法政大学学術機関リポジトリ
HOSEI UNIVERSITY REPOSITORY

PDF issue: 2025-06-01

長距離移動のための自律ナビゲーションシステムの開発

澤邊, 航 / SAWABE, Wataru

(発行年 / Year)
2010-03-24
(学位授与年月日 / Date of Granted)
2010-03-24
(学位名 / Degree Name)
修士(工学)
(学位授与機関 / Degree Grantor)
法政大学 (Hosei University)

2009 年度　修士論文

長距離移動のための
自律ナビゲーションシステムの開発

指導教授　渡辺嘉二郎　教授

法政大学大学院工学研究科
システム工学専攻修士課程
08R6108
サワベ　ワタル
沢辺　航

Development of Autonomous Navigation System for Long-distance Movement

Abstract

This paper describes a new implementation of the autonomous navigation system for long-distance movement such as Real World Robot Challenge (RWCR). In proposed system, to navigate safely, the current position of the mobile robot is corrected with the image template matching with the Normalized Cross-Correlation (NCC) between a probability map and stored global probability map. The probability map is generated by occupation probability based grid based on profile of the laser range finder. The probability map is updated obstacles information based on laser range finder uncertainty model. To confirm the validity of proposed autonomous navigation system, an electric wheelchair based mobile robot is used for implementation and tested actual outdoor experiments.

Key Words: *mobile robot， autonomous navigation system， occupation probability based grid*

－目次－

第１章　はじめに ..... - 3 -
　１．１　研究背景と目的 ..... - 3 -
　１．２　**RWRC** ..... - 4 -
第２章　システムと仮定の記述 ..... - 5 -
　２．１　**RWRC** の走行環境 ..... - 5 -
　２．２　提案する自律ナビゲーションシステム ..... - 6 -
　２．３　仮定の記述 ..... - 7 -
第３章　従来のナビゲーションシステム ..... - 8 -
　３．１　**Dead reckoning** 法 ..... - 9 -
　３．２　**Kalman Filter** ..... - 10 -
第４章　障害物を利用した確率マップの生成 ..... - 14 -
　４．１　障害物情報の時系列統合 ..... - 14 -
　４．２　障害物情報の不確かさモデル ..... - 15 -
　４．３　グローバル確率マップの生成 ..... - 17 -
第５章　マッチングによる自己位置補正・角度補正 ..... - 18 -
　５．１　自己位置補正 ..... - 18 -
　５．２　ジャイロ補正 ..... - 19 -
　５．３　実時間性の考慮 ..... - 20 -
第６章　移動ロボット制御 ..... - 21 -
　６．１　ウェイポイント設定 ..... - 21 -
　６．２　**Line of Sight** ..... - 21 -
第７章　実験と検証 ..... - 23 -
　７．１　シミュレーション実験 ..... - 23 -
　７．２　実機実験 ..... - 25 -
第８章　むすび ..... - 32 -
参考文献 ..... - 33 -
付録 ..... - 34 -
　Laser rangefinder の仕様 ..... - 34 -
　Optical fiber Gyroscope の仕様 ..... - 35 -
　Speedometer の仕様 ..... - 36 -
　Differential GPS の仕様 ..... - 37 -
　D/A 変換機の仕様 ..... - 37 -
　ベース車両の仕様 ..... - 38 -
謝辞 ..... - 39 -
活動実績 ..... - 40 -
研究業績 ..... - 41 -

第１章はじめに

１．１　研究背景と目的

移動は人間にとって快適な生活するために重要な行動である．しかし，現在の日本社会において，視覚・聴覚・下肢機能に障害を持つ方や高齢者の自立的な移動は困難となっている．このような認知・駆動に障害を持つ方と高齢者の移動支援を目的とした自律移動ロボットの開発・研究が近年盛んに行われている[1] [2].

人間の支援を目的とした移動ロボットは日常生活環境下での安全な動作が要求されるため，検証が難しく，未だ実用化へは至っていない．この背景のもと，2007 年から日常生活環境下における移動ロボットの実用化を目指した公道実験として，つくばチャレンジ RWRC(Real World Robot Challenge)が行われている[3]． RWRC では，移動ロボットが人の手を介さず，人間の生活環境下で安全に約 1km の長距離移動しなければならない．このとき移動ロボットは走行環境を判断すると同時に，正確な自己位置の推定を行うことが重要となる．

従来，移動ロボットの多くは，外界センサ（GPS）と内界センサ（車速計・ジャイロ）を Kalman Filter でセンサフュージョンすることで自己位置の推定を行い，自律走行する手法が最も一般的であった[4] [5] [6]．他に，Kalman Filter の制約条件を補った particle filter による自己位置推定法[7] [8]なども存在しているが，これらの手法は，最終的に GPS の受信精度に依存してしまい，木や建物などが多く存在する日常生活環境下での運用は困難となる．

近年注目されている自己位置推定法のひとつに Simultaneous Localization And Map building (SLAM)がある[9] [10]. SLAM とは,移動ロボットが時々刻々位置を変えても逐次的に環境マップを生成でき,同時にマップ内での自己位置を正確に求めることを可能にする問題の総称である．SLAM は環境地図を構築すると同時に自己位置推定を行うため，GPS などの受信精度にかかわらず，環境情報から正確な自己位置の推定をすることができる．

本研究は，RWRC の環境を想定し，事前に Laser range finder (LRF)から取得した周辺環境情報をもとに，GPS の受信精度に依存せず，リアルタイムで環境地図を構築することで，安全に長距離移動を可能にする自律ナビゲーションシステムを提案し，シミュレーションと実機へ実装することで検証を行う．

１．２　RWRC

RWRC は，自ら考え行動するロボット（自律型ロボット）が，実際に人が生活する街の中で，速度を競うのではなく，“安全かつ確実に動く”ことを目指す技術チャレンジである．これは人とロボットが共存する社会の実現を目標としているためであり，日本初の行動における移動ロボットの大規模な実証実験の場となっている．大会のルールとして，茨城県つくば市の中央公園に作られた全長 1km 走行コースにおいて，移動ロボットが自律で走行を行う．このとき最高速度は 4km/h 以内で 2 時間以内に走行することが要求される．

RWRC はトライアル走行と本走行からなる．トライアル走行は車両審査を兼ねた試験走行であり，走行コースの約 100m 地点まで到達することが課題として与えられる．トライアル走行を突破すると，本走行へ参加でき，トライアル走行の到達時間順に走行することとなる．Fig.1 に 2008 年度 RWRC の走行コースを示す．

Fig.1 RWRC 2008

第２章　　システムと仮定の記述

２．１　RWRC の走行環境

RWRC は, 生活環境下である茨城県つくば市の平坦な遊歩道を競技コースとして設定され, 1km のコースを 2 時間以内に，自律的に走行することが要求される．この走行区間は多くの障害物が存在しており，森林区間や建物などに囲まれた区間が広く存在するため，GPS の精度は期待できない．車道に面した歩道，細い遊歩道など，1~2m 幅の走行コースを左側通行しなければならないため，高精度な自己位置推定が必要となる．

また，RWRC では，試験走行が認められており，GPS・車速計・ジャイロによる座標情報，LRF による環境情報の取得など，データ収集を行うことが可能である．本研究においても，事前に走行コースの環境情報を LRF で取得し，GPS・車速計・ジャイロにより，座標情報を得る．Photo.1 に RWRC の走行環境を示す．

Photo.1 RWRC course environment

２．２　提案する自律ナビゲーションシステム

Fig.2 に提案するナビゲーションシステムを示す.

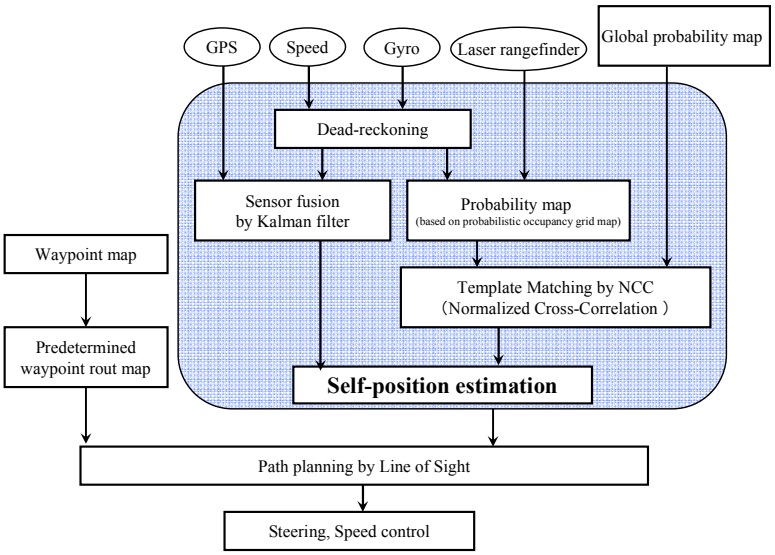


Fig. 2 Proposed navigation system

本研究では，LRF から取得した障害物情報を車速計・ジャイロを用いて時系列で統合し，リアルタイムで確率マップを生成する．この確率マップをテンプレートとし，事前に取得したデータから生成するグローバル確率マップとマッチングをとることで，GPS・車速計・ジャイロから推定される自己位置の補正を行う．移動ロボットは，あらかじめ複数の目標地点を設け，それら目標点を自律的に巡回して走行するウェイポイントナビゲーションを適応する．Fig.3 にウェイポイントナビゲーションの概要を示す.

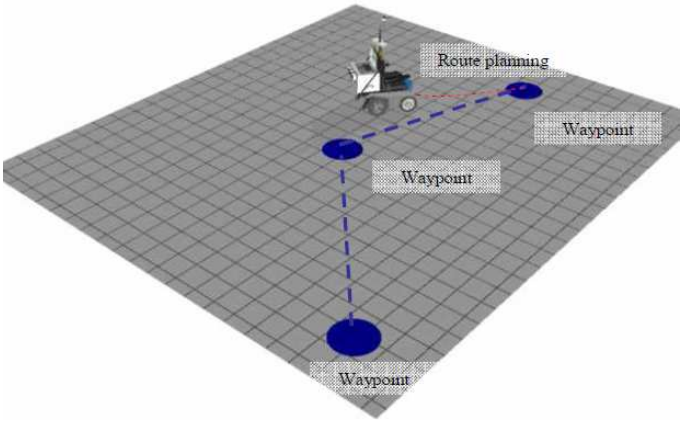


Fig.3　Route planning for mobile robot

２．３　仮定の記述

本研究における仮定を述べる. 提案する長距離移動のための自律ナビゲーションシステムについて，以下の 3 つの仮定を設ける.

(A1) 走行環境は二次元平面とする.
(A2) 確率マップ上における各グリッドでの障害物の存在確率は独立であり，そのグリッドに関する観測情報はそのグリッドの事象のみに依存する.
(A3) 環境情報を取得後，走行環境が大きく変化することはない.

仮定(A1)において，RWRC の競技環境は平坦な遊歩道のため，走行環境は二次元の平面空間とし，走行環境内に高低差はないものとする．仮定(A2)は，計算コストを減らし，確率マップをリアルタイムで生成するための仮定である．仮定(A3)は，事前に取得する環境データをもとに，自己位置補正を行うため，本研究ではデータ取得後，走行環境は大きく変化することはないとする.

