

AGVの自動走行のための直接接触を用いた地図修正手法の開発と評価

Development and Evaluation of a Map Correction Method for AGV Autonomous Navigation Using Direct Contact

○学 許 鵬飛 (信州大) 正 小林 拓也 (信州大) 正 河村 隆 (信州大)

Pengfei XU, Shinshu University, 20f2019a@shinshu-u.ac.jp

Takuya KOBAYASHI, Shinshu University, 22hs104c@shinshu-u.ac.jp

Takashi KAWAMURA, Shinshu University, kawamura@shinshu-u.ac.jp

In response to the critical labor shortage exacerbated by the decreasing number of agricultural workers and their aging demographic in Japan, this study introduces a novel map correction methodology to enhance the efficiency and reliability of AGV autonomous navigation within outdoor agricultural settings, particularly in mulberry fields. Leveraging the capabilities of Inertial Measurement Unit (IMU) for obstacle detection, the proposed method distinguishes traversable obstacles, such as weeds, by analyzing the direct contact-induced acceleration data obtained during the AGV's navigation, thereby facilitating an efficient path planning and execution process. Through comprehensive experimentation in both simulated and real-world environments, the research assesses the method's ability to identify obstacles effectively and update the SLAM (Simultaneous Localization and Mapping) generated maps accordingly. The findings demonstrate significant improvements in navigation performance, with particular emphasis on enhancing the accuracy of route planning in complex terrains and unstructured outdoor agricultural landscapes.

Key Words: Agricultural Robots, Simultaneous Localization and Mapping, Inertial Measurement Unit

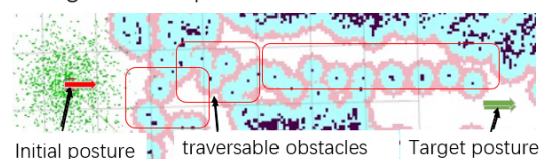
1 緒言

日本の農業では、農業従事者の減少と高齢化の進行により、労働力不足が深刻な問題となっている [1]。平成 27 年から令和 5 年にかけて、基幹的農業従事者は 175.7 万人から 116.4 万人へと減少した。特に 65 歳以上の従事者の割合が高く、平均年齢も上昇している [2]。この問題に対応するため、日本政府は農業現場における農業データの利活用の推進を支持している [3]。例えば、農業現場における土壌データ、気象データなどの農業データを活用することで、農作業の効率化やコストの削減を実現することができる。そこで、我々は桑畑での適用を例として、複数の AGV (Automatic Guided Vehicle) を使用して圃場を巡回し、桑の生育状態を自動で観察するシステムの構築を目指す。桑畑では、AGV が走破可能な雑草が存在する。しかし自動走行のために 2D-LiDAR (2D Laser Imaging and Detection Ranging) を用いて地図を作成する際、走破可能な雑草であっても走破不可能な障害物として記録されてしまう。そのため、本来走行可能な箇所を避けた遠回りな経路が生成され、非効率な走行となる可能性がある。

この問題を解明し、対策を検討するために、信州大学繊維学部附属農場の桑畑で AGV を使用した SLAM およびナビゲーションの実験を実施した。桑畑での AGV 実験結果では、図 1 に示すように、SLAM によって雑草などと思われる走破可能な障害物が地図に記録された。本来これらの障害物は実際の走行において、無視することができる。図 2 に示すように、実験的に手動で障害物を除去した場合、より効率的なナビゲーションをできるようになった。実際の運用では、走破可能な障害物を地図から取り除くことで、AGV の走行効率が向上することが示された。

本研究では、桑畑での AGV の効率的な自動走行を目指し、IMU を活用した障害物識別手法を提案する。具体的には、AGV が桑畑で走行する際、進行方向の雑草との直接接触を通じて得られる IMU の加速度データを分析し、加速度の絶対値が閾値以上の場合は進行方向に障害物が存在すると判断し、閾値未満の場合は障害物が存在しないと判断する。この手法を用いることで、SLAM で生成された地図の正確性を向上させ、桑畑をはじめとする屋外不整地での安定した自動走行を実現することが本研究の最終目的である。

