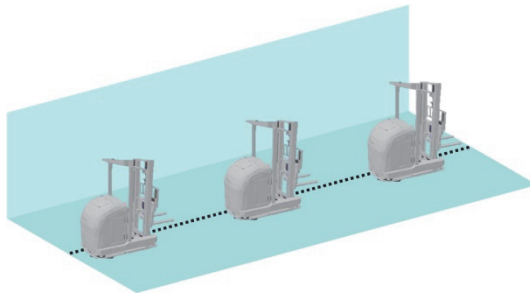


複数の無人ビークル協調作業のための群知能技術の開発

Development of The Swarm Intelligence System for Multi-vehicle Cooperation



藤島 泰郎*¹
Yasuo Fujishima

高木 一茂*¹
Kazushige Takaki

荒木 亮次*¹
Ryoji Araki

杉本 喜一*²
Kiichi Sugimoto

東 俊一*³
Shun-ichi Azuma

無人フォークリフトや水中ビークルなどの当社無人ビークル製品では、搬送効率最大化・ビークル同士の衝突防止・一定のフォーメーション維持等、複雑な制約条件下でビークル群を目標位置に到達させるため経路計画が必要となる。しかしサーバに情報を全て集約してサーバが全てのビークルの経路を計画すると、サーバに過負荷がかかる場合や通信が途絶した場合、サーバは経路計画処理ができず群全体が停止する。そのため、処理を各ビークルに分散することによる負荷低減が必要となる。本研究では、複雑な制約条件下における経路計画を各ビークルで分散処理させる、モデル予測経路計画に基づいた 10 数台規模のビークルの群知能技術を、名古屋大学との共同研究で開発した。

1. はじめに

物流現場では多数のフォークリフトが稼働しているが、近年の少子高齢化などの影響による作業員の減少が懸念されている。そのため、今後は多数の無人ビークルが走行するスマートファクトリーの実現を目指し、無人フォークリフト(Automated Guided Forklift; AGF)の導入による作業の一部自動化が必要になると考えられる。しかし現状の AGF は、走行エリアを占有して他の AGF はそのエリアに侵入できない排他制御方式に基づいた運用となっており、複数の AGF が同じ目的位置に向かって多数の荷物を搬送する場合の搬送効率は低い。このような課題を解決するために、我々は、複数の AGF の搬送効率を向上するための群知能技術の開発に取り組んできた。従来のクライアント・サーバ方式では、倉庫管理システムサーバが AGF 群全体の行動を制御することにより、AGF 群が互いに衝突せず隊列走行しながら大量の物資を運搬する、という動作を実現できるものと考えられる。しかしこの方式は AGF の台数に対するスケーラビリティがなく、例えば AGF の経路計画にモデル予測制御(Model Predictive Control; MPC)のような計算負荷の高い技術を導入すると、サーバの負荷が高くなりシステムがダウンする恐れがあった。

そこで本研究では、AGF の経路計画の MPC 処理を各 AGF に分散処理させることによる、負荷低減を図った。図1に、サーバ型 MPC と分散型 MPC の違いを示す。分散型 MPC を採用することにより、AGF の台数が増えても最適化すべき変数(将来の制御指令値の系列)の次元は常に同じであるため、将来のスマートファクトリーに適していると考ええる。本報では、このような分散型 MPC により複数の AGF の経路を計画する方法の一つについて述べ、15 台の AGF モデルを用いたシミュレーション結果を示す。

*1 ICT ソリューション本部 CIS 部

*2 ICT ソリューション本部 CIS 部 首席技師 技術士(情報工学部門)

