通信損失が発生した場合に失われた情報を計算量を必要とせず高い精度で外挿できる手法として、この場合においてロボットが目標形状に向けて指数関数的に収束していくという特性に着目した、指数関数を用いた外挿手法 (EE) を新たに導入し、この手法が従来の 0 次ホールドなどの単純な外挿手法と比べて高い精度でロボット位置を推定できることを確認した。

・前述の指数関数を用いた外挿手法における推定精度を更に高めるため、この手法を改良した外挿式を考案した。まずは前述の指数関数を用いた外挿式を用いて計算した外挿値を直前の変化方向とは逆方向に与える、改良された外挿式 (EEE) を提案しその有効性について検証を行った。続いて、外挿式として指数関数に加えて 1 次関数も加えた新たな外挿式 (EEF 及び EEEF) を考案し、その効果について検証を行った。その結果、外挿式として 1 次関数を加えることにより、指数関数のみを用いた外挿式と比べて推定精度を更に向上させることができた。

・当初、指数関数外挿式の適用対象としては、群ロボットが正三角形格子を目標形状とする formation 制御、即ちロボットが 1 階の微分方程式に従う場合を想定していたが、これをロボットが 2 階、 \dots 、 n 階の微分方程式に従う場合についても適用し、その数学的な安定性について示した。

10.2 今後の課題

今後の課題としては、至近のものとしてはロボット台数を増加させた場合に同様の結果が得られるかどうかということが挙げられる。今回のシミュレーションでは、扱うロボットの台数を $n = 10$ に限定して計算を行っているが、実際の応用を考えた場合には、ロボット台数 n の値を変化させた場合に同様の結果が得られるかどうかを検討する必要がある。また、本論文における研究においては、コンピュータによるシミュレーションを用いて提案手法の有効性について検証を行ったが、当然その次の段階として、実機を用いた実験による検証を行う必要があると考えられる。他にも、今回提案した手法によって目標形状への収束が保証される通信損失確率の上限値についても検証を行う必要があると考えられる。システム制御の分野において、システム内の通信に損失がある場合に、安定してそのシ

システムを制御することができる通信損失確率の上限値についての理論的な考察が文献 [83]-[86] などで行われている。これらの理論を今回提案した手法に応用することにより、実際に本論文の手法がどの程度の通信損失に対してまで有効であるかを検討する必要があると考えられる。また、今回はロボット間の情報交換として通信のみを仮定しているが、通信以外の方法により失われた情報を 2 次的に補完することができた場合、どのような影響を及ぼすかということについて検証する必要があると考えられる。

謝辞

本論文を書き上げる際には、私の所属する北陸先端科学技術大学院大学の方々を始め、様々な方のお世話になりました。

主指導教員である丁洛榮教授には、研究テーマの決定から研究の方針、論文の投稿先の決定、出張の手続きなど些細な点に至るまでお世話になりました。改めて感謝申し上げます。

副指導教員である浅野文彦准教授には、研究のみならず学生生活全般についてアドバイスを頂き、大変参考になりました。時には品川での浅野研のゼミにも参加させて頂き、自分の研究分野以外の議論に参加できたことは私にとって良い経験となりました。

副テーマ指導教員である Xavier Défago 准教授（現東京工業大学教授）には、副テーマ指導の際に毎月のように品川に足を運んでいただき、熱心にご指導頂き大変に感謝しております。Défago 先生には単に研究指導に留まらず、研究者としての心構えや研究の進め方についても適切なアドバイスを頂きました。改めて感謝申し上げます。

宮崎大学工学部の李根浩准教授には、論文の審査のみならず、journal 論文投稿の際にも大変お世話になりました。李先生との長時間に渡る議論のおかげで、論文の質を高め採録を決めることができました。JAIST を離れてからも多くの時間を私のために割いて頂き、改めて感謝申し上げます。

