

情報収集のためのロボットプラットフォーム

中山 悟¹ 柳田 将希¹ 平沼 拓也¹
住谷 拓馬¹ 中野 美由紀¹ 菅谷みどり¹

近年、安価なロボットが利用可能になり、多数のロボットを利用したサービスも検討されるようになってきた。ロボットは異なる物理条件下で動作することから、その動作や開発時の記録を逐次ログすることが求められる。しかし、現状ロボットを対象として、データを収集する場合には、データ収集のための API の設計及びプラットフォームが十分整備されておらず情報収集がしづらい問題がある。本研究では、汎用ロボットの基礎的なロギングのためのプラットフォーム RLS(Robot Logging System)を設計、実装し、その仕組みを利用してデータ収集を行った内容、および、今後のロボットの情報収集のプラットフォームについて議論した内容を報告する。

1. はじめに

近年、安価なロボットが利用可能になり、多数のロボットを利用したサービスも検討されるようになってきた。複数台のロボットを利用したサービスにおいては、サービス提供者側は、個々のロボットを管理するために、その情報収集が必要となる。

しかし、ロボットを対象として、データを収集する場合には、ハードウェアプラットフォームや、ソフトウェアプラットフォームが統一されていないため、統一的な形でログを収集することが困難である。また、ロボットを動作させるためには、一般的にミドルウェアが必要とされるが[1][2]ミドルウェアにロギングライブラリを追加し、それらのデータ収集を可能としたものは十分に提供されていない。

本研究では、ロボットのための共通のプラットフォーム開発に向けて、二つの試みを行った内容を報告し、今後のプラットフォームの設計の議論を行うことを目的とする。

2. 提案

全体システム構成を図 1 に示した。ロボットは、制動部の制御の計算を計算機と接続して行うことが一般的である。そこで、今回はまず、ロボットを計算機と接続し、計算機側で情報を収集し、処理する役割を担わせ、そこから命令を発してロボットを動作させる構成とした。また、この際に、ロボットは主に計算機とシリアル接続していることを前提とした。データの送受信は、ロボットのログを集約するサーバを想定し、そのデータをサーバに集約するものとした。

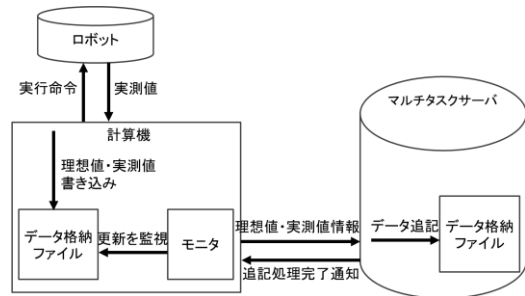


図 1 システム構成

3. 評価

3.1 ロボット API のログ収集の有効性の評価

複数人による同時開発を実現するために、ロボットを用いたプログラミングの授業で、授業受講者に課題を出し、その開発過程のログを収集し、これらのログを比較検討することを目的とした。課題は、前進 (`forwardDistance()`関数) のみを使って、可能な限り 2m に近い位置でロボットを停止させるものとした。課題作成にあたり、発進と停止時の誤差を縮めるために、加速と減速を考慮し作成したサンプルプログラムを提示した。授業は、芝浦工業大学のプログラミング演習 I の実習において 1 年生 108 人で、約 10 人ずつ 10 グループに分けて実施した。比較対象に、同情報工学科、3 年生の別の選択必修の授業にて約 80 人を 10 人ずつ 8 グループに分けてそれぞれ実施した。各グループから得られたデータの中から、直進の加速と減速を想定した課題で使われた `forwardDistance()`関数の速度パラメータを抽出し、関数の試

¹ 芝浦工業大学 情報工学科
Shibaura Institute of Technology, Information Science and Engineering

