

GMPS®を用いた自己位置推定ソフトウェアの開発と OSS 公開

Development of Localization Software Using GMPS® and Release as Open Source Software

棕本 博学* 安藤 孝幸* 浦川 一雄*
Hirotaka Mukumoto Takayuki Ando Kazuo Urakawa

Abstract

This report describes a localization method based on the Global Magnetic Positioning System (GMPS) using a magneto-impedance sensor, which was developed for automated driving systems to obtain precise vehicle positions under all conditions, including snow or rain. The aim of this study is to propose a localization method which can estimate yaw angle and absolute position of a vehicle using only one sensor and double magnetic markers. Assuming that a vehicle moves in a straight line, the yaw angle can be determined by measuring the lateral deviations from the two magnetic markers. Experimental results using an electronic wheelchair and GMPS demonstrate the stability and high accuracy of the proposed method. Additionally, the software of the proposed method is released as Open Source Software, aiming to reduce the initial cost of using GMPS and to facilitate collaborative software development.

1. はじめに

当社は自己位置推定に用いるセンサシステムとして、Fig. 1. のような高感度な MI センサが複数個アレイ状に配置された磁気センサモジュール（以下、GMPS センサ）と路面に設置された磁気マーカからなる磁気マーカシステム（GMPS）を開発している^{1,2,3}。GMPS は、車両の床下に搭載した GMPS センサで、路面に設置した磁気マーカを検出することで、目標軌跡に対する車両位置の横ずれ量や車両のヨー角を算出するものであり、mm 単位の高い自己位置推定精度が特長である。

従来はバスのような大型車両の床下に GMPS センサを 2m 間隔で前後 2 本搭載し、同様に 2m 間隔で路面に敷設した磁気マーカを 2 本の GMPS センサで同時検知することで、車両のヨー角を算出していた。しかしながら、工場・倉庫物流等で使用される小型車両においては、車両サイズの制約により GMPS センサを 2m 間隔で前後 2 本搭載することができない。そのため、GMPS センサ 1 本でも拡張カルマンフィルタを用いて位置推定する手法を当社は開発した⁴。しかし、他の自己位置推定手法（GNSS や LiDAR スキ

ャンマッチング）との冗長化手段として GMPS を使うためには、目標軌跡からの相対座標ではなく地図上の絶対座標を推定する必要があった。

また、自己位置推定は GMPS センサ内部ではなく自動運転車両側のソフトウェアに属する処理であるため、従来は GMPS を導入しようとするユーザ自身が、GMPS を用いた自己位置推定ソフトウェアを作成する必要があった。そのため、GMPS を試験的に使用するだけでも開発コストがかかってしまい、ユーザが GMPS に興味があったとしても試験を実施することが難しいという問題があった。一方、ロボット分野では OSS (Open Source Software) が広く普及しており、それぞれが開発したソフトウェアを互いに利用できるようにすることで、コミュニティ全体の開発効率を向上させている。

そこで、本開発では、GMPS センサ 1 本で地図上の絶対座標を精度良く推定する手法を開発した。また、その手法を用いた自己位置推定ソフトウェアを開発し、OSS として公開した⁵。

*未来創生開発部 モビリティシステム開発室
GMPS®は愛知製鋼株式会社の登録商標

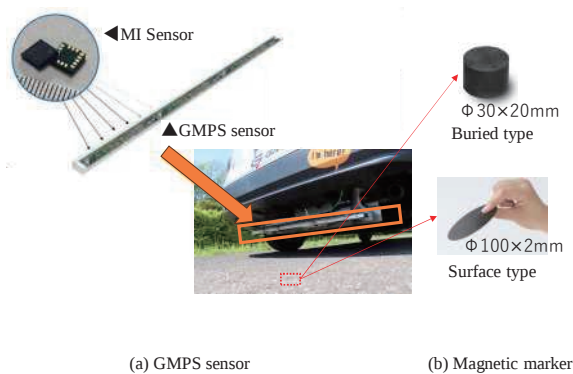


Fig. 1. Global Magnetic Positioning System (GMPS).

2. GMPS を用いた自己位置推定手法

本章では、OSS として公開した GMPS を用いた自己位置推定ソフトウェア（以下、`gps_localizer`）に沿って、GMPS センサ 1 本で地図上の絶対座標を推定する手法について説明する。2. 1 節では自己位置推定全体における `gps_localizer` の位置づけと、磁気マーカシステムの概要について説明する。2. 2 節では `gps_localizer` の前半部分である対応付けの処理について説明する。2. 3 節では `gps_localizer` の後半部分である磁気マーカ敷設座標をもとにした絶対座標の計算について説明する。

2. 1 システム概要

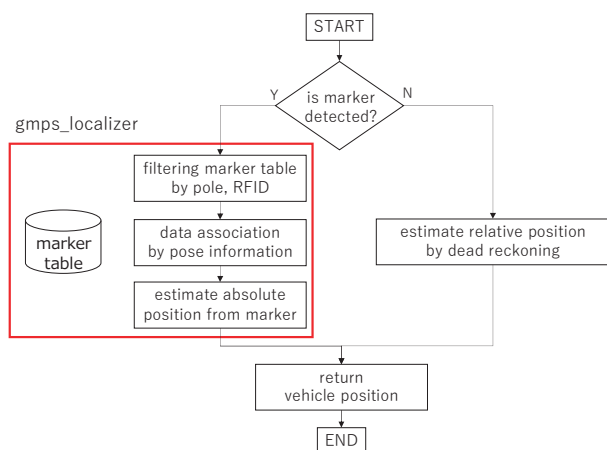


Fig. 2. Scope of `gps_localizer`.

