

# 卒 業 論 文

## 通信の不安定性を考慮した遠隔操作ロボット制御方式の実装と 評価

— 副題ああだこうだ —

22TC074 河部 修輔

指導教員 宮澤 元

2026 年 1 月

南山大学 理工学部 電子情報工学科

---

Title Foo and Bar

— Subtitle Bar or Foo —

20yyMTxxx NENZAN Taro

20yyMTxxx NENZAN Hanako

Supervisor SIDO Kyoju

Month 20yy

Department of Software Engineering

Faculty of Science and Engineering

Nanzan University

## 要約

ここに本論文の要約を書く。要約は論文のエッセンスを抜き出したものであるので、ここで取り扱う問題、その問題を解決するための手法、および主な成果が書かれていなければならない。要約を読むだけで、論文の概要が分かり、読者にとって興味を抱く内容か否かが分かるようになっている必要がある。

日本語と英語の両言語で要約を書くが、必ずしも 1 対 1 に対応する文章になっている必要はない。それぞれにふさわしい表現があるからである。

## Abstract

In this part, the abstract of this paper is described. Since ‘abstract’ means the essence or summary of the paper, the abstract should include the description on the problems treated in the paper, the author’s approach for solving those problems, and the main results. Readers will understand the outline of the paper, without reading other parts, and be able to decide whether they will have interests in the paper or not.

# 目次

第 1 章	はじめに	1
第 2 章	背景	2
2.1	自律移動ロボットの社会的背景	2
2.2	自律移動ロボットにおける認識の重要性	2
2.3	ロボットにおける計算資源の制約	2
2.4	クラウド技術の活用と期待	3
2.5	通信環境の不確実性	3
2.6	通信品質低下が与える影響	3
2.7	処理方式切替の有用性	3
2.8	処理方式切替に関する既存の考え方	4
2.9	固定的な判断基準の課題	4
2.10	本研究の位置づけ	4
第 3 章	設計	5
3.1	システム設計	5
3.2	切り替え機構の設計思想	5
3.3	通信品質指標に基づく切り替え設計	5
3.4	ローカル構成の設計	6
3.5	クラウド構成の設計	6
3.6	本章のまとめ	6
第 4 章	実装	8
4.1	実装概要	8
4.2	ハードウェアおよびソフトウェア構成	8
4.3	通信品質計測と動的切り替え機構	8
4.4	SLAM 処理の実装	9
4.5	本章のまとめ	9
第 5 章	実験・結果	10
第 6 章	考察	11
第 7 章	まとめ	12
	参考文献	13

# 第 1 章

## はじめに

- 1.0.1 研究の概要
- 1.0.2 研究の目的
- 1.0.3 本研究の貢献
- 1.0.4 本論文の構成

## 第2章

# 背景

### 2.1 自律移動ロボットの社会的背景

近年、物流倉庫や工場、医療・福祉、さらには災害現場等において、自律移動ロボットの活用が急速に拡大している。これらの分野では、人手不足の深刻化や作業効率向上への要求を背景として、人が行ってきた作業をロボットに代替させる動きが進んでいる。特に、搬送や巡回といった反復的な作業においては、ロボットによる自動化が高い効果を発揮すると期待されている。

また、少子高齢化が進む社会においては、将来的な労働力不足が懸念されており、ロボット技術の活用は重要な解決策の一つと考えられている。このような背景のもと、自律移動ロボットには、限られた人手による管理で長時間運用できることや、予期せぬ状況においても安定して動作することが求められている。そのため、ロボットが人の介入に依存せず、自律的に環境を認識し判断できる能力の重要性が高まっている。

### 2.2 自律移動ロボットにおける認識の重要性

自律移動ロボットが人と共存する環境で安全に動作するためには、周囲の状況を正確に把握し、それに基づいて適切な行動を選択する必要がある。ロボットは自身の位置を誤って認識した場合、目的地への到達に失敗するだけでなく、障害物との衝突や危険な挙動を引き起こす可能性がある。そのため、環境認識と自己位置推定は、自律移動ロボットにとって最も基本的かつ重要な機能の一つである。

