「i–Cart mini」をベースにした自律移動ロボットの開発

徳安 一, 河野 隆人, 田中 大揮, 山崎 飛鳥, 山下 優輝, 西田 健

九州工業大学工学部　CIR-KIT B



Fig.1 KIT-C5の外観

# はじめに

CIR-KIT Bは，九州工業大学工学部の学部生を中心としたロボット開発チームであり，屋内外の移動を行う福祉ロボットの開発に取り組んでいる．自律移動ロボットが人間の生活環境内で活動するにあたって必要とされる機能は安全性の確保を始めとして多くがあるが，本稿で報告するロボット（KIT-C5）は，その中でも最も基本的な機能である「地図情報に基づく自律移動と障害物回避機能」に焦点を当てた開発を行った．開発メンバは，つくばチャレンジに初参加であるため，自律走行による課題達成を目指した．横断歩道の横断と人物探索には取り組んでいない．

# KIT-C5の構成

KIT-C5の外観をFig.1に示す．本節ではその構成を説明する．

## ハードウェア

筑波大学知能ロボット研究室「T-frogプロジェクト」によって開発された「i-Cart mini」[1]を利用して開発した．これはオープンソースハードウェアであり，設計図や必要な部品リストが公開されているため，ハードウェアの設計工程を大幅に削減することができる．i-Cart miniは，二つのブラシレスモータを駆動するためにTF-2MD3-R3を搭載している．さらに，ロボット制御用にUbuntu 16.04をインストールしたラップトップを一台搭載した．一方で，標準の設計では要求される高さを満たさない．また，緊急停止スイッチを押しやすい箇所に搭載することが難しい．そこで， 600[mm]の高さのロボットの筐体を構成した．さらに，LRFを前方に一台搭載した（詳細は後述）．LRF搭載のために高さの調整可能なマウンタを製作した．ロボット筐体の外観をFig.2に示す．

外装はタイヤを含む車体全体を覆うカバーをプラスティック段ボールによって製作した．さらに，ロボットが小型であるため，存在を目立たせるために赤色の外装とした．雨対策は行っていない．実験走行日に雨が降ったが，ビニールシートで車体を覆うことで対処した．



Fig.2 作成したロボットの筐体

### 2.1.1.センサ

　北陽電機社測域センサUTM-30LX（以下URG）を搭載した．センサの写真をFig.3に示す．また，ホイールオドメトリはモータに搭載されているロータリエンコーダにより取得できる．

## 2.2ソフトウェア



Fig.3北陽電機社測域センサ

UTM-30LX

### 2.2.1. Yp-Spur

　Yp-Spur[2]は，筑波大学知能ロボット研究室によって開発された移動ロボットのためのミドルウェアである．制御基板TF-2MD-3R3と連携することで直進や旋回などの簡易な制御コマンドでロボットを制御することが可能である．また，制御基板で計算されたホイールオドメトリを取得することができる．オドメトリの計算に必要なパラメータはファイルから変更することができる．搭載したi-Cart miniをYp-Spurを利用して制御した．

### 2.2.2. ROS

KIT-C5のソフトウェア開発にはROS[3]を利用した．バージョンはKinetic Kameを利用した．ROSを利用することで，他のロボットで開発したソフトウェアや世界中で開発された様々なソフトウェアを再利用することができる．ROSでは自律移動ロボットのためのNavigationスタックなどが提供されており，自律移動の機能を比較的短い時間で構築することが可能である．他にもセンサ情報の可視化やデータロギングなどデバッグのためのツールも多く提供されている．また，Yp-Spurとの連携のために「ypspur\_ros\_bridge」が提供されている．このパッケージを利用することで，ROSの速度指令コマンドをYp-Spurに送ることができ，ROSのnavigationスタックで出力される速度指令をもとにロボットを制御することができる．

### 2.2.3. 自己位置推定

　自己位置推定にはamcl（adaptive Montecalro localization）パッケージを利用した．これは，あらかじめ取得した環境地図をもとに，オドメトリとURGによって得られる計測値をパーティクルフィルタにより統合し，ロボットの自己位置を推定する．