Fig. 2 は GMPS を用いた自己位置推定の大まかな処理のフローであり、赤枠の範囲が `gps_localizer` の扱う範囲である。自己位置推定は内界センサを用いて車両の位置を算出するデッドレコニングをベースとして、外界センサを用いたスターレコニングによってその位置を補正する手法で構成されている。GMPS は外界センサに相当する

GMPS センサと、インフラに相当する磁気マーカとを用いて地図上の位置を推定するため、スターレコニングの技術に該当する。`gps_localizer` は磁気マーカ直上を通過した瞬間の絶対座標を出力する。

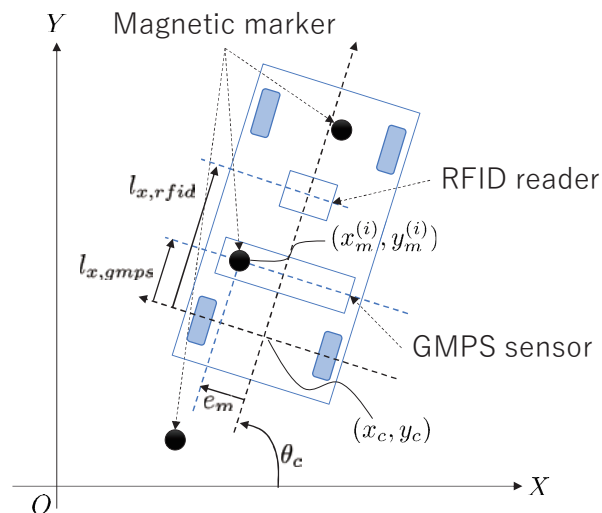


Fig. 3. Vehicle positions and yaw angle definition.

Fig. 3 は平面座標系における車両、GMPS センサ、RFID リーダ、および磁気マーカの位置関係を表したものである。なお磁気マーカは、一部に RFID タグを装着したものを使用し、RFID タグに書き込まれた情報を車両に搭載した RFID リーダで読み取れるようにしている。座標基準点における XY 座標およびヨー角を (x_c, y_c, θ_c) とし、 i 番目の磁気マーカの敷設座標を $(x_m^{(i)}, y_m^{(i)})$ とする。座標基準点から GMPS センサ中心までの縦方向のオフセットを $l_{x,gmps}$ とし、座標基準点から RFID リーダまでの縦方向のオフセットを $l_{x,rfid}$ とし、GMPS センサ中心から磁気マーカまでの横偏差を e_m とする。ただし、縦方向オフセット l_x は前進方向を正とし、横偏差 e_m は右方向を正とする。また、自動運転を開始してから現在までの走行距離を λ とする。

なお、磁気マーカの座標 $(x_m^{(i)}, y_m^{(i)})$ は、敷設後に GNSS 測量等によって事前に正確な座標を調べておき、極性や RFID タグ番号とあわせて磁気マーカテーブルを作成しておく必要がある。Table 1 に磁気マーカテーブルの例を示す。ここで、ID は磁気マーカの一意な番号であり、RFID は RFID タグに書き込まれた番号を表す。RFID の項が 0 であることは、RFID タグを含まない通常の磁気マーカであることを意味する。

Table 1. Example of table of magnetic markers

ID	X	Y	Pole	RFID
1	$x_m^{(1)}$	$y_m^{(1)}$	N	-0
2	$x_m^{(2)}$	$y_m^{(2)}$	S	1002
3	$x_m^{(3)}$	$y_m^{(3)}$	N	-0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Fig. 4 は gmps_localizer のフローチャートであり、Fig. 2 の赤枠内をより具体化したものになっている。GMPS が磁気マーカを検知してから位置情報を出力するまでには、
 (1) Table 1. の磁気マーカテーブルから正しい磁気マーカ番号を探索する対応付けの処理 (Data Association) と、
 (2) 磁気マーカの敷設座標と横偏差とから、車両の絶対座標を計算する処理 (Pose Measurement) の 2 つの処理が含まれる。以下の節では、それぞれの処理について説明する。

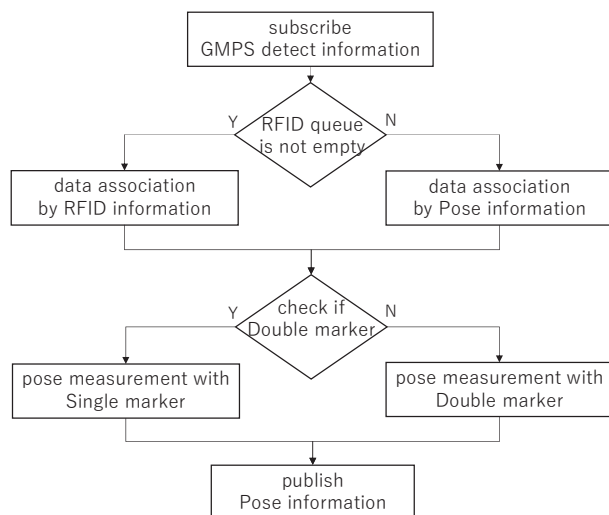


Fig 4. Flowchart of gmps_localizer.