Navigation2 map with traversable obstacles



Navigation route generated from Navigation2

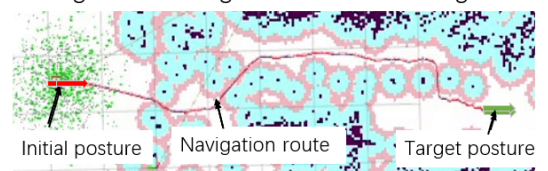
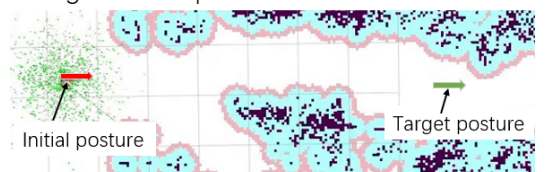


Fig.1 Map with traversable obstacles and the route generated by Navigation2

Navigation2 map with obstacles removed



Navigation route generated from Navigation2

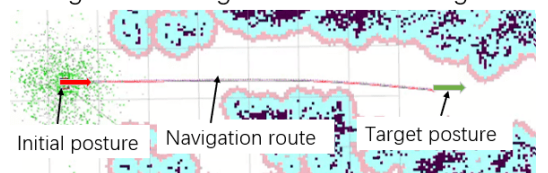


Fig.2 Map without traversable obstacles and the route generated by Navigation2

2 AGV 自動走行のための障害物識別手法の提案

AGV の自動走行において、走破可能な障害物を処理するため、IMU を活用した障害物識別手法を提案する。

2.1 関連研究

AGV の自動走行における障害物の識別に関する既存研究では、松村ら [4] は画像認識と走行経験を組み合わせた走破可能な障害物を認識する手法を提案した。しかし、この手法は画像処理に依存しており、明るさや桑の木の枝などによる遮蔽の影響が考えられる。桑畑のような環境では適応が困難である。そこで本研究では、AGV が障害物と直接接触した際の IMU 情報に基づき、障害物の識別手法を提案する。

2.2 AGV 定速移動中の加速度データの収集と障害物の識別

AGV が SLAM を行う際、AGV の進行方向についての加速度を IMU を用いて計測する。今回の実験では、IMU の X 軸の正方向を AGV の前進方向とした。AGV が走破可能な障害物上の走行を試みる場合には、AGV の進行方向の加速度の変動は小さい。一方、走破不可能な障害物の場合には、AGV が急に停止するため、AGV の進行方向の加速度が大きく変動するという特徴がある。よって、障害物と認識された物について AGV が走破可能か否かは、AGV の加速度の変化量に注目して判断することができる。この特徴を利用し、AGV が障害物と直接接触した際に IMU から取得できる加速度データを利用した障害物の識別手法を提案する。

本手法では、一定速度で走行する AGV が障害物上の走行を試みた際の加速度の値に対し、閾値 δ を設定する。 δ の値については、実際の運用環境や AGV の性能に応じて事前に決定する。加速度の絶対値 $|a|$ が閾値以上 ($|a| \geq \delta$) である場合は、進行方向に障害物が存在すると判断し、SLAM 地図上に障害物のマークを追加する。 $|a|$ が閾値未満 ($|a| < \delta$) である場合は障害物ではないと判断し、SLAM 地図からその情報を削除する。提案手法の全体的なプロセスをアルゴリズム 1 に示す。このプロセスにより、AGV が圃場での SLAM を行う際に走破可能な障害物を効果的に除去することができる。

