

NAIST-IS-MT0351113

修士論文

ステレオ視を用いた自己位置同定法 -屋内ナビゲーションロボットシステムへの応用-

増田 健司

2005年2月3日

奈良先端科学技術大学院大学
情報科学研究科 情報システム学専攻

本論文は奈良先端科学技術大学院大学情報科学研究科に
修士(工学) 授与の要件として提出した修士論文である。

増田 健司

審査委員： 小笠原 司 教授
木戸出 正繼 教授
金出 武雄 教授
加賀美 聡 助教授

ステレオ視を用いた自己位置同定法 -屋内ナビゲーションロボットシステムへの応用-*

増田 健司

内容梗概

本研究では移動ロボットの自己位置同定問題において3次元画像情報とオドメトリ情報によりロボットの位置を推定する手法を提案する。

ロボットにおける移動のための自律機能の一つとして自己位置同定問題は最も重要である。ロボットの自己位置同定問題は、ベイズ的なアプローチとマルコフ性の仮定を用いることでロボットの状態系列を表す確率密度関数を、状態遷移確率(状態遷移ダイナミクス)と観測確率(センサモデル)に分解し、表現することができる。

本手法ではステレオカメラからの3次元情報を用い、実環境を高さを含む2.5次元な立体的表現から自己位置認識問題にアプローチする。これにより通常用いられている平面センシングでは発見不可能な小さな障害物を観測できる。状態ダイナミクスおよびセンサモデルにはそれぞれ、オドメトリとステレオカメラからの観測データを用いる。距離画像による3次元情報を自己位置同定に利用するには、モデル化が複雑になるという問題がある。そこで本研究では、位置認識に必要な情報のみを抽出する事で、シンプルなモデリングを試みる。具体的には2次元平面に投影する事でロボットから障害物までの距離を各角度毎に求める。これをレンジプロファイルと定義する。このレンジプロファイルを環境地図とマッチングし、推定位置の確からしさを評価する。この時、取得した3次元情報により障害物の高さも考慮する事で、地図にない小さな障害物があっても推定結果に影響

* 奈良先端科学技術大学院大学 情報科学研究科 情報システム学専攻 修士論文, NAIST-IS-MT0351113, 2005年2月3日.

響を与えることなく，環境マップとマッチング行なうことが可能となる．また距離が進むにつれ，蓄積誤差により真値から外れるオドメトリにおいても，yaw 軸光ジャイロを用いることにより，そのモーションデータを補正する．回転方向の誤差は距離が進むにつれ位置同定に大きな影響を及ぼすため，高精度の自己位置同定を目指すためには，回転方向の誤差をなくすことは非常に重要である．ジャイロを用いるにより回転方向の蓄積誤差を大幅に抑えることができた．自己位置同定は以上のようなモデルをもとにパーティクルフィルタを用いて行なう．複数の候補点を配置し，その確からしさを評価する事で，先に述べたベイズアプローチによる確率密度関数を高速に近似することができた．

実際にこのアルゴリズムを移動ロボット NOMAD200 に実装し屋内実環境を走行する実験により評価を行なった．カメラからの環境データとオドメトリによる移動量を取得し，そのデータをもとにオフラインで自己位置同定を行い，精度を検証した．フロアのおおよそ一回り，距離にして 220[m] を走行し，誤差が蓄積せず最大位置誤差は 6[cm]，角度誤差は 6 度という精度で自己位置を同定可能なことが確認された．そこで自己位置同定，経路計画機能を統合し，ロボット本体に実装することで，オンラインで実環境内を走行し，同時に自己位置同定と経路計画を行なう事ができる移動ロボットシステムを構築した．

キーワード

自己位置同定，2.5 次元，ステレオ視，パーティクルフィルタ，移動ロボット

Stereo Vision System based Localization -Applications to indoor navigation robot system-*

Kenji Masuda

Abstract

Localization is a fundamental problem for an autonomous mobile robot.

The localization problem can be represented by a probability density function (PDF) which represents the robot state by using Bayes approach and the Markov assumption. The problem is divided into the state transition probability (called the motion model) and the perceptual likelihood (called the perceptual model).

This research use three dimensional information from a stereoscope system, and represents the environment using a 2.5 dimensional map. Therefore, this approach is able to observe small obstacles which might not be seen using a 2D planar sensor.

This research uses odometry and stereo vision sensor data for the motion model and the perceptual model. In applying the 3D data to the localization problem, the perceptual model can become complicated. Therefore, in order to simplify the model, this research reduces the dimensionality of information used in the localization process. Specifically, three dimensional data is projected on a two dimensional plane, and the distance within the field of view is computed from the robot to obstacles. This extracted information is named the "range-profile".

* Master's Thesis, Department of Information Systems, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-MT0351113, February 3, 2005.

The range-profile is matched with the existing map that represents a real environment, and a probability measure of the robot state hypothesis is evaluated. By considering the height of obstacles from the three dimensional information, this matching can be performed without influence from sensed obstacles which do not exist in the map. The error in the motion model is corrected the error using the yaw-axis of a fiber optical gyroscope. Errors in rotation are compounded with the distance traveled and therefore greatly affect the accuracy of position estimation. By using the gyros, rotation error was reduced significantly. The particle filter method is of localization applied to estimate the robot state with these models. The particle filter method , when combined with the motion and perceptual models is able to approximate an arbitrary PDF.

The localization algorithm was implemented on a mobile robot, NOMAD200, and were conducted in a real world environment experiments. The robot moved through the real world while collecting vision and odometry data over a 220[m] path for 13 sampled points, after collection, localization was performed offline. the maximum translation error in the localization estimate was 6 cm and orientation error was 6 degree. Localization using the stereo vision system does not accumulate error as the robot navigated. Finally, an online mobile robot system was developed with a path planning function.

Keywords:

localization, 2.5D, stereo vision, particle filter, mobile robot

目次

1. 序論	1
1.1 研究背景及び目的	1
1.2 関連研究	2
1.2.1 環境の観測	2
1.2.2 自己位置同定問題	3
2. ステレオ視によるパーティクルフィルタ自己位置同定	6
2.1 ベイズフィルタによる位置推定	7
2.2 ロボットモーションのモデリング	8
2.3 観測センサのモデリング	10
2.3.1 三角測量原理	10
2.3.2 センサモデルの生成	11
2.4 パーティクルフィルタ自己位置同定	13
3. 自己位置同定システムの実装	16
3.1 ノイズ除去	16
3.1.1 オクルージョンによるノイズの除去	16
3.1.2 外れ値ノイズの除去	17
3.2 観測データの2.5次元表現	18
3.2.1 カメラ座標系からロボット座標系への変換	19
3.2.2 デプスデータの離散化	20
3.2.3 各ピクセルにおける障害物の存在確率	21
3.2.4 ベイズ定理による障害物の存在確率の更新	23
3.2.5 レンジプロファイルへの登録	24
3.3 環境マップの作成	25
3.3.1 2次元グリッド環境マップ	26
3.3.2 環境マップのエッジ抽出	26

4. 実機による実環境での自己位置同定	30
4.1 ハードウェアおよびソフトウェア構成	30
4.1.1 NOMAD200	30
4.1.2 ステレオカメラ	31
4.1.3 ファイバーオプティカルジャイロ	31
4.1.4 ソフトウェア構成	33
4.2 オフライン自己位置同定	34
4.2.1 並進コースによる実験	34
4.2.2 並進・回転コースによる実験	34
4.2.3 考察	37
4.3 車輪のすべりの影響	38
4.3.1 車輪のすべり実験	39
4.4 地図に無い障害物の影響	40
4.4.1 走行実験	40
4.4.2 高さ情報の利用	42
4.5 オンライン自己位置同定	44
4.5.1 実際の走行結果	44
4.5.2 経路探索法	44
5. 結論	48
5.1 ステレオ視を用いた自己位置同定	48
5.2 今後の課題	49
謝辞	51
参考文献	54
付録	58
A. ベイズアプローチ	58
A.1 隠れ状態推定	58
A.2 マルコフ性の仮定	58

A.3 事後確率の漸化式	60
B. NOMAD200 の主な仕様	62
C. ステレオカメラ主な仕様	63
D. ファイバオプティカルジャイロの主な仕様	64

目 次

1	オドメトリ誤差説明図	9
2	モーションモデル	10
3	距離測定法説明図	11
4	レンジプロファイル説明図	12
5	距離による分散の決定説明図	13
6	パーティクルフィルタ説明図	14
7	アルゴリズムの挙動 1	15
8	アルゴリズムの挙動 2	15
9	視差ノイズ説明図	17
10	ノイズ処理説明図	18
11	グリッド中のデプスデータ	20
12	2.5次元マップ	21
13	シグモイド関数	22
14	カレントビジョンビュー: a) カメラ画像, b) 視差画像, c) グローバル座標に投影した各点, d) 各セルの確率	24
15	レンジプロファイルへの登録	25
16	マップ	26
17	マッチングエラー説明図	27
18	エッジマップ	29
19	NOMAD200	31
20	MEGA-DCS	32
21	JG-35FD	32
22	ソフトウェア構成図	33
23	走行コース 1	34
24	走行コース 1 における位置同定結果	35
25	走行コース 2	35
26	走行コース 2 における位置同定結果: a) 位置誤差, b) 角度誤差	36
27	走行経路の推定結果	36

28	走行経路の推定結果	37
29	実験コース	39
30	位置同定結果	39
31	実験環境	41
32	推定結果：a) 往路, b) 復路	41
33	位置推定誤差	42
34	高さ情報を加味した推定結果：a) 往路, b) 復路	43
35	高さ情報を加味した場合の位置推定誤差	43
36	オンライン自己位置同定	45
37	経路計画法説明図 1	46
38	経路計画法説明図 2	46
39	経路計画	47
40	隠れ状態推定のための動定期ベイジアンネット	60

表 目 次

1	NOMAD200 主な仕様一覧	62
2	MEGA-DCS の主な仕様一覧	63
3	ファイバオプティカルジャイロの主な仕様一覧	64

1. 序論

1.1 研究背景及び目的

現在，人間の生活する屋内環境でタスクを行なうロボットが多数あり，館内を案内するロボット MINERVA [1], RHINO [2] たけまる [3] や，来客に対してオフィス内をナビゲートする MOBSY [4] や改良 Pioneer-2 [1] などがある．このようなロボットに求められる基本機能として移動が挙げられる．与えられるさまざまなタスクに対して，移動は最も基本的な機能であり，実環境内を安全に移動するためには，環境に対する自分の位置を認識する自己位置同定能力が大変重要であり，移動ロボットにおいて非常に難しい問題である．

