2009年度 博士論文

# モデル予測制御に基づく複数移動体の編隊制御に関する研究

京都大学大学院　博士後期課程
工学研究科　　　機械理工学専攻
1060-21-8331 根 和幸
**指導教員** ： 松野 文俊教授

# 目次

# 図目次

# 表目次

# 第1章

# 序論

## 1.1　はじめに

複数の移動ロボットによる協調作業の有用性は，広範囲の環境センシング[1]，自己位置推定[2, 3]，マッピング[4]-[6]，広範囲の探索[7]，大質量物体の協調運搬[8]などさまざまな分野において指摘されている．これらは単体のロボットでは実現が困難なタスクで，単に作業速度・効率が向上が図れるというのみでなく，複数の移動ロボットが協調することで実現が可能となるタスクの例である．

複数の移動ロボットの協調作業のなかで最も基本的で重要な機能の１つに“ 集団での移動 ”が挙げられる．複数の移動ロボットが同じ環境中で作業を行う場合，どのようなタスクにおいても集団での移動は必要となる機能である．また，移動ロボット同士の衝突回避を考慮しなくてはならないといった単体ロボットの場合にはない課題が存在する．

複数移動体の“移動”に着目した研究は，生物の群れ行動[9, 10](例えばFig. 1.1の渡り鳥のV字飛行やFig. 1.2の魚の群れ)からヒントを得たようなアルゴリズム[11, 12]，群れ行動自体の数理モデル化[13]などが古くから報告されている．編隊(フォーメーション)制御はこのような生物規範のアルゴリズムから発展し，制御理論的アプローチから近年盛んに研究が行われている分野である．本研究も複数の移動ロボットの“移動タスク”に対して，編隊制御の観点から取り組むものである．

## 1.2　編隊制御

編隊(フォーメーション)制御とは移動体同士の相対位置関係を制御することで，移動体群としては一定の形状を維持しながら集団で移動をさせることを目的としたものである([14]-[44])．編隊制御を用いることで，移動体同士の相対位置関係が制御されるため移動体同士の衝突が回避され，安全に移動させることが期待できる．また，移動指令は個々の移動体ごとではなく移動体群全体に対して与えればよいので移動指令の簡単化，効率化も期待できる．さらに，特定の目標形状(編隊形状)を与えることで解析が容易となる

Fig. 1.1:Formation fight of migrating birds

ため，制御系設計の面から複数移動体の移動を取り扱いやすくする利点もある．このような利点から複数台での運用が考えられる，UGV[45]，飛翔体[46, 47]，自律型無人探査機[48, 49]，衛星[50, 51]など様々な移動体を対象とした研究が盛んに報告されている．また，特定の形状を利用することで形態把持を行うように物体を運搬する応用例も報告されている[52]-[54].

編隊制御に関する研究は制御方策の特徴から，大きく分けてLeader-Follower型に基づくもの[14]-[19]，Behavior-based　Approachに基づくもの[20]-[22]，グラフの定型性に基づくもの[23]-[25]などに分類されている．これらの手法の関連性について[26]-[28]などで解析が行われている．また，近年では，移動ロボットが有するセンシング能力や通信能力を陽に考慮した研究も報告されている([29]-[33]).

編隊制御に関する課題の 1 つに，移動体同士の衝突回避が挙げられる．例えば，ある初期状態から目標の編隊形状を形成する際には移動体同士の相対位置関係が変化するため，移動体同士の衝突回避を陽に考慮しなくてはならない([35]-[44])．また，編隊形状の切り替えは，環境中に存在する障害物を回避することと合わせて議論されることが多く，障害物回避と編隊形状

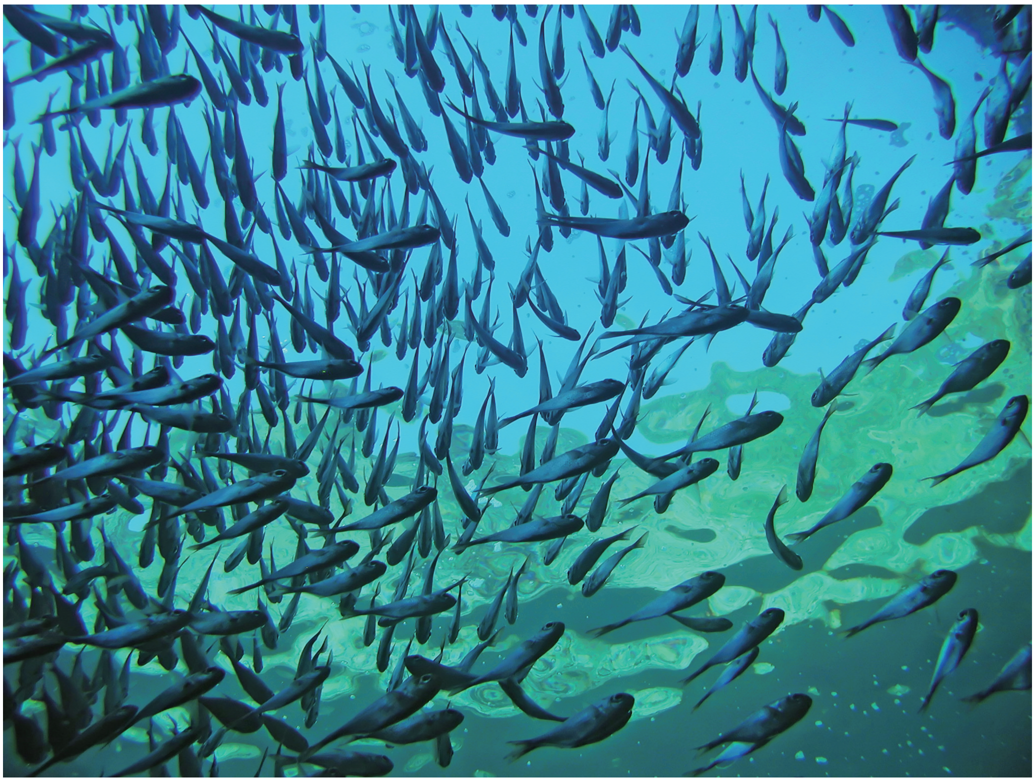


Fig. 1.2:Schooling of fishes

維持という相反する制御目的をどのように実現するか[41, 64]，環境に合わせて編隊形状をいかに生成・切り替えるか[42]-[44]などに着目した研究も報告されている．

移動体同士の衝突回避を考慮する手法としてよく用いられるのが人工ポテンシャル法に基づく手法である[35]-[40]. 人工ポテンシャル法とは移動体と目標位置に仮想的な引力ポテンシャルを，移動体間や移動体と環境間に仮想的な斥力ポテンシャルを作用させることで衝突を回避しながら目標位置まで収束させる方法である．参考文献[37]ではNavigation-function[38]を用いることで障害物が存在する環境中でも複数移動体の編隊走行を実現した．また，参考文献[39]では同様にNavigation-functionに基づく手法を用いることで局所的な情報のみを用いて衝突回避を実現しながら大域的な収束性を保証している．その他の衝突回避手法としては[41]-[44]などが報告されている．参考文献[41]は仮想的なバネーダンパ構造を導入することで移動体間および移動体環境間に仮想的な力を作用させるもので，移動体同士の衝突回避と障害物回避を実現している．参考文献[16]，[42]-[44]はハイブリッド制御に基づくものであり，複数の制御器を環境に応じて切り替えることで編隊形

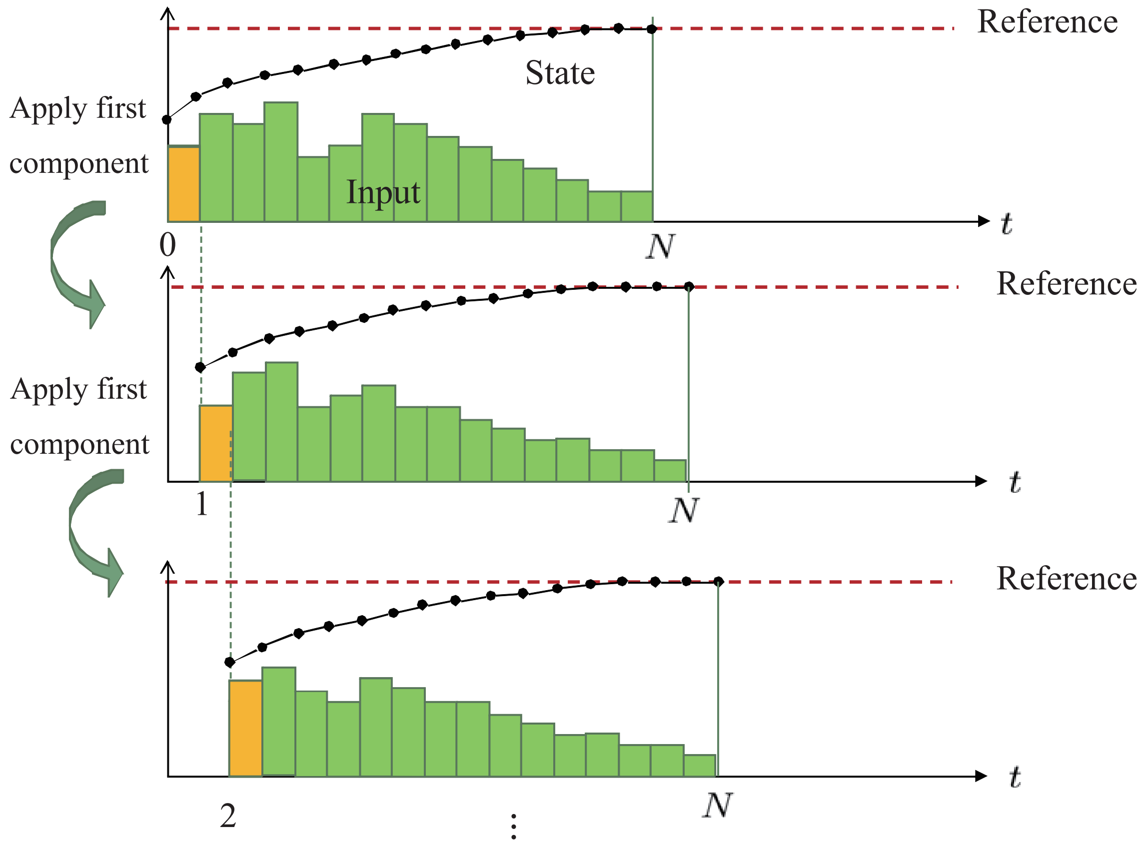



Fig. 1.3:Basic idea of MPC. $N$ is a number of prediction steps.

状を変化させ，障害物と移動体同士の衝突回避を行うものである．

## 1.3　モデル予測制御に基づく衝突回避

複数移動体同士の衝突回避問題を取り扱う系統的な手法としてモデル予測制御が挙げられる．モデル予測制御とは有限時間未来までの状態をオンラインで予測し，制約付きの最適制御問題を解くことで制御入力を決定する手法である (e.g. [55]-[57])．モデル予測制御では最適制御問題を各サンプリング毎に解き, その一番最初の入力のみを適用するということを繰り返し行う (Fig. 1.3 参照)．そのため，予測区間が時間経過とともに後退していくことから後退ホライズン制御 (Receding Horizon Control) とも呼ばれる．モデル予測制御は上述のようにオンラインで最適制御問題を解く必要があるため従来は反応速度が比較的遅い化学プラント等に用いられることが多かったが，

近年の計算機の発達に伴い，速度の速い機械システムへの適用例も報告されている (e.g. [58, 59] ).

移動体同士の衝突回避条件は0,1変数を含む不等式制約として記述することができるため，モデル予測制御により衝突回避問題を取り扱うことが可能である．このような不等式条件を用いた場合，最適制御問題は混合整数計画問題 (Mixed Integer Programming) として定式化することができる ([63]-[66]). モデル予測制御を用いることの利点として位置や速度，入力に関する制約のもとで移動体の軌道の最適性を陽に考慮することができることである [60, 61, 62]. したがって，人工ポテンシャル法に基づく手法 ([35]-[40]) などと比較し，移動体同士の相対距離の下限や移動体の持つ物理的限界を陽に設定できる点が長所であると考えられる．

その一方で，衝突回避問題が定式化される混合整数計画問題はNP困難 [70] な問題であるため，移動体の台数や予測時間の長さに応じて計算時間が急激に増加し，オンラインで解くことが困難となってしまうという問題が存在する．この問題に対しては，静止障害物回避問題を対象に混合整数計画問題を固定端の最適制御問題に帰着させることで計算量を低減化する研究 [67] などが報告されている．また，モデル予測制御に基づく衝突回避では，衝突回避のための不等式制約の下で最適な開ループ入力を求めるため，実装の際には離散時間の最適制御問題を解くことになる．すなわち，有限時間未来先までの離散時間の状態を予測し，その各予測時刻において衝突回避を考慮している．したがって，離散的な予測時刻でしか衝突回避を考慮していないため，得られた軌道は予測時刻間で衝突する可能性がある．特に，目標の状態への収束性を保証するには予測時間をある程度長く確保する必要があるため，サンプリング周期を十分短くできない場合があり，予測時刻間の衝突が深刻な問題となる．すなわち，本質的に離散化の影響を陽に考慮しなければ安全な衝突回避は実現できないと言える．静止障害物回避問題を対象に離散化の影響を考慮した研究が報告されているが [65, 66]，移動体同士の衝突回避に対して考慮した例はない．これは，移動体同士の衝突回避では自身の挙動だけでなく回避対象である他移動体の挙動も影響を与え，より取り扱いが困難となるためであると考えられる．

## 1.4　本論文の目的

本研究では，複数移動体の編隊制御としてLeader-Follower型を取り扱い，その編隊形状変化に伴う衝突回避問題に対し，モデル予測制御に基づく衝突回避手法を提案し，その有効性を実験により検証する．特に，モデル予測

Fig. 1.4:Segway RMP 200 (Segway Inc.)

制御に基づく手法で問題となる計算量や離散化の影響に着目し，それらを考慮した衝突回避手法を考える．まず，計算量の低減化に着目した衝突回避手法では，新たな衝突回避アルゴリズムを提案するとともに，最適制御問題の解法についても衝突回避問題の特性を考慮し改良することで，計算量低減化を図ることを考える．また，離散化が移動体同士の衝突回避に与える影響を明らかにし，予測時刻間での衝突回避を保証する制御手法を提案する．

## 1.5　本論文の構成

本論文は全5章で構成され，概要は以下の通りである

第2章では，制御対象と制御目的について述べる．制御対象として移動ロボットの移動機構に用いられることの多い2輪型車両を取り扱い(例えばFig. 1.4やFig. 1.5など)，Leader-Follower型の編隊制御手法に基づき目標の編隊状態を定義する．また，移動体同士の衝突回避や障害物回避が0,1変数を含む線形不等式制約で記述することができることを示す．また，非線形システムである2輪車両を取り扱うため，フィードバック線形化に基づいた制御則を提案し，線形化されたシステムに対してモデル予測制御を適用することで2輪車両の衝突回避を実現する制御方策を提案する．また，モデル予測制御により衝突回避問題が混合整数計画問題として定式化されることを示す．

Fig. 1.5:beego (Technocraft)

第3章では，モデル予測制御の基づく衝突回避を考慮した編隊制御手法を提案する．ここでは，計算量の低減化のため，各移動体が順番に独立した最適制御問題を解くアルゴリズムを提案する．また，その際の最適制御問題の可解性と閉ループ系の安定性についても述べる．次に，一般的な分枝限定法の問題点を明らかにし，衝突回避問題の特性を考慮した分枝限定法を提案する．提案する分枝限定法は衝突回避問題の特性を考慮した分枝ルールを適用するという簡単な改良ながらも計算量が低減化できるものである．それらの有効性をシミュレーションと実験により検証する．

第4章では，モデル予測制御による衝突回避のための最適制御問題を実装する際に生じる離散化の影響を陽に考慮した衝突回避手法を提案する．まず，問題の簡単化のために，1台の移動体が対象に障害物を回避しながら目標位置まで移動する静止障害物回避問題を取り扱い，障害物回避に離散化が与える影響を明らかにする．これに基づき，予測時刻間での障害物回避を考慮した2つの手法を提案する．次に，上述した1台の移動体の静止障害物回避問題で得られた知見を生かし，移動体同士の衝突回避問題に対して離散化の影響を陽に考慮した衝突回避手法を提案する．提案手法は移動体群が移動しながら動的に編隊形状を変化するような場合でも予測時刻間での衝突回避を実現できることを示し，実機によってもその有効性を確認する．

第5章では，各章で得られた結果をまとめ，モデル予測制御に基づく衝突回避手法の課題と今後の展望について述べる．

# 第2章

# 問題設定と制御方針

本章では，まず，本論文で取り扱う制御対象と目標の編隊状態の定義について述べる．本論文では，移動ロボットの形態として一般的に用いられることの多い2輪車両を制御対象とする．それら2輪車両群の編隊制御手法としてLeader-Follower型(e.g. [14]-[19])に基づくものを用いる．Leader-Follower型の編隊制御法では移動体群をLeaderとFollowerとに明確に分けるものである．Leaderとは外部から操作される移動体で，移動体群全体の目標値となる移動体である．一方，Folowerは，Leaderに一定の相対位置関係で自律的に追従する移動体である．Leaderに対してのみ目標軌道を与えるだけで，FollowerはLeaderに対して一定の位置関係で自律的に追従するため，移動体群全体として一定の幾何学的形状を維持したまま目標位置まで移動させることを可能とする手法である．したがって，制御系設計対象はFollowerであり，本研究では特に，編隊形状切り替えに伴うFollower同士の衝突回避を考えるものとする．このFollower間の衝突回避を取り扱うため，Follower同士の衝突回避条件を定義し，その衝突回避条件が0,1変数を含む不等式制約条件として記述可能であることを示す．また，環境中に存在する静止障害物も同様に0,1変数を含む不等式制約条件として記述可能で，同様に取り扱いが可能であることを示す.

次に，2輪車両の衝突回避を考慮するため，厳密なフィードバック線形化により移動体モデルを導出し，編隊形状の形成･維持における衝突回避をモデル予測制御により実現する制御方策を提案する．このとき，モデル予測制御では衝突回避制約を用いるとFollower間の衝突回避問題を一般的に混合整数計画問題として定式化可能であることも示す.

## 2.1 制御対象と制御目的

本論文では以下の$n$台の二輪車両型の移動体を考える．

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} \cos\theta_i & 0 \\ \sin\theta_i & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_i \\ \omega_i \end{bmatrix} \tag{2.1}$$

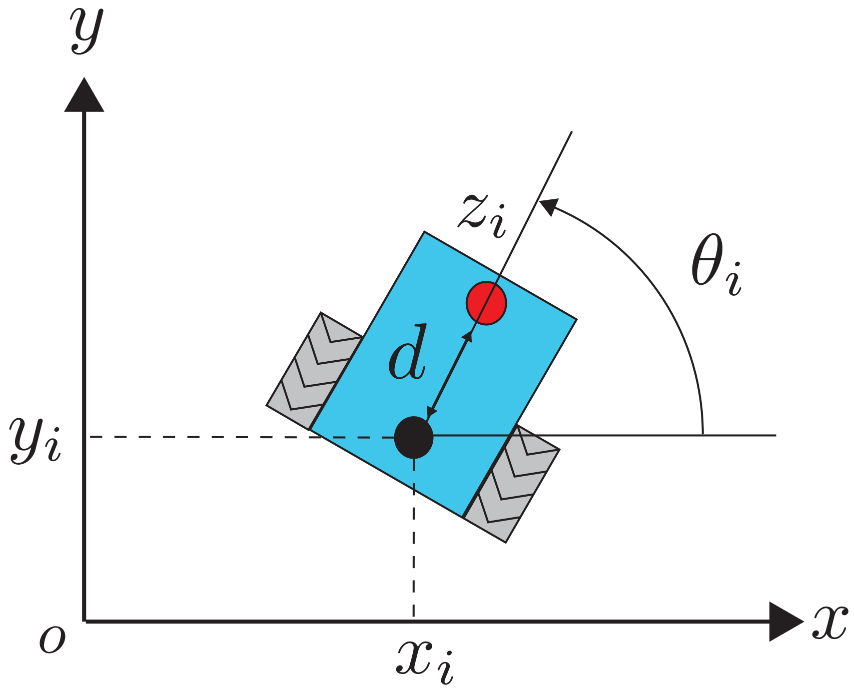



Fig. 2.1:Wheeled vehicle

ただし，$v_i$，$\omega_i$，$(x_i, y_i, \theta_i)$ はそれぞれ移動体 $i(= 1, \ldots, n)$ の並進速度，角速度，絶対座標系における位置・姿勢である (Fig. 2.1 参照)．また，編隊の基準となる以下のような2輪車両型の移動体を考える．

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} \cos\theta_r & 0 \\ \sin\theta_r & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} \tag{2.2}$$

ただし，$v_r$，$\omega_r$，$(x_r, y_r, \theta_r)$ はそれぞれ基準移動体の並進速度，角速度，絶対座標系における位置・姿勢である．以下では，(2.1)，(2.2) 式の2輪車両をそれぞれ，“ Follower $i$”, “ Leader”と呼ぶものとする．

Follower $i$ の目標位置は Fig. 2.2 のように Leader に固定した移動座標系 $(r, l)$ において $\zeta_i^d := (r_i^d, l_i^d)^T$ と表現される．すなわち Follower $i$ の目標軌道は絶対座標系において次式のように表される．

$$z_i^d := \begin{bmatrix} x_r + r_i^d \sin\theta_r + l_i^d \cos\theta_r \\ y_r - r_i^d \cos\theta_r + l_i^d \sin\theta_r \end{bmatrix} \tag{2.3}$$

本論文では，Follower 同士が衝突することなく，各々の Follower が

$$z_i := \begin{bmatrix} x_{vi} \\ y_{vi} \end{bmatrix} = \begin{bmatrix} x_i + d\cos\theta_i \\ y_i + d\sin\theta_i \end{bmatrix} \tag{2.4}$$

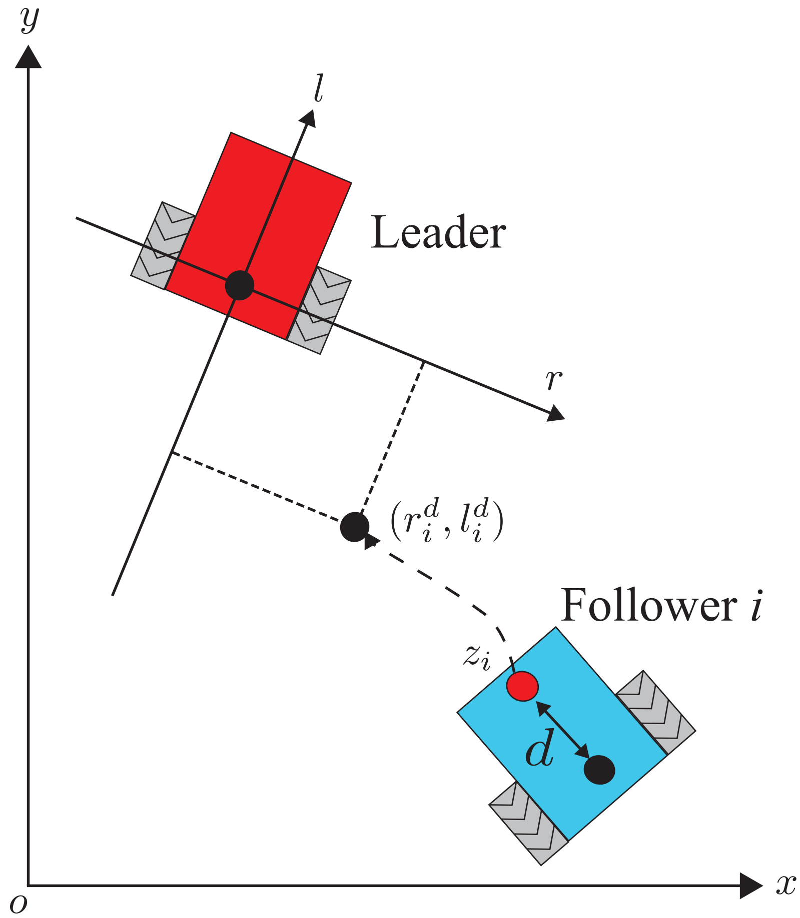



Fig. 2.2:Leader and follower

で表される制御点を目標軌道(2.3)に追従させることを制御目的とする．なお，移動体$i$の制御点を移動座標系において$\zeta_i := (r_i, l_i)^T$と表現するものとする．制御点(2.4)は$\zeta_i$を用いると以下のように表わすことができる．

$$z_i = \begin{bmatrix} x_r + r_i \sin\theta_r + l_i \cos\theta_r \\ y_r - r_i \cos\theta_r + l_i \sin\theta_r \end{bmatrix} \tag{2.5}$$

(2.3)式から分かるように，本研究では全てのFollowerがLeaderとの相対位置関係を直接制御する関係にある．より一般的には，参考文献[16]などのように近傍の移動体と相対位置関係を制御することが考えられるが，本研究では取り扱わないものとする．また，各Followerは他移動体に関する必要な情報を通信等により取得できるものとする．

## 2.2　衝突回避条件

ここではFollower同士の衝突回避条件を定義する．また，その衝突回避条件が0,1変数を含む線形不等式制約条件に変換することができることを示す．さらに，移動体同士の衝突回避と同様に環境中に存在する静止障害物との衝突回避条件も0,1変数を含む線形不等式制約条件に変換することができることを示す．なお，LeaderとFollowerとの衝突回避も同様に取り扱うことが可能であるが，参考文献[23]-[25]らと同様にLeaderは仮想的に与えることとし，考慮しないものとする．

### 2.2.1　衝突回避制約

Follower同士の衝突回避を考慮するため，Follower $i$と$j$が衝突しないための十分条件が二輪車両の大きさに基づき，次のように与えられているものとする(Fig. 2.3参照).

$$\|z_i - z_j\|_\infty \geq \psi, \quad \forall j \neq i \tag{2.6}$$

なお，(2.6)式の衝突回避制約は次の線形制約で表されることが知られている[63].

$$\begin{aligned} x_{vi} - x_{vj} &\leq M\kappa_{ij1} - \psi \\ -x_{vi} + x_{vj} &\leq M\kappa_{ij2} - \psi \\ y_{vi} - y_{vj} &\leq M\kappa_{ij3} - \psi \\ -y_{vi} + y_{vj} &\leq M\kappa_{ij4} - \psi \\ \sum_{p=1}^{4} \kappa_{ijp} &\leq 3 \end{aligned} \tag{2.7}$$

ただし，$\kappa_{ijp}$は$0, 1$のみをとる変数，Mは$z_i$のとりうる値よりも十分大きな正数である．なお，LeaderとFollowerの間の衝突も同様に考慮することが可能であるが，Leaderは仮想的に与えられることが多いため，本論文では考慮しないものとする．

### 2.2.2　障害物回避制約

環境中に存在する静止障害物も移動体同士の衝突回避と同様に記述することができる[64]．ここでは環境中に$m$個の障害物が存在すると仮定する．また，各障害物はそれを内包する長方形$O_j (j = 1, \ldots, m)$で近似され，その長方形の重心を$o_j := (x_{oj}, y_{oj})^T$，辺の長さを$(2\psi_{j1}, 2\psi_{j2})$と定義する(Fig. 2.4参照)．こ

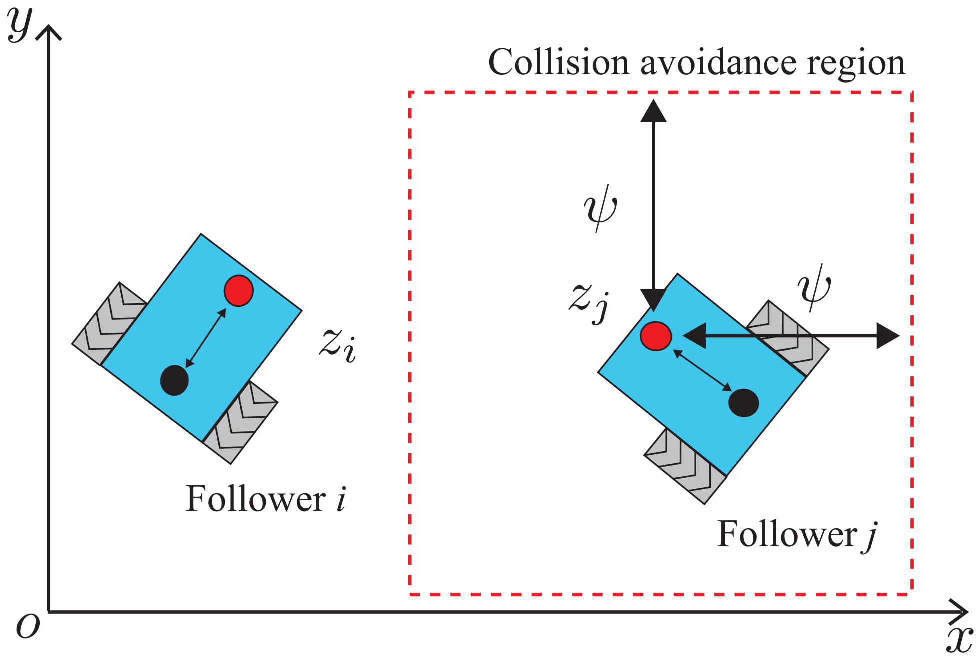



Fig. 2.3:Collision avoidance between follower $i$ and follower $j$

のとき，障害物 $j(=1,\ldots,m)$ と Follower $i$ が衝突しないための十分条件が以下のように与えられるものとする．

$$|x_{vi}-x_{oj}|\geq\psi_{j1}\quad\text{or}\quad|y_{vi}-y_{oj}|\geq\psi_{j2} \tag{2.8}$$

ただし，$\psi_{i1}$，$\psi_{i2}$ は障害物と移動体の大きさを考慮し適切に設定されているものとする [68]．(2.8) 式は次のようなベクトル形式で表すことができる．

$$\|D_{\psi_j}^{-1}(z_{vi}-o_j)\|_\infty\geq 1,\quad D_{\psi_j}:=\mathrm{diag}(\psi_{j1},\psi_{j2}) \tag{2.9}$$

さらに，長方形 $O_j$ が一定角度 $\phi_j$ 回転している場合には，以下のように表わすことができる．

$$\|D_{\psi_j}^{-1}R_{\phi_j}^{-1}(z_{vi}-o_j)\|_\infty\geq 1,\ R_{\phi_i}:=\begin{bmatrix}\cos\phi_j & -\sin\phi_j\\ \sin\phi_j & \cos\phi_j\end{bmatrix} \tag{2.10}$$

本論文で，静止障害物も取り扱う場合には，簡単化のため (2.9) 式の場合のみを用いることとする．

ここで，(2.8) 式は衝突回避制約 (2.7) と同様に 0, 1 変数を用いて以下のよう

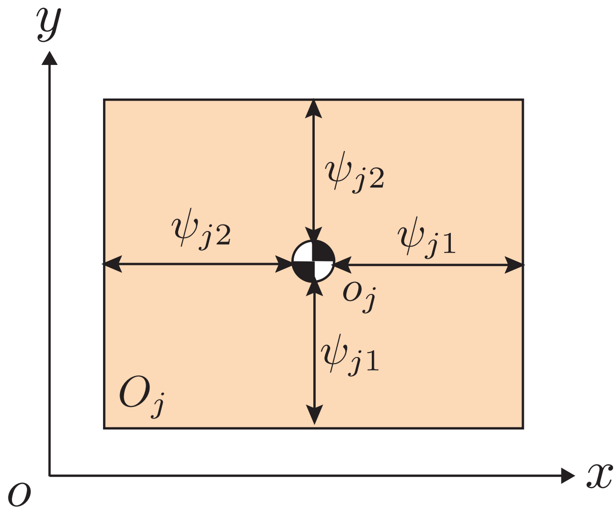



Fig. 2.4:Definition of the obstacles

に表現できる．

$$
\begin{aligned}
x_{vi} - x_{oj} &\le M\kappa_{ij1} - \psi_{j1} \\
-x_{vi} + x_{oj} &\le M\kappa_{ij2} - \psi_{j1} \\
y_{vi} - y_{oj} &\le M\kappa_{ij3} - \psi_{j2} \\
-y_{vi} + y_{oj} &\le M\kappa_{ij4} - \psi_{j2} \\
\sum_{p=1}^{4} \kappa_{ijp} &\le 3
\end{aligned}
\tag{2.11}
$$

ただし，$\kappa_{ijp}$ は $0, 1$ のみをとる変数，$M$ は左辺のとりうる値よりも十分大きな正数である．Fig. 2.5(i)-(iv) は (2.11) 式で $\kappa_{ijp} = 0$ $(p = 1, 2, 3, 4)$ となる場合に対応した領域を示したものである．また，幾何学的な条件から最大で2つの $\kappa_{ijp}$ が同時に0となる場合が存在する (Fig. 2.5(v)-(viii) 参照)．このように (2.11) 式では $\kappa_{ijp}$ の値により8つのモードが存在する．

## 2.3　制御方針

本論文では，Follower 同士の衝突回避を考慮するため，以下の制御方策を提案する．まず，フィードバック線形化により非線形システムを安定なシステムにする．次に，その線形システムに対してモデル予測制御を適用し，衝突回避制約 (2.7) を考慮した最適制御問題を解くことで，衝突回避しながら目標編隊を実現する制御入力を決定する．

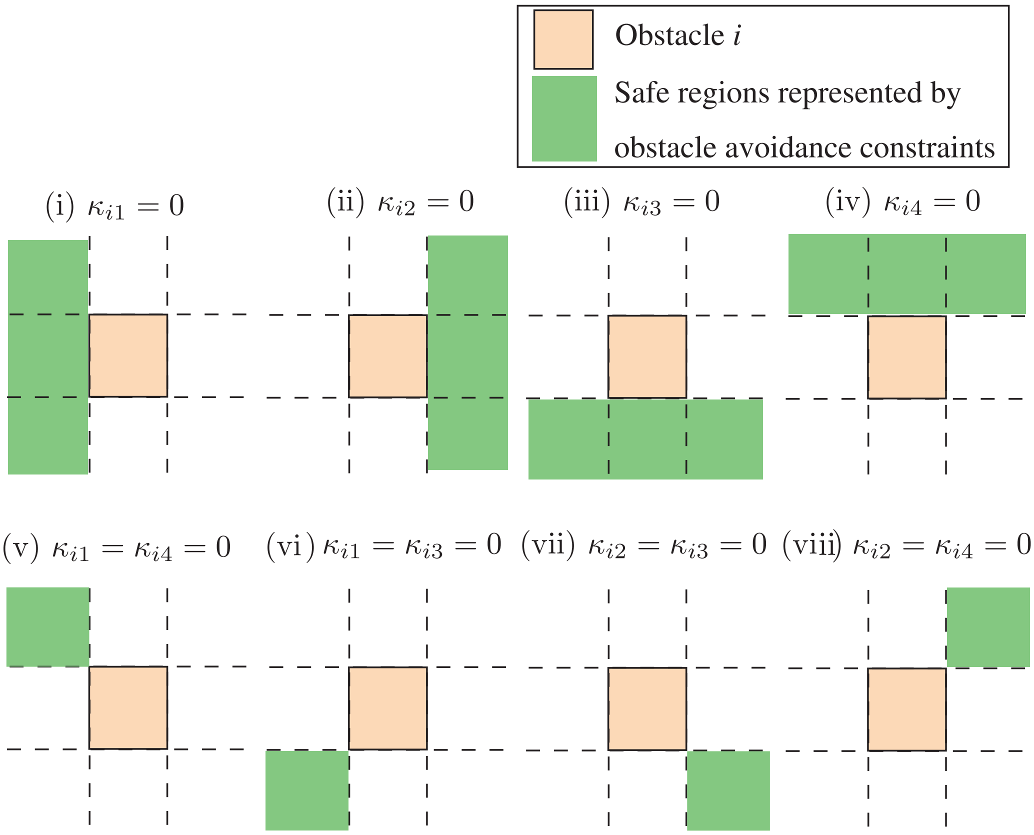



Fig. 2.5:Possible regions for the constraint (2.11)

### 2.3.1　フィードバック線形化

ここでは，フィードバック線形化によりシステムを線形化する．具体的には参考文献[16]の手法を改良し，Follower同士の衝突回避を考慮しやすいようにする．ここでは絶対座標系$(x, y)$と移動座標系$(r, l)$で定義した追従偏差$e_i := z_i - z_i^d$，$\tilde{e}_i := \zeta_i - \zeta_i^d$を用いたそれぞれの場合についてフィードバック線形化により，システムを安定化する制御則を導出する．この2つの制御則は取り扱う問題に応じて使い分けるものとする．たとえば，移動体同士の衝突回避のみを考慮する場合には移動座標系で表現すると，Leaderの有限時間未来先までの情報を必要としないため，より問題が取り扱いやすくなる．一方，環境中に存在する静止障害物回避も同時に考慮する場合には，絶対座標系での表現を用いる必要がある．障害物回避を考慮するには環境に固定した静止障害物と移動体との相対位置関係を記述できなくてはならないため絶対座標系での表現が必要となる．なお，2つの表現で収束の仕方に違いが出るが，どちらを用いた場合でも衝突が起きない限り指数的に目標位置まで

収束する．

### 絶対座標系 $(x, y)$ での表現

まず，絶対座標系で追従偏差を定義した場合の制御則を導出する．(2.3) 式の時間微分は，

$$\dot{z}_i^d = \begin{bmatrix} \dot{x}_r + r_i^d \dot{\theta}_r \cos\theta_r - l_i^d \dot{\theta}_r \sin\theta_r \\ \dot{y}_r + r_i^d \dot{\theta}_r \sin\theta_r + l_i^d \dot{\theta}_r \cos\theta_r \end{bmatrix} \tag{2.12}$$

となる．(2.2) 式より

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} \cos\theta_r & 0 \\ \sin\theta_r & 0 \end{bmatrix} \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} \tag{2.13}$$

であるので，(2.12) 式は

$$\begin{aligned} \dot{z}_i^d &= \begin{bmatrix} \dot{x}_r + r_i^d \omega_r \cos\theta_r - l_i^d \omega_r \sin\theta_r \\ \dot{y}_r + r_i^d \omega_r \sin\theta_r + l_i^d \omega_r \cos\theta_r \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta_r & r_i^d \cos\theta_r - l_i^d \sin\theta_r \\ \sin\theta_r & r_i^d \sin\theta_r + l_i^d \cos\theta_r \end{bmatrix} \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} \end{aligned} \tag{2.14}$$

となる．したがって，

$$F_i := \begin{bmatrix} \cos\theta_r & r_i^d \cos\theta_r - l_i^d \sin\theta_r \\ \sin\theta_r & r_i^d \sin\theta_r + l_i^d \cos\theta_r \end{bmatrix}, \quad u_r := \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} \tag{2.15}$$

を定義すると，

$$\dot{z}_i^d = F_i u_r \tag{2.16}$$

と表わせられる．

また，(2.4) 式の時間微分は，

$$\begin{bmatrix} \dot{x}_{vi} \\ \dot{y}_{vi} \end{bmatrix} = \begin{bmatrix} \dot{x}_i - d\dot{\theta}_i \sin\theta_i \\ \dot{y}_i + d\dot{\theta}_i \cos\theta_i \end{bmatrix} \tag{2.17}$$

となる．(2.1) 式より，

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} = \begin{bmatrix} \cos\theta_i & 0 \\ \sin\theta_i & 0 \end{bmatrix} \begin{bmatrix} v_i \\ \omega_i \end{bmatrix} \tag{2.18}$$

となるので，(2.17) 式は

$$\begin{aligned} \begin{bmatrix} \dot{x}_{vi} \\ \dot{y}_{vi} \end{bmatrix} &= \begin{bmatrix} \dot{x}_i - d\omega_i \sin\theta_i \\ \dot{y}_i + d\omega_i \cos\theta_i \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta_i & -d\sin\theta_i \\ \sin\theta_i & d\cos\theta_i \end{bmatrix} \begin{bmatrix} v_i \\ \omega_i \end{bmatrix} \end{aligned} \tag{2.19}$$

となる．したがって，

$$G_i := \begin{bmatrix} \cos\theta_i & -d\sin\theta_i \\ \sin\theta_i & d\cos\theta_i \end{bmatrix}, \quad u_i := \begin{bmatrix} v_i \\ \omega_i \end{bmatrix} \tag{2.20}$$

を定義すると，

$$\dot{z}_i = G_i u_i \tag{2.21}$$

と表わせられる．

(2.16), (2.21) 式より，追従偏差 $e_i := z_i - z_i^d$ の時間微分は次式で表される．

$$\dot{e}_i = G_i u_i - F_i u_r \tag{2.22}$$

ここで，$\alpha_i$ をシステムの新たな入力として Follower $i$ の制御入力 $u_i$ を

$$u_i = G_i^{-1}(-\lambda e_i + F_i u_r + \alpha_i) \tag{2.23}$$

とすると，(2.22), (2.23) 式より次式を得る．

$$\dot{e}_i = -\lambda e_i + \alpha_i \tag{2.24}$$

ただし $\lambda > 0$ は設計パラメータである．なお，$G_i$ は

$$\det G_i = d\cos^2\theta_i + d\sin^2\theta_i = d \tag{2.25}$$

であるので $d > 0$ ならば常に可逆である．

(2.24) 式の微分方程式の解は $\alpha_i = 0$ の場合には

$$e_i(t) = \exp(-\lambda t)e_i(0) \tag{2.26}$$

となり，$e_i$ が指数的に 0 に収束することを示している．ただし，この式では他 Follower との衝突の可能性を考慮していないので，他の Follower が接近している場合にはこの関係式は成り立たなくなる．

なお，制御入力 (2.23) の $G^{-1}F_i u_r$ は Leader に関するフィードフォワード項であり，これにより絶対座標系 $(x, y)$ での移動体間の相対位置関係が維持される．また，$\alpha_i$ がモデル予測制御で決定される衝突回避のための入力である．

### 移動座標系 $(r, l)$ での表現

次に追従偏差を移動座標系で定義した場合のシステムを安定化する制御則をフィードバック線形化により導出する．

ここでは移動座標系 $(r, l)$ における制御点 $\zeta_i$ に対し，新たな制御入力 $\tilde{\alpha}_i := (\tilde{\alpha}_r, \tilde{\alpha}_l)^T$ を与えることを考える．また，フィードバック項 $-\lambda\tilde{e}_i$ も加えることを

考える．このとき，移動座標系において追従偏差 $\tilde{e}_i := \zeta_i - \zeta_i^d$ の時間微分は次式で表わされる．

$$\dot{\tilde{e}}_i = -\lambda\tilde{e}_i + \tilde{\alpha}_i \tag{2.27}$$

次に (2.27) 式を実現するための制御入力 $u_i$ をフィードバック線形化により求める．

まず，(2.5) 式の時間微分は，

$$\begin{aligned} \dot{z}_i &= \begin{bmatrix} \dot{x}_r + r_i\dot{\theta}_r\cos\theta_r - l_i\dot{\theta}_r\sin\theta_r + \dot{r}_i\sin\theta_r + \dot{l}_i\cos\theta_r \\ \dot{y}_r + r_i\dot{\theta}_r\sin\theta_r + l_i\dot{\theta}_r\cos\theta_r - \dot{r}_i\cos\theta_r + \dot{l}_i\sin\theta_r \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta_r & r_i\cos\theta_r - l_i\sin\theta_r \\ \sin\theta_r & r_i\sin\theta_r + l_i\cos\theta_r \end{bmatrix}\begin{bmatrix} v_r \\ \omega_r \end{bmatrix} + \begin{bmatrix} \sin\theta_r & \cos\theta_r \\ -\cos\theta_r & \sin\theta_r \end{bmatrix}\begin{bmatrix} \dot{r}_i \\ \dot{l}_i \end{bmatrix} \end{aligned} \tag{2.28}$$

となる．ここで，

$$\tilde{E}_i := \begin{bmatrix} \cos\theta_r & r_i\cos\theta_r - l_i\sin\theta_r \\ \sin\theta_r & r_i\sin\theta_r + l_i\cos\theta_r \end{bmatrix},\ \tilde{F}_i := \begin{bmatrix} \sin\theta_r & \cos\theta_r \\ -\cos\theta_r & \sin\theta_r \end{bmatrix} \tag{2.29}$$

とおくと，(2.27)，(2.28) 式より，

$$\dot{z}_i = \tilde{E}_i u_r + \tilde{F}_i(-\lambda\tilde{e}_i + \tilde{\alpha}_i) \tag{2.30}$$

と表わせる．

ここで (2.30)，(2.21) 式より，Follower $i$ の制御入力は

$$u_i = G_i^{-1}(\tilde{E}_i u_r + \tilde{F}_i(-\lambda\tilde{e}_i + \tilde{\alpha}_i)) \tag{2.31}$$

と求めることができる．(2.27) 式より (2.26) 式と同様に $\tilde{\alpha}_i$ が 0 であり，かつ Follower 同士の衝突が起きない場合には $\tilde{e}_i$ は指数的に 0 に収束する．

制御入力 (2.31) の $G_i^{-1}\tilde{E}_i u_r$ 項は Leader の挙動に対するフィードフォワード項であり，これにより移動座標系 $(r, l)$ での移動体間の相対位置関係が維持される．また，$\tilde{\alpha}_i$ がモデル予測制御で決定される衝突回避のための入力である．

### 2.3.2 モデル予測制御に基づく一般的な衝突回避手法

次に線形化されたシステム ((2.24) 式もしくは (2.27) 式) に対してモデル予測制御を適用することで Follower 同士の衝突回避を実現する制御入力を計画する．例えば，線形化システム (2.24) が与えられた時，Follower 同士の衝突回避を実現する入力 $\alpha_i$ は一般的に以下のような最適制御問題 $P_0$ を解くことで決定することができる (e.g. [63])．ただし，以下の最適制御問題 $P_0$ は時刻 $t = k\delta$ において解かれるものとする．

衝突回避のための最適制御問題 $P_0$

$$\min_{\hat{\alpha}_1,\ldots,\hat{\alpha}_n} \sum_{i}^{n} \int_{t}^{t+T} \widehat{e_i}(\tau|k)^T Q\widehat{e_i}(\tau|k) + \widehat{\alpha}_i(\tau|k)^T R\widehat{\alpha}_i(\tau|k) d\tau \tag{2.32}$$

subject to

$$\dot{\widehat{e_i}} = -\lambda\widehat{e_i} + \widehat{\alpha}_i, \quad \widehat{e_i}(t|k) = e_i(t) \tag{2.33}$$

$$\|\widehat{e_i}(\tau|k) - \widehat{e}_j(\tau|k) + \mu_i(\tau|k) - \mu_j(\tau|k)\|_\infty \geq \psi \tag{2.34}$$

$$i = 1, \ldots, n, \ \forall j > i, \quad t \leq \tau \leq t + T$$

ただし

$$\mu_i(\tau|k) := \begin{bmatrix} \sin\theta_r(\tau|k) & \cos\theta_r(\tau|k) \\ -\cos\theta_r(\tau|k) & \sin\theta_r(\tau|k) \end{bmatrix} \begin{bmatrix} r_i^d \\ l_i^d \end{bmatrix} \tag{2.35}$$

である．$*(\tau|k)$ は時刻 $t = k\delta$ における $*(\tau\delta)$ の予測値を表わし，$*(k|k) := *(k\delta)$ である．また，等式制約 (2.33) は (2.24) 式で表わされる $e_i$ の予測モデルで，不等式制約 (2.34) は衝突回避制約であり，(2.7) 式の形式に変換される．したがって，不等式制約 (2.34) には 0,1 変数が含まれることになり，最適制御問題 $P_0$ は混合整数二次計画問題となる．なお，衝突回避だけでなく 2.2.2 節で示したような静止障害物も同時に考慮する場合には

$$\|D_{\psi_j}(\widehat{e_i}(\tau|k) - o_j + \mu_i(\tau|k))\|_\infty \geq 1, \ \ j = 1, \ldots, m \tag{2.36}$$

を課すことで実現することができる．モデル予測制御では Fig. 1.3 に示した陽に上記の最適制御問題を更新周期 $\delta$ 毎に解き，その最初の入力 $\widehat{\alpha}_i(k|k)$ を印加することを繰り返し行うことでフィードバック効果を得る．したがって，オンラインで実装するには少なくとも最適制御問題 $P_0$ は $\delta$ 以下で計算できなくてならない．

本論文で着目するのは，このような衝突回避のための最適制御問題であり，3 章では計算量の低減化の観点から衝突回避のための最適制御問題とその一般的な解法である分枝限定法を改良することを考える．4 章では，最適制御問題の“離散化”に着目し，離散化が与える衝突回避への影響を考慮した上で，予測時刻間での衝突回避を保証する最適制御問題について考える．

# 第3章
# 衝突回避問題に適合した分枝限定法に基づくモデル予測編隊制御

本章では計算量低減化の観点から，モデル予測制御に基づく新たな衝突回避のための最適制御問題を提案する．提案する手法は2.3.2節で示した最適制御問題とは異なり，各Followerが順番に独立した最適制御問題を解くものである．これにより各更新時刻で解く最適制御問題のサイズを小さく抑え，計算量の低減化を図る．また，その最適制御問題の可解性と目標の編隊への収束性についても解析する．次に，混合整数計画問題の一般的な解法である分枝限定法[70]の問題点をあげ，衝突回避問題に適合した新たな分枝限定法を提案し，シミュレーションにより計算量が低減化できることを示す．これらにより，従来は困難であった複雑な衝突回避が必要となるような編隊形状の切り替えについても実験で検証する．

## 3.1 衝突回避を考慮した編隊制御手法

ここでは2.2.1節で述べた衝突回避制約(2.7)に基づいて制御入力を決定する制御手法について述べる．2.3.2節で示したような，すべてのFollowerが入力 $\alpha_i$ $(i = 1, \ldots, n)$ を一つの最適化問題で同時に決定するのは計算量の面から実用的でない．そこで，Follower毎に独立な最適化問題に分割し求める方法を考える．ただし，すべてのFollowerが独立した最適化問題で同時に制御入力を決定すると，衝突回避を理論的に保証するのが困難であるため，各Followerは更新周期 $\delta$ ずつ時間をずらして順番に最適制御問題を解くことにする．また，本研究では2輪車両を線形化して，閉ループ系に対してモデル予測制御を適用している．したがってモデル予測制御で決定する入力 $\alpha_i$ と元々のシステムへの入力 $u_i$ との関係を求め，元々の入力が飽和しないような条件を導出する．これの条件をもとに制約を課すことで，制御入力の飽和を回避しながら衝突回避実現する手法を提案する．

### 3.1.1　提案する衝突回避手法

本節で提案する手法では，各々のFollowerが更新周期$\delta$ごとに順番に最適化問題を解き，衝突回避を行うものである．具体的にはFollower $i$は時刻$t = k_i\delta(k_i := sn + i - 1\ (s = 0, 1, \ldots))$において最適制御問題を解き，得られた最適制御軌道$\alpha_i^*$を次の更新時刻$t = (k_i + n)\delta$まで適用する．時刻$t = k_i\delta$における$e_i(\tau)$の予測値$\widehat{e_i}(\tau|k_i)$は計算が終わり次第すぐに他のすべてのFollowerに送信される．Follower $i$はFollower $j(j \neq i)$より送信された予測軌道$\widehat{e_j}$を用い，衝突回避を行う．なお，各Followerは，他のFollowerの位置などの必要な情報を取得することができるものとする．また，Leaderの有限時間未来先までの軌道$\theta_r(\tau)(t \leq \tau \leq t + T)$を取得することができるものとする．

Follower $i$の衝突回避アルゴリズムを次に示す.