大阪大学基礎工学部の新井健生教授、及び産業技術総合研究所の谷川民生先生には本論文審査の際にお世話になりました。私の論文審査のために遠方からお越し頂き、有益なご意見を頂いたことに感謝しております。

最後に、東京サテライトで共に学んだ学生の皆さんにも感謝申し上げます。博士後期課程では履修する授業が少なく、他の学生さんと出会う機会が少なかったのですが、それでも何人かの方とは知り合うことができ、社会人学生ならではの

悩みについて語り合うことで勉学を続ける意欲を分けて頂きました。

本論文は多くの方々の支えによって完成させることができました。本研究に関わった全ての方に対して、改めて感謝申し上げます。

参考文献

- [1] Y. Tan and Z.-Y. Zheng, Research Advance in Swarm Robotics, Defence Technology, vol.9, no.1, pp.18-39, 2013
- [2] M. Yogeswaran and S.G. Ponnambalam, Swarm Robotics: An Extensive Research Review, Advanced Knowledge Application in Practice, InTech, pp.259-278, 2010
- [3] E. Sahi, Swarm Robotics: From Sources of Inspiration to Domains of application, Lecture Notes in Computer Science. vol.3342, Springer-Verlag, pp.10-20, 2005
- [4] G. Beni and J. Wang, Swarm Intelligence in Cellular Robotic Systems, Proceedings of NATO Advanced Workshop on Robots and Biological Systems, 1989
- [5] L. Qiu, J. Chen, T. Iwasaki, and H. Fujioka, Developments in Control Theory Towards Glocal Control, The Institution of Engineering and Technology, 2012
- [6] S. Hara, J. Imura, and K. Tsumura, Multi-Resolved Dynamical system Theory for Large Scale Complex systems, Proceedings of SICE Annual Conference, pp.3115-3118, 2008
- [7] D. Tsubakino and S. Hara, Eigenvector-Based Intergroup Connection of Low Rank for Hierarchical Multi-Agent Dynamical systems, Systems & Control Letters, vol.61, no.2, pp.354-361, 2012
- [8] T. H. Kim, S. Hara, and Y. Hori, Cooperative Control of Multi-Agent Dynamical Systems in Target-Enclosing Operations Using Cyclic Pursuit Strategy, International Journal of Control, vol.83, no.10, pp.2040-2052, 2010

- [9] S. Hara, T. Hayakawa, and H. Sugata, Stability Analysis of Linear Systems with Generalized Frequency Variables and Its Application to Formation Control, Proceedings of the 46th IEEE Annual Conference on Decision and Control, pp.1459-1466, 2007
- [10] T. Hayakawa, T. Matsuzawa, and S. Hara, Formation Control of Multi-Agent Systems with Sampled Information, Mathematical Engineering Technical Report by Tokyo University, 2006-55, 2006
- [11] G. Lee and N. Y. Chong, A Geometric Approach to Deploying Robot Swarms, Annals of Mathematical and Artificial Intelligence, vol.52, no.2-4, pp.257-280, 2009
- [12] G. Lee and N. Y. Chong, Adaptive Flocking of Robot Swarms: Algorithms and Properties, IEICE Transactions on Communications, vol.E91-B, no.9, pp.2848-2855, 2008
- [13] G. Lee and N.Y Chong, Adaptive Self-Configurable Robot Swarms Based on Local Interaction, IEEE/RSJ International Conference of Intelligent Robots and Systems, pp.4182-4187, 2007
- [14] G. Lee S. Yoon, N. Y. Chong, and H.Christensen, A Mobile Sensor Network Forming Concentric Circles through Local Interaction and Consensus Building, Journal of Robotics and Mechatronics, vol.21, no.4, pp.469-477, 2009
- [15] X. Defago and S.Souissi, Non-Uniform Circle Formation Algorithm for Oblivious Mobile Robots with Convergence toward Uniformity, Theoretical Computer Science, vol.396, no.1-3, pp.97-112, 2008
- [16] G. Vassrhelyi, Cs. Viragh, N. Tarcai, T. Szorenyi, G. Somorjai, T. Nepusz and T. Vicsek, Outdoor Flocking and Formation Flight with Autonomous Aerial Robots, Proceedings of IEEE/RSJ International Conference of Intelligent Robots and Systems, 2014