上述したようなロボットは基本的に静的な環境をレーザレンジやソナーセンサを用いて2次元に観測し，自己位置同定を行なうものがほとんどである．しかし，実環境には大小さまざまな高さを持つ物体が存在し，椅子，靴のように特定な位置を持たないものが多く存在する．移動ロボットが将来さまざまなタスクに対応するために，次に求められる課題は以下のようなものが挙げられる．

- 立体的に環境を観測する事．
- 距離による蓄積誤差に強い事
- 動的環境に強い事

1つ目の課題として物体把持や障害物回避を考慮した場合に，環境を立体的に認識する事は非常に重要となる．位置同定と物体認識で別センサを使い分けることも考えられるが，システムが複雑大規模となる．よって，3次元データを用いて自己位置を同定し，付加機能にもこのデータを流用できることが理想である．

次に課題では自己位置同定には従来オドメトリなどの動力学モデルが用いられるが，これらには本質的な観測誤差からくる蓄積誤差の問題がある．よって，これらの誤差を解消するようなアルゴリズムが必要となる．

最後の課題として動環境の影響が挙げられる．環境を立体的に観測するという事は，よりセンシティブに環境を捉えることができるという利点があるが，位

位置同定の際，地図にない障害物または，特定の位置を持たない物体の影響を大きく受けるといふ欠点にもなる．よって，このような問題にも対処していく必要がある．

本研究では目的として，上述したような問題を解決するような，実環境を立体的に捉えるオンライン自己位置同定システムの構築を目指す．

1.2 関連研究

1.2.1 環境の観測

ロボットの自己位置を同定しようとするとき，周辺の空間モデルをいかに取得するかが問題となる．ロボットが環境情報を取得するためにさまざまなセンサが用いられるが，これらは測定誤差の影響を受け，少しのエラーがのちのち大きなエラーとなってあらわれるため，位置同定の結果において大きな影響を与える．よって，センサモデルにおいて，これを解決するのが第一の鍵となるなる．

本研究は屋内環境における位置同定問題を想定しているため，主に用いられるセンサとして Time of Flight（光飛行時間計測法）によるレンジファインダやソナーセンサなどの能動的手法や，ステレオ視を利用するカメラなどの受動的手法がある．能動的手法は比較的精度良く測定可能だが，その多くが平面センシングに使われている．しかし，実環境には様々な高さを持つ障害物が存在する．よって，これらのセンサからの観測データは2次元情報しか持たず，真に3次元の環境データを取得できたとはいえない．

レーザレンジファインダをスイングさせ3次元情報を取得する手法も提案されているが[5]，装置が複雑で大型であり，計算コストも大きい事から本研究では扱わない．

ステレオ視は既知の位置関係にある二つ以上のカメラから同時に得られた画像により，三角測量の原理を用いて三次元復元をおこなう手法である．特徴として，カメラの解像度分の距離が一度に取得可能である．一般的に画像中の各点の対応点探索が問題になるが，現在では再帰相関演算法や，LR-check 法を用いて実時間で精度の良い距離画像の生成が可能になっている [6]．

よって本研究はこのステレオ視を用いることにより、実環境をより立体的に捉え自己位置認識問題に適用する事を考える。この際、カメラから得た観測データは高さ情報を含む 2.5 次元表現で実環境を表現する。センサにカメラを用いる事により、視野内に存在する障害物であれば、小さなものでも発見できるという特徴を持つ。これはある高さの平面的な情報しか取得できない他の 2 次元ベースのセンサとは異なり、より環境にセンシティブであるという利点を持つ。

1.2.2 自己位置同定問題

ロボットの位置推定問題において、最も一般的に問題を捉えると、ある直接観測不可能な内部状態 (隠れ状態) を持つシステムについて、一連の観測データの系列から、隠れ状態の系列 (の一部) を推定する問題の一種とみなすことが出来る。ここで、観測データの系列とは離散時刻 t における観測データ (一般的にはセンサからの観測ベクトル)、隠れ状態の系列は隠れ状態変数をまとめた状態ベクトルである。この問題は隠れ状態系列が移動する (複数) 物体の位置や姿勢で、観測センサがレーダやカメラであるような場合には (複数) 目標追従 ((multiple)target tracking) 問題と呼ばれ、アンテナアレイ信号処理やコンピュータビジョン分野で盛んに研究が行なわれているが [7][8][9]、それ以外にも、 $\theta(t)$ や $y(t)$ を様々に設定することで、本研究のような移動ロボットの自己位置同定 [10][11][12][13][14]、移動しながら会話する複数話者の音声の同時認識 [15]、対話システムでのユーザの状況推定 [16][17] など、幅広い範囲の問題をこの形に帰着させることができる。

1990 年代に確率的な枠組み [18] が提案され、地図を広げてゆきながら自分の位置を同定する SLAM [19]、CML [20]、の問題がロバストに解決できることを示した。現在ではこの確率的手法が広く使われており、大別して以下のような方法が提案されてきている。

1. カルマンフィルタによる方法
2. EM アルゴリズムによる手法
3. 壁やドアなどの識別可能な物体間の関係を位置づける手法

1のカルマンフィルタによる手法[18]は昔からよく使われてきた手法であり、最も大きな利点はオンラインでマップ上の全ての事後事象を推定できるという事である。信頼性のあるロボットの状態を与える、カルマンフィルタは確実性の高い推定をすることができる。それは自己位置同定に伴うナビゲーション機能にも非常に有用な特徴となる。

最も大きな制限はガウス分布に従うということである。特に観測ノイズが独立でガウス分布であるという仮定は制限の鍵になる。もし、見分けのつかないランドマークを見てしまうと、推定はマルチモーダルな分布となり、それは単一モードのガウシアンに一致しないというような欠点がある。実用上におけるこの手法を考えると上記のようなマルチモーダルな環境は十分に存在する。また本質的にノイズを含むセンサから情報にも大変影響を受けてしまう。非線形なモーションモデルは線形近似するために、各モーションは微小区間で区切るといったカルマンフィルタの拡張型手法[21]も提案されているがそれらにおいてもこの影響は例外ではない。

3においても壁[22]、ドア[23]、エッジ[24]やその他のマーカー[25]というランドマークにより自己位置を推定する手法であるが、一定数のランドマークが常に確認できる範囲内にロボットがいる事、また、1同様見分けのつかないランドマークに非常に影響を受ける手法であり、オンライン処理など実環境を見据えたシステムに適応するには難しい。

よって、本研究では2の期待値を最大化する手法を用いる。この手法の特徴としては計算が簡単であるため、実装が簡単で計算コストも比較的軽くオンライン処理に向いていることである。その中でも、パーティクルフィルタ (particle filter) による位置推定法は、実環境に点在するマルチモーダルな分布に対応することができる。これは膨大な状態空間に対して局限された事後確率分布をオンラインで効率よく表現・計算する方法である。(研究分野によってモンテカルロフィルタ (Monte Carlo filter) [26]、Condensation アルゴリズム [8] などとも呼ばれている。パーティクルフィルタの基本的なアイデアは、状態空間に値を持つ多数の粒子 (パーティクル) の状態空間中の分布によって確率分布を近似表現することである。粒子の確からしさを評価し、確率の高そうな場所を保ち、確率の低いと思われる

場所を棄却しながら，時間とともにパーティクルとその存在確率を伝播していく手法である．

2. ステレオ視によるパーティクルフィルタ自己位置同定

まず、本研究で扱う問題を一般的な形で定式化する。1.1でも述べたように、ある直接観測不可能な内部状態 (隠れ状態) を持つシステムについて、時間1から T における一連の観測データの系列 $y(1), \dots, y(T)$ から、隠れ状態の系列 $\Theta(1), \dots, \Theta(T)$ (の一部) を推定する問題の一種とみなすことが出来る。ここで、 $y(t)$ は離散時刻 t における観測データ (一般的にはセンサからの観測ベクトル)、 $\Theta(t)$ は隠れ状態変数をまとめた状態ベクトルである。本研究の場合には、 $\Theta(t)$ はロボットの位置及び姿勢であり、個々のロボットの位置や姿勢 $\theta_i(t), (i = 1, \dots, N)$ をまとめたものになる。ここで N はロボットの取り得る状態の最大数を示す。

この問題はベイズフィルタ (Bayes filter) を用いることにより定式化することができる、また、実際にベイズフィルタを用いるときには以下のような時間的に不変な3つのモデリングが必要となる。

1. ロボットの初期位置
2. ロボットのモーションモデル
3. ロボットのセンサモデル

これらをいかにモデル化するかが位置同定推定において非常に重要となる。従来これらモデリング法には2つの段階がある。1つは理想的モデリングであり、もう1つはノイズを考慮に入れたモデリングである。本研究では実環境でのロボットシステムを想定している。よってノイズを考慮したモデリングが必要となるため、このノイズを微小なガウシアンノイズにより生成し、ランダム変数より与えることによりモデリングする。

また、センサモデルにおいてはレンジセンサを用いれば比較的簡単にモデル化することができるが、本研究では3次元情報を扱うためカメラを用いる。このような非線形なセンサはモデルが非常に複雑になってしまうという問題がある。よって、本研究では、位置同定に必要な情報を観測データより抽出することで、センサをシンプルにモデリングすることを試みる。実際にはステレオカメラから三角測量により距離画像を求め、各角度毎に障害物までの距離を求める。本研究では

これをレンジプロファイルと呼ぶ。これをあらかじめロボットに与えた環境マップとマッチングさせることによりモデリングを行なう。

このようにして、自己位置同定問題を定式化し、実際のロボットへの実装を試みる。しかし、ここでも問題が挙がる。ベイズフィルタは、過去の全ての事象に対しての状態系列を推定することができる。本研究のような、各時刻 t でとりうるロボットの状態数 N が無数に存在する場合は、計算量が C^N の形で増えてゆく。従ってベイズアプローチによる計算を厳密に実行できるのは N が小さい範囲だけである。