行ごとの推移を図2, 3に示した。

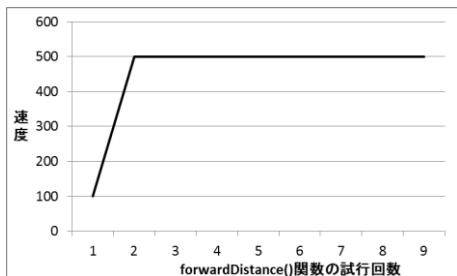


図2 加速と減速をしないプログラムの開発過程

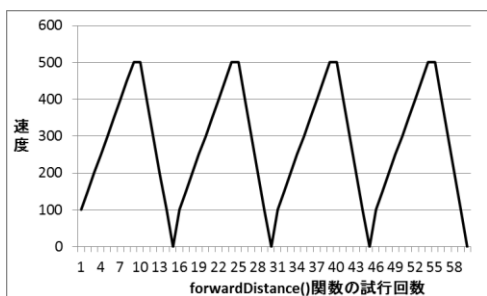


図3 加速と減速をするプログラムの開発過程

API のログを解析することで、プログラムの開発過程を知ることができることが分かった。このことから、ログを取得し、開発過程を理解することで、異なる物理条件下におけるロボットのチューニング過程を記録し、共有することができると考えられる。

3.2 ロボットにおける TCP/IP 通信の基本性能の調査

ロボットを動作させる環境では、現状主にシリアルが用いられている[3]。無線通信の実現方法には、より多くの帯域幅を利用することを前提とした、TCP/IP 通信があるが、iRobot Create をはじめとする多くのロボットには、TCP/IP 通信を行うためのミドルウェアが十分ではない。そこで、本研究では、TCP/IP 通信を行うためのシステム基盤であるソケット通信を設計、実装し、簡易的な評価を行うことで、その実用性について議論するものとした。

調査では、従来のシリアル通信に対して、TCP/IP 通信を行った場合の速度を比較することで、どの程度の速度比があるのかを明らかにすることを目的とした。

まず初めに、ソケット通信環境をロボットに実装した。ロボットには、命令を送る側、受信した命令を実施する側がある。これをサーバ、クライアントプログラムに当てはめるためには、命令を送る側と、実行を行う側を最初に区別して、役割を割り振る必要がある。我々は、クライアントから開始するクライアントサーバのモデルに従い、まずは、クライア

ントが準備完了の packets を送付した後、サーバ側が命令指示側として、クライアントに必要なデータを送付するものとした。これに対して、通常のシリアル通信は、特に区別しなくても、命令の送信側と受信側で役割分担がなされているため通信をそのまま行うものとした。

クライアント側の入力送信後からロボットが前進し始めるまでの時間をそれぞれ 10 回計測した結果を表 1 に示した。

尚、本実験は Intel Core i7, 2.5GHz, 2GB メモリのマシンで実施した。

表 1 実験結果

	シリアル通信のみ	C/S通信とシリアル通信
平均	265.5	355.9
最大	624.6	967.2
最低	32.9	42.0
標準偏差	224.7	287.7

実験結果から、サーバクライアント通信をロボットに移植したときの平均遅延時間は 90.4msec であった。双方向の通信を行う TCP/IP のオーバーヘッドが 1/2 以下であることから、ロボットプログラムの命令実行通信が利用できるレベルにあることを示している。オーバーヘッドを値のみからだけで判断することは困難であるが、TCP/IP を用いたサーバクライアントモデルをロボットに应用することは、現実的な選択肢となりうることを示した。

4. まとめ

ロボットを対象としたデータ収集のためのプラットフォームの実装における汎用性の問題点を述べ、プラットフォームを開発するに当たって要求される点として、ロボットのハードウェア構成を考慮したロギングの仕組みと、TCP/IP ベースの通信制御を挙げた。ロギングの仕組みについては、ロボット演習の実践授業で実験を行い、有効性が示された。通信制御については、シリアル通信と TCP/IP 通信の速度差から、大して遅延しないことが分かった。今後、ログの収集時間を評価し、リアルタイム性の向上を行う。

参考文献

- [1] Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, Andrew Ng: ROS: an open-source Robot Operating System.
- [2] N. Ando, T. Suehiro, K. Kitagaki, T. Kotoku, and W. -K. Yoon, "RT-Middleware: Distributed Component Middleware for RT (Robot Technology)", Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 3555-3560, 2005.
- [3] 内村裕, ロボット制御の視点で見た無線通信の課題と期待. 情報通信学会. 2013.