

# ちびチャレ 2018 成果報告

1 班 (有馬純平, 隼田駿大, 中本英晶, 小林航大)

## 1 緒言

近年, 自動運転車をはじめ, 人間の業務を代替する自律移動ロボットの開発が盛んに行われている. 今回は, それらのロボット技術に触れ, 研究への橋渡しとすべく, 「ちびチャレ」と称される教育研修を行った.

本研修では, 自律移動ロボットの基礎となる 3 つの技術 (自己位置推定, 大域経路計画, 局所経路計画) を学ぶことを目的とした. これらの技術は, ROS の navigation stack<sup>1)</sup> などのオープンソフトウェアを用いて容易に実装可能である. しかし, 本研修ではより理論的なことを理解するとともにコーディング技術を向上させるため, オープンソフトウェアは使用せずにそれらのシステムを作成した. ハードウェアとしては, iRobot 社の Roomba<sup>2)</sup> と北陽電機社の 2D-LiDAR の UTM-30LX<sup>3)</sup> とノート PC を用いた. そして, ROS で実装したシステムを動作させ, 最終的に明治大学生田キャンパス第二校舎 D 館 1F を 1 周させることを目標とした.

## 2 提案手法

本章では, 実装したプログラムのアルゴリズム, および実装に当たって工夫した点について説明する. システムの概要図を Fig. 1 に示す. ここで示す地図情報は, オープンソフトウェアの gmapping<sup>4)</sup> で作成した 2 次元地図である. 本システムでは, 事前情報として地図と経路点を与える.

以下の各節では, 実装した自己位置推定, 大域経路計画, 局所経路計画の 3 つの要素について説明する.

### 2.1 自己位置推定

自己位置推定には, パーティクルで信念を表現する Augmented Monte Carlo Localization<sup>5)</sup> のアルゴリズムを用いた. 今回の実装では, 観測ノイズを 4 種類想定した点と, ランダムパーティクルを追加する点の 2 点を工夫した. まず, センサの計測誤差を, 正しい計測時の小さな計測ノイズ, 地図上に存在しない物体による計測ノイズ, ガラスなどの透過による計測ノイズ, ランダムな計測ノイズの 4 種類について考慮した. この様に 4 種類考慮することによって, ノイズに強い尤度計算とした.

また, 今回走行する D 館 1F は特徴的な構造物が少なく, 位置推定に誤りが生じやすい. 通常の MCL では, パーティクルが 1 箇所にまとまった後, 位置推定に誤りが生じると復帰することができない. そこで, 今回のアルゴリズムではランダムなパーティクルを加えることで, パーティクルが 1 箇所にまとまった後に, 位置推定に誤りが生じてしまったとしても, 位置推定を回復する確率を増やし, ロバスト性を高めた. ランダムパーティクルは Eq. 1 の確率  $p$  で追加する. ここで  $\omega_{slow}$ ,  $\omega_{fast}$  はそれぞれ尤度の短期, 長期の平均値である.

