

インホイール駆動ユニットによる移動ロボットの開発と Cartographer と Autoware を用いた自律走行

原 祥堯 ^{†1}, 西村 健志 ^{†1}, 入江 清 ^{†1}, 吉田 智章 ^{†1}, 大和 秀彰 ^{†1}, 友納 正裕 ^{†1}

Development of an Mobile Robot Using In-Wheel Drive Units and Autonomous Navigation with Cartographer and Autoware

*Yoshitaka HARA ^{‡1}, Takeshi NISHIMURA ^{‡1}, Kiyoshi IRIE ^{‡1},
Tomoaki YOSHIDA ^{‡1}, Hideaki YAMATO ^{‡1}, Masahiro TOMONO ^{‡1}

千葉工業大学 fuRo アウトドア部 III

Abstract— We are developing a mobile robot capable of traveling on outdoor 3D terrain. Our original in-wheel drive units are used to realize a compact, lightweight, and high performance system. The robot performs environment recognition and behavior planning in 3D. We use Cartographer for Graph-based SLAM and Autoware for navigation. In Autoware, NDT Scan Matching and State Lattice Planner are adopted for localization and path planning. The robot achieved complete run in Tsukuba Challenge 2018.

Keywords: Outdoor 3D Terrain, Autonomous Navigation, Tsukuba Challenge

1. 緒言

筆者らは、移動ロボットの基盤技術を研究開発すると共に、各種のロボットシステムへの統合を行っている。本稿では、これらの技術の実証実験の場として取り組んでいる、つくばチャレンジ 2018 での自律移動ロボットの開発事例について報告する。

従来、車輪型移動ロボットの多くが水平面での走行を仮定し、2次元平面で地図構築や経路計画を行っていた。しかし特に屋外の走行環境には、坂(スロープ)、段差、立体交差、螺旋通路など、水平面でない3次元地形もある。本開発では、屋外の3次元地形での自律走行を目標としている。

ハードウェアに関しては、独自開発のインホイール型の駆動ユニットを用いて、小型軽量かつ高出力を実現した。ソフトウェアに関しては、屋外地形を走行するために、3次元での環境認識と3次元での行動計画を行った。

今回の開発では、State-of-the-Art (SOTA) の OSS をいくつか試用し、要求仕様を満たしたものを採用してシステムを構築した。グラフベース SLAM に Cartographer を、自己位置推定と経路計画に Autoware を用い、それぞれ必要な機能を独自に追加している。

2. ハードウェア

2.1 車体と搭載デバイス

Fig. 1 に、開発した自律移動ロボット Puffin の外観を示す。搭載しているセンサは、3D-LIDAR (Velodyne VLP-16)、距離画像カメラ (Intel RealSense D435)、IMU (Xsens MTi-3)、モータエンコーダ (後述のロータ角センサ) である。距離画像カメラは、今回は実験用のデータ取得のみに使用した。

車体は、アルミフレームによって拡張が容易な構造としつつ、樹脂カバーを用いて防水性能を確保した。後輪にサスペンション付きキャスターを用いると共に、前輪の空気圧調整により、凹凸の大きな未舗装路での走行性能を確保している。

電源として、リチウムイオンバッテリ (IDX E-HL9S、14.4

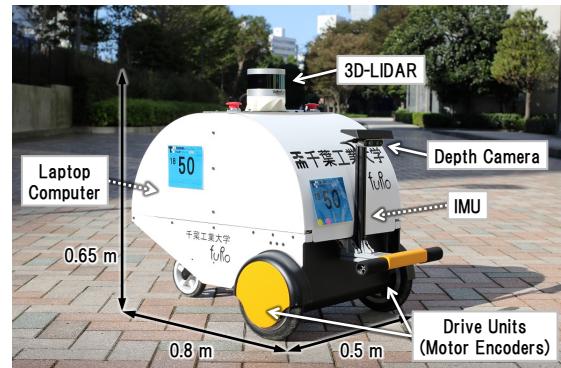


Fig. 1 自律移動ロボット Puffin



Fig. 2 駆動ユニット (モータ、減速機)