このような環境認識を実現する技術として、自己位置推定と環境地図作成を同時に行う SLAM が広く用いられている。SLAM は、ロボットが未知環境を移動しながら、自身の位置を推定しつつ地図を生成するための枠組みであり、多くの自律移動ロボットシステムにおいて中核的な役割を果たしている。SLAM の結果は、経路計画や障害物回避などの上位機能にも利用されるため、その精度と安定性はシステム全体の性能に大きな影響を及ぼす。

### 2.3 ロボットにおける計算資源の制約

一方で、SLAM を含む環境認識処理は、多量のセンサデータをリアルタイムで処理する必要があり、高い計算負荷を伴うことが知られている。実際の運用を想定した移動ロボットでは、小型化や軽量化が求められることが多く、搭載可能な計算機の性能には制約がある。また、バッテリー駆動で長時間稼働する必要があるため、消費電力の観点からも高性能な計算機の搭載は容易ではない。

このような制約の下では、ロボット単体で高度な処理を実行することが難しくなる場合がある。計算資源が不足すると、処理速度の低下や精度の劣化が生じ、結果としてロボットの行動に悪影響を及ぼす可能性がある。そのため、ロボットのハードウェア制約を考慮しつつ、必要な処理性能をどのように確保するかは、自律移動ロボットの設計における重要な課題である。

## 2.4 クラウド技術の活用と期待

この課題に対する解決策の一つとして、近年ではクラウド技術をロボット分野に応用するクラウドロボティクスが注目されている。クラウド環境では、高性能な計算資源を柔軟に利用できるため、ロボット単体では困難な処理を外部で実行することが可能となる。これにより、ロボットのハードウェア構成を簡素化しつつ、高度な機能を実現できる可能性がある。

特に SLAM のような計算負荷の高い処理においては、クラウド側で処理を行うことで、高精度な結果を得られることが期待されている。また、複数のロボットから得られた情報を集約することで、より広範な環境認識が可能になるなど、クラウド利用ならではの利点も指摘されている。このように、クラウド技術は自律移動ロボットの性能向上に寄与する重要な要素と考えられている。

## 2.5 通信環境の不確実性

しかしながら、クラウドを利用した処理は通信環境に強く依存するという課題を抱えている。移動ロボットは多くの場合、無線通信を用いて外部と接続されるため、通信遅延や通信速度の変動、さらには通信断が発生する可能性がある。これらの問題は、屋内外の環境差や電波干渉、アクセスポイントとの距離など、さまざまな要因によって引き起こされる。

さらに、移動ロボットは常に位置を変えながら動作するため、通信環境も時間とともに変化する。このような通信品質の変動は、クラウドに処理を依存するシステムにとって大きな不確実要因となる。通信が不安定な状況では、クラウド側での処理結果を適切なタイミングで受け取ることができず、ロボットの挙動に影響を及ぼす可能性がある。

## 2.6 通信品質低下が与える影響

通信品質が低下した状態でクラウド処理を継続すると、処理結果の遅延や欠落が発生する恐れがある。このような問題は、自己位置推定の精度低下や地図更新の遅れといった形で現れ、結果としてロボットの行動計画や制御に悪影響を及ぼす可能性がある。特に、人や障害物が存在する環境では、認識の遅れが安全性の低下につながる危険性も否定できない。

このため、通信環境が悪化した場合にどのように対応するかは、ロボットシステムの信頼性を左右する重要な課題である。通信状況に応じて処理方式を切り替えるといった柔軟な対応は、ロボットの自律性を高め、安定した運用を実現する上で有効な考え方である。

## 2.7 処理方式切替の有用性

通信品質の変動が避けられない環境において、単一の処理方式に依存したシステム設計は、安定した動作を維持する上で限界がある。特に、通信が安定している状況と不安定な状況とでは、ロボットに求められる処理の在り方が大きく異なる。このため、状況に応じて処理方式を切り替えるという考え方は、合理的かつ実用的な解決策の一つであると考えられる。

例えば、通信品質が十分に良好な場合には、外部の計算資源を活用することで、高精度な処理結果を得ることが可能となる。一方で、通信品質が低下した場合には、ロボット単体で完結する処理方式に切り替えることで、通信遅延や通信断の影響を回避できる。このように、複数の処理方式を適切に使い分けることで、システム全体の信頼性を向上させることが期待される。