*3 名古屋大学大学院 工学研究科 教授 工博

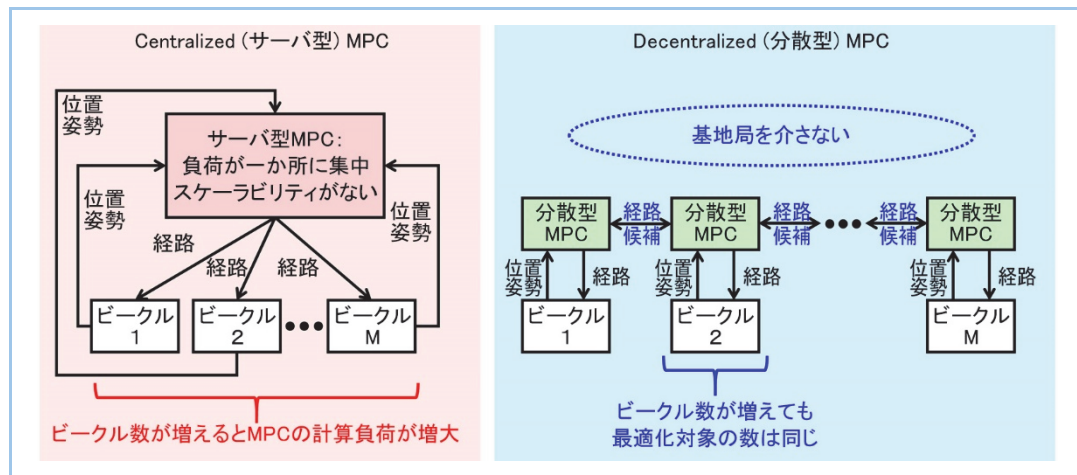


図1 サーバ型と分散型の違い

2. 前提条件・問題設定

本研究では、物流倉庫内における AGF の搬送効率[回/(h・台)]向上を例題とし、複数の AGF を独立に移動させるのではなく、互いに衝突しないよう一定のフォーメーションを維持したまま目的位置へ向かう場合を考える。図2に、本研究で想定したユースケースを示す。約100m 四方のエリア内を15台のAGFが走行し、AGFは同じ目的位置を有する5台で一つのチーム(小集団)を形成する。図2には15台のAGFが存在するが、これらは必ず三つのチームのいずれかに所属し、各チームには必ずリーダーが一つ存在し、リーダーは他のチームのリーダーとの車-車間通信ができるものと仮定する。また、リーダー以外のAGFをフォロワーと呼称する。フォロワーは同一チーム内においてのみ通信できるものとする。

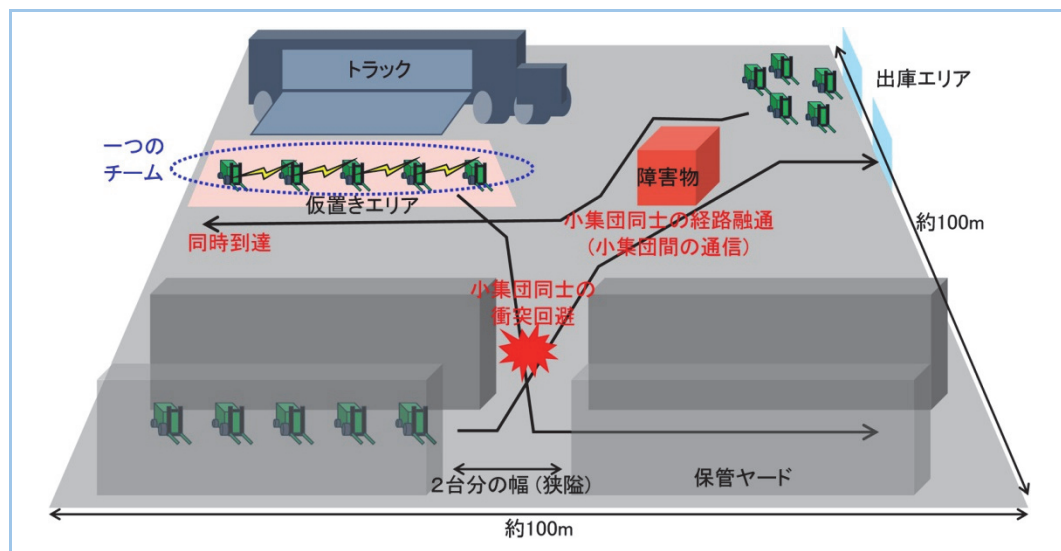


図2 複数のAGFによる荷物搬送の様子

このようなユースケースは、図3に示すような、異なる種類の荷物を載せたトラックからAGFが物流倉庫内に荷物を入庫する場合が該当する。図3は、在庫管理のしやすさ等の理由により、同じメーカーの荷物(図中同じ色で示した箱)は互いに近い場所で保管されている場合を想定した図である。同一メーカーの荷物を入庫エリアから同じ保管場所に効率良く搬送するためには、複数のAGFがチームを組み一定のフォーメーション(隊列走行等)を保って目的位置(保管場所)へ移動することが必要であると考えられるため、図2・図3のユースケースを本研究の例題として設定した。なお、図3破線で示したAGFの軌跡は、出庫時の様子を表している。出庫時は入庫時とは逆に、AGFは異なる保管場所から荷物を出庫エリアに集約する必要があるため、AGF群は“チ