パーティクルフィルタ (particle filter) は膨大な状態空間に対して局限された事後確率分布をオンラインで効率よく表現・計算する方法である。(研究分野によってモンテカルロフィルタ (Monte Carlo filter)[26], Condensation アルゴリズム [8] などとも呼ばれている。パーティクルフィルタの基本的なアイデアは、状態空間に値を持つ多数の粒子の状態空間中の分布によって確率分布を近似表現することである。

この方法を用いることにより、計算量の問題を回避することができる。また、パーティクルフィルタは 1.1 でも述べたように、マルチモーダルな環境に対してもロバストな手法であり、実環境下での自己位置同定において非常に適した手法である。このようにして、ロボットの確率密度関数を近似し、自己位置同定システム構築を行なう。

本論では、まず、ベイズフィルタに関して簡単に説明し、続いてモーションモデルおよびセンサのモデリング方法を述べる。最後にパーティクルフィルタの実装手順について説明する。実装の際には更にいくつか考慮すべき点があるが、それは 3 章で詳しく述べる。

2.1 ベイズフィルタによる位置推定

ベイズ的なアプローチでは時系列に時間 1 から T をとるとき、観測データの系列 $y(1), \dots, y(T)$ および隠れ状態の系列 $\Theta(1), \dots, \Theta(T)$ を確率変数として、それらの間の関係を確率密度関数によって捉えることで様々な推論を行なう。以下では簡単のために $y(1), \dots, y(T)$ を $y_{1:t}$ などと書く。

本研究では、観測データ $y_{1:t}$ から、 $\Theta(t)$ の事後分布 $p(\Theta(t), y_{1:t})$ を計算する問題を扱う。 $y(t)$ と $\Theta(t)$ の確率的な関係をモデル化するためにマルコフ性の仮定を用いると、事後確率 $p(\Theta(t), y_{1:t})$ は、

$$p(\Theta(t) | y_{1:t}) = \frac{1}{Z} p(y(t) | \Theta(t)) \sum_{\Theta(t-1)} p(\Theta(t) | \Theta(t-1)) p(\Theta(t-1) | y_{1:t-1})$$

という式で表すことができる。この式の導出についての詳しい説明は付録 A.1 に示す。

ここで $p(y(t) | \Theta(t))$ は観測確率であり、 \sum で囲まれる項は $p(\Theta(t) | \Theta(t-1))$ を成分とする状態遷移確率である、 Z は $\Theta(t)$ によらない正規化の定数である。

つまり、事後確率 $p(\Theta(t), y_{1:t})$ は、初期状態と状態遷移確率および観測確率の3つの分布により求めることができる。

ベイズアプローチは汎用的なアルゴリズムの枠組み（メタアルゴリズム）であり、具体的な問題に適用するためには、確率変数やその間の関係を具体的に決める必要がある。既に述べたように、本研究の問題においては、 $\Theta(t)$ は時刻 t における空間的位置と姿勢を表したものである。一方、カメラおよびオドメトリというセンサは決まっても、フィルタリングのための観測データ $y(t)$ として何を用いるかは考慮の余地がある。最も素朴な方法は、 $y(t)$ としてカメラやオドメトリが出力する生のデータを用いることであるが、その場合、観測確率は非常に複雑になる。そこで、全体の処理をいくつかの段階に分割して、まず、生のデータからロボットの位置、向きの状態の推定にとって重要な情報だけを抽出し、それを $y(t)$ として用いてフィルタリングする。

2.2 ロボットモーションのモデリング

モーションモデルにはオドメトリ情報をそのまま適応し、状態遷移確率を求める方法もあるが、基本的にオドメトリには各キャプチャー毎に Fig:1 に示すような誤差があり、そのエラーは積分誤差ととして蓄積してゆく。本研究では各キャプチャー毎のオドメトリ情報にガウス分布に従うランダムノイズをエラーとして加えてモーションモデルを定義する。

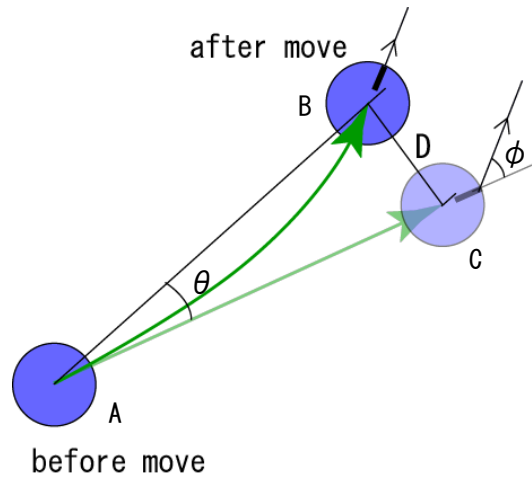


図 1 オドメトリ誤差説明図

従来のロボット運動学を 2 つの独立な 0 平均ランダム変数によって表すことができる．一つは移動量ノイズ，もう 1 つは回転ノイズである．今ロボットが A から C に移動するとする．しかし，実際の移動にはノイズがのり B に移動する事になる．この時，A と B を結んだ線と A の向いている方向の角度が θ である確率を P_{gs1} ，B の向いている方向と C の向いている方向の角度が ϕ の時の確率を P_{gs2} ，B と C の距離が D の時の確率を P_{gt} とすると，状態遷移確率 $p(\Theta(t) | \Theta(t-1))$ は

$$p(\Theta(t) | \Theta(t-1)) = P_{gs1} P_{gs2} P_{gt} \quad (1)$$

という式で表すことができる．Fig:2 に式 (1) により求めたモーションモデルを示す．三角をスタート地点とし， $5cm$ の移動量を与えたものである．確率を色の濃さで表しており，濃いほど確率が高い．

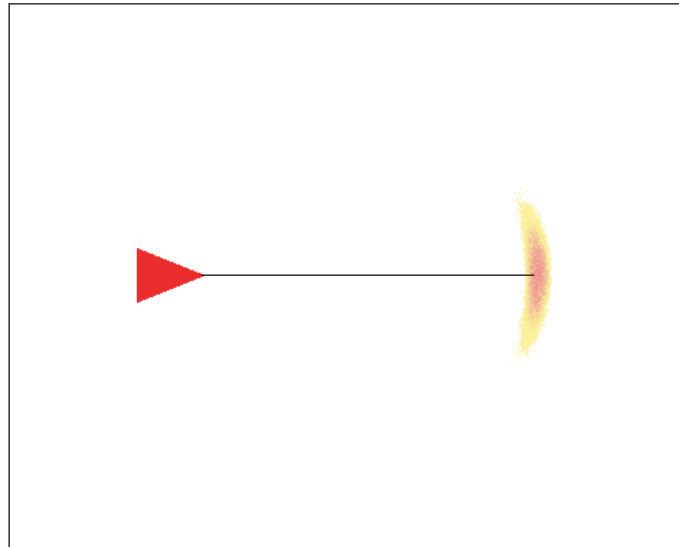


図 2 モーションモデル

2.3 観測センサのモデリング

2.3.1 三角測量原理

三角測量の原理を用いたステレオカメラによる距離測定法（3次元距離画像生成法）の説明を Fig.3 に示す。

2点から対象物への角度と2点間の距離が分かれば対象物の距離を計算することができる，2点の観測面上の位置を (x_a, y_a) , (x_b, y_b) , 注目点 P の3次元位置を (x, y, z) とすると式 (2) , 式 (3) , 式 (4) により注目点 P を求めることができる。

$$x = \frac{x_a L}{x_a - x_b} \quad (2)$$

$$y = \frac{y_a L}{x_a - x_b} \quad (3)$$

$$z = \frac{fL}{x_a - x_b} \quad (4)$$

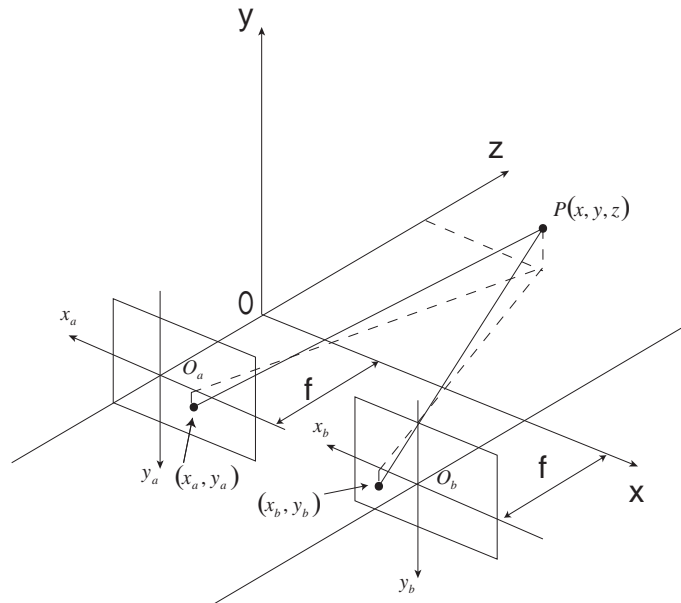


図 3 距離測定法説明図

2.3.2 センサモデルの生成

センサモデルには本章冒頭で述べたように，ステレオカメラデータから位置推定にとって重要な情報を抽出する必要がある．

本研究では抽出した情報をまとめたものをレンジプロファイルと呼ぶ．レンジプロファイルとは、ステレオカメラセンサの視野を分割し，各角度におけるロボットから障害物までの距離をまとめたものであり，これをマップとマッチングさせることにより，観測確率を求める．観測確率 $p(y(t) | \Theta(t))$ は以下のような式で表すことができる．

$$p(y(t) | \Theta(t)) = \prod_{i=1}^N P(d_i | s_t)$$

ここで， d_i は Fig:4 に示ようにレンジプロファイルにおいて角度 $i\theta/N$ でのロボットから障害物までの距離である． θ はステレオカメラの視野角であり， N は視野の分割数を表す．

