

令和3年度卒業論文
自律走行ロボットにおけるパラメータ調整が
地図作成に及ぼす影響

奈良女子大学 生活環境学部 情報衣環境学科生活情報通信科学コース 4回生
新出研究室 18480293 杉山華那依

令和4年2月

目次

1	はじめに	3
2	関連研究	4
3	使用ロボット	4
3.1	Arno	4
3.2	センサ	4
3.3	ROS	4
4	地図の作成と評価	5
4.1	地図作成	5
4.2	作成された地図の評価	6
5	結果	9
6	まとめ	9
7	謝辞	10

概要

近年、周囲環境の変化に対応しながら目標達成への最適なプラン選択をしてスタートからゴールまでの完走を目指す自律走行ロボットの研究が盛んに行われている。自律走行ロボットとは、与えられた知識や周囲の環境から自ら考えて状況を判断し、出発地点から目的地まで走行するロボットを指す。自律走行ロボットは、自身のタイヤの動き、センサで観測する実環境、走行経路の周辺地図をもとに自律走行を行う [4, 5]。

その中でも、ロボットに搭載されたセンサ情報を利用して走行経路の周辺地図を自ら作成し、その地図をもとに自律走行をするロボットがある。ここで作成された地図はその後の自律走行時に周囲環境と比較する際に利用され、自律走行において重要な役割を果たす。作成された地図の正確性は、センサに設定する各種パラメータに大きく影響を受け、またその正確性は自己位置推定や自律走行の安定性に深く関係する。しかし、パラメータの設定がどのように地図の正確性に影響するかに関する先行研究は現時点では見当たらなかった。本研究では、搭載されたセンサの計測範囲を指定するパラメータを調整して、作成される地図に現れる影響について調査した。

1 はじめに

近年、農業や宅配サービスなど様々な場面において、人間が直接介入せずとも自ら目標達成へのプラン選択をしながら行動ができるロボットが注目されている。このようなロボットは自律ロボットと呼ばれ、人間からの指示がなくても自らの置かれた実世界の環境を認識し、周囲の情報をもとに目標達成へ向けた最適な行動選択を求められる。特にスタートとゴールを指定して、与えられた地図とセンサで得た周囲環境の情報を元に走行を行うロボットは、自律走行ロボットと呼ばれる。

こういった自律走行ロボットの技術の進歩を目的として行われている、「つくばチャレンジ」という技術チャレンジがある [2]。これは 2007 年から毎年実施されており、つくば市内の遊歩道などの市街地で移動ロボットが自律走行するために、大学や企業、個人など 50 以上のチームが参加し、自律走行技術の発展のため技術交流や公開実験を行う場である。このつくばチャレンジと同趣旨の大会を関西地方でも実施しようと企画されたものが、「中之島ロボットチャレンジ」である [1]。中之島ロボットチャレンジは、人々の往来する実環境において自律移動ロボットが問題なく行動できる技術開発の公開実験を大学や企業向けに提供する大会であり、様々な研究開発機関が参加する技術交流の場を設けることで、ロボット開発技術のレベルを向上させることを目的としている。我々は、2021 年夏、秋、冬に開催された中之島ロボットチャレンジに北陽電機と共同のチームで参加した。そこで実験走行を行った結果、いくつかの問題点が確認された。その一つが、自律走行時に使用される地図の作成が正確に行えないと走行が安定して行えないという点である。

自律走行を行うためにロボットが用いる情報は、主に 3 つある [4, 5]。1 つ目はオドメトリと呼ばれる、どれくらい進んだかなどをタイヤから読み取ったもの、2 つ目はロボットに搭載されたセンサで観測する実環境、3 つ目は走行場所周辺の地図である。自律走行に使用される地図は、自律走行の前にあらかじめコースでロボットを走行させ、ロボットに搭載されたセンサで壁や通路を認識して作成される。このとき、センサの誤差や誤認識などにより、作成される地図には不正確さが生じる。また、地図の精度はセンサに設定される各種のパラメータに影響を受ける。ロボットの自律走行では、作成される地図の精度がその後の自律走行の正確性に影響を及ぼすことが我々の実験で確認されている。そこで本研究では、地図作成時に使用されるセンサの計測範囲のパラメータを調整し、作成される地図に現れる影響について調査した。