ーム”を組むことが困難な可能性があり、今後の課題として本研究の想定ユースケースから除外した。図2・図3のユースケースの前提条件を表1のように設定した。

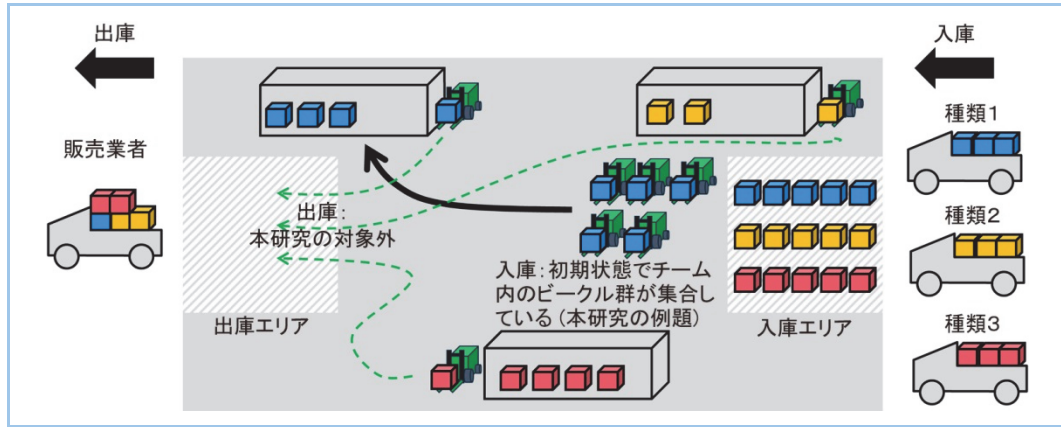


図3 複数のAGFによる入庫時の様子

表1 前提条件

No.	前提	備考
1	同一目的(目的位置など)を持つ AGF 群を一つのチームと位置付ける。	各チームの目的位置は既知情報である(管理システムから与えられる)。
2	AGF は走行時のチーム内フォーメーションを与えられている。	—
4	リーダーは、他のリーダー及び及び自チームのフォロワーとの通信が常時可能である。	リーダーは他チームのフォロワーとは通信しない。
5	フォロワーは自チームのリーダー・自チームの他のフォロワーとの通信が常時可能である。	フォロワーは他チームのリーダー・フォロワーとは通信しない。
6	AGF の自己位置姿勢及び速度情報は既知であり、計測誤差はない。	—
7	AGF 間の通信遅れはない。	—
8	AGF は倉庫内レイアウトの地図情報を保有しており、壁等の進入禁止エリアの座標の真値を知っている。	進入禁止エリア:壁・保管エリア・柵 等
9	AGF は他の AGF と衝突してはいけない。	—
10	AGF は非ホロノミックなシステムであり、真横に移動することができない。	駆動輪:前輪1輪・後輪2輪はパッシブ
11	AGF は旋回半径・速力とも制約があり、上下限を超えて動作することはできない。	旋回半径の制約は駆動輪の舵角制限に対応している。

以下に、本報中で使用する変数の定義を示す：

- $k \in R$: 離散時間
- $I \in R$: チームの数
- $i = 1, 2, \dots, I$: チームのインデックス
- $J \in R$: チーム内の AGF の数
- $j = 1, 2, \dots, J$: AGF のインデックス
- $L \in R$: AGF のホイールベース
- P_{ij} : AGF (i, j) の運動モデル
- C_{ij} : AGF (i, j) の群知能システム(本論文で述べるシステム)
- $p_{ij} = [x_{ij} \ y_{ij} \ \theta_{ij}]^T$: AGF (i, j) の2次元平面上の位置・姿勢
- $u_{ij} = [\phi_{ij} \ v_{ij}]^T$: AGF (i, j) の操舵角・車速
- $Z_i \in R, i = 1, 2, \dots, I$: チーム i の走行経路の長さ
- $w_i \in R^{Z_i \times 1}, i = 1, 2, \dots, I$: チーム i の走行経路の番号(ノード)
- N : AGF の経路の長さ
- $p_{ij}^-(k+n) \in R^{3 \times 1}, n = 1, 2, \dots, N$: AGF (i, j) の目標位置姿勢(詳細経路)