- [17] E. Ferrante, A. E. Turgut, C. Huepe, A. Stranieri, C. Pincioli, and M. Dorigo, Self-Organized Flocking with a Mobile Robot Swarm: A Novel Motion Control Method, *Adaptive Behavior*, vol.20, no.6, pp.460-477, 2012
- [18] G. Lee and N. Y. Chong, Decentralized Formation Control for Small-Scale Robot Teams with Anonymity, *Mechatronics*, vol.19, no.1, pp.85-105, 2009
- [19] B. Shucker, T. D. Murphey, and J. K. Bennett, Convergence-Preserving Switching for Topology-Dependent Decentralized Systems, *IEEE Transactions on Robotics*, vol.24, no.6, pp.1405-1415, 2008
- [20] D. Kurabayashi, K. Okita, and T. Funato, Obstacle Avoidance of a Mobile Robot Group a Nonlinear Oscillator Network, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.186-191, 2006
- [21] S. Camazine, J. L. Deneubourg, N. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau, *Self-Organization in Biological Systems*, Princeton University Press, 2001
- [22] G. Beni and J. Wang, Swarm Intelligence, In *Proceedings of the seventh Annual Meeting of the Tokyo, Japan*, pp.425-428, 1989
- [23] C. Kube and R. Zhang, Collective Robotics: From Social Insects to Robots, *Adaptive Behavior*, vol.2, pp.189-218, 1994
- [24] R. Alami, H. Asama, and R. Chatila, *Proceedings of the Seventh International Symposium on Distributed Robotic Autonomous Systems*, 2004
- [25] R. Arkin and G. Bekey, Robot Colonies, *Autonomous robots*, vol.4, 1997
- [26] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and Cayirci, A Survey on Sensor Networks, *IEEE Communication Magazine*, pp.102-115, 2002
- [27] G. Chirikjian, Kinematics of a Metamorphic Robot System, In *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 449-455, 1994

- [28] M. Yim, Y. Zhang, J. Lamping, and E. Mao, Distributed Control for 3D Metamorphosis, *Autonomous Robots*, vol.10, pp.41-56, 2001
- [29] T. Balch, Hierarchic Social Entropy: An Information Theoretic Measure of Robot Group Diversity, *Autonomous Robots*, vol.8, pp.209-238, 2000
- [30] K. Sugawara and M. Sano, Cooperative Acceleration of Task Performance: Foraging Behavior of Interacting Multi-Robot System, *Physica D*, vol.100, pp.343-354, 1997
- [31] I. Suzuki and M. Yamashita, Distributed Anonymous Mobile Robots: Formation of Geometric Patterns. *SIAM Journal on Computing* vol.28, no.4, pp.1347-1363, 1999
- [32] X. Défago and A. Konagaya, Circle Formation for Oblivious Anonymous Mobile Robots with No Common Sense of Orientation, In *Proceedings 2nd ACM International Workshop on Principles of Mobile Computing*, pp.97-104, 2002
- [33] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer, Pattern Formation by Autonomous Robots without Chirality. In *Proceedings of SIROCCO*, pp.147-162, 2001
- [34] S. Carpin and L. E. Parker, Cooperative Leader Following in a Distributed Multi-Robot System, In *Proceedings of IEEE International Conference on Robotics and Automation*, pp.2994-3001, 2002
- [35] Z. Cao, M. Tan, S. Wang, Y. Fan, and B. Zhang, The Optimization Research of Formation Control for Multiple Mobile Robots, In *Proceedings of 4th World Congress on Intelligent Control and Automation*, pp.1270-1274, 2002
- [36] F. Mondada, G. C.Pettinaro, A. Guignard, I. Kwee, D. Floreano, J. L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo, Swarm-bot: A New Distributed Robotic Concept, *Autonomous Robots* vol.17, no.2-3, pp.193-221, 2004