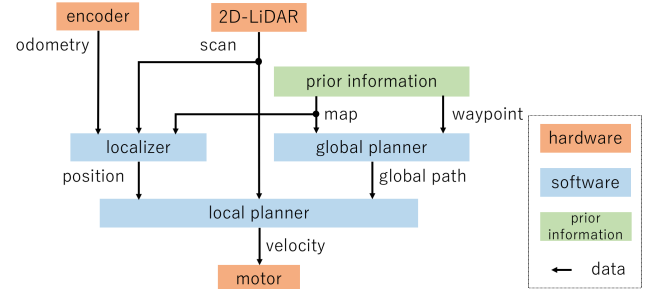


Fig. 1: システム概要

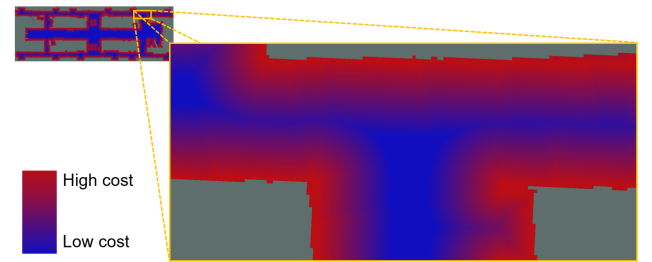


Fig. 2: コストマップ

$$p = \max(0.0, 1.0 - \omega_{fast}/\omega_{slow}) \quad (1)$$

### 2.2 大域経路計画

大域経路計画には, A\*アルゴリズム<sup>6)7)</sup>を用いた. 各セルにおけるノードの向きは 8 方向で区別した. ベースコストは, セルの区分線と平行な向きのノードで 1 とし, 斜め向きのノードで  $\sqrt{2}$  とした. ヒューリスティック値  $h$  は Eq. 2 に示すように現在位置  $(x, y)$  とゴール位置  $(gx, gy)$  との距離とした.

$$h = \sqrt{(gx - x)^2 + (gy - y)^2} \quad (2)$$

また, 壁付近のコストマップの値を上げることで, 最終的なパスが狭い通路や壁などの障害物の近くを通りにくく, 通路の中央を通るようにした. コストマップの値は, 壁からの距離に応じて直線的に減少するように設定した. 得られたコストマップを Fig. 2 に示す. ここでは, コストは赤, 青の順で大きい値を示す.

### 2.3 局所経路計画

局所経路計画には Dynamic Window Approach<sup>8)9)</sup>を用いる. Dynamic Window 内の速度  $v$ , 角速度  $\omega$  をサンプリングし経路を予測する. 予測経路に Eq. 3 の評価関数を適応してコストを計算し, コストが最少となる速度  $v$ , 角速度  $\omega$  を出力値とする.

$$G(v, \omega) = \alpha * dis(v, \omega) + \beta * obs(v, \omega) + \gamma * vel(v, \omega) \quad (3)$$

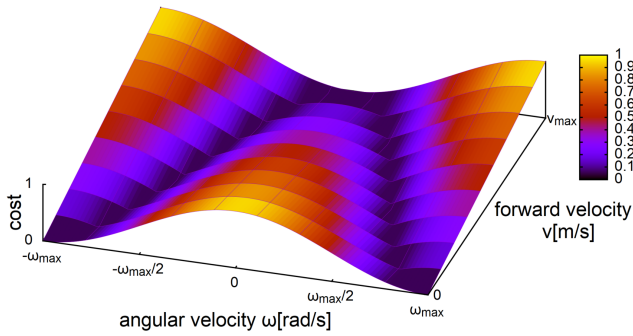


Fig. 3: velocity 関数におけるコストの算出

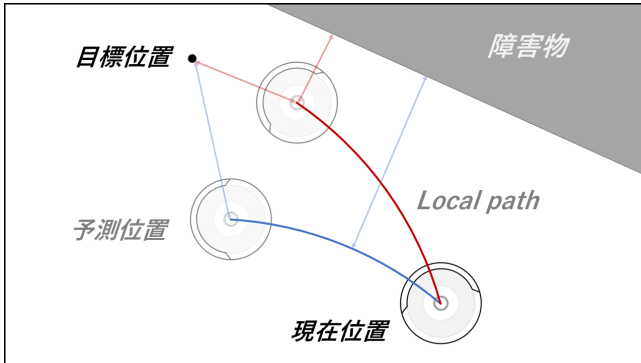


Fig. 4: 局所経路計画における評価

ここで,  $dis(v, \omega)$  は予測位置から目標地点までの距離によるコスト,  $obs(v, \omega)$  は予測経路から障害物までの最短距離によるコスト,  $vel(v, \omega)$  はサンプリングした速度  $v$  と角速度  $\omega$  の関係によるコストを表している. 角速度  $\omega$  の絶対値が大きいとき, 直進速度  $v$  は小さいものが適切と考え, Fig. 3 に示す関係でコストを算出した. サンプリングした速度と適切速度の差をコストとした. Fig. 4 に局所経路計画の評価のイメージを示す.

### 3 実験

今回実装した自己位置推定, 大域経路計画, 局所経路計画の評価を行うために, 以下の二つの実験を行う.

