



パーティクルフィルタによる自己位置同定とロバスト制御を組合せた果樹園におけるUGV巡回走行

UGV orchard circuit runs made possible by a combination of particle filter-enabled self-position identification and robust control system

石山 健二 神谷 剛志 深尾 隆則 倉舗 圭太

Abstract

The automation of agricultural work through the use of UGVs (Unmanned Ground Vehicles) has promise as a way to save labor. Operating on uneven or semi-natural terrain such as farm and orchard land requires technology to ensure that precise self-position identification is achieved even at times of sensor signal loss or noise and to ensure that the vehicle will run its prescribed path with certainty even when subjected to disturbance from uneven ground. In this paper we report on a development project that resulted in stable performance of autonomous runs of a circuit route in an orchard using a guidance system consisting of a combination of particle filter-enabled self-position identification and a robust control method based on a non-linear theory with course regeneration.

要旨

UGV(Unmanned Ground Vehicle)による農作業の自動化、省力化が期待されている。農地などの不整地・半自然環境でUGVを運用するには、センサ情報損失やノイズが発生しても正確に自己位置を認識し、路面から外乱を受けても確実に目標経路に追従する技術が必要になる。本研究ではパーティクルフィルタによる自己位置同定および非線形理論に基づくロバストな制御手法と経路再生成を組合せ、果樹園で安定した巡回走行を実現したのでその内容を報告する。

1 はじめに

わが国における農業従事者の減少および高齢化は深刻である。UGVによる農作業の自動化、省力化はその解決策の1つとして期待される。本研究は果樹園における生育管理、農薬散布、収穫などの作業自動化の基盤となるUGVの環境認識と制御システムに関するものである。

農業用UGVが自動走行するには正確に自己位置を認識する必要があり、主にRTK-GPS(Real Time Kinematic-GPS)やIMU(Inertial Measurement Unit)、オドメトリを用いる手法が提案されている。しかしRTK-GPSやIMUは高価であり、IMU、オドメトリは誤差が累積する問題がある。木が立ち並ぶ果樹園など、GPS情報が信頼できないケースもある。

本研究では2次元レーザレンジファインダ(以下LRF)とオドメトリ(車速、操舵角)、事前に用意した簡易な木のマップを組合せ、マーカなどにより自然環境に手を加えることなく自己位置を推定する。この場合計算手法として、例えば三角測量と拡張型カルマンフィルタ(以下EKF)を組合せる手法が考えられる。EKFは誤差の正規性(ガウス型)を前提としており、センサ情報とその分布に従って安定して取得できれば良い精度が得られる。しかし途中で情報が正常に取得できない、あるいは外れ値が生じるとその後の

推定がうまくいかない場合がある。つまり初期状態(別の言い方をすると前計算ステップの結果)がある程度正しく収束している必要がある。

本研究で対象とする果樹園では、真直ぐでない幹、垂下った枝葉、マップに存在しない障害物、路面傾斜・凹凸による姿勢変動などにより、常に正しい情報を取得できるとは限らない。このような環境を考慮しパーティクルフィルタ(以下PF)とマップマッチングを組合せて自己位置同定を行う。PFは確率論に基づく時系列フィルタの1つであり、計算機の性能向上を背景として様々な分野で成果を挙げている。また凹凸や滑りといった外乱に対応した経路追従を行うため、非線形理論に基づくロバストな制御手法および目標経路へなめらかに復帰する経路再生成法を採用する。

2 自己位置同定

PFは時系列ベイズフィルタの1つである。ベイズフィルタは直接観測できない対象システムの状態の確率分布を推定する。センサからの観測値 z_t が得られた時、事後の条件付確率分布 $p(x_t | z_t)$ は次式で再帰的に推定できる。

$$p(x_t | z_t) = \alpha \cdot p(z_t | x_t) \int p(x_t | x_{t-1}) p(x_{t-1} | z_{t-1}) dx_{t-1} \quad (2-1)$$

ここで $p(z_t | x_t)$ は観測モデルであり、観測値 z_t は同時刻の状態 x_t のみに依存するという性質を持つ。 $p(x_t | x_{t-1})$ は運動モデルであり、状態 x_t は直前の状態 x_{t-1} のみに依存するという性質を持つ。 α は正規化するための係数である。しかし一般に確率分布を解析的に表現するのは不可能である。また多次元ベクトルによる積分も問題を困難にする。PFは確率分布をパーティクル(サンプル値)の集合で近似し、計算機での処理を容易にしたものである。

パーティクル集合内で i 番目のものを $x^i = \{x, y, yaw\}^T$ と定義する。ベクトル要素はそれぞれ固定座標系での車両 x, y 座標、車両 yaw 角である。実装手順は以下の通りである。

- (1) 初期パーティクル集合の生成
- (2) 予測ステップ: 次式の車両運動モデルに従って各パーティクルを移動させる。

$$\dot{x} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2-2)$$

ここで v は車速、 ω は車両 yaw 角速度である。

- (3) 観測モデル(マップマッチング)に基づき、各パーティクルの尤度(確からしさ)を計算する。詳細は 2.1 で説明する。

- (4) 尤度の高い領域でサンプル数が多く、低い領域で少なくなるよう、パーティクルを選択する。
 - (5) 選択されたパーティクル集合を基に、乱数によるノイズを加えて新たな集合を生成する。
- (2)～(5)を繰り返す。図1に処理のイメージを示す。