**Step 0:** 初期時刻$t = 0$において$k := 0$および，

$$\widehat{\alpha_i}(\tau|0) := \lambda e_i(0), \quad 0 \leq \tau \leq T \tag{3.1}$$

$$\widehat{e_j}(\tau|0) := e_j(0), \quad 0 \leq \tau \leq T, \ \ j \neq i \tag{3.2}$$

とする．

**Step 1:** 時刻$t = k\delta$において,

・$k = i - 1\ (\mathrm{mod}\ n)$の場合, 後述の最適制御問題を解き,

$$\widehat{\alpha_i}(\tau|k) = \alpha_i^*(\tau|k) \tag{3.3}$$

$$\widehat{e_i}(\tau|k) = e_i^*(\tau|k), \ \ t \leq \tau \leq t + T \tag{3.4}$$

の更新を行い，$\widehat{e_i}(\tau|k)$を他のFollowerに送信する．ただし，$\alpha_i^*(\tau|k)$と$e_i^*(\tau|k)$は後述の最適制御問題により得られた最適軌道である．

・その他の場合, 時刻$p = k\ (\mathrm{mod}\ n)$におけるFollower $p$の$\widehat{e_p}(\tau|k)$を受信する．

**Step 2:** 時刻$t \leq \tau \leq t + \delta$において

$$\alpha_i(\tau) = \widehat{\alpha_i}(\tau|k) \tag{3.5}$$

を用い，(2.23)式の$u_i(\tau)$を適用する．

**Step 3:** $k = k + 1$ および，

$$\widehat{\alpha}_i(\tau|k) = \widehat{\alpha}_i(\tau|k-1) \tag{3.6}$$

$$\widehat{e}_j(\tau|k) = \widehat{e}_j(\tau|k-1), \quad t+\delta \le \tau \le t+T \tag{3.7}$$

$$\widehat{e}_j(\tau|k) = \exp(\lambda(t+T-\tau))\,\widehat{e}_j(t+T|k-1), \quad t+T \le \tau \le t+T+\delta \tag{3.8}$$

と更新し，Step 1 に続く．

ただし，Follower $j$ が $t+T \le \tau \le t+T+\delta$ における $\alpha_j(\tau)$ の情報を持っていないので，

$$\alpha_j(\tau) = 0, \quad t+T \le \tau \le t+T+\delta \tag{3.9}$$

と仮定することにより，(3.8) 式で用いる $e_j$ の予測値を計算する．また，ホライズン $T$ は衝突回避を行うため，$T \ge n\delta$ を満たすように選ぶ必要がある．なお，各 Follower の最適化及び入力適用のタイミングを Fig. 3.1 に示す．●印は最適化を行う時刻を，矢印は最適化により得られた入力を適用する時間を示したものである．

時刻 $t = k\delta(k = i-1) \pmod n$ における Follower $i$ の最適制御問題は次の混合整数二次計画問題 $P_1$ で表される．

衝突回避のための最適制御問題 $P_1$

$$\min_{\hat{\alpha}_i} \int_t^{t+T} \widehat{\alpha}_i(\tau|k)^T R \widehat{\alpha}_i(\tau|k) d\tau \tag{3.10}$$

subject to

$$\dot{\widehat{e}}_i = -\lambda \widehat{e}_i + \widehat{\alpha}_i, \quad \widehat{e}_i(t|k) = e_i(t) \tag{3.11}$$

$$\|\widehat{e}_i(\tau|k) - \widehat{e}_j(\tau|k) + \mu_i(\tau|k) - \mu_j(\tau|k)\|_\infty \ge \psi \tag{3.12}$$

$$\| - \lambda \widehat{e}_i(\tau|k) + \widehat{\alpha}_i(\tau|k)\|_\infty \le \eta \tag{3.13}$$

$$\|\widehat{e}_i(t+T|k)\|_\infty \le \gamma_i \tag{3.14}$$

$$\forall j \ne i, \quad t \le \tau \le t+T$$

であり，$R$ は任意の正定行列である．また，(3.11) 式の等式制約は (2.24) 式で表される $e_i$ の予測モデルである．(3.12) 式は (2.6) 式で示した衝突回避制約で，(2.7) 式で表される形式に変換される．(3.13) 式は (2.23) 式の制御入力 $u_i$ を制限するため導入した入力制約で，$\psi$, $\eta$ は設計者が定める正数である．(3.14) 式は閉ループ系の安定性を保証するために導入した終端制約である．なお，最適制御問題は絶対座標系で表現しているため，静止障害物回避も同時に

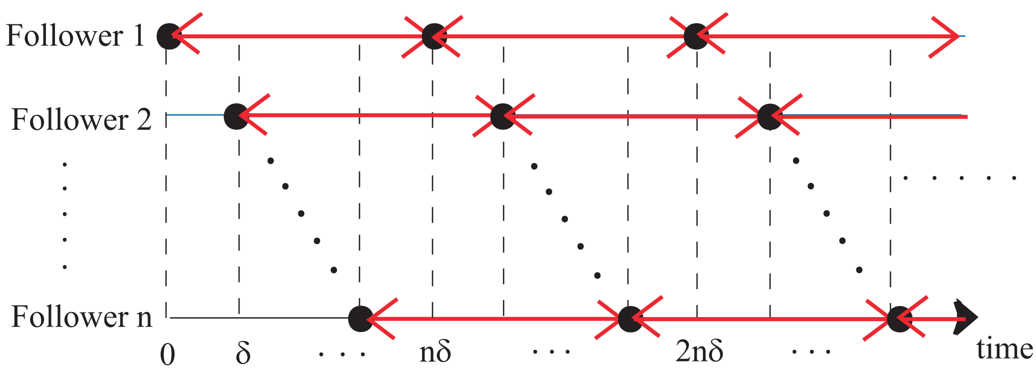



Fig. 3.1:Timing of optimization and input application. A mark • represents time when an optimization problem is solved, and the arrows represent the periods when optimal control sequences are applied

考慮することが可能である．その場合には(2.36)式と同様に

$$\|\widehat{e_i}(\tau|k) - o_i + \mu_i(\tau|k)\|_\infty \geq \psi \tag{3.15}$$

を課せばよい．ただし，本章では問題の簡単化のため，移動体同士の衝突回避のみ考慮することとする．

**注意1** 現実の多くのシステムでは，入力制約は(3.13)式の形ではなく，例えば

$$\|u_i(t)\|_\infty \leq \eta_u \quad (\eta_u \text{ は正数}) \tag{3.16}$$

のように，$u_i$の大きさが制限される形で与えられる．したがって，このようなシステムに対して本手法が正常に動作するためには，$u_i$に対して与えられた制約を大きく破らないように(3.13)式の$\eta$が設定される必要がある．特に，$\omega_r = \dot{\theta}_r = 0$の場合，すなわち隊形変化の間にLeaderの姿勢が変化しない場合，$\|u_i(t)\|_\infty$は

$$\begin{aligned}
&\|u_i(t)\|_\infty \\
&= \left\| \begin{bmatrix} \cos\theta_i \sin\theta_i \\ -\frac{1}{d}\sin\theta_i \frac{1}{d}\cos\theta_i \end{bmatrix} \begin{bmatrix} -\lambda e_{ix} + \alpha_{ix} + v_r\cos\theta_r \\ -\lambda e_{iy} + \alpha_{iy} + v_r\sin\theta_r \end{bmatrix} \right\|_\infty \\
&= \left\| \begin{bmatrix} \cos\theta_i(-\lambda e_{ix} + \alpha_{ix} + v_r\cos\theta_r) + \sin\theta_i(-\lambda e_{iy} + \alpha_{iy} + v_r\sin\theta_r) \\ -\frac{1}{d}\sin\theta_i(-\lambda e_{ix} + \alpha_{ix} + v_r\cos\theta_r) + \frac{1}{d}\cos\theta_i(-\lambda e_{iy} + \alpha_{iy} + v_r\sin\theta_r) \end{bmatrix} \right\|_\infty
\end{aligned} \tag{3.17}$$

のように表わされる．ここで，(3.17) 式，$(-\lambda e_{ix} + \alpha_{ix}) \leq \eta$，$(-\lambda e_{iy} + \alpha_{iy}) \leq \eta$ より

$$
\begin{aligned}
&\|u_i(t)\|_\infty \\
&\leq \max\{|\eta\cos\theta_i + v_r\cos\theta_i\cos\theta_r + \eta\sin\theta_i + v_r\sin\theta_i\sin\theta_r|, \\
&\qquad |-\frac{\eta}{d}\sin\theta_i - \frac{v_r}{d}\sin\theta_i\cos\theta_r + \frac{\eta}{d}\cos\theta_i + \frac{v_r}{d}\cos\theta_i\sin\theta_r|\} \\
&= \max\{|\eta(\cos\theta_i + \sin\theta_i) + v_r(\cos\theta_i\cos\theta_r + \sin\theta_i\sin\theta_r)|, \\
&\qquad |\frac{\eta}{d}(\cos\theta_i - \sin\theta_i) + \frac{v_r}{d}(\cos\theta_i\sin\theta_r - \sin\theta_i\cos\theta_r)|\} \\
&= \max\{|\sqrt{2}\sin(\theta_i + \frac{\pi}{4}) + v_r\cos(\theta_i - \theta_r)|, \\
&\qquad |\frac{\sqrt{2}\eta}{d}\cos(\theta_i + \frac{\pi}{4}) + \frac{v_r}{d}\sin(\theta_r - \theta_i)|\} \\
&\leq \max\{|\sqrt{2}\eta\sin(\theta_i + \frac{\pi}{4})| + |v_r\cos(\theta_i - \theta_r)|, \\
&\qquad |\frac{\sqrt{2}\eta}{d}\cos(\theta_i + \frac{\pi}{4})| + |\frac{v_r}{d}\sin(\theta_r - \theta_i)|\} \\
&\leq \max\{\sqrt{2}\eta + |v_r|, \frac{\sqrt{2}\eta}{d} + |\frac{v_r}{d}|\}
\end{aligned}
\tag{3.18}
$$

となり，$\eta$ を決める際の一つの目安となる．なお，(3.18) 式は $\theta_i$ の値が取り得る全ての値を考慮して見積もられた上界であるので，$\theta_i$ の値の範囲が限定できれば，より保守性の低い上界を見積もることが可能である．

### 3.1.2　可解性と収束性

ここでは前節で提案した手法について，その最適制御問題の可解性と閉ループ系の安定性を示す．そのため，まず，次の仮定が満たされるものとする．

**仮定 1** 終端集合 (3.14) の大きさ $\gamma_i$ $(i = 1, \ldots, n)$ およびホライズン $T$ が次の条件を満たすものとする．

$$\left\| \begin{matrix} r_i^d - r_j^d \\ l_i^d - l_j^d \end{matrix} \right\|_\infty \geq \sqrt{2}(\psi + \gamma_i + \gamma_j), \quad \forall j \neq i \tag{3.19}$$

$$\lambda\gamma_i \leq \eta, \quad T \geq n\delta \tag{3.20}$$

次に，以下の補題を導入する．

**補題 1** $\|e_i\|_\infty \leq \gamma_i$ および $\|e_j\|_\infty \leq \gamma_j$ $(i \neq j)$ が仮定 1 を満たすものとする．このとき，$\alpha_i = 0$ および $\theta_r \in [0, 2\pi]$ において次式が満たされる．

$$\|e_i - e_j + \mu_i - \mu_j\|_\infty \geq \psi \tag{3.21}$$

$$\|-\lambda e_i + \alpha_i\|_\infty \leq \eta \tag{3.22}$$

**証明：** まず，(3.19)式より，

$$\frac{1}{\sqrt{2}}\left\| \begin{array}{c} r_i^d - r_j^d \\ l_i^d - l_j^d \end{array} \right\|_\infty - \gamma_i - \gamma_j \geq \psi, \quad \forall j \neq i \tag{3.23}$$

である．これに $\|e_i\|_\infty \leq \gamma_i$ と $\|e_j\|_\infty \leq \gamma_j$ を用いると，

$$\frac{1}{\sqrt{2}}\left\| \begin{array}{c} r_i^d - r_j^d \\ l_i^d - l_j^d \end{array} \right\|_\infty - \|e_i\|_\infty - \|e_j\|_\infty \geq \psi, \quad \forall j \neq i \tag{3.24}$$

である．ここで，(2.35)式より，

$$\mu_i - \mu_j = \begin{bmatrix} \sin\theta_r & \cos\theta_r \\ -\cos\theta_r & \sin\theta_r \end{bmatrix} \begin{bmatrix} r_i^d - r_j^d \\ l_i^d - l_j^d \end{bmatrix} \tag{3.25}$$

であるので，

$$\begin{aligned} \|\mu_i - \mu_j\|_\infty &\geq \frac{1}{\sqrt{2}}\|\mu_i - \mu_j\|_2 \\ &= \frac{1}{\sqrt{2}}\left\| \begin{array}{c} r_i^d - r_j^d \\ l_i^d - l_j^d \end{array} \right\|_2 \geq \frac{1}{\sqrt{2}}\left\| \begin{array}{c} r_i^d - r_j^d \\ l_i^d - l_j^d \end{array} \right\|_\infty \end{aligned} \tag{3.26}$$

が得られる．(3.24)，(3.26)式より，

$$\begin{aligned} \psi &\leq \|\mu_i - \mu_j\|_\infty - \|e_i\|_\infty - \|e_j\|_\infty \\ &\leq \|\mu_i - \mu_j + e_i - e_j\|_\infty \end{aligned} \tag{3.27}$$

となり，(3.21)式が示された．また，(3.20)式より，

$$\| - \lambda e_i + \alpha_i\|_\infty \leq \lambda\|e_i\|_\infty \leq \lambda\gamma_i \leq \eta \tag{3.28}$$

となり，$\alpha_i = 0$ に対して(3.22)式が示された． □

最適制御問題(3.10)の可解性について次の定理を得る．

**定理1** 最適制御問題(3.10)が仮定1を満たし，かつ，各Follower $i$ ($i = 1, \ldots, n$) の初期の更新時刻 $t = (i-1)\delta$ において可解であるとする．このとき $t \geq n\delta$ におけるすべての最適制御問題(3.10)は可解となる．

**証明：** 数学的帰納法を用いて証明を行なう．まず，Follower $i$ ($= 1, \ldots, n$) は時刻 $t = k_i\delta \quad (k_i = sn + i - 1)$ において最適解を持つと仮定する．ただし $s$ は非負の整数である．3.1.1節の衝突回避アルゴリズムより Follower $i$($= 2, \ldots, n$) は $k_i\delta \leq \tau \leq k_i\delta + T$ において Follower 1 が

$$\widehat{\alpha}_1(\tau|k_i) = \begin{cases} \alpha_1^*(\tau|k_1), & k_i\delta \leq \tau \leq k_1\delta + T \\ 0, & k_1\delta + T < \tau \leq k_i\delta + T \end{cases} \tag{3.29}$$

を適用するものとし，入力 $\alpha_i^*(\tau|k_i)$ を決定する．従って，Follower $i(=2,\ldots,n)$ は Follower 1 と衝突することなく，$t=k_i\delta+T$ で終端集合

$$\Omega_i := \{z_i:\ \|z_i - z_i^d\|_\infty \le \gamma_i\} \tag{3.30}$$

に入る．また，$t=(k_1+n)\delta$ において Follower 1 が

$$\begin{aligned}&\widehat{\alpha}_1(\tau|k_1+n)\\ &=\begin{cases}\alpha_1^*(\tau|k_1), & (k_1+n)\delta \le \tau \le k_1\delta+T\\ 0, & k_1\delta+T<\tau\le(k_1+n)\delta+T\end{cases}\end{aligned} \tag{3.31}$$

を適用すれば，$(k_1+n)\delta \le \tau \le k_i\delta+T$ において Follower 1 は他の Follower と衝突することはない．また，補題1より，衝突回避制約 (3.12) は，$k_i\delta+T \le \tau \le (k_1+n)\delta+T$ においても Follower 1 と $i$ が各々の終端集合内にいるため，満たされる．従って，Follower 1 は $t=(k_1+n)\delta$ において実行可能解を少なくとも1つ持つことが示される．このときの (3.31) 式の入力 $\widehat{\alpha}_1$ は，補題1より，入力制約 (3.13) を満たすことが示される．

同様に，$t=(k_i+n)\delta$ において Follower $i(=2,\ldots,n)$ が

$$\begin{aligned}&\widehat{\alpha}_i(\tau|k_i+n)\\ &=\begin{cases}\alpha_i^*(\tau|k_i), & (k_i+n)\delta \le \tau \le k_i\delta+T\\ 0, & k_i\delta+T<\tau\le(k_i+n)\delta+T\end{cases}\end{aligned} \tag{3.32}$$

を適用すれば，$(k_i+n)\delta \le \tau \le k_j\delta+T$ において Follower $j(=i+1,\ldots,n)$ と衝突することはない．また，Follower $i$，$j$ は $k_j\delta+T<\tau\le(k_i+n)\delta+T$ において各々の終端集合内にいるため，Follower $i,j$ は衝突はすることはない．従って，すべての Follower $i\,(=1,\ldots,n)$ が $k_i=sn+i-1$ において最適解を持つならば，$k_i=(s+1)n+i-1$ において，すべての Follower は実行可能解を持つ．以上より，$s=0$ での可解性が示されれば，数学的帰納法により証明される．　□

また，閉ループ系の安定性について次の定理を得る．

**定理2** 定理1の条件が満たされる時，すべての Follower は衝突を生じず，目標位置に収束する．すなわち，

$$\lim_{t\to\infty} e_i(t)=0,\quad i=1,\ldots,n \tag{3.33}$$

**証明：** まず，評価関数 $J_i^*(s)$ が $s$ について非増加関数であることを示す.

$$J_i^*(s) := \int_{k_i\delta}^{k_i\delta+T} \alpha_i^*(\tau|k_i)^T R\alpha_i^*(\tau|k_i)d\tau \tag{3.34}$$

更新時刻$t = (k_i + n)\delta$において(3.31), (3.32)式の実行可能解

$$\overline{J}_i(s+1) := \int_{(k_i+n)\delta}^{(k_i+n)\delta+T} \widehat{\alpha}_i(\tau|k_i+n)^T R \widehat{\alpha}_i(\tau|k_i+n)d\tau \tag{3.35}$$

は次式を満たす.

$$\begin{aligned}
&\overline{J}_i(s+1) - J_i^*(s) \\
&= \int_{(k_i+n)\delta}^{(k_i+n)\delta+T} \widehat{\alpha}_i(\tau|k_i+n)^T R \widehat{\alpha}_i(\tau|k_i+n)d\tau \\
&\quad - \int_{k_i\delta}^{k_i\delta+T} \alpha_i^*(\tau|k_i)^T R \alpha_i^*(\tau|k_i)d\tau \\
&= -\int_{k_i\delta}^{(k_i+n)\delta} \alpha_i^*(\tau|k_i)^T R \alpha_i^*(\tau|k_i)d\tau \leq 0.
\end{aligned} \tag{3.36}$$

また, $\alpha_i^*(\cdot|k_i+n)$の最適性より

$$J_i^*(s+1) \leq \overline{J}_i(s+1) \tag{3.37}$$

が満たされる. (3.36), (3.37)式より,

$$J_i^*(s+1) \leq J_i^*(s)$$

が成り立ち, 評価関数は非増加であることが示された. 評価関数は非増加で, 下限が0で抑えられるため, ある定数$c \geq 0$に対して$s \to \infty$で$J_i^*(s) \to c$が満たされる. 従って, 任意の$\epsilon > 0$に対して

$$0 \leq J_i^*(s) - J_i^*(s+1) < \epsilon, \quad \forall s \geq s_1 \tag{3.38}$$

を満たす$s_1 > 0$が存在する.

一方, (3.36), (3.37)式より,

$$\begin{aligned}
&\int_t^{t+\delta} \alpha_i^*(\tau|k)^T R \alpha_i^*(\tau|k)d\tau \\
&= J_i^*(s) - \overline{J}_i(s+1) \leq J_i^*(s) - J_i^*(s+1)
\end{aligned} \tag{3.39}$$

また, $R > 0$, (3.38), (3.39)式より,

$$\lim_{s\to\infty} \alpha_i^*(\tau|k_i) \to 0, \quad k_i\delta \leq \tau < (k_i+n)\delta \tag{3.40}$$

従って, (3.3), (3.5)式より

$$\lim_{t\to\infty} \alpha_i(t) = 0 \tag{3.41}$$

が導かれる．これより，

$$\epsilon_1 < \lambda \min\{\gamma_{i1}, \gamma_{i2}\} \tag{3.42}$$

を満たす，任意の定数 $\epsilon_1 > 0$ に対して，次式を満たす，$t_1$ が存在する．

$$\|\alpha_i(t)\|_\infty \leq \epsilon_1, \quad \forall t \geq t_1 \tag{3.43}$$

ここで，$V(\cdot) := |\cdot|$ とすると，(2.24)，(3.43) 式より，

$$\begin{aligned}
\dot{V}(e_{i1}(t)) &= \frac{e_{i1}\dot{e}_{i1}}{|e_{i1}|} = -\lambda|e_{i1}| + \frac{e_{i1}}{|e_{i1}|}\alpha_{i1} \\
&\leq -\lambda V(e_{i1}(t)) + |\alpha_{i1}(t)| \\
&\leq -\lambda V(e_{i1}(t)) + \epsilon_1, \quad \forall t \geq t_1
\end{aligned}$$

が満たされる．比較定理[69]を用いると，次式が得られる．

$$\begin{aligned}
V(e_{i1}(\tau)) &\leq e^{-\lambda(\tau-t_1)}V(e_{i1}(t_1)) + \epsilon_1 \int_0^{\tau-t_1} e^{-\lambda s} ds \\
&= e^{-\lambda(\tau-t_1)}\left(V(e_{i1}(t_1)) - \frac{\epsilon_1}{\lambda}\right) + \frac{\epsilon_1}{\lambda}
\end{aligned} \tag{3.44}$$

(3.44) 式の右辺は $\tau \to \infty$ で $\epsilon_1/\lambda$ に収束する．定数 $\epsilon_1$ は任意に選べるため，次式を満たす $t_c$ が存在することが分かる．

$$V(e_{i1}(t)) \leq \gamma_{i1}, \quad \forall t \geq t_c \tag{3.45}$$

同様にして，$e_{i2}(t)$ についても，

$$V(e_{i2}(t)) \leq \gamma_{i2}, \quad \forall t \geq t_c \tag{3.46}$$

が導かれる．

以上より，$t \geq t_c$ での最適解は常に $\alpha_i^*(\tau) = 0$ となるため，(2.24) 式より $t \to \infty$ で $e_i(t) \to 0$ が導かれる． □

定理2より閉ループ系の漸近安定性が示される．

## 3.2 衝突回避問題に適合した分枝限定法

3.1.1 節で示したように衝突回避のための最適制御問題は移動体同士の衝突回避制約に0,1変数を含むため，混合整数二次計画問題と定式化される．ここでは混合整数二次計画問題の一般的な解法である分枝限定法の問題点を明らかにし，衝突回避問題の特性を考慮した新たな分枝限定法を提案する．これにより，最適制御問題の解法においても計算量の低減化を図る．

### 3.2.1 一般的な分枝限定法の問題点

3.1.1節の最適制御問題$P_1$は，(3.12)式の衝突回避制約が0，1変数を含む(2.7)式の形式に変換されるため以下のような混合整数二次計画問題$P_1'$となる.

衝突回避のための最適制御問題$P_1'$

$$\min_{\hat{\alpha}_i} \sum_{\tau=k+1}^{k+N} \widehat{\alpha}_i(\tau-1|k)^T R \widehat{\alpha}_i(\tau-1|k)$$

subject to eq. (3.11), (3.13), (3.14)

$$\begin{bmatrix} \widehat{z}_i(\tau|k) - \widehat{z}_j(\tau|k) \\ -\widehat{z}_i(\tau|k) + \widehat{z}_j(\tau|k) \end{bmatrix} \leq M \begin{bmatrix} \kappa_{ij1}(\tau|k) \\ \kappa_{ij3}(\tau|k) \\ \kappa_{ij2}(\tau|k) \\ \kappa_{ij4}(\tau|k) \end{bmatrix} - \begin{bmatrix} \psi \\ \psi \\ \psi \\ \psi \end{bmatrix}$$

$$\sum_{p=1}^{4} \kappa_{ijp}(\tau|k) \leq 3$$

$$\forall j \neq i, \quad \tau = k+1,\ k+2,\ \ldots,\ k+N$$

ただし，$\widehat{z}_i(\tau|k) = \widehat{e}_i(\tau|k) + \mu_i(\tau|k)$で，連続値変数の次元は$2N$，0，1変数の次元は$4N(n-1)$である．また，サンプリングタイム$\delta_m$で離散化し，予測ステップを$N$とする(すなわち$T = \delta_m N$). なお，更新周期$\delta$とモデル予測制御の離散化のサンプリング周期$\delta_m$の関係を示したのがFig. 3.2である．Fig. 3.2の矢印が予測ホライズンを示し，●印が予測を行う時刻を示している．このように実装上は離散的に衝突回避を考慮することとなる．

この問題は0，1変数を固定すると二次計画問題となり，原理的には0，1変数の全ての組み合わせを列挙することにより最適解を得ることができる．混合整数計画問題の標準的な解法である分枝限定法では，以下のような限定操作と分枝操作を繰り返すことにより，最適解が得られる可能性のない不必要な組み合わせをできるだけ省略することにより，計算時間を短縮することを目的としている[70].

**（分枝操作）** 一部の0，1変数を固定した子問題を生成する．具体的には，整数条件$\kappa = 0$ or $\kappa = 1$を$0 \leq \kappa \leq 1$に緩和した最適化問題（連続緩和問題）を解き，整数条件を満たさなかった0，1変数を1つ選んで0と1に固定することにより，新たな2つの子問題を生成する．

**（限定操作）** ある子問題の連続緩和問題の最適値は，その子問題の最適値の下界を与える．したがって，連続緩和問題の最適値がそれまでに得られた最適値（暫定値）よりも大きければ，その子問題からは最適解を

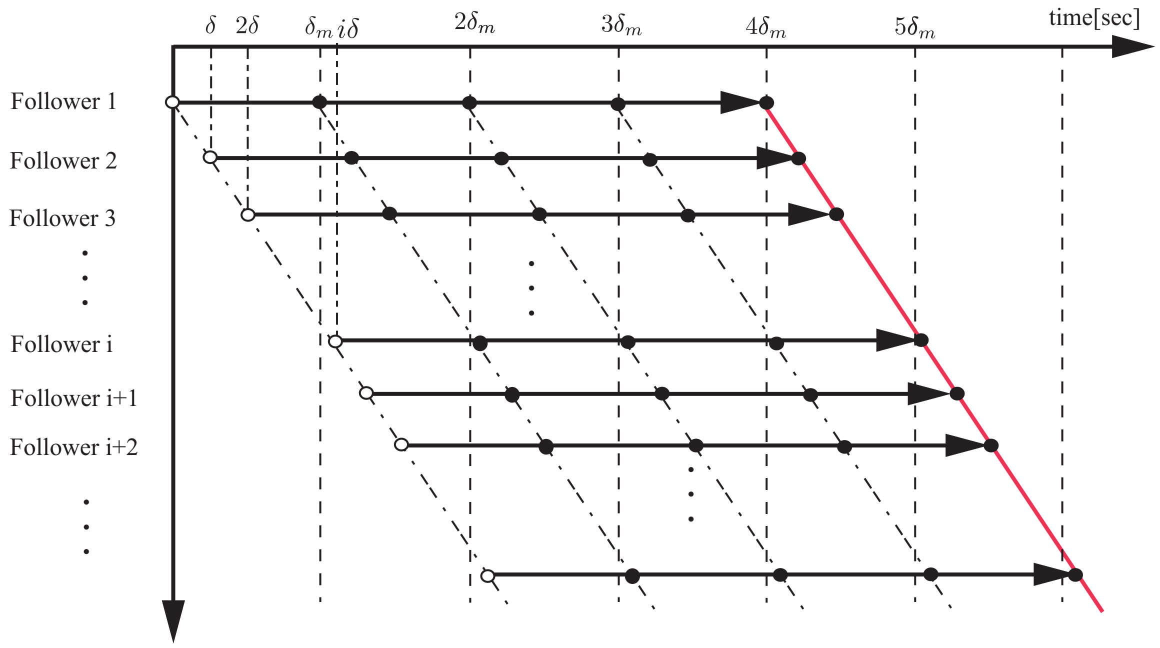



Fig. 3.2:Prediction horizon(arrows) and prediction intervals(mark •). Case of discretizing the optimization probelm with $\delta_m(\neq \delta)$

> 得られる可能性がないため，その子問題は終端する．その他，連続緩和問題の解が全て整数条件を満たした場合や，解が得られなかった場合も終端する．最終的に全ての子問題が終端したときの暫定解が元問題 $P_1'$ の最適解となる．

以上のように分枝限定法では子問題が早く終端すればするほど，計算量が短縮できることがわかる．ところが，上記の方法は一般的な混合整数二次計画問題に適用可能である反面，最適制御問題 $P_1'$ に対して子問題が早く終端する根拠はなく，最悪の場合，すべての0，1変数の組み合わせを列挙するのと同等の計算時間がかかってしまう．

一方，移動体が極度に密集している場合を除いて，全ての移動体に対する衝突回避を常に行わなければならないという状況は稀であるため，最適制御問題 $P_1'$ の衝突回避制約の中には実際には不必要な0，1変数が数多く含まれている．標準的な分枝限定法は全ての0，1変数に対して同様の分枝操作を行うため，衝突回避が実際には必要ない状況で，不必要な子問題が数多く生成されていると予想される．この問題点が顕著に表れている数値例をつぎに示す．

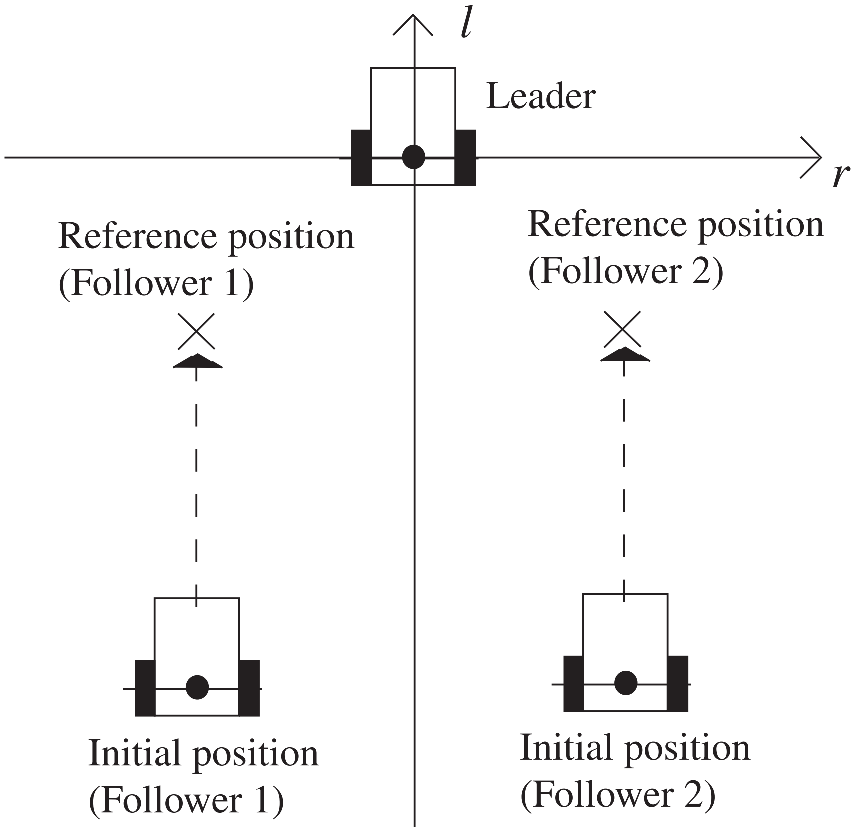



Fig. 3.3:Example of formation control

**例1** Followerを2台，予測ステップ数$N$を6とし，Follower 1, 2の初期状態をLeaderに固定した移動座標系においてそれぞれ$(-1.0, -3.0, \pi/2)$，$(1.0, -3.0, \pi/2)$とし，目標位置をそれぞれ$(-1.0, -1.0)$，$(1.0, -1.0)$とする（Fig. 3.3参照）．なお，その他のパラメータは$v_r = 0.0$，$w_r = 0.0$，$\psi = 0.5$，$M = 30$，$\delta = 0.25$，$\lambda = 0.3$，$\gamma_i = 0.1$，$\eta = 0.4$，$R = I_{2N}$とする．この例では，Fig. 4.2のように$l$軸と並行に直進する軌道が最適解であり，衝突回避が必要ないことは明らかである．実際，最適化問題から衝突回避制約を除いてモデル予測制御を実装すると，衝突することなく目標の編隊形状が達成でき，各時刻の最適化問題にかかった計算時間は最長で0.0122[sec]であった．なお，計算に用いたPCはCPU: Intel Pentium IV 3.0GHz，RAM: 1GBであり，二次計画法はMATLAB Optimization Toolbox[79]を使って解いた．一方，衝突回避制約を含めた混合整数二次計画問題を標準的な分枝限定法で解く場合，子問題が82個生成され，計算時間は1.27[sec]となった．なお，混合整数二次計画問題はMATLAB Hybrid Toolboxの混合整数二次計画ソルバー[80]を用いて解いた．

この例が示すように，標準的な分枝限定法は一般的な混合整数計画問題

に適用可能である反面，衝突回避問題に対して必ずしも効率のよい方法とはいえない．次節では，衝突回避問題に対して不必要な子問題の生成をできるだけ抑えることを目的とした分枝限定法について述べる．

### 3.2.2　提案する分枝限定法

前節でも述べたように，衝突回避のための最適制御問題においては，予測を行う全ての時刻に対して，全ての移動体との衝突回避制約が課せられており，実際には不必要な制約条件も数多く含まれている．我々が日常的に行っている衝突回避においても，衝突を予測してからどの方向に回避するかを決定することが多く，常に衝突回避を意識して移動することは稀であると思われる．

提案手法は，整数条件を緩和した問題ではなく，衝突回避制約を除いた問題を緩和問題とする．緩和問題を解いて得られた軌道が衝突回避制約を満たさない場合には，Fig. 3.4で表される4つの領域への回避を行う制約条件を追加した子問題を生成する．具体的には，

$$\begin{aligned} x_{vi} - x_{vj} &\leq M\kappa_{ij1} + M\kappa_{ij2} - \psi \\ y_{vi} - y_{vj} &\leq -M\kappa_{ij1} + M\kappa_{ij2} + M - \psi \\ -x_{vi} + x_{vj} &\leq M\kappa_{ij1} - M\kappa_{ij2} + M - \psi \\ -y_{vi} + y_{vj} &\leq -M\kappa_{ij1} - M\kappa_{ij2} + 2M - \psi \end{aligned} \tag{3.47}$$

において，$(\kappa_{ij1}, \kappa_{ij2}) = (0,0),\ (0,1),\ (1,0),\ (1,1)$と固定した制約条件を追加した4つの子問題を生成する．すなわち，衝突が予測されるまでは衝突回避を意識せずに移動し，衝突が予測されるに伴い，回避する方向を決定するという方法になっている．この方法により，衝突回避制約を新たに必要としない子問題は必ず終端するため，不必要な子問題の生成が抑えられると考えられる．

提案する分枝限定法のアルゴリズムを以下に示す.

**Step 0:** 初期化

リスト$L = \{P'_1\}$，暫定値$z^* = \infty$，暫定解$X^* = \phi$，番号$l = 0$とおく．

**Step 1:** 最適性判定

リスト$L = \phi$であれば終了．このとき暫定値$z^* < \infty$であれば，対応する暫定解$X^*$が$P_0$の最適解となる．一方，$z^* = \infty$であれば，$P_0$は実行可能解を持たず終了．また，$L \neq \phi$であればStep 2へ行く．

**Step 2:** 子問題の選択

リスト$L$から1つの子問題$P_k$を選び出して$L = L - \{P_k\}$とする.

**Step 3:** 限定操作

$P_k$から分枝操作で固定した衝突回避制約以外を取り除いた緩和問題$P'_k$を解く.このとき$P'_k$が実行可能解を持たなければ,Step 1に戻る.一方,最適値$z'_k$をもち,暫定値$z^*$に対して$z'_k \geqq z^*$であれば,Step 1に戻り,$z'_k < z^*$であればStep 4へ行く.

**Step 4:** 更新操作

$P'_k$の最適解$X'_k$が衝突回避制約を満たすか否か判定する.衝突回避制約を満たす場合,$z^* = z'_k$,$X^* = X'_k$とし,Step 1に戻る.一方,衝突回避制約を満たさない場合にはStep 5へ行く.

**Step 5:** 分枝操作

$P'_k$に対して,衝突回避制約を満足しない最小の$\tau$を$m$,そのときのFollowerを$j$とする.このとき,衝突回避制約(3.47)の$\kappa$を

$$[\kappa_{ij1}(m), \kappa_{ij2}(m)]^T = [0,0]^T,\ [0,1]^T,\ [1,0]^T,\ [1,1]^T$$

に固定した子問題$P_{l+1}$,$P_{l+2}$,$P_{l+3}$,$P_{l+4}$を生成する.また,リスト$L = L \cup \{P_{l+1}, P_{l+2}, P_{l+3}, P_{l+4}\}$,$l = l + 4$としてStep 1へ戻る.

以上のように提案手法は従来手法と同様に列挙法となっているため,従来法と同様に最適解を得ることができる.その一方で提案手法では,分枝操作は衝突回避制約を満たさない時刻$\tau$,Follower $j$に関する0,1変数に対してのみ行われる(Step 5).多くの衝突回避問題では移動体が密集していることは稀であり,軌道上で起きうる衝突も限られていることを考慮すると,提案手法では生成される子問題の数の低減化が期待できる.したがって従来手法と比較して計算量が低減化されると考えられる.次節では数値例により生成される子問題の数が低減されることを示す.

## 3.3 数値例

3.1.1節で提案した衝突回避手法と3.2.2節で提案した分枝限定法の有効性をシミュレーションで検証した.