第３章　　従来のナビゲーションシステム

本研究では，従来の自己位置推定法である，外界センサ（GPS）と内界センサ（車速計・ジャイロ）を Kalman Filter でセンサフュージョンすることで自己位置の推定を行う．この自己位置の推定結果は，事前に取得した環境データから生成するグローバル確率マップの限定処理に使用し，さらにこの推定結果を補正かけることで自律走行を行う．Fig.4 に従来のナビゲーションシステムの概要を示す．

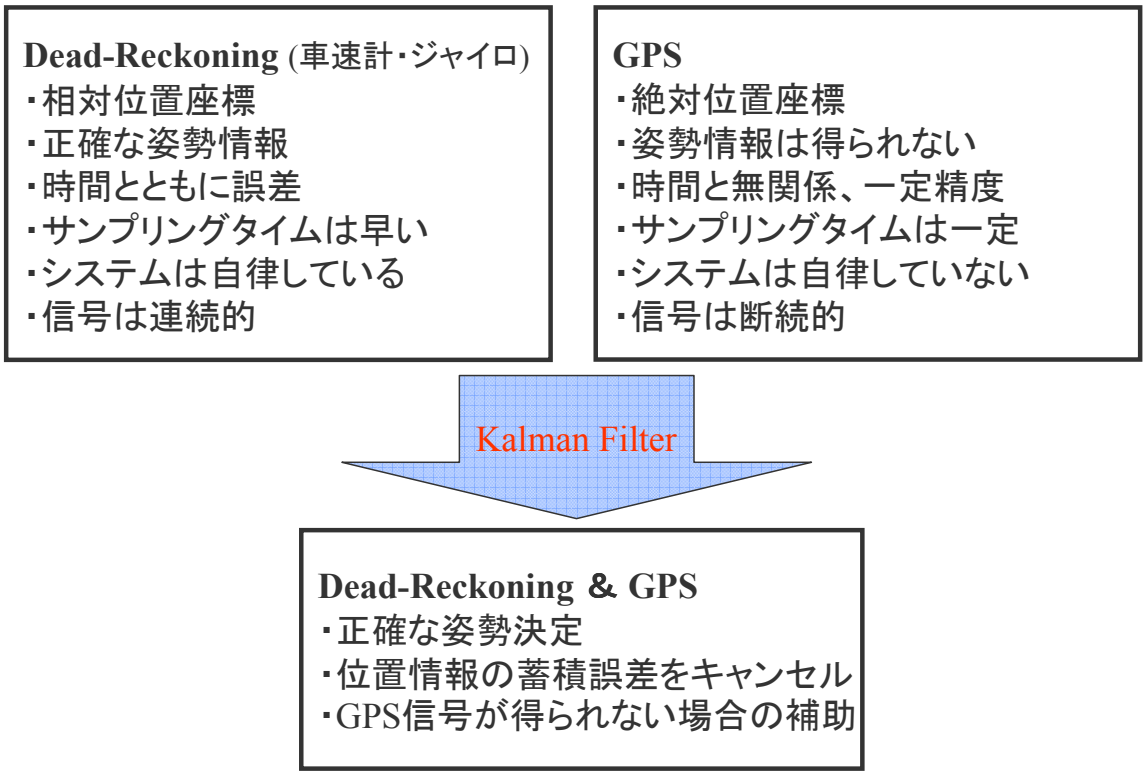


Fig.4　Outline of sensor fusion

３．１　Dead reckoning 法

内界センサ（車速計・ジャイロ）から自己位置を推定する手法である，Dead-Reckoning法を説明する．Dead-Reckoning法とは，ジャイロから得られる角度情報と車速計から得られる速度情報を積分することにより，車両の現在位置および姿勢（$x_{dr}(k), y_{dr}(k), \theta_{dr}(k)$）を推定する方法である．この方式の問題は，車輪径の誤差，車輪間隔の誤差，走行時のスリップ等の各種の誤差が累積していくことであるが，推定位置の補正を適時行うことができれば, 簡単で有効な方式である. 以下にDead-Reckoningに使用する変数と定数を定義し, Fig.5にDead-Reckoningのモデルを示す.

【変数と定数】

$x_{dr}(k), y_{dr}(k)$	:車両の相対位置座標
$\theta_{dr}(k)$	:車両の向き
τ	:サンプリング間隔
v	:車両の速度

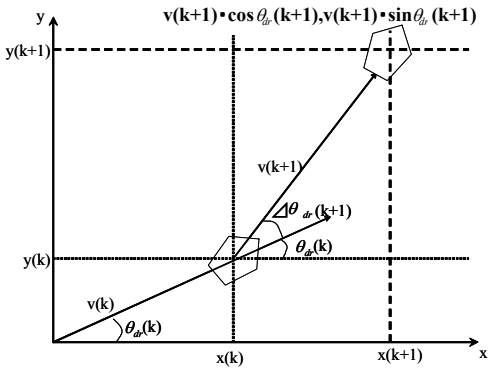


Fig.5 Dead-Reckoning

これより, 相対位置座標は式(1)として示すことができる.

$$
\begin{aligned}
x_{dr}(k+1) &= v(k)\cdot\tau\cdot\cos\theta_{dr}(k) + x_{dr}(k) \\
y_{dr}(k+1) &= v(k)\cdot\tau\cdot\sin\theta_{dr}(k) + y_{dr}(k)
\end{aligned}
\tag{1}
$$

Dead-reckoning法は, 回転速度, 速度などから経路を求める方式であるため, 時間が経つにつれ経路推定に誤差が蓄積することは避けられない. そのため, GPSなどといった外界センサとセンサフュージョンを行うことで, この欠点を補う必要がある.

３．２　Kalman Filter

外界センサ（GPS）と内界センサ（車速計・ジャイロ）をセンサフュージョンする際に使用するKalman Filterについて説明する．Kalman Filterとは，システムの状態推定を行うためのフィルタリング理論である．x_kを真の値の位置，現在の時刻kにおいて推定される位置を$x_k^{'}$とすると，過去から現在に至るまでのすべての観測値y_{k-m}の線形結合で現在の$x_k^{'}$を推定する．このとき時刻tにおける真の値x_kと推定値$x_k^{'}$の差の二乗の総和を最小にするようなQ_mを求める．ここで求めるべきQ_mは無限であるが，Kalman Filterを用いると現在の推定値は一つ前の推定値に既知の定行列をかけ，現在の観測値で決まる補正項を加えることにより簡単に求まる．これによって現在の推定値が求まると，次の時間$k+1$の推定値も同じように決まっていく．

x_k：真の値　　Q_m, G_{k-1}：係数行列　　y_{k-m}：観測値
$x_k^{'}$：推定値　　$correction$：補正項　　k：離散時間

$$x_k^{'} = \sum_{m=0}^{\infty} Q_m y_{k-m}$$

$$\left\langle \left\| x_k - x_k^{'} \right\|^2 \right\rangle \rightarrow \min \quad \square \; x_k^{'} = G_{k-1} x_{k-1}^{'} + correction$$

以下にシステムの基本的な変数と定数を定義する．

【共通変数と定数】

t	：連続時間
k	：車両センサ信号処理における離散時間
τ	：車両センサ信号処理におけるサンプル間隔

【地上座標系】

$[x(k), y(k)]$	：離散時刻kにおける地上水平面座標
$[x_M(k), y_M(k)]$	：DGPS により求められた座標
$\triangle d(k)$	：車両が時刻kから$k+1$の間に回る角度
$\triangle\theta(k)$	：車両が時刻kから$k+1$の間に回る角度
$\theta(k)$	：車両が向いている方向
$n_{\triangle d}(k)$	：$\triangle d(k)$に含まれる誤差
$n_{\triangle\theta}(k)$	：$\triangle\theta(k)$に含まれる誤差

【Kalman Filter】

l_k	：車両の状態ベクトル（位置と姿勢）
m_k	：観測ベクトル
z_k	：最適推定値
F_k	：状態遷移行列
G_k	：駆動行列
H_k	：観測行列
K_k	：カルマンゲイン
P_k	：推定誤差共分散行列
Q_k	：システムノイズの共分散行列
R_k	：観測ノイズの共分散行列
u_k	：システムノイズベクトル
v_k	：観測ノイズベクトル
$[n_x(k), n_y(k)]$	：$[x_M(k), y_M(k)]$に含まれる誤差

式(2) (3) に本研究の対象となる状態方程式と観測方程式を記す.

$$l_{k+1} = F_k l_k + u_k \quad \text{（状態方程式）} \tag{2}$$

$$m_k = H_k l_k + v_k \quad \text{（観測方程式）} \tag{3}$$

まず，状態方程式について定義する.

車両の状態ベクトルl_k，状態遷移行列F_kは式(4)，(5)となる.

$$l_k = [x(k) \quad y(k) \quad \cos\theta(k) \quad \sin\theta(k)]^T \tag{4}$$

$$F_k = \begin{bmatrix} 1 & 0 & \triangle d(k+1)\cdot\cos\triangle\theta(k+1) & -\triangle d(k+1)\cdot\sin\triangle\theta(k+1) \\ 0 & 1 & \triangle d(k+1)\cdot\sin\triangle\theta(k+1) & \triangle d(k+1)\cdot\cos\triangle\theta(k+1) \\ 0 & 0 & \cos\triangle\theta(k+1) & -\sin\triangle\theta(k+1) \\ 0 & 0 & \sin\triangle\theta(k+1) & \cos\triangle\theta(k+1) \end{bmatrix} \tag{5}$$

また，駆動行列G_k，システムノイズベクトルu_kはシステムノイズに関する項であり，式(6)として表される.

$$G_k = \begin{bmatrix} \cos\theta(k) & -\triangle d(k+1)\cdot\sin\theta(k) \\ \sin\theta(k) & \triangle d(k+1)\cdot\cos\theta(k) \\ 0 & -\sin\theta(k) \\ 0 & \cos\theta(k) \end{bmatrix} \tag{6}$$

$$u_k = \begin{bmatrix} n_{\triangle d}(k) & n_{\triangle\theta}(k) \end{bmatrix} \tag{7}$$

システムノイズの共分散Q_kは

$$Q_k = \begin{bmatrix} \sigma^2\{n_{\Delta d}(k)\} & 0 \\ 0 & \sigma^2\{n_{\Delta\theta}(k)\} \end{bmatrix} \tag{8}$$

式(2)，(4)～(7)より状態遷移モデルは式(9)として定義できる．

$$\begin{bmatrix} x(k+1) \\ y(k+1) \\ \cos\ \theta(k+1) \\ \sin\ \theta(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & \triangle\ d(k+1)\cdot\cos\ \triangle\theta(k+1) & -\triangle d(k+1)\cdot\sin\ \triangle\theta(k+1) \\ 0 & 1 & \triangle\ d(k+1)\cdot\sin\ \triangle\theta(k+1) & \triangle d(k+1)\cdot\cos\ \triangle\theta(k+1) \\ 0 & 0 & \cos\ \triangle\theta(k+1) & -\sin\ \triangle\theta(k+1) \\ 0 & 0 & \sin\ \triangle\theta(k+1) & \cos\ \triangle(k+1) \end{bmatrix} \begin{bmatrix} x(k) \\ y(k) \\ \cos(\ \theta) \\ \sin(\ \theta) \end{bmatrix} + \begin{bmatrix} \cos\ \theta(k) & -\triangle d(k+1)\sin\ \theta(k) \\ \sin\ \theta(k) & \triangle d(k+1)\cos\ \theta(k) \\ 0 & -\sin\ \theta(k) \\ 0 & \cos\ \theta(k) \end{bmatrix} \begin{bmatrix} n_{\triangle d}(k) \\ n_{\triangle\theta}(k) \end{bmatrix} \tag{9}$$

次に観測方程式について定義する．

観測ベクトルm_k，観測行列H_k，観測ノイズベクトルv_kは式(10)～(12)となる．

$$m_k = [x_M(k) \quad y_M(k)]^T \tag{10}$$

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{11}$$

$$v_k = [n_x(k) \quad n_y(k)]^T \tag{12}$$

観測ノイズの共分散行列R_kは

$$R_k = \begin{bmatrix} \sigma^2\{n_x(k)\} & 0 \\ 0 & \sigma^2\{n_y(k)\} \end{bmatrix} \tag{13}$$

(3)，(10)～(12)より観測方程式は次のように定義できる．

$$\begin{bmatrix} x_M(k) \\ y_M(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ y(k) \\ \cos\theta(k) \\ \sin\theta(k) \end{bmatrix} + \begin{bmatrix} n_x(k) \\ n_y(k) \end{bmatrix} \tag{14}$$

これらの予測値と観測値から，最適推定値z_kに対し，その誤差共分散のトレースを最小にする行列K（カルマンゲイン）は式(15)となる．

$$K = Q_k \cdot H_k{}^T \cdot (H_k \cdot Q_k \cdot H_k^T + R_k)^{-1} \tag{15}$$

このとき推定誤差共分散行列P_kは次のようになる．

$$P_k = Q_k - K \cdot H \cdot Q_k \tag{16}$$

これにより最適推定値z_kは式(17)から求めることができる．

$$z_k = l_k + K(m_k - Hz_k) \tag{17}$$

第４章　　障害物を利用した確率マップの生成

障害物の有無を確率的に表すマップとして,障害物存在確率マップ(Probabilistic Occupancy grid map)がある[11] [12]．障害物存在確率マップは複数回の観測を効率的に統合して生成されるグリッドマップである．RWRC では，走行環境がめまぐるしく変化するため，提案するシステムでは障害物存在確率マップを利用し，確率マップの生成を行い長距離移動における安定性の向上を図る．