2.3 SLAM 地図の処理

提案手法では、障害物識別後に SLAM による作成した地図の更新が必要である。SLAM によって作成された地図は Grid Map[5] である。Grid Map はロボットのナビゲーションや環境認識に広く使用されるデータ構造の一つであり、このマップは、環境を一定の大きさの格子に分割し、各格子が特定の状態（例えば障害物の有無）を表す。障害物の有無に応じ、Grid Map 上の対応する Grid の値を変更することで、AGV のナビゲーション精度を向上させる。

障害物が存在する場合、その位置の Grid の値を 100（障害物あり）に変更し、障害物が存在しない場合は 0（障害物なし）に変更する。Grid Map 上の座標は一次元配列で管理されており、実際の地図上の二次元座標 (x, y) を Grid Map の一次元配列上のインデックスに変換する必要がある。

座標 (x, y) から Grid Map 上の Index を求める式は式 1 の通りである。

$$\text{Index} = x \times \text{map.info.width} + y \quad (1)$$

ここで、 map.info.width は Grid Map の幅（X 軸方向の Grid 数）である。この変換により、任意の実際の座標を Grid map の対応する位置にマッピングすることができる。

アルゴリズム 2 は障害物情報の更新プロセスを示す。このプロセスを通じ、障害物の有無に基づいて SLAM 地図上の情報を適切に更新することができ、AGV はより正確に環境を認識し、安全にナビゲートすることが可能になる。

Algorithm 1 Obstacle Detection and Map Update

```
1: procedure DETECTANDUPDATE  
   (map, imu_data, threshold)  
2:   if  $|\text{imu\_data.x\_acceleration}| \geq \text{threshold}$  then  
3:     Identify as obstacle  
4:     Add obstacle mark on the map  
5:   else  
6:     Identify as non-obstacle  
7:     Remove obstacle information from the map  
8:   end if  
9:   return Updated map  
10: end procedure
```

Algorithm 2 Update Obstacle Information on Map

```
1: procedure UPDATEOBSTACLE(map, x, y, obstacle)  
2:    $\text{index} \leftarrow x \times \text{map.info.width} + y$   
3:   if obstacle then  
4:      $\text{map.data}[\text{index}] \leftarrow 100$   $\triangleright$  add obstacles  
5:   else  
6:      $\text{map.data}[\text{index}] \leftarrow 0$   $\triangleright$  clear obstacles  
7:   end if  
8: end procedure
```

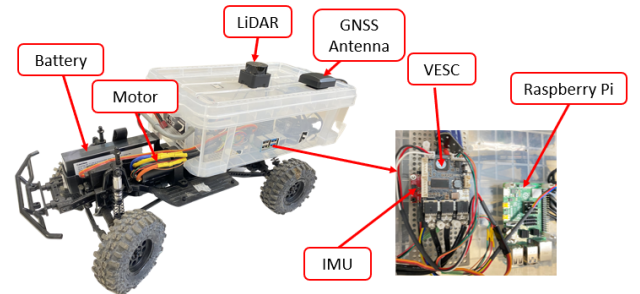


Fig.3 The hardware configuration of the AGV.

3 提案手法を検証するための実験

提案手法の有効性を確認するために、シミュレーション環境として Gazebo classic を使用し、実験を行った。また、実環境では、図 6a に示す実験環境で実験を行った。

3.1 実験に用いる AGV のハードウェア

本研究で使用する AGV のハードウェア構成を図 3 に示す。制御ユニットには、Gmade 社の GS02 をベースに、Raspberry Pi 4B を中心とするシステムが組み込まれている。また、駆動系には BLDC (Brushless DC) モータとして TAMIYA 社の TBLM-02S を搭載している。BLDC モータの制御には FLIPSKY 社の FSESC 4.12 を使用する。センサ系には、2D-LiDAR として YDLIDAR 社の T-mini Pro、IMU として Qwiic-ICM-20948 9DoF を使用し、GNSS(Global Navigation Satellite System) モジュールとしては Ublox 社の ZED-F9P を採用している。2D-LiDAR は環境の地図作成に使用される。IMU は AGV の傾きや加速度を検知し、走行中の姿勢を推定する。GNSS は屋外での位置情報を提供する。