### 2.2.4. 環境地図

　環境地図は「gmapping」パッケージを利用して二次元の占有格子地図として構成した．作成した地図にはノイズが混入していたため，何もない箇所で回避動作が発生する不具合が生じた．そこで，画像編集ソフトを用いて，作成された地図のノイズ部分を手作業で消去した．

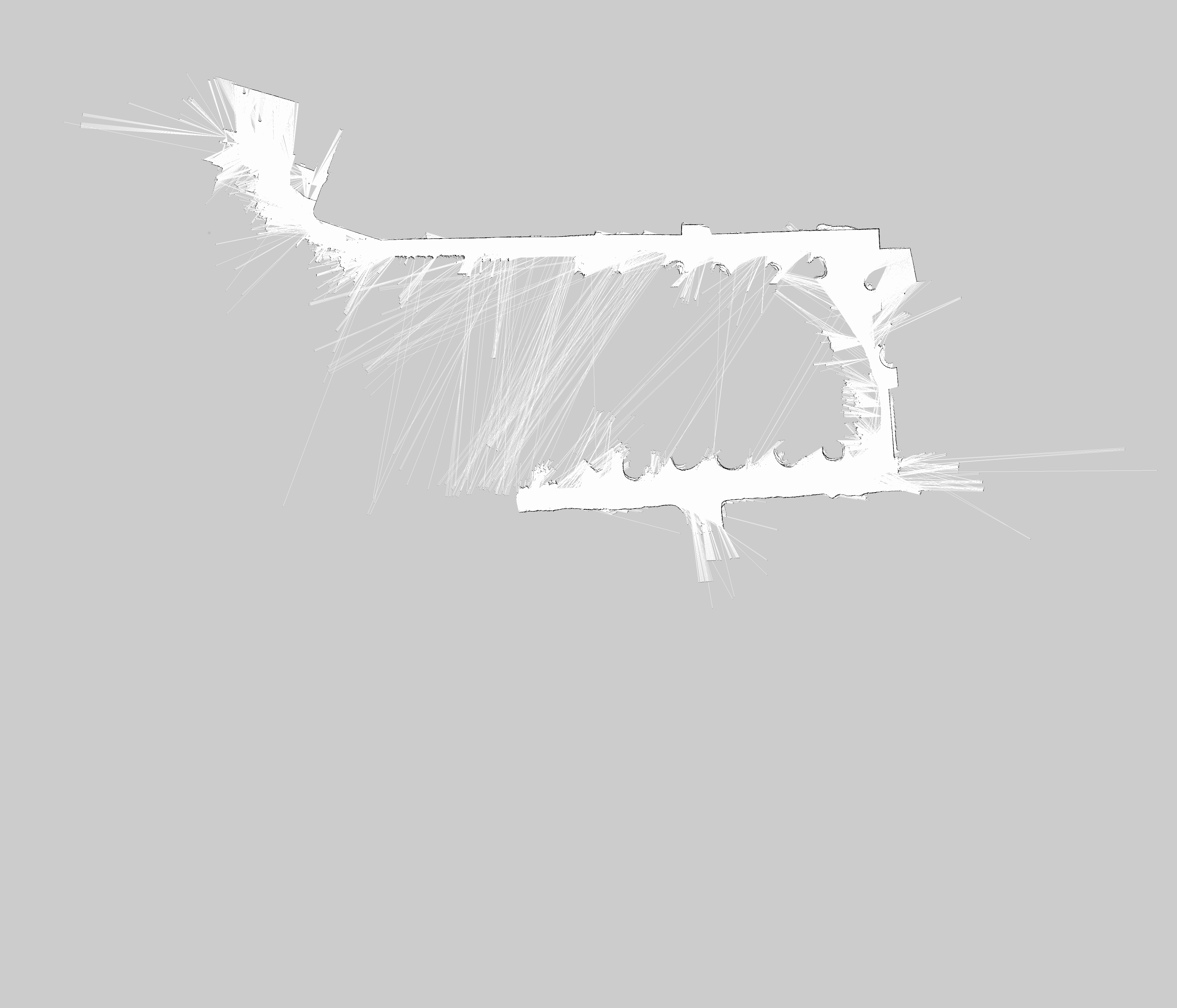


Fig.4 gmappingにより作成した大清水公園の地図

### 2.2.5. 経路計画

　環境地図上に設定した「waypoint」を順に辿るようにして経路が生成される．障害物回避やグローバルパスプランニングなどにはROSの「move\_base」で提供されているパッケージ群を利用した．これは，数メートル間隔で設定した「waypoint」を目指して走行し，その近くに到達すると，新たな「waypoint」を目標として再設定する．これを最終的なゴールに到達するまで繰り返すことによって，長距離の自律移動を実現した．