本報では、各チームのリーダーのインデックスを $i = 1$ とする。

3. 提案手法

3.1 開発方針

図2, 図3のユースケースを実現するためには, 以下の二つの課題が存在する:

課題① 経路計画処理を分散化した際に生じる, デッドロックの回避

課題② 複雑な制約条件下における, 分散化した際の個別経路計画処理の高速化

課題①の打ち手として, 他のチームが移動する可能性がある領域に侵入しないよう, チーム間で経路候補の情報交換を行い, 経路計画時の制約条件に反映することを考えた。また, 課題②の打ち手として, モデル予測経路計画を高速化するアルゴリズム Continuous/Generalized Minimal Residual (C/GMRES)⁽¹⁾を適用し, 個々の分散処理における経路計画の処理速度を向上させた。

群知能技術を実現する際, 各フォロワーが同時に経路を計画するアーキテクチャが考えられる。この方式では, リーダーはフォロワーの存在を考慮せず, 目標位置までの経路候補を生成する。しかし, そのままではリーダーはフォロワーの針路を妨害する可能性があるため, リーダーは計画した経路を自己位置姿勢とともにフォロワーに送信する。一方, フォロワーはリーダーの針路を妨害しないように, 所定のフォーメーションを達成できるよう, リーダーから受信した経路をオフセットしてリーダーに追従する。我々はリーダーの経路をチーム全体の経路とみなせることに着目し, リーダーが存在するという前提条件の下, 各フォロワーによる個別の経路計画や同一チーム内のフォロワー間の通信による経路の調整が不要な経路計画技術を開発した。

図4に, 我々が開発した経路計画システムのブロック線図を示す。図中の(a)と(b)はそれぞれリーダーとフォロワーのブロック線図を表している。図中, P_{ij} はAGF実機の運動モデルを表しており, 本研究では以下のノンホロノミックモデルを仮定している。

$$P_{ij}: \begin{cases} \dot{x}_{ij}(k) = v_{ij}(k) \cos \theta_{ij}(k) \\ \dot{y}_{ij}(k) = v_{ij}(k) \sin \theta_{ij}(k) \\ \dot{\theta}_{ij}(k) = (v_{ij}(k)/L) \tan \phi_{ij}(k) \end{cases}, i = 1, 2, \dots, I, j = 1, 2, \dots, J \text{ ここで, } \dot{x} \text{ は } x \text{ の時間微分}$$

を表す。また, 図中の u_{ij}^{ref} は制御指令値を表す。リーダーのシステム C_{i1} の入力は, 自己位置姿勢 $p_{i1}(k)$ に加えて, 他のチームとの接近を避けるために他のチームの走行経路の番号(ノード) $w_l, l \neq i$ も含む。他チームの走行経路の番号は, 倉庫管理システムから受信し, リーダーは自分のチームの走行経路番号を倉庫管理システムに送信する。一方, (b)に示すフォロワーのシステム $C_{ij}, j = 2, 3, \dots, J$ の入力は, 自己位置姿勢 $p_{ij}(k)$ に加えて, チーム内の他のビークル $m \neq j$ の位置姿勢 $p_{im}(k)$ 及びリーダーが出力するチーム全体の計画経路 $\hat{p}_{i1}(k+n), n = 1, 2, \dots, N$ も含む。

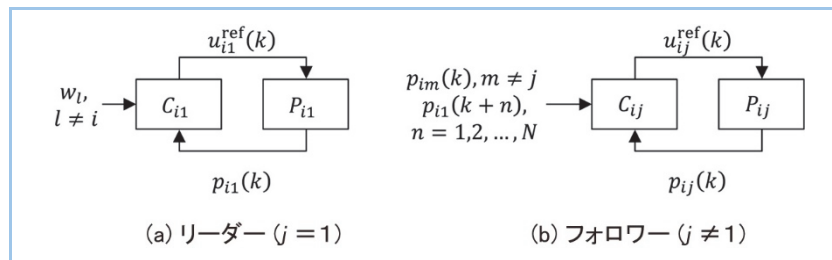


図4 提案法のブロック線図

3.2 チーム間の衝突回避手法

全ビークル間での衝突回避のための経路調整はスケーラビリティがないため, 無向グラフによるチーム単位での衝突回避のアプローチを採用した。まず, AGF 群が稼働する倉庫レイアウトの地図情報から, AGF が通行可能なエリアの代表点がグラフのノードとして定義されているものとする。同様に, ノード間の通行可能な道がグラフのエッジとして定義されているものと仮定する。チー