V、87 Wh) を 2 直列 2 並列で搭載している。将来的には組み込み PC への電源供給も想定した大容量であり、駆動系とセンサ系だけならば、バッテリ交換なしで一日中走行できる。

コンピュータは、実験の利便性からノート PC (Lenovo ThinkPad P51) を使用している。バッテリ駆動時間は、処理負荷や画面輝度によるが 3 時間程度であり、一日あたり 1~2 回のバッテリ交換を行っている。

2.2 駆動ユニット

車輪の駆動には、千葉工大 fuRo にて開発を進めているインホイール型の駆動ユニットを使用した。駆動ユニットの外観を Fig. 2 に、仕様を Table 1 に示す。

駆動ユニットは、モータ (ステータ・ロータ)、ロータ角セ

^{†1} 千葉工業大学 未来ロボット技術研究センター (fuRo)

^{‡1} Future Robotics Technology Center, Chiba Institute of Technology

Table 1 駆動ユニットの仕様

全体サイズ	$\phi 70 \text{ mm}$ 、D 60 mm 、940 g
モータ	3相ブラシレス（5極対）
ステータ・ロータ	$\phi 60 \text{ mm}$ 、積厚 20 mm
熱定格、トルク	400 W、 $0.92 \text{ Nm}/10\text{Arms}$
ロータ角センサ	磁気ディスク & ホール IC
減速機	サイクロイド機構、薄型1段構成
速比	1/15（最大効率 85% 以上）
出力部	フランジ式クロスローラ軸受

ンサ、減速機から構成される。いずれの構成要素においても専用設計を行っており、既製品の組み合わせでは難しい小型軽量化と機能最適化を追求している。

各モータは、連続電流 15 Arms、最大電流 30 Arms の駆動回路により、正弦波駆動している。ノート PC で車体速度を統括し、CANopen の通信により左右輪の各々で速度制御する。

3. ソフトウェア

3.1 グラフベース SLAM

事前に手動走行によってセンサデータを記録し、オフラインで地図構築（SLAM）を行って 3 次元地図をつくる。グラフベース SLAM として、Cartographer [1] を用いた。文献 [1] では 2 次元の手法が報告されているが、OSS [2] としては 3 次元版も実装されている。Cartographer は、スキャンマッチング（フロントエンド）とポーズ調整（バックエンド）による、グリッドベースの手法である。

Fig. 3 に、つくばチャレンジ 2018 環境で構築した 3 次元地図を平面に投影したものを見ます。走行距離 5 km 以上、約 1 時間半のデータから、経路のループを閉じた整合性のある地図を構築できています。ただしこの地図では、地面の占有空間（黒色）がほとんど消えてしまっています、また自由空間（白色）もほぼ残っていない。

このような歪みのない地図を構築するためには、内部のアルゴリズムを理解した上でパラメータ調整が必要でした。特に、Cartographer の逐次 SLAM は性能が不充分であるためオドメトリの寄与を大きくすることと、広大な環境に適応するためループ検出の対象領域（部分地図）と探索範囲を広げることなどが重要だった。

現状では、SLAM の結果をオペレータが確認して成否を判断し、自律走行に使用する地図を決定している。人手による地図の修正は行っていない。

なお、文献 [1] ではリアルタイム処理が可能と報告されているが、これは 2 次元の場合に限定される。3 次元ではリアルタイム処理は難しく、特にポーズグラフが大きくなると、実時間の数倍以上が掛かる。約 1 時間半の走行データに対して、パラメータにもよるが、オフライン処理に 3~24 時間程度を要した。

Cartographer 本体は 3 次元地図を出力せず（可視化や 2 次元投影は可能）、また構築した地図には通行人や他のロボットなどの移動物体も含まれてしまう。そこで本稿では、ポーズ調整を行ったポーズグラフと生のスキャン点群から、ボクセルフィルタによって移動物体を除去して点群地図を構築する。