2. 2 磁気マーカ検知と磁気マーカ番号との対応付け

磁気マーカは全て特性が同じ磁石であるため、GMPS センサだけではどの磁気マーカを検知したのか区別することができない。そのため、位置情報あるいは RFID タグ番号を用いてどの磁気マーカを検知したのかを特定する必要がある。このような、センサによる検知情報と被検知物との対応関係を特定する処理を対応付け (Data association) と呼ぶ。

本節では、Table 1. の磁気マーカテーブルを探索することで、磁気マーカ検知情報と磁気マーカ番号との対応付けを行う処理について説明する。

2. 2. 1 対応付け手法の選択

はじめに、位置情報を用いて対応付けを行うか、RFID タグ番号を用いて対応付けを行うかを条件によって選択する。ここでの条件は、GMPS センサが磁気マーカを検知する前に、RFID リーダが磁気マーカに付属する RFID タグを検知していたかどうかである。全ての磁気マーカに RFID タグが付属しているとは限らないため、対応付けは位置情報によって行うのが基本である。もし RFID タグを検知していた場合には、RFID タグ番号による対応付けを試みる。なお、実際の gmps_localizer は RFID タグ番号による対応付けに失敗した場合、位置情報による対応付けも試みる。

2. 2. 2 位置情報による対応付け

Fig. 5. は位置情報による対応付けの模式図である。青の●印は磁気マーカの予測位置を、黒の●印は実際の磁気マーカの敷設位置を表している。磁気干渉防止やコスト低減の観点から、磁気マーカは一定以上の距離間隔をとって敷設される。例えば、公道で用いられる磁気マーカの最小敷設間隔は 1m である。したがって、50cm 程度の精度で自己位置推定ができていれば、現在の自己位置から最近傍の磁気マーカを探索することで、対応付けが可能である。

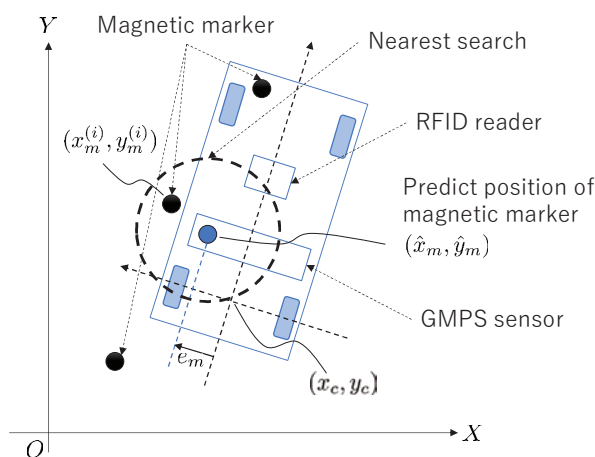


Fig 5. Data association with pose information.

具体的な処理は以下の通りである。

(1) 磁気マーカの予測位置の計算

Fig. 3. に図示されるように、磁気マーカテーブルを探索するためのクエリとなる磁気マーカ予測位置($\hat{x}_m, \hat{y}_m$)は、デッドレコニング等により推定した現在の自己位置推定

値 $(\hat{x}_c, \hat{y}_c, \hat{\theta}_c)$ からの平行移動によって次のように計算できる。

$$\hat{x}_m = \hat{x}_c + l'_{x,gmps} \cos \hat{\theta}_c + e_m \sin \hat{\theta}_c \quad (1)$$

$$\hat{y}_m = \hat{y}_c + l'_{x,gmps} \sin \hat{\theta}_c - e_m \cos \hat{\theta}_c \quad (2)$$

ここで、 $l'_{x,gmps}$ は検知遅延距離 l_d を考慮した実質的な縦方向オフセットであり、Fig. 6. に図示されるように、縦方向オフセット $l_{x,gmps}$ から検知遅延距離 l_d を差し引くことで得られる。

$$l'_{x,gmps} = l_{x,gmps} - l_d \quad (3)$$

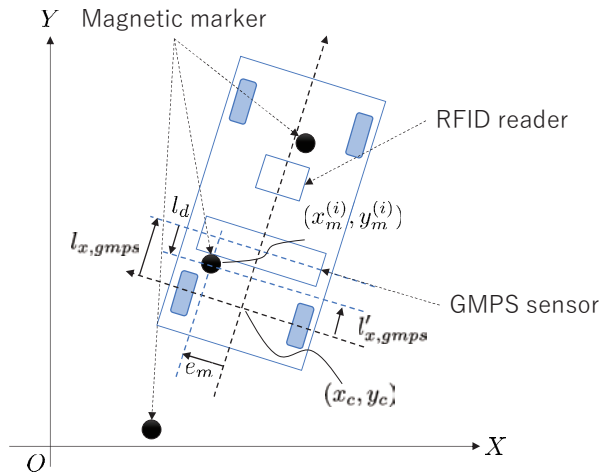


Fig. 6. Sensor delay.