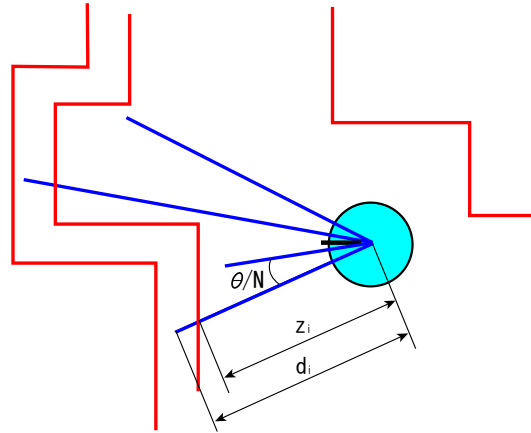


図 4 レンジプロファイル説明図

ある角度における観測距離 d_i の確率 $P(d_i|s_t)$ はガウス分布により以下のように与えることができる。

$$P(d_i|s_t) = \frac{1}{\sqrt{q\pi\sigma^2}} \left(e^{-\frac{0.5(d_i-z_i)^2}{\sigma^2}} \right)$$

z_i はマップにおけるロボットの位置候補点から角度 i での障害物までの距離である。 σ^2 はステレオカメラより得られる距離画像における分散で、以下の式であたえる

$$\sigma^2 = T^{MIN} + \mu d_i^2$$

T^{MIN} は探索エリアの最小サイズを表す。

ステレオカメラの性質上、遠くほど精度が悪い。つまり分散を距離に依存して決定する必要があり、その説明を Fig:5 に示す。視野を分割した時に端から i, j 番目のレンジプロファイルに登録されている距離を d_i, d_j とすると、 i, j では Fig:5 のよう距離に応じて探索領域を変更しマッチングを行なう事で、各角度におけるマッチングの信頼性を平滑化させる共に、計算コストの軽量化を図る。

このようにして与えられた状態遷移ダイナミクス $p(\Theta(t)|\Theta(t-1))$ およびセンサモデル $p(y(t)|\Theta(t))$ を用いて、各パーティクル毎 $p(\Theta(t)|y_{1:t})$ を求める。その中で最も確率の大きいものをロボット位置とする。

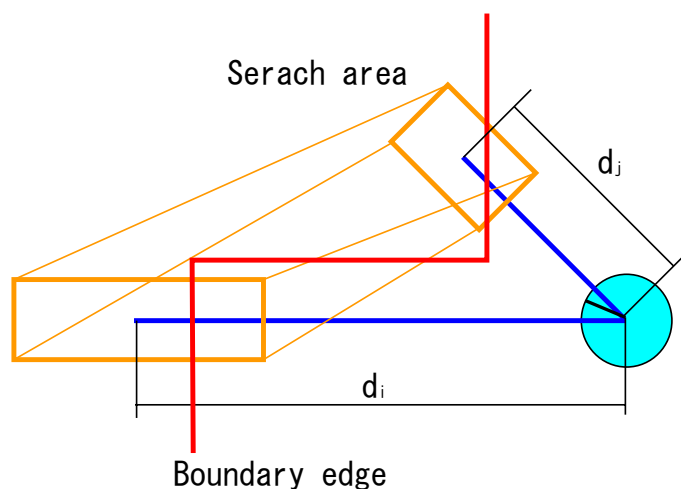


図 5 距離による分散の決定説明図

2.4 パーティクルフィルタ自己位置同定

状態変数のダイナミクスを表す状態遷移確率 $p(\Theta(t) | \Theta(t-1))$ および観測確率 $p(y(t) | \Theta(t))$ が与えられたときに、 $p(\Theta(t) | y_{1:t})$ を近似計算するためのパーティクルフィルタの具体的なアルゴリズムとしては様々なものが提案されているが、典型的なものは下記のような、importance sampling と bootstrap selection を組み合わせた方法である [27] [14].

処理の流れを以下に示すと共にイメージを Fig:6 に示す .

初期化 状態空間の値を持つ粒子を M 個用意し、各粒子の状態空間地をランダムに (初期確率が与えられている場合はそれに従って) 初期化する .

逐次計算 時間を追って以下を繰り返す .

Predict 状態遷移確率に従って、各粒子の状態をランダムに遷移させる .

Measure ある粒子の遷移した状態を Θ とするとき、観測値 $y(t)$ を使って $p(y(t) | \Theta)$ を計算し、粒子全体について和が1になるように正規化した値をその粒子のスコア (importance factor) とする .

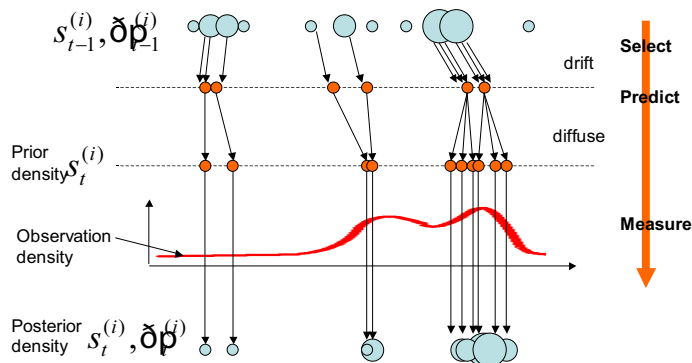


図 6 パーティクルフィルタ説明図

Reselect 粒子のスコアを確率として用いて，M 個の粒子のをリサンプリングする．

なお，ほぼ同様のアルゴリズムによって， $p(\Theta_{1:t}(t) | y_{1:t})$ などを近似計算することも可能である [27]．

もう少し詳しくアルゴリズムの挙動について述べる．Fig:7, Fig:8 は簡単のため 1 次元のロボットを例にとり，パーティクルフィルタのアルゴリズムの挙動を示したものである．

基本的にロボットと環境の状態 (地図とロボットの位置) を上段に，そのときの確率分布を下段の二段構成で記している．はじめに Fig:7 では，初期状態を示しており，パーティクルは一様にランダムに配置されている．ここから自己の移動モデルから確率分布を横に移動するとともに移動の誤差モデルを用いて統計的に伝播したものが Predict の欄である．続いて Measure の欄でセンサーを用いて世界を観測する．このときには観測の確率モデルを用いて，ロボットの観測データと，ロボットが各パーティクルの位置にいるとしたときの予想される観測データとの相関をとり，その場所にいる確からしさを計算する．最後に Reselect のフェーズで乱数を用いながら，確率の高そうな場所を残していく．

次に，Fig:8 では，最上段に Fig:7 の最下段を再び記す．Predict の欄では再びロボットは少し移動し，移動モデルから確率分布を横に移動しながら統計的伝播

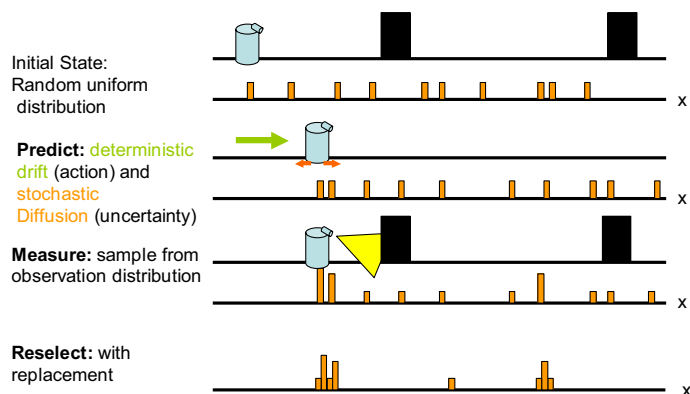


図 7 アルゴリズムの挙動 1

を行う．次に Measure で観測により相関をとり，また Reselect で確率の高そうな場所を残していく．以上のような流れで自己位置同定を行なう．

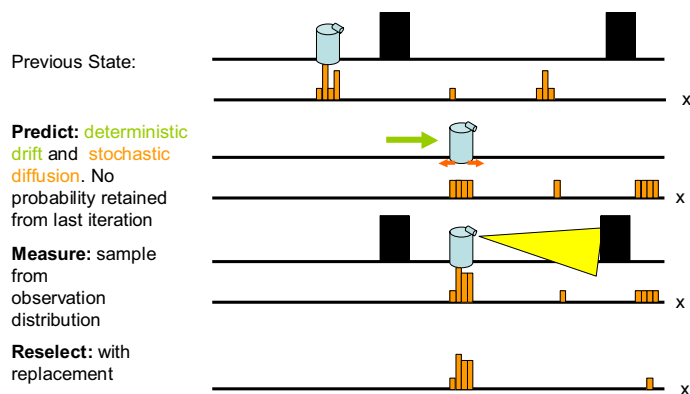


図 8 アルゴリズムの挙動 2

3. 自己位置同定システムの実装

本手法を実装するにあたり，いくつかの考慮すべき点が挙がる．対処すべき問題は以下の二つであり，

- 距離画像中のノイズ除去
- 3次元観測データの2.5次元グリッドマップへの変換
- 環境マップの作成

である．本章ではまず，本研究で扱う2種類のノイズ処理についての対処法を述べた後，ステレオカメラ画像の2.5次元グリッドマップへの変換法，環境マップ生成について順に述べる．

3.1 ノイズ除去

2.3.1のような方法を用いて得た距離画像内にはノイズが存在する．本研究ではロボットの自己位置を同定する時にステレオカメラから得た距離画像と環境マップをマッチングさせる．この時，距離画像中にノイズが存在する有無が地図とのマッチング精度に大きく影響してくる．よって，距離画像中のノイズはできる限りその存在を特定し除去する必要がある．

本研究では距離画像中に存在するノイズを以下のように2種類に分類した．1つはエッジ部分に起こるオクルージョンノイズ．もう一つは外れ値ノイズとし，各ノイズごとに除去処理を行なう．

3.1.1 オクルージョンによるノイズの除去

まず，オクルージョンノイズに対する処理について説明する．Fig.9に示すようにオクルージョンにより，左右一方のカメラでしか見えない箇所が存在した場合 Fig.9 中のような誤ったデプスデータが算出されてしまう．本研究ではこれをオクルージョンノイズと呼ぶ．