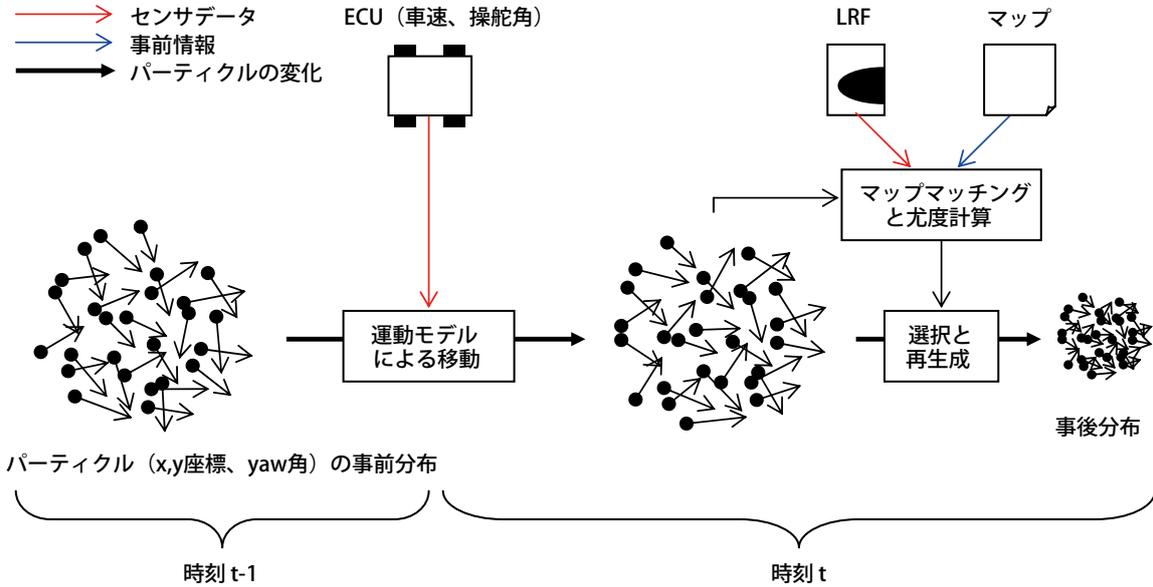


図1 パーティクルフィルタによる自己位置同定の流れ

2.1 マップマッチングによる尤度計算

事前に果樹園のマップを準備する。マップは幹の高さ約60cm付近の中心座標で定義される。つまり木を点とみなし、その本数分の x, y 座標(固定座標系)の並びで表現する。レーザ点群を距離の近い物同士でクラスタリング(図2左上)し、レーザクラスタと各パーティクル x^i 目線でのマップ(図2左下)とを比較して x^i の尤度とする(図2右)。

具体的には、マップ内で定義される各木に対して最近傍クラスタを見つけ、その距離を用いて信頼度を計算する。マップ内全ての木に対して信頼度の総和を取り、さらに全てのパーティクルに対して尤度の総和が1となるよう正規化する。

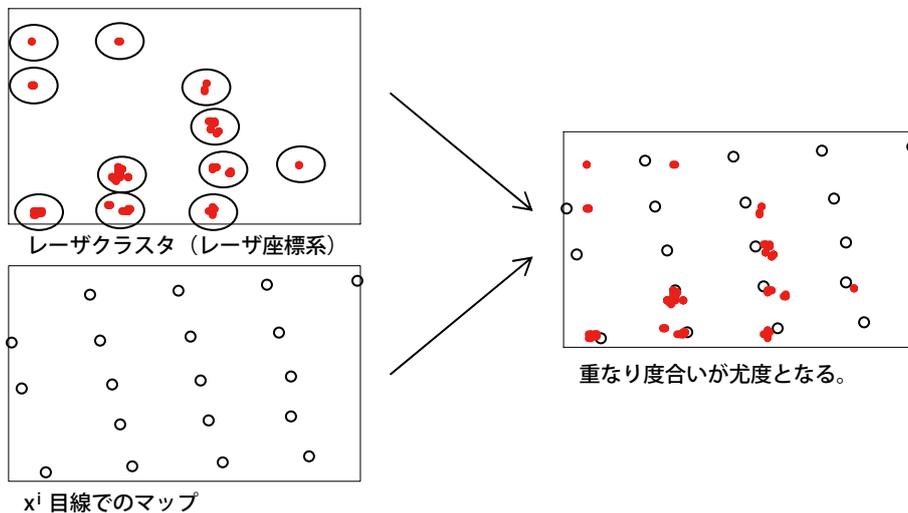


図2 マップマッチング

2.2 従来手法との比較

今回評価のために三角測量とEKFを組合せた手法と比較した。これはマップに対応するクラスタを抽出後、全ての2個の組合せに対して三角測量を行い、各組合せの信頼度による重み付き平均を取る。その結果とオドメトリ情報をEKFに入力し自己位置を推定するものである。使用するマップとレーザ点群をクラスタリングする処理は同じである。比較結果は4章で述べる。