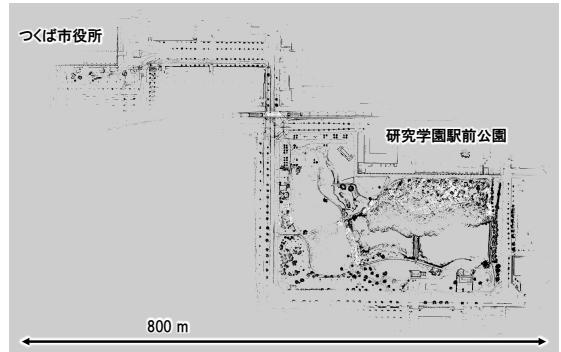


Fig. 3 3 次元 SLAM で構築したつくばチャレンジ 2018 環境の地図

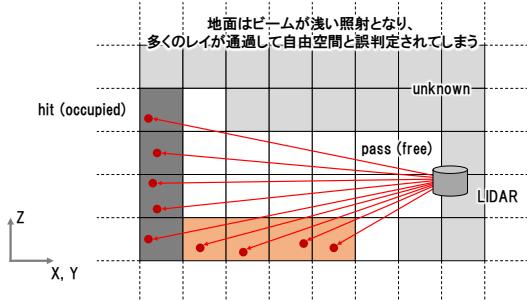


Fig. 4 占有確率を求める逆計測モデルの側面図

3.2 ボクセルフィルタによる移動物体除去

ポーズグラフ（走行軌跡）に対してスキャン点群を配置することで、点群地図を構築する。この際に、点群地図に移動物体が残ると自己位置推定や経路計画の邪魔になるため、除去できることが望ましい。よって、各ボクセルに対してレイキャスティングを行い、占有確率に基づくバイナリペイズフィルタによって移動物体を除去する。

ここで、従来のバイナリペイズフィルタによる占有確率の計算では、誤って地面なども消えてしまうという問題が発覚した。Fig. 4 に、占有確率を求める逆計測モデルの側面図を示す。地面のセルは、ビームが浅い照射となってしまうため、ヒット（占有）に対して他の多くのレイが通過してパス（自由）となる。これにより、地面が自由空間と誤判定され、消えてしまう。

これは空間をグリッド分割することに伴う、本質的な課題である。ビームが浅い照射となる物体は静止していても消えやすく、逆に垂直に照射される物体は移動していても残りやすい。3 次元地図での地面に限らず、2 次元地図で壁際を走行した際にも、同様の問題が発生する。

そこで提案手法では、各点の法線方向と入射角を利用して、この問題に対処する。バイナリペイズフィルタで占有確率を求める逆計測モデルにおいて、ビームの入射角を考慮することで、地面などの静止物体の点群を正しく残しつつ移動物体を適切に除去できる。この手法の詳細は、別論文（ロボティクスシンポジウム 2019）の一部として報告予定である。

Fig. 5 に示す生の点群地図には、通行人や他のロボットなどの移動物体の軌跡が線となって残ると共に、空中には細かなノイズの点群が含まれている。Fig. 6 や Fig. 7 に示す従来手法では、移動物体は消えているが、地面も消えてしまっている。一方で Fig. 8 に示す提案手法では、移動物体は消えつつ、地面が残っていることが分かる。