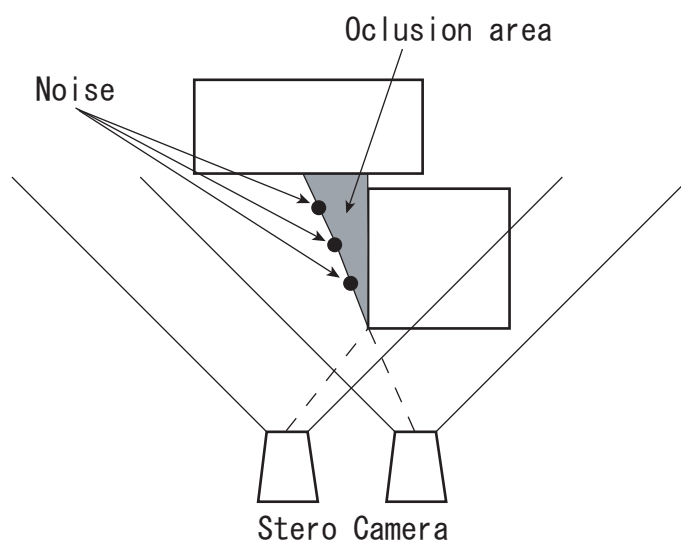


図 9 視差ノイズ説明図

連続し隣り合うデプスデータは次に述べる外れ値ノイズでない限り，緊密に並んでいるはずである．よって，隣り合う2点の距離がある一定以上の大きさを持つ時これを視差ノイズとして除去した．本研究では評価値を経験的に $50[mm]$ とし，処理を行なう．

3.1.2 外れ値ノイズの除去

次に外れ値ノイズの除去について述べ，その方法を Fig.10 に示す．

焦点の中心 O ，評価点を A ，基準近隣点 P からなる三角形 OAP と，撮像面上の評価点の位置 a ，基準近隣点 p からなる三角形 Oap の相似比をもちいて，閾値 S を式 (5) のように定義する．

$$S = \frac{z}{df} \quad (5)$$

基準近隣点までの間にある他の点と評価点との距離 D を評価する．評価点がノイズでなければ，距離 D は閾値よりも小さくなるはずである．この評価を評価

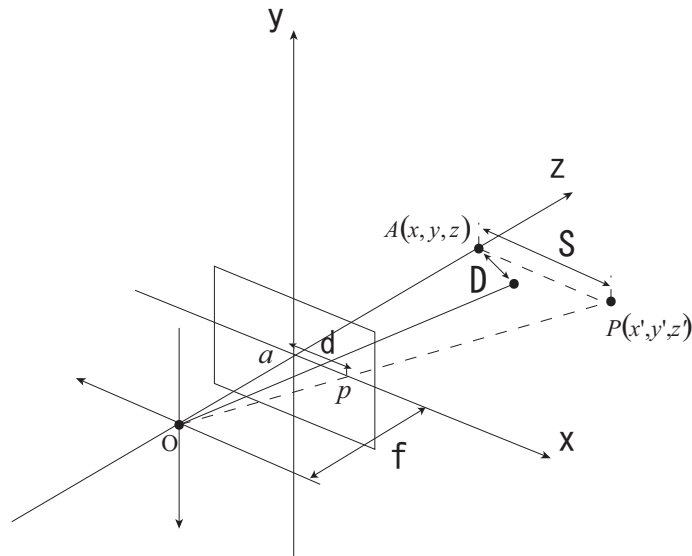


図 10 ノイズ処理説明図

点に対し行い．評価点が近隣の点に対して外れた点かどうかを判定する．つまり，ノイズかどうか判定する．本研究では近隣に存在する点が 20 点以下の時その点をノイズとして判定した．ノイズであると判定された点は削除し，これを距離画像中の全ての点に対して行うことでノイズを除去することができる．

3.2 観測データの 2.5 次元表現

観測確率の導出は 2 で述べたように，ステレオカメラからの観測データと地図とのマッチングを行う．よって 2 次元の地図とマッチングさせるため，ステレオカメラより取得した 3 次元データをグリッドマップ表現に加工する必要がある．

まず，2.3.1 で述べた方法により得たデプスデータを地図と同じピクセルスケールのグリッドマップに投影する．この時，デプスデータが持つ高さ情報も各グリッドが保持する事で 2.5 次元な観測視野マップ (以後 2.5 次元マップ) を作成する．さらに，2.5 次元マップの各グリッドに障害物の占有確率を持たせる．ステレオカメラから得られるデプスデータは 3.1 でも述べたとおり，非常にノイズの多いデー

タとなっている．そのため，ある瞬間瞬間での観測データのみで2.5次元マップを更新するには非常に信頼度が低い．よって本研究ではハイトマップをベイズの定理を用いて更新する事で信頼度を高める事でさらにノイズ影響の軽減を図る．

このようにして作成した2.5次元マップから，視野を分割し，各角度毎に障害物までの距離を判定する．つまりレンジプロファイルを作成する．

3.2.1 カメラ座標系からロボット座標系への変換

取得した距離データはカメラ座標系をもとにした座標である．よって，取得した3次元距離画像データを，ロボットを中心としたグローバル座標系に変換する必要がある．本研究においてカメラはロボットの頭部に搭載されており，前方下方の地面を見下ろすように設置されている．そのためカメラはロボットの進行方向と平行に固定されている，よって本研究ではヨー角を考慮せずに，ピッチ角とロール角の座標変換を行う．

座標変換は回転処理を行ってから平行移動を行う．地上面に対してカメラのピッチ角を θ とし，ロール角を ϕ とし，距離画像中のあるピクセルの座標を (x, y, z) とすると，回転処理後のカメラ座標系での座標 (x', y', z') は以下のように表すことができる．

$$\begin{aligned} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} &= \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \\ &= \begin{bmatrix} \cos \phi & -\sin \phi \cos \theta & \sin \phi \sin \theta \\ \sin \phi & \cos \phi \cos \theta & -\cos \phi \sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \end{aligned} \quad (6)$$

最後に平行移動をおこなう．本研究ではグローバル座標系（ロボット座標系）において，原点（ロボットの中心）より $(x, y, z) = (-50, 950, -250)$ の位置にカメラが設置されている．よって，グローバル座標系における各点の座標 (X, Y, Z) は以下のような式で表すことができる．

$$\left[X, Y, Z \right] = \left[x' - 50, y' + 950, z' - 250 \right] \quad (7)$$

3.2.2 デプスデータの離散化

デプスデータはステレオマッチングにより得られた視野内のオブジェクトの3次元位置を表すデータであり，単位は [mm] である．しかし，マッチングさせる地図のグリッドマップであるため，デプスデータをグリッドマップに投影する必要がある．本研究では1ピクセル100[mm]としている．つまり Fig.11 に示すように1つのピクセル中には複数のデプスデータが存在することになる．よって，視野内のデプスデータをグリッド毎に分離するつまり離散化する必要がある．

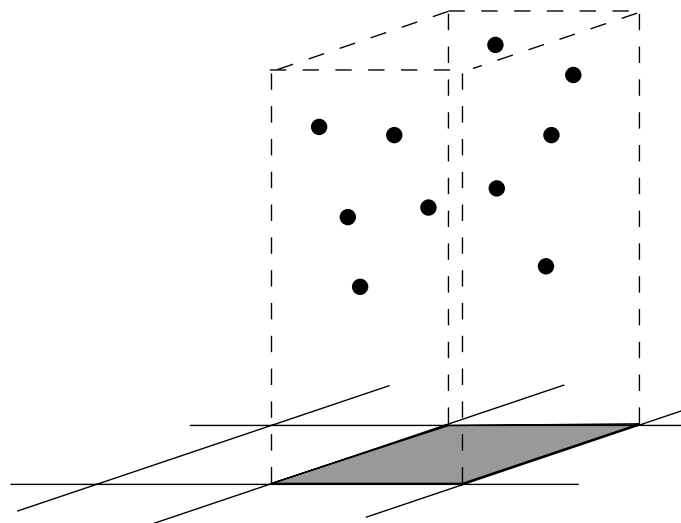


図 11 グリッド中のデプスデータ

本研究では単純にデプスデータを所属する各グリッドに落とし込む際，小数点以下の端数は四捨五入により切り捨て，くり上げをおこなう．その際，あるピクセル内のデプスデータで最も大きな高さを持つものの値を，そのグリッドの高さ情報として登録する．このようにしてカメラによる視野マップを生成し，これを 2.5 次元マップと呼び，Fig:12 に示す．

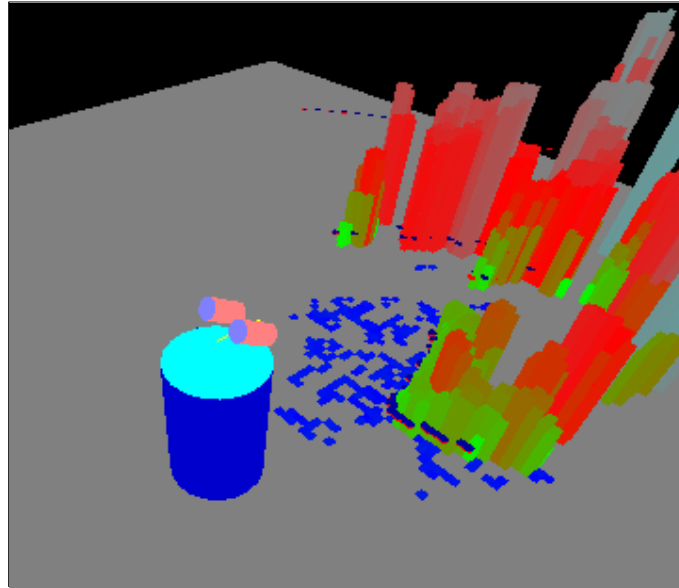


図 12 2.5 次元マップ

3.2.3 各ピクセルにおける障害物の存在確率

3.2.2 で作成した 2.5 次元マップに障害物の存在確率を与える．あるグリッドにおける障害物の存在確率 P_c ，は高さ確率 P_h ，そのグリッドに含まれるデプスデータの高さにおける標準偏差確率 P_{sd} ，視点からの距離確率 P_d の 3 つのパラメータの積で式 (8) のように与えられる．

$$P_c = P_h \cdot P_{sd} \cdot P_d \quad (8)$$

各パラメータの値域は 0 から 1 であり，それぞれはシグモイド関数によって与えられる．シグモイド関数とは 0 から 1 までの値域を持つ微分可能な関数であり Fig:13 ような関数をいう．