3 制御

本研究では前輪操舵車両(4輪バギー)の経路追従問題に逆最適制御を適用した。

3.1 逆最適制御

最適制御では次式で表されるシステムに対して、(1)(2)の条件を満たすフィードバックコントローラ $u(x)$ を設計する。

$$\dot{x} = f(x) + g(x)u \quad (3-1)$$

(1) $u(x)$ は系を目標状態に漸近収束させる。

(2) $u(x)$ は次の評価関数 J を最小化する。

$$J = \int_0^{\infty} \{ l(x) + u^T R(x) u \} dt \quad (l(x) \geq 0, R(x) > 0) \quad (3-2)$$

しかし $u(x)$ を求めるにはHamilton-Jacobi-Bellman方程式を解く必要があり、これは一般に困難である。

逆最適制御は最初に安定化フィードバックコントローラを設計し、これによりある評価関数が最適化されることを示すという、通常とは逆の手順を取るものである。またこのようにして得られたコントローラは sector margin $(1/2, \infty)$ と呼ばれる安定余裕を持ち、車両のモデル化誤差に対してロバスト性を持つ。不整地における凹凸や車輪の滑りを車両モデルの変動とみなせばこの特性は有用である。

【定義1】sector margin

図3に示す外乱を受ける閉ループ系 (H, K, Δ) が、 $diag \{ \gamma_1(\cdot), \dots, \gamma_m(\cdot) \}$ の形で表されるどんな外乱 Δ に対しても大域的に漸近安定である時、非線形フィードバックシステム (H, K) は sector margin (α, β) を持つ。ここで $\gamma_i(\cdot)$ は (α, β) に属する局所的にLipschitzで静的な非線形関数である。

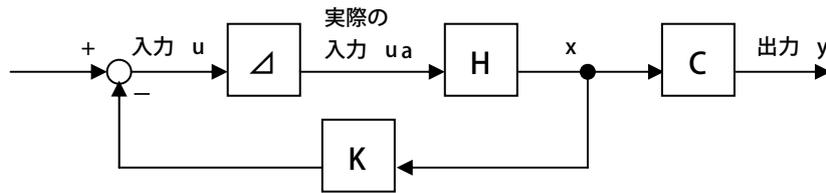


図3 入力の不確かさを含む非線形フィードバックシステム

逆最適制御によるコントローラ設計ではControl Lyapunov Function (CLF)を見つけることがポイントとなる。

【定義2】Control Lyapunov Function

全ての $x \neq 0$ に対して次の関係を満たす、滑らかで正定かつ放射状に非有界な関数 $V(x)$ はシステム(3-1)に対するControl Lyapunov Functionとなる。

$$L_g V(x) = 0 \Rightarrow L_f V(x) < 0 \tag{3-3}$$

ここで、 $L_g V(x)$, $L_f V(x)$ はLie微分 $g \frac{\partial}{\partial x} V(x)$, $f \frac{\partial}{\partial x} V(x)$ である。

この時、次の入力

$$u = -p(x) b(x) \tag{3-4}$$

は、システムを漸近安定化し、かつ次の評価関数を最小化する。

$$J = \int_0^{\infty} \left\{ \frac{1}{2} p(x) b^T(x) b(x) + \frac{1}{2 p(x) u^T u} \right\} dt \tag{3-5}$$

ここで

$$p(x) = \begin{cases} c_0 + \frac{a + \sqrt{a^2 + (b^T b)^2}}{b^T b} & (b \neq 0) \\ c_0 & (b = 0) \end{cases} \tag{3-6}$$

$$a(x) = L_f V(x) \tag{3-7}$$

$$b^T(x) = L_g V(x) \tag{3-8}$$

$c_0 > 0$ である。

3.2 前輪操舵車両モデルへの逆最適制御適用

図4に車両モデルを示す。横加速度が大きくなければ左右輪の運動特性はほぼ同じなので、それらを集約し2輪で表現する。このモデルの運動方程式は次式で表される。

$$\frac{d}{dt} \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (3-9)$$

$$\omega = \frac{v}{L} \tan \phi \quad (3-10)$$

ここで、(x,y)は固定座標系での車両位置、 θ は車両yaw角、 v は車速、 ω は車両yaw角速度、 ϕ は操舵角、 L はホイールベースであり、車輪の滑りは考慮しない。計算により得られる制御入力を(v, ω)、実際に出力されるものを(v_a, ω_a)とおくと以下の関係がある。

$$\begin{bmatrix} v_a \\ \frac{v_a}{L} \tan \phi_a \end{bmatrix} = \begin{bmatrix} R_v & 0 \\ 0 & L/L' \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (3-11)$$