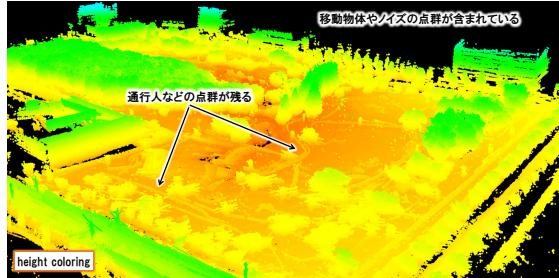


Fig. 5 3 次元グラフベース SLAM で構築した生の点群地図

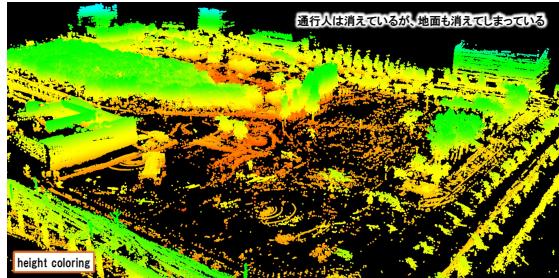


Fig. 6 統計占有モデル（従来手法）によるボクセルフィルタ

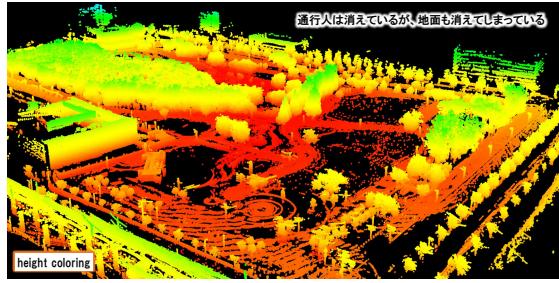


Fig. 7 ベイズ占有モデル（従来手法）によるボクセルフィルタ

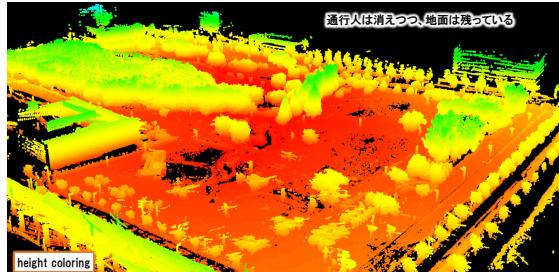


Fig. 8 ビーム入射角を考慮したベイズ占有モデルによるボクセルフィルタ

3.3 自己位置推定と経路計画

自律走行機能のベースとして、Autoware [3] を用いた。Autoware は、ROS 上に各種の自律走行ソフトを実装したパッケージ群である。各種の自己位置推定や経路計画の他、物体認識などの機能を持つ。使用する際のパッケージの組み合わせ方は、色々とあり得る。本稿では、lidar_localizer と lattice_planner を使用し、諸々の不具合修正や機能追加を行ってシステムに統合した。

自己位置推定には、lidar_localizer パッケージの 3 次元 NDT スキャンマッチングを用いた。Fig. 9 に、NDT スキャンマッチングの模式図を示す。NDT は、対応点の最近傍探索が不要で、計測点がある地図のボクセルに対応付けるため、地図が大規模でも処理が高速である。実装としては、PCL の関数を呼び出している。

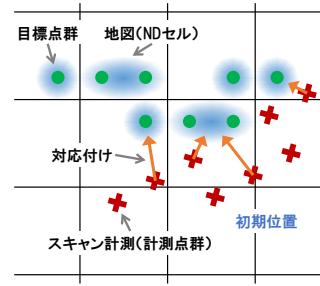


Fig. 9 NDT スキャンマッチングによる自己位置推定

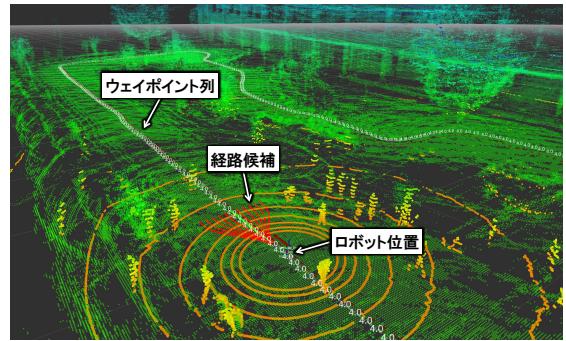


Fig. 10 State Lattice Planner による経路計画

経路計画には、lattice_planner パッケージによる State Lattice Planner を用いた。Fig. 10 に、経路計画の様子を示す。事前に与えられたウェイポイント列に対して、局所的に複数の経路候補を生成している。ウェイポイントは 3 次元だが、経路計画を行う各時刻において、ロボット位置の局所平面に投影することで 2 次元の経路として扱う。また、この State Lattice Planner では、3 次スプライン曲線で経路を生成している。

