

自律移動ロボットの多機能なタスクを遠隔実行可能なスマートフォン向けロボットアプリケーションの開発

猪口 岳広[†] 松原 佑樹[†] 森岡 一幸[†]

明治大学[†]

1. はじめに

近年、さまざまな分野でロボット技術が発展し、飲食店などで人の代わりに接客を行う接客ロボットなどが人の生活を助けている。また、複数台のロボットを人がスマートフォンのアプリケーションから簡単に操作でき、ユーザの持つ目的を果たすために必要なタスクを遠隔操作により実行するシステムが求められる。本稿では、自律移動ロボットが持つ多機能なタスクをスマートフォンのアプリケーションから遠隔実行することができ、ユーザの目的に応じて操作するロボットを複数台の中から選択することができるシステムとロボットの自律走行システムのサポートや ROS プログラムの遠隔実行を行うスマートフォンアプリケーションを開発する。ROS プログラムの遠隔実行を実現することで、この世に存在する ROS プログラムにより開発されたロボットシステムをスマホアプリから簡単に取り扱うことができる。

2. 提案する遠隔実行スマホアプリ

2.1 ロボットネットワーク

本稿では、ロボットネットワークとして複数台の移動ロボットとユーザをつなぐことができ、ユーザが目的に応じて操作したいロボットを容易に選択することができる Rowma システム [1][2] を用いている。Rowma システムは、接続した任意の ROS ベースロボット間やユーザとロボット間でトピック通信を行ったり、ROS システム内のノードの遠隔実行をしたりできる、ROS ロボットの管理システムである。

2.2 スマートフォンアプリケーション

2.1 節で説明した Rowma システムは、Kotlin の SDK を提供している。開発したアプリケーションは Android studio で開発しており、言語は Kotlin を用いている。開発したスマートフォンアプリケーションを図 1 に示す。Rowma に接続した際に各ロボットには、ID が割り振られており、Rowma では、その ID によりロボットを識別する

Development of a smartphone application to remotely launch multiple functions of autonomous mobile robots
Takahiro INOKUCHI[†], Yuuki MATSUBARA[†], Kazuyuki MORIOKA[†]
Meiji University[†]



図 1. 接続中のロボット一覧

ことで、任意のロボットの操作を可能にしている。また、スマートフォンのアプリケーションで Rowma に接続しているロボットの ID 一覧を取得することができるため、図 1 のような接続中のロボット一覧を表示することができる。また、画面中央にボタンを配置し、ボタンを押すことでそれぞれロボット呼び出し、任意のゴール地点への自律走行、ロボットの ROS ノードの遠隔実行を行う画面へと遷移する。これらの三つのロボットシステムについては後述する。

2.3 スマホアプリによる遠隔実行

本稿では、自律移動ロボット上の多機能なタスクを遠隔実行することを目的としている。開発したスマホアプリでは、主に 2 つの遠隔実行を行っている。1 つ目は、ROS コマンドの遠隔実行である。ROS を用いたロボット開発において主に使用される ROS コマンドは `roslaunch` コマンドと `roslaunch` コマンドであり、これらの ROS コマンドをスマホアプリから遠隔実行することができる。2 つ目は、スマホアプリから自律走行に必要なロボットの初期位置やゴール地点を遠隔で設定することである。スマホアプリからロボットに対して ROS トピックを送信することで実現している。

3. ROS コマンドの遠隔実行

3.1 システム概要

本章では、2.3 節で述べた ROS コマンドの遠隔実行について説明する。遠隔実行システムの概要を図 2 に示す。2.1 節で述べたようにロボットが Rowma に接続すると各ロボットに ID が割り振られ、接続中のロボットを管理している。2.2 節