ここで、 $v, v_a \neq 0$ とすると、 $R_v = v_a / v$ であり、 L' は次式で表される。

$$L' = \frac{L}{R_v} \frac{\tan \phi}{\tan \phi_a} \quad (3-12)$$

式(3-11)は定義1において、 $\gamma_1 = R_v, \gamma_2 = L / L'$ と置いたものである。したがって、 $1/2 < R_v < \infty, 0 < L' < 2L$ が満たされていれば安定性が保証される。車両yaw角速度の不確かさをホイールベースの変動に置換えれば、本制御手法は凹凸や車輪の滑りに対するロバスト性を持つ。ただし定義により R_v, L は静的でなければならず、急変する場合などは安定性が保証されるとは限らない。

経路追従問題において、実車両は目標経路上を移動する仮想の参照ロボットを追うように制御される(図5)。参照ロボットの運動モデルは、実車同様次式で表される。添字 r は参照ロボットであることを示す。

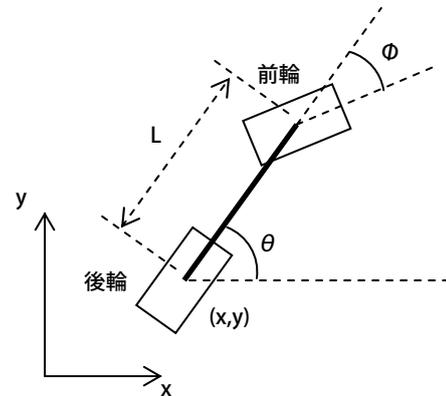


図4 前輪操舵モデル

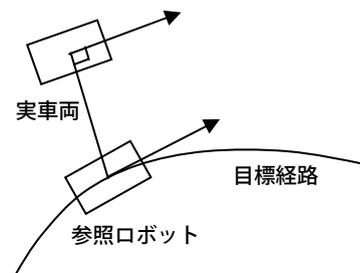


図5 経路追従における参照ロボットの概念

$$\frac{d}{dt} \begin{bmatrix} x_r \\ y_r \\ \theta_r \end{bmatrix} = \begin{bmatrix} \cos \theta_r & 0 \\ \sin \theta_r & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} \quad (3-13)$$

この時、参照ロボットと実車の相対誤差を次式で定義する。

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos \theta_r & \sin \theta_r & 0 \\ -\sin \theta_r & \cos \theta_r & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - x_r \\ y - y_r \\ \theta - \theta_r \end{bmatrix} \quad (3-14)$$

e_1, e_2, e_3 はそれぞれ実車に対する前後、左右、yaw角の誤差を表す。式(3-14)を時間で微分し、

$$h(z) = \frac{z}{1+z^2} \quad (3-15)$$

$$h' = dh / dz \quad (3-16)$$

$$\tilde{e}_3 = e_3 + h(e_2 v) \quad (3-17)$$

による変換を行う。また参照ロボットを実車の真横に置くことにより、 $e_1=0$ とする。すると誤差に関する次の方程式が得られる。

$$\frac{d}{dt} \begin{bmatrix} e_2 \\ \tilde{e}_3 \end{bmatrix} = \begin{bmatrix} v \sin(\tilde{e}_3 - h(e_2 v)) \\ -\omega_r + h'\{v^2 \sin(\tilde{e}_3 - h(e_2 v)) + e_2 v\} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \omega \quad (3-18)$$

これは式(3-1)のアフィンシステムの形式となっており、次式で表すVはCLFとなる。

$$V(e_2, \tilde{e}_3) = \frac{1}{2} k_2 e_2^2 + \frac{1}{2} k_3 \tilde{e}_3^2 \quad (3-19)$$

ここで、 $k_2, k_3 > 0$ である。

これを式(3-4),(3-6),(3-7),(3-8)に適用して得られるコントローラは誤差 e_1, e_2, e_3 を0に漸近させる。

3.3 経路再生成

目標経路との誤差が増大した時、そこへ復帰するための修正軌道を、走行距離に対して区分ごとに転

舵速度が一定(速度一定の場合)となるクロソイド曲線を組合せて生成する(図6)。修正軌道上に沿って参照ロボットを動かすことにより過渡状態を指定する。これによりアクチュエータへの負荷変動を小さくし、挙動を安定化することができる。

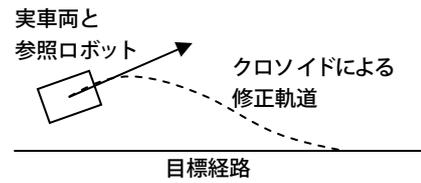


図6 経路再生の概念

4 実験内容と結果

実験に用いたUGV(図7)は2次元LRFとGPS受信機、制御用の小型PC(Intel Core2Duo, 1GB RAM)を搭載する。LRFは最大検出距離30m、スキャン範囲270deg、角度分解能0.25deg、スキャン周期25msecである。GPSは自己位置同定の精度を評価する目的のみに使い、基準局を設置しDGPSによる補正を行った。ボンネット上に生育調査カメラを搭載するが本報告では触れない。