経路に追従する速度を決定するために、ポテンシャル場を用いるように改良した。ロボット位置中心のポテンシャル場を生成し、周囲の障害物に応じて速度を決定する。これにより、障害物の近くでは減速して走行し、開けた環境では所定の速度でスムーズに走行できる。

3.4 自律走行の全体システム

Fig. 11 に、自律走行システムの構成を示す。プロセス間通信(IPC)には、独自開発の fuRom [4] と ROS [5] を併用した。

走行系の基盤となる車体制御には、差動駆動のキネマティクス(FK, IK)とオドメトリを実装している。オドメトリとして、車体静止時の IMU 自動キャリブレーション機能を持つジャイロ併用車輪オドメトリを開発した。

障害物検出は、3 次元スキャンから地面を除去し、衝突する高さの点群を抽出している。また LIDAR の最短測定距離(0.5 m 程度)未満にある障害物を検出するため、目前の路面検出を行い、路面がない場合も障害物として扱う。ロボットに人などが接近した際の安全確保には、この機能が重要であった。

つくばチャレンジの選択課題であるチェックポイント通過を達成するためにも、走行経路は臨機応変に変更できることが望ましい。今回の開発では、行動計画をシェルスクリプトとして実装し、走行経路の切り替えや一時停止でのオペレータ操作入力に対応した。チェックポイント間のウェイポイント列は、現状ではグラフベース SLAM のポーズグラフを分割して生成している。