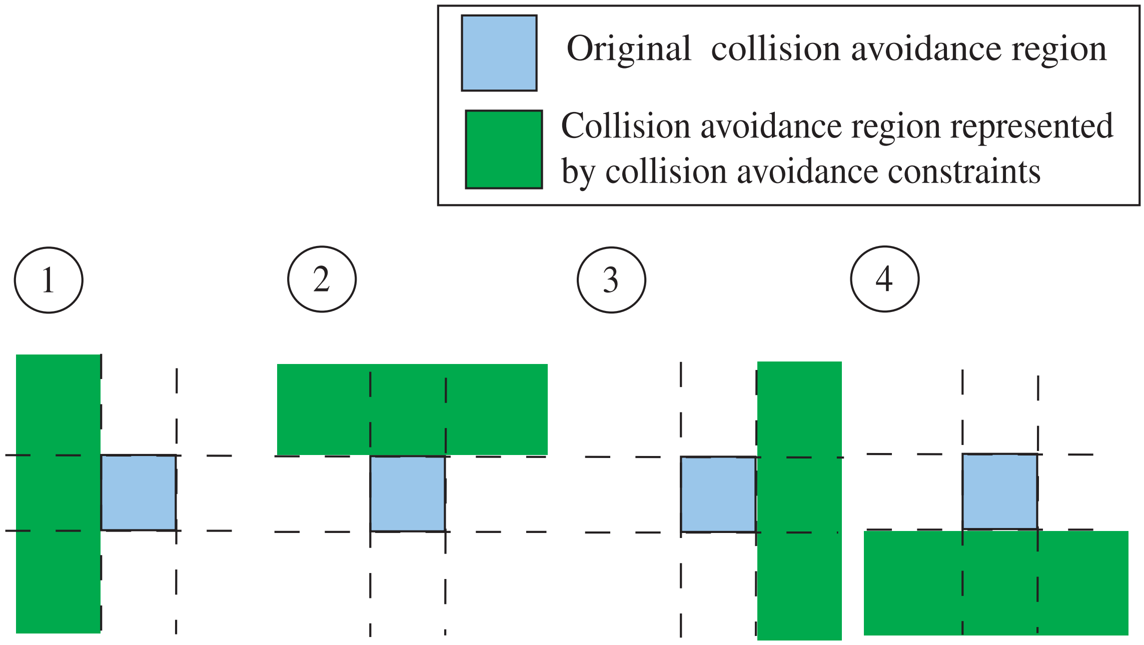



Fig. 3.4:Collision avoidance region represented by eq.(3.47)

### 3.3.1　衝突回避手法の検証

まず，3.1.1節で提案した手法の有効性を確認するためMATLABを用い，Followerが3台の場合についてシミュレーションを行った．ただし，混合整数二次計画問題を解くソルバーには分枝限定法に基づいた[80]を用いた．

Leaderを絶対座標系で$(0,0,0)$に配置し，$x$軸に沿って6[m/sec]で等速直線運動を実現させるように設定した．また，3台のFollowerの初期状態を移動座標系で，各々，$(2,0,\pi/2)$，$(0,0,\pi/2)$，$(-2,0,\pi/2)$とし，目標位置を移動座標系で各々，$(-2,0)$, $(0,0)$, $(2,0)$とした．その他のパラメータは，$d = 0.15$，$\psi = 1$，$M = 30$，$T = 0.4$，$\delta = 0.1$，$\lambda = 0.3$，$\gamma_i = 0.2$である．ただし，この例では$\eta = \infty$とし，入力制約(3.13)は用いなかった．以上のパラメータを用い，最適制御問題(3.10)をサンプリング周期$\delta_m = 0.1$[sec]で離散化し，シミュレーションを行った．

Fig. 3.5，Fig. 3.6は各々，絶対座標系でのFollowerの軌跡，Leaderに固定した移動座標系でのFollowerの軌跡を示したものである．また，Fig. 3.5の丸印は一定時間毎のFollowerの位置を示している．これらの図より，Follower 1，3がFollower 2と衝突しないように各々の位置を入れ替えるように移動し，目標の編隊が達成されていることが確認できる．また，この際の，Follower間の最小距離$\min_{i,j} \|z_i - z_j\|_\infty$を示したものがFig. 3.7である．この図より制約(3.12)が満たされていることが確認できる．

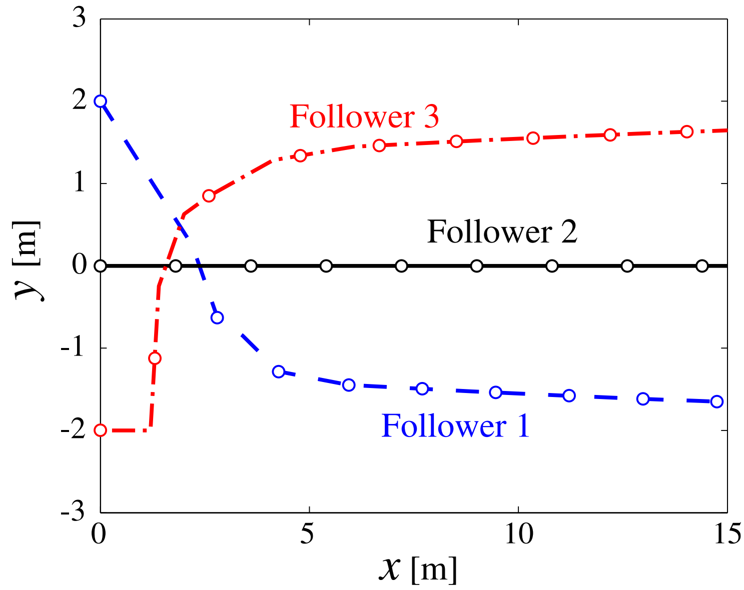



Fig. 3.5:$x$-$y$ plot at the global frame (Simulation)

### 3.3.2　計算量の評価

次に3.2.2節で提案した分枝限定法の有効性の検証を行った．具体的には3.1.1節の衝突回避問題 $P_1$ を3.2.2節で提案した方法と従来手法のそれぞれを用いて解いた場合のシミュレーション結果を示す．ここではFollowerが3台の場合と4台の場合について，それぞれ予測ステップ数 $N$ を5〜10に変化させシミュレーションを行った．このときの各時刻における最適制御問題を解くのに要した最大の計算時間及び子問題の数の比較を行った．なお，Leaderの制御入力 $u_r$ は衝突回避問題の計算量には大きな影響を与えないため，ここではLeaderが固定の場合の結果のみを示すこととする．

混合整数二次計画問題を解くソルバーには従来手法としてMATLAB Hybrid Toolboxの混合整数二次計画ソルバー(miqp.m)[80]を用いた．また，提案手法ではmiqp.mと同じ二次計画ソルバーを用い，miqp.mの分枝操作を変更したソルバーを作成して計算を行った．なお，計算に用いたPCはCPU: Intel Pentium IV 3.0GHz，RAM: 1GBである．

#### Followerが3台の場合

Follower 1, 2, 3の初期状態はLeaderに固定した移動座標系 $(r, l)$ においてそれぞれ $(1.0, 0.0, \pi/2)$，$(0.0, -1.0, \pi/2)$，$(-1.0, 0.0, \pi/2)$ とし，目標位置は移動座標系 $(r, l)$ においてそれぞれ $(-1.0, 0.0)$，$(0.0, 1.0)$，$(1.0, 0.0)$ とした．その他のパラメータ

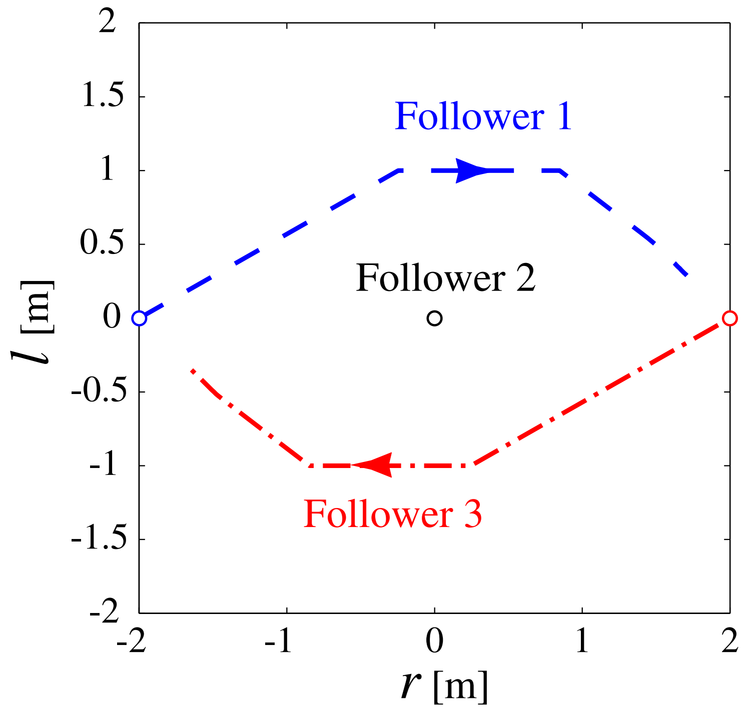



Fig. 3.6:$x$-$y$ plot at the local frame (Simulation)

は $v_r = 0.0$, $w_r = 0.0$, $\psi = 0.5$, $M = 30$, $\delta = 0.25$, $\lambda = 0.3$, $\gamma_i = 0.1$, $\eta = 0.4$, $R = I_{2N}$ である．3.1.1 節の最適制御問題 $P_1$ の離散化はサンプリング周期 $\delta_m = 1.0$[sec] で行った．なお，0-1 変数の総数の変化による影響を除くため従来手法・提案手法ともに (3.47) 式の形式の制約を衝突回避制約として用いた．

シミュレーションの結果を Table 3.1 に示す．Table 3.1 は各時刻の最適制御問題を解くのに要した最長の計算時間とその際に解いた緩和問題の数を示したものである．この表から従来手法では予測ステップ $N$ の増加に伴い計算時間が急激に増加してしまうことが確認できる．一方，提案手法では，計算時間の増加が抑えられていることが確認できる．また，従来手法では解いた緩和問題の数が予測ステップの増加とともに急激に増加していることが確認できる．一方，提案手法では解いた子問題の数が低減化されており，子問題の数が低減化されたことで計算時間が低減化されたことが確認できる．

**Follower が 4 台の場合**

Follower 1, 2, 3, 4 の初期状態は Leader に固定した移動座標系 $(r, l)$ においてそれぞれ $(1.0, 0.0, \pi/2)$，$(0.0, -1.0, \pi/2)$，$(-1.0, 0.0, \pi/2)$，$(0.0, 1.0, \pi/2)$ とし，目標位置は

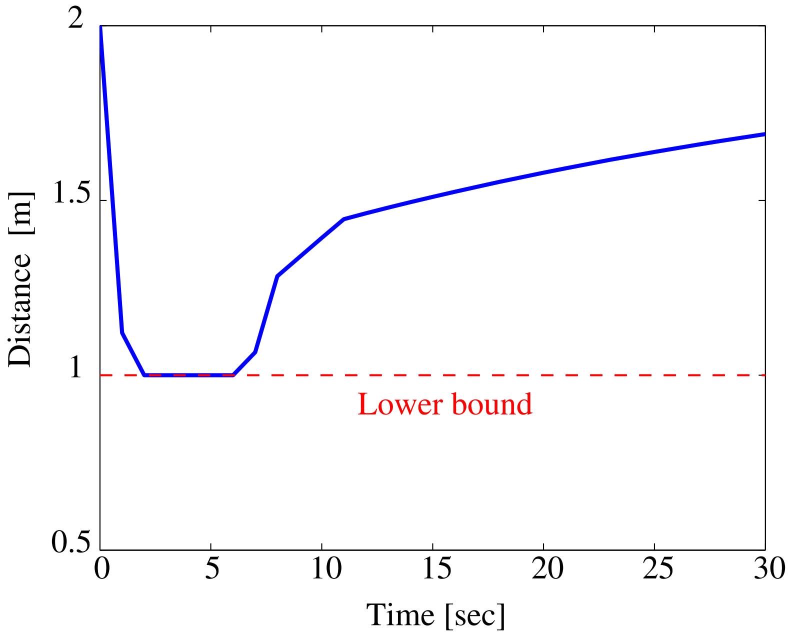



Fig. 3.7:Minimum distance among followers(Simulation)

移動座標系$(r,l)$においてそれぞれ$(-1.0,0.0)$，$(0.0,1.0)$，$(1.0,0.0)$，$(0.0,-1.0)$とした．その他のパラメータおよび条件は3台の場合と同じである．

シミュレーションの結果をTable 3.2に示す．Table 3.2は各時刻の最適化問題にかかった最長の計算時間とその際に解いた緩和問題の数を示したものである．ただし，予測ステップが5の場合は実行不可能であった．この表から従来手法では，予測ステップ$N$の増加に伴い計算時間が急激に増加してしまうことが確認できる．特に$N \geq 8$では計算が困難となってしまった．一方，提案手法では，計算時間の増加が抑えられ$N \geq 8$でも計算を行うことが可能であることが確認できる．また，提案手法では子問題の数が低減化されており，3台の場合の結果と同様に子問題の数が低減化されたことで計算時間が低減化されたことが確認できる．

## 3.4　実験

実験により提案した編隊制御手法の有効性を検証した．ここではFollowerが2台の場合とFollowerが3台の場合について実験を行った．

なお，3台の場合には，前節の結果より一般的な分枝限定法ではオンラインで解くことが困難であることが示されている．したがって，Followerが3台の場合には3.2.2節で提案した分枝限定法を用いて最適制御問題を解くこと

Table 3.1:Maximum computation time and number of subproblems(n=3)

| Prediction step | Computation time[s] | | Number of subproblems | |
|---|---|---|---|---|
| | Standard | Proposed | Standard | Proposed |
| 5 | 1.84 | 0.111 | 187 | 17 |
| 6 | 5.01 | 0.136 | 391 | 17 |
| 7 | 12.2 | 0.196 | 859 | 21 |
| 8 | 43.3 | 0.669 | 1840 | 73 |
| 9 | 151 | 1.26 | 5300 | 113 |
| 10 | 336 | 1.33 | 10000 | 113 |

Table 3.2:Maximum computation time and number of subproblems(n=4)

| Prediction step | Computation time[s] | | Number of subproblems | |
|---|---|---|---|---|
| | Standard | Proposed | Standard | Proposed |
| 5 | — | — | — | — |
| 6 | 245 | 0.346 | 22500 | 45 |
| 7 | 1350 | 0.591 | 102000 | 69 |
| 8 | * | 1.21 | * | 113 |
| 9 | * | 1.81 | * | 141 |
| 10 | * | 5.48 | * | 401 |

とした．

### 3.4.1 実験システムの概要

実験に用いたシステムをFig. 3.8に示す．実験には市販のラジコン戦車(TAMIYA社製Tiger I)を改造したものを用いた．また，各移動体の位置，姿勢情報はカメラ画像をもとにしたリアルタイム3次元位置･姿勢計測システム((株)応用計測研究所Quick Mag IV)を用いて取得した. Followerの制御入力の計算はシステムの簡略化のため1台のPC(CPU: Intel Pentium IV 3.0GHz, RAM: 1GB)で行った．計算機からの出力はシリアル通信(RS232C)により，各移動体に搭載したマイコン(H8/3048F)に送信される．さらにH8/3048Fマイコンからモータドライバに対しそれに応じたPWM信号を出力し，速度制御を実現するものである．なお，実験スペースの関係上，本節で示す実験は全てLeaderは静止して動かないものとした．

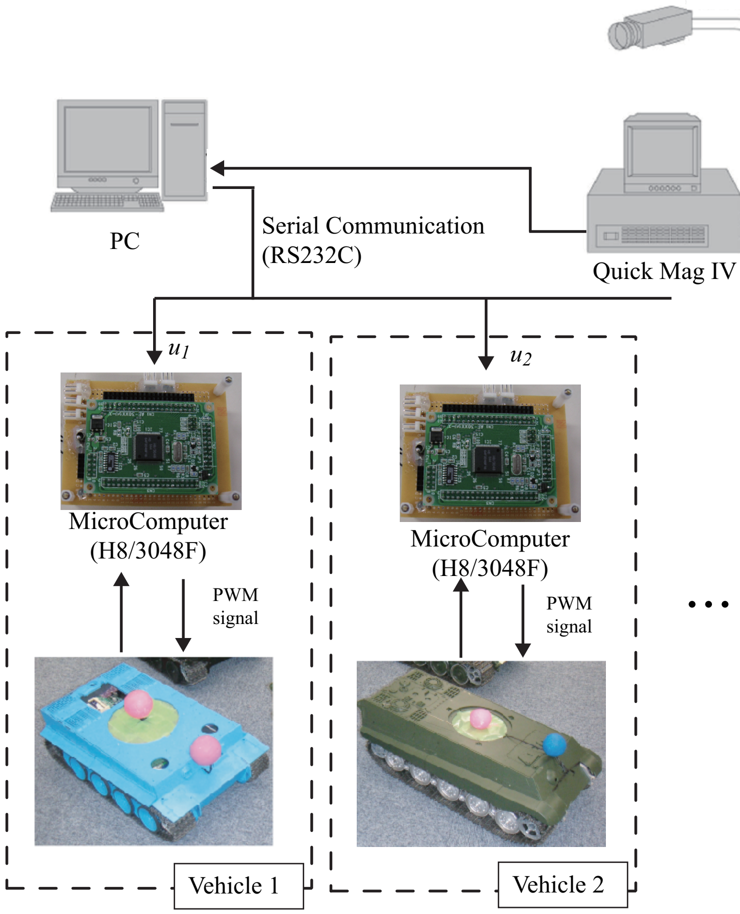



Fig. 3.8:Experimental system

### 3.4.2 Followerが2台の場合

まず，Followerが2台の場合の実験結果を示す．

実験に用いたパラメータを以下のとおりである．Follower 1，2の初期状態は各々，移動座標系において，$(0.5, -1.75, 3\pi/4)$，$(-0.5, -1.75, \pi/4)$，目標位置は各々，移動座標系において$(-0.5, -0.5)$，$(0.5, -0.5)$とした．その他のパラメータは，$d = 0.15$，$\psi = 0.45$，$M = 30$，$T = 3.0$，$\delta = 0.25$，$\lambda = 0.4$，$\gamma_i = 0.125$，$\eta = 0.5$である．本実験ではLeaderを実際に設置しているが，これはLeaderの位置を示すためであり，動作はしない．なお，混合整数二次計画問題を解くソルバーにはシミュレーションと同様のものを用いた．これをMATLAB Compiler[81]を用い変換し，実行プログラムを作成した．

まず，この条件下で，3.1.1節の衝突回避を行わない場合$(i.e. \alpha_i = 0)$について実験を行った．このときのFollowerの軌跡を示した図がFig. 3.9である．この図

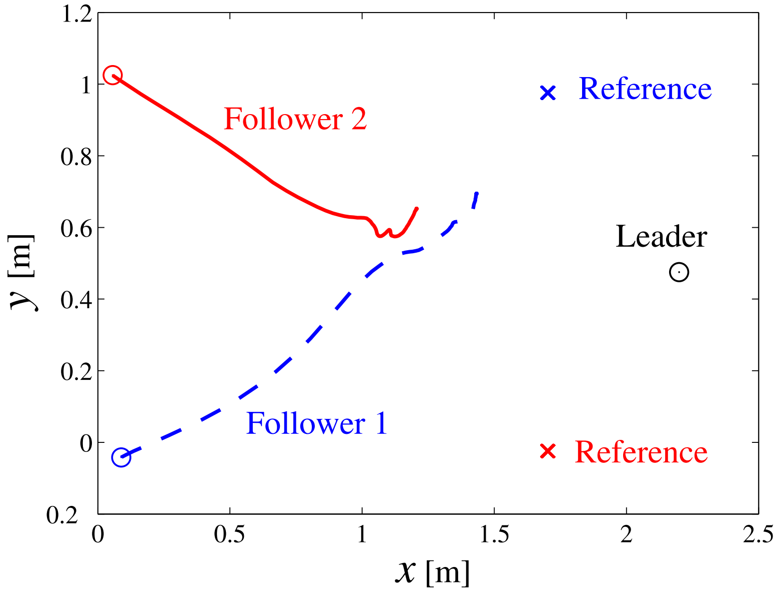



Fig. 3.9:Paths of followers for $\alpha_i = 0$

よりFollower 1とFollower 2の衝突が生じ，目標の編隊が達成できないことが確認できる．

次に提案手法を適用した場合について，最適制御問題$P_1$をサンプリング周期$\delta_m = 1.0$[sec]で離散化し実験を行った．Fig. 3.10，Fig. 3.11はそれぞれ絶対座標系でのFollowerの軌跡，Follower間の最小距離を示したものである．また，Fig. 3.10の丸印は一定時間毎の各Followerの位置を示している．これらの図よりFollower同士，衝突することなく目標の位置に収束していることが確認できる．なお，各最適化問題を解くのに要した時間は最大，Follower 1で0.215[sec]，Follower 2で0.192[sec]であった．また，この実験では，システムの入力に$|\omega_i| \leq 1.0$という制約があるため，入力制約(3.13)が特徴的な役割を果たしている．Fig. 3.12，Fig. 3.13はそれぞれ入力制約がない場合のFollowerの軌跡，Follower 2の制御入力の角速度$\omega_2$を示した図である．Follower 1は入力制約を侵害することは無かったが，Follower 2は，これらの図が示すように入力制約(3.13)が無い場合，$\omega_2$の入力制約を大きく破ってしまうため，Fig. 3.10と比較し大きくオーバーシュートを生じていることが確認できる．

### 3.4.3　Followerが3台の場合

次にFollowerが3台の場合の実験結果を示す．編隊形状の初期状態および目標状態によっても移動体同士の衝突回避の難易度も変化するので，ここで

Table 3.3:Experimental parameter of each experiments

| Parameter | Case 1 | Case 2 | Case 3 | Case 4 |
|---|---|---|---|---|
| $\delta$[sec] | 0.25 | 0.25 | 0.25 | 0.25 |
| $N$ | 5 | 5 | 6 | 6 |
| $n$ | 3 | 3 | 3 | 3 |
| $\lambda$ | 0.3 | 0.3 | 0.3 | 0.3 |
| $\gamma$ | 0.1 | 0.1 | 0.1 | 0.1 |
| $M$ | 0.5 | 0.5 | 0.5 | 0.5 |
| $\eta$ | 0.4 | 0.4 | 0.4 | 0.4 |
| $d$[sec] | 0.15 | 0.15 | 0.15 | 0.15 |
| $R$ | I | I | I | I |

Table 3.4:Initial and reference position of followers at global coodinate

| | | Case 1 | Case 2 | Case 3 | Case 4 |
|---|---|---|---|---|---|
| Leader | Initial position | (3.5,1.0,0) | (3.5,1.0,0) | (3.5,1.0,0) | (3.5,1.0,0) |
| Follower 1 | Initial position | (2.0,2.0,0.0) | (0.5,1.0,0.0) | (1.75,0.0,pi/2) | (1.0,0.0,0.0) |
| | Reference position | (2.0,2.0) | (2.5,1.0) | (1.75,2.0) | (2.25,2.0) |
| Follower 2 | Initial position | (2.0,1.0,0.0) | (1.5,1.0,0.0) | (0.75,1.0,0.0) | (1.0,1.0,0.0) |
| | Reference position | (2.0,1.0) | (0.5,1.0) | (2.75,1.0) | (2.25,1.0) |
| Follower 3 | Initial position | (2.0,0.0,0.0) | (2.5,1.0,0.0) | (1.75,2.0,-pi/2) | (1.0,2.0,0.0) |
| | Reference position | (2.0,0.0) | (1.5,1.0) | (1.75,0.0) | (2.25,0.0) |

は初期状態および目標の編隊形状を変更したCase 1〜Case 4の計4つの場合について実験を行った．実験に用いたパラメータをTable 3.3，Table 3.4に示す．

実験は各パターンについて衝突回避アルゴリズム未適用の場合と衝突回避アルゴリズム適用の場合について行い，それらの比較を行った．衝突回避アルゴリズムを適用した場合は，3.1.1節の最適制御問題$P_1$をサンプリング周期$\delta_m = 1.0$[sec]で離散化して実験を行った．各々の実験結果を以下に示す．

なお，衝突回避アルゴリズム未適用の場合は衝突回避アルゴリズム適用の比較を行うため，スタートする時刻をサンプリングタイム$\delta$分ずつずらし実験を行った．これにより，スタート時の時間差の影響を排除し同等の条件下で比較できるようにした．また，混合整数二次計画問題のソルバーには3.3節で作成したソルバーを使用した．これを，MATLAB Compiler[81]を用いて変換し，実行ファイルを作成した．

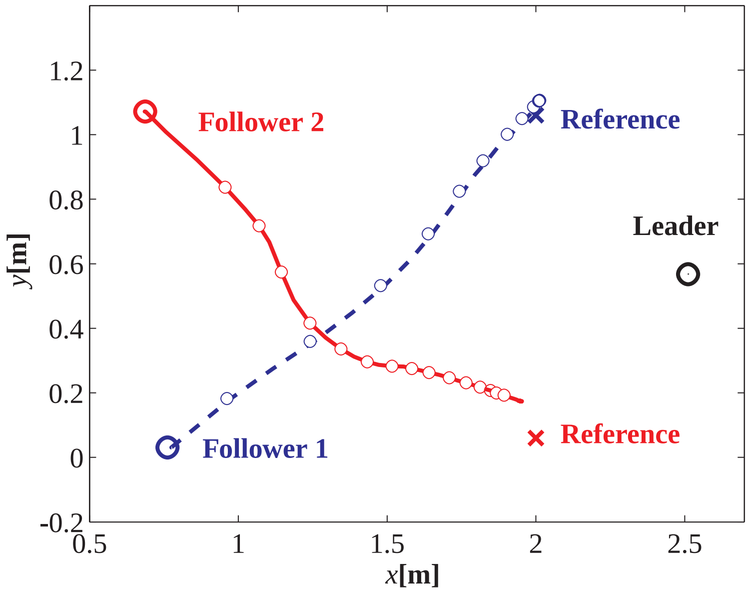



Fig. 3.10:Paths of followers (Experiment)

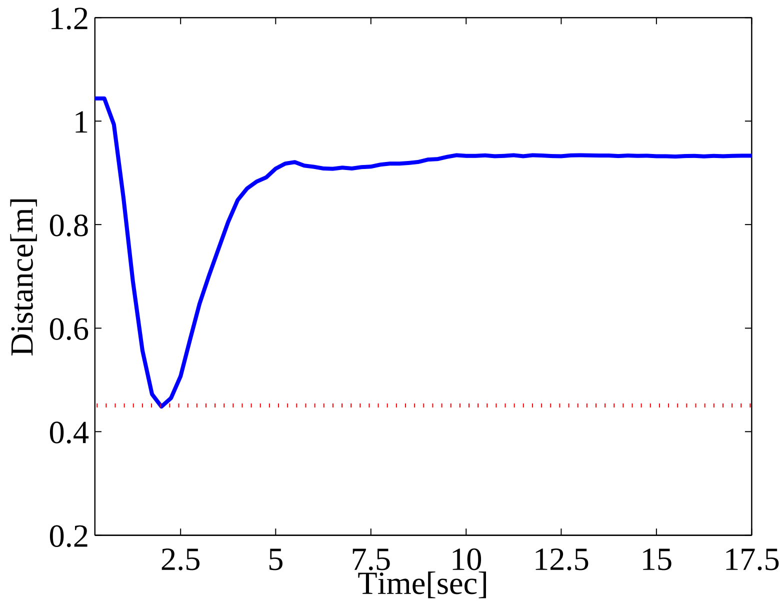



Fig. 3.11:Minimum distance among followers (Experiment)

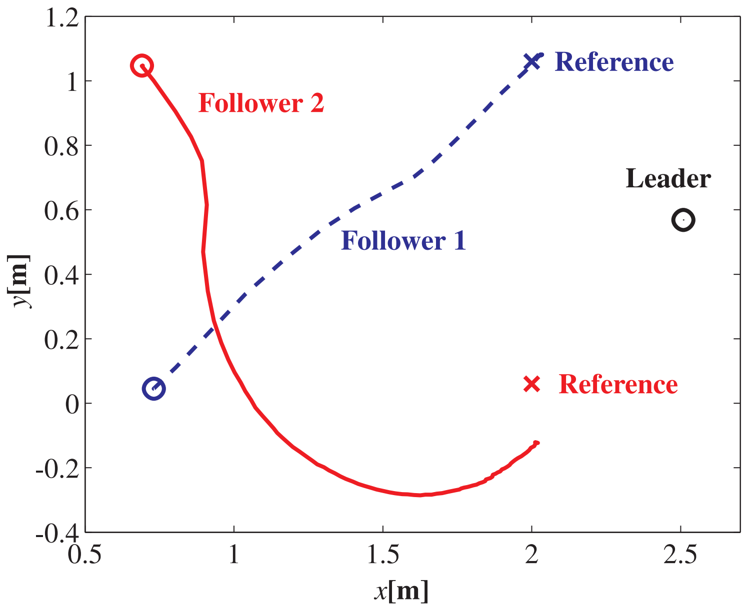



Fig. 3.12:Paths of followers (without input constraints)

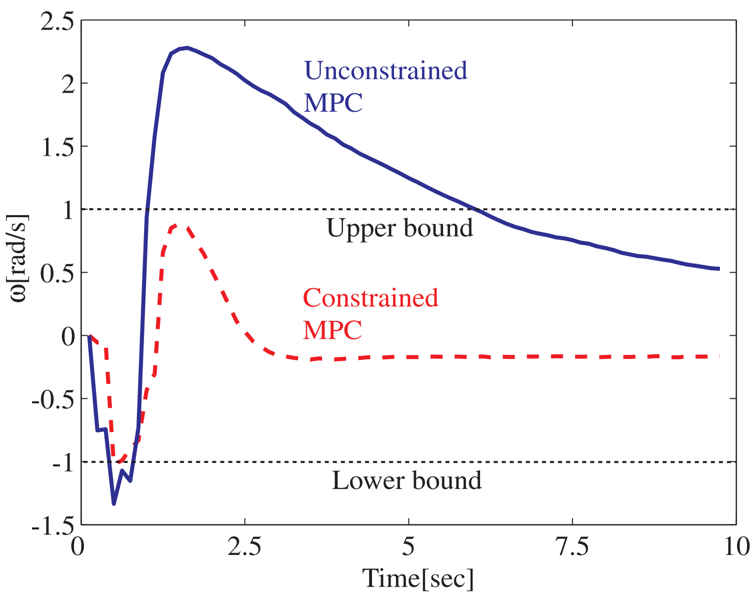



Fig. 3.13:$\omega$ of the follower 2

**Case 1**

Case 1の実験結果をFig 3.14〜Fig. 3.18に示す．Fig 3.14，Fig. 3.15はそれぞれ衝突回避アルゴリズム未適用の場合の移動体の軌跡，Follower間の最小距離を示したものである．Fig 3.16，Fig. 3.17はそれぞれ衝突回避アルゴリズムを適用した場合の移動体の軌跡，Follower間の最小距離を示したものである．Fig 3.14，Fig 3.16の∘印は1[sec]毎の各Followerの位置をしたものである．

Fig. 3.18は衝突回避アルゴリズムを適用した場合の動作の様子を示したものである．Fig. 3.18中の移動体(青)がFollower 1，移動体(黒)がFollower 2，移動体(緑)がFollower 3である．なお，Leaderは仮想的に与えられるものと仮定しているが，ここではLeaderの位置を示すため実際にLeaderとなる移動体(銀)を配置した．

Fig. 3.14，Fig. 3.15から衝突回避アルゴリズム未適用の場合では，移動体同士が衝突してしまい目標位置に収束することができないことが確認できる．一方，Fig. 3.16，Fig. 3.17から衝突回避アルゴリズム適用の場合では，移動体同士の衝突が回避され目標位置に収束することができていることが確認できる．

また，移動体の姿勢は制御していないためFig. 3.18からわかるようにFollower 1が後ろ向きに目標位置に向けて移動していることが確認できる．

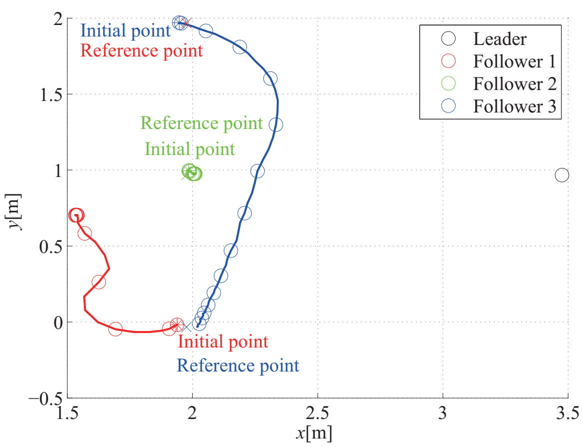



Fig. 3.14:XY-plot of Case 1 without collision avoidance

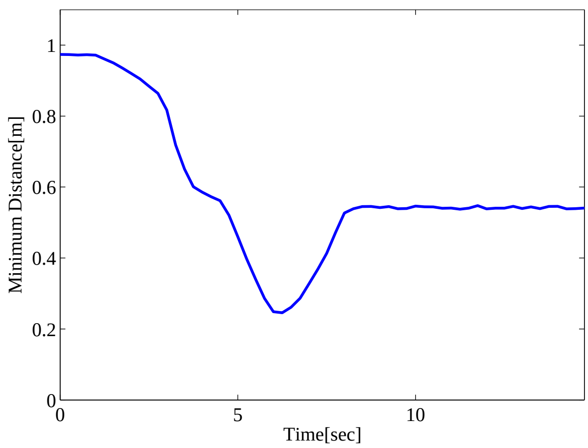



Fig. 3.15:Minimum distance among followers of Case 1 without collision avoidance

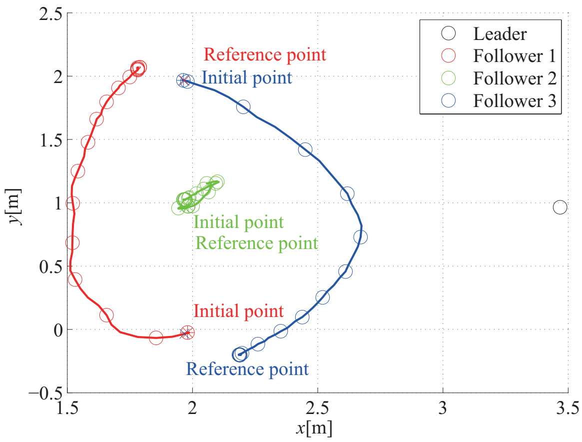



Fig. 3.16:XY-plot of Case 1 with collision avoidance

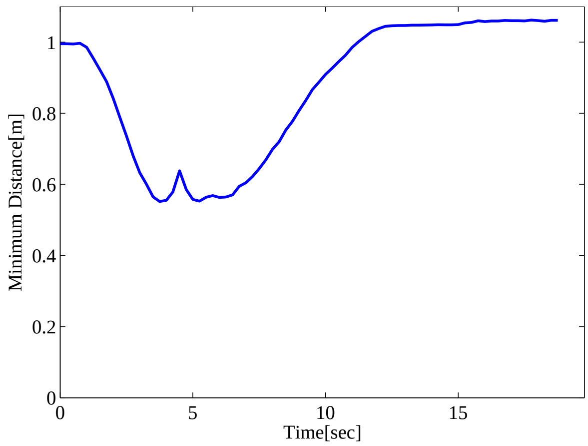



Fig. 3.17:Minimum distance among followers of Case 1 with collision avoidance

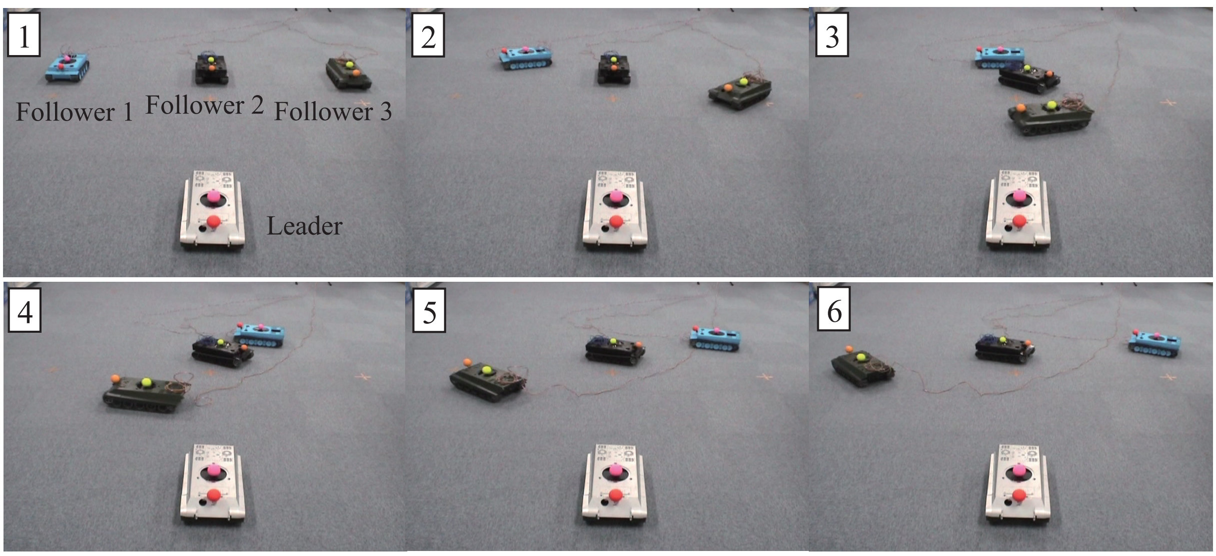



Fig. 3.18:The motion of followers in Case 1 with collision avoidance

### Case 2

Case 2の実験結果をFig. 3.19〜Fig. 3.23に示す．Fig. 3.19，Fig. 3.20はそれぞれ衝突回避アルゴリズム未適用の場合の移動体の軌跡，Follower間の最小距離を示したものである．Fig. 3.21，Fig. 3.22はそれぞれ衝突回避アルゴリズムを適用した場合の移動体の軌跡，Follower間の最小距離を示したものである．Fig 3.19，Fig 3.21の∘印は1[sec]毎の各Followerの位置をしたものである．Fig. 3.23は衝突回避アルゴリズムを適用した場合の動作の様子を示したものである．Fig. 3.23中の移動体の割り当てはCase 1と同様である．

Fig. 3.19，Fig. 3.20から衝突回避アルゴリズム未適用の場合では，移動体同士が衝突してしまい目標位置に収束することができないことが確認できる．一方，Fig. 3.16，Fig. 3.17から衝突回避アルゴリズムを適用した場合では，移動体同士の衝突が回避され目標位置に収束することができていることが確認できる．ただし，離散化の影響で衝突回避制約を一部侵害してしまっていることも確認することができる．しかし，あらかじめ衝突回避制約の大きさを大きめに設定しているため実際に移動体同士が衝突することはなかった．

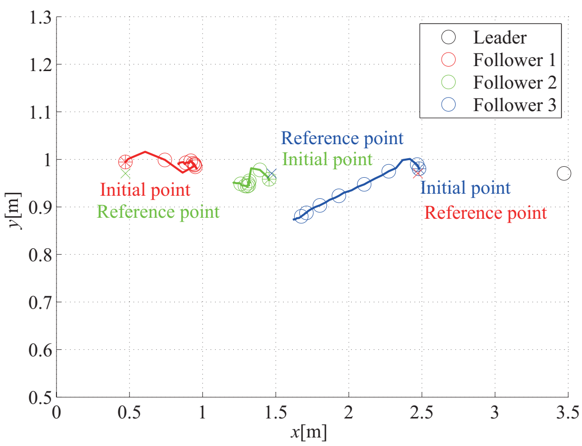



Fig. 3.19:XYplot of Case 2 without collision avoidance

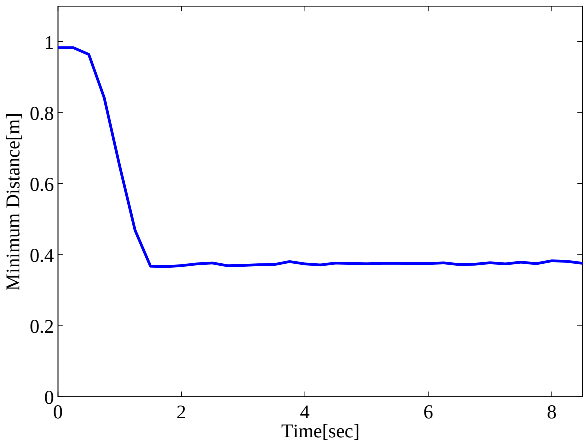



Fig. 3.20:Minimum distance among followers of Case 2 without collision aoidance

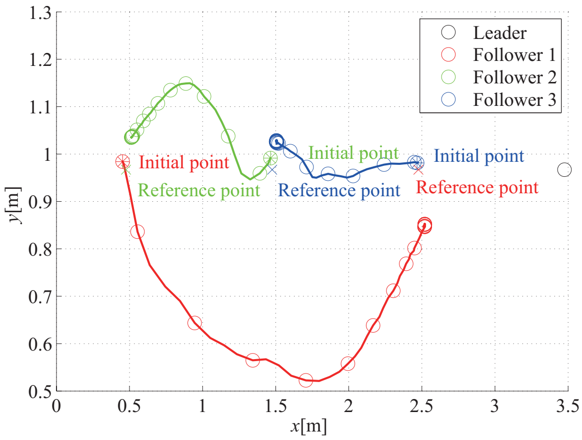



Fig. 3.21:XYplot of Case 2 with collision avoidance

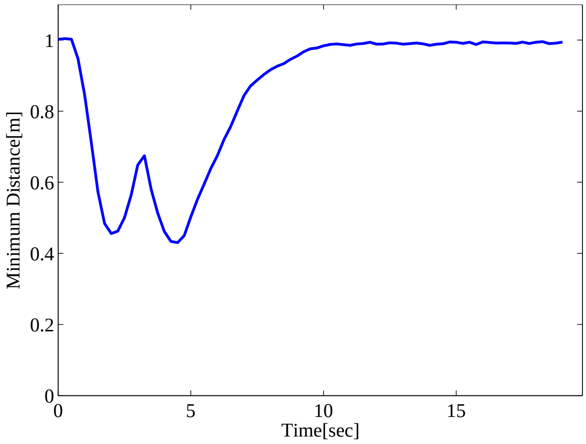



Fig. 3.22:Minimum distance among followers of Case 2 with collision aoidance

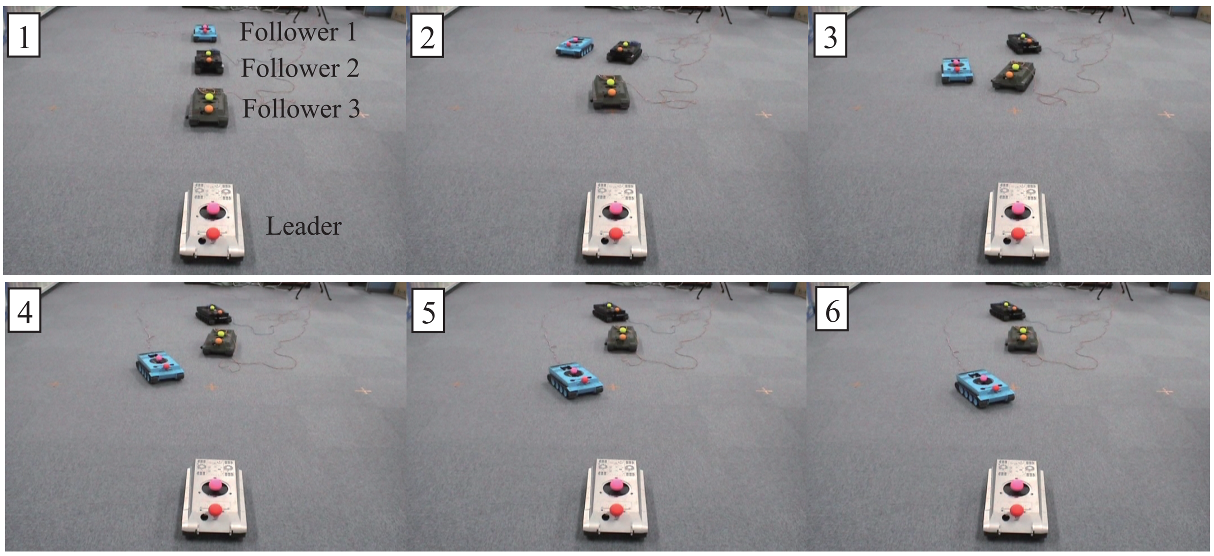



Fig. 3.23:The motion of followers in Case 2 with collision avoidance

### Case 3

Case 3の実験結果をFig. 3.24〜Fig. 3.28に示す．Fig. 3.24，Fig. 3.25はそれぞれ衝突回避アルゴリズム未適用の場合の移動体の軌跡，Follower間の最小距離を示したものである．Fig. 3.26，Fig. 3.27はそれぞれ衝突回避アルゴリズムを適用した場合の移動体の軌跡，Follower間の最小距離を示したものである．Fig 3.24，Fig 3.26の∘印は1[sec]毎の各Followerの位置をしたものである．Fig. 3.28は衝突回避アルゴリズムを適用した場合の動作の様子を示したものである．Fig. 3.28中の移動体の割り当てはCase 1と同様である．

Fig. 3.24，Fig. 3.25から衝突回避アルゴリズム未適用の場合では，移動体同士が衝突してしまい目標位置に収束することができないことが確認できる．一方，Fig. 3.26，Fig. 3.27から衝突回避アルゴリズムを適用した場合では，移動体同士の衝突が回避され目標位置に収束することができていることが確認できる．

