(305) 通信 その他

LoRa を用いた複数ロボット間位置情報共有システムの試作
Prototyping of Position Sharing Method for Multi-Robot System based on LoRa

松藤 はるか † 稲田 隼輔 † 都築 伸二 † 曽利 仁 ‡ 井上 浩行 ‡ 杉本 大志 †
Haruka MATSUFUJI † Shunsuke INADA † Shinji TSUZUKI † Hitoshi SORI ‡ Hiroyuki INOUE ‡
Masashi SUGIMOTO †
† 愛媛大学大学院　理工学研究科
‡ 津山工業高等専門学校　総合理工学科

1　まえがき

複数個で協調的に動作する自律分散ロボットは，単体ロボットに比べ複雑高度なタスクを速やかに達成する事が可能であると考えられている．その協調動作を実行するためにはロボット間通信が不可欠であり，時分割多元接続 (Time Division Multiple Access: TDMA)，周波数分割多元接続 (Frequency Division Multiple Access: FDMA)，搬送波感知多重アクセス (Carrier Sense Multiple Access: CSMA) などの MAC プロトコルが適用されている．協調ロボットの応用例として近年注目されているスマート農業でも様々なロボットを 1 つのネットワークへ接続して一体的に運用する事が試みられている [1]．本グループが開発する除草ロボットの概念を図 1 に示す．このうちロボット間を含む M2M 通信では，一般的に LTE/3G 回線が基本ネットワークとして用いられることが多いが，新規に導入する際はコストが増加する場合がある．また，その農地がサービスエリア外であり適用できないと言う場合もある．

近年注目されている Internet of Things(IoT) では，低消費電力且つ広域通信を同時に実現する Low Power Wide Area(LPWA) 技術が積極的に実装されており，代表的な無線規格の一つである Long Range(LoRa) は日本では CSMA を採用している．この手法の問題点としては，端末数の増加に比例してデータパケットの衝突が増加する点が挙げられる．しかし，自律分散ロボットを多数用意して作業を行わせる事は，各ロボットのメンテナンスや面積単位での費用対効果の面から一般的に考えられない．更に LoRa は，LTE/3G 回線のエリア外でも使用したいエリアに基地局を設置すれば携帯通信網を補完することができる．

これまでに本グループでは農作業ロボットの自律行動アルゴリズムの構築を遂行しており [2][3][4]，作業用ロボットのマイコンとして Raspberry Pi，Arduino を使用してきたが，複数台のロボットをデータ共有させる形での農作業システムの構築は検討してこなかった．そこで本稿では，ロボット間で協調動作を実行する第一段階として，Raspberry Pi と Arduino を利用し，LoRa Private を用いる事で，ロボット間でデータが共有できることを確認する．これにより，初期コストを低減でき，なおかつ LTE/3G 回線のエリア外でも

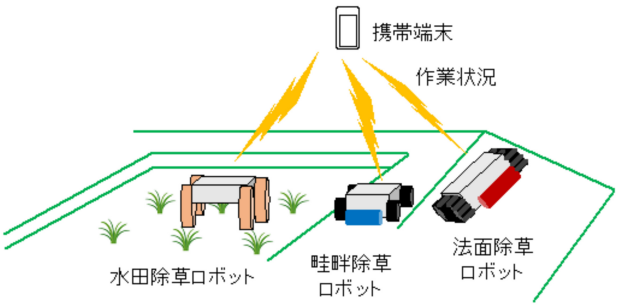

図 1: 開発するシステムが対象とする範囲 [2]

通信が可能である LoRa を利用した自律分散型農作業ロボットシステムの構築について検討する．

2　システム構成

システム構成図を図 2 に示す．Arduino で 9 軸センサモジュール（MPU9255) と，GPS モジュール (NEO-6M-0-001) からデータを取得する．Arduino と 9 軸センサモジュールは I2C 通信をしており，Arduino と GPS モジュールはシリアル通信を行っている．Arduino で取得したデータをシリアル通信で Raspberry Pi に送信する．Raspberry Pi と LoRa Private モジュール間はシリアル通信 (UART) しており，LoRa Private 間で Raspberry Pi で受け取ったセンサデータを送受信する．本節ではそれぞれの機能について説明する．