以下に確率マップの生成で用いる変数と定数を定義する．

【変数と定数】

O ： 過去の観測値列

i ： 任意のグリッド番号

E_i ： i 番目のグリッドに障害物が存在する事象

$\overline{E_i}$ ： i 番目のグリッドに障害物が存在しない事象

o ：最新情報において障害物を観測した事象

o_i ： 最新情報において i 番目のグリッドに障害物を観測した事象

$\overline{o}$ ：最新情報において障害物を観測しない事象

$\overline{o_i}$ ： 最新情報において i 番目のグリッドに障害物を観測しない事象

４．１　障害物情報の時系列統合

本研究では，時系列統合に車速計・ジャイロを用いる．これは速度計・ジャイロの短期的な精度が良いためである．車速計・ジャイロにより移動ロボットの自己位置と向きが与えられ，観測（障害物情報）が得られたもとでの各グリッドの障害物の存在確率について考える．i 番目のグリッドに障害物が存在する事象を E_i とする．過去の観測値列を O とし，最新観測において障害物を観測した場合を o として，O に続いて o が得られたとき，E_i となる確率 $P(E_i \mid o, O)$ はベイズの定理により式(18)と表せる．

$$P(E_i \mid o, O) = \frac{P(o \mid E_i, O) P(E_i \mid O)}{P(o \mid O)} \tag{18}$$

ここで$P(o \mid E_i, O)$は，実際に障害物が存在するときに観測する確率を，$P(E_i \mid O)$は事前確率を表す．本研究では，事前確率$P(E_i \mid O)$の初期値として 0.5 と与えている．式(18)は仮定(A2)を用いれば式(19)と表せる．

$$P(E_i \mid o_i, O) = \frac{P(o_i \mid E_i, O)P(E_i \mid O)}{P(o_i \mid E_i, O)P(E_i \mid O) + P(o_i \mid \overline{E_i}, O)P(\overline{E_i} \mid O)} \tag{19}$$

ここでo_iはi番目のグリッドで障害物を観測した場合を表す．また，$P(o_i \mid \overline{E_i}, O)$は存在しない障害物を誤観測する確率を表し，$P(\overline{E_i} \mid O)$項は，

$$P(\overline{E_i} \mid O) = 1 - P(E_i \mid O) \tag{20}$$

を表す．本研究における確率マップは，式(2)を用いて障害物が観測されたグリッドを更新し，以下の式(21)を用いて障害物が観測されないグリッドの更新を行う．このとき，仮定(A2)より確率の更新はグリッドごとに独立に行う．

$$P(E_i \mid \overline{o_i}, O) = \frac{P(\overline{o_i} \mid E_i, O)P(E_i \mid O)}{P(\overline{o_i} \mid E_i, O)P(E_i \mid O) + P(\overline{o_i} \mid \overline{E_i}, O)P(\overline{E_i} \mid O)} \tag{21}$$

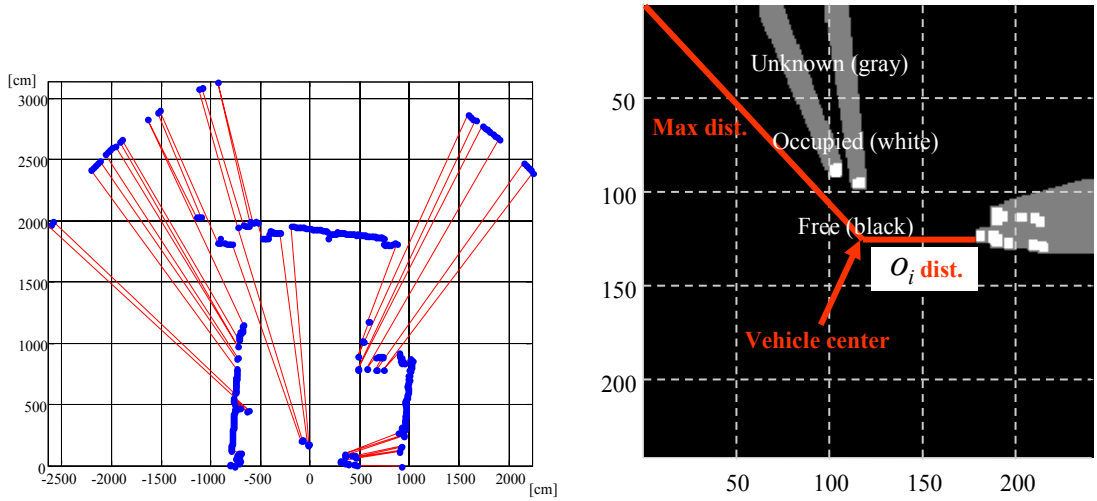
ここで，式(21)の各項は式(20)，式(22)より求まる．

$$\begin{aligned} P(\overline{o_i} \mid E_i, O) &= 1 - P(o_i \mid E_i, O) \\ P(\overline{o_i} \mid \overline{E_i}, O) &= 1 - P(o_i \mid \overline{E_i}, O) \end{aligned} \tag{22}$$

グリッドの更新を行うための式(19)，式(21)を計算するには，$P(o_i \mid E_i, O)$と$P(o_i \mid \overline{E_i}, O)$の項を定める必要がある．この項は，障害物情報の不確かさのモデルに相当する．

４．２　障害物情報の不確かさモデル

障害物が存在するときに観測する確率$P(o_i \mid E_i, O)$と，存在しない障害物を誤観測する確率$P(o_i \mid \overline{E_i}, O)$を，障害物情報ごとに定める．本研究では，障害物情報は LRF から取得する．LRF から得られるデータをxy座標系に変換し格子化を行う．このとき格子化データは 1 グリッドあたり 5cm とする．　Fig.6 (a)に LRF データのxy座標，(b)に格子化したデータを示す．



(a) xy coordinate (b) grid data (5cm/grid)

Fig.6 LRF data

LRF は水平面上を操作するため，Fig.6 (b)に示されるように，障害物情報は Occupied / Free / Unknown の 3 パターンとなる．この中で Occupied に事象 o を割り当て，Free に事象 $\bar{o}$ を割り当てる．Unknown は観測において何の情報も得られないため，グリッドの更新は行わない．また，Fig.6 (a)に示されるように，LRF はレーザの跳ね返り方によって障害物が観測できない場合がある．本研究では障害物情報の不確かさモデルを式(23)が示す値に設定した．

$$P(o_i \mid E_i, O) = 0.98 - 0.93 \times \frac{\text{Dist. to observed } o_i}{\text{Max dist. to observed grid}} \qquad (23)$$
$$P(o_i \mid \overline{E_i}, O) = 0.05$$

この不確かさモデルは，実際に障害物が存在するときに観測する確率を，移動ロボットからの距離と比例させて最大 0.98 から最小 0.05 と定めている．また，存在しない障害物を誤観測する確率を 0.05 と定めている．

Fig.7 は式(23)を用いた障害物マップ上のグリッド更新モデルである．横軸は更新前のグリッドの障害物存在確率を示し，縦軸は更新後のグリッドの障害物存在確率を示す．Fig.7 の左上の実線は，事象 o が起きた後の障害物存在確率 $P(E_i \mid o, O)$ であり，右下の破線は事象 $\bar{o}$ 起きた後の障害物存在確率 $P(E_i \mid \bar{o}, O)$ である．障害物存在確率は移動ロボットに近いほど更新変化量が大きくなっている．

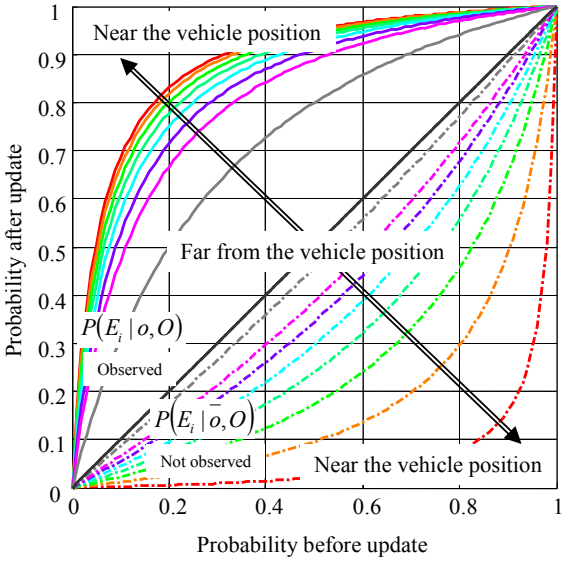


Fig.7 Grid update model for obstacles map

４．３　グローバル確率マップの生成

障害物情報を時系列統合し，不確かさモデルから確率マップが生成される．これを従来の GPS・車速計・ジャイロより推定された自己位置をもとにマップの重ね合わせを行い，グローバル確率マップを生成する．Fig.8 にグローバル確率マップ生成の処理を示す．

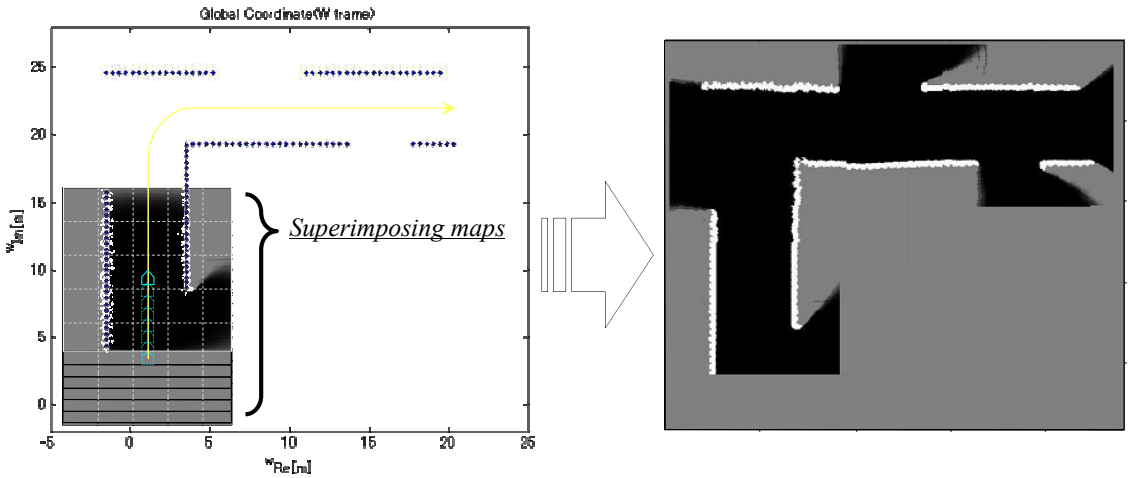


Fig.8 Global map generation

単純に確率マップの重ね合わせからグローバル座標系における確率マップを生成したため，マップ上にゆがみなどが生じる場合がある．これは時系列統合に使用するジャイロ・車速計の誤差などが原因として考えられるが，確率マップは同一コース上において障害物情報が与えられる場合，再現性が高いため，特に問題とはならない．

第５章　　マッチングによる自己位置補正・角度補正

５．１　自己位置補正

相関係数を用いたマッチングに正規化相互相関 NCC (Normalized Cross-Correlation)がある[13]．NCC は，データ各点の輝度値から局所的な平均値を引き，分散値の類似度でマッチングを取るため，ノイズに強いという特徴を持つ．確率マップでは観測毎に更新を行う性質上，ノイズが含まれることがあるため，本研究では NCC を利用したマッチングを行う．

NCC は，テンプレートの大きさを $M \times N$，テンプレートの位置 (i, j) における画素値を $T(i, j)$，探索対象データの画素値を $I(i, j)$ とすると，以下の式(24)で定義される．

$$R_{ncc} = \frac{\sum_{j=0}^{N-1} \sum_{i=0}^{M-1} \left(I(i, j) T(i, j) \right)}{\sqrt{\sum_{j=0}^{N-1} \sum_{i=0}^{M-1} I(i, j)^2 \times \sum_{j=0}^{N-1} \sum_{i=0}^{M-1} T(i, j)^2}} \tag{24}$$

相関係数 R_{ncc} は理想的に一致した場合，最大値１となる．本研究では，探索対象データをグローバル確率マップ，テンプレートデータをリアルタイムで生成される確率マップとしてサンプリング毎に NCC マッチングを行い，自己位置を補正する．Fig.9 に NCC によるマッチング例を示す．