ム間の衝突回避について、図5を用いて説明する。図中のノード#1 のエリアにチーム1が存在し、ノード#2 のエリアにチーム2が存在している場合を考える。初期状態では、チーム1・2のいずれも図中左側に示すグラフ G_0 を保有している。ある時刻においてチーム1に目的位置のノード番号(図中#4 のノード)が与えられた際、チーム1のリーダーはそのノード#4 までの最短経路をダイクストラ法などグラフ探索アルゴリズムにより求め、図5に示すように#1・#3・#4 のノード系列を得る。このノード番号の系列(ウェイポイント)が、図4に示した w_l である。最短経路探索後、チーム1のリーダーは、自分のチームが通過する予定のウェイポイントを他の全てのチームのリーダーに送信した後、移動を開始する。次に、他のチームのリーダーは、チーム1から送信されたノードと接続されているエッジ(図5中の破線で示したエッジ)を全て削除し、新たなグラフ G_1 を得る。これにより、チーム1の通過予定のノードを他のチームが通過できなくなるため、チーム間での接近が発生し難くなる。その後、チーム2が目的位置のノード#6 を与えられた際は、グラフ G_1 を用いて最短経路を探索することで、図中右側の赤線で示したように#2・#5・#8・#7・#6 のウェイポイントを得る。この方法の課題はチームが通行する経路が他のチームにとって常に通行禁止となる点である。本課題に対しては、例えば移動速度を考慮してチームが走行する時間帯だけ他チームのグラフからエッジを削除すれば、他チームも時間差でその経路を走行できるようになる。

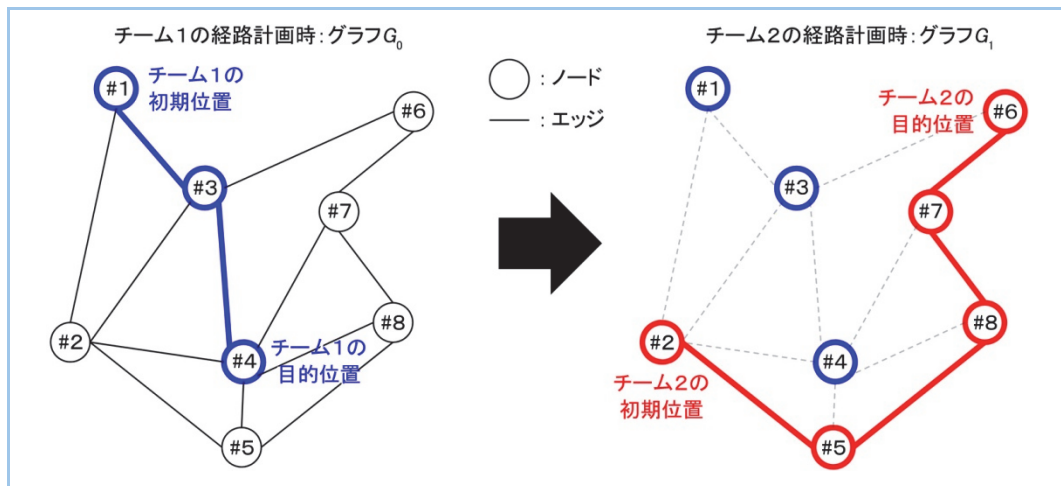


図5 チーム間の衝突防止のための無向グラフ

3.3 チーム内の経路計画

AGF はリーダー・フォロワー問わず互いの衝突や禁止領域への進入を避ける必要があるため、我々は MPC のアプローチを採用することとした。

$$C_{ij}: U_{ij}(k) = \underset{U_{ij}}{\operatorname{argmin}} f_{ij}(U_{ij}, p_{ij}(k)), \text{ subject to } g_{ij}(U_{ij}, p_{ij}(k)) < 0 \quad \text{ここで} \quad U_{ij}(k) \triangleq$$