Fig:13 よりシグモイド関数は以下のような式で表される．

$$f(x) = \frac{1}{1 + e^{(a(-x+a))}} \quad (9)$$

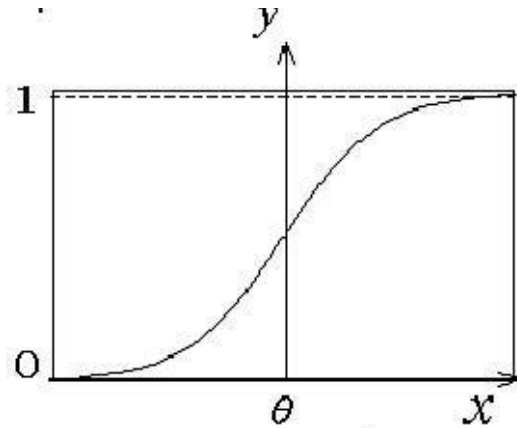


図 13 シグモイド関数

ここで a は比例定数を表す． θ はオフセット θ のとき 0.5 をとる．本研究では 0.5 を境に障害物の存在の有無を表すこととし， $\theta = 0.5$ のときは，障害物の存在の有無を決定できないとして評価する．

シグモイド関数を用いて各パラメータを求めてゆく．グリッドが持つ高さを h とし，床面ノイズオフセット θ とする．ここで床面ノイズオフセットとはノイズによる床の高さのバラツキ度を表す．本研究では経験的に $0 \sim 100[mm]$ の高さをもつピクセルは床面と見なすため， $\theta = 100$ とする．よって，グリッドの高さ確率 P_h は以下のように与えられる．

$$P_h = \frac{1}{1 + e^{(0.05(-h+100))}} \quad (10)$$

グリッドに含まれるデプスデータの高さにおける標準偏差 P_{sd} は標準偏差を S_d ，式 (10) のように， θ を偏差オフセットとして， $\theta = 500[mm]$ とすると以下のように与えられる．

$$P_{sd} = 1 - \frac{1}{1 + e^{(0.05(-S_d+500))}} \quad (11)$$

ここで、標準偏差確率は偏差が小さいほどその障害物の存在確率がもっともらしいので、確率を反転させるために最後に 1 から確率を引いておく。

最後に視点から距離による確率を求める。式 (11) と同様に確率を反転させる。ステレオカメラにおいては距離が遠くなるにつれ精度が落ちることが知られている。つまり距離が近いほど信頼度が高いことを表す。距離を d とする。図 13 よりシグモイド関数は $d = 0$ の時 0.5 をとる性質がある。距離確率においては $d = 0$ の時、最も高い確率 1 をとるから右辺を 2 倍しておく。よって視点からの距離確率 P_d は

$$P_d = 2 \left(1 - \frac{1}{1 + e^{(0.0005(-d))}} \right) \quad (12)$$

として与えることができる。式 (10)、式 (11)、式 (12) の得た各確率を式 (8) に代入し、各ピクセルにおける障害物の存在確率を求める。この確率は先で述べるレンジプロファイルの作成時に利用する。

3.2.4 ベイズ定理による障害物の存在確率の更新

毎回のセンシングにより 2.5 次元マップは更新される。ここで、前回の視野情報を利用して今回の 2.5 次元マップを作成するため、本研究ではベイズの定理を利用して更新を行なう。

ベイズの定理とは未来の事象の確率を過去の事象の確率により計算する手法である。よって、新に更新されたグリッドにおける障害物の存在確率 P_t は、3.2.3 で求めた存在確率 P_c 、と過去の 2.5 次元マップ上におけるそのグリッドの存在確率 P_o により計算することができる。ベイズの定理より計算式を以下に示す。

$$P_t = \frac{P_o P_c}{(P_o P_c) + (1 - P_o)(1 - P_c)} \quad (13)$$

このようにして、2.5 次元マップ上の各グリッドを更新する。図 14 にカメラ画像、視差画像、それにより生成される投影画像と各セルの確立を示したマップを示す。

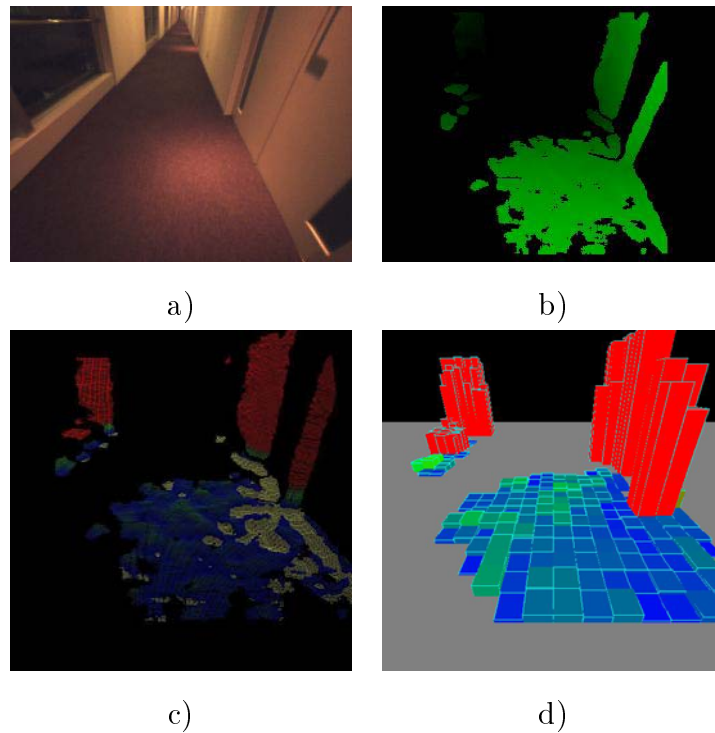


図 14 カレントビジョンビュー: a) カメラ画像, b) 視差画像, c) グローバル座標に投影した各点, d) 各セルの確率

3.2.5 レンジプロファイルへの登録

3.2.2 で作成した 2.5 次元マップを用いてレンジプロファイルを作成していく。視点を中心に視野を 40 分割する。本研究では視点から $3[m]$ 以内のデプスデータを利用して、2.5 次元マップを作成している。ステレオカメラより取得できる視野角は 80 度であり、これを 40 分割すると 1 つ 2 度となる。視点から $3[m]$ のポイントでは 2 度進むと約 $100[mm]$ ずれる事になる。マップは 1 ピクセル $100[mm]$ としているので、ピクセル毎に評価するには重複も無く、荒くも無くちょうど良い。よって、視野を 40 分割する。

次に各角度毎に視点から障害物までの距離を調べてゆく。Fig??に示すように評価角度を $i\theta$ とし、視点からの距離を t とすると、調べるピクセルの座標 (x, y) は

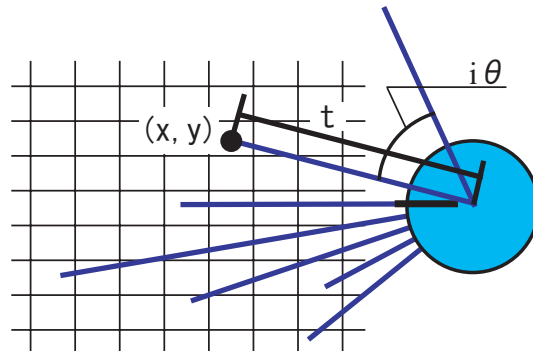


図 15 レンジプロファイルへの登録

$$\begin{pmatrix} x, & y \end{pmatrix}^T = \begin{pmatrix} t \cos \theta, & t \sin \theta \end{pmatrix}^T \quad (14)$$

で求めることができる．ここで T は転置を表す．この時 3.2.3 で述べたグリッドの障害物存在確率を判定する．本研究では存在確率が 0.5 より大きな値を持つとき，そのピクセルを占有する障害物があると判定する．ここで，存在確率 0.5 は障害物の有無の確率が等しく判断できない場合を表す．初期状態においては全てのグリッドは存在確率を 0.5 としている．もしもそのグリッドに障害物データがあればその高さデータと，視点からの距離をレンジプロファイルに登録する．

前にも述べたようにマップは 1 ピクセル 100[mm] としているため，ここで評価も 100[mm] ずつ，このように視点から 3[m] まで 100[mm] ずつ調べ，障害物までの距離をレンジプロファイルに登録してゆく．

3.3 環境マップの作成

先に述べたように観測確率を求めるためには，観測データとマップのマッチングが必要である．よって環境マップを用意する必要がある．しかし，このままレンジプロファイルとマッチングさせると誤認識をおこす問題が生じる．レンジプロファイルは視野分割した各角度のロボットから障害物表面までの距離をまとめ

たものである．よって，環境地図においても壁などの障害物の表面の情報を知ることが重要となる．本研究では，作成した環境マップにエッジ抽出処理を施し，マッチングに用いる．以下に詳しく説明する．

3.3.1 2次元グリッド環境マップ

マップは筆者が作成し用意した．本研究では観測データとのマッチングに2次元のマップを用いる．Fig:16 に実際に作成したマップを示す．

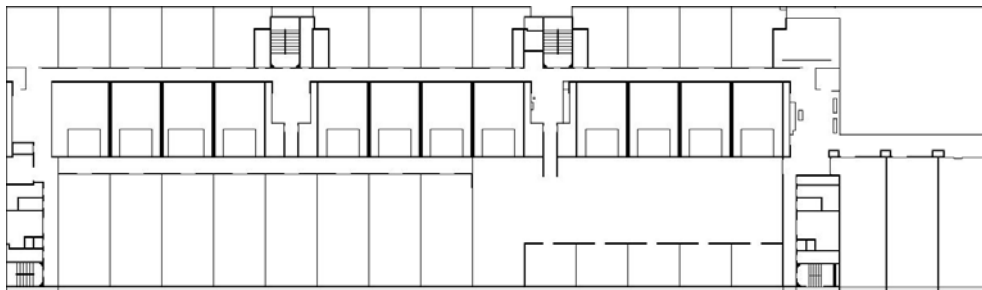


図 16 マップ

用意したマップは産業技術総合研究所3階の間取り図である．建築図面を利用してペイントエディタにより作成した．Fig:16の白色は走行可能領域，黒色は走行不可能領域，つまり障害物を表している．ファイル形式はグレースケールのpgmファイルである．1ピクセル100mmとしマップサイズは 1216×352 [pixel]となっている．