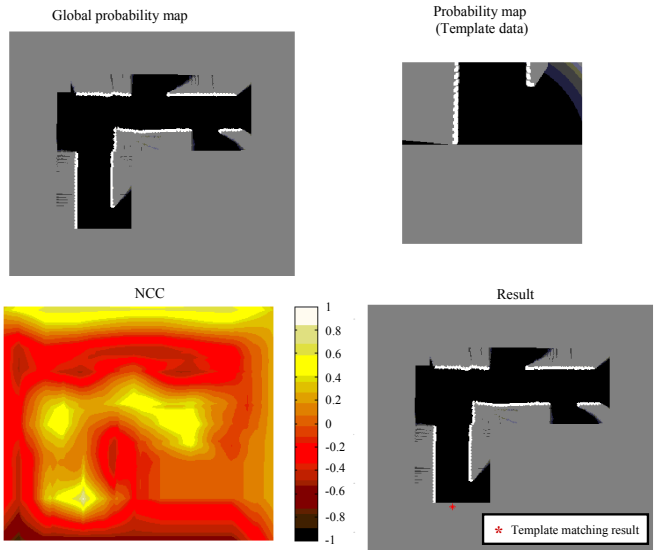


Fig.9 NCC matching result

探索対象データであるグローバル確率マップは，事前のデータ取得時に得られた環境情報から構築される．このことから，本研究で行う補正は，一度完走することができた地図内の自己位置情報へ強引に修正を行う行為であると言える．

また，本研究では移動ロボットが動作を始める直前にグローバル確率マップとマッチングをとり，グローバル確率マップ内の初期位置を定義する．その初期位置を移動ロボットのスタート地点である(0,0)座標と対応を付け，自己位置の補正を行う．

５．２　ジャイロ補正

ジャイロは振動や温度ドリフトなどにより，誤差が生じる．この誤差は，走行中累積誤差として自己位置推定に影響があるため，長距離移動時では無視することができない．本研究では設定した間隔のみで補正処理を行う．ジャイロの補正は以下の 3Step で構成する．Fig.10 にジャイロ補正の処理フローを示す．

Step1　車両を停止させる．

Step2　確率マップを-5～5 deg，0.5deg 刻みで回転させその都度 NCC マッチングをとる．

Step3　相関係数の中で最大となる値からオフセット角を割り出し，補正角を決定する．

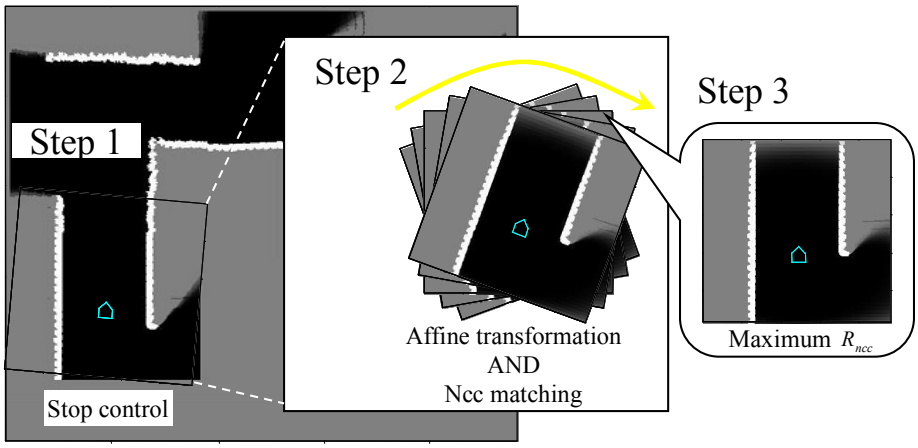


Fig.10 Gyro error correction

Step1 は，移動ロボットが移動中にジャイロ補正処理を行うには，処理時間がかかりすぎてしまうため，車両を停止させる．また，Step2 における回転角の幅と刻み角は，使用するジャイロの特性と設定するジャイロ補正の間隔からを見て判断しなければならないパラメータとなる．本研究では第 7 章で使用する実機の特性から設定する．

５．３　実時間性の考慮

NCC によるマッチングの処理時間は，探索対象データ・テンプレートデータのサイズに大きく依存する．本研究では実時間性を考慮し，探索対象データであるグローバル確率マップの探索範囲の限定化を行い，局所的なマッチングを行う．限定処理は，移動ロボットが 1 サンプリングに進む最大距離を目安にサイズを決定し，推定された自己位置座標を中心に設定し，サンプリング毎に更新を行う．Fig.11 に限定処理を示す．

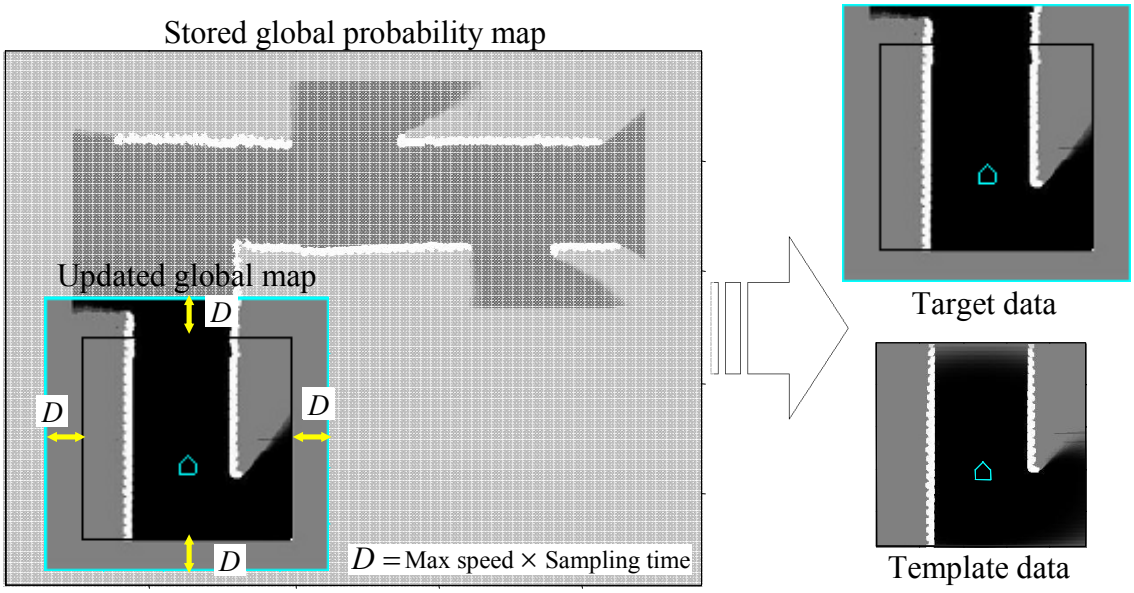


Fig.11 Update global probability map

Table.1 に限定処理を行う場合と行わない場合の処理時間の比較を示す．グローバル確率マップの限定処理を行うことで，実機の実時間制御が可能となる．

Table.1 Processing time of template matching

	Unprocessed data 1000×1000	**Processed data** 260×260
Template data 240×240	2.84 sec	0.243 sec

第６章　　移動ロボット制御

６．１　ウェイポイント設定

RWRC は走行コースのみが与えられ，そこから事前に取得する環境情報及び自己位置情報をもとに，ウェイポイントの設定を行う必要がある．本研究では事前に取得する走行コースの環境情報から生成されるグローバル確率マップを使用し，ウェイポイントの設定を行う．

６．２　Line of Sight

安定した長距離移動を行うため，移動ロボットのステアリング，速度制御はLine of Sight (LOS)を採用する[14] [15]．以下にLOSで使用する変数と定数を定義し，Fig.12にLOSの制御方法を示す．

【変数と定数】

k	:事前に登録した Waypoint 番号
x_k, y_k	:今現在目指している k 番目の Waypoint 座標
x_{k-1}, y_{k-1}	:通過直後の $k-1$ 番目の Waypoint 座標
r	:移動ロボットの中心円の半径
x_{est}, y_{est}	:推定された移動ロボットの自己位置
x_{los}, y_{los}	: k 番目と $k-1$ 番目の Waypoint をつなぐ直線と中心円の接点
θ	:移動ロボットのステアリング角

ステアリング角 θ は以下の式(25)と表せる．

$$\theta = \arctan\left(\frac{y_{los} - y_{est}}{x_{los} - x_{est}}\right) \tag{25}$$

ここで接点 x_{los}, y_{los} は式(26) (27) の関係より得られる．

$$\left(x_{los} - x_{esr}\right)^2 + \left(y_{los} - y_{esr}\right)^2 = r^2 \tag{26}$$

$$\left(\frac{y_{los} - y_{k-1}}{x_{los} - x_{k-1}}\right) = \left(\frac{y_k - y_{k-1}}{x_k - x_{k-1}}\right) = \text{constant} \tag{27}$$

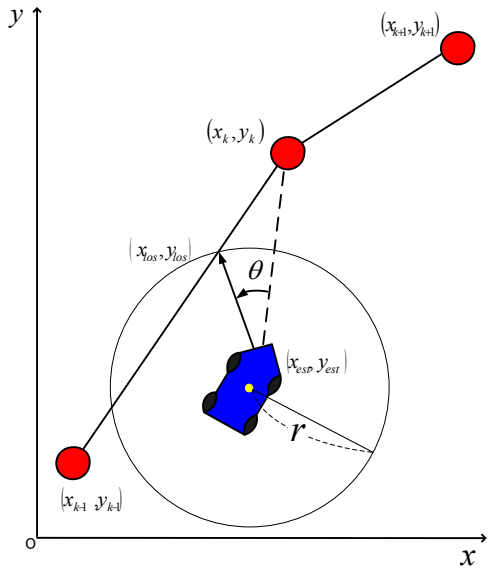


Fig.12 Conceptual diagram of Line of Sight

LOS は，車両中心の円の半径とウェイポイントのみから移動ロボットの制御量を決定できるため，システムを簡略化することができる．本研究では，Waypoint ごとに円の半径・最高速度を設定し，移動ロボットの制御性の向上を図る．

第７章　　実験と検証

７．１　シミュレーション実験

提案するシステムが移動ロボットに適応可能か検証するため，GPS の受信精度が著しく悪い場合と，ジャイロに積分誤差が加わった場合のシミュレーション実験を行う．

GPS の受信精度が著しく悪い場合のシミュレーション実験は，マッチングによる自己位置補正の有用性を検証するために行う．このとき GPS，車速計，ジャイロおよびレーザレーダの各センサにランダムなノイズを加える．このノイズは，MATLAB の randn 関数を用いて正規分布の乱数を用いる．レーザレーダは，時間に無相関はセンサであり計測される距離データにランダムなノイズを加える．

ジャイロに積分誤差が加わった場合のシミュレーション実験は，ジャイロ補正の有用性を検証するために行う．このときジャイロの積分誤差を想定し，サンプリング毎に 0.25deg 加算し,他のセンサのノイズは無いものとする.ジャイロ補正処理は 10 サンプル毎に行う．

このシミュレーション実験では共通して，移動ロボットが移動時に生成する確率マップを 240×240(12m×12m)サイズ，グローバルマップの限定処理から得られる探索対象データを 260×260 (13m×13m)サイズと設定する．RWRC の規定を満たせるよう最大速度は 1km/h と設定する．また，実機の実時間制御を考慮し，事前に処理時間の計測を行った．結果を Fig.13 に示す．これより，実機でのデータ取得を考慮し，サンプリング間隔を 0.8sec と設定する．マッチング処理は動き出しの確率マップのバッファを得るため，数サンプリング従来の手法で動作した後，開始する．シミュレーション実験では 20 サンプル以降，マッチング処理を行う．また，シミュレーション実験における実験環境は，第 4 章で生成したグローバル確率マップと障害物情報を用いる．

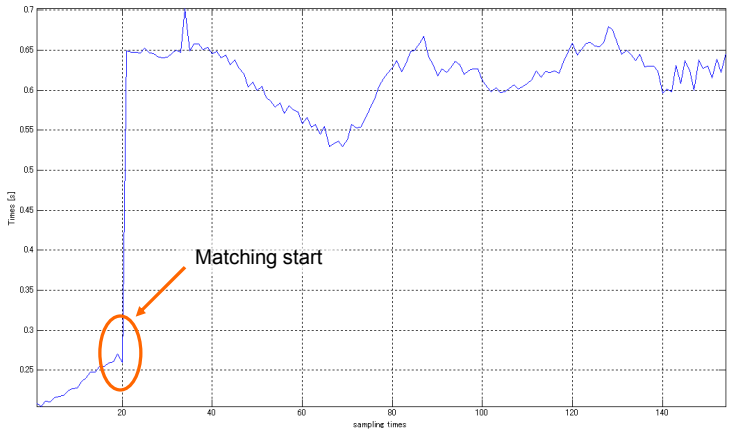


Fig.13 Processing time of proposed navigation system

Fig.14 に GPS の受信精度が著しく悪い場合のシミュレーション実験の結果を示す.