(2) 磁気マーカ予測位置を用いた最近傍探索

磁気マーカ予測位置 $(\hat{x}_m, \hat{y}_m)$ との距離が最小であるような磁気マーカを Table 1. の磁気マーカテーブルから探索することで、磁気マーカ番号 i を特定する。

$$i = \arg \min_j \sqrt{(x_m^{(j)} - \hat{x}_m)^2 + (y_m^{(j)} - \hat{y}_m)^2} \quad (4)$$

(3) 距離による誤検知判定

最近傍探索で得られた i 番目の磁気マーカと磁気マーカ予測位置との距離が閾値以下であることを確認する。閾値を超えていれば誤検知として棄却する。閾値の目安は、想定される磁気マーカ敷設間隔の半分である。`gmps_localizer` のデフォルト設定は 1.0m としている。

(4) 極性による誤検知判定

最近傍探索で得られた i 番目の磁気マーカ極性と、GMPs

センサの検知情報に含まれる極性とが位置しているかを確認する。一致していなければ誤検知として棄却する。

以上により、今回の磁気マーカ検知に対応する磁気マーカ番号 i が特定され、磁気マーカ敷設座標 $(x_m^{(i)}, y_m^{(i)})$ を得ることができる。

2. 2. 3 RFIDによる対応付け

RFID タグを検知している場合には、RFID タグ番号で磁気マーカの対応付けができる。

具体的な処理は以下の通りである。

(1) RFID タグ番号と磁気マーカ検知の対応付け

2. 1. の Fig. 3. より、RFID リーダと GMPs センサとは前後方向に離れて搭載される。したがって、ある磁気マーカの RFID タグを RFID リーダが検知するタイミングと、その磁気マーカを GMPs センサが検知するタイミングとは、多少の時間差がある。そのため、まずは今回の磁気マーカ検知に対応する RFID タグ検知がどれであるかを特定し、RFID タグ番号と磁気マーカの検知情報との対応付けを行う必要がある。

ある磁気マーカの RFID タグを RFID リーダで検知してから、その磁気マーカを GMPs センサで検知するまでに車両が走行する距離は、各センサとの縦方向オフセットの差分 $l_{x,rfid} - l_{x,gmps}$ に等しい。よって、RFID タグを RFID リーダで検知した際の走行距離を λ_{rfid} とし、磁気マーカを GMPs センサで検知した際の走行距離を λ_{gmps} とすると、

$$\lambda_{gmps} - \lambda_{rfid} = l_{x,rfid} - l_{x,gmps} \quad (5)$$

が成り立つかどうかによって、今回の磁気マーカ検知に対応する RFID タグ検知を特定することができる。式(5)によって特定された RFID タグ検知情報に含まれる RFID タグ番号が、磁気マーカテーブル探索のクエリとなる。

(2) 磁気マーカテーブルの探索

上記 (1) の処理によって探索した RFID タグ番号と同じ番号を持つ磁気マーカを、Table 1. の磁気マーカテーブルから探索する。処理としては単純に番号の一致を調べればよい。これにより RFID タグ番号をもとに磁気マーカ番号 i を特定できる。

(3) 距離と極性による誤検知判定

2. 2. 2. の位置情報による対応付けと同様に、上記

(2) で探索した i 番目の磁気マーカについて、磁気マーカ予測位置との距離や極性の一致を確認する。

以上により、今回の磁気マーカ検知に対応する磁気マーカ番号 i が特定され、磁気マーカ敷設座標 $(x_m^{(i)}, y_m^{(i)})$ を得ることができる。

なお、現状の `gps_localizer` は磁気マーカより先に RFID タグを検知することを前提とした処理構造になっている。そのため、原則として RFID リーダを GPS センサよりも前方に搭載する必要がある。

2. 3 観測座標の計算

本節では、2. 2 節で特定した磁気マーカの敷設座標 $(x_m^{(i)}, y_m^{(i)})$ と横偏差 e_m とから地図上の絶対座標を推定する処理を説明する。なお、以下では磁気マーカが 1 個の場合を単体 (Single) マーカ、磁気マーカが 2 個の場合を二連 (Double) マーカと呼称している。単体マーカの場合はヨー角を推定することができないため XY 座標のみを推定するが、二連マーカの場合は XY 座標に加えてヨー角も推定する。