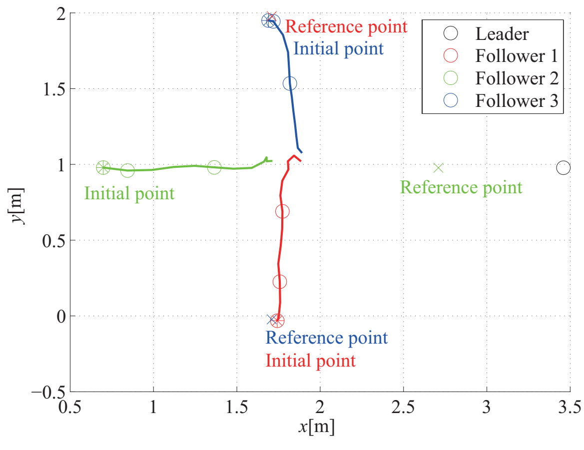



Fig. 3.24:XYplot of Case 3 without collision avoidance

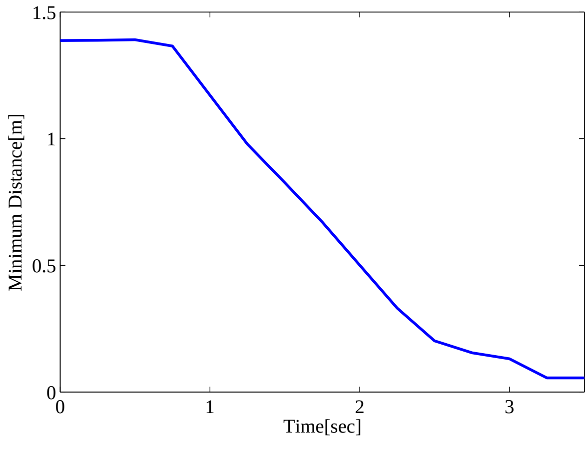



Fig. 3.25:Minimum distance among followers of Case 3 without collision avoidance

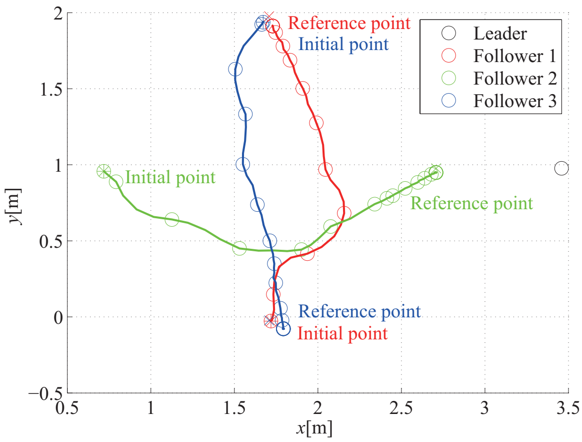



Fig. 3.26:XYplot of Case 3 with collision avoidance

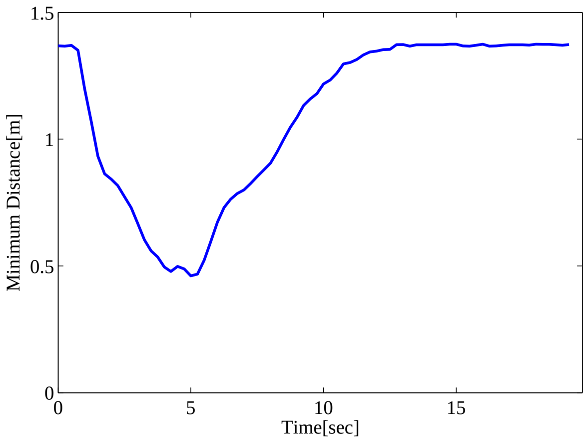



Fig. 3.27:Minimum distance among followers of Case 3 with collision avoidance

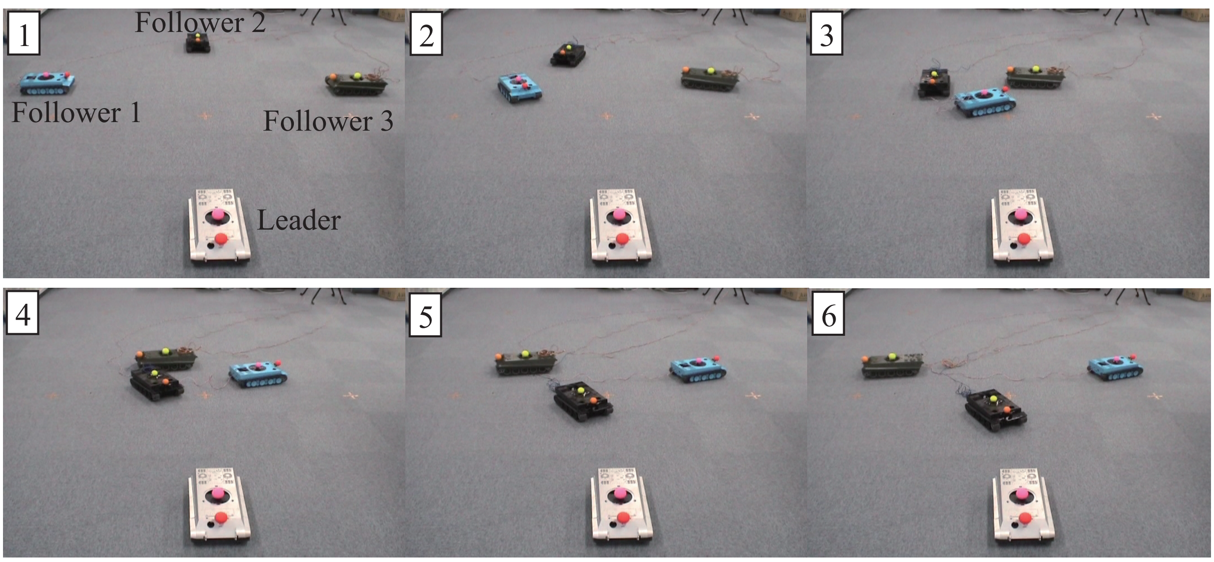



Fig. 3.28:The motion of followers in Case 3 with collision avoidance

### Case 4

Case 4の実験結果および実験風景をFig. 3.29～Fig. 3.33に示す．Fig. 3.29，Fig. 3.30はそれぞれ衝突回避アルゴリズム未適用の場合の移動体の軌跡，Follower間の最小距離を示したものである．Fig. 3.31，Fig. 3.32はそれぞれ衝突回避アルゴリズムを適用した場合の移動体の軌跡，Follower間の最小距離を示したものである．Fig 3.29，Fig 3.31の∘印は1[sec]毎の各Followerの位置をしたものである．Fig. 3.33は衝突回避アルゴリズムを適用した場合の動作の様子を示したものである．Fig. 3.33中の移動体の割り当てはCase 1と同様である．

Fig. 3.29，Fig. 3.30から衝突回避アルゴリズム未適用の場合では，移動体同士が衝突してしまい目標位置に収束することができないことが確認できる．一方，Fig. 3.31，Fig. 3.32から衝突回避アルゴリズムを適用した場合では，移動体同士の衝突が回避され目標位置に収束することができていることが確認できる．

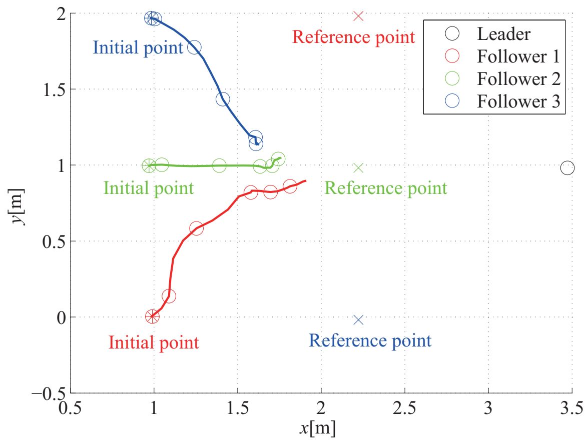



Fig. 3.29:XYplot of Case 4 without collision avoidance

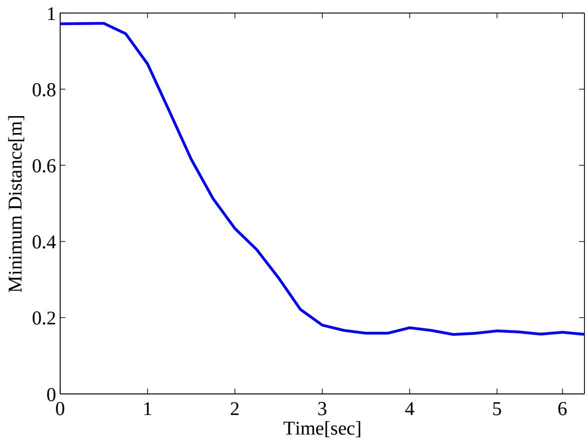
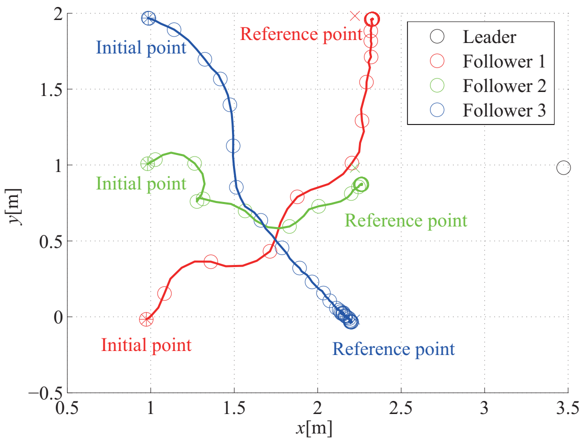



Fig. 3.30:Minimum distance among followers of Case 4 without coillision avoidance



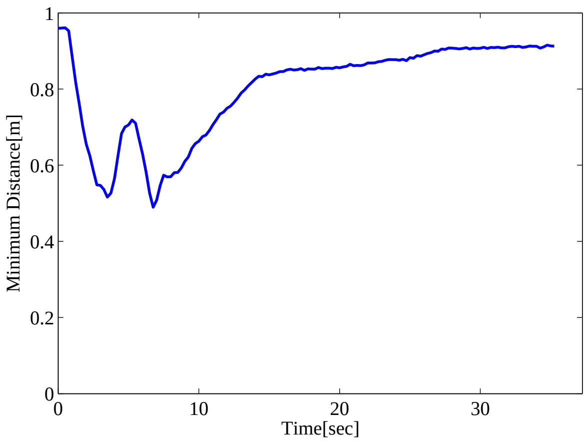




Fig. 3.31:XYplot of Case 4 with collision avoidance

Fig. 3.32:Minimum distance among followers of Case 4 with coillision avoidance

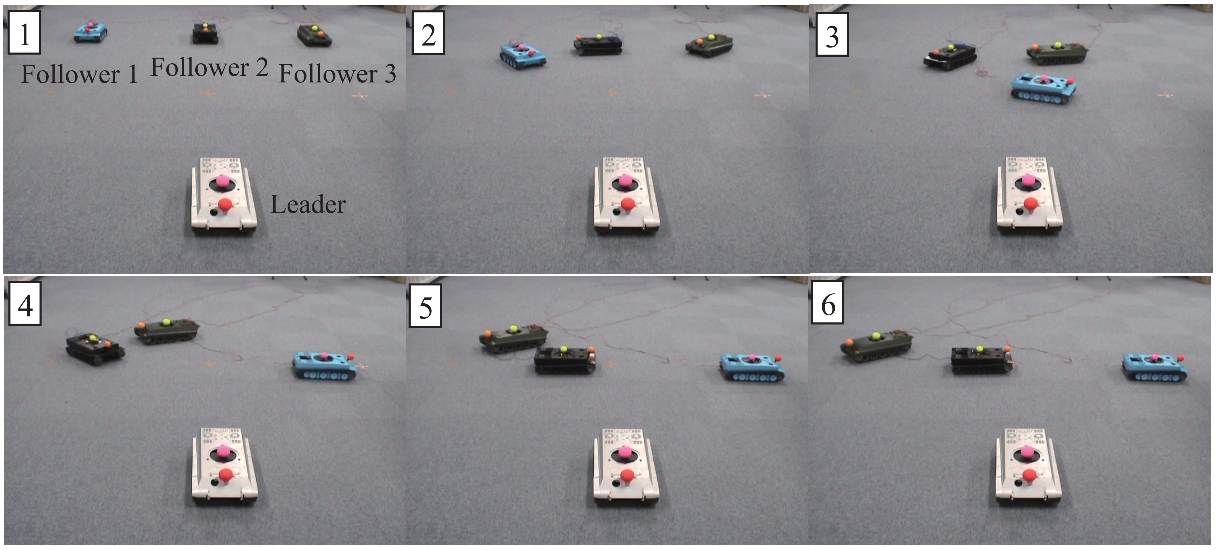



Fig. 3.33:The motion of followers in Case 4 with collision avoidance

### 3.4.4 考察

本節では，Followerが2台と3台の場合に対して行った実験結果を示した．各パターンとも衝突回避なしの場合には移動体同士の衝突が発生し，目標位置に収束することができなかった．一方，衝突回避ありの場合には移動体同士の衝突を回避し，目標の編隊を形成することができ，衝突回避アルゴリズムの有効性を確認することができた．特に従来の分枝限定法ではオンラインで解くことが困難であったFollowerが3台の場合でもオンラインで衝突回避が実現できることが示された．

その一方で次のような問題点も挙げられる．それは離散化の影響のため衝突回避制約を連続時間では満たさない部分が生じてしまっていることである．最適制御問題では離散的な予測時刻において衝突回避制約を満たす軌道を計画する．言い換えるとその予測時刻間では衝突回避は全く考慮していない．したがって，連続時間では侵害する場合が存在し，予測周期$\delta_m$を十分小さく設定できない場合にはこの影響を無視できなくなる．特に，本実験では予測間隔を$\delta_m = 1.0$[sec]と大きく設定していたため，この影響が如実に表れていたと考えられる．一例としてCase 1の場合のFollower 2は上記の影響が無ければ初期位置=目標位置であるため移動する必要が無いが，実際には多少移動していることがFig. 3.16からも確認できる．

本実験では上記の離散化の影響に対応するため，予測間隔より小さい周期$\delta = 0.25$[sec]で最適制御問題を再計算する手法を用いた．この手法ではFig. 3.2のように予測時刻が更新周期分ずれていくような最適制御問題を解くこととなり，衝突の危険性が低減化できる．ただし，予測を行っているにも関わらず，衝突が起きる直前に急に予測軌道が変更されるため，移動体の挙動に悪影響を与えてしまう可能性がある．特にCase 4のような複雑なパターンの場合，衝突を回避するため急に進路を変更したり，停止するなどの挙動が確認されている．

この問題を解消するひとつの方法は上述のとおり最適制御問題のサンプリングタイム$\delta_m$を十分小さし，離散化の影響を小さくすることである．しかしながら，可解性と安定性を保証するために導入している終端制約(3.14)により予測ホライズン$T$を長くとらなければならず，オンラインで解くためには$\delta_m$を十分小さくできない場合が存在する．したがって，最適制御問題のサンプリングタイムを十分小さく設定できない場合でも，予測時刻間の衝突回避も保証する制御方策が必要であることが本実験からも確認できる．

## 3.5 まとめ

本章では，複数移動体の衝突回避問題をモデル予測制御により定式化した際の計算量の増加の問題に着目し，計算量の低減化を考慮した衝突回避アルゴリズムを提案した．提案手法は新たな衝突回避のための最適制御問題の提案と，最適制御問題の一般的な解法である分枝限定法を改良する2点から構成されるものである．まず，提案した衝突回避のための最適制御問題では，Followerが順番に問題を解くような形式を導入することで，1つ1つの最適化の負荷を小さくするものである．また，そのときの最適制御問題の可解性と閉ループ系の安定性を解析し，最初のステップで全てのFollowerが最適解を持つことができれば目標の編隊形状が実現できることを示した．また，閉ループ系の入力$\alpha_i$と元々のシステムの入力$u_i$との関係を求め，$u_i$が飽和しないような制約を課した．次に衝突回避問題の特性を考慮した簡単な分枝，限定ルールを導入することで計算量を低減化を図る分枝限定法を提案した．また，シミュレーションにより，一般的な分枝限定法と比較し計算量が低減化できることを示した．さらに，これら計算量低減化アルゴリズムを用いて一般的な分枝限定法ではオンラインで実行することが困難であった，Followerが3台の場合について衝突回避アルゴリズムの有効性を実機により確認した．実験においては最適制御問題の離散化による影響が衝突回避に悪影響を与えていること確認された．このことからも次章で説明するような離散化を陽に考慮したアルゴリズムが必要であることが分かった．

今後の課題としては，最適制御問題を順番に解くのではなく各移動体が分散的・同時的に最適制御問題を解くような制御方策に改良することが挙げられる．また，実験においては1台のPCで解いたが，各移動体で最適制御問題を解く場合には時間遅れなどが生じる可能性がある．したがって，ロバスト性の向上を図っていくことも重要な課題であると考えられる．

# 第4章

# 離散化を陽に考慮したモデル予測制御に基づく衝突回避

前章のモデル予測制御に基づく衝突回避手法では，衝突回避のための不等式制約の下で最適な開ループ入力を求めるため，実装の際には離散時間の最適制御問題を解くことになる．すなわち，有限時間未来先までの離散時間の状態を予測し，その各予測時刻において衝突回避を考慮している．したがって，離散的な予測時刻でしか衝突回避を考慮していないため，得られた軌道は予測時刻間で衝突する可能性がある．特に，目標の状態への収束性を保証するには予測時間をある程度長く確保する必要があるため，計算時間の関係でサンプリング周期を十分短くできない場合があり，予測時刻間の衝突が深刻な問題となる．すなわち，本質的に離散化の影響を陽に考慮しなければ安全な衝突回避は実現できない．

本章では，最適制御問題の離散化がFollower同士の衝突回避に与える影響を考察し，その影響を陽に考慮した衝突回避手法を提案する．まず，問題を簡単化するため，1台の移動体に対して静止障害物を回避させながら目標位置まで誘導する静止障害物回避問題を考える．静止障害物回避は1台の移動体を除いて残りの全ての移動体が目標位置で停止している場合と同等の問題となるため，衝突回避問題の特殊なケースともとらえることができる．この問題に対して離散化が与える影響を明らかにし，予測時刻間においても障害物回避の保証を与える2つの手法を提案する．1つは障害物の近傍で移動体の最大移動速度を制限する方法である．もう1つは障害物回避制約の遷移を制限することで，予測時刻間で衝突が起きる可能性を排除する方法である．これらの手法ではいずれも最適化問題を繰り返し解くことなく，障害物と衝突しない軌道が得られるという利点をもつ．次にこの知見を用いて，Follower同士の衝突回避問題においても予測時刻間での衝突回避を保証する手法を提案する．また，前章の手法と同様の条件で最適制御問題の可解性や閉ループ系の安定性が保証されることも示す．それら提案手法の有効性を数値例と実験により検証する．

## 4.1　1台の移動体の障害物回避問題

本節では，Fig. 4.1のように1台の移動体が静止障害物を回避しながら目標位置に誘導する軌道計画問題を考える．ただし，静止障害物は2.2.2節のように定義されるものとする．このような静止障害物回避を考慮した軌道計画法は数多く報告されている(e.g. [71]-[75])．また，モデル予測制御に基づく軌道計画法も報告されている(e.g. [61],[65, 66])．2.2.2節で述べたように静止障害物回避も移動体同士の衝突回避と同様に0, 1のみを取る変数(0, 1変数)を含む不等式制約で表現でき，障害物回避問題も混合整数計画問題として定式化される．

静止障害物回避に対する離散化の影響を陽に考慮した例としては参考文献[65, 66]などが報告されている．参考文献[65]では，予測時刻間でも障害物と衝突しないような軌道を計画するため，障害物を仮想的に大きくするものであり，軌道が必要以上に保守的になってしまうという問題点が残されている．また，参考文献[66]では予測時刻間での障害物回避を保証する2つの手法を提案している．1つは，衝突が予測される予測時刻間のサンプリングタイムを小さくするという手順を繰り返すものである．もう1つは，予測時刻間で衝突が起こる障害物のみ仮想的に拡大するという手順を繰り返す手法である．これらの手法では衝突の起きない軌道が得られるまで繰り返し混合整数計画問題を解く必要があるため，問題によっては大きな計算時間を要する可能性がある．

まず，この問題において離散化による影響がどの程度障害物回避に影響するかを明らかにする．また，その性質を考慮し，離散化を行っても連続時間上で障害物回避を保証する2つの方法，“可変最大速度法”と“遷移制約法”を提案する．これらの手法は，従来手法とは異なり，いずれも最適化問題を繰り返し解くことなく，障害物と衝突しない軌道が得られるという利点をもつ．

### 4.1.1　離散化が与える障害物回避への影響

ここでは1台の移動体のみ考えるので予測モデルとして(2.24)式の代わりに以下を用いるものとする．

$$\dot{e} = \alpha \tag{4.1}$$

ただし，$e := z_v - z^d$である[1]．また，(2.24)式に含まれていたフィードバック項$-\lambda e$は議論の簡単化のため除くものとする．また，目標位置$z_d$は絶対座標系

[1]表記の簡単化のため，4.1節の障害物回避問題においては変数から移動体の番号を示す添え字$i$は除く．

で固定であるので移動体への制御入力として (2.23) 式ではなく，Leader の挙動に関するフィードバック項を除いた以下を用いるものとする．

$$u = G^{-1}\alpha \tag{4.2}$$

このとき，障害物回避を行うため，移動体はサンプリングタイム $\delta$ 毎に以下に示す最適制御問題を解き，その最初の成分 $\alpha(k)$ を適用する．時刻 $t = k\delta$ における障害物回避を考慮した軌道計画問題 $P_2$ は以下のような混合整数二次計画問題として定式化することができる[2]．

軌道計画のための最適制御問題 $P_2$

$$\min_{\widehat{\alpha}} \sum_{\tau=k+1}^{k+N} \left(\widehat{e}(\tau|k)^T Q\widehat{e}(\tau|k) + \widehat{\alpha}(\tau-1|k)^T R\widehat{\alpha}(\tau-1|k)\right) \tag{4.3}$$

subject to

$$\widehat{e}(\tau|k) = \widehat{e}(\tau-1|k) + \delta_m\widehat{\alpha}(\tau-1|k),\ \widehat{e}(k|k) = e(t) \tag{4.4}$$

$$||D_{\psi_i}^{-1}(\widehat{e}(\tau|k) + z_d - o_i)||_\infty \geq 1 \tag{4.5}$$

$$||\widehat{\alpha}(\tau-1|k)||_\infty \leq \eta \tag{4.6}$$

$$||\widehat{e}(k+N|k)||_\infty \leq \gamma \tag{4.7}$$

$$i = 1,\ \ldots,\ m,\ \ \tau = k+1,\ k+2,\ \ldots,\ k+N$$

ただし，$N$ は予測ステップであり，$\widehat{e}(\tau|k)$ は時刻 $t = k\delta$ における $e(\tau\delta)$ の予測値を表している．また，$Q \in \mathbb{R}^{2N\times 2N}$，$R \in \mathbb{R}^{2N\times 2N}$ はそれぞれ対称行列，正定対称行列である．

(4.4) 式は $\widehat{e}$ の予測モデルであり，(4.1) 式を零次ホールドによりサンプリングタイム $\delta_m$ で離散化したものである．なお，議論の簡単化のため，$\delta = \delta_m$ とする．また，不等式制約 (4.5) は (2.9) 式の障害物回避制約で，(2.11) 式または (3.47) 式の形式に変換される．不等式制約 (4.6) は入力を制限するために導入した制約である．不等式制約 (4.7) は軌道の収束性を保証するための終端状態への制約である．

最適制御問題 $P_2$ では各予測時刻において障害物回避制約が課されている．すなわち，離散的な間隔でしか障害物回避を考慮していないため，予測時刻間では障害物との衝突が発生する可能性がある．実際に従来のモデル予測制御手法を適用したシミュレーション結果を Fig. 4.2 に示す．Fig. 4.2 の丸印は予測を行った時刻での移動体の位置であり，これら予測時刻では障害物回避制約を満足しているが，予測時刻間では障害物と衝突が起きていること

[2]閉ループ系の安定性を保証するには評価関数に適切な終端コストを導入すればよい [76]．本節では説明の簡単化のため，終端コストは導入していない．

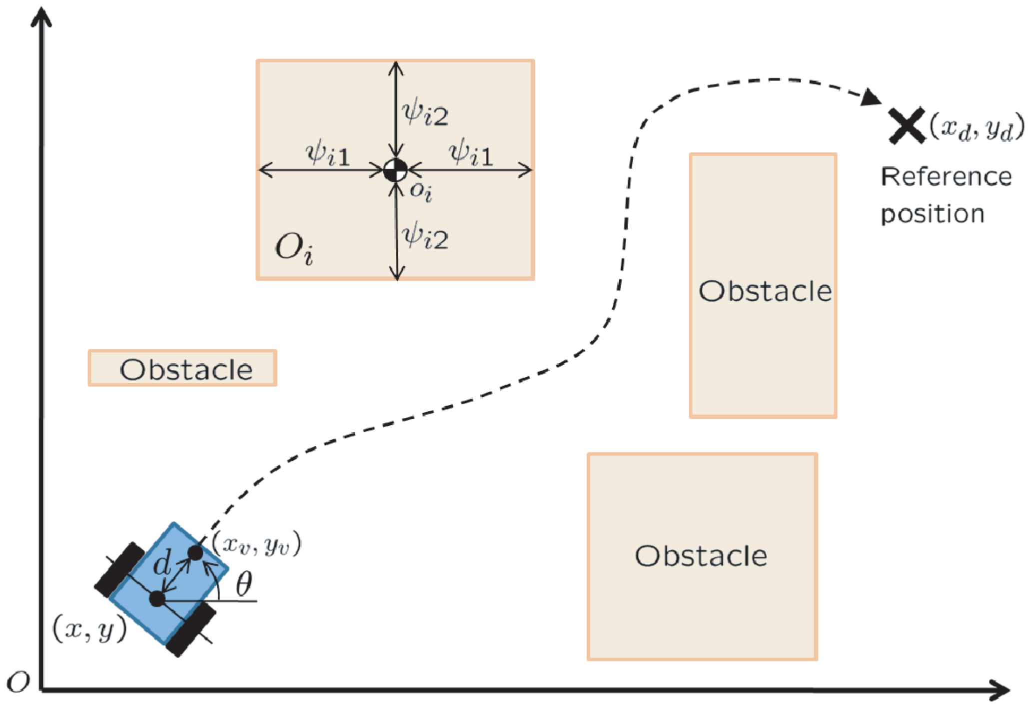



Fig. 4.1:Trajectory generation for single vehicle with obstacle avoidance

が確認できる．

この問題を解決する方法の1つに，仮想的に障害物の大きさを拡大することで元々の障害物回避制約が予測時刻間でも満足されるようにする手法がある [65]．これは，Fig. 4.3 のように仮想的に障害物を一定量$\beta$だけ拡大するものである．ここで，拡大量$\beta$が満たすべき条件を次の補題に示す.

**補題 2** 予測時刻$t = \tau\delta(\tau = k+1, k+2, \dots, k+N)$において障害物回避制約 (4.5) が満たされるものとする．このとき，予測時刻間で障害物と衝突しないための拡大量$\beta$の十分条件は以下のように得られる．

$$\beta > \max\left(\frac{\delta\eta}{2}, \Phi\right) \tag{4.8}$$
$$\Phi := \max_i \left(\frac{\sqrt{2}\delta\eta - 2\min(\psi_{i1}, \psi_{i2})}{2}\right), \quad i = 1, \dots, m$$

**証明：** 予測時刻間の入力$\alpha$は一定であるため，(4.1) 式の予測時刻間の軌道は予測時刻間を結ぶ線分となる．この線分が，拡大した長方形と交わる最小の長さを$\Delta l_{min}$と定義する．この$\Delta l_{min}$には障害物の大きさに応じて，2通りの場合が存在する．1つは Fig. 4.3(a) のように拡大前の障害物の頂点を通る場合である．もう1つは Fig. 4.3(b) のように拡大前の障害物の内側を通過する場合

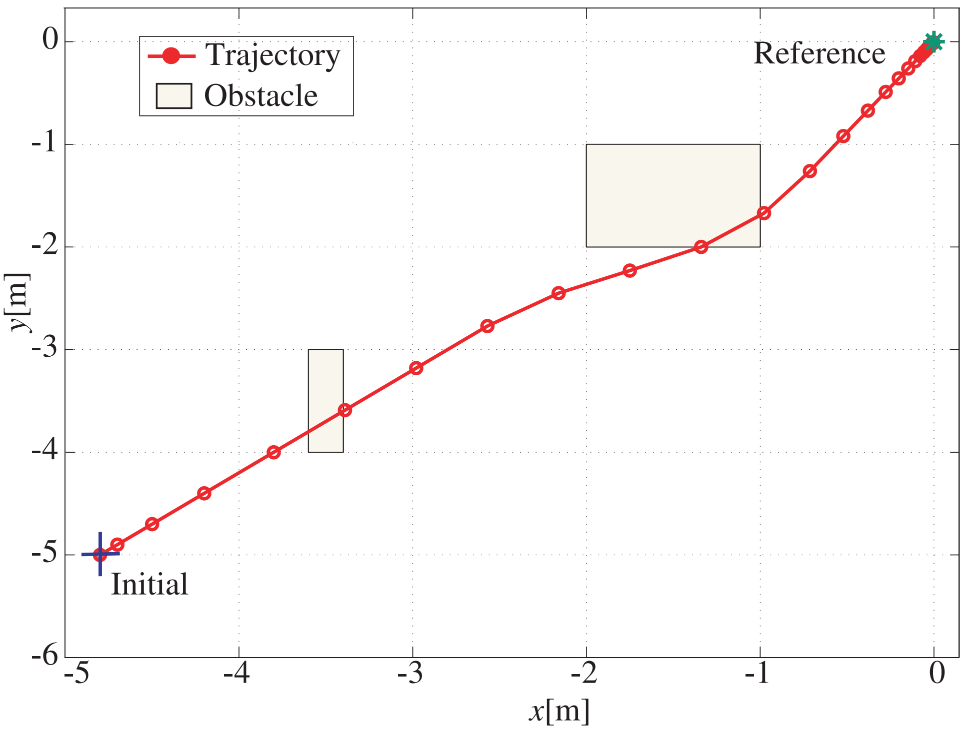



Fig. 4.2:Example of bad trajectory generation

である．仮定より，各予測時刻 $t = \tau\delta$ での移動体の予測位置は拡大した長方形の外側に存在するため，$\Delta l_{min}$ が 1 サンプリングタイム $\delta$ で移動可能な最大距離 $\Delta l = \delta\sqrt{\eta^2 + \eta^2} = \sqrt{2}\delta\eta$ より大きければ予測時刻間での障害物回避が保証される．

まず，Fig. 4.3(a) の場合を考える．この場合，$\Delta l_{min} = 2\sqrt{2}\beta$ となることから，$\Delta l_{min} \geq \Delta l$ より

$$\beta \geq \frac{\delta\eta}{2} \tag{4.9}$$

が十分条件となる．同様に，Fig. 4.3(b) の場合には，

$$\Delta l_{min} = 2\beta + 2\min(\psi_{i1},\ \psi_{i2}) \tag{4.10}$$

となるため，$\Delta l_{min} > \Delta l$ より

$$\beta > \frac{\sqrt{2}\delta\eta - 2\min(\psi_{i1},\ \psi_{i2})}{2} \tag{4.11}$$

が十分条件となる．

以上より全ての障害物に対して (4.9), (4.11) 式が満たされれば衝突が起きないため，(4.8) 式が $\beta$ の十分条件となる．　□

仮想的に障害物を拡大する手法では，大きな拡大量$\beta$が全ての障害物に適用された場合，得られる軌道が極度に保守的になってしまうという問題点が存在する．補題1で示したように拡大量$\beta$は，サンプリングタイムと最大速度に依存しているため，これらの値を小さくすることで拡大量の低減化は可能である．しかし，収束性を保証する終端制約(4.7)により，予測区間の終端には目標位置近傍に到達しなくてはならず，必ずしも十分にサンプリングタイムや最大速度を小さく設定できない場合が存在する．したがって，このような場合でも保守性を低く抑えながらも陽に障害物回避の保証を与える手法が必要となる．一方，参考文献[66]では衝突が予測される部分のサンプリングタイムを次第に小さくする方法と，衝突が予測される障害物を仮想的に拡大していく手法を提案している．これらの手法では衝突の起きない軌道が得られるまで混合整数計画問題を繰り返し解く必要があるため，問題によっては大きな計算時間を要する可能性がある．次節では，混合整数計画問題を繰り返し解くことなく，予測時刻間での障害物回避を保証する2つの手法を提案する．

### 4.1.2 可変最大速度法

本節で提案する可変最大速度法は障害物の拡大量を動的に変化させるものである．本手法では予測時刻間での障害物回避を保証するため，障害物の拡大量に応じて移動体の最大速度$\eta$も補題2に基づき動的に変化させる．

具体的には，補題2を満たす拡大量$\beta$を用い，障害物回避制約(4.5)を以下のように修正する．

$$\|D_{\psi_i}^{-1}(\widehat{\nu}(\tau|k))(\widehat{e}(\tau|k) + z_d - o_i)\|_\infty \geq 1 \tag{4.12}$$
$$D_{\psi_i}(\widehat{\nu}) := D_{\psi_i} + \beta(1 - \widehat{\nu})I_2$$

ただし，$\hat{\nu}(\tau|k) \in [0, 1]$は障害物の拡大量を調整するために導入した変数である．(4.12)式から長方形の辺の長さは$\widehat{\nu}$に応じて$2\beta(1 - \widehat{\nu}(\tau|k))$増加することになる．また，(4.12)式により障害物の拡大量が変化するため，$t \in ((\tau - 1)\delta, \tau\delta)$における最大速度，すなわち$\widehat{\alpha}(\tau - 1|k)$も補題2に基づいて修正される必要がある．ただし，補題2とは異なり，障害物の大きさが予測時刻$(\tau - 1)\delta$，$\tau\delta$で変わる可能性があるため，入力制約は以下のように，より拡大量が小さい場合に対応させる必要がある．

$$\|\widehat{\alpha}(\tau - 1|k)\|_\infty \leq \eta(1 - \max\{\widehat{\nu}(\tau - 1|k), \widehat{\nu}(\tau|k)\}) \tag{4.13}$$

さらに，軌道が過度に保守的になるのを防ぐため，(4.3)式の評価関数を考慮して$\hat{\nu}$を決定する．

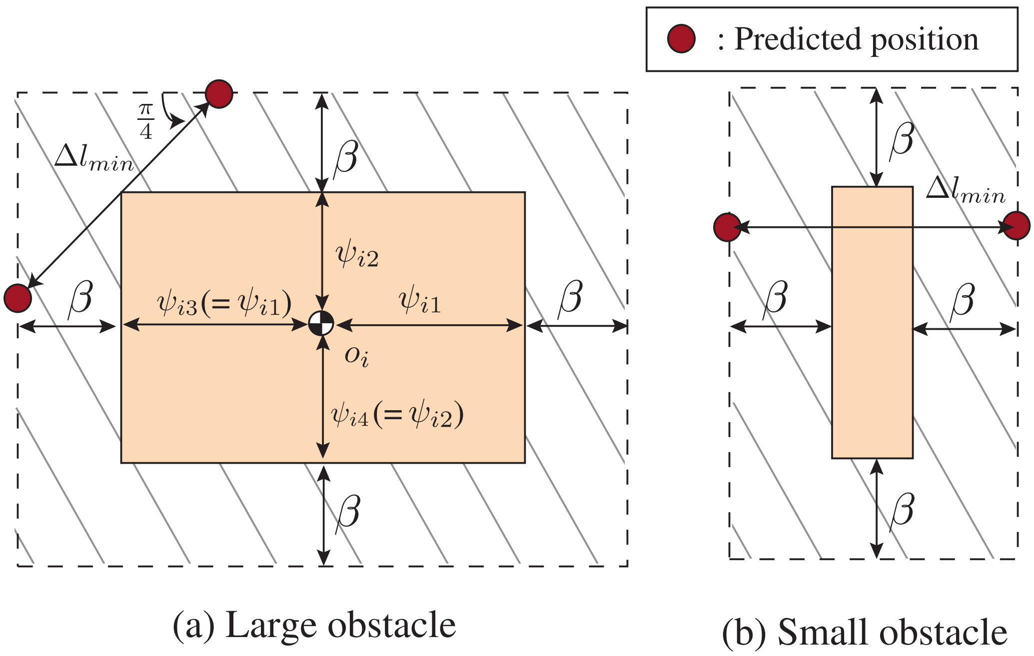



Fig. 4.3:Enlarged obstacle (shaded area) and $\Delta l_{min}$

以上より時刻 $t = k\delta$ における可変最大速度法に基づく軌道計画問題 $P_3$ は以下のように表わされる．

可変最大速度法に基づく最適制御問題 $P_3$

$$\min_{\widehat{\alpha},\widehat{\nu}} \sum_{\tau=k+1}^{k+N} \Big(\widehat{e}(\tau|k)^T Q\widehat{e}(\tau|k) + \widehat{\alpha}(\tau-1|k)^T R\widehat{\alpha}(\tau-1|k)$$
$$+\widehat{\nu}(\tau-1|k)^T S\widehat{\nu}(\tau-1|k)\Big) \tag{4.14}$$

subject to $eq.$ (4.4), (4.7)

$$\|D_{\psi_i}^{-1}(\widehat{\nu}(\tau|k))(\widehat{e}(\tau|k) + z_d - o_i)\|_\infty \geq 1 \tag{4.15}$$
$$\|\widehat{\alpha}(\tau-1|k)\|_\infty \leq \eta(1-\widehat{\nu}(\tau-1|k)) \tag{4.16}$$
$$\|\widehat{\alpha}(\tau-1|k)\|_\infty \leq \eta(1-\widehat{\nu}(\tau|k)) \tag{4.17}$$
$$0 \leq \widehat{\nu}(\tau-1|k) \leq 1 \tag{4.18}$$
$$i = 1, \ \ldots, m, \quad \tau = k+1, \ k+2, \ \ldots, k+N$$

ただし，$\widehat{\nu}(k|k)$ は $e(k\delta)$ から決定される値で，$S \in \mathbb{R}^{N\times N}$ は対称行列である．不等式制約 (4.15) は (4.12) 式の障害物回避制約であり，不等式制約 (4.16)，(4.17) は (4.13) 式と等価である．

ここで，可変最大速度法について以下の定理が得られる．

**定理3** 時刻$t = k\delta$において$z_v$が障害物回避制約を満たすものとする．また，障害物の拡大量$\beta$が(4.8)式を満たすとする．このとき，最適制御問題$P_3$の実行可能解は$t \in [k\delta, (k+N)\delta]$において障害物と衝突しない軌道となる．

**証明：** 仮定より時刻$t = \tau\delta$ ($\tau = k+1,\ k+2,\ \ldots,\ k+N$) および$t = k\delta$において衝突は起きない．したがって，時刻$t \in ((\tau-1)\delta,\ \tau\delta)$において障害物との衝突が起きないことを示せばよい．

補題2の証明と同様にFig. 4.3(a)(b)の場合に対してそれぞれ$\Delta l_{min} \geq \Delta l,\ \Delta l_{min} > \Delta l$を示せば，時刻$t \in ((\tau-1)\delta, \tau\delta)$での障害物回避が保証される．ただし，可変最大速度法では障害物回避制約(4.15)より時刻$(\tau-1)\delta,\ \tau\delta$での障害物の拡大量が異なる可能性がある．したがって，$\beta(1-\widehat{\nu}(\tau-1|k))$と$\beta(1-\widehat{\nu}(\tau|k))$の両方に対して$\Delta l_{min} \geq \Delta l$(または$\Delta l_{min} > \Delta l$)が満たされることを示す必要がある．ここで，$\Delta l_{min}(\tau-1)$と$\Delta l_{min}(\tau)$をそれぞれ$\beta(1-\widehat{\nu}(\tau-1|k)),\ \beta(1-\widehat{\nu}(\tau|k))$に対応する$\Delta l_{min}$の値とする．ここでは，一般性を失うことなく$\widehat{\nu}(\tau|k) \geq \widehat{\nu}(\tau-1|k)$と仮定する．このとき，入力制約(4.13)より移動体の最大速度が$\eta(1-\widehat{\nu}(\tau|k))$であるため，1サンプリングタイムで移動可能な最大距離は$\Delta l = \sqrt{2}\delta\eta(1-\widehat{\nu}(\tau|k))$となる．

まず，Fig. 4.3(a)の場合には，

$$\Delta l_{min}(\tau-1) \geq \Delta l_{min}(\tau) = 2\sqrt{2}\beta(1-\widehat{\nu}(\tau|k)) \tag{4.19}$$

となるため，$\Delta l_{min}(\tau) \geq \Delta l$を示せばよい．仮定より$\beta > \delta\eta/2$であり，(4.18)式より$\widehat{\nu}(\tau|k) \leq 1$であるため

$$\Delta l_{min}(\tau) - \Delta l = \sqrt{2}(2\beta - \eta\delta)(1-\widehat{\nu}(\tau|k)) \geq 0 \tag{4.20}$$

となる．同様にFig. 4.3(b)の場合には，

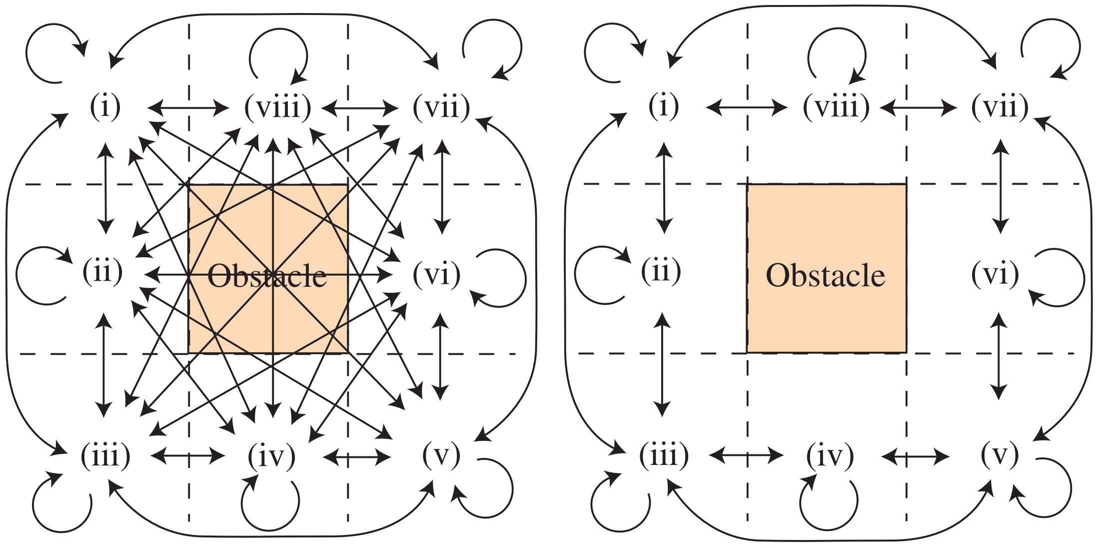
$$\begin{aligned}\Delta l_{min}(\tau-1) &\geq \Delta l_{min}(\tau)\\ &= 2\beta(1-\widehat{\nu}(\tau|k)) + 2\min(\psi_{i1},\ \psi_{i2})\end{aligned} \tag{4.21}$$

となるため，$\Delta l_{min}(\tau) > \Delta l$を示せばよい．仮定より(4.11)式が満たされ，(4.18)式より$0 \leq \widehat{\nu}(\tau|k) \leq 1$であるため

$$\begin{aligned}\Delta l_{min}(\tau) - \Delta l = &\ 2\widehat{\nu}(\tau|k)\min(\psi_{i1},\ \psi_{i2})\\ &+ \left(2\beta - \sqrt{2}\eta\delta + 2\min(\psi_{i1},\ \psi_{i2})\right)(1-\widehat{\nu}(\tau|k)) > 0\end{aligned}$$

となる．なお，$\widehat{\nu}(\tau|k) \leq \widehat{\nu}(\tau-1|k)$の場合には，以上の式中の$\widehat{\nu}(\tau|k)$を$\widehat{\nu}(\tau-1|k)$に替え，(4.19)式と(4.21)式の不等式$\Delta l_{min}(\tau-1) \geq \Delta l_{min}(\tau)$を$\Delta l_{min}(\tau) \geq \Delta l_{min}(\tau-1)$に替えることによって同様に示される．したがって時刻$t \in ((\tau-1)\delta, \tau\delta)$での障害物回避が保証される．

以上より，$t \in [k\delta, (k+N)\delta]$において障害物と衝突しないことが示された．□




(a) Standard (b) Proposed

Fig. 4.4:Transition of the obstacle avoidance constraints

**注意2** 可変最大速度法はより複雑な形状の障害物に対しても拡張は可能である．拡張する場合には，補題2のように$\beta$が満たすべき十分条件を導出することが必要となる．$\beta$が満たすべき十分条件が求まれば，同様の議論で適用が可能となる．

### 4.1.3 遷移制約法

次に，有効となる障害物回避制約の予測時刻間での遷移に着目した手法を提案する．

2.2.2節で述べたように，障害物回避制約(2.11)には0,1変数の組み合わせに応じて8つのモードが存在する．また，予測時刻間でのモード変化については制約が存在しないため，Fig. 4.4(a)のようにすべてのモード間で遷移が可能である．ここで，問題となるのがこれらのモード遷移の中に障害物と衝突するものが含まれていることである．例えば，Fig. 4.4(a)の(ii)→(viii)や(ii)→(vi)では，その過程で障害物と衝突する危険性がある．

遷移制約法は，障害物回避制約(2.11)の$\kappa_{ij}$の変化に制約を加えることでFig. 4.4(b)に示すような安全な遷移条件のみ選択されるようにするものである．例えば，$\kappa_{i1} = 0, \kappa_{ij} = 1\ (j \neq 1)$に対応する領域(ii)に移動体が存在する場合には，同様に$\kappa_{i1} = 0$である領域(i)，(ii)，(iii)のいずれかに移動が可能となる．$\kappa_{i1} = \kappa_{i4} = 0, \kappa_{i2} = \kappa_{i3} = 1$に対応する領域(i)に移動体が存在する場合には，$\kappa_{i1} = 0$

または$\kappa_{i4}=0$である領域(i), (ii), (iii), (vii), (viii)のいずれかに移動が可能となる．言い換えると，Fig. 4.4(b)の遷移条件は以下のように表すことができる．

$$\tilde{\kappa}_{ij}=\kappa_{ij}=0, \quad \exists j\,(=1,2,3,4) \tag{4.22}$$

ただし，$\tilde{\kappa}_{ij}$は1つ前の予測時刻での$\kappa_{ij}$の値を表したものである．この条件は以下の不等式制約により実現ができる．

$$\begin{aligned}\tilde{\kappa}_{i1}+\kappa_{i1} &\le M\rho_{i1}+M\rho_{i2}\\ \tilde{\kappa}_{i2}+\kappa_{i2} &\le -M\rho_{i1}+M\rho_{i2}+M\\ \tilde{\kappa}_{i3}+\kappa_{i3} &\le M\rho_{i1}-M\rho_{i2}+M\\ \tilde{\kappa}_{i4}+\kappa_{i4} &\le -M\rho_{i1}-M\rho_{i2}+2M\end{aligned} \tag{4.23}$$

ただし，$\rho_{i1}, \rho_{i2}$は新たに導入した0, 1変数である．

以上より時刻$t=k\delta$における遷移制約法に基づく軌道計画問題$P_4$は以下のように表わされる．

遷移制約法に基づく最適制御問題$P_4$

$$\min_{\widehat{\alpha}} \sum_{\tau=k+1}^{k+N}\left(\widehat{e}(\tau|k)^T Q\widehat{e}(\tau|k)+\widehat{\alpha}(\tau-1|k)^T R\widehat{\alpha}(\tau-1|k)\right) \tag{4.24}$$

subject to $eq.$ (4.4), (4.6), (4.7)

$$\begin{bmatrix}\widehat{e}(\tau|k)+z_d-o_i\\ -\widehat{e}(\tau|k)-z_d+o_i\end{bmatrix} \le M\begin{bmatrix}\kappa_{i1}(\tau|k)\\ \kappa_{i3}(\tau|k)\\ \kappa_{i2}(\tau|k)\\ \kappa_{i4}(\tau|k)\end{bmatrix}-\begin{bmatrix}\psi_{i1}\\ \psi_{i2}\\ \psi_{i1}\\ \psi_{i2}\end{bmatrix} \tag{4.25}$$

$$\sum_{j=1}^{4}\kappa_{ij}(\tau|k)\le 3$$

$$\begin{bmatrix}\kappa_{i1}(\tau-1|k)+\kappa_{i1}(\tau|k)\\ \kappa_{i2}(\tau-1|k)+\kappa_{i2}(\tau|k)\\ \kappa_{i3}(\tau-1|k)+\kappa_{i3}(\tau|k)\\ \kappa_{i4}(\tau-1|k)+\kappa_{i4}(\tau|k)\end{bmatrix} \le \begin{bmatrix}M & M\\ -M & M\\ M & -M\\ -M & -M\end{bmatrix}\begin{bmatrix}\rho_{i1}(\tau|k)\\ \rho_{i2}(\tau|k)\end{bmatrix}+\begin{bmatrix}0\\ M\\ M\\ 2M\end{bmatrix} \tag{4.26}$$

$$i=1,\ \ldots,\ m,\ \ \tau=k+1,\ k+2,\ \ldots,\ k+N$$

ただし，$\kappa_{ij}(k|k)$は$\widehat{e}(k\delta)$が(4.25)式を満足するように決定される値である．不等式制約(4.25)は(2.11)式の形式で表された障害物回避制約で，不等式制約(4.26)は(4.23)式の遷移条件に関する制約である．

ここで，遷移制約法について以下の定理が得られる．

**定理 4** 時刻 $t = k\delta$ において $z_v$ が障害物回避制約を満たすものとする．このとき，最適制御問題 $P_4$ の実行可能解は $t \in [k\delta, (k+N)\delta]$ において障害物と衝突しない軌道となる．

**証明：** 最適制御問題 $P_4$ の実行可能解は時刻 $t = \tau\delta$ において障害物回避制約を満たす．また，仮定より初期時刻 $t = k\delta$ においても障害物回避制約を満たす．さらに，不等式制約 (4.26) より，時刻 $t = (\tau-1)\delta$ から $t = \tau\delta$ での領域の遷移は Fig. 4.4(b) のどれかに制限される．ここで $t \in [(\tau-1)\delta, \tau\delta]$ における挙動は前述のように両端を結ぶ線分となることから，$t \in [(\tau-1)\delta, \tau\delta]$ においても障害物回避制約を満たす.