Estimated Trajectory　　Estimation errors

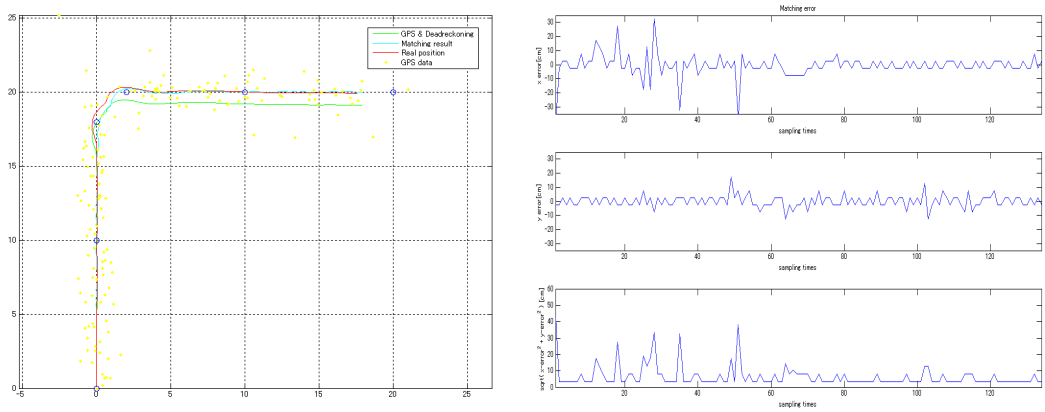


Fig.14 Simulation result of proposed system

従来の手法から得られる自己位置推定結果はセンサのノイズにより，理論上の自己位置から離れてしまった．提案するシステムから得られる自己位置推定の結果は理論上の自己位置とほとんど重なり，従来の自己位置推定結果をサンプリング毎に補正できていることが確認できる．

Fig.15 にジャイロに積分誤差が加わった場合のシミュレーション実験の結果を示す.

Estimated Trajectory　　Estimation gyro errors

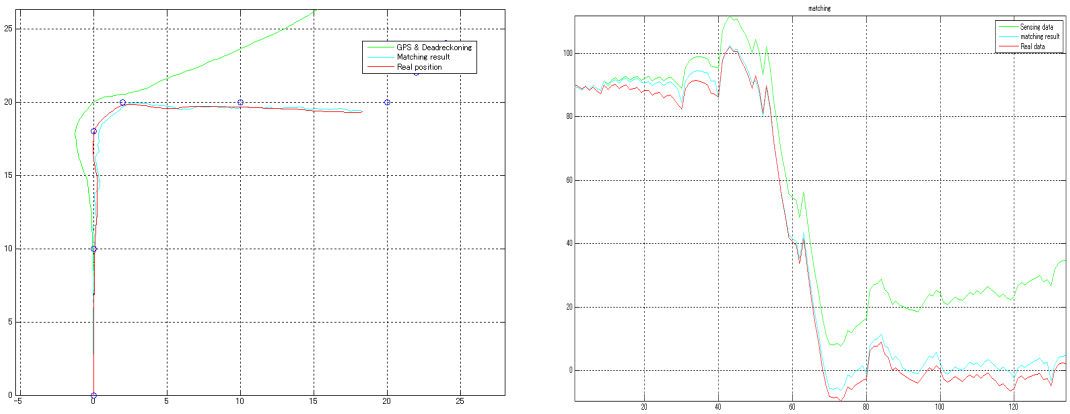


Fig.15 Simulation result of gyro error correction

従来の手法はジャイロの積分誤差の影響により推定位置が理論上の自己位置から大きく離れてしまった．ジャイロ補正処理により，ジャイロの積分誤差を補正し，正確な自己位置推定が行えていることが確認できた.

以上のシミュレーション実験により，提案するシステムは正確な自己位置推定を行うことができ，実機へ実装し検証することが可能であると判断できた.

７．２　実機実験

提案するシステムを検証するため，実機実験を行う．Fig.16 に福祉車両をベースとした移動ロボットを示す．移動ロボットは障害物検出センサとして前方に LRF，角度センサとして光レートジャイロ，速度センサとしてロータリーエンコーダを用いた車速計，絶対位置センサとして GPS を搭載する．移動ロボットの背面には DA 変換モジュールが接続された Laptop PC (Core 2 Duo processor 2.6GHz,　3GB RAM) が搭載されている．移動ロボットは，各センサから取得した情報を Laptop PC にて処理し，DA 変換モジュールを通して車両を制御する．

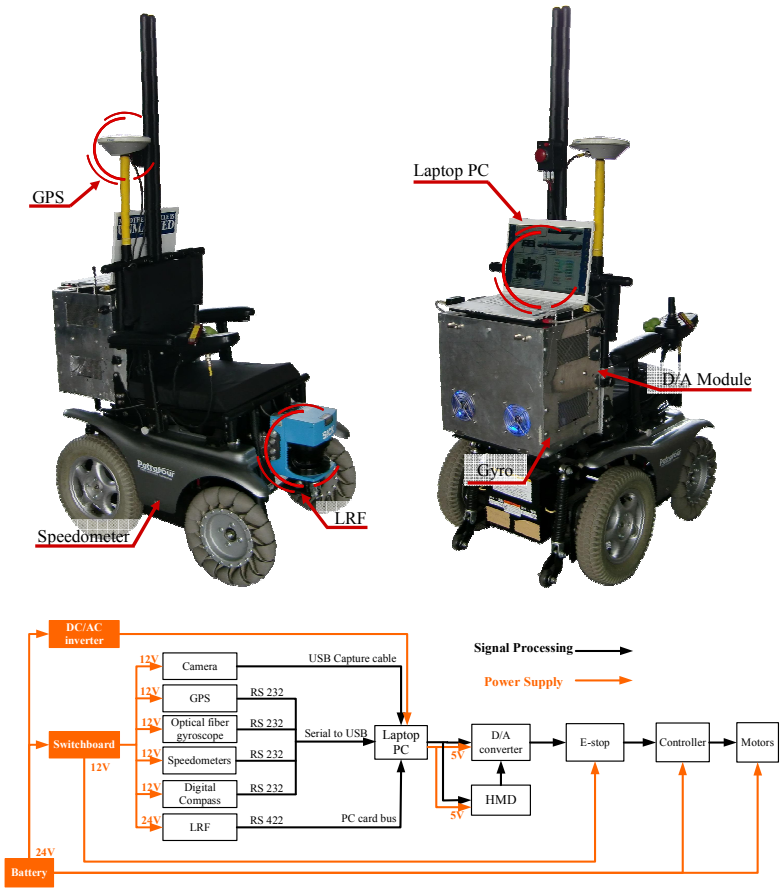


Fig.16　Developed mobile robot

RWRC を想定した実験環境として，法政大学小金井キャンパス内に全長約 200m のコースを設定する．Fig.17 に走行コースを示し，Fig.18 に事前に取得した環境データから生成したグローバル確率マップを示す．実験環境は，建物に囲まれた状態での自律走行になるため，GPS の受信が期待できなく，RWRC の走行環境よりも困難である．

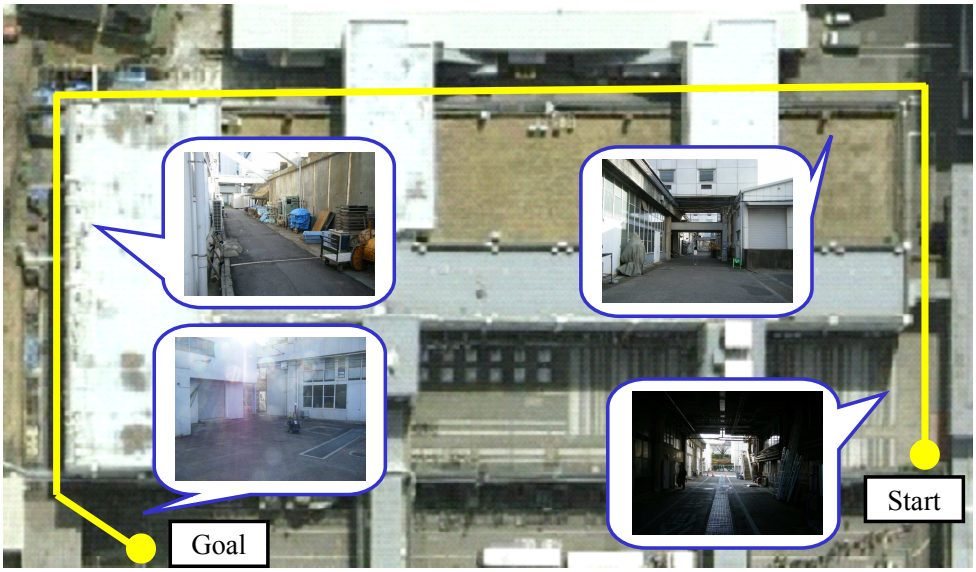
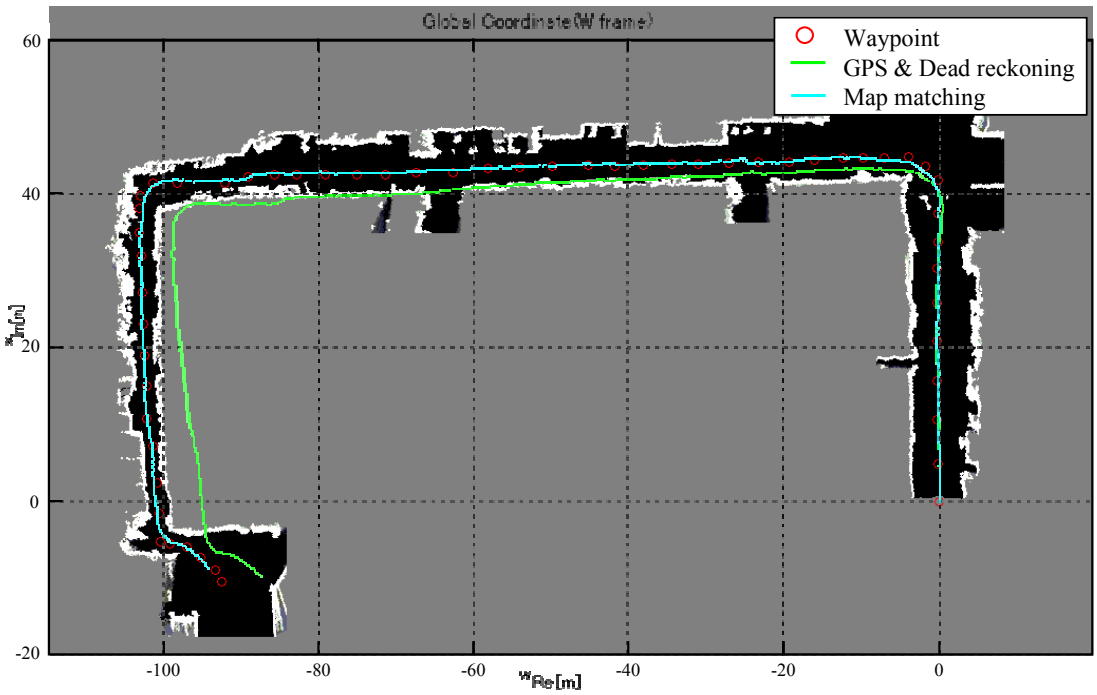