3.3.2 環境マップのエッジ抽出

グレースケールの2次元環境をマップを作成したが，そのままレンジプロファイルとのマッチングを行なうと，マッチングを誤ってしまう．

Fig:17のような十分に厚さを持つ障害物が存在するとして，その時のロボットの位置におけるレンジプロファイルとのマッチングを考える．

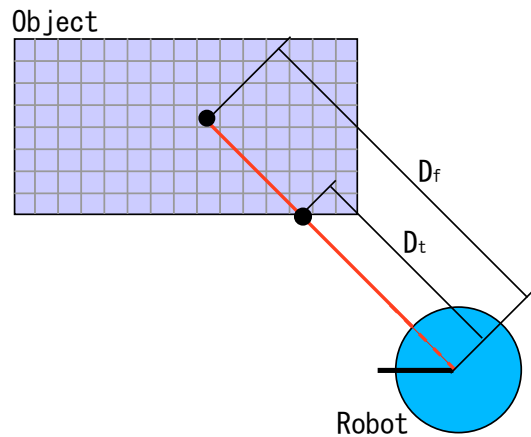


図 17 マッチングエラー説明図

本研究ではマッチングの際，あるパーティクルが持つレンジプロファイルが示す各角度ごとの距離を基準として，その点から地図上の障害物の距離までを評価する．

この時，ある角度方向でのマッチングで距離データが障害物内部で入り込むような長さを持っていた場合，マッチングはその場で距離 D_f で成立してしまう．しかし，実際にロボットから障害物までの距離は D_t であり，誤ったマッチング結果を算出してしまふ事となる．

よって，3.3.1 で作成したマップからエッジ部分を抽出する必要がある．

本研究ではマップのエッジ抽出にソーベル・フィルタ (Sobel Edge detector と呼ばれる) を用いる．ソーベルフィルタについて簡単に説明をする．

画像の中のある領域の境界 (エッジ) を画像処理により検出するとき，領域の境界では，画素の輝度値の変化が大きいため，画素値の変化に対して微分演算を行えば，エッジの検出を行うことが知られている。ただし，デジタル化された画像に対する計算機による処理では，微分演算の代わりに差分演算を行なうことによってこれを行うことが可能になる．しかし，通常の差分演算を行なうと，画面に含まれる雑音成分にも反応してしまうため，雑音の低減とノイズの除去の両方の働きを持つフィルタがいくつか提案されている．ソーベルフィルタはその中の

一つとしてよく利用されているフィルタである。

ソーベルフィルタは、ある注目画素を中心とした上下左右9つの画素の輝度値に対して、垂直方向、水平方向の二つの係数行列を用いてそれぞれ乗算し、結果を合計する手法である。水平方向 g_H 、垂直方向 g_V の係数行列は

$$g_H = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad g_V = \begin{bmatrix} -1 & -2 & 1 \\ 0 & 0 & 0 \\ -1 & 2 & 1 \end{bmatrix} \quad (15)$$

で与えられる。注目画素の座標値 (x, y) の輝度値 $f(x, y)$ とすると、各画素と水平方向係数 g_H をそれぞれ乗算し、その結果を合計した値 g_{HS} は

$$\begin{aligned} g_{HS} &= (-1) \times f(x-1, y-1) + (0) \times f(x, y-1) + f(x+1, y-1) \\ &\quad + (-2) \times f(x-1, y) + (0) \times f(x, y) + (2) \times f(x+1, y) \\ &\quad + (-1) \times f(x-1, y) + (0) \times f(x, y) + f(x+1, y) \end{aligned} \quad (16)$$

となる。垂直方向の合計値 g_{VS} も同様に求めると、注目画素の画素値は以下の式で求めることができる。

$$g = (g_{HS}^2 + g_{VS}^2)^{\frac{1}{2}} \quad (17)$$

以上に示すようなフィルタをかけることにより、3.3.1で作成した2次元環境マップを Fig:18 のようなエッジマップに変換する。

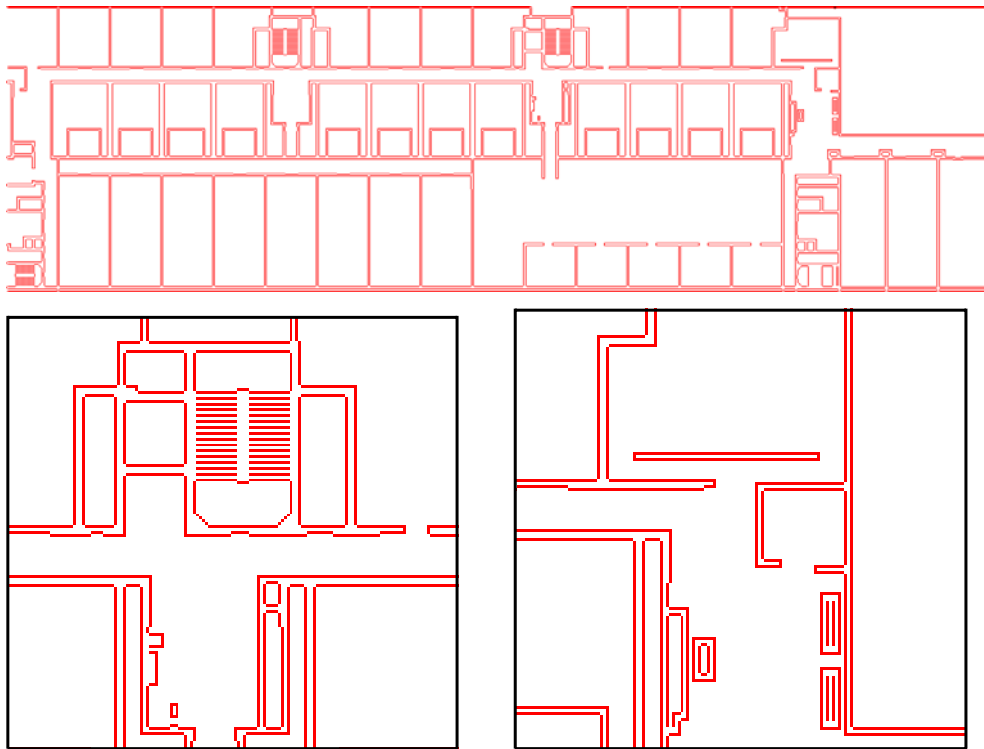


図 18 エッジマップ

4. 実機による実環境での自己位置同定

本章では実環境での走行実験およびオンラインシステムについて述べる．走行実験に用いるハードウェア，ソフトウェアの構成を説明し，精度検証，実環境における有効性を実証する実験について述べる．ロボットの操作はジョイスティックにより無線通信で外部のノートPCより行なう．このようにして，実環境を走行しデータを保存する．その後オフラインによる自己位置同定を行い，実際のロボット位置と推定結果を比較することにより行なう．その後本手法で提案する手法を実際にロボットに実装しオンラインシステムとして構築する．

4.1 ハードウェアおよびソフトウェア構成

実験に用いるハードウェアとソフトウェア構成の説明をする．ハードウェア構成は主に車輪型ロボットに各センサを取り付けた物となっている．センサ類および無線通信用機器は全てロボット頭部に装着した，ロボット内部にはPCと電源ユニット，モータドライバ，バッテリー×5つからなる．

ソフトウェア構成では核となるメインプログラムがセンシングと位置同定の処理を管理する2つのコントローラーからなっている．

4.1.1 NOMAD200

本研究で用いたロボットは車輪型移動ロボット NOMAD200 を Fig:19 に示す．NOMAD200 の外形は，直径 460[mm]，高さ 880[mm]，重量 59[kg] の円筒形である．3つの車輪で全方向に移動でき最大速度は約 0.5[km/h] である．頭部には 4.1.2 で示したステレオカメラを搭載しており，前方下方の地面を見下ろすように設置されている．また，頭部の中央に 1軸光ジャイロを搭載している．CPU は Pentium 800MHz, Memory は 384MB, OS は *LinuxGNUDebian2.4.18* を用いる．ロボットとの通信は無線であり，通信方式は *IEEE802.11a* である．ステーションとして頭部に NEC Aterm WL54TE を接続する．付録 B の Table:1 にロボットの仕様を示す．

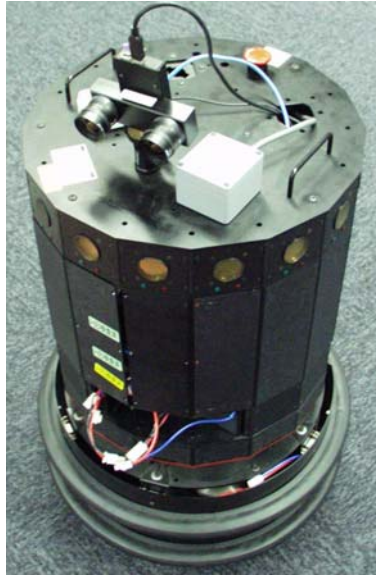


図 19 NOMAD200

4.1.2 ステレオカメラ

本研究で用いたカメラは、Fig.20 に示す株式会社リバスト社の IEEE 1394 接続カメラ、メガピクセルデジタルステレオヘッドを用いた。本研究では 4.1 に述べたように NOMAD200 頭部に装着し、フレームフォーマット 320×240 で環境データを取得する。付録 C の Table: 2 にカメラの仕様を示す。

4.1.3 ファイバーオプティカルジャイロ

本研究では Fig.21 に示す日本航空電子工業株式会社の RS-232C 接続ジャイロ、JG-35FD を用いた。これを NOMAD200 の頭部中心に固定し、オドメトリによる回転方向の変移量を補正する形で用いる。付録 D の Table 3 にジャイロの仕様を示す。



☒ 20 MEGA-DCS



☒ 21 JG-35FD

4.1.4 ソフトウェア構成

次にソフトウェア構成について説明する．本研究で提案するシステムは主にセンサからのキャプチャーを担当する Sensor controller 部と自己位置同定を担当する Computation Controller 部の 2 つのフェーズからなる．Fig:22 に本システムのソフトウェア構成図を示す．Sensor controller はオドメトリ，ジャイロ，ステレオカメラの初期化やキャプチャーをして，Computation Controller にデータを渡す，Computation Controller 部はセンサからの観測データを参照し，位置同定を行なう．また，与えるフロアマップもこのフェーズで処理され，マッチングに用いられる．先のオンラインシステムでは 4.5 で述べる経路計画フェーズもこの Computation Controller が管理する．