2 関連研究

地図の精度が自律ロボットの走行に与える影響を調べた研究としては、江口らによる DGPS とスキャンマッチングを組み合わせて占有格子地図を作成することでより安定した自己位置・姿勢を推定する研究 [3] などが存在する。しかし、センサのパラメータ設定が地図に与える影響を調べた研究は現時点では見当たらなかった。

3 使用ロボット

3.1 Arno

本実験で使用するのは、北陽電機開発の自律走行ロボット Arno(図 1) である。本体の前後に、3.2 節で述べる 2 台の測域センサを搭載している。また、制御には PC 上の Ubuntu 18.04.5 LTS を用いている。地図作成や自律走行でない手動での走行の際には、人間がコントローラーを通じて動かすことが可能である。Arno のタイヤは 3 つあり、前方二輪による前輪駆動である。



図 1: Arno(北陽電機)

3.2 センサ

Arno は、二次元測域センサ URM-40LC-EW(北陽電機) を本体の前方中央と右後方の合計 2 台搭載している。これは、最大で距離 40m、270° の範囲を計測可能な屋外用レーザスキャナである [6]。地表より約 43cm の高さに設置されており、そこから扇形にレーザーを放射して周囲の障害物等を検知することができる。2 台がそれぞれ 270° の範囲を計測することで、ロボットの周囲の全範囲を計測することができる(図 2)。これらの計測範囲は、パラメータを調整することで変更可能である。

3.3 ROS

Arno を含む自律走行ロボットは、ROS(Robot Operating System) というロボット開発用のソフトウェアプラットフォームを使用して制御する。ROS は、C++ や Python でプログラムを作成することができる。

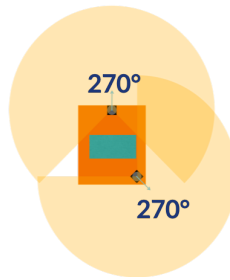


図 2: 最大センサ計測範囲

パッケージという基本単位があり、その中に様々な機能を実装していく。プロセスの実行の最小単位はノードと呼ばれる。このノード同士が相互にやりとりすることで、データの受け渡し等を行うことができる [4, 7]。launch ファイルというファイルを使用すると、ロボットを制御するために必要な複数のノードを一括して起動することもできる。また、param と呼ばれるパラメーターの数値を変更することで、センサから受け取る値の調整が可能となる。

ROS はデフォルトの状態でもいくつかのパッケージを有するため、それらを用いてロボットを制御することも可能である。しかし、ロボットの形状やセンサの種類によっては、自作の独自ノードを使用することも可能である。Arno の制御は、標準のノードおよび北陽電機から提供されるいくつかの独自ノードで行われており、今回の研究にもそれらを使用した。

4 地図の作成と評価

4.1 地図作成

奈良女子大学構内の池を中心としたコース (図 3) をコントローラー制御で走行し、地図を作成した。

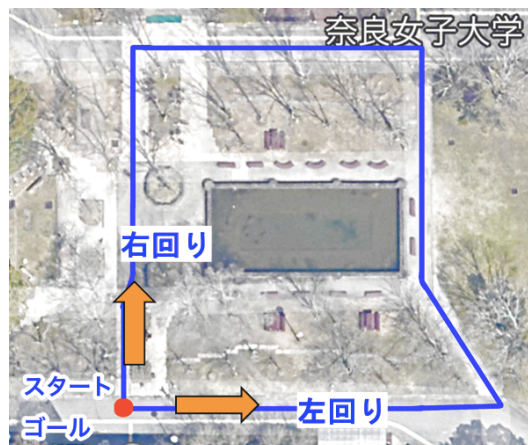


図 3: 地図作成に使用したコース (奈良女子大学構内)(Google map を加工)