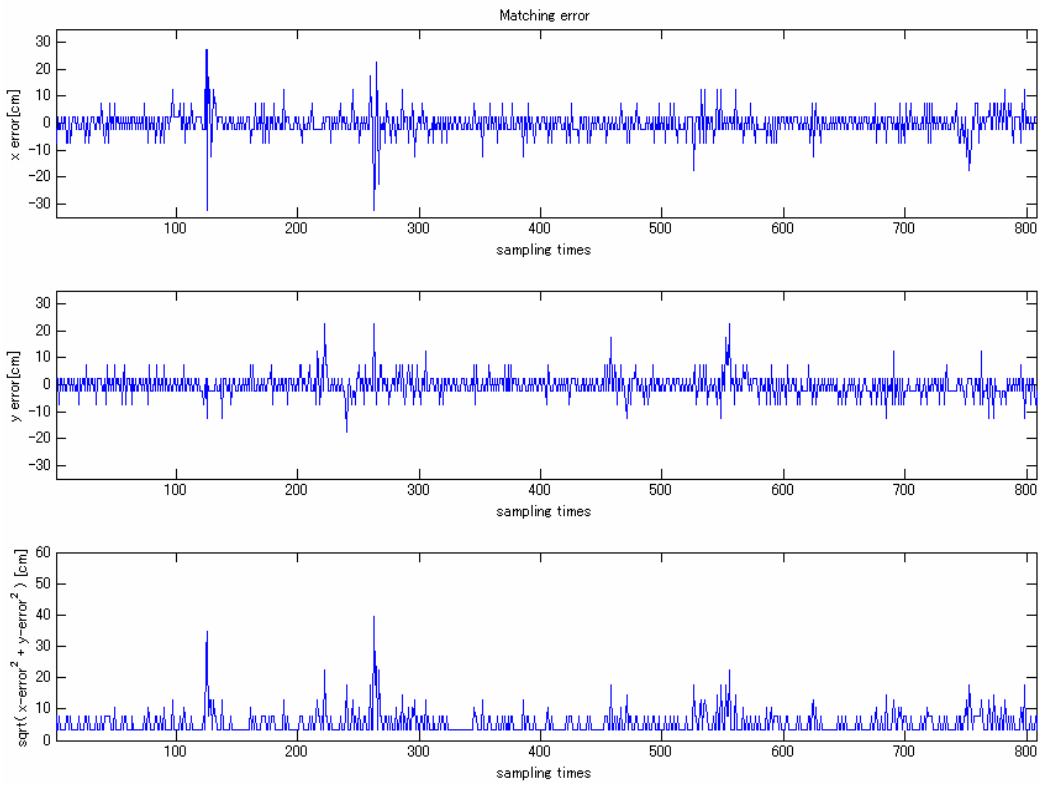
Fig.17 Experimental environment

Fig.18 Global probability map

シミュレーション実験と同様に，移動ロボットがリアルタイムで生成する確率マップは 240×240(12m×12m)サイズ，最大速度は 1km/h，サンプリング間隔は 0.8sec とする．これよりグローバルマップの限定処理から得られる探索対象データは 260×260 (13m×13m)サイズと設定する．また，ジャイロ補正処理はジャイロの特性から 170 サンプル毎に行い，確率マップが全面に生成される 30 サンプリング目からマッチング処理をスタートさせる．Fig.19 (a)に走行結果，(b)にマッチング結果を示す．



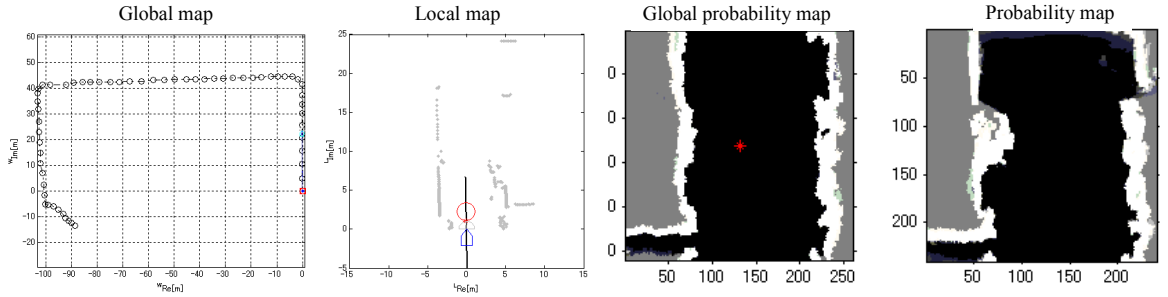
(a) Estimated Trajectory



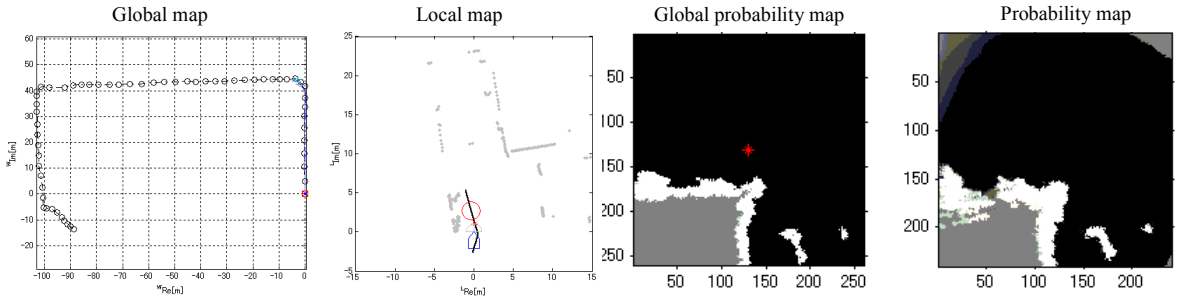
(b) Estimation errors

Fig.19 Experiment result

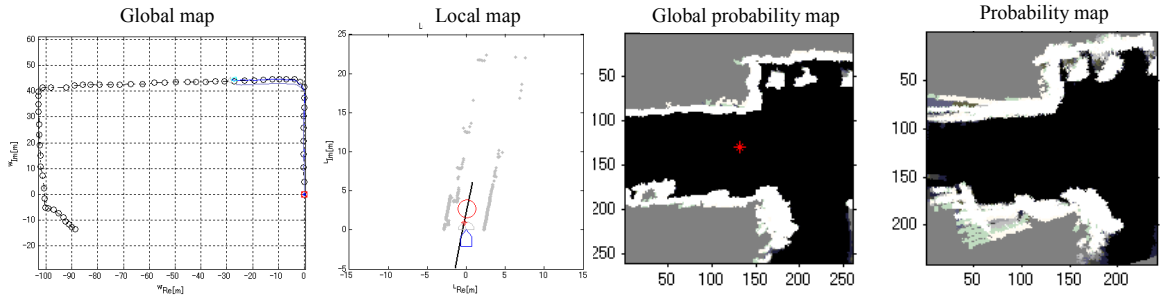
Fig.20 にグローバル座標系，ローカル座標系，自己位置補正マッチングを 100 フレームごとに 800 フレームまで示す．Fig.21 に 170，340，510，680 フレームに実行されたジャイロ補正の処理結果を示す．Fig.22 に実装した自律ナビゲーションシステムの処理時間を示す．