# 実験結果

## 実験走行結果

　第6回と第7回の実験走行に参加した．実験走行では，ゲーム機用コントローラを用いてロボットを操縦し，大清水公園の地図を作成した．作成した地図をFig.4に示す．大清水公園を一周することができなかったため，大清水公園外の地図は取得できていない．i-Cart miniとYp-Spurによって得られるホイールオドメトリが十分に高精度だったため，精度の良い地図を作成することに成功した．また，作成した地図を利用して， 20[m]程度の自律走行に成功した．

## 本走行結果

本走行では，スタート直後にロボットが後方に進んでしまい，出走を失敗した．オペレータの操作ミスが原因として考えられる．プログラムの起動手順が複雑であり，それらの手順の自動化が十分ではなかった．プログラム起動手順の自動化，簡易化が今後の課題である．

# 考察

自律走行距離が伸びなかった原因として，ハードウェアの問題が大きい．i-Cart miniに施した追加の改造により，ロボットの重心位置が前方および上方に移動したため，走行時にロボットがピッチ方向に大きく振動した．そのため，URGの計測データに振動による計測変化が乗じられ，自己位置推定が失敗する場合が多く発生した．

また，車体下部に搭載したバッテリに縁石などが引っ掛かり，動けなくなってしまう問題も発生した．

屋内では十分な自律走行性能を有していたが，屋外環境での実験が十分でなかったため，事前にこれらの問題を把握できていなかった．

# 開発体制について

学部1および2年生を中心として開発が行われた．いずれのメンバも基本的なC言語の理解があるのみで，ROSなどには不慣れであった．チュートリアル等やWikiなどによりこれらの学習を進め，最終的には自律走行ロボットの開発を達成することができた．

# 今後の課題

## ハードウェア

外装は強度不足，防水加工が難しい，整備性が悪いという問題があった．今後はFRPの一体成型による外装の制作に取り組みたい．また，重心位置を考えた構造に変更することが考えられる．

## センサ

自己位置推定の安定性向上や自律走行時に死角ができないためにURGを複数取り付けることを検討したい．さらに，GPS測位モジュールなども搭載し，URGによる自己位置推定が破たんした場合にも自律走行が続けられる構成とすることが考えられる．

## 開発体制

ハードウェアの製作が遅れたために，ソフトウェアの十分な開発期間が取れなかった．そのため，実験も十分に行うことができなかった．スケジュールの管理が重要な課題である．また，シミュレーション環境を準備できなかったため，ハードウェアとソフトウェアの開発を平行して進めることが出来なかった．例えばgazeboなどを用いることによってロボットの動的なシミュレーション環境を構築したい．

# まとめ

オープンソースハードウェアであるi-Cart miniとYp-Spur，ROSを組み合わせることによってロボット開発に不慣れなメンバでもロボットの自律走行を達成させることができた．

# 謝辞

　つくばチャレンジ実行委員会および関係者の皆さま，URGを貸し出していただいた北陽電機株式会社様にはこの場を借りてお礼申し上げます．

# 参考文献

[1] T-frog Project, “ロボットフレーム i-Cart mini”, [Online]. Available: http://t-frog.com/products/icart\_mini/

[2] Iida, Shigeki, and Shinichi Yuta. "Vehicle command system and trajectory control for autonomous mobile robots." Intelligent Robots and Systems' 91.'Intelligence for Mechanical Systems, Proceedings IROS'91. IEEE/RSJ International Workshop on. IEEE, 1991.

[3] Open Source Robotics Foundation “ROS” 2016. [Online]. Available: http://www.ros.org/