- [37] G. Baldassarre, V. Trianni, M. Bonani, F. Mondada, M. Dorigo, and S. Nolfi, Self-Organized Coordinated Motion in Groups of Physically Connected Robots, *IEEE Transactions on Systems, Man, and Cybernetics. Part B*, vol.37, no.1, pp.224-239, 2007
- [38] Y. Ikemoto, Y. Hasegawa, T. Fukuda, and K. Matsuda, Graduated Spatial Pattern Formation of Robot Group, *Information Science*, vol.171, no.4, pp.431-445, 2005
- [39] A. Ishiguro and T. Kawakatsu, Self-Assembly through the Interplay between Control and Mechanical Systems, In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.631-638, 2006
- [40] M. Shimizu, T. Mori, and A. Ishiguro, A Development of a Modular Robot that Enables Adaptive Reconfiguration, In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.174-179, 2006
- [41] B. Werger and M. J. Mataric, From Insect to Internet: Situated Control for Networked Robot Teams, *Annals of Mathematics and Artificial Intelligence*, vol.31, pp.173-198, 2001
- [42] T. Balch and M. Hybinette, Social Potentials for Scalable Multi-Robot Formations, In *Proceedings of IEEE International Conference on Robotics and Automation*, pp.73-80, 2000
- [43] A. Howard, M. J. Mataric, and G. S. Sukhatme, Mobile Sensor Network Deployment Using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem, In *Proceedings of 6th International Symposium on Distributed Autonomous Robotic Systems*, pp.299-308, 2002
- [44] W. M. Spears and F. D. Gordon, Using Artificial Physics to Control Agents, In *Proceedings IEEE International Conference on Information, Intelligence, and Systems*, pp.281-288, 1999

- [45] W. Spears, D. Spears, J. Hamann, and R. Heil, Distributed, Physics-Based Control of Swarms of Vehicles. *Autonomous Robots* vol.17, no.2-3, pp.137-162, 2004
- [46] K. Fujibayashi, S. Murata, K. Sugawara, and M. Yamamura, Self-Organizing Formation Algorithm for Active Elements, In *Proceedings of 21st IEEE Symposium on Reliable Distributed Systems*, pp.416-421, 2002
- [47] J. McLurkin and J. Smith, Distributed Algorithms for Dispersion in Indoor Environments Using a Swarm of Autonomous Mobile Robots, In *Proceedings of 7th International Symposium on Distributed Autonomous Robotic Systems*, pp.831-890, 2004
- [48] B. Shucker, T. Murphey, and J. K. Bennett, Switching Rules for Decentralized Control with Simple Control Laws, In *Proceedings of American Control Conference*, pp.1485-1492, 2007
- [49] J. Reif and H. Wang, Social Potential Fields: A Distributed Behavioral Control for Autonomous Robots, *Robotics Autonomous Systems*, vol.27, no.3, pp.171-194, 1999
- [50] Y. Zou and K. Chakrabarty, Sensor Deployment and Target Localization Based on Virtual Forces, In *Proceedings of IEEE Infocom Conference*, pp.1293-1303, 2003
- [51] Y. F. Zheng and W. Chen, Mobile Robot Team Forming for Crystallization of Protein. *Autonomous Robotics*, vol.23, pp.69-78, 2007
- [52] N. Heo and P. K. Varshney, A Distributed Self Spreading Algorithm for Mobile Wireless Sensor Networks, In *Proceedings of IEEE Wireless Communication and Networking Conference*, pp.1597-1602, 2003
- [53] R. Cohen and D. Peleg, Local Algorithms for Autonomous Robots Systems, *Lecture Notes in Computer Science*, vol.4056, pp.29-43, 2006