以上より，$t \in [k\delta, (k+N)\delta]$ において障害物と衝突しないことが示された．□

**注意 3** 本章では長方形状の障害物のみ取り扱っているが，本節の遷移制約法は容易に凸 $n_i$ 多角形形状の障害物 [64] に拡張することが可能である．これは，遷移制約 (4.22) を以下のように変更することで実現できる．

$$\tilde{\kappa}_{ij} = \kappa_{ij} = 0, \quad \exists j\ (= 1, \ldots, n_i) \tag{4.27}$$

### 4.1.4　可変最大速度法と遷移制約法の比較

可変最大速度法では，障害物近傍を速度を抑えて走行するため，より安全な軌道を計画することが可能であるといえる．本手法では $\hat{v}$ に対して重み $S$ を評価関数に組み込んでおり，$S$ を適切に設定することで，障害物の近傍を通りやすいか，離れて走行するかを変更することができると考えられる．すなわち軌道の安全性を調整することが可能であると考えられる．ただし，サンプリングタイムが大きくなるにつれて速度が制限される領域が大きくなってしまうため，サンプリングタイムを極度に大きくした場合には常に低速度でしか走行できなくなってしまう可能性がある．

遷移制約法では，障害物を拡大する必要がないため，サンプリングタイムの影響を受けない．一方，衝突の危険性のある遷移条件をすべて削除しているため，保守的になる場合が存在する．例えば，Fig. 4.4 において衝突することなく (ii)→(viii) と遷移できる場合でも (ii)→(i)→(viii) の経路が選択されることとなる．また，障害物回避制約の遷移条件に 0, 1 変数を用いているため最大速度制限法と比較し，計算量が増大する可能性があるが，モード遷移ルール [77] などを活用することで計算量の低減化が可能であると考えられる．以上より，遷移制約法は大まかに目標までの経路を求めたい場合などに適した手法であると考えられる．

## 4.2 複数移動体の衝突回避問題

次に，編隊形状切り替えに伴うFollower同士の衝突回避問題に対する離散化の影響を考慮した制御手法を考える．

衝突回避問題の場合，回避対象である他Followerの挙動も影響を与えるため静止障害物回避と比較し，より取り扱いが困難となる．すなわち，衝突回避問題の場合，回避対象のFollowerの挙動を最適制御問題で考慮することが必要である．次節では，回避対象のFollowerの挙動を考慮する1つの方法を示し，その問題点を明らかにする．また，その問題点を避けるため，前節で提案した遷移制約法に基づく衝突回避手法を提案する．

### 4.2.1 予測時刻間の衝突回避

3.1.1節で提案した制御手法では各Followerが順番に最適制御問題を解いていく．そのため，Follower $i$ の最適制御問題においてはFollower $j(\forall j \neq i)$ の予測位置情報を持っていることになる．したがって，Follower $j$ の予測位置情報を用いて，Follower $j$ の挙動が衝突回避に与える影響を考慮すること可能である．具体的には連続する予測時刻間の衝突回避領域を覆う拡大衝突回避領域を作成し，新たにその領域を回避するような制約を課すことが考えられる．ここで，拡大衝突回避領域は，以下のように定義されるものである．

$$\widehat{\mathrm{CO}}_j(\tau|k) \supseteq P_j(\tau|k),\ P_j(\tau|k) := \left\{(x,y) \mid \cup CO_i(\frac{\hat{t}}{\delta} \mid k), \hat{t} \in [(\tau-1)\delta, \tau\delta]\right\} \quad \forall j \neq i,\ \tau = k+1,\ldots,k+N \tag{4.28}$$

ただし，$CO_j(\frac{\hat{t}}{\delta}|k)$ は予測時刻 $\hat{t}$ におけるFollower $j$ の衝突回避領域を示すものとする，また，$\widehat{CO_j}$ は拡大衝突回避領域を示し，連続する予測時刻間の衝突回避領域の和集合 $P_j(\tau|k)$ の上位集合である．障害物の予測時刻間での挙動が線形である場合，時刻 $\tau\delta, (\tau+1)\delta$ での衝突回避領域の計8頂点の凸包[78]を計算することで $P_j(\tau|k)$ を得ることができる. この場合，Follower $j$ の挙動に応じて $P_j$ は長方形状または6角形状となる(Fig. 4.5参照). 衝突回避制約(2.6)と同じく拡大衝突回避領域 $\widehat{CO_j}$ を長方形状で表現した場合にはFig. 4.6のように表わされる．ただし，その長方領域の重心位置 $\widehat{co}_j$ と辺の長さ $(2\widehat{\psi}_{cj1}, 2\widehat{\psi}_{cj2})$ を以下のように定義する．

$$\widehat{co}_j(\tau|k) := [\widehat{x}_{cj}(\tau|k), \widehat{y}_{cj}(\tau|k)]^T = \frac{1}{2}\Big(z_j(\tau-1|k) + z_j(\tau|k)\Big) \tag{4.29}$$

$$\widehat{\psi}_{cj1}(\tau|k) := \Big|x_j(\tau|k) - \widehat{x}_{cj}(\tau|k)\Big| + \psi \tag{4.30}$$

$$\widehat{\psi}_{cj2}(\tau|k) := \Big|y_j(\tau|k) - \widehat{y}_{cj}(\tau|k)\Big| + \psi \tag{4.31}$$

$$\forall j \neq i,\ \tau = k+1,\ldots,k+N$$

他Followerの挙動による影響を考慮するには，予測時刻$\tau\delta$に対して，$\widehat{CO}_j(\tau-1|k)$，$\widehat{CO}_j(\tau|k)$の計$2m$個の拡大衝突回避領域との回避制約を新たに課す必要がある．すなわち，時刻$t=k\delta$における最適制御問題には以下の衝突回避制約が課されることとなる．

$$\begin{aligned} |x_{vi}(\tau|k)-\widehat{x}_{cj}(\tau-1|k)| &\leq \widehat{\psi}_{cj1}(\tau-1|k) \\ \text{or}\quad |y_{vi}(\tau|k)-\widehat{y}_{cj}(\tau-1|k)| &\leq \widehat{\psi}_{cj2}(\tau-1|k) \end{aligned} \tag{4.32}$$

$$\begin{aligned} |x_{vi}(\tau|k)-\widehat{x}_{cj}(\tau|k)| &\leq \widehat{\psi}_{cj1}(\tau|k) \\ \text{or}\quad |y_{vi}(\tau|k)-\widehat{y}_{cj}(\tau|k)| &\leq \widehat{\psi}_{cj2}(\tau|k) \end{aligned} \tag{4.33}$$

$$\forall j \neq i,\ \tau = k+1,\ldots,k+N$$

しかしながら，この方法では，回避対象の挙動によっては回避領域が大きくなり，過度に保守的な設計となってしまう可能性がある．また，各予測時刻に対して2倍の衝突回避制約が課されるため計算量が増加してしまう可能性がある．さらに，最適制御問題$P_0$のような一度に全ての移動体の入力を決定する問題に対しては適用することができない点もデメリットとして挙げられる．

上記の問題を避けるため，次節では遷移制約法と予測モデルの性質を活用し，予測時刻間での衝突回避を保証する手法を提案する．

### 4.2.2 遷移制約法に基づく衝突回避手法

本節では最適制御問題$P_1$を4.1.3節で提案した遷移制約法に基づき修正することで，予測時刻間でも衝突回避の保証を与え，本質的に安全な軌道を計画できるようにする．また，修正した最適制御問題においてもその可解性や閉ループ系の安定性について最適制御問題$P_1$と同等の条件のもと保証されることを示す.

まず, 4.1.3節の遷移制約法は予測モデルの性質を利用することで予測時刻間での障害物回避を保証した．そこで，Follower同士の予測時刻間での衝突回避の保証を与えるため，(2.27)式のモデルを用いるように修正し，フィードバック項$-\lambda\tilde{e}_i$を除いた以下の式を予測モデルとして用いるものとする．

$$\dot{\tilde{e}}_i = \tilde{\alpha}_i \tag{4.34}$$

また，(2.31)式からフィードバック項$-\lambda\tilde{e}_i$を除いた以下の式をFollowerへの制御入力として用いるようにする．

$$u_i = G_i^{-1}(\tilde{E}_i u_r + \tilde{F}_i \tilde{\alpha}_i) \tag{4.35}$$

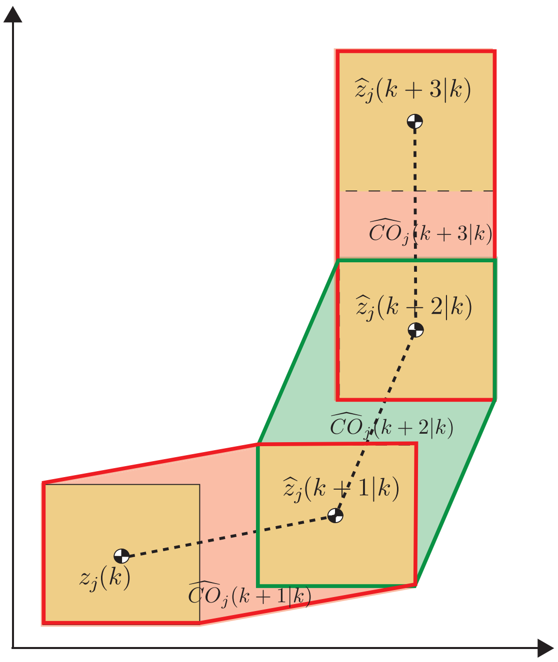



Fig. 4.5:Expanded collision avoidance region (calculated by using convex hull)

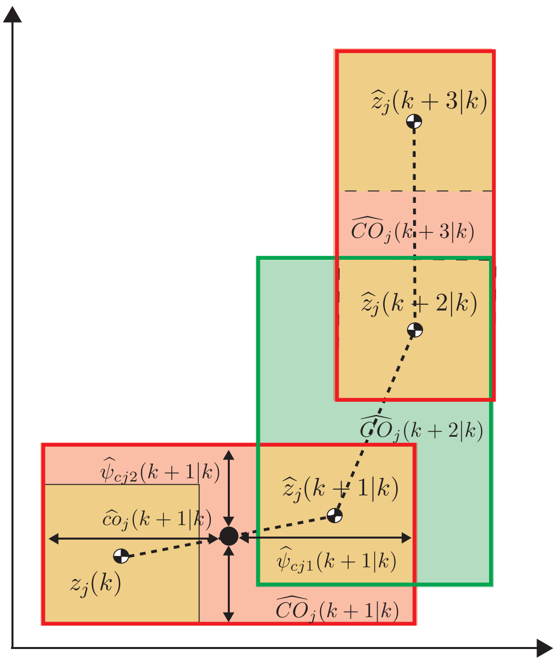



Fig. 4.6:Expanded collision avoidance region (eq. (4.29)-(4.31))

また，衝突回避アルゴリズムを以下のように修正し用いることとする．

**Step 0:** 初期時刻 $t=0$ において $k:=0$ および，

$$\widehat{\alpha}_i(\tau|0) := 0, \quad \tau = 0,\dots,N \tag{4.36}$$

$$\widehat{e}_j(\tau|0) := \tilde{e}_j(0), \quad \tau = 0,\dots,N, \quad j \neq i \tag{4.37}$$

とする．

**Step 1:** 時刻 $t = k\delta$ において，

- $k = i-1 \pmod n$ の場合，後述の最適制御問題を解き，

$$\widehat{\alpha}_i(\tau-1|k) = \alpha_i^*(\tau-1|k) \tag{4.38}$$

$$\widehat{e}_i(\tau|k) = e_i^*(\tau|k), \ \tau = k+1,\dots,k+N \tag{4.39}$$

  の更新を行い，$\widehat{e}_i(\tau|k)$ を他の Follower に送信する．ただし，$\alpha_i^*(\tau|k)$ と $e_i^*(\tau|k)$ は後述の最適制御問題により得られた最適軌道である．

- その他の場合，時刻 $p = k \pmod n$ における Follower $p$ の $\widehat{e}_p(\tau|k)$ を受信する．

**Step 2:** 時刻 $k\delta \le t < (k+1)\delta$ において

$$\tilde{\alpha}_i(t) = \widehat{\alpha}_i(k|k) \tag{4.40}$$

を用い，(4.35) 式の $u_i$ を適用する．

**Step 3:** $k = k+1$ および，

$$\widehat{\alpha}_i(\tau-1|k) = \widehat{\alpha}_i(\tau-1|k-1) \tag{4.41}$$

$$\widehat{e}_j(\tau|k) = \widehat{e}_j(\tau|k-1),\ \tau = k,\dots,k+N-1 \tag{4.42}$$

$$\begin{aligned}&\widehat{e}_j(k+N|k) = \\ &\quad A\widehat{e}_j(k+N-1|k-1) + B\widehat{\alpha}_j(k+N-1|k-1)\end{aligned} \tag{4.43}$$

と更新し，Step 1 に続く．

なお，$\widehat{e}_i(\tau|k)$ は時刻 $t = k\delta$ で予測した時刻 $\hat{t} = \tau\delta$ での $e_i$ の予測値を示すものとする．また，(4.43) 式では，Follower $i$ が持っていない $\tau = k+N-1$ における $\widehat{\alpha}_j(\tau|k-1)$ の情報を，

$$\widehat{\alpha}_j(\tau|k-1) = K_f\widehat{e}_j(\tau|k-1) \tag{4.44}$$

と仮定することで予測値を計算するものとする．ただし，$K_f = -(R + B^T P_f B)^{-1} B^T P_f A$ であり，$A$，$B$ は (4.49) 式で表わされる行列，$P_f$ は以下の代数リカッチ方程式の解である．

$$P_f = A^T P_f A - A^T P_f B(R + B^T P_f B)^{-1} B^T P_f A + Q \tag{4.45}$$

また，$Q$，$R$ は評価関数 (4.48) で用いられる重み行列である．

ここで，静止障害物回避の場合と同様に，衝突回避制約 (2.7) に含まれる 0, 1 変数 $\kappa_{ijp}$ に対する以下の条件を最適制御問題に課すことを考える．

$$\kappa_{ijp}(\tau-1|k) = \kappa_{ijp}(\tau|k) = 0,\ \exists p(=1,2,3,4) \tag{4.46}$$

また，この条件は 0, 1 変数を用いて以下のような線形の不等式制約として記述することができる．

$$\begin{aligned}
\kappa_{ij1}(\tau-1|k) + \kappa_{ij1}(\tau|k) &\le M\rho_{ij1} + M\rho_{ij2} \\
\kappa_{ij2}(\tau-1|k) + \kappa_{ij2}(\tau|k) &\le -M\rho_{ij1} + M\rho_{ij2} + M \\
\kappa_{ij3}(\tau-1|k) + \kappa_{ij3}(\tau|k) &\le M\rho_{ij1} - M\rho_{ij2} + M \\
\kappa_{ij4}(\tau-1|k) + \kappa_{ij4}(\tau|k) &\le -M\rho_{ij1} - M\rho_{ij2} + 2M
\end{aligned} \tag{4.47}$$

ただし，$\rho_{ij1}$，$\rho_{ij2}$ は新たに導入した0,1変数である．なお，遷移制約(4.47)を課した時の衝突回避領域のモード変化を示したのがFig. 4.7，Fig. 4.8である．Fig. 4.6と比較し，保守性が低減化できていることがわかる．

以上より，最適制御問題 $P_1$ は，以下の遷移制約法に基づく最適制御問題 $P_5$ に修正される．

**予測時刻間の衝突回避を考慮した最適制御問題 $P_5$**

$$\min_{\widehat{\alpha}_i}\Big\{\sum_{\tau=k+1}^{k+N}\Big(\widehat{e}_i(\tau-1|k)^T Q\widehat{e}_i(\tau-1|k)+\widehat{\alpha}(\tau-1|k)^T R\widehat{\alpha}(\tau-1|k)\Big) + \widehat{e}_i(k+N|k)^T P_f \widehat{e}_i(k+N|k)\Big\} \tag{4.48}$$

subject to

$$\widehat{e}_i(\tau|k) = A\widehat{e}_i(\tau-1|k) + B\widehat{\alpha}_i(\tau-1|k),\ \widehat{e}_i(k|k) = \tilde{e}_i(k\delta) \tag{4.49}$$

$$\|\widehat{\alpha}_i(\tau-1|k)\|_\infty \le \eta \tag{4.50}$$

$$\|\widehat{e}_i(k+N|k)\|_\infty \le \gamma_i \tag{4.51}$$

$$\begin{bmatrix} \widehat{e}_i(\tau|k)-\widehat{e}_j(\tau|k)+\zeta_i^d-\zeta_j^d \\ -\widehat{e}_i(\tau|k)+\widehat{e}_j(\tau|k)-\zeta_i^d+\zeta_j^d \end{bmatrix} \le M\begin{bmatrix} \kappa_{ij1}(\tau|k) \\ \kappa_{ij3}(\tau|k) \\ \kappa_{ij2}(\tau|k) \\ \kappa_{ij4}(\tau|k) \end{bmatrix} - \begin{bmatrix} \psi \\ \psi \\ \psi \\ \psi \end{bmatrix} \tag{4.52}$$

$$\sum_{p=1}^{4}\kappa_{ijp}(\tau|k) \le 3$$

$$\begin{bmatrix} \kappa_{ij1}(\tau-1|k)+\kappa_{ij1}(\tau|k) \\ \kappa_{ij2}(\tau-1|k)+\kappa_{ij2}(\tau|k) \\ \kappa_{ij3}(\tau-1|k)+\kappa_{ij3}(\tau|k) \\ \kappa_{ij4}(\tau-1|k)+\kappa_{ij4}(\tau|k) \end{bmatrix} \le \begin{bmatrix} M & M \\ -M & M \\ M & -M \\ -M & -M \end{bmatrix}\begin{bmatrix} \rho_{ij1}(\tau|k) \\ \rho_{ij2}(\tau|k) \end{bmatrix} + \begin{bmatrix} 0 \\ M \\ M \\ 2M \end{bmatrix} \tag{4.53}$$

$$\forall j \ne i,\ \tau = k+1,\ k+2,\ \ldots,\ k+N$$

ただし，等式制約(4.49)は予測モデル(4.34)を零次ホールドによりサンプリング周期 $\delta_m$ で離散化したものであり，$A = I_2$，$B = \delta_m I_2$ である．ここでは議論の簡単化のため $\delta = \delta_m$ と仮定する．また，$R > 0$，$Q \ge 0$，$\eta > 0$，$\gamma_i \ge 0$ は設計パラメータである．予測モデル(4.34)では遷移制約法を適用するため，フィードバック項 $-\lambda\tilde{e}_i$ を除いている．そのため，評価関数(4.48)には状態に関する項 $\widehat{e}_i(\tau|k)^T Q\widehat{e}_i(\tau|k)$ を追加している．また，不等式制約(4.50)は入力制約で，不等式制約(3.13)からフィードバック項 $-\lambda\tilde{e}$ を除いた形式に修正している．不等式制約(4.51)は終端制約である．不等式制約(4.52)は衝突回避制約(2.7)である．不等式制約(4.53)は(4.47)式の遷移条件に関する制約である．なお，$\kappa_{ijp}(k|k)$ は時

刻 $t = k\delta$ での移動体の位置から決定される値である

なお，Leader との衝突回避を考慮する場合，Leader は移動座標系では原点で停止しているため静止障害物として取り扱うことができる．したがって，定理4より遷移制約法を用いることで予測時刻間での Leader との衝突回避も保証は直ちに導かれる．

ここで，最適制御問題 $P_5$ の実行可能解の予測時刻間での衝突回避について次の補題が得られる．

**補題 3** 時刻 $t = k\delta$ において Follower $i$ が衝突回避制約 (4.52) を満足しているとする．このとき，時刻 $t = k\delta$ における Follower $i$ の最適制御問題の実行可能解は Follower $j(\forall j \neq i)$ と衝突しない軌道となる．

**証明：** 仮定より予測時刻 $\hat{t} = \tau\delta(\tau = k+1, \ldots, k+N)$ および時刻 $t = k\delta$ では衝突が起きない．したがって予測時刻間 $\hat{t} \in ((\tau-1)\delta, \tau\delta)$ で衝突が起きないことを示せばよい．

遷移制約 (4.53) より予測時刻 $\hat{t} = (\tau-1)\delta$，$\tau\delta$ では $\kappa_{ijp_j}(\tau-1|k) = \kappa_{ijp_j}(\tau|k) = 0$ となる $p_j \in \{1, 2, 3, 4\}$ が存在する．ここでは，$\kappa_{ij2}(\tau-1|k) = \kappa_{ij2}(\tau|k) = 0$ を仮定する．このとき，衝突回避制約 (4.52) から，

$$\widehat{r_{ij}}(\tau-1|k) := \widehat{r_i}(\tau-1|k) - \widehat{r_j}(\tau-1|k) \geq \psi \tag{4.54}$$

$$\widehat{r_{ij}}(\tau|k) := \widehat{r_i}(\tau|k) - \widehat{r_j}(\tau|k) \geq \psi \tag{4.55}$$

が成り立つことが分かる．ただし，$[\widehat{r_i}(\tau|k), \widehat{l_i}(\tau|k)]^T = \widehat{e_i}(\tau|k) + \zeta_i^d$ である．また，予測モデル (4.49) より，予測時刻間での $r$ 軸方向の入力は，それぞれ $\widehat{\alpha}_{ir}(\tau-1|k)$，$\widehat{\alpha}_{jr}(\tau-1|k)$ となる．ここで，

$$\Delta r_{ij} := \widehat{r_{ij}}(\tau|k) - \widehat{r_{ij}}(\tau-1|k) \tag{4.56}$$

とおくと，

$$\Delta r_{ij} = \delta(\widehat{\alpha}_i(\tau-1|k) - \widehat{\alpha}_j(\tau-1|k)) \tag{4.57}$$

となる．

まず，$\Delta r_{ij} \geq 0$ の場合，Follower $i$ と Follower $j$ は互いに $r$ 軸方向の相対距離を拡大するように移動する．このとき，(4.54) 式の両辺に $\frac{(\hat{t}-(\tau-1)\delta)\Delta r_{ij}}{\delta}(\geq 0)$ を加えると，

$$\widehat{r_{ij}}(\tau-1|k) + \frac{(\hat{t}-(\tau-1)\delta)\Delta r_{ij}}{\delta} \geq \psi + \frac{(\hat{t}-(\tau-1)\delta)\Delta r_{ij}}{\delta} \geq \psi \tag{4.58}$$

となる．また，予測時刻間での$\widehat{r}_{ij}$は(4.57)式より，

$$
\begin{aligned}
\widehat{r}_{ij}(\frac{\hat{t}}{\delta}|k) = \quad & \widehat{r}_i(\frac{\hat{t}}{\delta}|k) - \widehat{r}_j(\frac{\hat{t}}{\delta}|k) \\
= \quad & \widehat{r}_i(\tau-1|k) + (\hat{t}-(\tau-1)\delta)\widehat{\alpha}_i(\tau-1|k) \\
& -\widehat{r}_j(\tau-1|k) - (\hat{t}-(\tau-1)\delta)\widehat{\alpha}_j(\tau-1|k) \\
= \quad & \widehat{r}_{ij}(\tau-1|k) + \frac{(\hat{t}-(\tau-1)\delta)\Delta r_{ij}}{\delta}
\end{aligned}
\tag{4.59}
$$

であるので，(4.58)，(4.59)式より，

$$
\widehat{r}_{ij}(\frac{\hat{t}}{\delta}|k) \geq \quad \psi \tag{4.60}
$$

が得られる．一方，$\Delta r_{ij} < 0$の場合，Follower $i$とFollower $j$は$r$方向の相対距離を縮める方向に移動する．そのため，(4.54)式の下界を(4.57)式から，より正確に求めると，

$$
\widehat{r}_{ij}(\tau-1|k) \geq \psi - \Delta r_{ij} \tag{4.61}
$$

となる．(4.61)式の両辺に$\frac{(\hat{t}-(\tau-1)\delta)\Delta r_{ij}}{\delta}$を加えると，

$$
\widehat{r}_{ij}(\tau-1|k) + \frac{(\hat{t}-(\tau-1)\delta)\Delta r_{ij}}{\delta} \geq \psi - \frac{(\delta-\hat{t}+(\tau-1)\delta)\Delta r_{ij}}{\delta} \tag{4.62}
$$

となる．ここで$\Delta r_{ij} < 0$であるため，

$$
\frac{(\delta-\hat{t}+(\tau-1)\delta)\Delta r_{ij}}{\delta} < 0 \tag{4.63}
$$

となる．(4.62)，(4.63)式より

$$
\widehat{r}_{ij}(\tau-1|k) + \frac{(\hat{t}-(\tau-1)\delta)\Delta r_{ij}}{\delta} > \psi \tag{4.64}
$$

となる．(4.59)，(4.64)式より，

$$
\widehat{r}_{ij}(\frac{\hat{t}}{\delta}|k) > \psi \tag{4.65}
$$

が得られる．したがって，(4.60)，(4.65)式より予測時刻間でも$\kappa_{ij2}$に対応する衝突回避制約が満たされる．同様に，$\kappa_{ijp_j}(\tau-1|k) = \kappa_{ijp_j}(\tau|k) = 0,\ p_j \in \{1,3,4\}$の場合についても成り立つことが示される．したがって，$\hat{t} = [(\tau-1)\delta, \tau\delta]$において衝突回避が保証される．

以上より，$\hat{t} = [k\delta, (k+N)\delta]$においてFollower $i$とFollower $j(\forall j \neq i)$が衝突しないことが示された． □

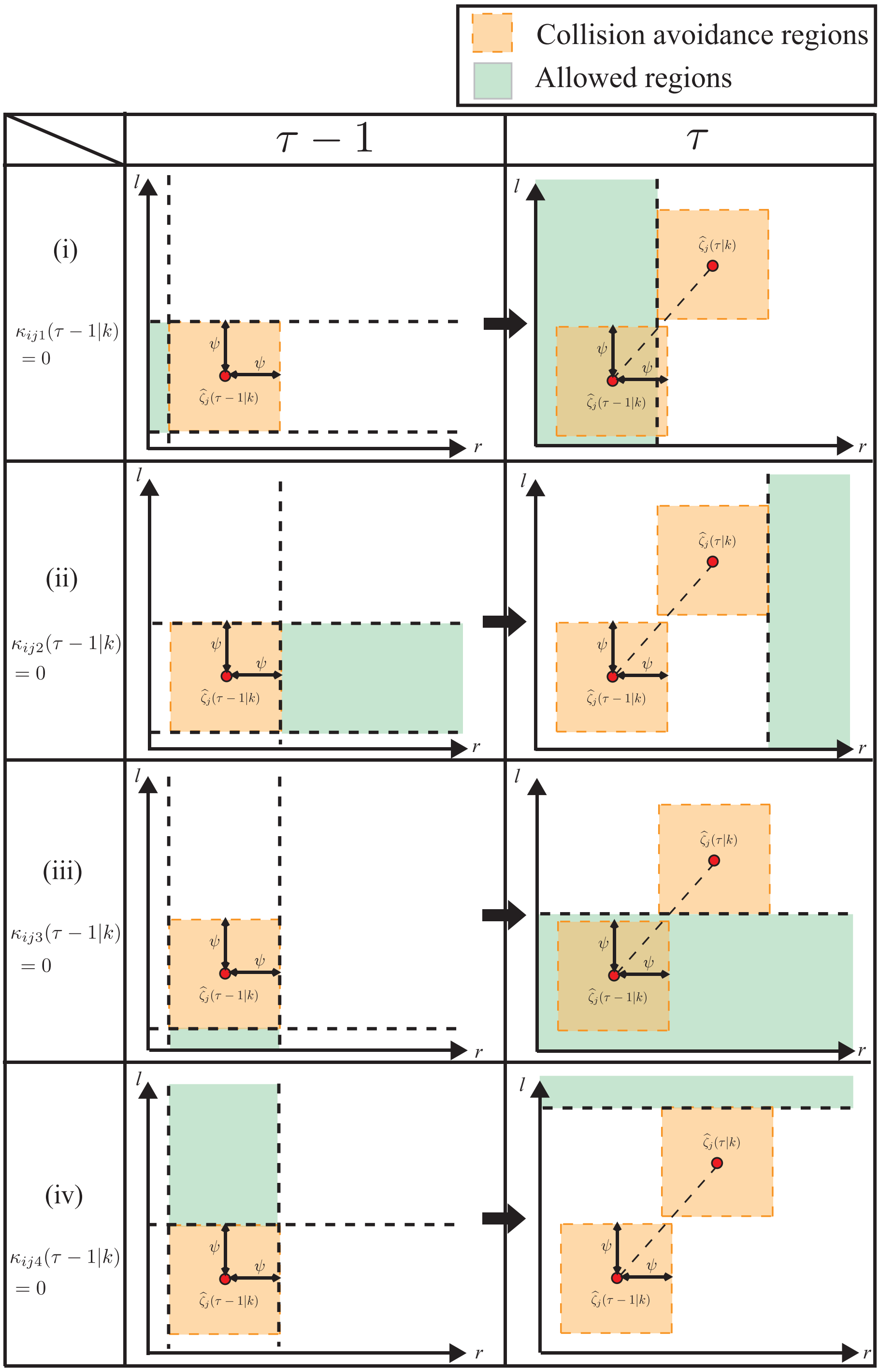



Fig. 4.7:Allowed region by constraint (4.47). Case of only one $\kappa_{ijp}$ becomes 0.

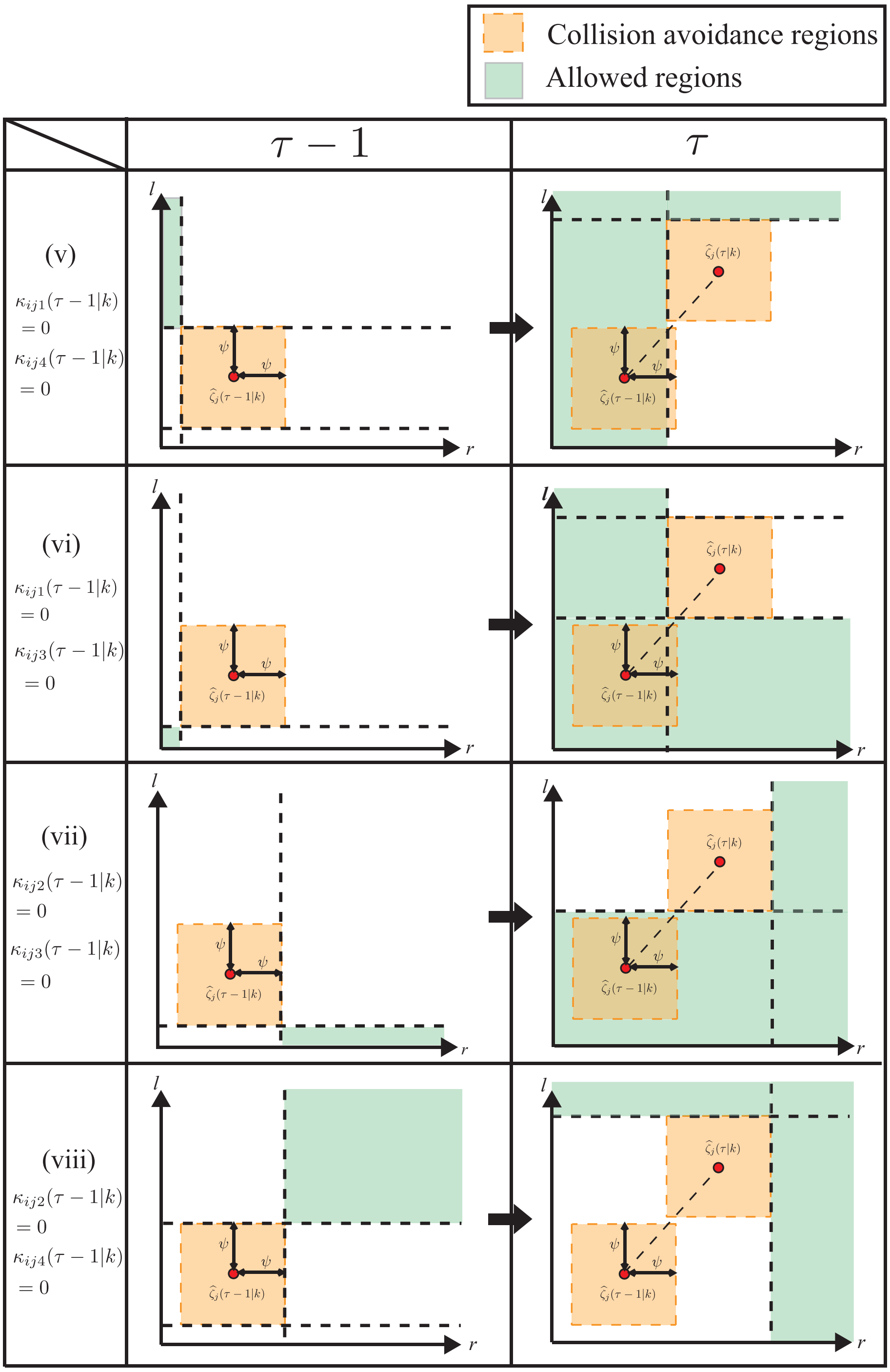



Fig. 4.8:Allowed region by constraint (4.47). Case of two $\kappa_{ijp}s$ become 0 at the same time.

### 4.2.3　可解性と収束性

ここでは，最適制御問題$P_5$の可解性と閉ループ系の安定性について述べる．まず，可解性と安定性を示すため，最適制御問題$P_5$が次の仮定を満たすものとする．

**仮定2** 終端集合(4.51)の大きさ$\gamma_i$ $(i = 1, \ldots, n)$および予測ステップ数$N$が次の条件を満たすものとする．また，重み行列$Q$，$R$は対角行列であるとする．

$$\|\zeta_i^d - \zeta_j^d\|_\infty \geq \psi + \gamma_i + \gamma_j, \quad \forall j \neq i \tag{4.66}$$

$$\|K_f \gamma_i\|_\infty \leq \eta, \quad N \geq n \tag{4.67}$$

次に以下の補題を導入する．

**補題4** $\|\tilde{e}_i\|_\infty \leq \gamma_i$および$\|\tilde{e}_j\|_\infty \leq \gamma_j$ $(i \neq j)$が仮定2を満たすものとする．このとき，終端時刻以降において次式が満たされる．

$$\|\tilde{e}_i - \tilde{e}_j + \zeta_i^d - \zeta_j^d\|_\infty \geq \psi \tag{4.68}$$

$$\|\tilde{\alpha}_i\|_\infty \leq \eta \tag{4.69}$$

**証明：** まず，(4.66)式より，

$$\|\zeta_i^d - \zeta_j^d\|_\infty - \gamma_i - \gamma_j \geq \psi \tag{4.70}$$

である．ここで，$\|\tilde{e}_i\|_\infty \leq \gamma_i$，$\|\tilde{e}_j\|_\infty \leq \gamma_j$を用いると，

$$\begin{aligned} \|\zeta_i^d - \zeta_j^d\|_\infty - \|\tilde{e}_i\|_\infty - \|\tilde{e}_j\|_\infty &\geq \psi \\ \|\tilde{e}_i - \tilde{e}_j + \zeta_i^d - \zeta_j^d\|_\infty &\geq \psi \end{aligned} \tag{4.71}$$

となり，(4.68)式が示された．また，(4.44)式より終端時刻以降では制御則(4.44)が用いられるため，

$$\|\tilde{\alpha}_i\|_\infty \leq \|K_f \gamma_i\|_\infty \leq \eta \tag{4.72}$$

となり，(4.69)式が示された． □

また，遷移制約(4.53)に関する以下の補題5，6を導入する．

**補題5** 時刻$t = k\delta$においてFollower $i$が最適制御問題を解いて予測軌道$E_i := [\widehat{e_i}(k+1|k), \ldots, \widehat{e_i}(k+N|k)]$を得たとする．また，時刻$t = (k+1)\delta, \ldots, (k+n-1)\delta$においてFollower $j (\forall j \neq i)$が最適制御問題を解いて予測軌道$E_j$を得たとする．このとき，$E_i$は$E_j$に対しても遷移制約(4.53)を満たす．

**証明:** 予測軌道$E_j$は$E_i$と衝突しないように計画したものであるため，$E_j$は$E_i$に対して遷移制約(4.53)を満足する．したがって，次式を満たす$p_j(\tau|\hat{k}) \in \{1,2,3,4\}$が存在する．

$$\begin{aligned} &\kappa_{jip_j(\tau|\hat{k})}(\tau-1|\hat{k}) = \kappa_{jip_j(\tau|\hat{k})}(\tau|\hat{k}) = 0 \\ &\quad \tau = \hat{k}+1,\ldots,\hat{k}+N \\ &\quad \hat{k} = \begin{cases} k+j-i & \text{if } j \in \{i+1,\ldots,n\} \\ k+j+(n-i) & \text{if } j \in \{1,\ldots,i-1\} \end{cases} \end{aligned} \tag{4.73}$$

ここで，衝突回避領域は(4.52)式より$r, l$軸に平行な正方領域である．言い換えると，各衝突回避領域は互いに平行な正方領域であるため，Follower $i$から見ると$p_j(\tau|\hat{k})$に対応する辺に平行なもう1辺に対応する0,1変数が0となる．すなわち，

$$\begin{aligned} &\kappa_{ijq_j(\tau|k)}(\tau-1|\hat{k}) = \kappa_{ijq_j(\tau|k)}(\tau|\hat{k}) = 0 \\ &q_j(\tau|\hat{k}) = \begin{cases} \{1,2\}\backslash p_j(\tau|\hat{k}) & \text{if } p_j(\tau|\hat{k}) \in \{1,2\} \\ \{3,4\}\backslash p_j(\tau|\hat{k}) & \text{if } p_j(\tau|\hat{k}) \in \{3,4\} \end{cases} \end{aligned} \tag{4.74}$$

が成り立つ．したがって，時刻$t = k\delta$で得られたFollower $i$の予測軌道$E_i$は時刻$t = (k+1)\delta,\ldots,(k+n-1)\delta$で得られたFollower $j$の予測軌道$E_j$に対しても遷移制約(4.53)を満たすことが示された． □

**補題6** 仮定2が満たされ，$\|\tilde{e}_i\|_\infty \le \gamma_i$および$\|\tilde{e}_j\|_\infty \le \gamma_j (\forall j \neq i)$であるとする．このとき，Follower $i$とFollower $j$は互いに遷移制約(4.53)を満たす.