Arno には様々なパラメータが存在するが、実際に地図作成に関わっているパラメータのうち特に今回はセンサの計測範囲の角度を決定するパラメータを調整し、各条件ごとに作成された地図の形状について調査し

表 1: センサの観測範囲のパラメータ設定

角度の min 値	角度の max 値	計測範囲
-30°	30°	60°
-60°	60°	120°
-90°	90°	180°
-120°	120°	240°
-135°	135°	270°

た。Arno に搭載されているセンサは前後合わせて 2 台あり、どちらも最大 270° の範囲を計測することができる。この計測範囲のパラメータは max 値と min 値を設定することが可能で、本研究では計測範囲として、min -30° から max 30° まで (計測範囲 60°)、min -60° から max 60° まで (計測範囲 120°)、min -90° から max 90° まで (計測範囲 180°)、min -120° から max 120° まで (計測範囲 240°)、min -135° から max 135° まで (計測範囲 270°) の 5 通りのパラメータ (表 1) を設定し、スタート地点を同じとする右回り・左回りの合計 10 通りの条件を用意した。条件 1 通りごとに 4 回の実験を行い、合計 40 枚の地図を作成した。

4.2 作成された地図の評価

作成された地図の評価は、以下の過程により行った。

4.2.1 地図比較の準備

まず、地図の加工・編集を容易に行うには、扱う地図のサイズを揃えておくことが望ましい。そこで、地図の最大サイズを知るため、読み込まれた画像の縦横の画素数を数え、その中で縦横それぞれの最大値を出力するプログラムを作成した。これを使用して 40 枚の地図の縦横の最大サイズが求めたので、それよりも大きく、今後の比較で扱いやすいサイズとして、縦横それぞれ 1900 ピクセルになるように画像サイズを調整することとした。そのため、PIL(Python Image Library) を用いて画像サイズを一括で 1900 × 1900 に揃え、空白部分には指定した色を補完するプログラムを作成した。また、作成されたばかりの地図のファイル形式は PGM だが、地図の編集にあたって、ペイントソフトを利用する際に扱える PNG などのファイル形式に一括で変換するプログラムも作成した。

4.2.2 節および 4.2.3 節で述べる地図の作成にあたっては、前もってこれらのプログラムを用いて地図のサイズと形式を揃えてから行った。

4.2.2 基準地図の作成

地図の比較の基準とするため、できるだけ正確な地図が必要だと考えた。今回は Google map の航空写真と、実際に現地を確認した地形の特徴などをもとに、比較の基準となる地図 (以降「基準地図」と呼ぶ) を作成した。MediBangPaint Pro、ibisPaint 等のペイントソフトを使用し、航空写真からわかる特徴を線画に書き起こした。航空写真撮影時点と実際に走行した時点で微妙に異なる部分については個別に編集した。また、実際に作成した地図ではセンサの計測範囲の都合上、地図の端に放射状に線が伸びているが、これは地図を作成するごとに変化するため、基準地図では省略した。

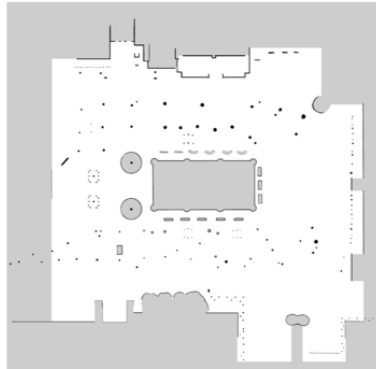


図 4: 基準地図

4.2.3 合成地図の作成

地図の評価は、4.1 節に延べた条件 1 通りごとに 4 枚ずつ作成した地図を重ね合わせて作成した地図 (以降「合成地図」と呼ぶ) に対して行った。

重ね合わせには MediBangPaint Pro を使用した。線画抽出機能で 4 枚の地図画像を読み込み、地図中の奈良女子大学記念館に当たる部分と学術情報センター前の掲示板に当たる部分を目印とし、グリッドに合わせて 4 枚それぞれの角度と位置を目視で揃えた。拡大縮小や縦横比の変更はせず、回転と平行移動のみで変形を行った。また、変形の際に空白になった背景部分には、もとの地図の背景と同じ灰色を補完した。