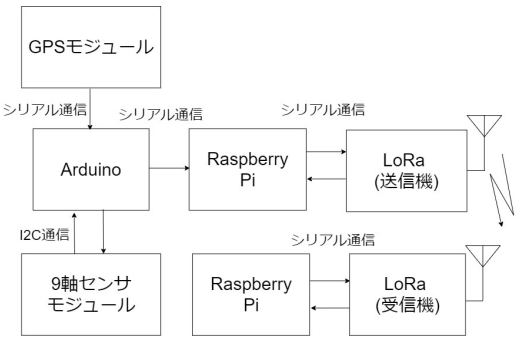

図 2: システム構成図

2.1 Arduino の機能

Arduino は 9 軸センサモジュールと GPS モジュールからデータを取得する．9 軸センサモジュールで加速度，ジャイロ，地磁気を読み取り, GPS モジュールで，時刻，緯度，経度，高度，速度を取得する．情報取得のフローチャートを図 3 に示す．本研究に於いて，I2C 通信に必要な”Wire.h”，GPS データ取得に必要な”TinyGPS” というライブラリを導入している [5]．また，Baudrate は 9600bps に設定し，30sec. 以上のタイミングで情報交換をしている．

9 軸センサの校正は，センサを水平状態にして行う．ジャイロ，加速度センサの offset 値，地磁気センサの感度補正値を取得する．地磁気センサの offset 値を取得するためにセンサが全方向を向くよう 8 の字に回転させ，得られた値の平均をとり，地磁気センサの offset 値を取得する．それぞれのセンサの offset 値，地磁気センサの感度補正値を用いて校正を行う．

一方，GPS モジュールは起動してから数分は正しいデータを得ることができない．TinyGPS 内で GPS モジュールの有効なデータを取得できたかどうか判別できるため，有効なデータのみを出力し，無効なデータは空白を出力するようにしている．また, 時刻は世界標準時になっていたので，日本時間に準じるよう調整している．この二つのモジュールで取得したデータを，30 秒毎にシリアル通信 (USB) で Raspberry Pi に送信する．

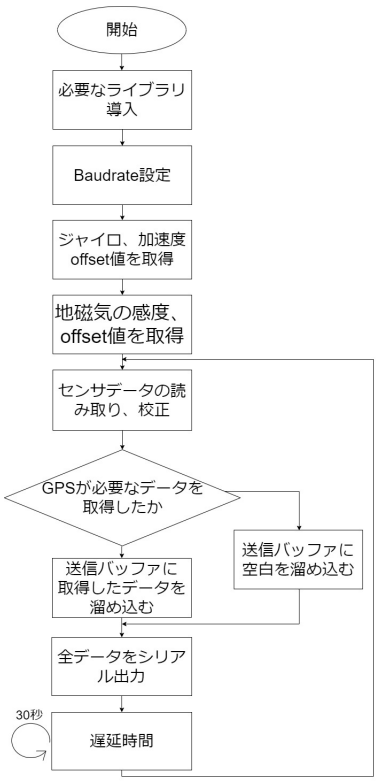

図 3: Arduino 側のフローチャート

2.2 Raspberry Pi の機能

Raspberry Pi は LoRa Private とシリアル通信 (UART) をしている．Baudrate は 9600bps である．LoRa Private の設定は Spreading Factor(SF)=12, Radio Frequency(RF)=925MHz, Band Width(BW)=125kHz に設定している．使用したモジュールは GREEN HOUSE 社の RM-92A である．送信側，受信側のフローチャートをそれぞれ図 4(a), (b) とする．送信側では，Raspberry Pi は Arduino と USB 接続，LoRa Private とは UART 接続をしているため，まずはじめにシリアルポートを開く．LoRa Private の受信機のデバイス ID を指定し，Arduino から取得したデータをバイト列に変換をして送信する．受信側では, はじめにリセットピンを HIGH 出力にする．その後シリアルポートを開き，LoRa Private が受信したデータを Raspberry Pi で読み取る．取得したデータを CSV ファイルで保存する．30 秒間隔で受信ができるはずなので，万が一 1 分以内に受信ができなかった場合は，Ping を送信し，Received Signal Strength Indicator(RSSI)，Signal Noise Ratio(SNR) を確認し，受信成功したデータと同じく CSV ファイルに保存する．