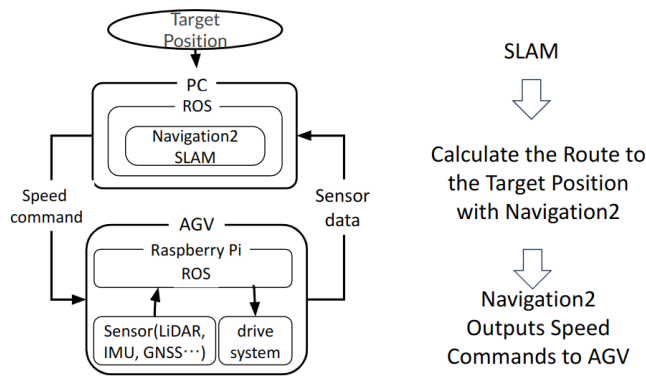


Fig.4 System Configuration

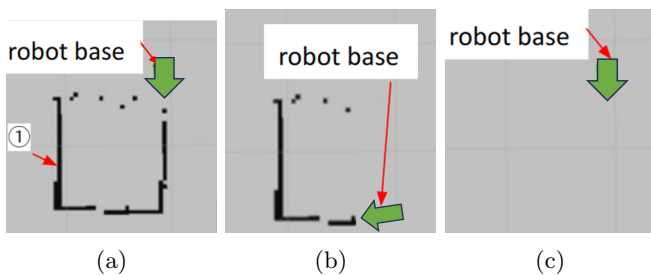


Fig.5 Validation of the proposed method in a simulation environment: (a) shows the initial SLAM map with traversable obstacles erroneously marked, (b) shows the process of removing such obstacles from the map, and (c) shows the final map with traversable obstacles removed, demonstrating the method's effectiveness.

3.2 自動走行システムの構成

自動走行システムの概要を図4に示す。ソフトウェア部分において ROS 2 (Robot Operating System 2[6]) をシステムの基盤とする。SLAM Toolbox[7] を本実験での SLAM アルゴリズムとして選定した。自動走行のためのナビゲーションと経路計画のためのアプリケーションには Navigation2[8] を利用した。AGV 内部の Raspberry Pi 4B は無線 LAN を介して ROS 2 で規格化されたデータ形式でセンサデータを PC に送信する。

AGV の自動走行は地図作成 (SLAM), 地図を用いた経路生成, 経路への追従制御という 3 つの要素で考えられる。PC で Raspberry Pi 4B からのセンサデータを処理し, 自己位置推定および桑畑の地図作成を行う。Navigation2 は目標位置の情報を受け取り, SLAM で作成した地図と AGV の位置姿勢情報を使用し, 経路計算と経路追従のための AGV への速度指令の出力を行う。

3.3 シミュレーション環境における提案手法の検証

提案手法の有効性を確認するため, シミュレーション環境として, Gazebo を用いて, 実験を行った。IMU で取得するデータの閾値を 0.7 m/s^2 と設定し, AGV は 0.2 m/s の速度で走行しながら地図を作成する。図 5a に示されているのは SLAM による作成した初期地図である。図中の黒い標識が走行可能な障害物を示し, 灰色の領域が空白地域である。図 5a の①に示すように, 実際には走破可能である領域は障害物として識別されたことが明らかになった。提案手法を用いて, 図 5b に示すように, その領域の障害物情報を地図から除去された。AGV がこれらの領域を走行した結果, 図 5c に示すように, 障害物標識が地図から除去された地図が生成された。