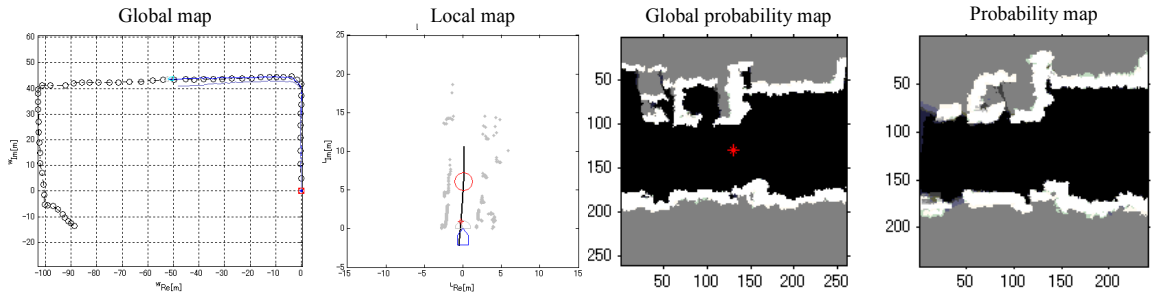
100 frame



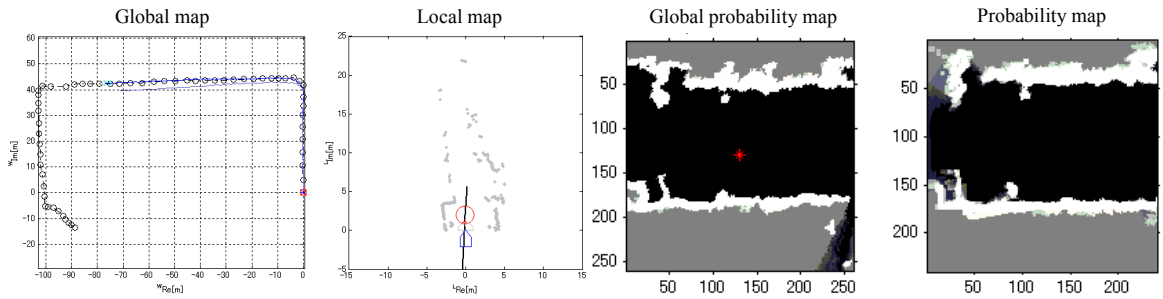
200 frame



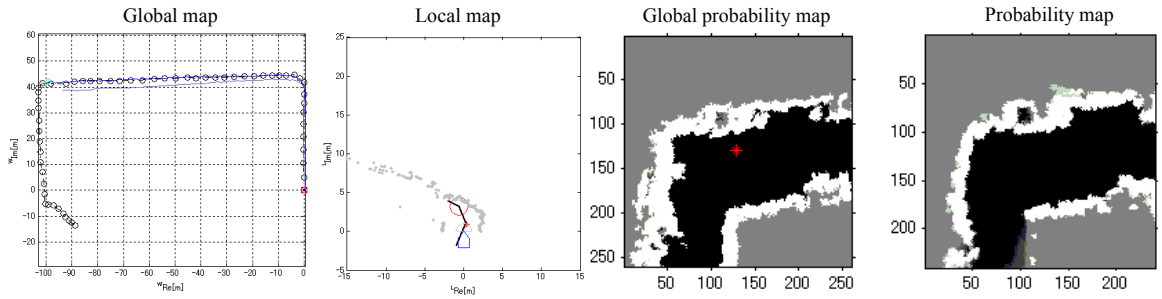
300 frame



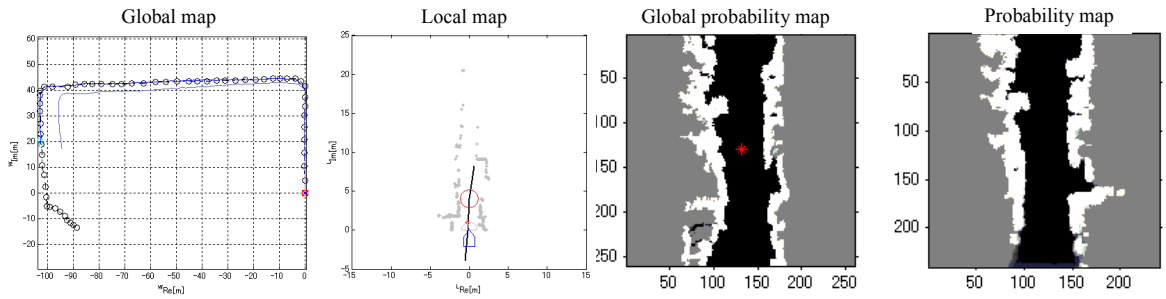
400 frame



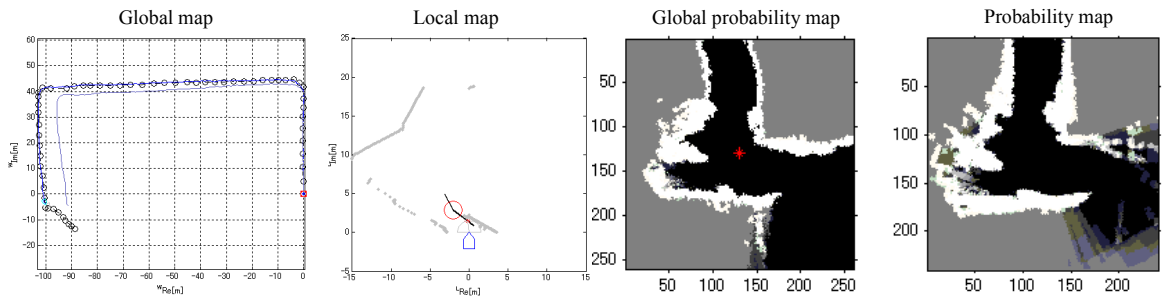
500 frame



600 frame



700 frame



800 frame

Fig.20 Template matching result

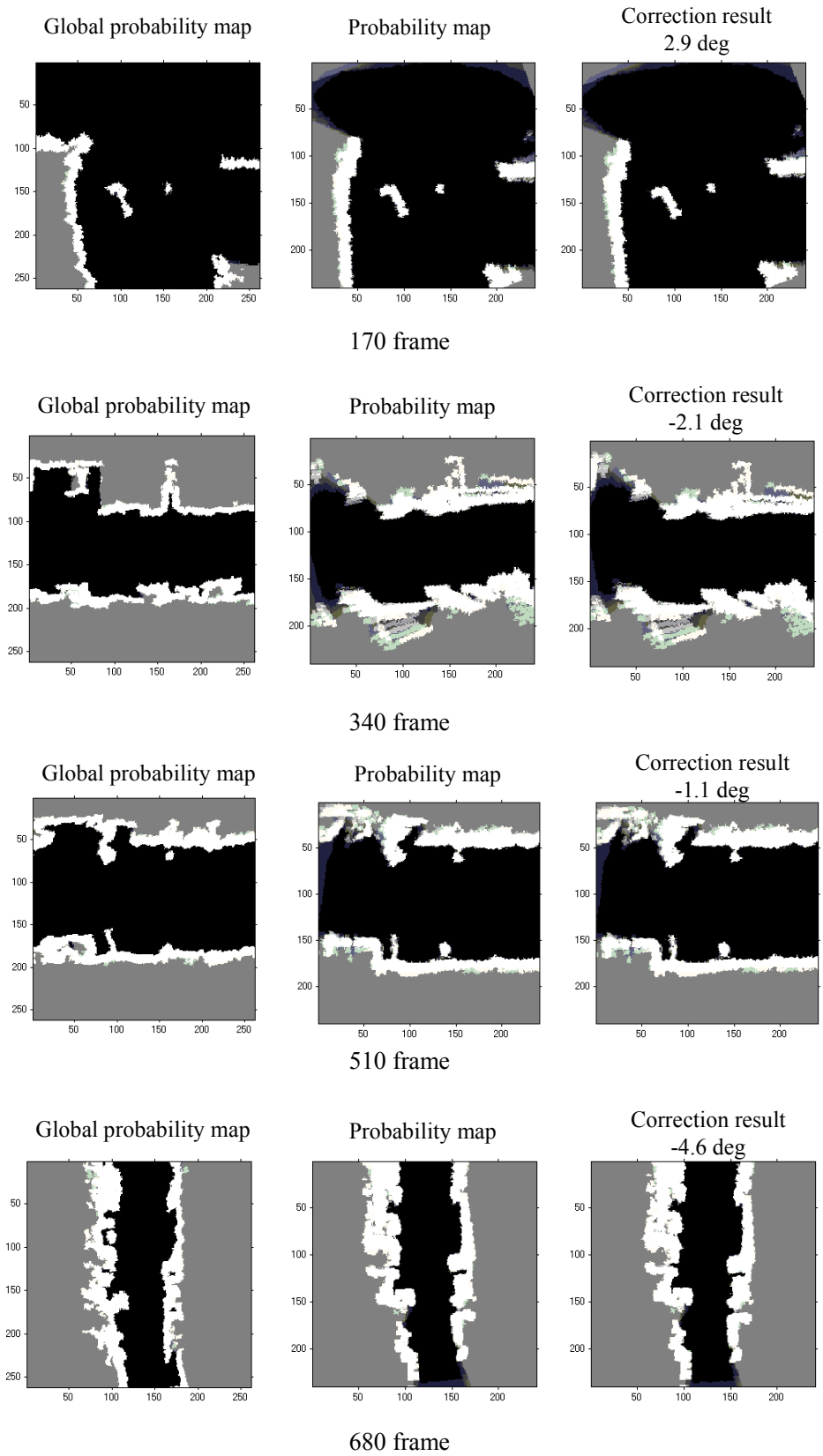


Fig.21 Gyro error correction

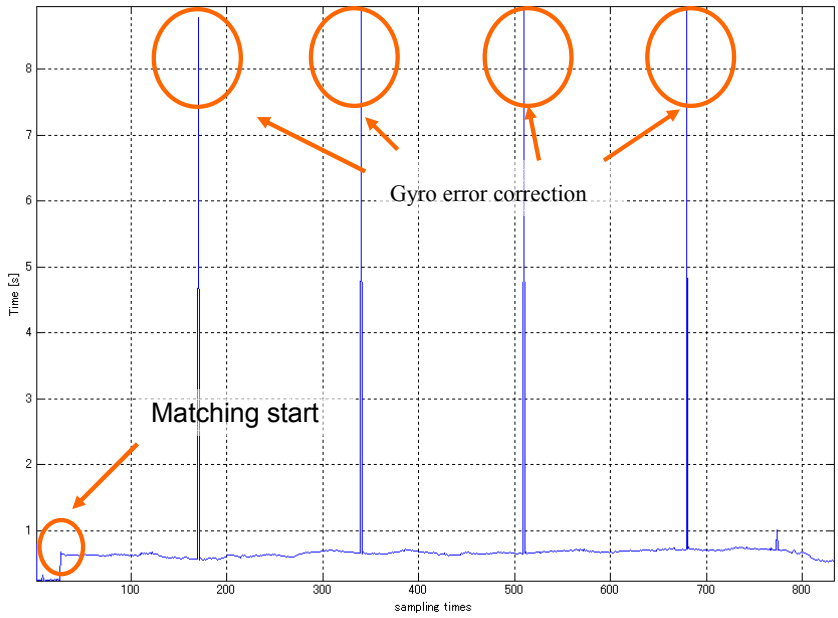


Fig.22 Processing time of implemented system

移動ロボットは，約 12 分の走行時間で実時間制御を行い，安全にゴール地点へ到達することができた．提案したシステムは，リアルタイムで生成した確率マップと事前に生成したグローバル確率マップをマッチングすることにより，自己位置を推定し，補正することができた．自己位置補正及びジャイロ補正に影響は無かったが，Fig.20 に示すように，リアルタイムで生成した確率マップに乱れが若干生じる区間があった．これは振動による車速計・ジャイロのノイズと 0.8sec と設定したサンプリング間隔が原因だと考えられる．また，従来の手法[6]で得られる自己位置の推定結果は，10 点目の Waypoint で走行が不可能になることが確認できた．これは，実験環境では，ほとんど GPS を受信することができなかったため，車速計・ジャイロの積分誤差が影響したものと考えられる．

第８章　　むすび

本研究では日常生活環境下における長距離移動を目的とした自律ナビゲーションシステムを提案し，実機へ実装を行った．実機実験を通し，事前に生成したグローバル確率マップと，リアルタイムで生成した確率マップを NCC によりマッチングをかけることで，約 200 メートル近い長距離のコースを安全に自律走行できることが確認できた．今後は，システムの高速化及び，カメラなど他の障害物検出センサを確率マップへ統合し，システムの拡張を行いたい．また，NDT マッチング[16]など，他のマッチング方法との比較を行い，本手法の有用性を検証していきたい．

参考文献

1) 畑瀬他：インテリジェント車いすロボット TAO Aicle，日本ロボット学会誌，Vol.24，No.2，pp.18-20，2006

2) 神山他：視覚障害者向けインテリジェント車椅子"ひとみ"の研究開発，電子情報通信学会技術研究報告，Vol.106，No.408，pp.23-28，2006

3) インターネットウェブサイト　つくばチャレンジ　：http://www.robomedia.org/challenge/index.html

4) Ochieng, W.Y. et al.: Integration of GPS and dead-reckoning for real-time vehicle performance and emissions monitoring, GPS Solutions, Vol.6, pp.229 – 241, 2003

5) Miyazaki, Y. et al.: Outdoor waypoints navigation for intelligent wheelchair by using Differential GPS and INS, SCIS and ISIS 2004, TUP-4-5, 2004

6) 沢辺他：自律走行制御のための Simulink ブロックセットの開発と実装，　第 52 回日本知能情報ファジイ学会知的制御研究部会，pp. 1-2，2007

7) Sawabe, W. et al.: Application of particle filter to autonomous navigation system for outdoor environment, SICE Annual Conference 2008, pp. 93-96, 2008

8) 後藤善孝：屋外環境下での自律ナビゲーションシステムへの particle filter の適用，法政大学大学院工学研究科紀要，Vol. 50，2009

9) Castellanos, J. et al.: Simultaneous localization and map building for mobile robots: a landmark based approach, IEEE Conf. Robot. Automat Workshop, W4, 2000

10) 樽床祐樹 : Simultaneous Localization And Map building による自律ナビゲーションシステムの開発，法政大学大学院工学研究科紀要，Vol. 48，2007

11) S. Thrun: Learning occupancy grids with forward sensor models, Autonomous Robots, Vol. 15, pp.111-127, 2003

12) 原口他：センサの見通し可能性を考慮した確率的モデルによる障害物地図の作成，日本ロボット学会誌，Vol. 25，No.4，pp. 632-642，2007

13) J. P. Lewis: Fast template matching, Vision Interface, pp. 120–123, 1995

14) 藤本他　：つくばチャレンジのための自律移動ロボット：JW-Future，計測自動制御学会論文集，Vol.46，No.1，pp.52-54，2010

15) Wilson, P.A. et al.: A line of sight counteraction navigation algorithm for ship encounter collision avoidance, Journal of Navigation, Vol.56, No.1, pp.111-121, 2003.

16) Biber, P. et al.: The Normal Distributions Transform A New Approach to Laser Scan Matching Proceeding ,2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.2743-2748，2003

付録

本論で提案した長距離移動のための自律ナビゲーションシステムの検証実験に用いた移動ロボットについて述べる．この移動ロボットのベースは，関東自動車株式会社製の 4WD 車いすを用い，各種センサを搭載することで自律ナビゲーションを実行可能にしている．

Laser rangefinder の仕様

本研究で使用した Laser rangefinder である，Sick 社製のレーザ距離計 LMS200 の仕様を Table.2 に示す．

Table.2 レーザ距離計の仕様

角度分解能	0．25 度/0．5 度/1 度　設定可能
外形寸法	155×210×156mm　（W×H×L）
インタフェース	ＲＳ232/ＲＳ422
レーザ保護クラス	1（本質的に安全）
最大角度	180 度
使用周囲温度	0 度～＋50 度
保護構造	ＩＰ65
測定距離(m)	80
分解能(mm)	10
応答時間	53ms/26ms/13ms 角度分解能による
保管周囲温度	-30 度～＋70 度
電源電圧	24Ｖ　ＤＣ±15％　（最大リップル 500mV）
スイッチング出力	PNP3 点；代表値 24Ｖ　ＤＣ
システム誤差	代表値±15mm
伝送速度	9．6/19．2/38．4/500 kbps
重量	約 4．5ｋｇ
備考	ＬＭＳ200 は，レーザレーダのように周囲を 2 次元的にスキャンする，非接触レーザ測定システムである．温度範囲 0 度～50 度で作動し，アクティブ・スキャニング・システムのため，パッシブの機器のようにターゲットを必要としない．

Optical fiber Gyroscope の仕様

本研究で使用した Optical fiber Gyroscope である，日立電線株式会社製の光ファイバージャイロ HOFG-3 の仕様を Table.3 に示す.

Table.3 光ファイバージャイロの仕様

項目	単位	規格
外観寸法	mm	88×88×65
インタフェース	—	RS232C
角速度測定範囲	deg/s	±100
角度測定範囲	deg	±360
電源電圧	V	＋10～＋18
最小検出感度	deg/s	0．05 以下
角速度直線性	%	±1．0 以内
零点ドリフト	deg/h	10 以下
起動時間	min	1 以内
電源消費電流	mA	500 以下
使用温度	℃	-10～70
使用湿度	%	20～80RH，結露しないこと
保存温度	℃	-20～70
保存湿度	%	10～80RH，結露しないこと

Speedometer の仕様

本研究で使用した車速計は，ロータリーエンコーダ(50P/R)と H8 マイコン(H8 3052)から構成されている．Fig.23 に回路図，Fig.24 に移動ロボットの駆動部におけるロータリーエンコーダの設置方法について示す.