で述べたようにスマートフォンのアプリケーションでその接続中の ID 一覧を受け取ることができるが、スマホアプリで受け取った際には、接続中のロボットの ID に紐づく形で、その ID を持つロボット上で実行可能なすべての ROS コマンドを取得することができる。これらの実行可能コマンドをロボット上のターミナルで実行するように命令する機能が Rowma の Kotlin SDK に用意されているため、任意のロボット上で任意の ROS コマンドを遠隔実行することが可能である。開発したアプリケーションでの遠隔実行の様子を図 2 に示す。図 1 中の中央部のボタンのうち、「ROS ノードの遠隔実行」を選択した後に、画面上部の「接続中の Robot UUID 一覧」から遠隔実行を行うロボットの ID を選択すると、そのロボット上で実行可能な `roslaunch` コマンドと `roslaunch` コマンドが表示され、実行したいコマンドを選択することでスマホアプリからの ROS コマンドの遠隔実行を実現している。

3.2 動作確認実験 1

本稿では、遠隔実行可能な ROS コマンドのうち、`roslaunch` コマンドの実行の動作確認実験を行う。Yujin Robot 社製の Turtlebot2[3]で実行可能な `roslaunch` コマンドのうち、自動で充電スタンドへドッキングすることのできる `roslaunch` コマンドを実行する。この自動ドッキングのプログラムはあらかじめインストールしておく。3.1 節で説明した手順で、自動ドッキングに必要な `roslaunch` コマンドをスマホアプリから実行すると、Turtlebot が充電スタンドへと自動でドッキングする様子を確認することができた。このように、自分以外の方が開発した ROS パッケージをアプリで取り扱うことができる。

4. 任意の目的地への自律走行

4.1 システム概要

開発したスマホアプリでは、自律走行に必要なロボットの初期位置とゴール地点をスマホアプリから遠隔で設定し、設定したゴール地点へ

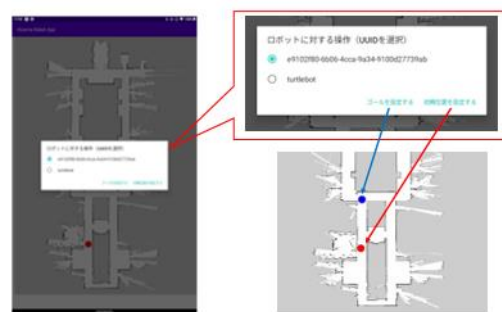


図 3. スマホアプリからの目的地設定

の自律走行を行っている。このとき、自律走行に必要な `roslaunch` コマンドを 3 章で述べた遠隔実行システムを用いてスマホアプリから実行することでロボット上の PC で必要な作業を最小限に減らしている。

4.2 動作確認実験 2

本実験では、明治大学中野キャンパス 11 階において図 3 中の青点で示した地点にゴール地点を設定し、図 4 中の赤点で示した地点からロボットの自律走行を行った。実験では、スマホアプリで設定したゴール地点へと走行することができた。

5. 考察とまとめ

実験 1 では、実行したいプログラムを対象のロボットの PC 上にインストールするだけでスマホアプリからそのコマンドを遠隔実行することができることが分かる。また、実験 2 では、従来ロボットの PC 上で必要であった ROS コマンドの実行やゴール地点の設定がスマホアプリから遠隔実行することができることが分かる。これらの機能は、ロボットが複数台に増えても簡単に操作したいロボットを切り替えることができ、大規模なロボット実験を容易にしてくれることが分かる。

本稿では、本来ロボット上の操作が必要であった ROS コマンドの実行やロボットの自律走行に必要なゴール地点の設定をスマホアプリから遠隔実行し、作業の効率化を図った。今後はスマホアプリの UI の改良や、複数台のロボットを用いた大規模な走行実験などを行っていく。

参考文献

- [1] Ryota Suenaga, Kazuyuki Morioka, 「Rowma: A Reconfigurable Robot Network Construction System」, IEEE/SICE International Symposium on System Integrations (SII2021), pp. 537-542, 2021.01, Online, IEEE Xplore
- [2] <https://github.com/rowma>
- [3] <https://github.com/yujinrobot/kobuki>



図 2. スマホアプリからの遠隔実行