しかし、線画抽出の際に輝度によって不透明度の変更が行われた地図画像を重ねているため、灰色同士が重なり合っている部分等はもとの地図と比べて大きく灰色の濃度が変わってしまった。そこで、今後の比較を容易にするため、画像の三値化を行うプログラムを作成し、これを用いて画像の編集を行った。OpenCV を用いて、閾値の設定を行い、明度が一定値以下の画素は黒色 (0)、一定値以上の画素は白色 (255)、それ以外の画素はもとの地図で使用されているものと同じ灰色 (205) に置き換えた。今回は 110 と 132 を閾値に設定し、重ね合わせた地図に用いられる色を三値化した。

以上の手順を 10 通りの条件で作られた地図全てに対して行い、10 枚の合成地図を作成した。

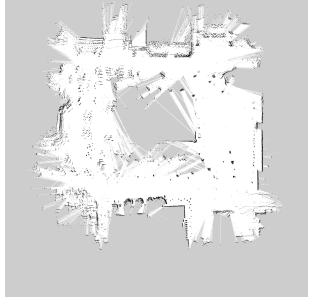
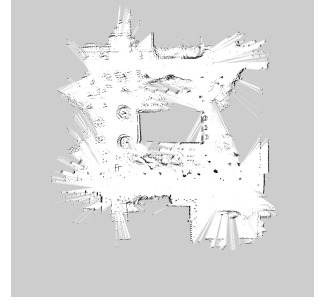
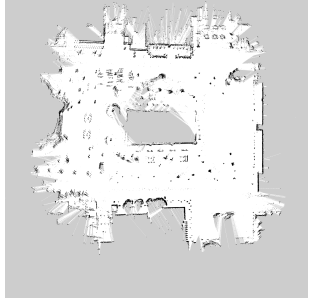
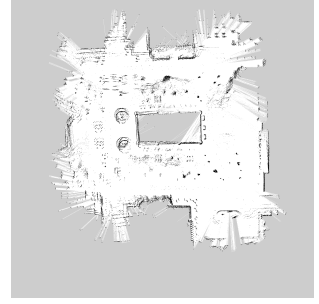
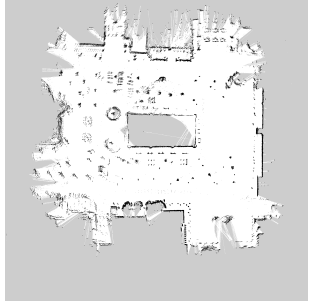
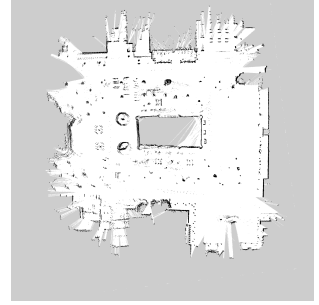
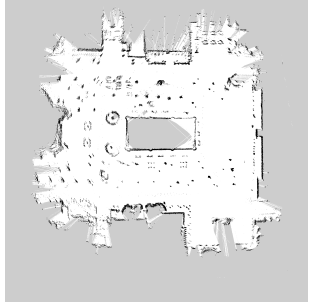
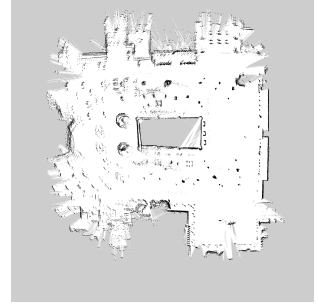
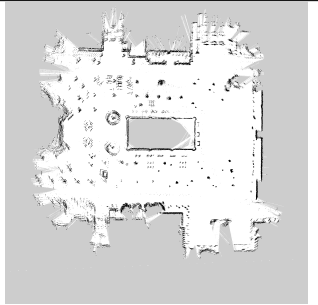
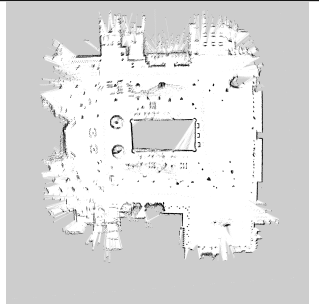
図 2 左の列が左回りの条件での合成地図、右の列が右回りの条件での合成地図である。今回実験に使用したコースの中心には池があり、地図上でその池に当たる部分は地図中央にある長方形となっている。右回りと左回りで差が出た原因としては、センサの取り付け位置が左右非対称であるため、センサの計測範囲が左右で異なることが考えられる。