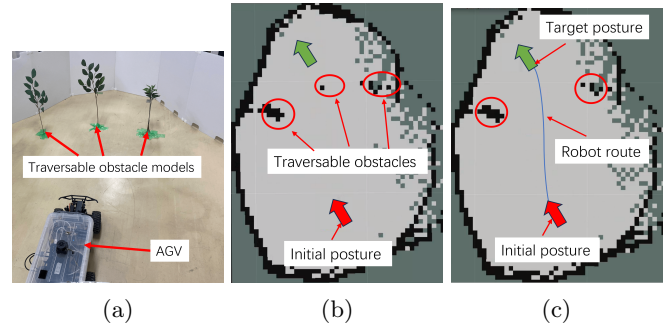


Fig.6 Comparative views of the field and SLAM maps: (a) the actual field, (b) initial mapping with obstacles, (c) Subsequent to obstacle removal.

3.4 実環境における提案手法の検証

実環境における実験では, 図 6a に示する環境で実験を行った。加速度の閾値を 0.7 m/s^2 に設定し, AGV が 0.2 m/s の速度で走行しながら地図を作成する条件のもとで行われた。

障害物識別処理なしのマッピングでは, 走破可能であるにも関わらず障害物として検出された領域が確認された (図 6b)。提案手法を用いることで, AGV がこれらの領域を通過する際に障害物のマッピングが地図から自動的に消去されることが確認された (図 6c)。

この結果から, 提案手法が走破可能な障害物をリアルタイムで識別し, SLAM 地図から除去することの有効性が示された。これにより, AGV は実環境での自動走行の精度を向上させることができる。

4 結言

本研究により提案された手法は, SLAM Toolbox を用いて作成された地図修正することを目的としている。シミュレーション環境および実環境での実験を通じ, 提案手法が走破可能な障害物を効果的に識別し, SLAM 地図から除去できることが示された。この結果, AGV の自動走行におけるナビゲーション性能が向上し, 特に桑畑のような不整地や複雑な環境における経路計画の効率性が改善されることが確認された。

提案手法は, 加速度センサを利用して障害物の有無を判断し, 障害物の識別情報に基づいて地図の更新を行う。このプロセスにより, AGV は雑草などの, 検出されたが走破可能な障害物を無視し, よりスムーズに目的地までの経路を計画および走行することが可能になる。今後の研究では, 走破可能な障害物が存在する桑畑において, 自動走行を行い, 桑畑の生育状態を自動的に観察するシステムを構築する。

参考文献

- [1] 農林水産省, スマート農業の展開について. https://www.soumu.go.jp/main_content/000775128.pdf, (Accessed on 25/1/2024).
- [2] 農林水産省, 農業労働力に関する統計 <https://www.maff.go.jp/j/tokei/sihyo/data/08.html>, (Accessed on 10/3/2024).
- [3] 農林水産省, 農業データの利活用の推進について <https://www.maff.go.jp/j/kanbo/smart/attach/pdf/index-146.pdf> (Accessed on 11/3/2024).
- [4] 松村和紀, 小林祐一. 画像認識と走行経験に基づく車両 型自律移動ロボットのための走破可能な障害物の認識. 日本機械学会ロボティクス・メカトロニクス講演会講演論文集, 2015(1): ROMBUNNO.1P1-K04, 2015.
- [5] P. Fankhauser and M. Hutter, "A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation", in Robot Operating System (ROS) – The Complete Reference (Volume 1), A. Koubaa (Ed.), Springer, 2016.
- [6] Steven Macenski, Tully Foote, Brian Gerkey, Chris Lalancette, William Woodall. Robot Operating System 2: Design, architecture, and uses in the wild. *Science Robotics*, 7(66): eabm6074, 2022,

- [7] Steve Macenski and Ivona Jambrecic. SLAM Toolbox: SLAM for the dynamic world. *Journal of Open Source Software*, 6(61): 2783, 2021.
- [8] Macenski, Steven and Martin, Francisco and White, Ruffin and Ginés Clavero, Jonatan. The Marathon 2: A Navigation System. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.