図7 実験車両

4.1 果樹(洋梨)園での巡回走行実験

図8に示すマップと目標経路を予め用意する。経路は6本の直線で構成されており、次の直線への移行部分は一定操舵角による旋回でつなぐ。設定車速1m/sec~2m/secで、合計5回の試行を実施した。図8のスタート地点、A地点、B地点の写真をそれぞれ図9,10,11に示す。枝分かれした幹や垂下った枝葉、脚立などが存在したり、木の間隔が疎であったり、センシングに厳しい環境であることが窺える。

図12,13はそれぞれPF、EKFで自己位置同定しながら巡回走行(1m/sec)した時の推定結果と実際の走行軌跡(DGPS)である。PFによる推定が第1直線で乱れているのは、経路上の窪みや傾斜によるもの

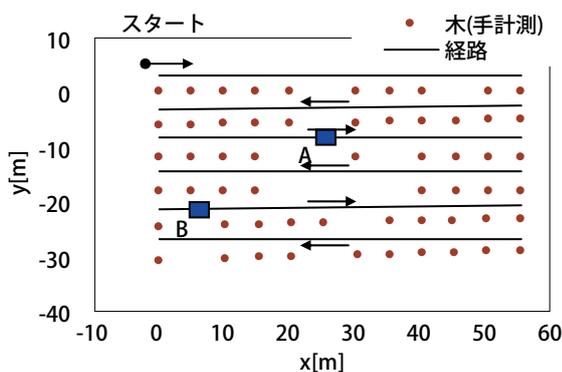


図8 マップと目標経路



図9 スタート地点と座標設定



図 10 図 8 の A 地点



図 11 図 8 の B 地点

である。EKFの場合はオドメトリに対応する共分散を小さめに設定しているため滑らかである。EKFによる推定において、第5直線の一部で大きな誤差が発生しているのは、マップに対応するレーザクラスタが少なくなるためと考えられる(図14)。またマップが点で定義されているのに対して、クラスタは枝分かれ、枝葉の垂下りなどによる広がりを持つ。このような箇所では三角測量は困難である。EKFと組合せても大きな誤差となった。これに対してPFによる推定は安定していた。

旋回部分では両手法とも推定が乱れる箇所がある。見える木が少なくなることに加え、その箇所に傾斜が存在するためである。このような条件でEKFはたまに自己位置を見失うケースがあったが、PFは多少の乱れはあっても見失うことはなくロバストであった。なお、推定軌跡が膨らんでいるのはLRFとGPSアンテナ搭載位置の違いによるものである。

精度について、一般にセンシングが安定している場合はEKFを用いる方が良いが、今回の様な環境ではPFで同程度の良い結果が得られた。表1に直線区間(x=5~51m:両端から2つ目の木と木の間)の位置推定誤差と経路追従誤差の比較結果を示す。それぞれDGPS軌跡および目標経路に対するy軸方向誤差の標準偏差である。標準偏差は各直線に対して計算し6本分の平均を取った。PFの数値がやや小さいが、参照したDGPSの水平方向誤差RMS(受信機の出力値)が約17cmであったので有意な差とはいえない。

一部推定が乱れる箇所が存在しても、経路再生成(図17)により車両の過剰反応が抑制されている。さらに逆最適制御の安定余裕と併せ、凹凸や滑りなど外乱を受けるコースで安定して走行することができた。ただし2m/secでは少し蛇行が大きくなることがあった。上記と同じ直線区間での位置推定誤差と経路追従誤差の時間変化を図15(1m/sec)、図16(2m/sec)に示す。

なお、使用したパーティクル数は300、予測からパーティクル再生成まで含む1回の繰返し処理時間は平均15.7msecであった。レーザスキャン周期25msec内で処理可能であるが、制御の遅れにつながる可能性がある。2m/secでの蛇行原因と考えられるので処理の効率化が課題である。一方、三角測量とEKFを組合せた処理時間は0.5msec未満であり、制御への影響はほとんどない。環境に応じて処理を切替えるのも解決策の1つとして挙げられる。

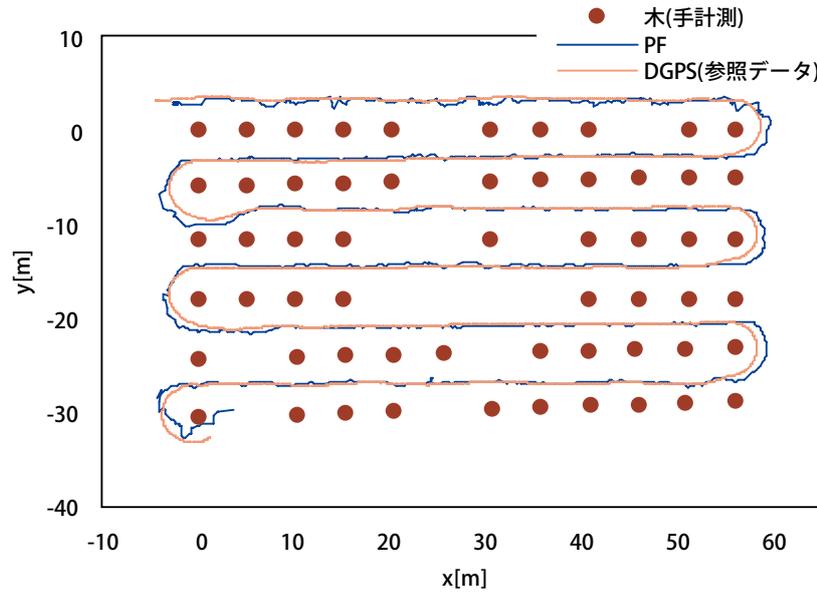


図 12 PF を用いた巡回走行 (1m/sec)