2. 3. 1 観測座標の計算手法の選択

詳細は後の 2. 3. 3 で説明するが、二連マーカを用いてヨー角を推定するためには、二つの磁気マーカを直進しながら検知していることが前提として必要である。そのため、手前の磁気マーカを検知してから次の磁気マーカを検知するまでの走行が直進とみなせるかどうかを確認する。直進とみなせる場合は二連マーカの処理へ、そうでない場合は単体マーカの処理へ進む。

直進かどうかは以下 2 つの条件によって判定している。

(1) ヨー角の変化量

$i - 1$ 番目の磁気マーカを検知したときのヨー角を θ_{i-1} とし、 i 番目の磁気マーカを検知したときのヨー角を θ_i とすると、その差分 $\Delta\theta_i$ は

$$\Delta\theta_i = |\theta_i - \theta_{i-1}| \quad (6)$$

である。 $\Delta\theta_i$ が閾値以下であれば車両は直進していたとみなす。

なお、操舵角やヨーレートを用いることでも同様の判定ができる。ヨー角の差分を用いる方法は、処理の簡易性と入力数を増やさずに済むことに利点がある。

(2) 磁気マーカ間の距離

ヨー角を計算するには 2 つの磁気マーカ間の距離は離れている方が望ましい。しかし、距離が離れるほど細かい修正舵を行っている可能性が高くなるため、ある程度短い距離であることを条件にしている。 $i - 1$ 番目の磁気マーカを検知してから i 番目の磁気マーカを検知するまでの走行距離を L_i とすると、

$$L_i = \lambda_{i,gmps} - \lambda_{i-1,gmps} \quad (7)$$

である。 L_i が閾値以下であるかを判定する。`gps_localizer` のデフォルト設定は 2.5m としている。

2. 3. 2 単体マーカ

2. 3. 1 の判定で二連マーカとみなすことができない場合は、単体マーカとして XY 座標のみを推定する。

具体的な処理は以下の通りである。

(1) ヨー角の取得

単体マーカの場合はヨー角の推定ができないため、ヨー角を外部から入力する必要がある。2. 2. 2 の位置情報による対応付けの場合と同様に、デッドレコニング等により推定した現在の自己位置推定値 $(\hat{x}_c, \hat{y}_c, \hat{\theta}_c)$ を用いる。

$$\theta_c = \hat{\theta}_c \quad (8)$$

(2) 磁気マーカから座標基準点までの平行移動

ヨー角 θ_c が得られれば、Fig. 3. のように、縦方向のオフセット $l_{x,gmps}$ と横偏差 e_m を用いて平行移動することで、マーカ敷設位置から現在の自己位置を以下のように計算することができる。

$$x_c = x_m^{(i)} - l'_{x,gmps} \cos \theta_c - e_m \sin \theta_c \quad (9)$$

$$y_c = y_m^{(i)} - l'_{x,gmps} \sin \theta_c + e_m \cos \theta_c \quad (10)$$

ここで、 $l'_{x,gmps}$ は検知遅延距離 l_d を考慮した縦方向オフセットであり、式(3)で与えられる。式(9), (10)を式(1), (2)と比較すると、平行移動の向きが反転しているだけで本質的には同じ計算をしていることが分かる。

なお、式(9)から明らかのように、検知遅延時間の補正においては縦方向の検知遅延距離 l_d のみを考慮しており、検知遅延時間内のヨー角の変化を考慮していない。検出遅延時間を t_d とし、車両速度を v_x とすると、 $t_d = l_d/v_x$

である。高速走行中であれば、磁気マーカの検知遅延時間 t_d は小さくなるため、ヨー角はほとんど変化せず問題とはならない。しかし、ロボットのような低速車両であれば検知遅延時間 t_d が大きくなってしまうため、その時間内の走行によるヨー角の変化を無視できない場合がある。そのため、カーブ中に敷設した磁気マーカを用いて自己位置推定を行う際には注意する必要がある。

2. 3. 3 二連マーカ

単体マーカの場合はヨー角の推定をすることができないが、二連マーカの場合は 2 つの磁気マーカ間を直進したという仮定の下でヨー角を推定することができる。

具体的な処理は以下の通りである。

(1) ヨー角の計算

Fig. 7. は二連マーカを直進して検知する模式図である。 e_1 は $i-1$ 番目の磁気マーカに対する横偏差、 e_2 は i 番目の磁気マーカに対する横偏差であり、この 2 つの磁気マーカ間の距離を L とする。この磁気マーカ間距離 L は式(7)のように走行距離の差分から求めることもできるが、ここでは磁気マーカの敷設座標を使って計算する。

$$L = \sqrt{(x_m^{(i)} - x_m^{(i-1)})^2 + (y_m^{(i)} - y_m^{(i-1)})^2} \quad (11)$$