図のように、センサ計測範囲の角度のパラメータの変更では、地図の大きな歪みには繋がらなかった。しかし、池の形状等は角度が小さいと精度が悪いというのが人間が見て分かる。そこで、この比較を客観化することを試みた。

4.2.4 比較プログラムの実装

比較の対象とするのは、1 枚の基準地図と 10 枚の合成地図である。これらを客観的に測定・比較する手法を検討したが、作成した地図が実環境をどの程度正確に反映しているかを客観的に評価している既存研究は現時点では確認できなかった。そのため、合成地図を少しずつ動かしたデータを基準地図と比較するという方法を実装した。

表 2: 各条件で得られた合成地図

計測角度	左回り	右回り
60°		
120°		
180°		
240°		
270°		

10枚の合成地図を元に、 x 軸 y 軸それぞれ -20 ピクセルから 20 ピクセルまで 1 ピクセルごとの平行移動、 -5 度から 5 度まで 0.2 度ごとの回転移動、 0.8 倍から 1.2 倍まで 0.05 倍差ごとの拡大縮小を組み合わせ、この変形で、背景が欠けた部分はもとの地図で用いられている灰色を補完して、合成地図 1 枚あたり 77 万 1579 通りの比較用地図を作った。これらの比較用地図と基準地図との比較は、全ての画素のうち、数値が一致しているものの割合を一致率として出すという手法で行った。このうち、一致率の高かったものを合成地図 1 枚ごとに選び、そのときの移動量等も記録した。これらの過程は、画像を毎回生成・保存すると負荷がかかるため、OpenCV を用いて画像を配列として扱うようにして実装した。

5 結果

図5は10通りの条件ごとに合成地図から生成した比較用地図と基準地図を比較した一致率のうち、それぞれの条件ごとに一致率が最高のもを示したグラフである。横軸が条件、縦軸が一致率を表す。このうち最大値は 120° 、右回り、 $(1, -1)$ 移動、回転 0 度、 1 倍のときの 90.11% 、最小値は 60° 、左回り、 $(20, 20)$ 移動、回転 0 度、 0.9 倍のときの 86.11% だった(表3)。このように、いずれの条件でも一致率は約 86% から 90% の間に収まり、地図同士の一貫率にはほとんど差が出なかった。