$[u_{ij}(k) \ u_{ij}(k+1) \ \dots \ u_{ij}(k+N-1)]^T$ は計画経路に対応した AGF の制御指令値の系列を表す。制御指令値と計画経路を対応付ける運動モデルは P_{ij} である。リーダーとフォロワーでは役割が異なるため、最適化の評価関数 f_{ij} 及び制約条件 g_{ij} はリーダーとフォロワーで異なる。リーダーはチーム全体の経路を決める役割を有するため、次のノードまでチーム全体が到達できるよう、次のノードの位置と経路の偏差を最小化する評価関数を採用した。また、禁止領域に侵入しないように制約条件を設定した。一方、フォロワーは、リーダーから受信した計画経路・所望フォーメーションの情報に基づいて、自分の経路を生成する。所望フォーメーションは、リーダーからの相対距離・相対方位により規定されるものとする。時々刻々と変化する状況に合わせて各 AGF はこの最適化問題を解き続ける必要があるため、C/GMRES を用いて処理速度を向上させた。

3.4 提案手法の全体像

図6に、以上をまとめた経路計画アルゴリズムを示す。リーダーは目的位置が倉庫管理システ

ムから与えられると、その場所までの最短経路をグラフ探索で求め、その経路上の次のノード位置までの詳細経路を C/GMRES を用いて求める。リーダーは、倉庫管理システムもしくは他のチームのリーダーに自分のチームが占有するノード・エッジの情報を送信することで、他のチームとの接触を回避する。一方、フォロワーはリーダーが作成した経路を受信すると、その情報に基づいて全てのフォロワーが同時に自身の経路を計画し、予め決められたフォーメーションとリーダーの位置に基づいて、経路上の目標位置姿勢を設定する。そしてビークル同士の衝突を回避しつつ、目標位置姿勢へ到達できる速度・操舵指令を求める。

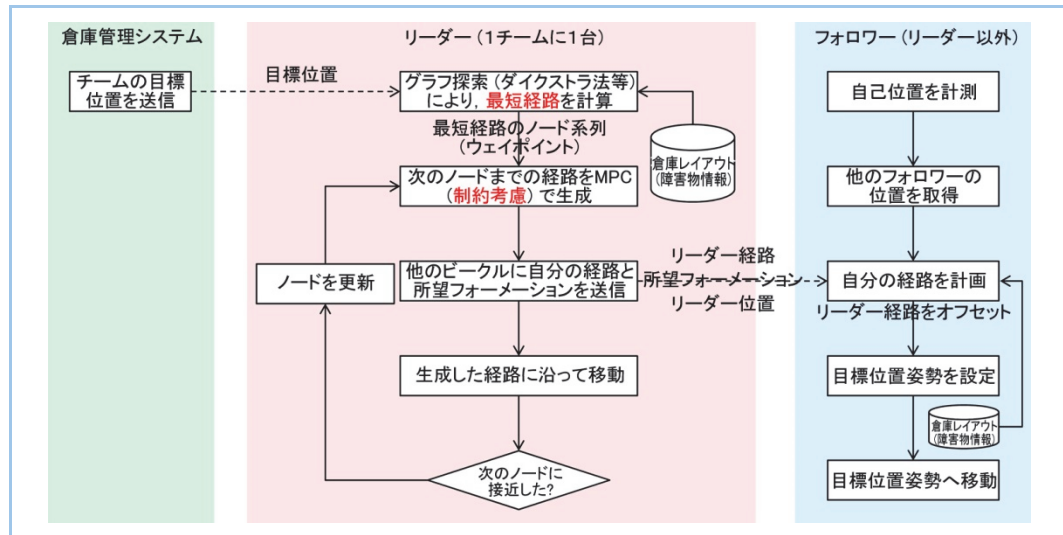


図6 提案法の処理フロー

4. ROS/Gazebo を用いたシミュレーション結果

3章で述べた群知能技術を ROS (Robot Operating System)⁽²⁾で実装し、Gazebo⁽³⁾を用いたシミュレーションで開発技術の有効性を評価した。表2にシミュレーション条件を示す。1チームのビークル (AGF) 台数・チーム数は、十数台規模の AGF が稼働する将来のスマートシステムを想定し、それぞれ5台・3チーム (計 15 台) に設定した。それらのチームをそれぞれ A・B・C と呼称する。速度上限・舵角上限・ホイールベース・車幅はいずれも実機と同一値とし、これらのパラメータを持つ AGF モデルを Gazebo で作製した。実機と同じ環境を模擬するため、各 AGF に ROS ノード (制御装置) を一つ割当て、その中で経路計画を処理させることで、各実機への組込みを表現した。