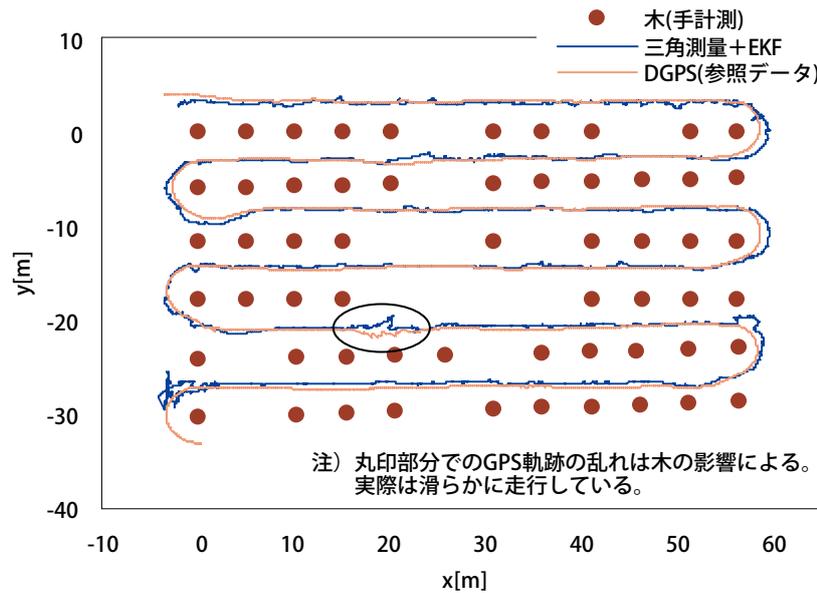


図 13 EKF を用いた巡回走行 (1m/sec)

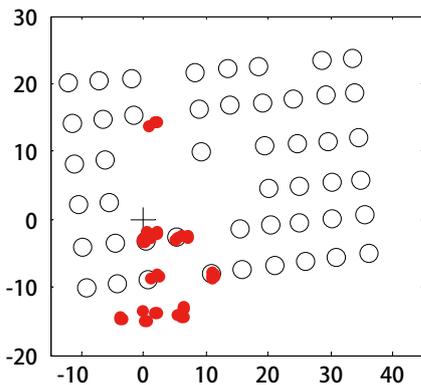


図 14 図 13 丸印付近での観測

表 1 自己位置推定誤差と経路追従誤差

	設定車速 [m/sec]	位置推定誤差 σ [m]	追従誤差 σ [m]
PF	1.0	0.154	0.0928
	1.5	0.146	0.122
	2.0	0.185	0.295
EKF	1.0	0.191	0.133
	2.0	0.224	0.226

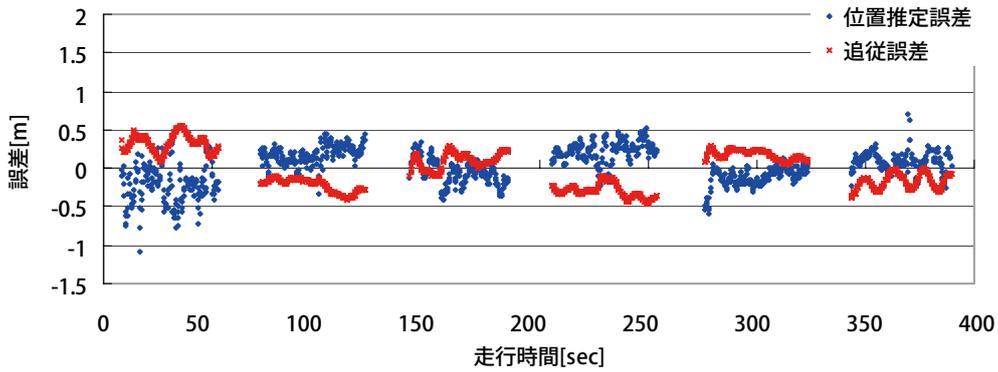


図 15 PF を用いた場合の誤差の時間変化 (1m/sec)