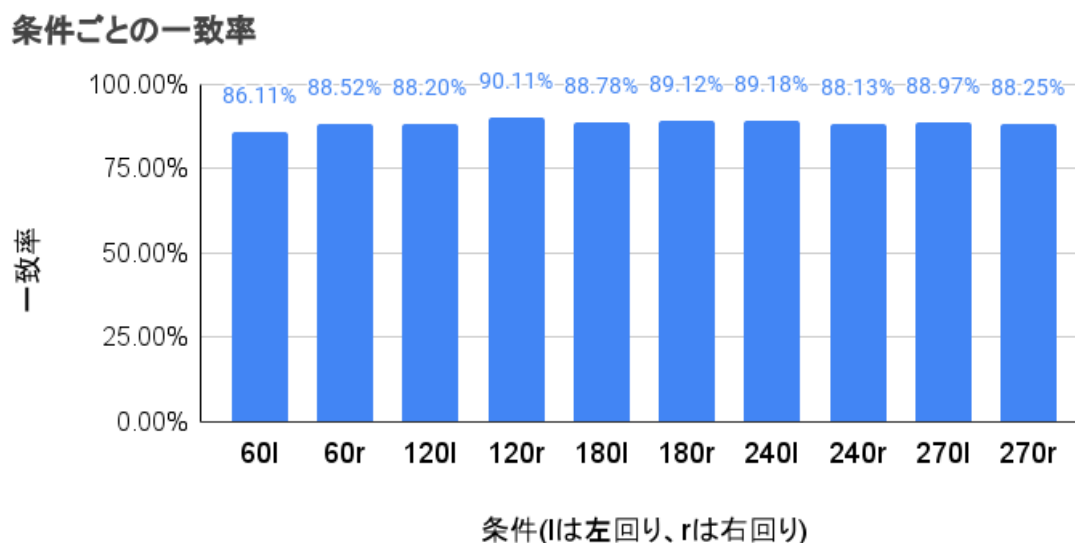


図5: 合成地図と基準地図の一致率を表すグラフ

6 まとめ

センサの計測範囲によって地図の正確性が影響を受ける傾向は見られたが、今回実装した手法を用いた比較では、センサの計測範囲によって地図の一致率には大きく差は出なかった。これは、地図に通路以外を表す灰色の部分が多く、そこを一致と判定するためと考えられる。今後の課題として、地図の本質である通路部分のみの比較を行うなど、比較手法の再検討が考えられる。また、比較後に最も成績の良い地図で実際に自律走行をしてその正確性を検証したり、地図作成時と自律走行時で異なるセンサの観測範囲を設定したときの自律走

表 3: 一致率最大時の条件

角度	左・右回り	平行移動 (x, y)	回転	拡大縮小	一致率
60°	左回り	(20, 20)	0°	0.9 倍	86.11%
60°	右回り	(-14, -1)	0°	1 倍	88.52%
120°	左回り	(16, 19)	0°	0.9 倍	88.20%
120°	右回り	(1, -1)	0°	1 倍	90.11%
180°	左回り	(19, 12)	0°	0.9 倍	88.78%
180°	右回り	(0, -8)	0°	1 倍	89.12%
240°	左回り	(20, 2)	0°	0.9 倍	89.18%
240°	右回り	(20, 3)	0°	0.9 倍	88.13%
270°	左回り	(20, 15)	0°	0.9 倍	88.97%
270°	右回り	(20, 7)	0°	0.9 倍	88.25%

行の正確性の検証なども考えられる。さらに、今回対象としたセンサの計測角度以外のパラメータが地図の正確さに与える影響を調査することも課題である。

7 謝辞

本研究および本論文の執筆にあたって、とても丁寧に指導して下さった新出尚之准教授に心から感謝いたします。また、本研究に使用した Arno 及び関連プログラム等を提供して下さった北陽電機の皆様にも、お礼申し上げます。ありがとうございました。

参考文献

- [1] Nakanoshima Robot Challenge. 中之島ロボットチャレンジの概要. <https://www.nakanoshima-rc.jp/concept.html>. (2022 年 2 月 11 日閲覧).
- [2] つくばチャレンジ実行委員会事務局. つくばチャレンジ概要 — つくばチャレンジ 2021. <https://tsukubachallenge.jp/2021/about/overview>. (2022 年 2 月 11 日閲覧).
- [3] 江口純司, 尾崎功一. DGPS 測位点の精度評価に基づく占有格子地図の作成手法の構築. 日本機械学会論文集 (C 編), 第 78 巻, pp. 129–138.
- [4] 原祥堯. ROS を用いた自律走行. 日本ロボット学会誌, 2017.
- [5] 松本勉, 油田信一. 経路地図に従った移動ロボットの自律走行システム. 日本ロボット学会誌, 第 5 巻, pp. 351–360, 1987.
- [6] 北陽電機株式会社. URM-40LC/LCN-EW. <https://www.hokuyo-aut.co.jp/search/single.php?serial=189#program>. (2022 年 1 月 27 日閲覧).
- [7] 上田隆一. 移動ロボットのための ROS パッケージの紹介と実機への導入方法. 計測と制御, 第 57 巻, pp. 715–720, 2018.