表2 シミュレーション条件

項目	値	単位
1チームのビークル台数	5	台
チーム数	3	チーム
AGF のホイールベース	1.8	m
AGF の車幅	1.0	m
チーム内フォーメーション	隊列	—

図7に、シミュレーション結果のスナップショットを示す。図中の Goal A, B, C はチーム A, B, C のリーダーの目的位置を表す (リーダーは先頭の AGF)。図中の (a) は初期状態を表している。次に、(b) ではチーム B と C が接近しているが、互いに衝突しない状況を示している。(c) では目的位置に向かって Team B が狭隘部へ侵入し、各 AGF は壁や他の AGF に衝突せず走行できている。最後に、(d) で全てのチームが目標位置に到達している。

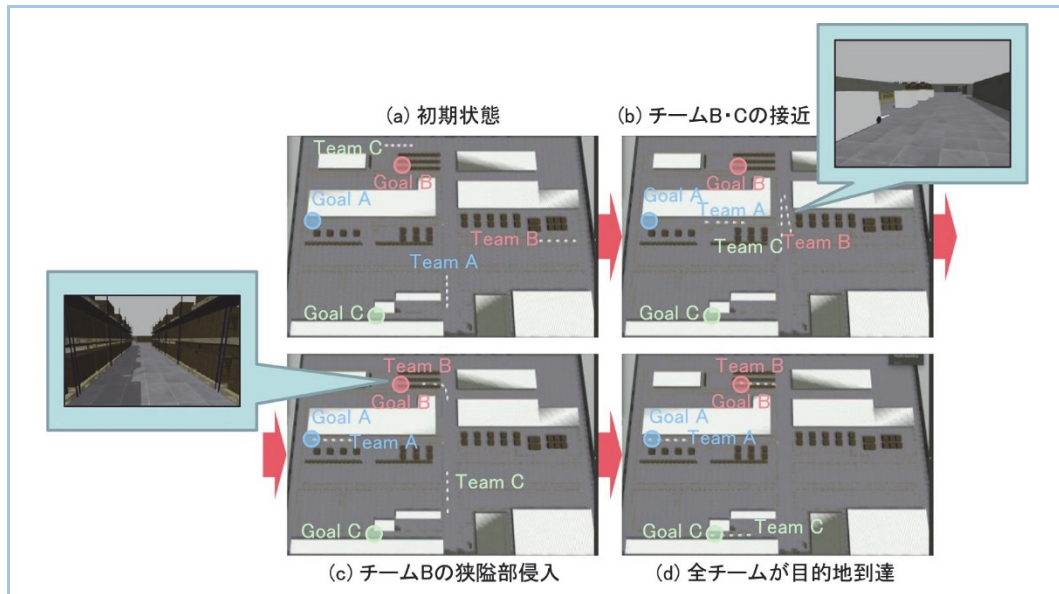


図7 Gazebo シミュレーションのスナップショット

5. まとめ

無人ビークル製品を対象とした、群知能技術を開発した。本技術は、リーダーが存在することが前提であり、リーダーは C/GMRES を用いて高速にモデル予測経路計画を実行することで、チーム(リーダー1台+フォロワー4台)全体の経路を高速に逐次更新するものである。モデル予測制御に基づく経路計画のみでは、入り組んだ区画を走行する問題は解けないため、グラフ探索と組み合わせることで、最短経路選択・壁との衝突及びビークル間の衝突回避を両立した。フォロワーはリーダーが生成した経路を元に、所望のフォーメーションに応じてオフセットさせた経路を生成することで、フォーメーション維持・衝突回避・スムーズな旋回を実現した。本技術の有効性を ROS/Gazebo シミュレーションにより評価した結果、遠方の目的位置まで各チームにフォーメーションを維持させたまま到達できることを確認した。今後、スマートファクトリーの実現に向け、群知能技術の実用化のための開発を進めていく。

参考文献

- (1) 大塚敏之ほか, 実時間最適化による制御の実応用, コロナ社, (2015)
- (2) ROS.org Powering the world's robots
<https://www.ros.org/>
- (2) Gazebo
<http://gazebo-sim.org/>