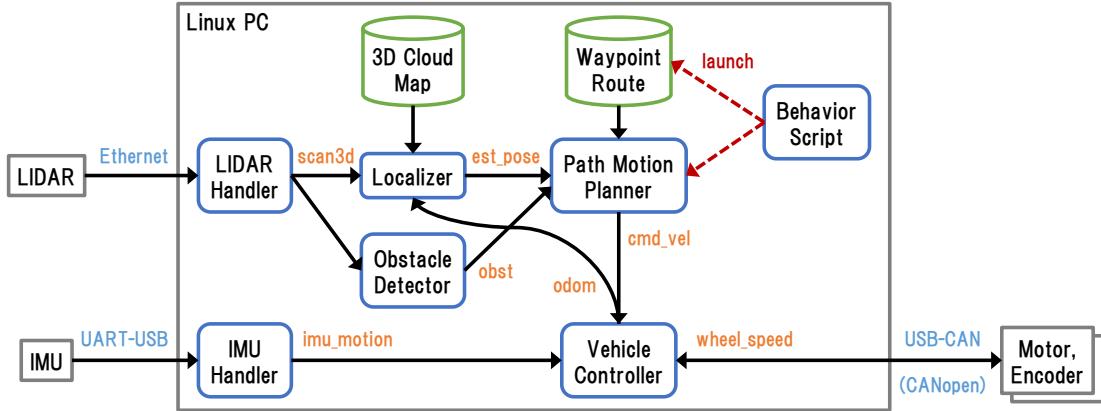


Fig. 11 自律走行システムの構成

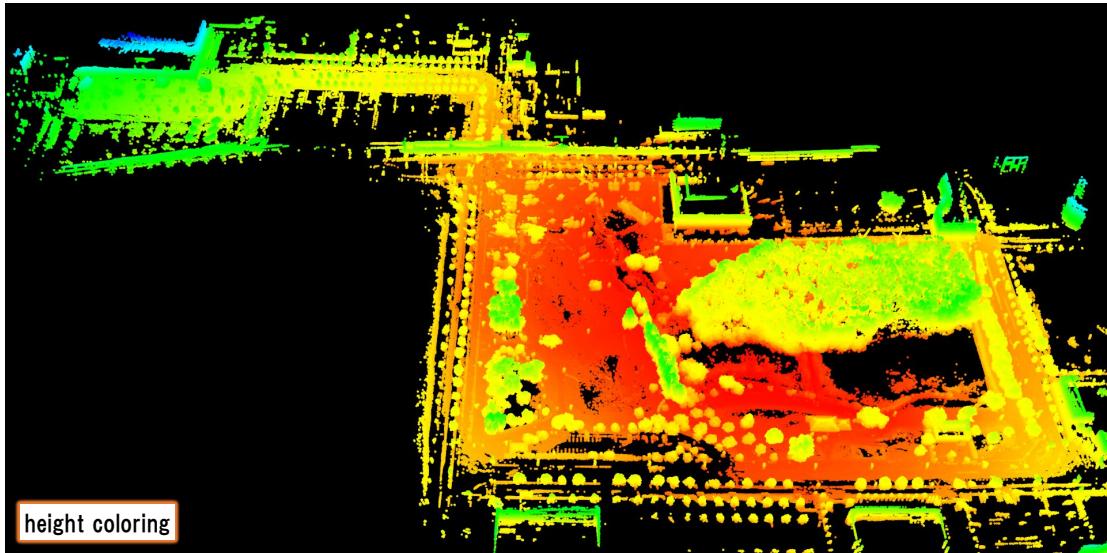


Fig. 12 ビーム入射角を考慮したペイズ占有モデルでの 3 次元点群地図の全体像

4. 実験結果

Fig. 12 に、つくばチャレンジ 2018 環境で構築した 3 次元の点群地図を示す。走行距離 5 km 以上、約 1 時間半のデータに対して、手動での修正なしに経路のループを閉じた地図を構築できている。また、ビームの入射角を考慮したバイナリペイズフィルタにより、通行人や他のロボットなどの移動物体を適切に除去すると共に、地面などの静止物体の点群が正しく残っている。

つくばチャレンジ 2018 本走行では、完走かつチェックポイント通過の選択課題を達成することができた。システムが完成したのが直前だったため、記録走行は 1 回も実施していない。実験としては何度か自律走行を行い、雨天時も含めて、スタートからゴールまで森林内も込みのひと通りの経路を走行できることを確認した。

ただし、ゴール前の広大な広場で自己位置推定が破綻する場合があった。およそ 4 回の走行に 1 回程度、この問題が発生し、自律走行の継続が困難な状況に陥る場合があった。

5. 結言

本稿では、つくばチャレンジ 2018 での自律移動ロボットの開発事例について述べた。ハードウェアに関しては、独自開発のインホイール型の駆動ユニットを用いて、小型軽量かつ

高出力を実現した。ソフトウェアに関しては、屋外地形を行けるために、3 次元での環境認識と 3 次元での行動計画を行った。グラフベース SLAM に Cartographer を、自己位置推定と経路計画に Autoware を用い、諸々の機能を独自に追加してシステム統合した。

今後の課題としては、自己位置推定の安定性の向上がある。広大な広場で自己位置推定が破綻する問題に対処するため、ペイズフィルタによる確率的融合が必要である。

また、現状ではウェイポイント列をポーズグラフから生成しているため、教示経路が目標経路となる。少数の目標地点を与えるだけで自動的に経路を生成する、長距離の 3 次元経路計画ができると良い。これについては、別論文（ロボティクスシンポジア 2019）で報告予定である。

その他、信号認識横断と探索対象発見にも取り組みたい。

参考文献

- [1] Wolfgang Hess, Damon Kohler, Holger Rapp, and Daniel Andor: “Real-Time Loop Closure in 2D LIDAR SLAM”, *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2016.
- [2] Cartographer. <https://github.com/googlecartographer>
- [3] Autoware. <https://github.com/CPFL/Autoware>
- [4] 入江 清: “ROS との相互運用性に配慮した共有メモリによる低遅延プロセス間通信フレームワーク”, 日本ロボット学会学術講演会予稿集, 2017.
- [5] ROS. <http://www.ros.org/>