また、処理方式切替は、計算資源や通信資源を効率的に利用するという観点からも有用である。常に高負荷な処理をクラウドに依存するのではなく、必要に応じて処理の分担を変更することで、無駄な通信や計算を削減

できる可能性がある。このような柔軟な運用は、実環境における長時間運用を想定したロボットシステムにおいて重要な要素である。

以上の理由から、通信状況に応じて処理方式を切り替えるというアプローチは、自律移動ロボットの安定運用を実現する上で有効な手段であり、本研究が着目する基本的な考え方の一つである。

## 2.8 処理方式切替に関する既存の考え方

これまでに、通信品質を指標として処理方式を切り替える手法が検討されてきた。一般的には、通信遅延や応答時間などの指標を用い、それらがあらかじめ定められた基準を超えた場合に処理方式を変更するという考え方が採用されている。このような方法は、判断基準が明確であり、システム設計が比較的容易であるという利点を持つ。

また、固定的な基準を用いることで、処理方式の切替を単純化できる点も評価されている。しかしながら、このようなアプローチは、通信環境が常に一定であることを前提としており、実際の運用環境においては必ずしもその前提が成り立たない場合がある。

## 2.9 固定的な判断基準の課題

実環境における通信品質は、短時間で大きく変動することが多い。一時的な通信遅延の増加や瞬間的な品質低下は頻繁に発生し、これらをすべて異常として扱うことは現実的ではない。固定的な判断基準を用いた場合、このような一時的な変動に過剰に反応し、不必要な処理方式の切替が頻発する可能性がある。

このような頻繁な切替は、システム全体の安定性を損なう要因となる。また、異なる環境や運用条件に対して同一の基準を適用することは、環境適応性の観点からも課題が残る。このように、固定的な基準のみに基づく判断には限界がある。

## 2.10 本研究の位置づけ

以上の背景を踏まえると、通信品質の変動を考慮し、状況に応じて柔軟に処理方式を選択できる仕組みが求められている。特に、通信状態を瞬間的な値のみで判断するのではなく、時間的な変化を考慮した判断を行うことが重要である。このような考え方は、実環境における自律移動ロボットの安定運用において、本質的な課題の一つである。

本研究は、この課題に着目し、通信品質の変動を考慮した処理方式選択の在り方について検討するものである。次章では、本研究の目的を明確に示す。

## 第 3 章

# 設計

### 3.1 システム設計

本研究では、ローカル構成およびクラウド構成の 2 つの処理構成を切り替えて利用する自律移動ロボットシステムを提案する。本システムは、ロボットが取得したセンサデータを用いて自己位置推定および環境地図生成を行う点では共通しているが、処理を実行する計算資源の違いによりローカル構成とクラウド構成に分かれている。

クラウド構成は高精度な SLAM 処理を可能とする一方で、通信品質に強く依存する。そのため、通信状態が不安定な状況で頻繁に処理構成を切り替えることは、システムの安定性やリアルタイム性を低下させる要因となる。

本研究では、通信状態を表す指標を用いて処理構成を自動的に切り替える機構を設計し、不必要な切り替えを抑制しつつ、安定性と精度の両立を目指す。

### 3.2 切り替え機構の設計思想

本研究における切り替え機構の設計において最も重視する点は、通信状態の一時的な変動によって不必要な構成切り替えが発生することを防ぐことである。

通信環境は時間とともに変動するため、瞬間的な通信遅延の増加のみを基準として処理構成を切り替えると頻繁な切り替えが発生し、システム全体の挙動が不安定となる可能性がある。特に SLAM 処理では、構成の切り替えに伴う処理状態の変化が、自己位置推定の精度低下を招く恐れがある。

そこで本研究では、通信品質を表す指標を時間特性の異なる複数の観点から評価し、切り替え判断に用いる。具体的には、通信状態の長期的な安定性を表す静的指標と、通信遅延の短期的な変動を捉える動的指標を組み合わせることで、不必要な切り替えを抑制する設計とする。