**証明：** (4.66)式より,

$$r_i^d - r_j^d \ \ge\ \psi + \gamma_i + \gamma_j \tag{4.75}$$

$$-r_i^d + r_j^d \ \ge\ \psi + \gamma_i + \gamma_j \tag{4.76}$$

$$l_i^d - l_j^d \ \ge\ \psi + \gamma_i + \gamma_j \tag{4.77}$$

$$-l_i^d + l_j^d \ \ge\ \psi + \gamma_i + \gamma_j \tag{4.78}$$

の4つの不等式のうち少なくとも1つの不等式が満たされる．(4.75)式が満たされる場合を考えると，$|r_i - r_i^d| \le \gamma_i$および$|r_j - r_j^d| \le \gamma_j$より

$$r_i - r_j \ge (r_i^d - \gamma_i) - (r_j^d + \gamma_j) \ge \psi \tag{4.79}$$

が成り立ち，衝突回避制約(2.7)の$\kappa_{ij2}$が常に0になりうる．同様に，(4.76)〜(4.78)式が満たされる場合にはそれぞれ$\kappa_{ij1}$，$\kappa_{ij4}$，$\kappa_{ij3}$が常に0になりうる．す

なわち，$\|\tilde{e}_i\|_\infty \le \gamma_i$ および $\|\tilde{e}_j\|_\infty \le \gamma_j (\forall j \neq i)$ である場合，$\tilde{e}_i$，$\tilde{e}_j$ によらず，常に $\kappa_{ijp} = 0$ となる $p \in \{1,2,3,4\}$ が存在する．

以上より，終端集合内に存在するFollower同士は遷移制約(4.53)を満たすことが示された．□

最適制御問題の可解性について以下の定理が得られる．

**定理5** 最適制御問題の終端集合の大きさ $\gamma_i$ $(i = 1,\ldots,n)$ および予測ステップ $N$ が仮定2を満たすものとする．また，各Follower $i$ $(i = 1,\ldots,n)$ の初期の更新時刻 $t = k_i\delta$ $(k_i = sn + i - 1, s = 0)$ での最適制御問題が可解であるとする．このとき $t = k_i\delta$ $(k_i = sn + i - 1,\ s \in \mathbb{N}^+)$ におけるすべての最適制御問題は可解となり，Follower同士の衝突が起きない．

**証明：** 数学的帰納法を用いて証明を行なう．まず，Follower $i$ $(= 1,\ldots,n)$ は時刻 $t = k_i\delta$ $(k_i = sn + i - 1,\ s \in \mathbb{N}^0)$ において最適解を持つと仮定する．

4.2.2節の衝突回避アルゴリズムよりFollower $i(= 2,\ldots,n)$ は $k_i\delta \le \hat{t} \le (k_i + N)\delta$ においてFollower 1が

$$\widehat{\alpha}_1(\frac{\hat{t}}{\delta}|k_i) = \begin{cases} \alpha_1^*(\tau|k_1),\ \tau\delta \le \hat{t} < (\tau+1)\delta, \tau \in \{k_i,\ldots,k_1+N-1\} \\ K_f\widehat{e}_1(\tau|k), \\ \quad \tau\delta \le \hat{t} < (\tau+1)\delta, \tau \in \{k_1+N-1,\ldots,k_i+N-1\} \end{cases} \tag{4.80}$$

を適用するものとし，入力 $\alpha_i^*(\hat{t}|k_i)$ を決定する．したがって，補題3よりFollower $i(= 2,\ldots,n)$ はFollower 1と衝突することなく，$t = (k_i + N)\delta$ で終端集合

$$\Omega_i := \{\zeta_i :\ \|\zeta_i - \zeta_i^d\|_\infty \le \gamma_i\} \tag{4.81}$$

に入る．また，$t = (k_1 + n)\delta$ においてFollower 1が

$$\begin{aligned} &\widehat{\alpha}_1(\frac{\hat{t}}{\delta}|k_1+n) \\ &= \begin{cases} \alpha_1^*(\tau|k_1),\ \tau\delta \le \hat{t} < (\tau+1)\delta, \tau \in \{k_1+n,\ldots,k_1+N-1\} \\ K_f\widehat{e}_1(\tau|k), \\ \quad \tau\delta \le \hat{t} < (\tau+1)\delta, \tau \in \{k_1+N-1,\ldots,k_1+n+N-1\} \end{cases} \end{aligned} \tag{4.82}$$

を適用すれば，補題3，5より $(k_1 + n)\delta \le \hat{t} \le (k_i + N)\delta$ においてFollower 1は他のFollowerと衝突することはない．また，補題4より，衝突回避制約(4.52)は，$(k_i + N)\delta \le \hat{t} \le (k_1 + n + N)\delta$ においてもFollower 1と $i$ は各々の終端集合内にいるため，満たされる．このときの(4.82)式の入力 $\widehat{\alpha}_1$ は，補題4より，入力制約(4.50)

を満たすことが示される．また，補題6より $(k_1+n)\delta \le \hat{t} \le (k_i+N)\delta$ でも遷移制約(4.53)も満たすことが示される．したがって，Follower 1は $t=(k_1+n)\delta$ において実行可能解を少なくとも1つ持つことが示される．

同様に，$t=(k_i+n)\delta$ においてFollower $i(=2,\ldots,n)$ が

$$\begin{aligned}&\widehat{\alpha}_i(\frac{\hat{t}}{\delta}|k_i+n)\\&=\begin{cases}\alpha_i^*(\tau|k_i),\ \tau\delta\le\hat{t}<(\tau+1)\delta, \tau\in\{k_i+n,\ldots,k_i+N-1\}\\ K_f\widehat{e}_i(\tau|k),\\ \quad\tau\delta\le\hat{t}<(\tau+1)\delta,\tau\in\{k_i+N-1,\ldots,k_i+n+N-1\}\end{cases}\end{aligned} \tag{4.83}$$

を適用すれば，補題3，5より $(k_i+n)\delta\le\hat{t}\le(k_j+N)\delta$ においてFollower $j(=i+1,\ldots,n)$ と衝突することはない．また，Follower $i$，$j$ は $(k_j+N)\delta\le\hat{t}\le(k_i+n+N)\delta$ において各々の終端集合内にいるため，Follower $i$，$j$ は衝突することはない．

したがって，すべてのFollower $i\ (=1,\ldots,n)$ が $k_i=sn+i-1$ において最適解を持つならば，$k_i=(s+1)n+i-1$ において，すべてのFollowerは実行可能解を持つ．以上より，$s=0$ での可解性が示されれば，数学的帰納法により証明される． □

また，閉ループ系の安定性について以下の定理が得られる．

**定理6** 定理5の条件が満たされるとき，全てのFollowerは衝突せずに目標位置に収束する．すなわち

$$\lim_{t\to\infty}\tilde{e}_i(t)=0,\ i=1,\ldots,n \tag{4.84}$$

**証明：** 漸近安定性を示すためには以下の(i)〜(iv)を示せばよい[55].

1. 終端集合では状態制約が満たされる．
2. 終端集合では終端制御器(4.44)が入力制約を満足する．
3. 終端制御器(4.44)のもとで終端集合が正の不変集合となる．
4. 終端コスト $F_i(k_i):=\|\widehat{e}_i(k_i+N|k_i)\|_{P_f}$ に関するステップ変化，

$$\widehat{F}_i := F_i(k_i+n)-F_i(k_i)+\sum_{\tau=k_i+N}^{k_i+n+N-1}(\|\widehat{e}_i(\tau|k_i)\|_Q+\|K_f\widehat{e}_i(\tau|k_i)\|_R) \tag{4.85}$$

が $\widehat{F}_i\le 0$ となる．

まず，補題4および補題6より終端集合では衝突回避制約(4.52)，遷移制約(4.53)が満たされるため(i)が示される．同様に補題4より(ii)が示される．また，仮定2より$Q$，$R$，$A$，$B$が全て対角行列であるので$K_f$，$P_f$も対角行列となる．すなわち，終端時刻以降で用いられる制御則(4.44)は$r, l$軸で独立な設計となる．制御則(4.44)は各軸で漸近安定であるので，終端集合は正の不変集合となり(iii)が示される．(iv)は$P_f$が(4.45)式の解であることから明らかである．

以上より，漸近安定性が示される．　□

定理5，6は定理1，2と比較し，離散化による影響を陽に考慮した結果となっている．

### 4.2.4　考察

前節では，更新周期$\delta$と予測周期$\delta_m$が一致している場合について，予測時刻での衝突回避や実行可能性，閉ループ系の安定性について議論を行った．ここでは，更新周期$\delta$と予測周期$\delta_m$に異なる値を設定した場合について考察を行う．また，0,1変数$\kappa_{ijp}(k|k)$の決定方法について議論を行う．

#### 更新周期と予測周期が異なる場合($\delta \neq \delta_m$)

予測モデル(4.49)は更新周期$\delta$で(2.27)式の離散化を行ったものである．予測モデル(4.49)を用いた場合，予測ホライズンを長くするには$\delta$を長く設定する必要があり，問題によっては最適制御問題を解く周期$n\delta$が過度に大きくなってしまうことが考えられる．このような場合，予測モデル(2.27)の離散化を更新周期$\delta$とは異なる予測周期$\delta_m(> \delta)$で行うことが望ましいと考えられる．予測周期$\delta_m$で離散化した場合，予測モデル(4.49)は以下のように修正される．

$$\widehat{e_i}(\tau|k) = A\widehat{e_i}(\tau-1|k) + B_1\widehat{\alpha_i}(\tau-1|k) \qquad (4.86)$$
$$\tau = k+1, \ldots, k+N$$

ただし，$B_1 = \delta_m I_2$である．予測モデル(4.86)を用いた場合の例をFig. 4.9に示す．Fig. 4.9の矢印が予測ホライズンを示したものである．また，●印が予測を行う時刻であり，入力$\widehat{\alpha_i}$が更新される時刻を示したものである．このように予測モデル(4.86)を用いることで，予測ホライズン$N\delta_m$を長く確保しながら，最適制御問題を解く周期$n\delta$を短くすることが可能となる．しかしながら，Fig. 3.2から，各Followerが入力$\widehat{\alpha_i}$を更新する予測時刻が異なるため(4.57)

式が成り立たないことが分かる．したがって，単に予測周期と更新周期に異なる値を設定した場合には補題3が成り立たないことになる．

そこで，本論文では予測周期$\delta_m(>\delta)$で離散化を行う場合，予測ステップの1ステップ目を可変にすることを考える．具体的には，予測の1ステップ目のみ以下の$\tilde{\delta}_m$で離散化を用い，以降は予測周期$\delta_m$で離散化を行うものとする．

$$\tilde{\delta}_m := \delta_m - \mathrm{mod}(k\delta, \delta_m) \tag{4.87}$$

ただし，(4.87)式は時刻$t = k\delta$における最適制御問題での1ステップ目の予測周期を示したものである．このとき，予測モデル(4.86)は以下のように修正される．

$$\begin{aligned} \widehat{e}_i(k+1|k) &= A\widehat{e}_i(k|k) + B_2\widehat{\alpha}_i(k|k) \\ \widehat{e}_i(\tau+1|k) &= A\widehat{e}_i(\tau|k) + B_1\widehat{\alpha}_i(\tau|k) \\ \tau &= k+1, k+2, \ldots, k+N-1 \end{aligned} \tag{4.88}$$

ただし，$B_2 = \tilde{\delta}_m I_2$である．また，更新周期$\delta$と予測周期$\delta_m$が異なる値であるため,（4.88）式の$\widehat{e}_i(\tau|k)$の表記は時刻$t = k\delta$における$\tilde{e}_i(k\delta + (\tau - k)\delta_m)$の予測値を示すものとする．予測モデル(4.88)を用いた場合の例をFig. 4.10に示す．Fig. 4.10より，各Followerが入力$\widehat{\alpha}_i$を更新する予測時刻が一致するため(4.57)式が成り立つことが分かる．したがって，補題3が成り立ち，予測時刻間での衝突回避が保証される．なお，予測モデル(4.88)を用いる場合には最適制御問題$P'_5$を解くこととなる．

**予測時刻間の衝突回避を考慮した最適制御問題$P'_5$**

$$\begin{aligned} \min_{\widehat{\alpha}_i} \Bigg\{ \sum_{\tau=k+1}^{k+N-1} \Big( \|\widehat{e}_i(\tau|k)\|_Q + \|\widehat{\alpha}_i(\tau|k)\|_R \Big) + \|\widehat{e}_i(k|k)\|_Q \\ + \|\widehat{\alpha}_i(k|k)\|_{\tilde{R}} + \|\widehat{e}_i(k+N|k)\|_{P_f} \Bigg\} \end{aligned} \tag{4.89}$$

subject to (4.50), (4.51), (4.52), (4.53), (4.88)

ただし，$\tilde{R} := \frac{\tilde{\delta}_m}{\delta_m}R$, $\|\widehat{e}_i(\tau|k)\|_Q := \widehat{e}_i^T(\tau|k)Q\widehat{e}_i(\tau|k)$である．なお，ここでは更新周期$\delta$と予測周期$\delta_m$が異なる値であるため,（4.89）式の$\widehat{e}_i(\tau|k)$の表記は時刻$t = k\delta$における$\tilde{e}_i(k\delta + (\tau - k)\delta_m)$の予測値を示すものとする．

### 0,1変数$\kappa_{ijp}(k|k)$の決定方法

遷移制約(4.53)で用いられる0,1変数$\kappa_{ijp}(k|k)(\forall j \neq i)$は初期状態から決定される値である．具体的には，時刻$t = k\delta$における状態から衝突回避制約(4.52)

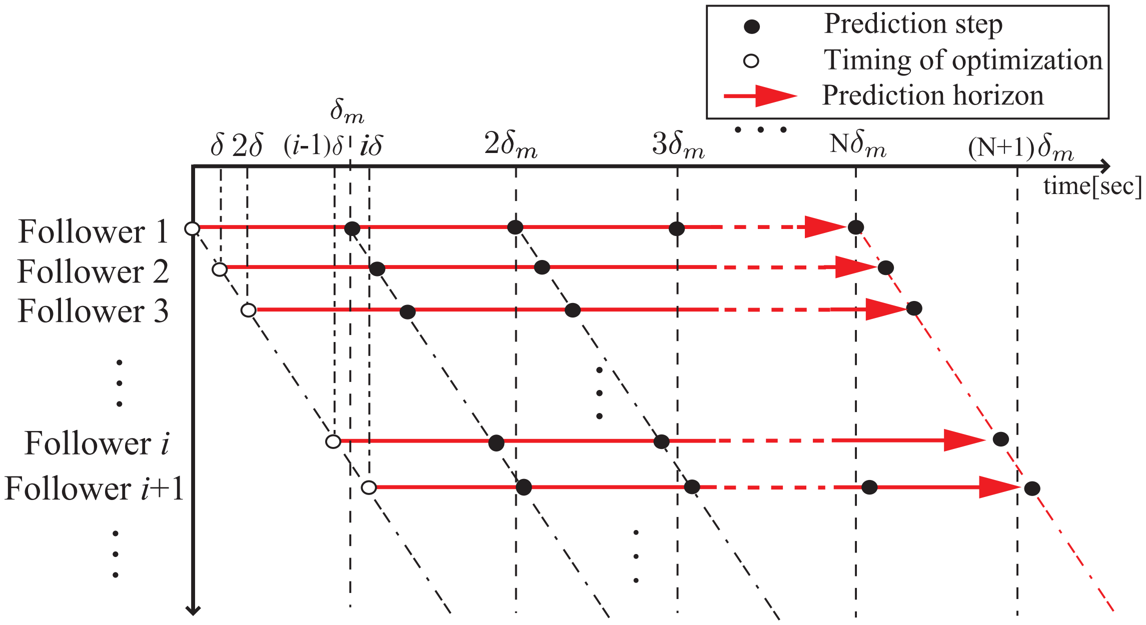



Fig. 4.9:Discretized prediction horizon with $\delta_m$

を満たすか否か判定し，満たす制約に対応する$\kappa_{ijp}(k|k)$を0とし，満たさない制約に対応する$\kappa_{ijp}(k|k)$を1と設定する．しかしながら，モデル化誤差や計算遅れ等の影響で初期時刻においてFollower $j$との衝突回避制約(4.52)を満たさないことが考えられる．この場合，$\kappa_{ijp}(k|k) = 1(\forall p \in \{1, 2, 3, 4\})$となるため，遷移制約(4.53)が満たされなくなり実行不可能となる．このように，初期時刻で衝突回避制約を侵害した場合には，$\kappa_{ijp}(k|k)$の決定条件を緩和する必要がある．

ここで，衝突回避制約の侵害量$\lambda_{ijp}$を以下のように定義する．衝突回避制約が満たされない場合には$\lambda_{ijp} > 0$となる．

$$\lambda_{ij1}(k|k) \ := \ r_i(k|k) - r_j(k|k) + \psi \tag{4.90}$$

$$\lambda_{ij2}(k|k) \ := \ -r_i(k|k) + r_j(k|k) + \psi \tag{4.91}$$

$$\lambda_{ij3}(k|k) \ := \ l_i(k|k) - l_j(k|k) + \psi \tag{4.92}$$

$$\lambda_{ij4}(k|k) \ := \ -l_i(k|k) + l_j(k|k) + \psi \tag{4.93}$$

本論文では侵害量$\lambda_{ijp}$が最も小さい制約に対応する$\kappa_{ijp}$を0に設定することを考える．すなわち，

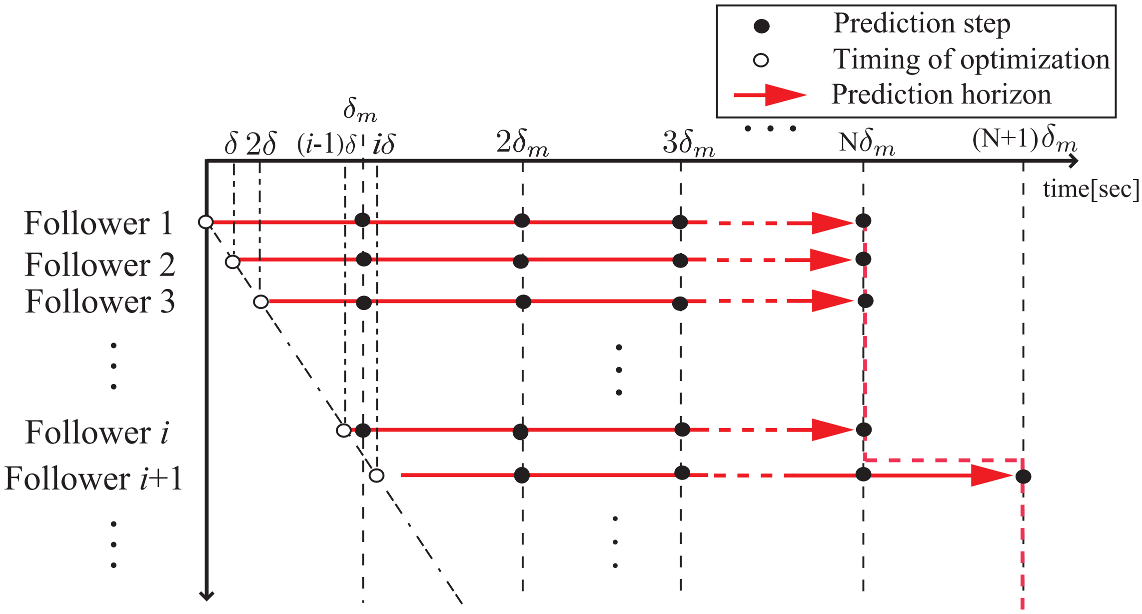



Fig. 4.10:Discretized prediction horizon with $\tilde{\delta}_m$ and $\delta_m$

$$\kappa_{ijp}(k|k) = \begin{cases} 0, \text{ if } \ p = \underset{q\in\{1,2,3,4\}}{\arg\min}\,\lambda_{ijq}(k|k) \\ 1, \text{ otherwise} \end{cases} \tag{4.94}$$

とすることを考える．このとき，衝突回避制約と遷移制約により $\lambda_{ijp}(k+1|k) \leq 0 (p = \underset{q\in\{1,2,3,4\}}{\arg\min}\,\lambda_{ijq}(k|k))$ となる．したがって，定理3と同様の議論により

$$\lambda_{ijp}(k|k) > \lambda_{ijp}(\frac{\hat{t}}{\delta}|k) > \lambda_{ijp}(k+1|k), \ k\delta < \hat{t} < (k+1)\delta \tag{4.95}$$

となり，初期の侵害量以上に制約を侵害することはない．以上より，初期時刻での侵害量が一定以下であれば，衝突回避制約を侵害した場合でも上記の方法で安全な軌道が計画できることとなる．

## 4.3　数値例

提案した制御手法の有効性をシミュレーションにより検証する．ここでは，4.1節で提案した静止障害物のための制御手法と4.2節で提案した複数移動体同士の衝突回避のための制御手法のそれぞれについて行ったシミュレーション結果を示す．なお，シミュレーションにはMATLABを用い，最適化問題を解くためのソルバーにはCPLEXを用いた[82].

### 4.3.1 単体の移動体の障害物回避

ここでは，4.1.2節で提案した可変最大速度法と4.1.3節で提案した遷移制約法を適用した結果を示す．また，比較のために，予測時刻間の障害物回避を考慮しないで行った結果も示す．なお，計算に用いたPCはCPU: Intel Core2 Duo 2.0GHz，RAM: 2GBである

シミュレーションに用いたパラメータは以下の通りである．移動体の初期状態を絶対座標系で，$(-6.0, -6.0, 0.0)$とし，目標位置を絶対座標系で，$(0.0, 0.0)$とした．その他のパラメータは，$d = 0.2$，$\eta = 0.41$，$\beta = 0.205$，$Q = I$，$R = I$，$S = I$，$M = 30$，$N = 20$，$\delta = 1.0$，$\gamma = 0.5$である．障害物はFig. 4.11のように5個設置した．Fig. 4.11の障害物周囲の点線は可変最大速度法で最大速度が制限される領域である．

Fig. 4.11中の実線は予測時刻のみ障害物回避を考慮した場合の移動体の軌跡を示している．Fig. 4.11より予測時刻では障害物回避制約を満たしているが，予測時刻間では衝突が発生していることが確認できる．一方，Fig. 4.11中の破線，一点鎖線はそれぞれ可変最大速度法，遷移制約法を適用した場合の移動体の軌跡を示している．これら提案手法では予測時刻だけでなく予測時刻間でも衝突が起きていないことが確認できる．

なお，Fig. 4.12はこのときの移動体への制御入力$u$を示したものである．この図より可変最大速度法では障害物4と5の間を通過した15[sec]付近において他手法と比較し速度が制限されていることが確認できる．そのため他手法と比較し，収束までに時間を要していることも確認できる．

また，可変最大速度法について$S = 100I$に変更して行ったシミュレーション結果をFig. 4.13に示す．$S = I$の場合には障害物4と5の間を通り抜ける軌道が選択されていた．一方，重み$S$を大きくした場合，障害物5の下側を通過し障害物から離れた，より安全性の高い軌道が選択されていることがFig. 4.13より確認できる．この結果より評価関数の重みにより，軌道の安全性を調整できることが確認できる．ただし，障害物の“角”ではかなり障害物と接近していることも確認できる．これは，障害物の拡大量$\beta$として条件(4.8)で与えられる下限値を用いたためである．より安全な軌道を計画したい場合には$\beta$をより大きくすることで，障害物との距離の下限を大きくすることが可能である．なお，Fig. 4.14はこのときの移動体への制御入力$u$を示したものである．Fig. 4.12，Fig. 4.14より重みが$S = I$の場合と比較し，実際に大きな速度が出力されていることが確認できる．

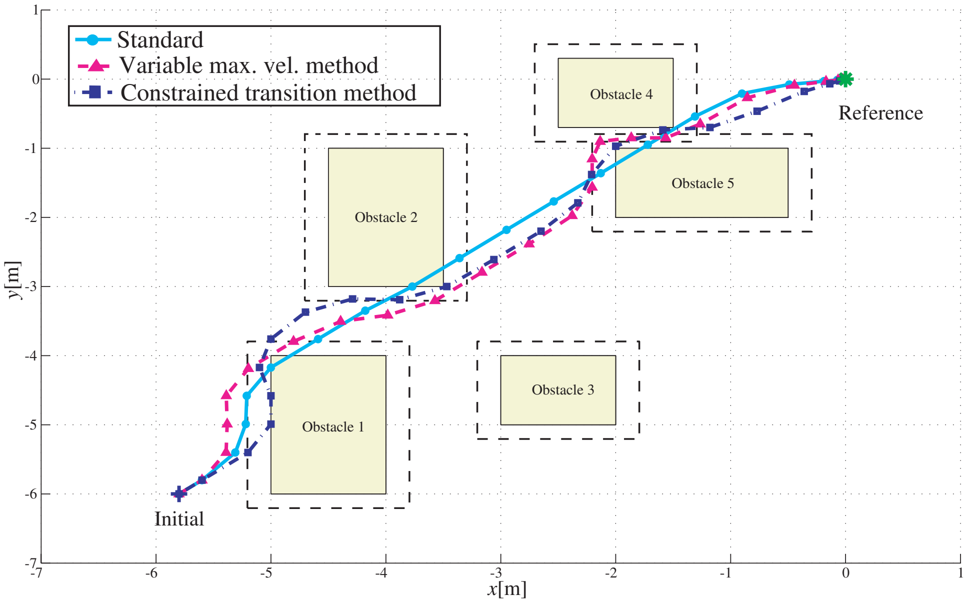



Fig. 4.11:Path of the vehicle in simulation

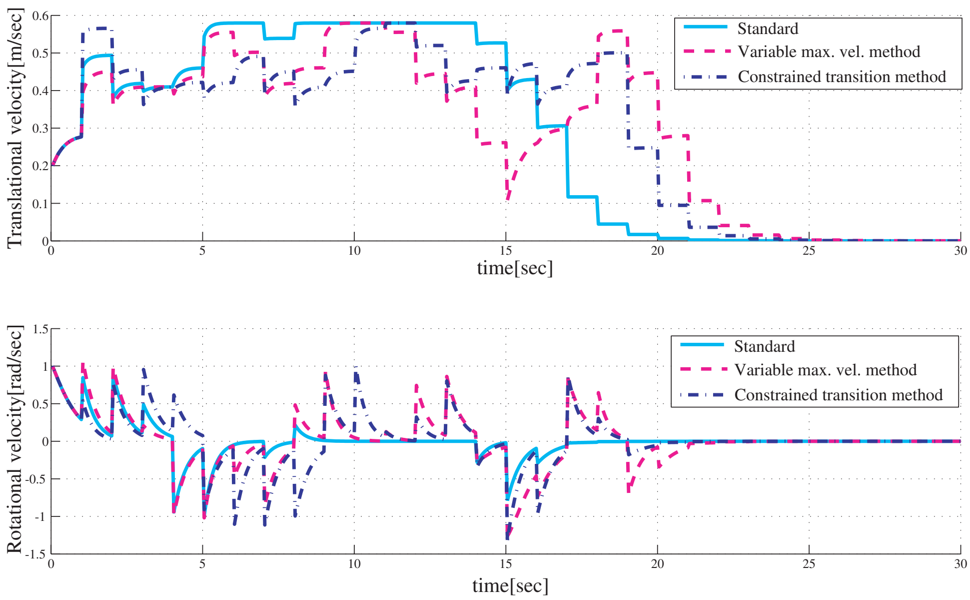



Fig. 4.12:Velocity input of the vechile in simulation

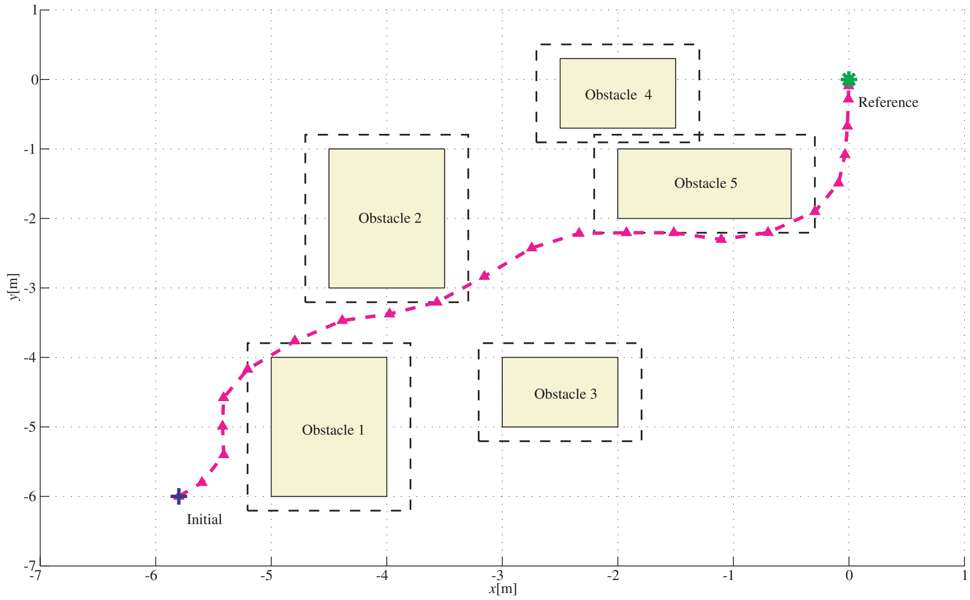



Fig. 4.13:Another simulation result of the proposed method in section. 4.1.2

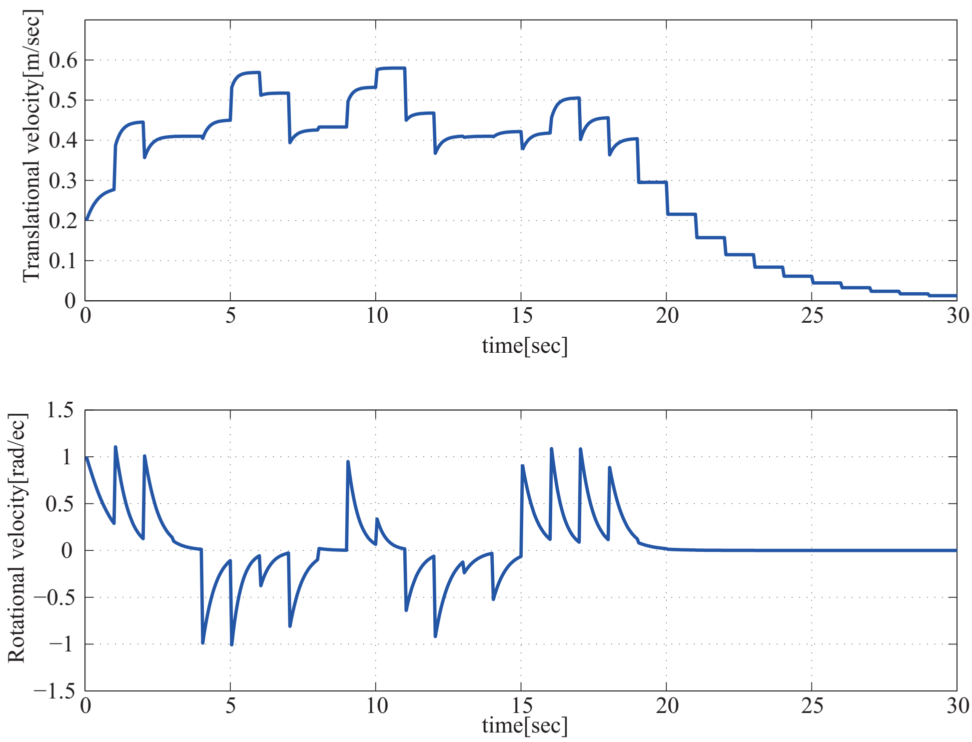



Fig. 4.14:Velocity input of the vechile in simulation

### 4.3.2 複数移動体同士の衝突回避

次に4.2.2節で提案した遷移制約法に基づく衝突回避手法を適用した結果を示す．ここでは，Followerが3台の場合についてシミュレーションを行った．ここでは比較のため，最適制御問題$P_5$で遷移制約(4.53)を課さなかった場合(Simulation 1)と最適制御問題$P_5'$を解いた場合の結果(Simulation 2, 3)を示す．なお，計算に用いたPCはCPU:Intel Core2 Duo 3.16GHz，RAM:4GBである．

**Simulation 1**

まず，遷移制約(4.53)を課さなかった場合の結果を示す．シミュレーションに用いたパラメータは以下のとおりである．Leaderを絶対座標系において$(0.0, 0.0, 0.0)$に配置し，入力$u_r = (0.2, 0.1)^T$を印加するものとする．また，3台のFollowerの初期状態を絶対座標系でそれぞれ，$(-3.0, -1.0, 0.0)$，$(-3.0, 0.0, 0.0)$，$(-3.0, 1.0, 0.0)$とし，目標位置を移動座標系において$(-1.0, -0.6)$，$(0.0, -0.6)$，$(1.0, -0.6)$とした．また，更新周期$\delta = 0.1$[sec]，予測ステップ数$N = 6$，予測周期$\delta_m = 1.2$[sec]とした．そのほかのパラメータは$d = 0.2$，$\psi = 0.5$，$M = 30$，$\gamma_i = 0.05$，$\eta = 0.6$，$R = 10I$，$Q = I$，$K_f = -0.262I$，$P_f = 3.18I$である．

シミュレーション結果をFig. 4.15，Fig. 4.16に示す．Fig. 4.15，Fig. 4.16はそれぞれ，絶対座標系でのFollowerの制御点$z_i$の軌跡，その際のFollower間の最小距離$\min_{ij} \|\zeta_i - \zeta_j\|_\infty$を示したものである．Fig. 4.16より，遷移制約(4.53)を課さなかった場合にはFollower同士が衝突していることが分かる．

**Simulation 2**

次に，Simulation 1と同様のパラメータで最適制御問題$P_5'$を解いた場合のシミュレーション結果をFig. 4.17，Fig. 4.18に示す．Fig. 4.17，Fig. 4.18はそれぞれ，絶対座標系でのLeaderの位置$(x_r, y_r)$とFollowerの制御点$z_i$の軌跡を示したもの，その際のFollower間の最小距離$\min_{ij} \|\zeta_i - \zeta_j\|_\infty$を示したものである．Fig. 4.17，Fig. 4.18より衝突を回避しながら目標の形状を達成できていることが確認できる．また，Fig. 4.18よりSimulation 1とは異なり，予測時刻間でも衝突回避制約が満たされていることも確認できる．

**Simulation 3**

また，異なるパラメータについて最適制御問題$P_5'$を解いた場合の結果を示す．シミュレーションに用いたパラメータは以下のとおりである．Leaderを絶対座標系において$(0.0, 0.0, 0.0)$に配置し，入力$u_r = (0.2, 0.1)^T$が印加されて

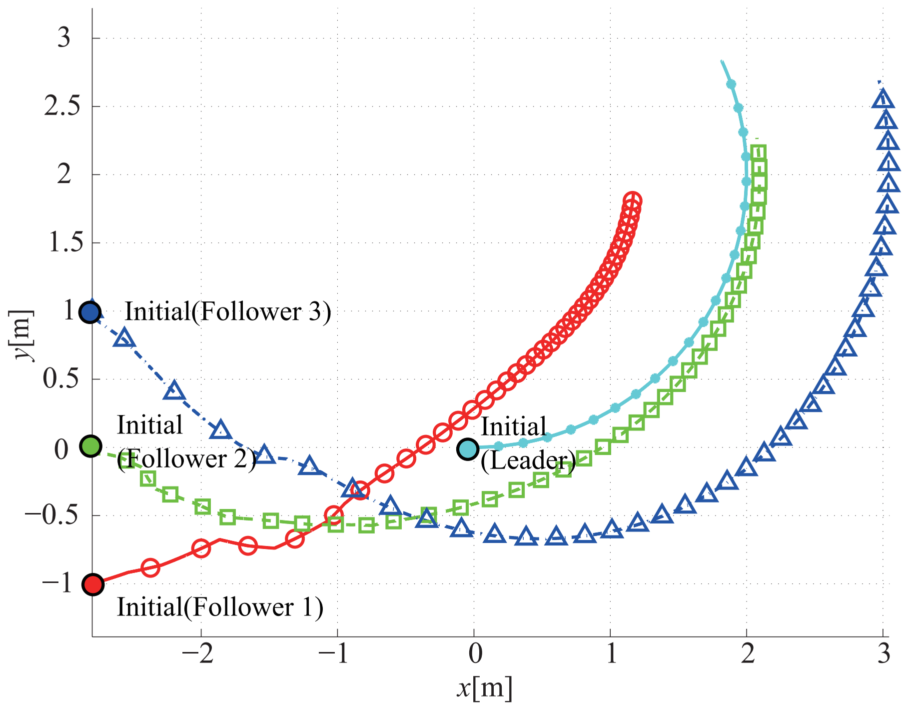



Fig. 4.15:Paths of the vehicles at the global frame (Simulation 1)

いるものとする．また，3台のFollowerの初期状態を絶対座標系でそれぞれ，(−1.0, −1.0, 0.0)，(−2.0, 0.0, 0.0)，(−1.0, 1.0, 0.0)とし，目標位置を移動座標系において(0.0, −1.2)，(0.0, −0.6)，(0.0, −1.8)とした．その他のパラメータは$\gamma_i = 0.05$とした以外はSimulation 1と同じである．

シミュレーション結果をFig. 4.19〜Fig. 4.21に示す．Fig. 4.19〜Fig. 4.21はそれぞれ，絶対座標系でのLeaderの位置$(x_r, y_r)$とFollowerの制御点$z_i$の軌跡，Leaderに固定した移動座標系でのFollowerの制御点$\zeta_i$の軌跡，その際のFollower間の最小距離$\min_{ij} \|\zeta_i - \zeta_j\|_\infty$を示したものである．Fig. 4.19，Fig. 4.20よりLeaderが移動した場合でも衝突を回避しながら目標の形状を達成できていることが確認できる．また，Fig. 4.21より予測時刻間でも衝突回避制約が満たされていることが確認できる．

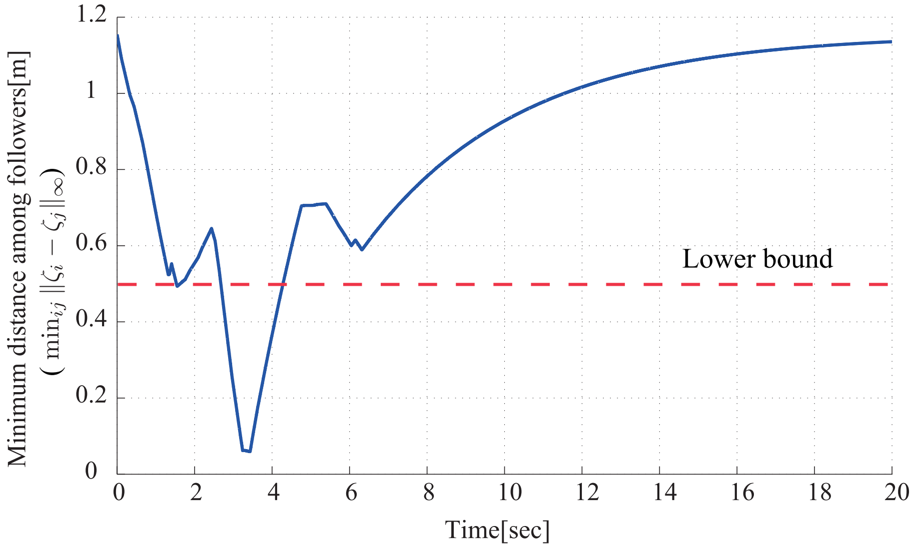



Fig. 4.16:Minimum distance among followers (Simulation 1)

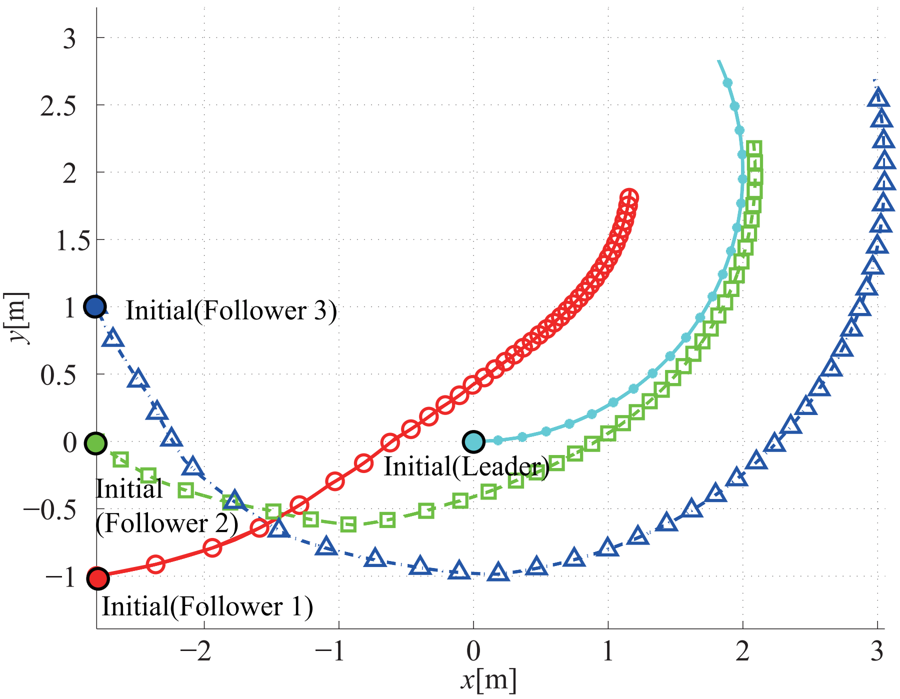



Fig. 4.17:Paths of the vehicles at the global frame (Simulation 2)

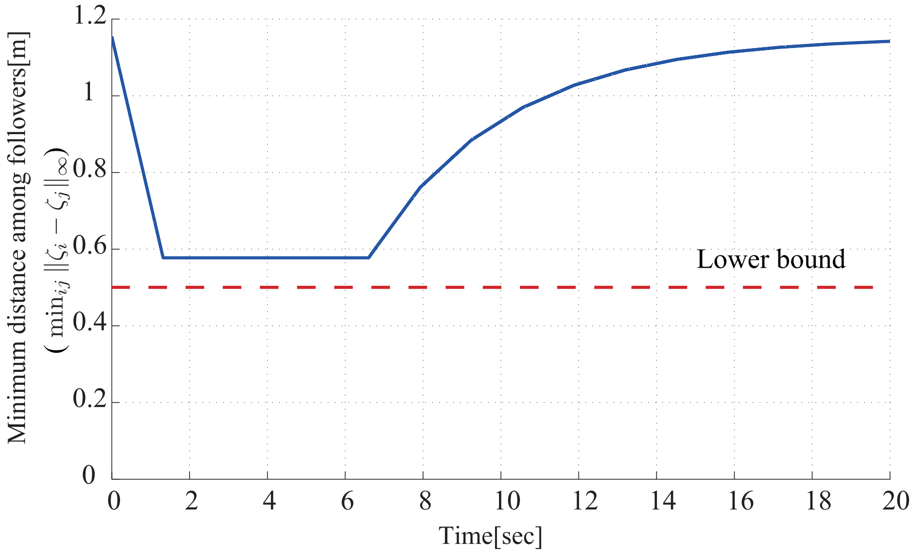



Fig. 4.18:Minimum distance among followers (Simulation 2)

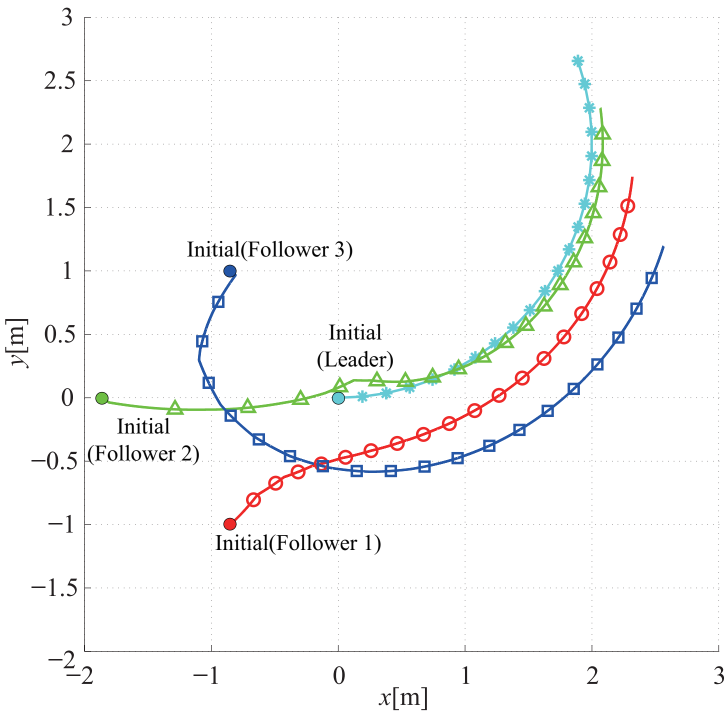



Fig. 4.19:Paths of the vehicles at the global frame (simulation 3)

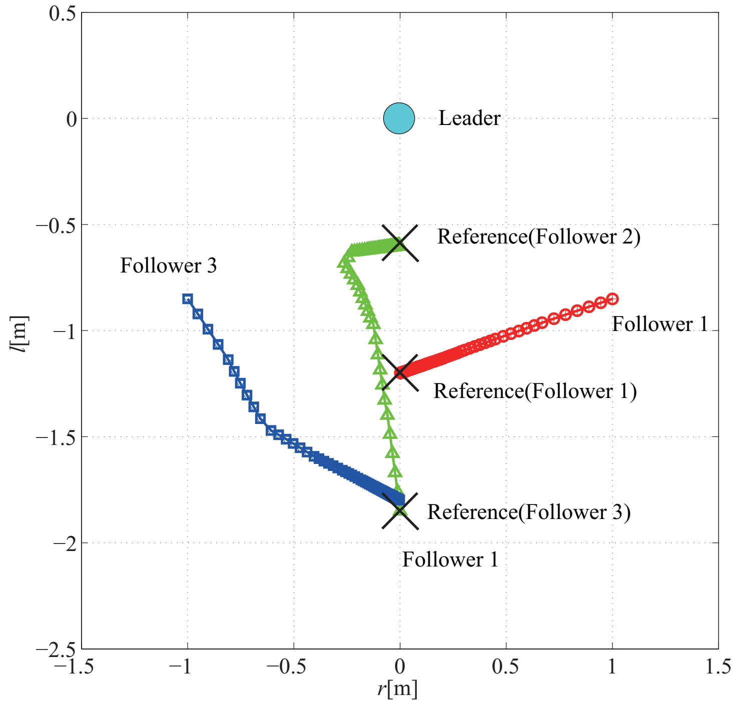



Fig. 4.20:Paths of the vehicles at the local frame (simulation 3)

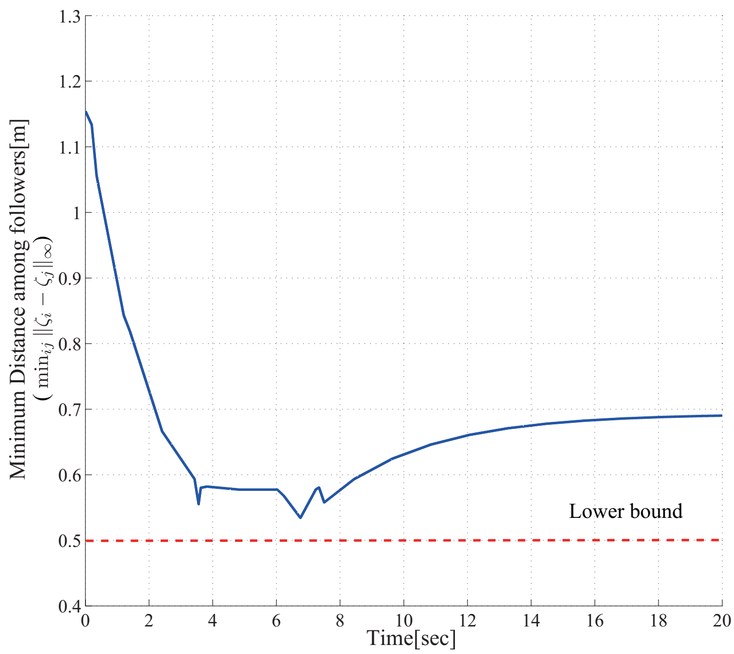



Fig. 4.21:Minimum distance among followers (simulation 3)

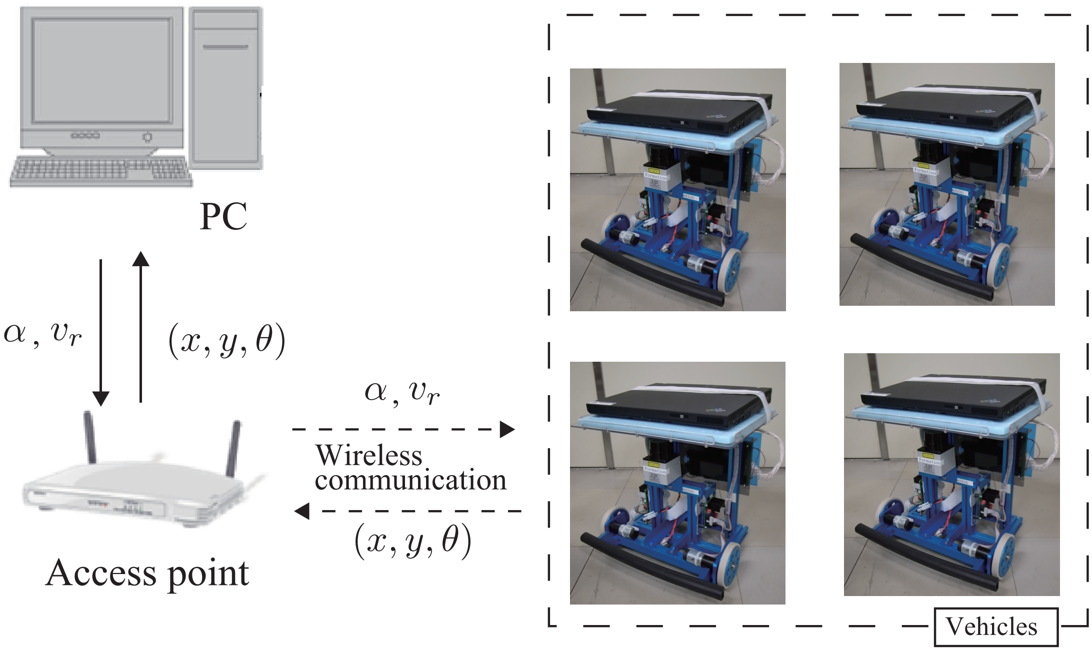



Fig. 4.22:Experimental system

## 4.4　実験

本節では実験により4.1節で提案した静止障害物のための制御手法と4.2節で提案した複数移動体同士の衝突回避のための制御手法の有効性を検証した．

4.4.1節では実験に用いた実験システムの概要を説明する．また，4.4.2節では，静止障害物のための制御手法について行った4.4.3節では複数移動体同士の衝突回避のための制御手法について行った実験結果を示す.