#### 3.1 障害物回避実験

本実験では, 障害物回避性能を評価するため, Fig. 5 に示すような回避実験を行った.  $D$  は障害物間の幅であり,  $d$  は, 大域経路線と障害物間の中心線の幅である.  $D = 0.4, 0.5, 0.6[m]$ ,  $d = 0.0, 0.5, 1.0[m]$  とし, 各条件で回避可能かどうかを 4 回走行させて調べる. その結果, 障害物を通過することのできた割合を Table 1 に示す. 結果から分かるように,  $D = 0.6[m]$ ,  $d = 0.5[m]$  まで確実に突破することができた.  $d = 1.0[m]$  の条件では, 障害物前で停止し回転し続けてしまった. これは, スペースを見つけることができず, ロボットが行き止まりと認識しているためだと考えられる.

Table 1: 障害物回避の結果

障害物回避		スペースの位置 $d[m]$		
結果		0.0	0.5	1.0
障害物間距離 $D[m]$	0.4	50%	0%	0%
	0.5	100%	25%	0%
	0.6	100%	100%	0%

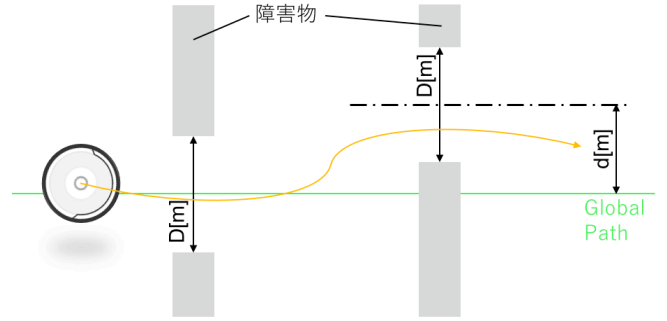


Fig. 5: 障害物回避実験

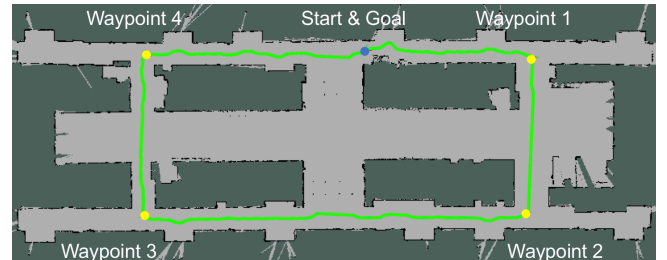


Fig. 6: 大域経路

#### 3.2 周回実験

本実験では, 生田キャンパス D 館 1F を 1 周させる実験を 10 回行う. Fig. 6 にスタート・ゴール地点, および経路点 4 点を与えた時の 1 周の大域経路 (緑線) を示す. 実験を行った結果, 10 回とも完走することができた. 1 周の平均走行時間は 4 分 30 秒だった. 自己位置推定は, 小さな段差などでタイヤが空転する場合を除き, 明らかな誤差が発生することはなかった. また, 自己位置推定に誤差が生じた場合でも短時間で復帰できることが確認できた. さらに, 最初に計算して求めた大域経路に沿って走行することも確認できた.

### 4 結言

本研修では Roomba に 2D-LiDAR を搭載し, 自律移動プログラムを実装した. 実機を用いた実験により実装したシステムが有用であることを確認できた. また本研修を通じて, 自律移動ロボットの基礎となる技術の基礎的な知識とそれらを実装する力を身につけることができ, 今後自律移動ロボットの研究を行なっていくための橋渡しとなった.

### 参考文献

- 1) <http://wiki.ros.org/ja/navigation>
- 2) <https://www.irobot-jp.com>
- 3) <https://www.hokuyo-aut.co.jp/search/single.php?serial=21>
- 4) <http://wiki.ros.org/gmapping>
- 5) Sebastian Trun, Wolfram Burgard, Dieter Fox, "Probabilistic Robotics", MIT Press, vol.1, pp.250-261, 2005
- 6) Peter E. Hart, Nils J. Nilsson, Bertram Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", IEEE Transactions of Systems Science and Cybernetics, 1968.
- 7) <https://classroom.udacity.com/courses/cs373>
- 8) <http://myenigma.hatenablog.com/entry/20140624/1403618922>
- 9) Dieter Fox, Wolfram Burgard, Sebastian Thrun, "The Dynamic Window Approach to Collision Avoidance", IEEE Robotics and Automation Magazine, 1997.