さらに、本研究では、切り替え判断そのものを目的とするのではなく、切り替え回数を必要最小限に抑えることを設計上の重要な指針とする。構成切り替えは、SLAM 処理の再初期化や内部状態の変化を伴うため、頻繁に発生すると自己位置推定の連続性が損なわれる可能性がある。

そのため、本研究では、通信状態が「悪化し始めた瞬間」ではなく、「一定時間以上にわたり悪化が継続している状態」を検出することを重視する。動的 RTT を時間的に評価する設計は、このような状態判定を可能とし、結果として不必要な構成切り替えの抑制につながる。

### 3.3 通信品質指標に基づく切り替え設計

本研究では、ローカル構成とクラウド構成の切り替え判断に用いる通信品質指標として、パケットロス率および RTT を採用する。これらの指標はそれぞれ異なる時間特性を持ち、切り替え設計において異なる役割を担う。



### 3.3.1 静的指標としてのパケットロス率

パケットロス率は、一定期間における通信の成功率を表す指標であり、通信経路の安定性を評価するために用いる。パケットロスが継続的に発生する環境では、クラウド構成におけるセンサデータや処理結果の欠落が生じやすく、SLAM 処理の信頼性が低下する。

そのため本研究では、パケットロス率を通信環境の良否を判断するための静的な指標として位置付け、一定期間の統計値に基づいて評価を行う。これにより、一時的な通信変動ではなく、通信環境そのものがクラウド利用に適しているかを判断する。

### 3.3.2 動的指標としての RTT

RTT は、ロボットとクラウド間の通信遅延を表す指標であり、ネットワーク負荷や混雑状況に応じて短時間で変動する特徴を持つ。

本研究では、RTT を切り替え判断における動的な指標として用いる。ただし、瞬間的な RTT の増加のみを基準として構成を切り替えると、通信状態の一時的な揺らぎに反応して不必要な切り替えが頻発する可能性がある。

そこで、RTT を時間的に評価し、一定期間における変動傾向を基に切り替えを判断する設計とする。これにより、一過性の遅延増加では切り替えを行わず、通信状態が継続的に悪化した場合のみローカル構成へ移行することが可能となる。

このように、RTT を動的に扱うことで、システムの安定性を維持しつつ、不要な構成切り替えを抑制することを本研究の目的とする。

本研究では、短時間で変動する RTT を主指標として扱い、通信環境そのものの良否を示すパケットロス率を補助的な指標として用いる点に特徴がある。

## 3.4 ローカル構成の設計

ローカル構成は、通信品質の低下が検出された場合においても、ロボットが自律的に動作を継続できるよう設計する。本構成では、センサデータの取得、SLAM 処理、および自己位置推定結果の利用をすべてロボット上で完結させる。

動的 RTT の評価により、通信遅延が継続的に増加していると判断された場合、ローカル構成へ切り替えることで、処理遅延による影響を回避する。

## 3.5 クラウド構成の設計

クラウド構成は、通信環境が安定している状況において、高精度な SLAM 処理を実現することを目的として設計する。パケットロス率が低く、かつ RTT が安定して低い場合に、クラウド構成を選択する。

動的 RTT を用いた評価により、通信遅延が一時的に増加した場合でも、即座に構成を切り替えることなく、クラウド構成を維持できる。これにより、不要な切り替えを抑制しつつ、高精度な処理を継続可能とする。

## 3.6 本章のまとめ

本章では、本研究で提案する自律移動ロボットシステムの設計について述べた。本システムは、ローカル構成とクラウド構成の 2 つの構成から成り、それぞれに明確な役割を持たせることで安定性と精度の両立を目指す。

ローカル構成は、通信環境に依存せずに動作可能な基盤として設計し、クラウド構成は、高精度な処理を必

要とする場面においてローカル構成を補助する役割を担う。さらに、通信品質に応じて両構成を切り替える設計を採用することで、実環境における通信変動に柔軟に対応可能なシステムとした。