- [54] G. L. Wang, G. Cao, and T. L. Porta, Movement-Assisted Sensor Deployment, In Proceedings of IEEE Infocom Conference, pp.2469-2479, 2004
- [55] E. Martison and D. Payton, Lattice Formation in Mobile Autonomous Sensor Arrays. Lecture Notes in Computer Science, vol.3342, pp.98-111, 2005
- [56] P. Fazli, A. Davoodi, P. Pasquier, and A. K. Mackworth, Complete and Robust Cooperative Robot Area Coverage with Limited Range, IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.5577-5582, 2010
- [57] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, Coverage Control for Mobile Sensing Networks, IEEE Transactions on Robotics and Automation, vol.20, No.2, pp.243-254, 2004
- [58] G. Hollinger and S. Singh, Multi-Robot Coordination with Periodic Connectivity, IEEE International Conference on Robotics and Automation, pp.4457-4462, 2010
- [59] J. H. Seok, J. Y. Lee, C. oh, J. J. Lee, and H. J. Lee, RFID Sensor Deployment Using Differential Evolution for Indoor Mobile Robot Localization, IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.3719-3724. 2010
- [60] R. Fujisawa, H. Imamura, and F. Matsuno, Cooperative Transportation of Robot Swarm Using Pheromone Communication, Proceedings of International Symposium on Distributed Autonomous Robotic Systems, pp.559-570, 2012
- [61] F. Ducatelle, G. A. D. Caro, C. Pinciroli, F. Mondada, and L. Gambardella, Communication Assisted Navigation in Robotic Swarms: Self-Organization and cooperation, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.4981-4988, 2011
- [62] N. Hoff, A. Sagoff, R. J. Wood, and R. Nagpal, Two Foraging Algorithms for Robot Swarms Using Only Local Communication, Proceedings of IEEE International Conference on Robotics and Biomimetics, pp.123-130, 2010

- [63] R. Iyengar, K. Kar, and S. Banerjee, Low-Coordination Topologies for Redundancy in Sensor Networks, The ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp.332-342, 2005
- [64] I. Stojmenovic and J. Wu, Broadcasting and Activity Scheduling in Ad Hoc Networks, Mobile Ad Hoc Networking. pp.205-229, IEEE press, 2004
- [65] 浅野, 計算幾何：理論の基礎から実装まで, 共立出版, 2007
- [66] 城, 秋本, 有限要素法のための整合分割プログラム, 精密機械学会誌, vol.46, no.3, pp.286-289, 1980.
- [67] N. Ramakoti, A. Vinay, and R. K. Jatoth, Particle Swarm Optimization Aided Kalman Filter for Object Tracking, Proceedings of IEEE International Conference on Advances in Computing, Control, and Telecommunication Technologies, pp.531-533, 2009
- [68] L. C. Carrillo-Arce, E. D. Nerurkar, J. L. Gordillo, and S. I. Roumeliotis, Decentralized Multi-Robot Cooperative Localization Using Covariance Intersection, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.1412-1417, 2013
- [69] E. D. Nerurkar and S. I. Roumeliotis, A Communication-Bandwidth-Aware Hybrid Estimation Framework for Multi-Robot Cooperative Localization, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.1418-1425, 2013
- [70] K. Y. K. Leung, T. D. Barfoot, and H. H. T. Liu, Decentralized Localization of Sparsely-Communicating Robot Networks: A Centralized-Equivalent Approach, IEEE Transactions on Robotics, vol.26, no.1, pp.62-77, 2010
- [71] A. Marjovi, J. Nunes, P. Sousa, R. Faria, and L. Marques, An Olfactory-Based Robot Swarm Navigation Method, Proceedings of IEEE International Conference on Robotics and Automation, pp.4958-4963, 2010