Fig. 7. に示すように、車両のヨー角 θ_c は、 $i-1$ 番目の磁気マーカと i 番目の磁気マーカとを結んだ直線方向 θ_{marker} と、横偏差 e_1, e_2 によって規定されるズレ量 $\Delta\theta$ に分解することができる。よって、

$$\theta_c = \theta_{marker} + \Delta\theta \quad (12)$$

である。ただし、

$$\theta_{marker} = \tan^{-1} \left(\frac{y_m^{(i)} - y_m^{(i-1)}}{x_m^{(i)} - x_m^{(i-1)}} \right) \quad (13)$$

$$\Delta\theta = \sin^{-1} \left(\frac{e_2 - e_1}{L} \right) \quad (14)$$

である。

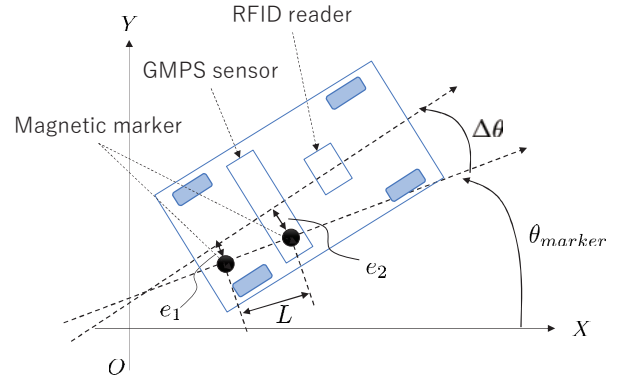


Fig. 7. Yaw estimation using double marker.

(2) 磁気マーカから座標基準点までの平行移動

ヨー角を式(12), (13), (14)から得られるものを使う以外は、単体マーカの場合と同じ処理になる。式(9), (10)を再掲する。

$$x_c = x_m^{(i)} - l'_{x,gmps} \cos \theta_c - e_m \sin \theta_c \quad (9)$$

$$y_c = y_m^{(i)} - l'_{x,gmps} \sin \theta_c + e_m \cos \theta_c \quad (10)$$

単体マーカの場合と同様に、二連マーカにおいても検知遅延時間内のヨー角の変化を無視している。また、特に式(14)によるヨー角のズレ量の計算において、直進の仮定が崩れると誤差が生じる。そのため、2. 3. 1の判定処理において直進中に検知された磁気マーカであることを確認している。

3. 実車実験

本章では、GMPS センサを搭載した電動車椅子（以下、実験車両）を用いて自動運転を行った実験（以下、本実験）について説明する。

Fig. 8. は実験車両の概観である。電動車椅子の底面には、GMPS センサが座標基準点である後輪軸中心から前方 200mm の位置に、RFID リーダが座標基準点である後輪軸中心から前方 100mm の位置に搭載されている。2. 2. 3 で述べた通り、gps_localizer は磁気マーカよりも RFID タグを先に検知することを前提とした処理構造になっているため、本来であればこのような配置は望ましくない。しかし、本実験で使用した磁気マーカの検知遅延距離が 100mm であるため $l_{x,rfid} - (l_{x,gmps} - l_d) = 0$ となること、および本実験で使用した RFID リーダの検知範囲が 100mm 四方程度であることから、設計上は RFID タグの検知の方が早い。実際に、本実験において RFID タグの検知が磁気マーカの

検知に遅れることは稀であった。

3. 1 実験条件

Fig. 9. は実験環境の写真である。当社技術センターの一面に、直径 60mm の貼付型磁気マーカを敷設してテストコースを準備した。見通しのきく位置にトータルステーションを設置し、実験車両の背もたれ上部に設置したプリズムを観測した。なお、プリズムは座標基準点である後輪軸中心に対し後方 176mm の位置に取り付けている。

本実験では、自己位置推定により求めた後輪軸中心の座標をプリズムの位置に変換した座標値と、トータルステーションにより観測したプリズムの座標値とを比較することにより、GPS を用いた自己位置推定の精度を検証している。

3. 2 実験結果

本節では、磁気マーカによる位置補正を行わずデッドレコニングのみで自己位置推定および自動運転を行った実験結果(a)と、磁気マーカによる位置補正を用いて自己位置推定および自動運転を行った実験結果(b)とを示す。

3. 2. 1 磁気マーカによる位置補正無し

Fig. 10. は磁気マーカによる位置補正を行わず、デッドレコニングのみで 4 周した際の自己位置推定結果の例である。青の実線は自己位置推定（デッドレコニングのみ）により推定した実験車両の位置を、赤の○印と破線はトータルステーションによる観測を、黒の△は床面に敷設した磁気マーカの位置を表している。車両は(0, 0)の位置から X 軸正方向に発進し、一辺が 3.5m の正方形に収まる直線と円弧からなるコースを、磁気マーカを検知しながら走行するように設定している。Fig. 10. より、デッドレコニングで推定した青線は安定した走行奇跡を描いているが、トータルステーションの赤破線は周回を重ねるごとに少しずつ斜めに傾いている。これは、現実の実験車両はトータルステーションが示すように徐々に走行位置がずれてしまっているが、デッドレコニングのみでは自己位置を正しく推定できていないということを意味する。車輪速を用いたデッドレコニングのみによって完璧に自己位置を推定することは困難であり、安定した自動運転のためにはスターレコニングによって位置を補正することが必要である。