### 4.4.1　実験システムの概要

実験システムの概要をFig. 4.22に示す. 実験には左右独立駆動型2輪車両“beego(TechnoCraft社製)”を使用した．ここでは，システムの簡単化のため最適制御問題は外部に設置した1台の計算用PC(CPU:Intel Core2 Duo 3.16GHz, RAM:4GB)で解き，無線LAN(IEEE 802.11a)により各移動体にその最適解を送信するものとした．また，各移動体の自己位置情報はデッドレコニングによる推定値を用い，計算用PCに送信されるものとする．最適制御問題を解くためのサブルーチンにはMatlab Engine[84]とCPLEXを用いた．これらはOpenRTM-aist[83]を用い，以下に示すようなモジュールとして実装を行った．

### ソフトウェアモジュールの構成

実験で用いたOpenRTMに基づくソフトウェア構成を説明する．OpenRTMとは、ロボットを構成する要素（アクチュエータやセンサなど）やロボットを制御するソフトウェアをコンポーネントとして部品化するためのミドルウェアである．これを用いることで，部品化されたソフトウェアコンポーネントを組み合わせ、多様な機能を持つロボットシステムを容易に構築することが可能となる[83]．また，コンポーネント間の通信はミドルウェア側が実現するため，移動体間の通信が必要な衝突回避問題などでも容易に開発が可能となる．

実験に用いたRTコンポーネントをFig. 4.23に示す．各ブロックが“コンポーネント”を表わし，コンポーネント間をつなぐ実線が“通信のライン”を表わす．以下に開発した各コンポーネントの概要を示す．

- **MIQP :** 最適制御問題を解くためのコンポーネントで，内部でMATLAB engine[84]を介し，MATLAB上でCMEX化したCPLEXにより最適制御問題を解き，その最適解を出力．これにより，最適制御問題を解くコードはシミュレーションと同じものが使用可能．

- **LeaderController :** Leaderに対する速度指令値を出力．ここではあらかじめ指定した値を出力．

- **FollowerController :** “MIQP”により出力された最適制御問題の解に基づきFollowerに対する制御入力を計算・出力．

- **beego :** 受信した速度指令値をもとにbeegoを制御し，デッドレコニングによる推定自己位置を出力．

- **MultiRobotMux :** 受信した複数台の推定自己位置情報をまとめて出力．

Fig. 4.23の点線で囲まれたコンポーネントは各移動体に搭載したPC上で実行され，“MIQP”と“MultiRobotMux”は外部に設置した計算用PC上で実行される．単体の移動体の軌道計画の場合には1台分のコンポーネントのみ実行することで対応が可能である．なお，beegoモジュールの代わりに独自に開発したシミュレータモジュール“MultiSimulator”を接続することも可能となっている(Fig. 4.24参照)．このシミュレータを活用することで容易にテストが可能となっている(Fig. 4.25参照)．

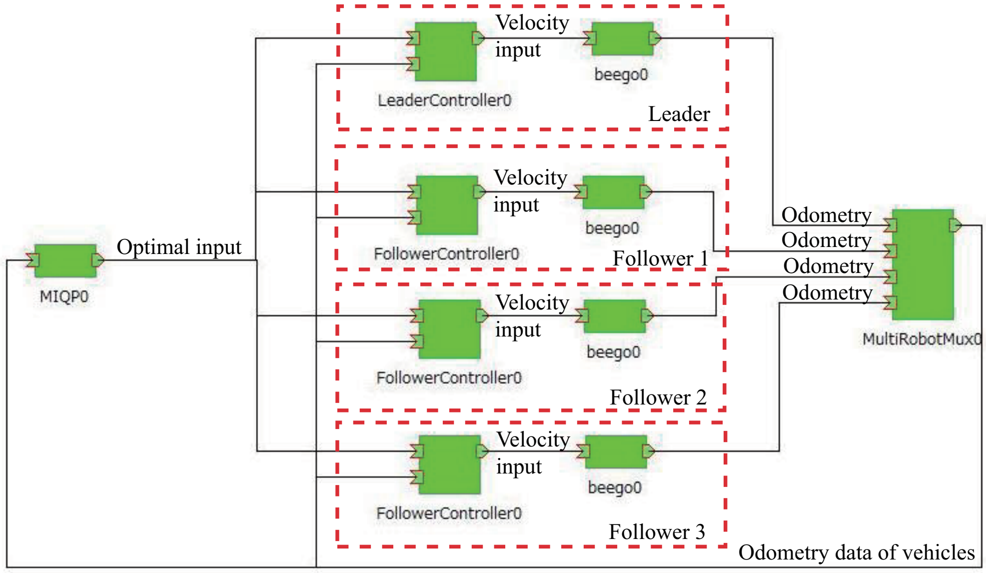



Fig. 4.23:Software components for experiment

### 4.4.2　単体の移動体の障害物回避

まず，単体の移動体が障害物回避を行う実験結果を示す．実験に用いたパラメータは以下の通りである．移動体の初期状態を絶対座標系で，$(-3.95, 0, 0)$ とし，目標位置を $(0, 0)$ とした．その他のパラメータは，$d = 0.2$，$\eta = 0.41$，$\beta = 0.205$，$Q = I$，$R = I$，$S = I$，$M = 30$，$N = 12$，$\delta = 1.0$，$\gamma = 0.5$ である．また，障害物は Fig. 4.26 に示すように仮想的に2個設置した．Fig. 4.26 の障害物周囲の点線は可変最大速度法で最大速度が制限される領域である．

Fig. 4.26 の実線は予測時刻での障害物回避のみ考慮した場合の移動体の軌跡を示しており，障害物との衝突が起きてしまっていることが確認できる．ここでは障害物は仮想的に設置していたため衝突後も移動を継続しているが，実際には衝突が起きた時点で移動の継続が不可能となる．

一方，破線と一点鎖線はそれぞれ可変最大速度法と遷移制約法を適用した場合の結果を示しており，これら2つの提案手法では障害物との衝突が起きていないことが確認できる．特に，可変最大速度法では障害物近傍において最大速度が制限されるため，障害物と距離を置いた軌道となっていることが確認できる．実際，移動体の入力を示した Fig. 4.27 より，可変最大速度法では障害物近傍を走行した3〜8[sec] の間，速度が低く制限されていることが確認できる．このように，可変最大速度法では障害物近傍で速度を落として走行するため遷移制約法と比較し，収束までに要する時間が大きく

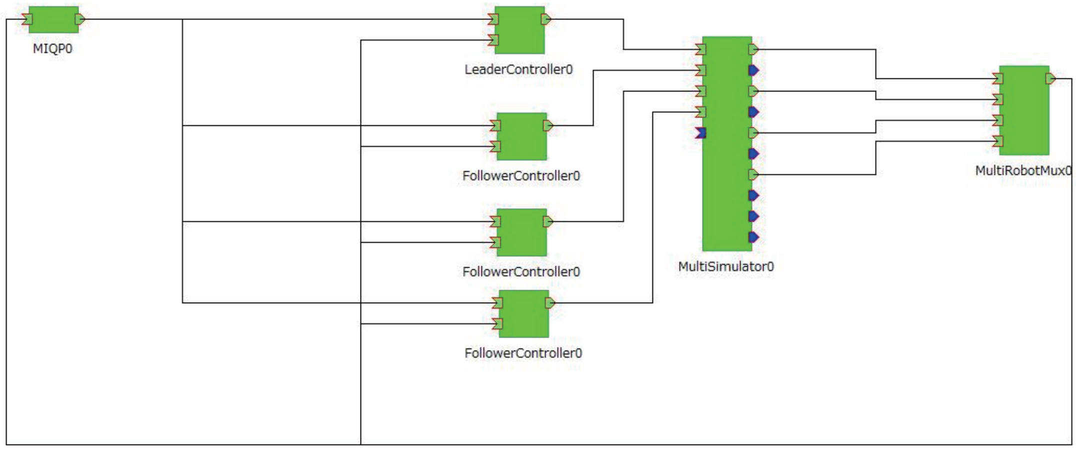



Fig. 4.24:Software components for experiment(with simulator)

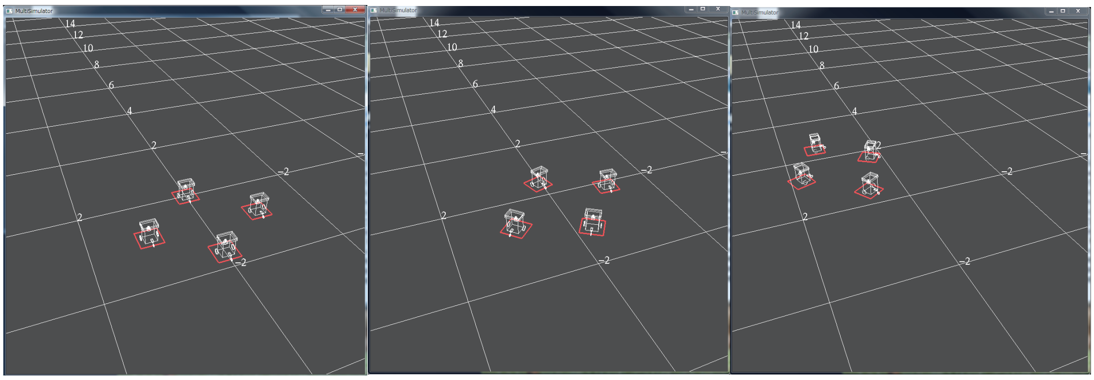


Fig. 4.25:Snapshots of simulator

なることも確認できる．なお，最適化問題を解くの要した最大の計算時間はそれぞれ，従来手法：0.151[sec]，可変最大速度法：0.170[sec]，遷移制約法：0.199[sec]であった．

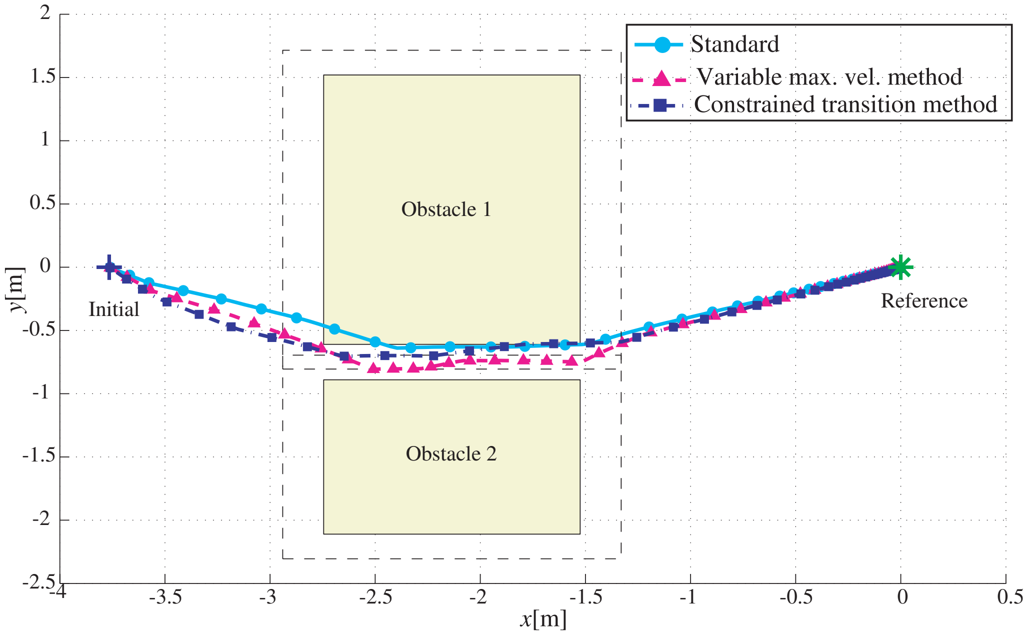



Fig. 4.26:Path of the vehicle in experiment

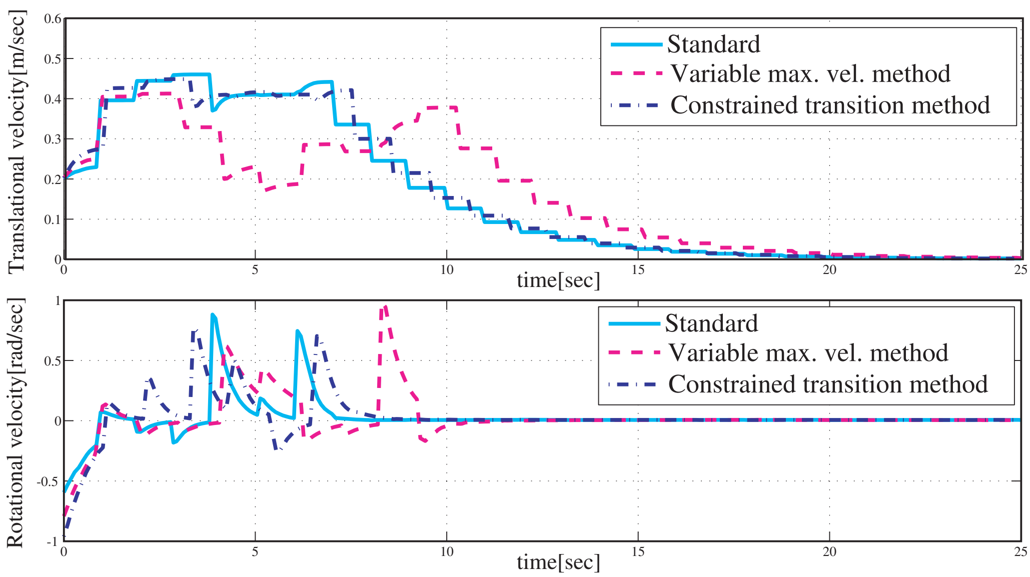



Fig. 4.27:Velocity input of the vehicle in experiment

### 4.4.3　複数移動体の衝突回避

次に“beego”を4台使用し4.2.2節の手法を実験により検証した．4台のうち，1台をLeader，残りの3台をFollowerとあらかじめ設定した．以下では，Leaderの挙動に影響を受けず，連続時間上で衝突回避がなされることを示すため，Leaderが固定の場合(Experiment 1)とLeaderが移動する場合(Experiment 2)の2通りの結果を示す.

#### Leaderが固定の場合

Leaderを絶対座標系において(0.0, 0.0, 0.0)に配置し，移動しないものとする．また，3台のFollowerの初期状態を絶対座標系でそれぞれ，(−2.4, −1.2, 0.0)，(−2.4, 0.0, 0.0)，(−2.4, 1.2, 0.0)とし，目標位置を移動座標系において(−1.2, −0.6)，(0.0, −0.6)，(1.2, −0.6)とした．更新周期$\delta = 0.2$[sec]，予測ステップ$N = 6$，予測周期$\delta_m = 3.0$[sec]とした．そのほかのパラメータは$d = 0.25$，$\psi = 0.5$，$M = 30$，$\gamma_i = 0.25$，$\eta = 0.3$，$R = 10I$，$Q = I$，$K_f = 0$，$P_f = I$である．

実験結果をFig. 4.28〜Fig. 4.31に示す．Fig. 4.28は絶対座標系でのLeaderの位置$(x_r, y_r)$とFollowerの制御点$z_i$の軌跡を示したものである．Fig. 4.29はFollower間の最小距離$(\min_{ij}\|\zeta_i - \zeta_j\|_\infty)$を示したものである．Fig. 4.30はFollowerの制御入力$u_i$を示したものである．Fig. 4.31は実験の様子を示したものである．

Fig. 4.28より目標の編隊形状を達成できていることが確認できる．また，Fig. 4.29より実際にFollower同士の衝突が生じていないことが確認できる．ただし，わずかながら衝突回避制約を侵害してしまっていることも確認できる．衝突回避制約を侵害した原因としては計算による遅れ，モデル化誤差などが考えられる．しかしながら最大でも約0.04[m]程度と微小であるため十分許容範囲内であると考えられる．また，4.2.2節で述べたように衝突回避制約を侵害した場合でも，遷移制約の特性から他移動体から離れる方向に軌道を計画するので，大きな問題は生じないと考えられる．

#### Leaderが移動する場合

次にLeaderが移動する場合について実験を行った．ここでは実験スペースの関係上，Leaderは$x$軸の正方向に0.1[m/sec]で等速直線運動するものとする．そのほかのパラメータはExperiment 1と同じである．

実験結果をFig. 4.32〜Fig. 4.36に示す．Fig. 4.32は絶対座標系でのLeaderの位置$(x_r, y_r)$とFollowerの制御点$z_i$の軌跡を示したものである．Fig. 4.33はLeaderに固定した移動座標系での各Followerの制御点$\zeta_i$の軌跡を示したものである．Fig. 4.34はFollower間の最小距離$(\min_{ij}\|\zeta_i - \zeta_j\|_\infty)$を示したものである．Fig. 4.35

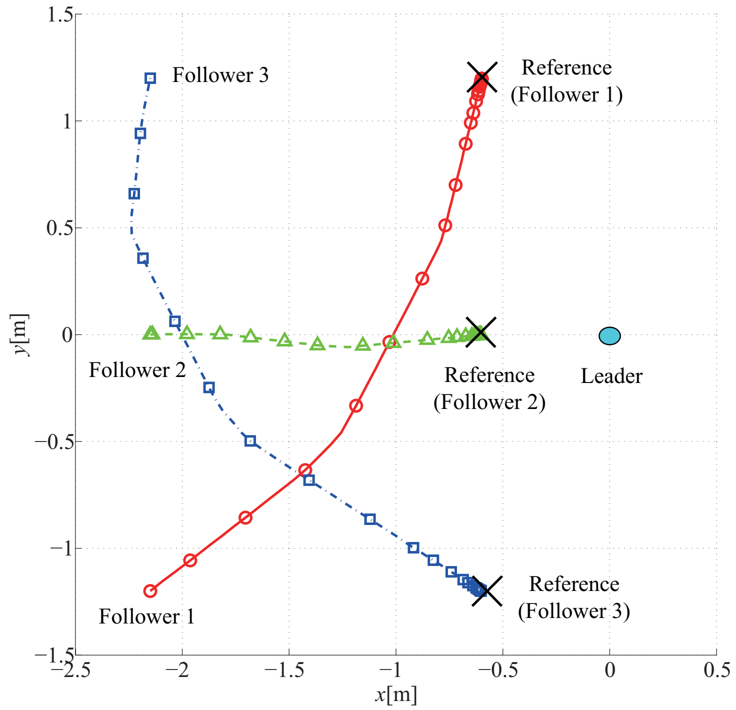



Fig. 4.28:Paths of the vehicles at the global frame (experiment 1)

はFollowerの制御入力$u_i$を示したものである．Fig. 4.36は実験の様子を示したものである．

Fig. 4.32，Fig. 4.33より目標の編隊形状を達成できていることが確認できる．また，Fig. 4.34より実際に移動体間の衝突が生じていないことが確認できる．Leaderが固定の場合と同様にLeaderが移動する場合でもわずかながら衝突回避制約を侵害してしまっているが，最大でも約0.04[m]程度であり十分許容範囲内であると考えられる．また，Fig. 4.30とFig. 4.35を比較するとLeaderが移動する場合には制御入力が大きくなっていることが確認できる．これはLeaderが移動する場合にはFollowerの制御入力にLeaderの移動速度に関するフィードフォワード項$\tilde{E}_i u_r$が付加されるためである．本実験ではLeaderの移動速度を0.1[m/sec]と小さく設定したため入力制約値$\eta$を変更しなかったが，より高速でLeaderが移動する場合には入力制約値$\eta$を小さく設定する必要がある．入力制約値$\eta$を小さくした場合，収束までにかかる時間が大きくなるため，さらに多くの予測ホライズンが必要となる．このような場合には，安全に予測ホライズンを長くとれる提案手法は特に有用となると考えられる．

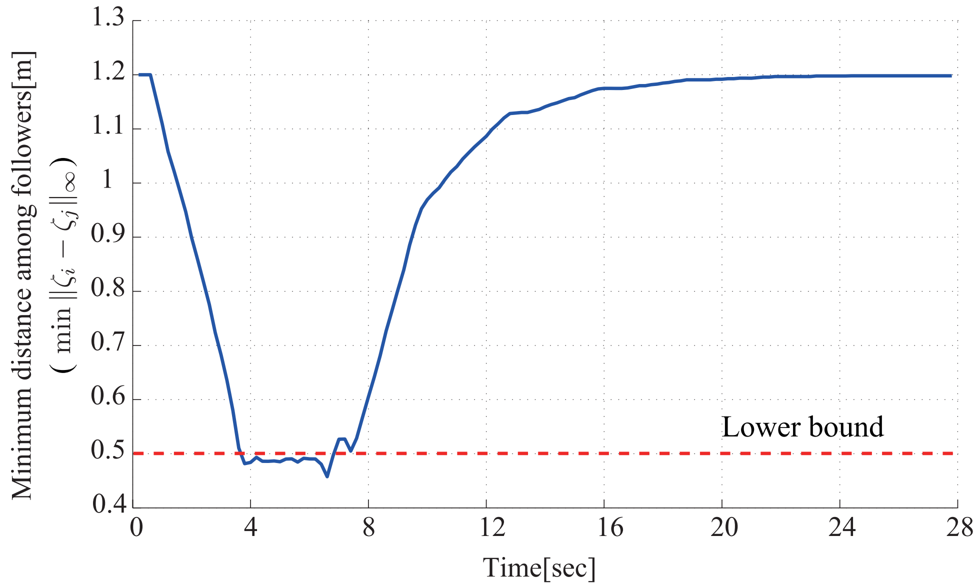



Fig. 4.29:Minimum distance among followers (experiment 1)

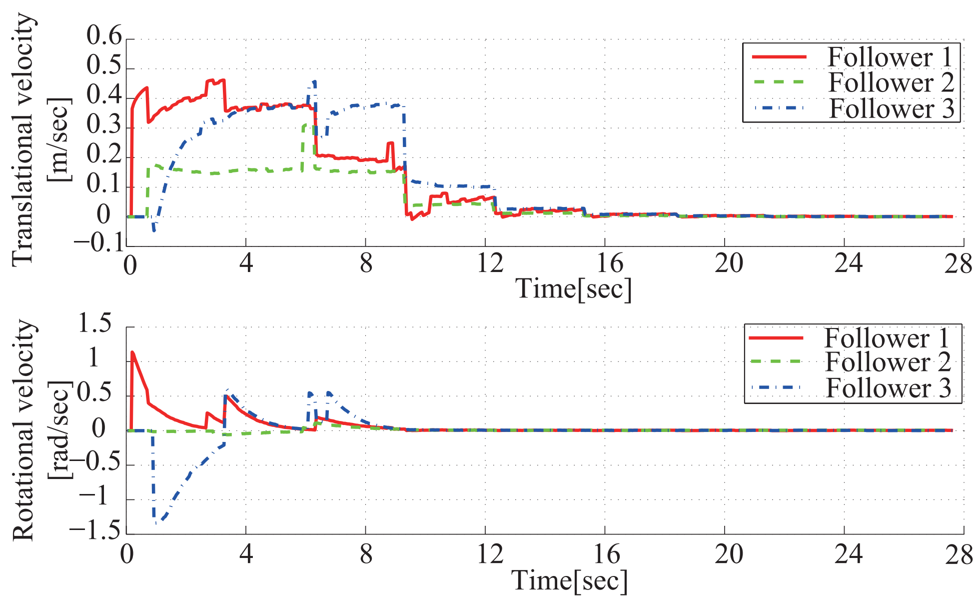



Fig. 4.30:Velocity inputs of followers (experiment 1)

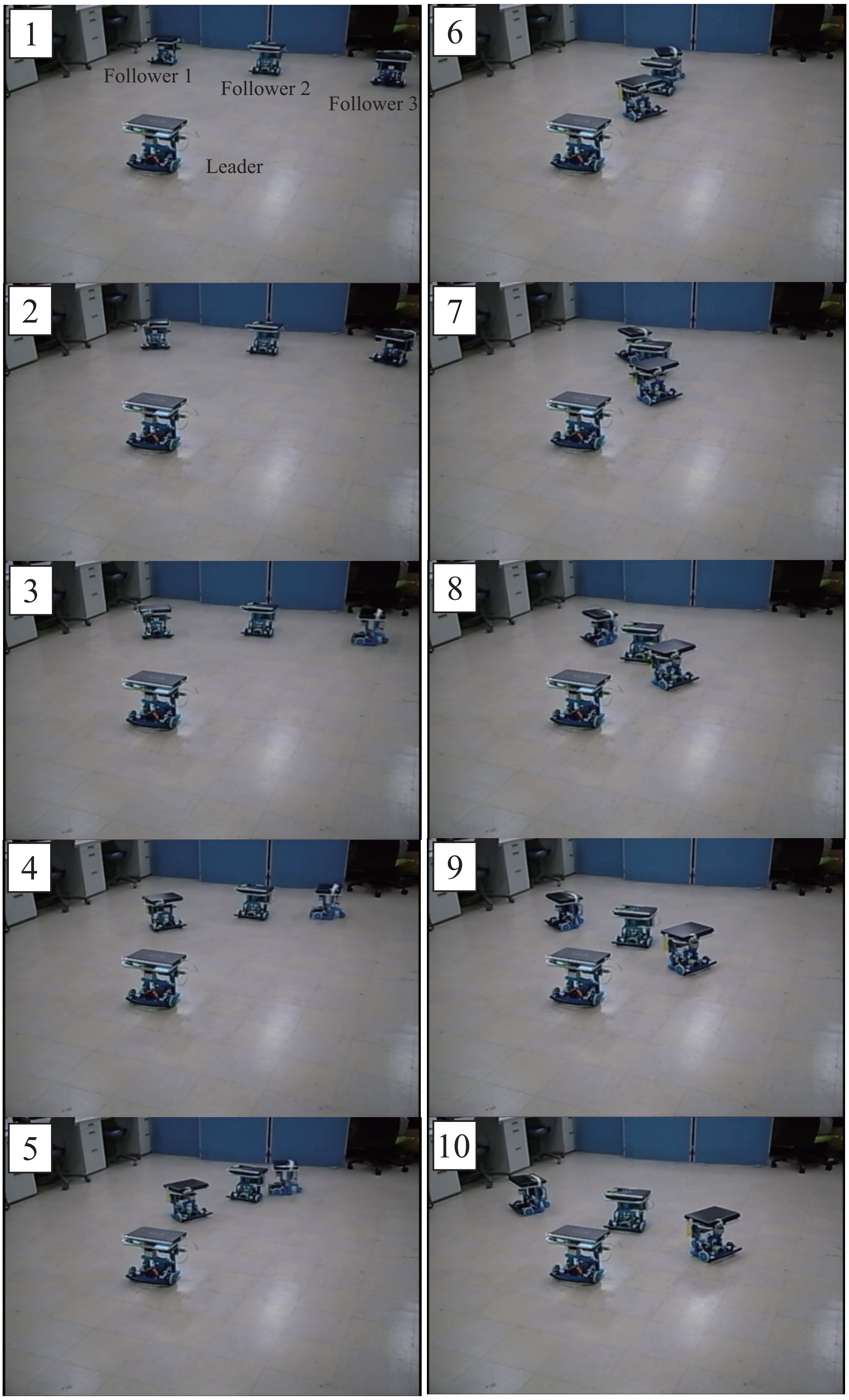



Fig. 4.31:The motion of vehicles in experiment 1

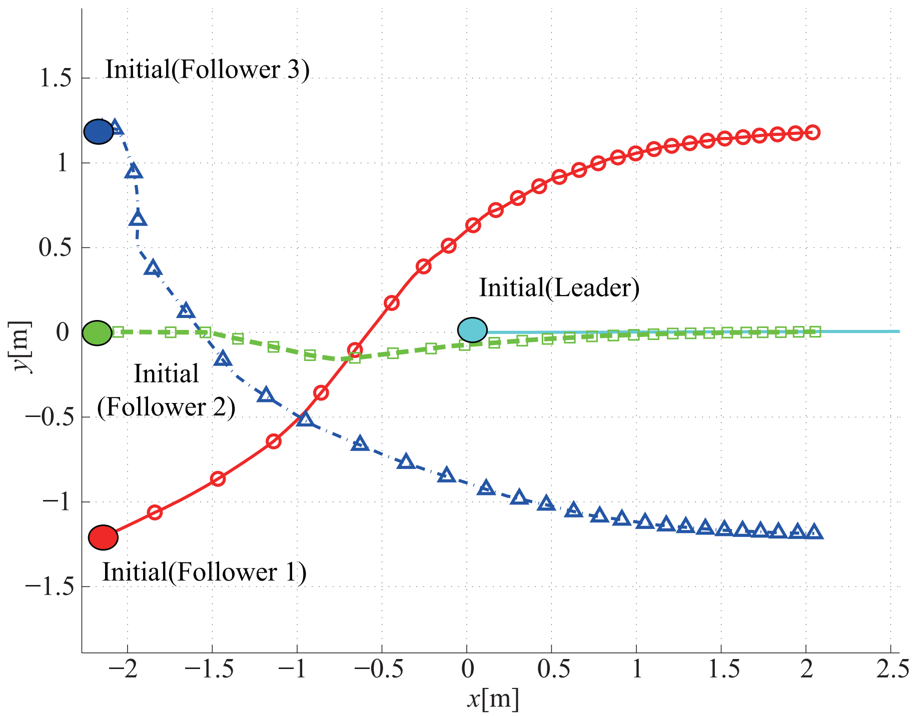



Fig. 4.32:Paths of the vehicles at the global frame (experiment 2)

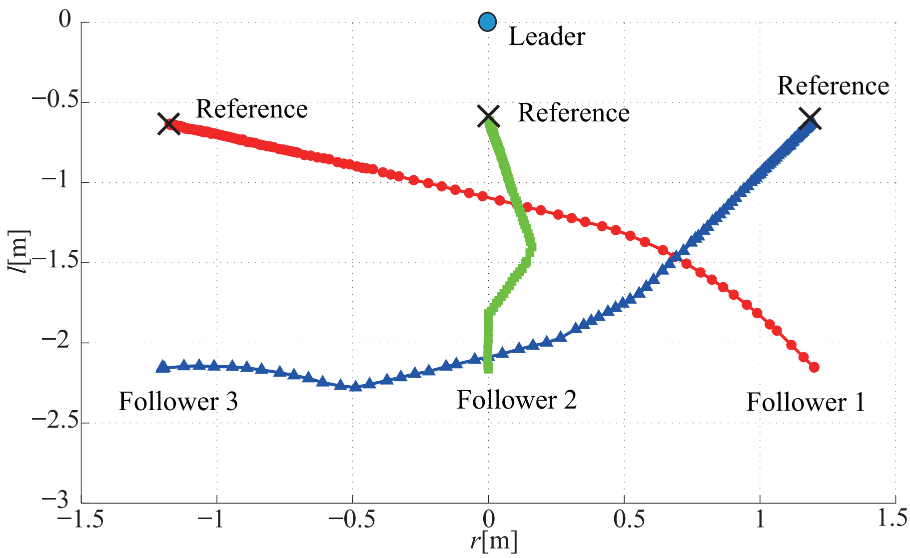



Fig. 4.33:Paths of the vehicles at the local frame (experiment 2)

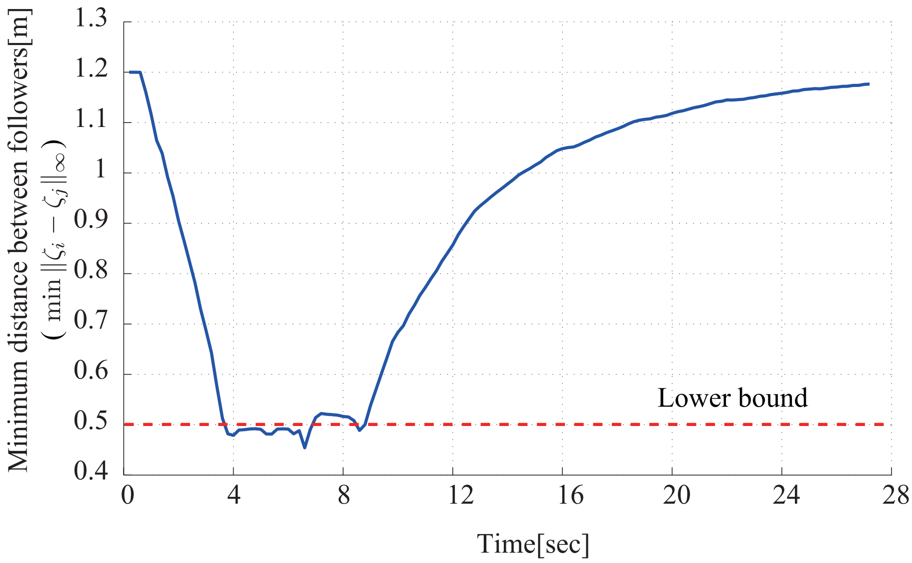



Fig. 4.34:Minimum distance among followers (experiment 2)

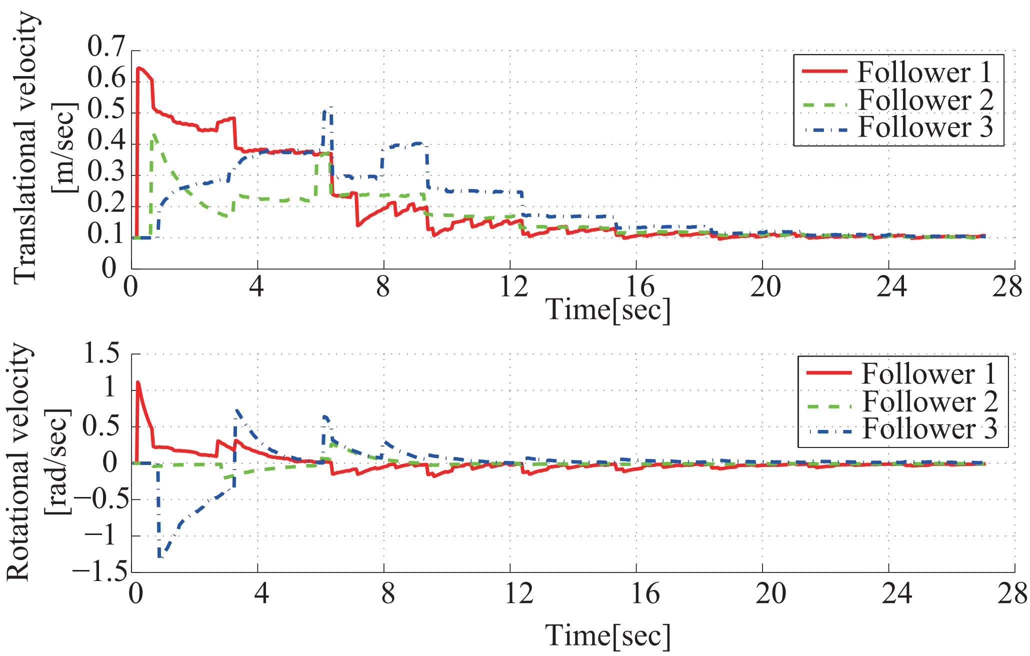



Fig. 4.35:Velocity inputs of followers (experiment 2)

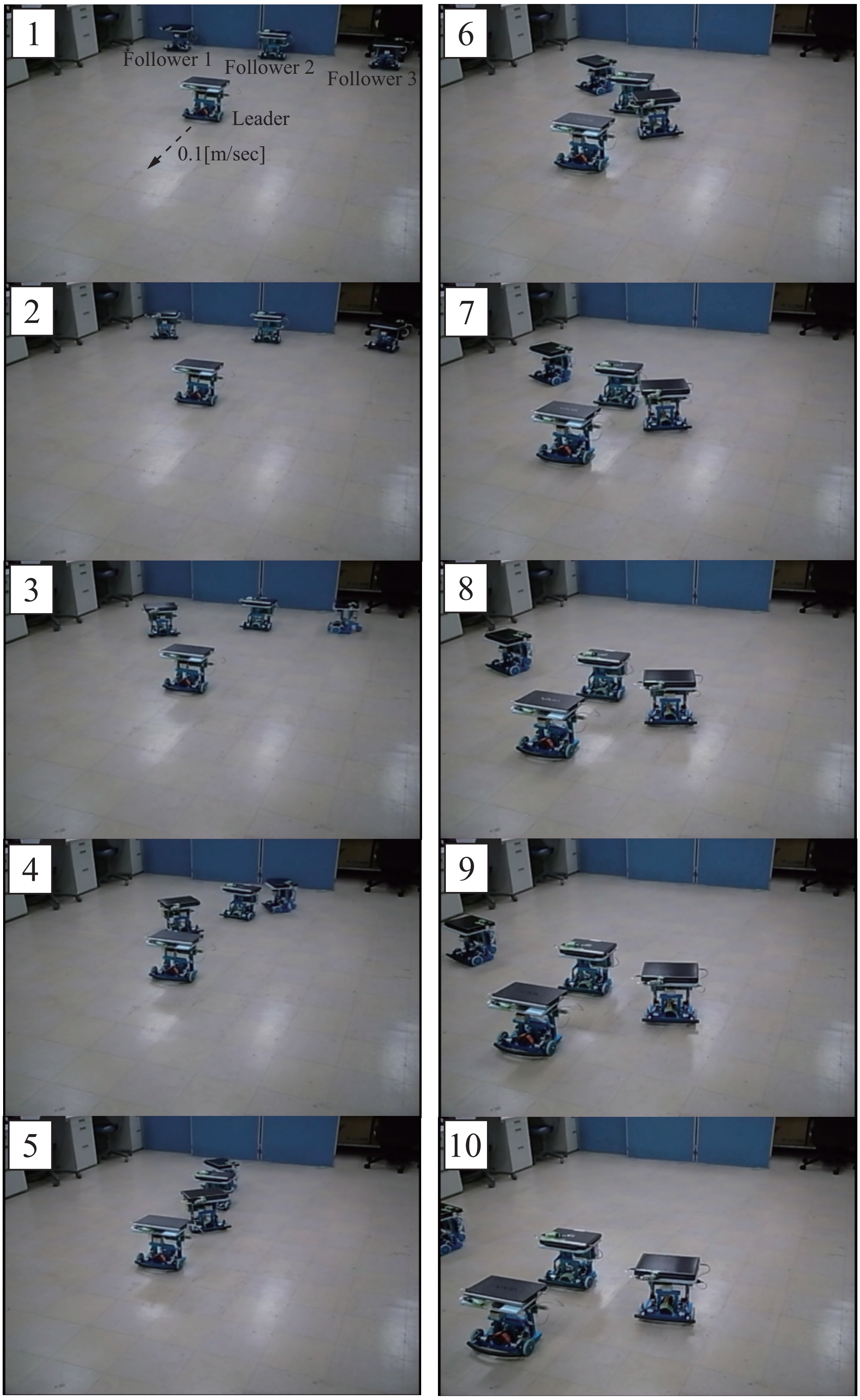



Fig. 4.36:The motion of vehicles in experiment 2

## 4.5 まとめ

本章では，モデル予測制御に基づく最適制御問題を実装する際に生じる，離散化の影響を陽に考慮した制御手法を提案した．

まず, 1台の移動体が静止障害物を回避しながら目標位置に移動する軌道計画問題に対し，連続時間上で障害物回避を保証する2つの手法を提案した．１つは可変最大速度法で障害物近傍でのみ最大速度を低減化することで保守性を低く抑えながらも予測時刻間の回避を保証するものである．もう１つの手法は遷移制約法で，障害物回避制約に含まれる0,1変数の遷移を制約することで安全な経路しか生成されないようにするものである．

次に，静止障害物回避に対して得られた知見を生かし，編隊形状切り替えに伴うFollower同士の衝突回避問題に対する遷移制約法に基づく手法を提案した．また，Follower同士の衝突回避においても連続時間上で衝突を起こさない安全な軌道が計画できることを示した．さらに，最適制御問題の可解性や閉ループ系の漸近安定性についても議論し，従来と同等の条件のもと保証が与えられることを示した．なお，本章では3.1.1節の衝突回避アルゴリズムについて議論を行ったが，2.3.2節の最適制御問題$P_0$のような1度に全ての移動体の制御入力を決定する形式に対しても適用可能である．これらの有効性をシミュレーションと実験により検証した．実験においては，通信遅れやモデル化誤差などの影響により衝突回避制約が侵害された場合でも遷移制約により確実に衝突が回避される方向に軌道が再計画されることが確認できた．

本章では，予測区間を一様に離散化することを基本に議論を行った．より直感的には現在から近い時刻は細かく，離れた場合には粗くすることが望ましいと考えられる，このように離散化を適当に変化させるなど，よりよい制御性能を実現する上には今後改良を図っていく必要があると考えられる．

なお，本章で提案した手法は予測モデルの性質に基づいたものである．より複雑な予測モデルに対して，提案手法を拡張することは今後の課題の１つであると考えられる．また，複数移動体の衝突回避問題に対しては遷移制約法のみ適用を行ったが，今回適用しなかった可変最大速度法を衝突回避手法に拡張することも今後の課題である．本章では離散化の影響を静止障害物回避とFollower同士の衝突回避とに分けて議論を行ったが，実際の環境中を走行する場合にはその両方を同時に考慮する必要がある．このように移動体同士の衝突回避と障害物回避を同時に考慮可能な制御系に統合していくことも今後の方向性の１つであると考えられる．

# 第5章
# 結論

## 5.1 まとめ

本論文では複数の移動体が編隊形状を形成・維持しながら移動する際に生じる移動体同士の衝突回避問題に対して，モデル予測制御に基づく衝突回避手法を提案した．特に，計算量低減化，離散化という観点からモデル予測制御に基づく新たな衝突回避手法を提案した．

第2章では，2輪車両を制御対象とし，2輪車両同士の衝突回避をモデル予測制御で取り扱うことを可能とする制御方策を提案した．提案手法は，フィードバック線形化とモデル予測制御を組み合わせたものであり，衝突回避だけでなく環境中に存在する障害物も同様に考慮できることを示した．

第3章では，モデル予測制御に基づく手法の計算量の問題に着目し，各移動体が順番に独立した最適制御問題を解くアルゴリズムを提案した．この手法は各更新時刻で解く最適制御問題のサイズを小さく抑え，計算量低減化を図るものであった．また，その最適制御問題の可解性と閉ループ系の安定性についても示し，各Followerが初期ステップで最適解を持つことができれば目標の編隊形状が達成できることを理論的に示した．次に，衝突回避問題の特性を考慮した分枝限定法を提案した．提案した分枝限定法は衝突回避問題の特性を考慮した分枝・限定ルールを導入するというものであり，シンプルな改良ながらも一般的な分枝限定法と比較し，大きく計算量が低減化できることをシミュレーションにより確認した．さらに，実験により提案した衝突回避手法の有効性を検証した．特に一般的な分枝限定法では困難であったFollowerが3台の場合でもオンラインで衝突回避が実現できることを示した．

第4章では，モデル予測制御に基づく衝突回避や障害物回避手法を実装する際に生じる離散化による影響を陽に考慮した制御方策について述べた．まず，問題の簡単化のために，1台の移動体が静止障害物を回避しながら目標位置まで移動する静止障害物回避問題を取り扱い，離散化が障害物回避に与える影響を明らかにした．この離散化による影響を考慮し，予測時刻間での障害物回避を考慮した2つの手法を提案した．1つは可変最大速度法で，障害物の近傍でのみ最大速度を低減化することで軌道の保守性は低く

抑えながらも，予測時刻間での障害物回避を保証した．もう1つは遷移制約法で，障害物回避制約に含まれる0,1変数の遷移を制約することで予測時刻間での障害物回避を保証した．次に，この知見に基づき，複数の移動体同士の衝突回避問題に対する，予測時刻間での衝突回避を保証する手法を提案した．提案した手法は遷移制約法に基づくものであり，予測時刻間での衝突回避だけでなく，最適制御問題の可解性と安定性についても保証を与えるものであった．これらの有効性はシミュレーションと実験により確認し，移動体群が移動しながら動的に形状を変化させような場合でもFollower同士の衝突を回避しながら目標の編隊形状を実現できることを示した．提案手法により本質的に安全な障害物回避や衝突回避が実現できたと考えられる．

以上，まとめると本論文では次のような点を提案・主張できたと考えられる．

- 2輪車両の衝突回避をモデル予測制御により考慮可能とする制御方策を提案した点
- 衝突回避の特性を考慮し，計算量低減化を図る制御方策と解法を提案した点
- 離散化が衝突回避に与える影響を検証し，予測時刻間でも衝突回避を保証する制御方策を提案した点
- 提案手法を実移動体に適用し有効性，妥当性を実機で示した点

## 5.2　今後の課題と展望

今後の課題としては，以下の項目が挙げられる．

3.1.1節では最適制御問題を順番に解く手法を提案したが，より一般的には各移動体が分散的・同時的に最適制御問題を解くことが望ましいと考えれる．このような枠組みに改良することが課題として挙げられる．また，各移動体が分散的・同時的に最適制御問題を解いた場合には通信の遅れなどが生じ，衝突回避に悪影響を与える可能性がある．したがって，ロバスト性の向上を図っていくなども今後の重要な課題であると考えられる．

4.2.2節では遷移制約法に基づく複数移動体同士の衝突回避手法を提案したが，もう１つの手法である可変最大速度法は拡張しなかった．可変最大速度法では，回避対象の近傍で速度が低減されるため，より安全性の高い衝突回避が実現できると期待できる．したがって，可変最大速度法を複数移動体同士の衝突回避に拡張することが今後の課題に挙げられる．

また，提案した衝突回避手法は予測区間を一様に離散化するものであったが，より直感的には現在から近い時刻は“細かく”，離れた場合には“粗く”することが望ましいと考えられる，このように離散化を適当に変化させるなど，よりよい制御性能を実現するためには今後改良を図っていく必要があると考えられる．さらに，本論文で提案した手法は1次系の予測モデルの性質に基づいたものである．運動学だけでなく動力学を考慮する場合など，より複雑な予測モデルを考える必要がある．このような系に対して提案手法を拡張することも今後の課題の１つである．

今後の展望としては，

- 障害物の考慮：
  本論文で提案した手法は特定の編隊形状を維持したまま障害物を回避することになる．この手法ではLeaderの挙動によってはFollowerが追従できない場合が存在する可能性がある．１つの解決法としてはLeaderの軌道計画に落とし込む[85]などが考えられるが，環境に応じて編隊形状を自動で生成し，動的に形状を変化させるようなことが望ましいと考えられる．これを行うことで，より一般的な環境での複数移動体の移動が実現できると考えられる．