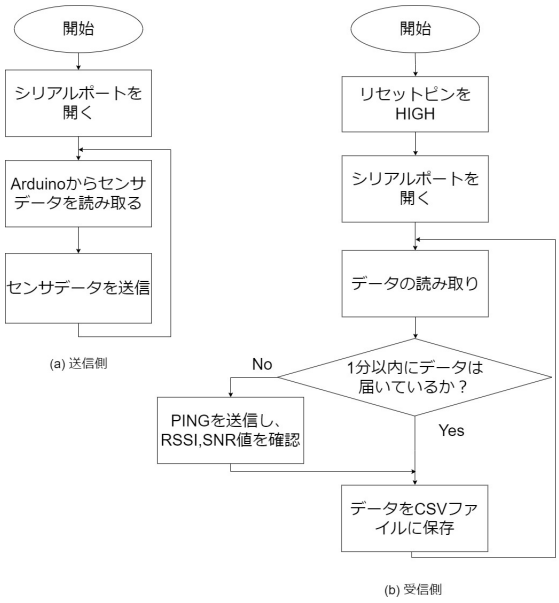

図 4: Raspberry Pi 側のフローチャート

2.3 実験結果

LoRa Private 間 30cm 間隔を空け、通信実験を行った．実際に共有し，保存した csv ファイルの中身の一部を表 1，2 に示す．

保存したデータは，ID(送信元のデバイス ID)，Datetime(GPS がデータを取得した日時)，Long(緯度)，Lat(経度)，Ax，Ay，Az(x 軸，y 軸，z 軸方向の加速度)，Gx，Gy，Gz(x 軸，y 軸，z 軸方向の角速度)，Mg(地磁気センサの値を加速度センサ値を用いて傾斜補正し

表 1: 保存した CSV ファイルのデータの一部 (その 1)

ID	Datetime	Long[dec]	Lat[dec]	Ax[G]	Ay[G]	Az[G]
2	2019/10/16 16:10:29	33.849098	132.77054	0.07	0.09	1
2	2019/10/16 16:10:59	33.849064	132.77061	0.07	0.1	1
2	2019/10/16 16:11:29	33.849007	132.77046	0.08	0.09	1.01
2	2019/10/16 16:11:59	33.849049	132.77028	0.07	0.1	1
2	2019/10/16 16:12:29	33.849072	132.77077	0.07	0.09	1

表 2: 保存した CSV ファイルのデータの一部 (その 2)

Gx[dec/s]	Gy[dec/s]	Gz[dec/s]	Mg[rad]	RSSI[dBm]	SNR[dB]
0	0.13	0.06	2.32	-117	2
0.01	0.16	0.01	2.32	-117	2
0	0.1	0.06	2.31	-117	2
0.06	0.06	0.1	2.32	-117	2
0.11	0.2	-0.06	2.31	-117	3

た際のコンパスデータ) 及び，受信した際の RSSI と SNR である．受信機側で送信機側の GPS，9 軸センサ情報を受信し，保存することで，データの共有ができたということを確認した．

また，1 分連続受信失敗した場合のときに PING を送信すると，以下のように PONG が返ってくる．”b’$RFRX,0002,RECEIVED_RPONG,0000*24 Seq:190 RSSI:-93 SNR:28 HOP:0\r\n’” この PONG により通信相手デバイスの電源 ON，OFF 状態や，RSSI，SNR を確認することができる．

3 むすび

本稿では，Raspberry Pi と Arduino を使い，LoRa Private 間で GPS と 9 軸センサ情報を送受信し，Raspberry Pi でデータを保存し，共有することができた．

今後は実際に水田用除草ロボットの複数台での位置情報を共有できることを確認する予定である．

参考文献

[1] 農林水産技術会議,「スマート農業技術の開発・実証プロジェクト」及び「スマート農業加速化実証プロジェクト」について, http://www.affrc.maff.go.jp/docs/smart_agri_pro/smart_agri_pro.htm, 2019.08.22 閲覧.

[2] Masashi SUGIMOTO, et. al, A Study for Development of Autonomous Paddy-weeding Robot System -An experimentation for autonomously straight-running based on compass-compensation-, International Journal of New Computer Architectures and their Applications, vol. 8 No. 4, pp.186-197, 2019.

[3] Masashi SUGIMOTO, Shinji TSUZUKI, Shiro URUSHIHARA, Kazunori HOSOTANI, Manabu KATO, Hitoshi SORI, Hiroyuki INOUE, An Experimental Study for Exploration-oriented Behavior in Maze-solving using Reinforcement Learning, In Proceedings of The Fifth International Conference on Electronics and Software Science ICESS2019, pp.47-53, 2019.

[4] 杉本 大志, 猪木 康弘, 竹内 貫太, 山地 駿徹, 加藤 学, 井上 浩行, 漆原 史朗, 細谷 和範, 曽利 仁, 画像認識を用いた 農作業ロボットにおける旋回動作についての実験的研究, 2018 年度精密工学会中国四国支部 -香川地方学術講演会-, pp.19-20, 2018.

[5] Arduino wisdom and gems by Mikal Hart，”Tiny GPS”，Arduiniana，http://arduiniana.org/libraries/tinygps/，2019.10.22 閲覧.