3. 2. 2 磁気マーカによる位置補正有り

Fig. 11. は磁気マーカによる補正を用いて自動運転で 10 周した際の自己位置推定結果の例である。青線は自己位置推定（GPS およびデッドレコニング）により推定した実験車両の位置を、赤の○印と破線はトータルステーションによる観測を、黒の△印は磁気マーカの敷設位置を表す。Fig. 10. の場合と異なり、今回は磁気マーカによる位置補正を行ったことで安定して周回を重ねることができたため、トータルステーションの赤破線の軌跡が重なっていることが分かる。また、例えば $(x, y) = (-1.0, +1.5)$ のところで青線の自己位置推定がトータルステーションの参照位置と乖離しているが、磁気マーカを検知して座標が変化するとトータルステーションの参照位置と一致する。これは、磁気マーカ間をデッドレコニングで走行している内に自己位置にズレが生じるものの、磁気マーカを検知して位置を補正することにより、正しい位置を推定できていることを意味する。

Fig. 12. は、Fig. 11. と同じ走行データにおいて、GPS センサで検知した磁気マーカからの横偏差 e_m をプロットしたものである。横軸は磁気マーカ番号、縦軸は横偏差である。Fig. 12. より、初期位置の影響で 1 周目の横偏差は他の周回の軌跡から外れているが、2 周目から 10 周目の横偏差はほとんど重なっていることが分かる。これは、実験車両が 2 周目以降の各周回において同じ軌跡で走行していたことを意味しており、Fig. 11. のトータルステーションの軌跡が重なっていることと符合する。このような再現性の高さが、GPS による自己位置推定の重要な特徴の一つである。

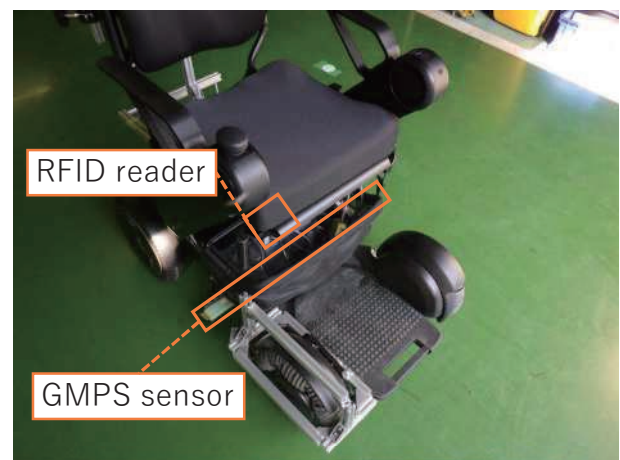


Fig. 8. Test vehicle.

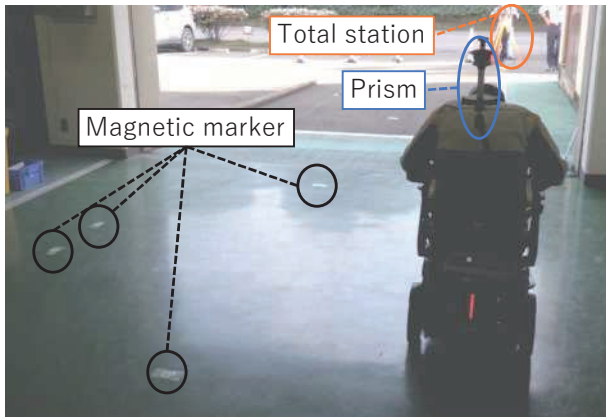


Fig. 9. Experiment field.

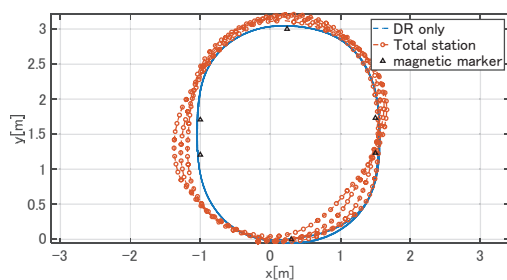


Fig. 10. Experiment (a) only dead reckoning.

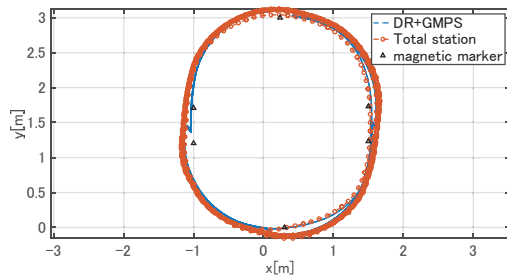


Fig. 11. Experiment (b) GMPS and dead reckoning.

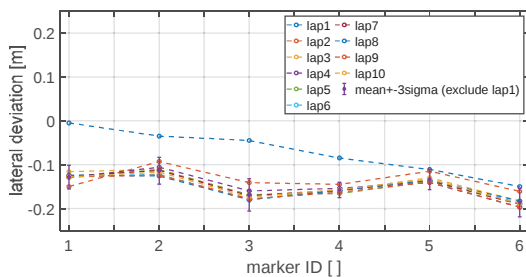


Fig. 12. Lateral deviation measured by GMPS at experiment (b).