- 環境情報の抽出：
  現状のモデル予測制御では暗に障害物情報といった環境情報はあらかじめ数式的に与えられるという仮定を必要としている．同様に，移動体の自己位置は外部から与えられるという仮定も必要としている．これらの仮定は，実環境では満たすことが困難な条件である．現実には不確かさを含むセンサ情報から環境情報の構築や自己位置推定を行わなくてはならない．すなわち，実環境で用いるには，移動体の自己位置推定と環境地図作成が必要不可欠な要素となる．これらを同時に実現する手法としてSLAM(Simultaneous localization and mapping)が知られている[86]．SLAMでは移動体の自己位置や環境地図は確定的に得られるのではなく，確率分布として表現される．すなわち，自己位置や障害物の位置の不確かさが陽に表現されるという利点がある(Fig. 5.1参照)．この不確かさを考慮することで，より安全性の高いロバストな編隊制御が実現できると期待できる．したがって，移動体の自己位置推定と環境地図構築をオンラインで実行可能なシステムの構築を行うとともに，移動体及び障害物の位置の不確かさを考慮した編隊制御則を構築することが必要であると考えられる．

- 非線形モデル予測制御への拡張：
  本論文では線形のモデル予測制御に基づき複数移動体同士の衝突回避

Fig. 5.1:Example of map generated by SLAM

や移動体の入力制約などを考慮した．より制御対象の特性を直接的に考慮するためには非線形モデル予測制御[87, 88]に拡張してくことが今後必要であると考えられる．特に，ソルバーの性能の向上に伴い，オンラインで解ける問題の幅が今後さらに広がって行くと考えられ，非線形モデル予測制御に基づく衝突回避も今後実現可能となっていくと考えられる．

などが挙げられる．以上のような課題を解決することで，より一般的な環境での運用が可能なモデル予測制御に基づく編隊制御系が実現できると考えれる．

# 付録A
# 最適制御問題の行列表現

本文中で示した最適制御問題は本章で示すような行列形式に変換された後，混合整数計画問題ソルバーによって解かれる．ここでは，最適制御問題 $P_0$ を例にその行列表現について説明する．ただし，衝突回避制約と障害物回避制約は(2.7)式の形式に変換されるものとする．

なお，以降で用いる $\mathbf{0}_{i\times j}$ は全ての要素が0の $i\times j$ 行列を，$\mathbf{1}_{i\times j}$ は全ての要素が1の $i\times j$ 行列，$I_j$ は $j\times j$ の単位行列を示すものとする．

## A.1 予測値の行列表現

以下では離散時刻予測モデル(3.11)を

$$e_i(\tau|k) = Ae_i(\tau-1|k) + B\alpha_i(\tau-1|k) \tag{A.1}$$

の形式で表わすものとする．ただし，$e_i \in \mathbb{R}^{2\times1}$，$\alpha_i \in \mathbb{R}^{2\times1}$，$A \in \mathbb{R}^{2\times2}$，$B \in \mathbb{R}^{2\times1}$ である．

このとき時刻 $t = k\delta$ における $e_i$ の予測値 $\mathbf{X_i}$ は初期状態 $e_i(k|k)$ と入力列 $\mathbf{U_i}$ を用いて以下のように表わすことができる．

$$\mathbf{X_i} = \mathbf{A}e_i(k|k) + \mathbf{B}\mathbf{U_i},\ \tau = k, \cdots, k+N-1 \tag{A.2}$$

ただし

$$\mathbf{X_i} := \begin{bmatrix} e_i(k+1|k) \\ e_i(k+2|k) \\ \vdots \\ e_i(k+N|k) \end{bmatrix} \in \mathbb{R}^{2N\times1},\ \mathbf{U_i} := \begin{bmatrix} \alpha_i(k|k) \\ \alpha_i(k+1|k) \\ \vdots \\ \alpha_i(k+N-1|k) \end{bmatrix} \in \mathbb{R}^{2N\times1}$$

$$\mathbf{A} := \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix} \in \mathbb{R}^{2N\times2},\ \mathbf{B} := \begin{bmatrix} B & 0 & \dots & 0 \\ AB & B & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ A^{N-1}B & A^{N-2}B & \dots & B \end{bmatrix} \in \mathbb{R}^{2N\times2N}$$

である．

## A.2　入力制約(3.13)の行列表現

入力制約は(3.13)式より

$$\|-\lambda e_i(\tau|k)+\alpha_i(\tau|k)\|_\infty \le \eta,\ \tau=k,\cdots,k+N-1 \tag{A.3}$$

と表わされる．(A.2)式に基づき(A.3)式を$e_i(k|k)$と$\mathbf{U_i}$で表現すると

$$(-\lambda\mathbf{C_2}\mathbf{B_2}+\mathbf{C_1})\mathbf{U_i} \le \mathbf{E_1}+\lambda\mathbf{C_2}\mathbf{A_2}e_i(k|k) \tag{A.4}$$

となる．ただし，

$$\mathbf{C_1} := \mathrm{diag}(C,C,\cdots,C)\in\mathbb{R}^{4N\times 2N},\ C := \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix}$$

$$\mathbf{E_1} := \eta\mathbf{1}_{4N\times 1},\ C_2 := \begin{bmatrix} \mathbf{C_1} & 0_{4N\times 2} \end{bmatrix}\in\mathbf{R}^{4N\times(2N+2)}$$

$$\mathbf{A_2} := \begin{bmatrix} I_2 \\ A \end{bmatrix}\in\mathbb{R}^{(2N+2)\times 2},\ \mathbf{B_2} := \begin{bmatrix} 0_{2\times 2N} \\ \mathbf{B} \end{bmatrix}\in\mathbb{R}^{(2N+2)\times 2N}$$

である．

## A.3　終端制約(3.14)の行列表現

終端制約は(3.14)式より

$$\|e_i(k+N|k)\|_\infty \le \gamma_i \tag{A.5}$$

と表わされる．ここで，左辺の$e_i(k+N|k)$は(A.2)式より初期状態$e_i(k|k)$と入力列$\mathbf{U_i}$で表現することが可能である．したがって，(A.5)式を以下のように表わすことができる．

$$\mathbf{C_3}\mathbf{B}\mathbf{U_i} \le \mathbf{E_2}-\mathbf{C_3}\mathbf{A}e_i(k|k) \tag{A.6}$$

$$\mathbf{C_3} := \begin{bmatrix} \mathbf{0}_{4\times 2(N-2)} & C \end{bmatrix}\in\mathbb{R}^{4\times 2N},\ \mathbf{E_2} := \gamma_i\mathbf{1}_{4\times 1}$$

## A.4　衝突回避制約(3.12)の行列表現

まず，回避対象のFollowerが1台の場合について考える．この場合，衝突回避制約は

$$\mathbf{C_1}\mathbf{X_i} \le \mathbf{M_1}\mathbf{\Phi_1}+\mathbf{\Psi_1}+\mathbf{C_1}\mathbf{X_j} \tag{A.7}$$

$$\mathbf{D_1}\mathbf{\Phi_1} \le 3I_N \tag{A.8}$$

と行列表現される．ただし，

$$\mathbf{D_1} := \mathrm{diag}(\mathbf{1}_{1\times 4}, \mathbf{1}_{1\times 4}, \cdots, \mathbf{1}_{1\times 4}) \in \mathbb{R}^{N\times 4N},\ \mathbf{M_1} := MI_{2N}$$

$$\mathbf{\Phi_1} := \begin{bmatrix} \begin{bmatrix} \kappa_{ij1}(k+1|k) \\ \kappa_{ij2}(k+1|k) \\ \kappa_{ij3}(k+1|k) \\ \kappa_{ij4}(k+1|k) \end{bmatrix}^T & \cdots & \begin{bmatrix} \kappa_{ij1}(k+N|k) \\ \kappa_{ij2}(k+N|k) \\ \kappa_{ij3}(k+N|k) \\ \kappa_{ij4}(k+N|k) \end{bmatrix}^T \end{bmatrix}^T \in \{0,1\}^{4N\times 1}$$

$$\mathbf{\Psi_1} := \begin{bmatrix} \psi \\ \psi \\ \vdots \\ \psi \end{bmatrix} \in \mathbb{R}^{4N\times 1},\ \mathbf{X_j} := \begin{bmatrix} e_j(k+1|k) - z_i^d + z_j^d \\ e_j(k+2|k) - z_i^d + z_j^d \\ \vdots \\ e_j(k+N|k) - z_i^d + z_j^d \end{bmatrix} \in \mathbb{R}^{2N\times 1}$$

であり，$\mathbf{X_j}$ はFollower jの状態にFollower $i$ の目標位置 $z_i^d$ とFollower $j$ の目標位置 $z_j^d$ が付加されたものである．ここで左辺の $\mathbf{X_i}$ を(A.2)式より，$\mathbf{U_i}$，$\mathbf{\Phi}_1$ を用いて表現すると以下のようになる．

$$\begin{bmatrix} \mathbf{C_1 B} & -\mathbf{M_1} \\ 0 & \mathbf{D_1} \end{bmatrix} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi_1} \end{bmatrix} \leq \begin{bmatrix} -\mathbf{C_1 A} \\ 0 \end{bmatrix} e_i(k|k) + \begin{bmatrix} \mathbf{\Psi_1} \\ 3\mathbf{1}_{N\times 1} \end{bmatrix} + \begin{bmatrix} \mathbf{C_1} \\ 0 \end{bmatrix} \mathbf{X_j} \tag{A.9}$$

次に，$n-1$ 台の回避すべき移動体が存在した場合は，(A.9)式を $n-1$ 回課すことで実現が可能である．したがって，Follower $\forall j \neq i$ との衝突回避制約は以下のような行列形式に変換される．

$$\begin{bmatrix} \mathbf{C_4 B} & -\mathbf{M_2} \\ 0 & \mathbf{D_2} \end{bmatrix} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi_2} \end{bmatrix} \leq \begin{bmatrix} -\mathbf{C_4 A} \\ 0 \end{bmatrix} e_i(k|k) + \begin{bmatrix} \mathbf{\Psi_2} \\ 3\mathbf{1}_{(n-1)N\times 1} \end{bmatrix} + \begin{bmatrix} \mathbf{C_4} \\ 0 \end{bmatrix} \mathbf{X_{js}} \tag{A.10}$$

ただし，

$$\mathbf{M_2} := \mathrm{diag}(\mathbf{M_1}, \mathbf{M_1}, \cdots, \mathbf{M_1}) \in \mathbb{R}^{2(n-1)N\times 2N}$$

$$\mathbf{D_2} := \mathrm{diag}(\mathbf{D_1}, \mathbf{D_1}, \cdots, \mathbf{D_1}) \in \mathbb{R}^{(n-1)N\times 4N}$$

$$\mathbf{C_4} := \begin{bmatrix} C_1 \\ C_1 \\ \vdots \\ C_1 \end{bmatrix} \in \mathbb{R}^{4(n-1)N\times 2N},\ \mathbf{\Phi_2} := \begin{bmatrix} \mathbf{\Phi_1} \\ \mathbf{\Phi_1} \\ \vdots \\ \mathbf{\Phi_1} \end{bmatrix} \in \{0,1\}^{4(n-1)N\times 1}$$

$$\mathbf{\Psi_2} := \begin{bmatrix} \mathbf{\Psi_1} \\ \mathbf{\Psi_1} \\ \vdots \\ \mathbf{\Psi_1} \end{bmatrix} \in \mathbb{R}^{4(n-1)N\times 1},\ \mathbf{X_{js}} := \begin{bmatrix} \mathbf{X_1} \\ \mathbf{X_2} \\ \vdots \\ \mathbf{X_n} \end{bmatrix} \in \mathbb{R}^{2(n-1)N\times 1}$$

である．また，$\mathbf{X_{js}}$ はFollower j($\forall j \neq i$)に関する $\mathbf{X_j}$ を並べた行列である．

## A.5　障害物制約(3.15)の行列表現

衝突回避制約と同様に，まず，障害物が1つしか存在しない場合について考える．障害物$j$のみが存在する場合，障害物回避制約(A.9)式と同様の行列形式で表現できる．

$$\begin{bmatrix} \mathbf{C_1B} & -\mathbf{M_1} \\ 0 & \mathbf{D_1} \end{bmatrix} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi_3} \end{bmatrix} \leq \begin{bmatrix} -\mathbf{C_1A} \\ 0 \end{bmatrix} e_i(k|k) + \begin{bmatrix} \mathbf{\Psi_3} \\ 3\mathbf{1}_{N\times1} \end{bmatrix} + \begin{bmatrix} \mathbf{C_1} \\ 0 \end{bmatrix} \mathbf{X_{oj}} \tag{A.11}$$

ただし，

$$\mathbf{\Phi_3} := \begin{bmatrix} \begin{bmatrix} \kappa_{io1}(k+1|k) \\ \kappa_{io2}(k+1|k) \\ \kappa_{io3}(k+1|k) \\ \kappa_{io4}(k+1|k) \end{bmatrix}^T & \cdots & \begin{bmatrix} \kappa_{io1}(k+N|k) \\ \kappa_{io2}(k+N|k) \\ \kappa_{io3}(k+N|k) \\ \kappa_{io4}(k+N|k) \end{bmatrix}^T \end{bmatrix}^T \in \{0,1\}^{4N\times1}$$

$$\mathbf{\Psi_3} := \begin{bmatrix} \begin{bmatrix} \psi_{io1}(k+1|k) \\ \psi_{io1}(k+1|k) \\ \psi_{io2}(k+1|k) \\ \psi_{io2}(k+1|k) \end{bmatrix}^T & \cdots & \begin{bmatrix} \psi_{io1}(k+N|k) \\ \psi_{io1}(k+N|k) \\ \psi_{io2}(k+N|k) \\ \psi_{io2}(k+N|k) \end{bmatrix}^T \end{bmatrix}^T \in \mathbf{R}^{4N\times1}$$

$$\mathbf{X_{oj}} := \begin{bmatrix} o_j(k+1|k) - z_i^d \\ o_j(k+2|k) - z_i^d \\ \vdots \\ o_j(k+N|k) - z_i^d \end{bmatrix} \in \mathbb{R}^{2N\times1}$$

である．

同様に障害物が$m$個存在する場合には，(A.11)式を$m$回繰り返せばよいことになる．したがって，$m$個の障害物との障害物回避制約は以下のような行列形式に変換される．

$$\begin{bmatrix} \mathbf{C_5B} & -\mathbf{M_3} \\ 0 & \mathbf{D_3} \end{bmatrix} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi_4} \end{bmatrix} \leq \begin{bmatrix} -\mathbf{C_5A} \\ 0 \end{bmatrix} e_i(k|k) + \begin{bmatrix} \mathbf{\Psi_4} \\ 3\mathbf{1}_{mN\times1} \end{bmatrix} + \begin{bmatrix} \mathbf{C_5} \\ 0 \end{bmatrix} \mathbf{X_{os}} \tag{A.12}$$

ただし,

$$\mathbf{M_3} := \mathrm{diag}(\mathbf{M_1}, \mathbf{M_1}, \cdots, \mathbf{M_1}) \in \mathbb{R}^{2mN\times 2N}$$
$$\mathbf{D_3} := \mathrm{diag}(\mathbf{D_1}, \mathbf{D_1}, \cdots, \mathbf{D_1}) \in \mathbb{R}^{mN\times 4N}$$
$$\mathbf{C_5} := \begin{bmatrix} C_1 \\ C_1 \\ \vdots \\ C_1 \end{bmatrix} \in \mathbb{R}^{4mN\times 2N},\ \mathbf{\Phi_4} := \begin{bmatrix} \mathbf{\Phi_3} \\ \mathbf{\Phi_3} \\ \vdots \\ \mathbf{\Phi_3} \end{bmatrix} \in \{0,1\}^{4mN\times 1}$$
$$\mathbf{\Psi_4} := \begin{bmatrix} \mathbf{\Psi_{o1}} \\ \mathbf{\Psi_{o2}} \\ \vdots \\ \mathbf{\Psi_{om}} \end{bmatrix} \in \mathbb{R}^{4mN\times 1},\ \mathbf{X_{os}} := \begin{bmatrix} \mathbf{X_{o1}} \\ \mathbf{X_{o2}} \\ \vdots \\ \mathbf{X_{on}} \end{bmatrix} \in \mathbb{R}^{2mN\times 1}$$

である.また,

## A.6 評価関数(3.10)の行列表現

ここでは$e_i$に関する項を含む評価関数$J$の行列形式を示す.

$$J = \sum_{\tau=k+1}^{k+N} e_i(\tau|k)^T Q e_i(\tau|k) + \alpha_i(\tau-1|k)^T R \alpha_i(\tau-1|k) \tag{A.13}$$

なお,評価関数(3.10)には$e_i$に関する項が含まれていないが,ここでは説明のため付加している.評価関数(3.10)の場合には以下の式において$Q=0$とすればよい.

評価関数(A.13)を$\mathbf{X_i}$と$\mathbf{U_i}$を用いて表現すると

$$J = \mathbf{X_i^T Q X_i} + \mathbf{U_i^T R U_i} \tag{A.14}$$

と表わされる.ただし,$\mathbf{Q} := \mathrm{diag}(Q, Q, \cdots, Q) \in \mathbb{R}^{2N\times 2N}$, $\mathbf{R} := \mathrm{diag}(R, R, \cdots, R) \in \mathbb{R}^{2N\times 2N}$である.さらに,$\mathbf{X_i}$を(A.2)式より初期状態$e_i(k|k)$と$\mathbf{U}_i$で書き改めると

$$\begin{aligned} J &= (\mathbf{A}e_i(k|k) + \mathbf{BU_i})^T \mathbf{Q}(\mathbf{A}e_i(k|k) + \mathbf{BU_i}) + \mathbf{U_i^T R U_i} \\ &= \begin{bmatrix} e_i(k|k)^T & \mathbf{U}_i^T \end{bmatrix} \left( \begin{bmatrix} A^T & B^T \end{bmatrix} \mathbf{Q} \begin{bmatrix} A \\ B \end{bmatrix} + \begin{bmatrix} 0 & \\ & \mathbf{R} \end{bmatrix} \right) \begin{bmatrix} e_i(k|k) \\ \mathbf{U}_i \end{bmatrix} \end{aligned} \tag{A.15}$$

となる.ここで

$$L := \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{A}^T & \mathbf{B}^T \end{bmatrix} \mathbf{Q} \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} + \begin{bmatrix} 0 & \\ & \mathbf{R} \end{bmatrix}$$

とおき，(A.15) 式を整理すると

$$J = \mathbf{U_i}^T L_{22} \mathbf{U_i} + 2e_i(k|k)^T L_{12} \mathbf{U_i} + \text{const.} \tag{A.16}$$

となる．なお，(A.16) 式の左辺第3項は初期状態のみに依存して決定される項である．

混合整数計画問題では連続値変数$\mathbf{U_i}$だけでなく，衝突回避制約(A.10)および障害物回避制約(A.12)に含まれる0,1変数$\mathbf{\Phi} := [\Phi_2^T \ \Phi_4^T]^T$も同時に決定する．$\mathbf{\Phi}$を考慮すると最終的に評価関数は以下のように表わされる．

$$J = \begin{bmatrix} \mathbf{U_i}^T & \mathbf{\Phi}^T \end{bmatrix} \mathbf{H} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi} \end{bmatrix} + \mathbf{F} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi} \end{bmatrix} \tag{A.17}$$

ただし，

$$H := \begin{bmatrix} L_{22} & \\ & 0 \end{bmatrix},\ F := \begin{bmatrix} 2e_i(k|k)^T L_{12} & 0 \end{bmatrix} \tag{A.18}$$

## A.7　最終的な行列表現

最終的に最適制御問題$P_0$は(A.4), (A.6), (A.10), (A.12), (A.17) 式より以下のような行列で表現される．

$$\min_{\mathbf{U_i},\mathbf{\Phi}} \left( \begin{bmatrix} \mathbf{U_i}^T & \mathbf{\Phi}^T \end{bmatrix} \mathbf{H} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi} \end{bmatrix} + \mathbf{F} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi} \end{bmatrix} \right) \tag{A.19}$$

subject. to

$$\begin{bmatrix} -\lambda\mathbf{C_2B_2} + \mathbf{C_1} & 0 & 0 \\ \mathbf{C_3B} & 0 & 0 \\ \mathbf{C_4B} & -\mathbf{M_2} & 0 \\ 0 & \mathbf{D_2} & 0 \\ \mathbf{C_5B} & 0 & -\mathbf{M_3} \\ 0 & 0 & \mathbf{D_3} \end{bmatrix} \begin{bmatrix} \mathbf{U_i} \\ \mathbf{\Phi} \end{bmatrix} \leq \begin{bmatrix} \lambda\mathbf{C_2A_2} \\ -\mathbf{C_3A} \\ -\mathbf{C_4A} \\ 0 \\ -\mathbf{C_5A} \\ 0 \end{bmatrix} e_i(k|k) + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \mathbf{C_4} & 0 \\ 0 & 0 \\ 0 & \mathbf{C_5} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{X_{js}} \\ \mathbf{X_{os}} \end{bmatrix} + \begin{bmatrix} \mathbf{E_1} \\ \mathbf{E_2} \\ \mathbf{\Psi_2} \\ 3\mathbf{1}_{(n-1)N\times 1} \\ \mathbf{\Psi_4} \\ 3\mathbf{1}_{mN\times 1} \end{bmatrix}$$

このように最適制御問題で決定する変数として連続値変数$U_i$だけでなく，0,1変数$\mathbf{\Phi}$も含まれるため，混合整数二次計画問題となる．

# 参考文献

[1] J. Clark and R. Fierro: Copperative hybrid control of robotic sensors for perimeter detection and tracking, *Proc. of American Control Conference*, pp.3500-3505 (2005)

[2] S. I. Roumeliotis and G. A. Bekey: Distributed multirobot localization, *IEEE Trans. on Robotics and Automation*,18(5), pp.781-795 (2002)

[3] D. C. K. Yuen and B. A. Macdonald: Vision-based localization algorithm based on landmark matching, triangulation, reconstruction, and comparison, *IEEE Trans. on Robotics*,21(2), pp.217-228 (2005)

[4] S. Thrun, W. Burgard, and D. Fox: A Real-Time Algorithm for Mobile Robot Mapping With Applications to Multi-Robot and 3D Mapping, *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, pp. 321–328 (2000)

[5] J. Ryde and H. Hu: Cooperative Mutual 3D Laser Mapping and Localization, *Proc. of IEEE Int. Conf. on Robotics and Biomimetics*,pp1048-1053 (2006)

[6] A. Howard: Multi-robot Simultaneous Localization and Mapping using Particle Filters, *Int. Journal of Robotics Research*, 25(12), pp. 1243-1256 (2006)

[7] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun: Collaborative multi-robot exploration, *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 476–481 (2000)

[8] N. Miyata, J. Ota, T. Arai, and H. Asama: Cooperative transport by multiple mobile robots in unknown static environments associated with real-time task assignment, *IEEE Trans. on Robotics and Automation*, 18(5), pp.769-780 (2002)

[9] H.Weimerskirch, J.Martin, Y.Clerquin, P.Alexander, and S. Jiraskova: Energy saving in flight formation, *Nature*, 414, pp. 679–698 (2001)

[10] 有元：魚はなぜ群れで泳ぐか，大修館書店 (2007)

[11] C.W. Reynolds: Flocks, Herds, and schools: Adistributed behavioral model, *Computer Graphics*, 21-4, pp. 25–34 (1987)

[12] D. Gr*ü*bbaum: Schooling as a strategy for taxis in a noisy environment, *Evolutionary Ecology*, 12,pp. 503–522 (1998)

[13] T. Vicsek, A. Czirok, E. Ben-Jacobs, I. Cohen, and O. Sochet: Novel type of phase transition in a system of self-driven particles, *Phys. Rev. Lett.*, 75(6),pp. 1226–1229 (1995)

[14] P. K. C. Wang and F. Y. Hadaegh : Coordination and control of multiple microspacecraft moving in formation, *The Journal of the Astronautical Sciences*, 44(3), pp. 315–355 (1996)

[15] J. P. Desai, J. P. Ostrowski, and V. Kumar : Modeling and control of formations of nonholonomic mobile robots, *IEEE Trans. on Robotics and Automation*, 17(6), pp. 905–908 (2001)

[16] A. K. Das, R. Fierro, and V. Kumar : A vision-based formation control framework, *IEEE Trans. on Robotics and Automation*, 18(5), pp. 813–825 (2002)

[17] R. Cui, S. S. Ge, B. V. E. How, and Y. S. Choo : Leader- follower formation control of underactuated auvs with leader position detection, *IEEE Int. Conf. on Robotics and Automation*, pp. 979–984(2009)

[18] H. J. Min, A. Drenner, and N. Papanikolopoulos : Vision-based leader-follower formations with limited information, *IEEE Int. Conf. on Robotics and Automation*, pp. 351–356(2009)

[19] H. G. Tanner, G. J. Pappas, and V. Kumar : Leader-to-formation stability, *IEEE Trans. on Robotics and Automation*, 20(3), pp. 443–455 (2004)

[20] T. Balch and R. C. Arkin: Behavior-based formation control for multirobot teams, *IEEE Trans. on Robotics and Automation*, 14(6), pp. 926–939 (1998)

[21] L. E. Parker : ALLIANCE: An architecture for fault tolerant multirobot cooperation, *IEEE Trans. on Robotics and Automation*, 14(2), pp. 220–240 (1998)

[22] M. Schneider-Fontán and M. J. Matarić : Territorial multi-robot task division, *IEEE Trans. on Robotics and Automation*, 14(5), pp. 815–822 (1998)

[23] M. A. Lewis and K.-H. Tan : High precision formation control of mobile robots using virtual structures, *Autonomous Robots*, 4(4), pp. 387–403 (1997)

[24] P. Ögren, M. Egerstedt, and X. Hu : A control lyapunov function approach to multiagent coordination, *IEEE Trans. on Robotics and Automation*, 18(5), pp. 847–851 (2002)

[25] W. Ren and R. Beard : Decentralized scheme for spacecraft formation flying via the virtual structure approach, *AIAA Journal of Guidance, Control, and Dynamics*, 27(1), pp. 73–82 (2004)

[26] J. R. T. Lawton, R. W. Beard, and B. J. Young : A decentralized approach to formation maneuvers, *IEEE Trans. on Robotics and Automation*, 19(6), pp. 933–941 (2003)

[27] R. W. Beard, J. Lawton, and F. Y. Hadaegh : A coordination architecture for spacecraft formation control, *IEEE Trans. on Control Systems Technology*, 9(6), pp. 777–790 (2001)

[28] W. Ren : Consensus based formation control strategies for multi-vehicle systems, *Proc. of American Control Conference*, pp. 4237–4242 (2006)

[29] T. Eren, P. N. Belhumeur, B. D. O. Anderson, and A. S. Morse : A framework for maintaining formations based on rigidity, *Proc. of the IFAC World Congress*, pp. 2752–2757 (2002)

[30] J. A. Fax and R. M. Murray : Information flow and cooperative control of vehicle formations, *IEEE Trans. Automatic Control*, 49(9), pp. 1465–1476 (2004)

[31] R. Olfati-Saber, J. A. Fax, and R. M. Murray : Consensus and cooperation in networked multi-agent systems, *Proc. IEEE*, 95(1), pp. 215–233 (2007)

[32] K. D. Listmann, M. V. Masalawasa, and J. Adamy : Consensus for formation control of nonholonomic mobile robots, *IEEE Int. Conf. on Robotics and Automation*, pp. 3886–3891(2009)

[33] S. Martinez, J. Cortes, and F. Bullo : Motion coordination with distributed information, *IEEE Control Systems Magazine*, 27(4), pp. 75–88 (2007)

[34] V. Gervasi and G. Prencipe: Coordination without communication:the case of flocking problem, *Discrete applied mathmatics*, 143, pp.203-223 (2004)

[35] J. Barraquand, B. Langlois, and J. C. Latombe : Numerical potential field techniques for robot path planning, *IEEE Transactions on Systems, Man and Cybernetics*, 22(2), pp. 224–241 (1992)

[36] N. E. Leonard and E. Fiorelli : Virtual leaders, artificial potentials and coordinated control of groups, *Proc. IEEE Int. Conf. Decision and Control*, pp. 2968–2973 (2001)

[37] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopoulos: Nonholonomic navigation and control of cooperating mobile manipulators, *IEEE Trans. on Robotics and Automation*, 19(1), pp. 53–64 (2003)

[38] E. Rimon and D. E. Koditschek: Exact robot navigation using artificial potential functions, *IEEE Trans. on Robotics and Automation*, 8(5), pp. 501–518 (1992)

[39] H. G. Tanner and A. Kumar: Formation stabilization of multiple agents using decentralized navigation functions, *Robotics: Science and Systems I*, S. Thrun, G. Sukhatme, S. Schaal and O. Brock (eds), MIT Press, pp. 49–56 (2005)

[40] E. G. Hernandez-Martinez and E. Aranda-Bricaire: Non-collision conditions in multi-agent robots formation using local potential functions, *IEEE Int. Conf. on Robotics and Automation*,pp. 3776–3781 (2008)

[41] P. Urcola, L.Riazuelo, M.T.Lázaro, and L.Motano; Cooperative navigation using environment compliant robot formations, *IEEE Int. Conf. on Intelligent Robots and Systems*, pp. 2789–2794 (2008)

[42] S. S. Ge and F. L. Lewis: Autonomous mobile robots, sensing, control, decision making and applications, Chapter 11: Multi-robot cooperation; pp. 417–459 , Taylor and Francis (2006)

[43] J.Shao, G. Xie, J. Yu, and L. Wang: Leader-following formation control of multiple mobile robots; *Proc. of the IEEE Int. Sym. on Intelligent Control*, pp. 808–813 (2005)

[44] J. Jongsuk and T. Mita : Tracking Control of Multiple Mobile Robots: A case Study of Inter-Robot Collision-Free Problem, *Proc. of IEEE ICRA*, pp2885–2890 (2001)

[45] L. Barnes, M. Fields, and K. Valavanis: Unmanned ground vehicle swarm formation control using potential fields, *Proc. of the 15th Med. Conf. on Control and Automation*, T33-004 (2007)

[46] J. Jurachart, 美多, 山北: 切り替え型制御による衝突回避を考慮した複数編隊無人飛行機の形成制御, システム制御情報学会論文誌, 17(1), pp. 26–38 (2004)

[47] J.How, E. King, and Y. Kuwata: Flight Demonstrations of Cooperative Control for UAV Teams, *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, (2004)

[48] E. Fiorelli, N. E. Leonard, P. Bhatta, D. Paley: Multi-AUV Control and Adaptive Sampling in Monterey Bay, *Proc. IEEE Autonomous Underwater Vehicles 2004: Workshop on Multiple AUV Operations*, pp. 134–147 (2004)

[49] D. B. Edwards, T.A. Bean, D.L. Odell, and M.J. Anderson: A Leader-Follower Algorithm for Multiple AUV Formations, *IEEE/OES Autonomous Underwater Vehicles*, pp. 40–46 (2004)

[50] S. M. Veres and N. K. Lincoln: Vision Assisted Satellite Formation Control, *Proceedings of IEEE Conference on Decision and Control*, pp. 5712–5717 (2006)

[51] J. J. C. M. Bik, P. N. A. M. Visser, and O. Jennrich: LISA satellite formation control, *Adbances in Space Research*, 40, pp. 25–34 (2007)

[52] P. Song and R. V. Kumar: A potentional field based approach to multi-robot manipulation, *IEEE Int. Conf. on Robotics and automation*,Volume 2, pp. 1217-1222 (2002)

[53] A. Sudsang, F. Rothganger, and J. Ponce: Motion planning for disc-shaped robots pushing a polygonal object in the plane, *IEEE Trans. on Robotics and Automation*,18(4), pp. 550-562 (2002)

[54] G.A.S. Pereira, M.F.M. Campos and V. Kumar: Decentralized algorithms for multi-robot manipulation via caging, *International journal of Roboticas Research*, 23(7-8), pp. 783-795 (2004)

[55] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O. Scokaert: Constrained model predictive control: Stability and optimality, *Automatica*, 36(6), pp.789–814 (2000)

[56] J.M. Maciejowski : Predictive Control with Constraints, Prentice Hall (2002)

[57] H. Fukushima and R. R. Bitmead: Robust constrained predictive control using comparison model, *Automatica*, 41(1), pp.97–106 (2005)

[58] H. Seguchi and T. Ohtsuka: Nonlinear Receding Horizon Control of an Underactuated Hovercraft, *International Journal of Robust and Nonlinear Control*,13, pp. 381-398(2003)

[59] G. Klancar and I. Skrjanc: Tracking-error model-based predictive control for mobile robots in real time, Robotics and Autonomous Systems, 55, pp.460–469(2007)

[60] A. Richards and J. How : Model predictive control of vehicle maneuvers with guaranteed completion time and robust feasibility, *Proc. of American Control Conference*, pp. 4034–4040 (2003)

[61] A. Richards and J. How : A decentralized algorithm for robust constrained model predictive control, *Proc. of American Control Conference*, pp. 4261–4266 (2004)

[62] Y. Kuwata, T. Schouwenaars, and J. How: Distributed robust receding horizon control for multivehicle guidance;, *IEEE Trans. on Control Systems Technology*, 15(4), pp. 627–641 (2007)

[63] T. Schouwenaars, B. D. Moor, E. Feron, and J. How : Mixed Integer Programming for Multi-Vehicle Path Plannning, *Procedings of the European Control Conference*, pp. 2603–2608 (2001)

[64] T. Kopfstedt, M. Mukai, M Fujita, and O. Sawody: A networked formation control for groups of mobile robots using mixed integer programming, *Proc. of the IEEE Int. Conf. on Control Applications*, pp. 579–584 (2006)

[65] T. Schouwenaars, J. How, and E. Feron : Receding horizon path planning with implicit safty guarantees, *Proc. of American Control Conference*, pp. 5576–5581 (2004)

[66] M.G.Earl and R.D'Andrea : Iterative MILP Methods for Vehicle-Control Problems, *IEEE Trans. on Robotics*, 21(6), pp. 1158–1167 (2005)

[67] H. L. Hagenaars, J. Imura, and H. Nijmeijer : Approximate continuous-time optimal control in obstacle avoidance by time/space discretization of non-convex state constraints, *IEEE Int. Conf. Control Applications*, pp. 878-883 (2004)

[68] M. D. Berg, O.Cheong, M. V. Kreveld, and M. Overmars: Computational geometry:Algorithms and applications, Springer (2008)

[69] R. K. Miller and A. N. Michel: Ordinary differential equations, Academic Press (1982)

[70] B. Korte and J. Vygen : Combinatorial optimization : Theory and algorithms, Springer-Verlag (2002)

[71] Y. Guo and T. Tang : Optimal trajectory generation for nonholonomic robots in dynamic environments, *IEEE Int. Conf. on Robotics and Automation*, pp. 2552–2557 (2008)

[72] D. Wooden and M. Egerstedt : Oriented visibility graphs : Low-complexity planning in real-time environments, *IEEE Int. Conf. on Robotics and Automation*, pp. 2354–2359 (2006)

[73] O. Khatib : Real-time obstacle avoidance for manipulators and mobile robots, *Int. Journal of Robotics Research*,5(1), pp. 90–98 (1986)

[74] J. J. Kuffner : Efficient optimal search of uniform-cost grids and lattices, *IEEE/RSJ Int. Conf. on Inttelligent Robots and Systems*, pp. 1946–1951 (2004)

[75] D. Ferguson and A. Stentz : Using interpolation to improve path planning: The field d* algorithm, *Journal of Field Robotics*, 23(2), pp. 79–101 (2006)

[76] P. O. M . Scokaert and J. B. Rawlings: Constrained linear quadratic regulation, *IEEE Trans. on Automtic Control*, 43(8), pp. 1163–1169 (1998)

[77] M. Ishikawa, T. Seki, J. Imura, and S. Hara: An efficient algorithm for optimal control of hybrid dynamical systems utilizing mode transition rule, *IEEE Conf. on Decision and Control*, pp. 1866–1871(2004)

[78] 譚, 平田: 計算幾何学入門　幾何アルゴリズムとその応用, 森北出版 (2001)

[79] MATLAB Optimization Toolbox User's Guide,The MathWorks, Inc.

[80] A.Bemporad : MIQP.m, http://www.dii.unisi.it/hybrid

[81] MATLAB Compiler User's Guide,The MathWorks, Inc.

[82] CPLEX, http://www.ilog.co.jp/products/cplex/, ILOG

[83] N. Ando, T.Suehiro, K. Kitagaki, T. Kotoku, and W. Yoon: RT-Component object model in RT-Middleware - Distributed component middleware for RT (Robot Technology)-, *IEEE Proc. of Int. Symposium on Computational Intelligence in Robotics and Automation*,We-B2-5(2005)

[84] MATLAB External Interfaces User's Guide,The MathWorks, Inc.

[85] P. Ogren and N. E. Leonard: Obstacle avoidance in formation, *IEEE Int. Conf. on Robotics and Automation*, pp. 2492–2497 (2003)

[86] S.Thrun, W. Burgard, and D. Fox: Probablistic robotics, MIT Press (2005)

[87] 大嶋, 関: モデル予測制御-V-非線形モデル予測制御, システム/制御/情報, 47(1), pp. 52–57 (2003)

[88] J. L. Piovesan and H. G. Tanner : Randomized model predictive control for robot navigation, *IEEE Int. Conf. on Robotics and Automation*, pp. 94–99(2009)

# 業績一覧

## 博士論文関連業績

### 論文誌

[1 ] 根和幸，福島宏明，松野文俊：衝突回避を考慮した複数移動体のモデル予測編隊制御；計測自動制御学会誌，42(8)，pp.877–883(2006)

[2 ] 根和幸，福島宏明，松野文俊：衝突回避問題に適合した分枝限定法に基づく複数移動体の編隊制御；計測自動制御学会誌，44(1)，pp.36–43(2008)

[3 ] 根和幸，福島宏明，松野文俊：予測時刻間の障害物回避を考慮したモデル予測制御に基づく軌道計画法; 計測自動制御学会誌，45(8)，pp.406–413(2009)

[4 ] 根和幸，福島宏明，松野文俊：予測時刻間の衝突回避を考慮した複数移動体のモデル予測編隊制御；計測自動制御学会誌，(投稿中)

### 国際会議

[1 ] H. Fukushima, K. Kon, and F. Matsuno: Distributed Model Predictive Control for Multi-Vehicle Formation with Collision Avoidance Constraints, *Proc. IEEE Conf. on Decision and Control*, pp. 5480-5485 (2005)

[2 ] K. Kon, H. Fukushima, and F. Matsuno: Multi-vehicle formation control based on branch-and-bound method compatible with collision avoidance problem, *IEEE Conf. on Intelligent Robots and Systems*, pp. 3771-3776 (2007)

[3 ] K. Kon, H. Fukushima, and F. Matsuno: Trajectory Generation based on Model Predictive Control with Obstacle Avoidance between Prediction Time Steps, *SYROCO*, F3B3 (2009)

### 国内会議

[1 ] 根和幸，福島宏明，松野文俊，衝突回避問題に適合した分枝限定法に基づく複数移動体の編隊制御，計測自動制御学会　第7回制御部門大会，71-1-3，調布 (2007)

[2 ] 根和幸，佐藤徳孝，福島宏明，Chattarjee Ranajit，五十嵐広希，松野文俊，金城隆也，田所諭，高森年，自律と操縦に対応した移動ロボット用RTCの開発第4報：編隊制御モジュール群，第26回日本ロボット学会学術講演会，1F3-10，(2008)

# 博士論文関連外業績

## 国際会議

[1 ]H. Fukushima, K.i Kon, F. Matsuno,Y. Hada, K. Kawabata, and H. Asama: Constrained Predictive Control for Multi-Vehicle Formation and an Autonomous Blimp, *SICE - ICASE International Joint Conference*, pp. 4515-4520 (2006)

[2 ]N. Shiroma, K. Kon, and F. Matsuno: Environment data collection and its use for robot teleoperation, *The 3rd International Conf. Ubiquitous Robots and Ambient Intelligence*, MP2-5 (2006)

[3 ]K. Kon, Y. Urano, N. Shiroma, N. Sato, Y. Fujino, H. Fukushima, and F. Matsuno: Development of Robot Teleoperation System in Bad Viewing Condition, *IEEE Conf. on Robotics and Biomimetics*, pp. 427-432 (2006)

[4 ]H. Fukushima, K. Kon, Y. Hada, F. Matsuno, K. Kawabata, and H. Asama: State-Predictive Control of an Autonomous Blimp in the Presence of Time Delay and Disturbance, *IEEE Conf. on Control Applications*, pp. 188–193 (2007)

[5 ]N. Sato, K. Kon, H. Fukushima, F. Matsuno: Map-based Navigation Interface for Multiple Rescue Robots, *IEEE Int. Workshop on Safety, Security, and Rescue Robotics*, pp. 152–157 (2008)

[6 ]H. Mizumoto, N. Sato, K. Kon, H. Mano, H. Shin, R. Chatterjee, and F. Matsuno: Flexible Interface for Multiple Autonomous and Teleoperated Rescue Robots, *IEEE Conf. on Robotics and Biomimetics*, pp. 1844–1849 (2008)

[7 ]H. Mano, K. Kon, N. Sato, M. Ito, H. Mizumoto, K. Goto, R. Chatterjee, and F. Matsuno: Treaded Control System for Rescue Robots in Indoor Environment, *IEEE Conf. on Robotics and Biomimetics*, pp. 183–1843 (2008)

## 国内会議

[1 ]城間直司，根和幸，松野文俊：環境地図構築とその遠隔操作利用，日本機械学会第16回インテリジェント・システム・シンポジウム講演会，pp. 253–256 (2006)

[2 ]宮澤克規，根和幸，佐藤徳孝，伊藤誠崇，水本尚志，真野隼人，大原伸介，藤田充典，松野文俊：屋外環境における複合センサ群を用いた自律移動ロボットの開発RWRC (Real World Robot Challenge) に向けて，計

測自動制御学会　第8回システムインテグレーション部門講演会, 3C3-1 (2007)

[3 ] 宮澤克規，根和幸，佐藤徳孝，伊藤誠崇，水本尚志，真野隼人，大原伸介，藤田充典，松野文俊：つくばチャレンジ2007に向けた屋外型自律移動ロボットFUMA type-Rの開発と成果報告，つくばチャレンジ開催記念シンポジウム，バンダイナムコゲームス未来研究所ファンシアター (2008)

[4 ] 佐藤徳孝，根和幸，福島宏明，Chattarjee Ranajit，五十嵐広希，松野文俊，長谷川晶一，金城隆也，田所諭，高森年：自律と操縦に対応した移動ロボット用RTCの開発第3報複数ロボットのための地図上ナビゲーションインターフェイスモジュール，第26回日本ロボット学会学術講演会，1F-09 (2008)

[5 ] 根和幸，佐藤徳孝，五十嵐広希，岩切淳，Chattarjee Ranajit，松野文俊，金城隆也，田所諭，高森年：自律と操縦に対応した移動ロボット用RTCの開発第7報：RWRCにおける屋外自律ナビゲーションシステムの開発，計測自動制御学会　第9回システムインテグレーション部門講演会，1I4-3，岐阜 (2008)

[6 ] 真野隼人，根和幸，佐藤徳孝，伊藤誠崇，水本尚志，後藤清宏，松野文俊：屋内環境におけるレスキューロボットの遠隔・自律切り替えシステム　計測自動制御学会　第9回システムインテグレーション部門講演会，2A3-1，岐阜 (2008)

[7 ] 水本尚志，佐藤徳孝，根和幸，真野隼人，新隼人，Ranajit Chatterjee，松野文俊：複数台の自律・遠隔レスキューロボットのための柔軟な操作インタフェースの開発，計測自動制御学会　第9回システムインテグレーション部門講演会，2A3-2，岐阜 (2008)

[8 ] 根和幸，佐藤徳孝，五十嵐広希，岩切淳，後藤清宏，金井僚太郎，Chatterjee Ranajit，松野文俊，金城隆也，田所諭，高森年：屋外自律移動ロボット用RTCの開発とRWRCでの実証実験, つくばチャレンジ開催記念シンポジウム, バンダイナムコゲームス未来研究所ファンシアター (2009)

[9 ] 後藤清宏，根和幸，松野文俊, 自律と操縦に対応した移動ロボット用RTCの開発第10報:速度制約領域を考慮した自律移動ロボットの行動計画，第27回日本ロボット学会学術講演会，3D1-07，横浜 (2009)

[10 ] 佐藤徳孝，根和幸，松野文俊，齋藤俊久，田所論，高森年，自律と操縦に対応した移動ロボット用RTCの開発第12報ロボット操縦用iPhone通信モジュール，第27回日本ロボット学会学術講演会，3D1-06，横浜 (2009)

# 謝　辞

本研究を遂行するにあたり，長きに渡り適切な御指導と御助言を頂きました京都大学工学研究科松野文俊教授，福島宏明助教に心より感謝申し上げます．本論文の執筆に当たって様々な御意見，御助言を頂きました，京都大学工学研究科椹木哲夫教授，泉田啓教授に深く感謝致します．また，研究を進めるに当たって御指導，御助言を頂きました，神戸大学工学研究科横小路泰義教授，電気通信大学知能機械工学科長谷川晶一准教授，慶応大学メディアデザイン研究科稲見昌彦教授に深く感謝致します．研究全体を通じよき御意見，御指導を頂きました，茨城大学知能システム工学科城間直司准教授，岡山大学自然科学研究科亀川哲志講師，山梨大学医学工学総合研究部大原伸介助教，岐阜大学人間情報システム工学科遠藤孝浩助教，八戸工業大学工学研究科藤澤隆介助教，京都大学工学研究科佐藤徳孝特定助教に深く感謝いたします．

また，議論や実験をはじめとする様々なご協力を頂いた田中基康氏，ともに研究活動に励んだ松野研究室，横小路研究室の皆様に感謝致します．

最後に，学部から9年間という長きにわたり生活面を支えていただいた両親に感謝の意を表し，本論文の締めくくりとさせて頂きます．