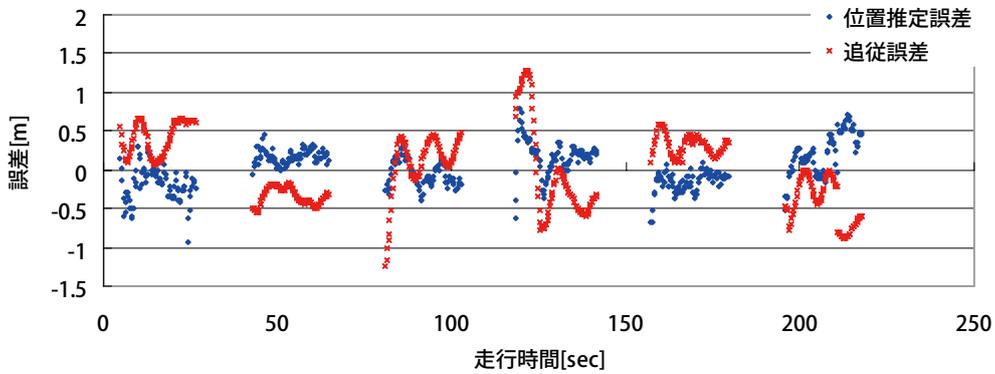


図 16 PF を用いた場合の誤差の時間変化 (2m/sec)

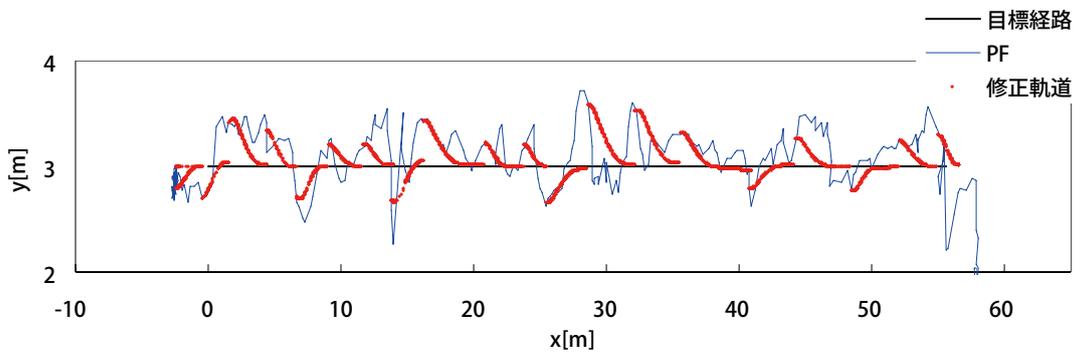


図 17 第 1 直線における追従誤差発生時の修正軌道 (1m/sec)

4.2 初期位置、yaw角誤差を与える実験

マップを参照して自己位置同定する場合、車両の正しい初期位置、yaw角を与えておく必要がある。そうしなければレーザクラスタとマップの各木の対応付けができない。現場での使用を想定すると、木の配置を目安に車両を置くことが考えられるが、位置や向きのずれに気を使わずに済むのが望ましい。本実験では図18に示す様に、プログラムに与える初期位置、yaw角を決めておき($x=-2\text{m}$, $y=-3\text{m}$, $\text{yaw}=0\text{deg}$)、実際の車両位置、yaw角を色々変えて正しく走行開始できるかテストした。図中の“成功”とは図19に示すように初期位置誤差を修正し、その後の巡回走行に成功したことを意味する。“1個ずれる”とは図20に示すように誤差を修正できず、木の間隔分誤って認識し続けたことを意味する。図は例とし

てx方向のずれを示すが、y方向のずれも発生し得る。“理論境界”とは仮の初期位置から木の間隔半分の距離をオフセットしたラインである。車両をこの境界内に置けば誤差がうまく修正されるべきであるが、境界に近い位置でずれてしまうケースもある。しかしここまでずらすと人の感覚でも明らかに分かるので、実用上問題ないと考えられる。