- [72] D. Fox , S. Thrun , W. Burgard, and F. Dellaert, Particle Filters for Mobile Robot Localization, Sequential Monte Carlo Methods in Practice, Springer-Verlag, pp.470-498, 2001
- [73] 片山, 非線形カルマンフィルタ, 朝倉書店, 2011
- [74] J. E. Slotine and W. Li, *Applied nonlinear control*. Prentice-Hall, 1991
- [75] H. Cramér: Mathematical Methods of Statistics, Princeton University Press, (1946)
- [76] 原, 位置推定における統計的推定理論, IEICE Fundamentals Review, vol.4, no.1, pp.32-38, 2010
- [77] D. L. Snyder and I. B. Rhodes, Filtering and Control Performance Bounds with Implications on Asymptotic Separation, Automatica, vol.8, pp.747-753, 1972
- [78] M. Zakai and J. Ziv, Lower and Upper Bounds in the Optimal Filtering Error of Certain Diffusion Processes, IEEE Transactions on Information Theory, vol.IT-18, pp.325-331, 1972
- [79] B. Z. Bobrovsky and M. Zakai, A lower Bound in the Estimation Error for Markov Processes, IEEE Transactions on Automatical Control, vol.AC-20, pp.785-788, 1975
- [80] I. B. Rhodes and A. S. Gilrman, Cone-Bounded Nonlinearities and Mean Square Bounds - Estimation Lower Bound, IEEE Transactions on Automatical Control, vol.AC-20, pp.632-642, 1975
- [81] J. Taylor, The Cramér-Rao Estimation Error Lower Bound Computation for-Deterministic Nonlinear System, IEEE Transaction on Automatic Control, **24**-2, pp.343-344(1979)
- [82] K. Oh, M. Park, H. Ahn, A Survey of Multi-Agent Formation Control, Automatica, vol.53, pp.424-440, 2015

- [83] B. Azimi-Sadjadi, Stability of Networked Control Systems in the Presence of Packet Losses, Proceedings of IEEE 42nd International Conference on Decision and Control, pp.676-681, 2003
- [84] O. C. Imer, S.Yuksel, and T. Basar, Optimal Control of LTI Systems over Unreliable Communication Links, Automatica, vol.42, no.9, pp.1429-1439, 2006
- [85] H. Ishii, Stabilization under Shared Communication with Message Losses and Its Limitations, Proceedings of IEEE 45th International Conference on Decision and Control, pp.4974-4979, 2006
- [86] N. Elia, Remote Stabilization over Fading Channels, Systems & Control Letters, vol.54, no.3, pp.238-249, 2005
- [87] H. L. Van Trees, Detection, Estimation, and Modulation Theory, PartI, New York:Wiley,1968
- [88] G. Sansone and R. Conti, Non-Linear Differential Equation, Oxford, England: Pergamon, 1964

本研究に関する発表論文

- [1] 神ノ尾 淳, 丁 洛榮, 李 根浩, 群ロボットの隊形制御における指数関数を用いた外挿手法の提案, 計測自動制御学会論文集, vol.52, no.1, pp.28-36, 2016
- [2] 神ノ尾 淳, 李 根浩, 丁 洛榮, 地理的に制約のある環境下における群ロボットの自己組織化, 日本機械学会論文集, 投稿中
- [3] A. SHINNOH ,N.Y. CHONG, and G. Lee, Communication Packet Loss Concealment for Pattern Generation with Robotic Swarms, IEEE International Conference on Advanced Intelligent Mechatronics, pp.537-542, 2015
- [4] A. SHINNOH , N.Y. CHONG, and G. Lee, Application of Exponential Extrapolation to n-Order Differential Equation, IEEE/SICE International Symposium on System Integration 2016, 投稿中
- [5] 神ノ尾 淳 , Défago Xavier, 丁 洛榮 , 李 根浩, 群ロボットの目標形状への収束度の評価手法-正三角形格子を例として-, ロボティクス・メカトロニクス講演会 2014, 2A2-J01
- [6] 神ノ尾 淳 , 丁 洛榮 , 李 根浩, 群ロボットにおける通信損失時の推定手法の比較-正三角形格子を例として, ロボティクス・メカトロニクス講演会 2015 , 2A1-K01