次章では、本章で示した設計方針に基づき、ローカル構成およびクラウド構成の具体的な実装について述べる。

## 第 4 章

# 実装

### 4.1 実装概要

本章では、本研究で提案する自律移動ロボットシステムの実装について詳述する。本システムは、ローカル構成およびクラウド構成の 2 種類の処理構成を、通信品質および SLAM 精度に応じて動的に切り替えることが可能である。その実現のため、ハードウェア構成、ソフトウェア構成、通信計測、および切り替え機構の実装に分けて説明する。また、SLAM 処理の精度を保ちつつ、 unnecessary 切り替えを抑制する工夫についても述べる。

### 4.2 ハードウェアおよびソフトウェア構成

ロボットは Raspberry Pi 5 を搭載し、OS には Ubuntu 20.04 を使用する。搭載センサは 2D LiDAR のみであり、ROS 2 を介して `/scan` トピックからデータを取得する。LiDAR は `ldlidar_stlros2` ドライバを用い、リアルタイムで環境情報を配信する。

ローカル構成では、SLAM 処理や TF 座標系の管理を Raspberry Pi 上で完結させる。TF 座標系は `'base_link'`、`'base_laser'`、`'odom'` など複数の static TF を定義し、起動時には古い TF プロセスを自動的に終了する仕組みを実装している。これにより、重複 TF による座標変換の不整合を防ぎ、自己位置推定の安定性を確保する。

クラウド構成では、Raspberry Pi とクラウドサーバ間の通信に CycloneDDS を利用し、ROS 2 ノード間の直接通信を実現している。k3s クラスタ上で SLAM 処理を分散実行することで、高性能な CPU やメモリを活用し、ローカル構成に比べて精度の高い自己位置推定と地図生成を可能としている。

### 4.3 通信品質計測と動的切り替え機構

通信品質の評価には、パケットロス率と RTT を用いる。パケットロス率は静的指標として一定期間の統計値を計算し、通信経路の安定性を評価する。RTT は動的指標として短時間で変動する遅延を評価し、過去の平均値や変動傾向を考慮してヒステリシス制御に用いる。

処理構成の切り替えは Python ノードによって定期的に実行される。具体的には、Ping コマンドによりクラウドサーバの応答を取得し、パケットロス率および平均 RTT を算出する。さらに、SLAM 精度指標である `/slam_quality` トピックの値を組み合わせ、`'dynamic_comm_thresh'` および `'dynamic_slam_thresh'` を計算して、切り替え判断に使用している。

切り替えの際には、現在稼働中の SLAM プロセスや TF publisher を安全に終了し、新しい構成を起動する。SIGINT や SIGTERM などの信号に対応した安全シャットダウン機能も実装されており、異常終了時でも自己位置推定や地図生成が破綻しないよう設計されている。

## 4.4 SLAM 処理の実装

ローカル構成では、‘slam\_toolbox‘をオンライン非同期モードで動作させ、LiDAR データをリアルタイムに処理する。クラウド構成では、より高精度な SLAM 処理を行うため、サーバ上で最適化回数を増やすなどの設定を施し、ローカルよりも精度を重視した地図生成を実現している。

処理切り替え時には、既存 SLAM プロセスを安全に終了し、必要に応じてクラウド側またはローカル側で新たに SLAM を起動する。これにより、自己位置推定の連続性を維持しつつ、通信状態や SLAM 精度に応じた最適な構成を選択できる。

## 4.5 本章のまとめ

本章では、本研究における自律移動ロボットシステムの実装について述べた。Raspberry Pi 5 上でのローカル処理、クラウド構成の k3s 管理、CycloneDDS による通信、通信品質評価に基づく動的切り替え、LiDAR を用いた SLAM 処理および SLAM 精度指標の統合により、安定性と精度を両立させたシステムを構築した。次章では、本章で実装したシステムを用いた評価実験について述べる。

## 第 5 章

# 実験・結果

## 第 6 章

## 考察

## 第 7 章

### まとめ

## 参考文献

- [1] 愛 上男：“プログラミング論”，情報処理論文誌, Vol.1 No.1, pp.1-10 (1991.1).
- [2] Z.MrX：to Make Programming Much Easier, <http://www.abc.def.gh/ijkl.html>.
- [3] 柿 久華子：“万能方法論”，日本出版 1990.