4. OSS 公開

本章では、前章までで述べた GMPS 自己位置推定のためのソフトウェアである `gmps_localizer` を OSS として公開した経緯を紹介する。

GMPS センサは磁気マーカに対する横偏差 e_m を CAN 等で出力するが、その横偏差を受信するインターフェース部分のソフトウェア（以下、Driver）及び横偏差から自己位置に変換する部分のソフトウェア（以下、Localizer）は、ユーザ自身が作成していた。そのため、ユーザの GMPS を導入するためのソフトウェア開発に係る初期コストが高いという問題があった。

一方で、Fig. 13. に示すように、ロボット分野においては、各センサメーカーが自社のセンサを使用するための Driver を開発し、OSS として公開することがよく行われている。さらに、LiDAR 等のセンサにおいては自己位置推定のための Localizer についても OSS が公開されている。したがって、ユーザはこれらの LiDAR 用 Driver および Localizer をダウンロードして簡単なセットアップをするだけで、すぐにセンサを使い始めることができる。

そこで、当社は GMPS センサのための Driver および Localizer である `gmps_driver`、`gmps_localizer` の2つのソフトウェアを開発し、2024 年 9 月に OSS として公開した⁵⁶⁾。`gmps_driver` は CAN のフレームデータを送受信する役割、`gmps_localizer` は本報告で述べてきた自己位置推定の役割を担っている。これらのソフトウェアは web から誰でも無料でダウンロードできるようになっている。これにより、ユーザは GMPS の導入にあたって自らソフトウェア開発を行う必要がなく、ダウンロードしたソフトウェアを使うだけで済むため、GMPS 導入のためのソフトウェア開発に係る初期コストを大きく低減できる。

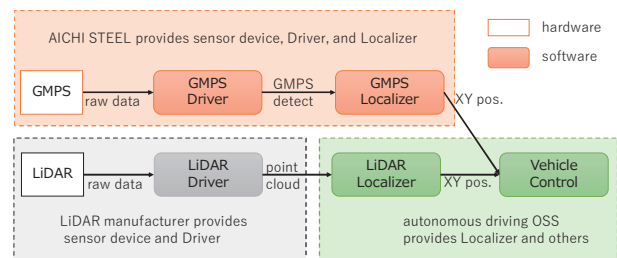


Fig. 13. Software supply chain.

5. おわりに

本開発では、GNSS センサ 1 本を用いて地図上の絶対座標を精度良く推定する手法を提案した。実験結果から、GNSS を用いた自己位置推定は高い精度だけでなく、高い再現性も有していることが示された。さらに、開発したソフトウェアを OSS として公開することで、ユーザはソフトウェアをダウンロードして簡単なセットアップを行うだけで GNSS センサを使用することが可能となり、GNSS 導入のためのソフトウェア開発に係る初期コストを大きく低減することができる。

自動運転を含むロボット分野では OSS の普及が進んでおり、開発を効率化するため OSS をベースに自動運転車両を構築しているユーザは少なくない。そのようなユーザにとっては、必要なソフトウェアが OSS で提供されているかどうかはセンサ選択における重要な指標となる。

当社は、高精度な自己位置推定を低コストで実現可能な GNSS センサや磁気マーカといったハードウェアから必要なソフトウェアまでを一貫して提供することで、自動運転の社会実装に貢献することを目的として、開発を続けて行く。

6. 謝辞

OSS として公開した `gnss_driver`, `gnss_localizer` の両ソフトウェアの開発においては、埼玉工業大学自動運転技術開発センターの渡部大志教授ならびに崔英泰殿の多大なるご支援をいただいた。ここに深い感謝の意を表する。

参考文献

- 1) M. Yamamoto, T. Nagao, and H. Aoyama, “Ultra-Low field magnetic guidance system,” in Proc. 23rd ITS World Congress (2016), Australia.
- 2) M. Yamamoto, T. Nagao, and H. Aoyama, “Ultra-low field magnetic guidance system operatable in harsh weather conditions,” in Proc. 26th ITS World Congress (2019), Singapore.
- 3) T. Ando, H. Mukumoto, K. Aoki, S. Okazaki, T. Nagao, H. Aoyama, M. Yamamoto, K. Nakano, “Localization using global magnetic positioning system for automated driving bus and intervals for magnetic markers,” *IEEE Trans. Intell. Veh.* (2023), vol.8, no.1, pp.878-887.
- 4) T. Ando, H. Mukumoto, S. Okazaki, T. Nagao, H. Aoyama, M. Yamamoto, “Localization Using GNSS with Extended Kalman Filter for Automated Driving Systems,” *愛知製鋼技報* (2023), No.39.
- 5) H. Mukumoto, “aichisteel-gnss/gnss_localizer: ROS2 package to estimate pose by GNSS sensor,” https://github.com/aichisteel-gnss/gnss_localizer, (2024).
- 6) H. Mukumoto, “aichisteel-gnss/gnss_driver: ROS2 driver for GNSS sensor,” https://github.com/aichisteel-gnss/gnss_driver, (2024).