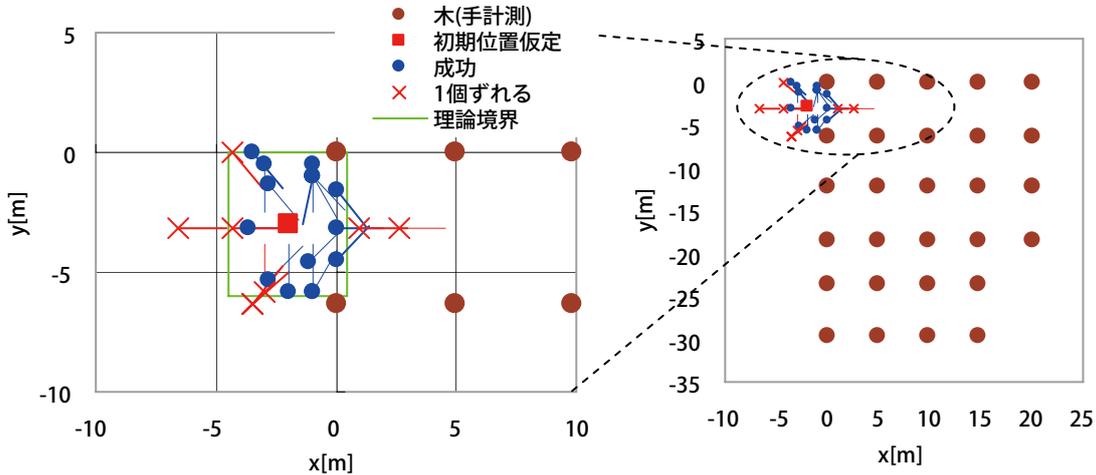


図 18 初期位置仮定 ($x=-2, y=-3, \text{yaw}=0\text{deg}$) に対し、実際の車両位置・yaw 角を変えてスタート

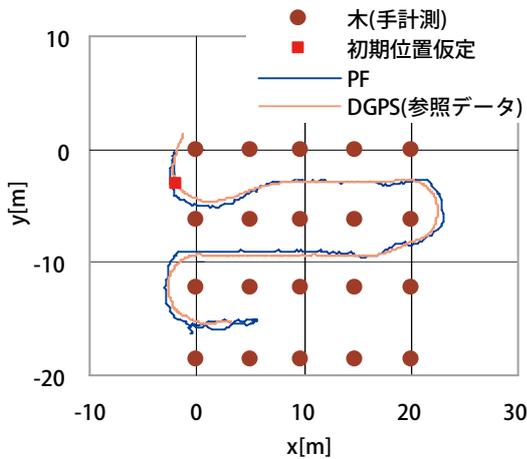


図 19 $x=-1, y=-1, \text{yaw}=-100\text{deg}$ からスタート

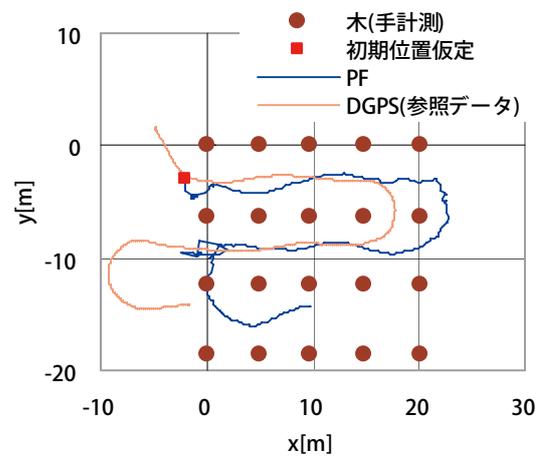


図 20 $x=-4.3, y=0, \text{yaw}=-50\text{deg}$ からスタート

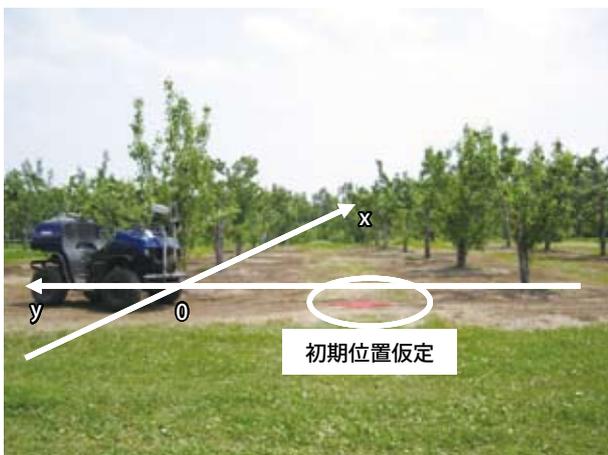


図 21 $x=-1, y=-1, \text{yaw}=-100\text{deg}$ からスタート

5 おわりに

二次元レーザレンジファインダとパーティクルフィルタによる自己位置同定を採用し、拡張型カルマンフィルタと比較することによって、センシングが不安定な環境および初期誤差が大きな場合の有効性を示した。さらにロバストな制御手法と経路再生成を組合せることにより、外乱に対して正確で安定した巡回走行を達成した。

またこれらの技術を採用することで高価な計測器が不要になるというメリットもあり、不整地・半自然環境での作業自動化が現実的なものに近づく。今後PF処理高速化などによる、より速い車速での安定した走行、また事前マップの効率的な取得方法なども課題となる。

謝辞

本研究は神戸大学と共同で行ったものであり、深尾隆則准教授と同研究室の学生に多大な協力を頂きました。この場にて厚く御礼申し上げます。

参考文献

- [1] Kurashiki,K., Fukao,T., Nagata,J., Ishiyama,K., Kamiya,T., Murakami,N.: Laser-based Vehicle Control in Orchard, AGRICONTROL 2010
- [2] Thrun,S., Burgard,W., Fox,D.: Probabilistic ROBOTICS (2005)
- [3] Welch,G., Bishop,G.: An Introduction to the Kalman Filter. Technical report, University of North Carolina (2001)
- [4] 倉鋪, 深尾, 大須賀, 石山, 竹原, 今井: 不整地を想定した移動ロボットのロバスト制御, 日本機械学会論文集C編, Vol. 74, No. 747, pp. 2705.2712, 2008

■著者



石山 健二
Kenji Ishiyama
技術本部
研究開発統括部
安全技術研究部



神谷 剛志
Tsuyoshi Kamiya
技術本部
研究開発統括部
安全技術研究部



深尾 隆則
Takanori Fukao
神戸大学大学院
工学研究科
機械工学専攻



倉鋪 圭太
Keita Kurashiki
神戸大学大学院
工学研究科
機械工学専攻