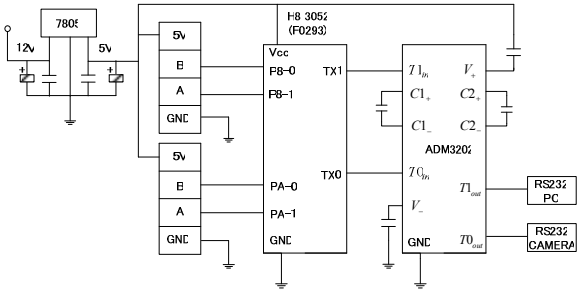


Fig.23 車速計の回路

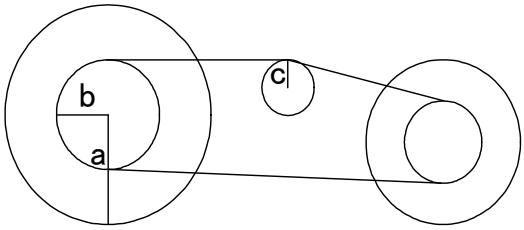


Fig.24 ゴム，後輪の内輪，外輪の半径

H8 マイコンのソフトウェアにより，50P/R のロータリーエンコーダを 4 逓倍処理することで 200P/R として扱っている．またこの回路から移動ロボットに設置されたカメラへ信号を送り，ズーム設定を行うこともできる.

ロータリーエンコーダの出力パルス(移動距離)をコンピュータ側で処理し，速度へ変換する．パルスの計算方法は後輪の外輪 A の半径を a(m)，後輪の内輪 B の半径を b(m)，ゴム C の半径を c(m)とした場合，後輪の内輪 B が 1 回転する時ゴム C は b/c 回転する．そのためゴム C が 1 回転する時後輪の内輪 B は c/b 回転する．ここで後輪の内輪 B が 1 回転する時 $2\pi a$ (m)進むので，ゴム C が 1 回転した時 $2\pi a \cdot c/b$ (m)進むことになる.

ここでゴム C が 1 回転した時に発生するパルス数を測定した結果, 左が 182(パルス/回転), 右が 184(パルス/回転)となる．そこで 1 パルスあたり左は $2\pi ac/182b$ (m)，右は $2\pi ac/184b$ (m)進む．よって，

$$左速度＝平均パルス\times\frac{2\pi a\times c}{182\times b}\times\frac{1}{0.008}$$

$$右速度＝平均パルス\times\frac{2\pi a\times c}{184\times b}\times\frac{1}{0.008}$$

となる(0．008(s)；ロータリーエンコーダのサンプリングタイム)． ここで a=0．175(m), b=0．055(m)，左のゴムの半径 c=0．0165(m)，右のゴムの半径 c=0．017(m)なので

$$左速度＝平均パルス\times\frac{2\pi\times 0.175\times 0.0165}{182\times 0.055}\times\frac{1}{0.008}$$

$$右速度＝平均パルス\times\frac{2\pi\times 0.175\times 0.017}{184\times 0.055}\times\frac{1}{0.008}$$

となる (実際には実験値である 182，184 を実験により変化させて精度を増す).

Differential GPS の仕様

本研究で使用した Differential GPS である, Trimble 社製　BD950 の仕様を Table.4 に示す.

Table.4 GPS 仕様

項目	規格
寸法	100mm×80mm×17mm
電源	+4．5 VDC～+32 VDC 5 VDC で通常 1．0 ワット (L1 のみの場合)5VDC で通常 1．5 ワット(L1/ L2，RTK の場合)
捕捉	24 チャンネル　L1，C/A，L1/L2 全波長位相
初期化時間	通常 30 秒未満
通信	RS-232 ポート×4，通信速度 115200bps 以下
出力フォーマット	NMEA-0183:AVR GSV HDT VGK VHD ROT GGK GGA ZDA VTG GST PJT PJK GSOF 出力および RT17 出力
動作温度	-40□～+75□
保管温度	-55□～+85□
振動	MIL 810 D，Tailored 動作時　ランダム 6．2gRMS，待機時　ランダム 8gRMS
機械的衝撃	MIL 810 D，動作時±40g，待機時±75g
DGPS 時	誤差 1m 以内，最大 20Hz，遅延 20ms

D/A 変換機の仕様

本研究で使用した CONTEC 社製の AD 変換器(ADI12-8(USB)GY)の仕様を Table.5 に示す.

Table.5 DA 変換器の仕様

入力形式	バス絶縁型電圧/電流出力
出力レンジ	電圧：　バイポーラ ±10V，±5V ユニポーラ 0 - 10V, 0 - 5V (出力電流 ±5mA) 電流：　0 - 20mA
出力インピーダンス	電圧レンジ：10W(Max．)
出力チャネル	4 チャネル
分解能	12 ビット
変換精度	電圧レンジ ±3LSB，電流レンジ ±5LSB
セトリング時間	電圧レンジ 10μsec，電流レンジ 20μsec
データバッファ	256K データ (262，144 データ)
内部サンプリングタイマ	10msec - 1，073，741，824msec

ベース車両の仕様

“Omnix2007”のベースとなった，関東自動車株式会社製 4WD 電動車いす Patrafour の仕様を以下の Table.6 に示す.

Table.6 Patrafour の仕様

形式・種類		KMC-0001 型・KMC
寸法 (mm)	前兆×全幅×全高	長 980×幅 625×高 930
	アームレストクッション高さ	シート上面より 150 から 300 5 段階調節
重量 (Kg)	本体	70
	バッテリ含む	100
前輪径(前後輪空気入りタイヤ) (mm kpa)		後輪 3．00-8-4PR (径 350 空気圧 294～314)
フレーム構造及び寸法(mm)		上下分割式：シート，アームレスト，レッグパイプ着脱 シート幅 390 シート奥行き 400 バックレスト高さ 435
リクライニング		5 段階調節機構(85～105 度)
駆動方式		4 輪駆動
制動方式		モータ発電及び電磁ブレーキによる制動方式
制御方式		ジョイスティックコントローラによる全方向電子制御方式
駆動モータ		DC ブラシレス 24V 280W×2(30 分定格)
バッテリ		LC-XC1238AJN 形鉛蓄電池
充電器		電子タイマー付自動充電方式(車載可能)
受電時間		8～12 時間
速度		最高速 6km/h 速度切り替え 3 段階：3．0，4．5，6．0 (4．5～6．0 変更可)
連続走行距離(km)		算出条件：常温で乗車重量 75kg，最高速度，平坦路直進時，バッテリ新品満充電から 70%放電まで 30km
実用登坂角度(度)		10
段差乗越高さ：前進時		80
溝乗越幅(mm)		100
最小回転半径(mm)		測定条件：フットレストの最も外側の軌跡 790
使用者最大体重(kg)(積載物含)		100 以下

謝辞

本研究を進めるにあたり，御指導を頂いた渡辺嘉二郎教授，小林一行教授に深く感謝します. 両先生には学部 4 年の 2007 年から 3 年間, Intelligent ground Vehicle Competition(IGVC) や RWRC(Real World Robot Challenge) 参加というすばらしい機会を与えていただき，研究の面だけでなく大変お世話になり，重ねて感謝申し上げます.

ビークル班の先輩である小竹亮太郎さん，大西将さん，後藤善孝さん，坂崎栄信さん，杉浦実さん，御園祐介さんには学会，IGVC，RWRC，参加にあたっての技術的・能力的な面から，プライベートに至るまで気に掛けて頂きまして，この場をかりて深く感謝申し上げます. 気さくに接してくださるだけではなく，適切な技術指導をしていただいたことで，精神的にも・能力的にも大きく成長できたと思います.

2008 年度 RWRC に初参加したときから共にビークル班で活動をしてきた，後輩の岡村麻代さん，中田貴人君，長谷川功介君，山川拓哉君に深く感謝します．世界的な疫病のため参加には至りませんでしたが，2009 年度 IGVC 参加へ向けた取り組みは，皆さんの協力のおかげで，やりきることができました．本当にありがとう．またそれと同時に，私の力量不足から後輩に迷惑をかけることが多々あったため，申し訳なくも思います．今後も中田貴人君，長谷川功介君，山川拓哉君は，大会参加へ向けた活動において，お互いの長所を認め合い，短所を補い合うことで，チームの中における個の力を高め，立派なエンジニアになってほしいと思います．期待しています.

最後に，学部・大学院含め 6 年間，不自由なく勉学に集中できる環境を提供していただいた両親・家族に感謝します.

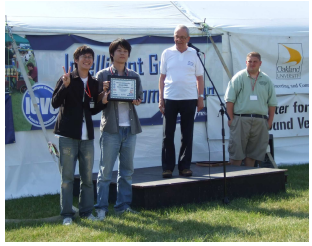
沢辺　航

活動実績

The 15th Annual Intelligent Ground Vehicle Competition

Design Competition　Presenter 担当

Design Competition　準優勝



The 16th Annual Intelligent Ground Vehicle Competition

Navigation Challenge　担当

Design Competition　担当

Navigation Challenge 3 位

Real World Robot Challenge (RWRC) 2008

Team Leader

電子回路担当

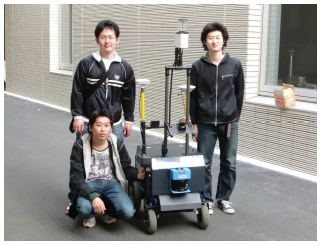
プログラム担当

The 17th Annual Intelligent Ground Vehicle Competition

Team Leader

Navigation Challenge　担当

世界的疫病のため出場辞退



Real World Robot Challenge (RWRC) 2009

電子回路担当

プログラム担当

研究業績

2007 年度（学部 4 年）

【国内会議】

・IGVC2007 に向けた自律走行車の開発
　沢辺　航・坂崎栄信・小竹亮太郎・渡辺嘉二郎・小林一行
　第 51 回知的制御研究会(SIC07-1)

・自律走行制御のための Simulink ブロックセットの開発と実装
　沢辺航,　御園祐介,　渡辺嘉二郎,　小林一行
　第 52 回知的制御研究会(SIC07-2)

・Simulink による自律環境認識および自律走行制御システムの開発
　沢辺航，御園祐介，小林一行，渡辺嘉二郎
　第 30 回ファジィ・ワークショップ

2008 年度（修士 1 年）

【国内会議】

・IGVC2008 に向けた自律走行車の開発
　内海恒亮・沢辺　航・渡辺嘉二郎・小林一行
　第 53 回知的制御研究会(SIC08-1)

・電動車いすのための荷重移動操作インターフェースの開発
　沢辺　航・小林 一行・渡辺 嘉二郎
　第 31 回ファジィワークショップ

・レーザーポインタを使った電動車いす操作インターフェースの提案
　福田芳久，内海恒亮，坂崎栄信，沢辺航，大久保友幸，小林一行，渡辺嘉二郎
　第 33 回ファジィワークショップ講演論文集

【国際会議】

・Application of Particle Filter to Autonomous Navigation System for Outdoor Environment,
　Wataru Sawabe , Yoshitaka Goto, Kazuyuki Kobayashi, Kajiro Watanabe
　SICE Annual Conference 2008

・Application of Particle Filter to GPS Navigation System
Wataru Sawabe, Yoshitaka Goto, Kazuyuki Kobayashi, Kajiro Watanabe, Tomoyuki Ohkubo
SCIS & ISIS 2008

2009 年度（修士 2 年）
【国内会議】

・IGVC2009 へ出場へ向けた JAUS 準拠自律走行車の開発
山川拓哉, 沢辺　航, 長谷川功介,渡辺嘉二郎, 小林一行
第 55 回知的制御研究会(SIC09-1)

・画像認識による JAUS 準拠車両の走行制御
長谷川功介，山川拓哉，中田貴人，沢辺航，小林一行，渡辺嘉二郎
第 56 回知的制御研究会（SIC09-2）

・JAUS 準拠の自律移動ロボットの開発,
山川　拓哉，長谷川　功介，沢辺　航，中田　貴人，小林　一行，渡辺　嘉二郎
第 10 回システムインテグレーション部門講演会 (SI2009)

・移動ロボットのための全方位カメラ画像を用いた速度計測法
中田 貴人，山川 拓哉，長谷川 功介，沢辺 航，大久保 友幸，小林 一行，渡辺 嘉二郎
第 35 回ファジィ・ワークショップ