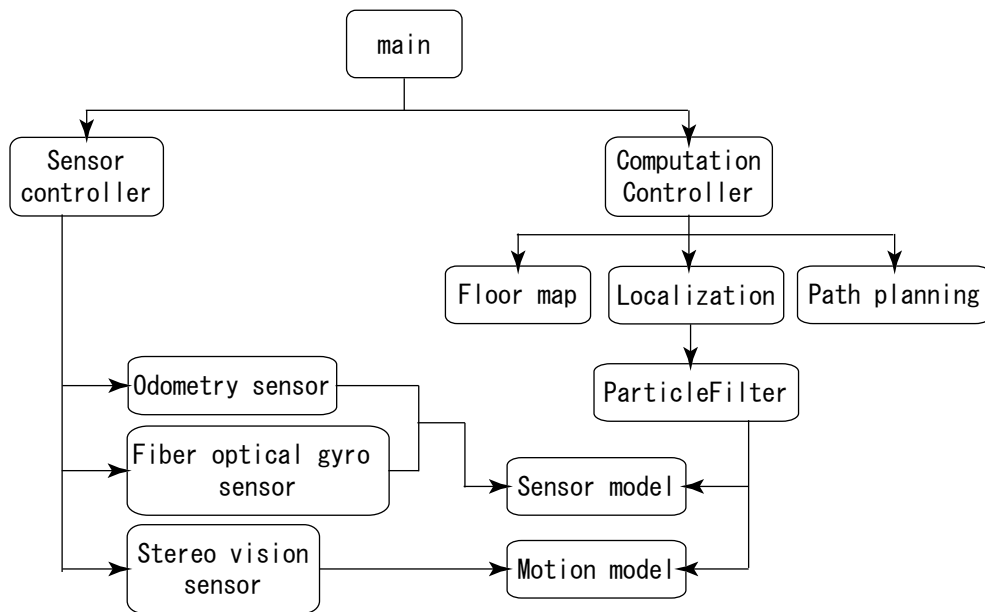


図 22 ソフトウェア構成図

以上のようなアルゴリズムを実際に実装して，実験を行う．まず，NOMAD200 により環境データを取得しオフラインにて自己位置同定を行い，本手法の検証を行なう．続いて，NOMAD200 に本体に本手法を実装し，オンラインな移動ロボットシステムを構築した．この際，経路計画アルゴリズムも実装した．

4.2 オフライン自己位置同定

実装したシステムについて位置同定精度の検証実験を行った。屋内実環境(産業技術総合研究所 3F)を走行し、環境データを取得する。その後これらのデータを用いて自己位置同定をオフラインで行い、ロボットの位置を推定しその精度を真値と比較する。真値は走行コース途中数点でのロボット位置を手計測したものをを用いる。パーティクルの数は500点とした。

4.2.1 並進コースによる実験

まず、主に並進移動のみのコースを2往復半した。距離にして約160[m]走行した。走行経路を Fig:23 に示すと共に、同定結果と真値との差を Fig:24 に示す。赤線はオドメトリのみで同定を行なった結果と真値との誤差を表しており、青線は本手法による同定結果と真値との誤差を示している。積分誤差が溜まるオドメトリと違い、本研究の手法では誤差が蓄積していかない事が確認できる。最大誤差は31[cm]であった。

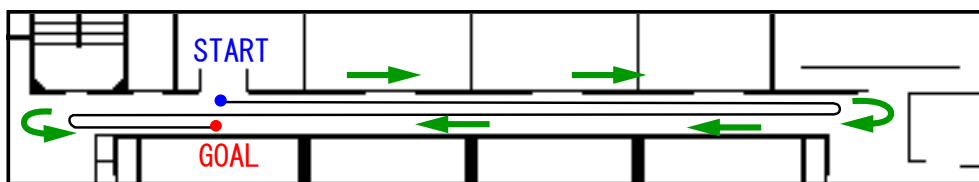


図 23 走行コース 1

4.2.2 並進・回転コースによる実験

次に回転も含めたより長い距離を走行し、誤差の蓄積度合いを見ていく。また、今回の実験ではステレオカメラのキャプチャレートを上げて環境データを取得する。走行距離は約220[m]である。1つ目の実験と同様に走行経路を Fig:25 に示

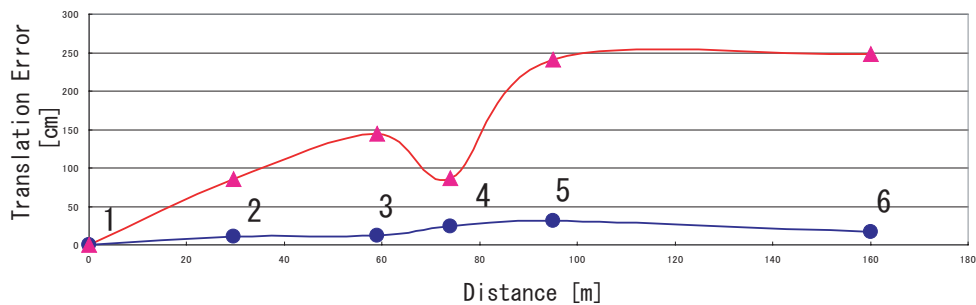


図 24 走行コース 1 における位置同定結果

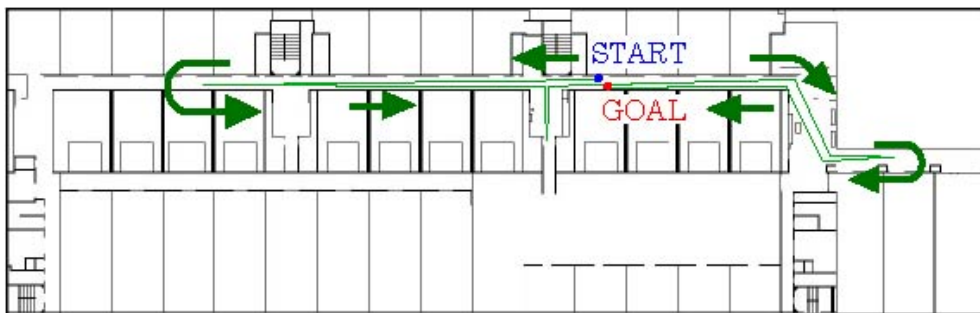
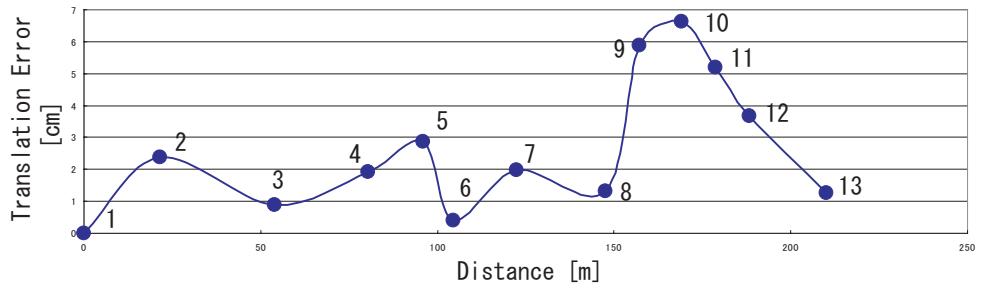


図 25 走行コース 2

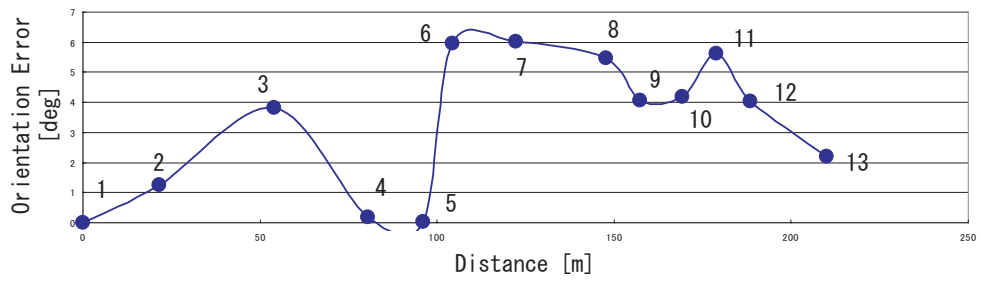
すと共に、同定結果と真値との差を Fig:26 に示す。今回は走行経路中に回転移動が含まれるため角度誤差も求めた。

今回の実験でも位置、角度ともに距離によって誤差が蓄積していかない事が確認できる。また、レートを上げたために精度も前回の実験よりも改善されている。最大位置誤差は $6[cm]$ 、角度誤差は $6[deg]$ であった。

本研究手法により推定した走行経路結果を Fig:27 に示す。青線はオドメトリ情報のみを用いて同定を行なった結果であり、赤線は本研究での手法による同定結果を表す。また、緑で表した点は真値である各ロボット位置を手計測したポイントを示している。



a)



b)

図 26 走行コース 2 における位置同定結果: a) 位置誤差, b) 角度誤差

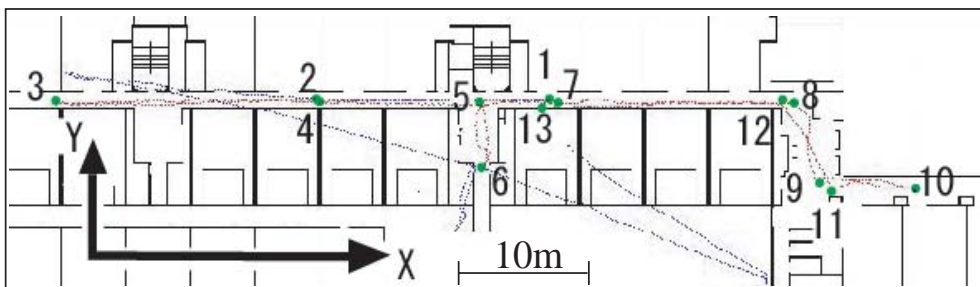


図 27 走行経路の推定結果

4.2.3 考察

以上の2つの実験により、本手法では距離が進むにつれて誤差が蓄積しないことが確認できた。また、レートを上げる事によって精度の改善も見ることができた。誤差の精度としては精度が比較的良いとされるレーザレンジファインダで行なわれるものと同程度の精度を得る事ができ、実環境でも障害物に衝突することなく走行できることが確認できた。

しかし、これはチェックポイント(手計測を行なった地点) に対する検知である。Fig:27の推定経路において他の部分を見てみると、この時Fig:28に示すように手計測箇所以外で推定経路が大きくぶれる箇所が存在する。

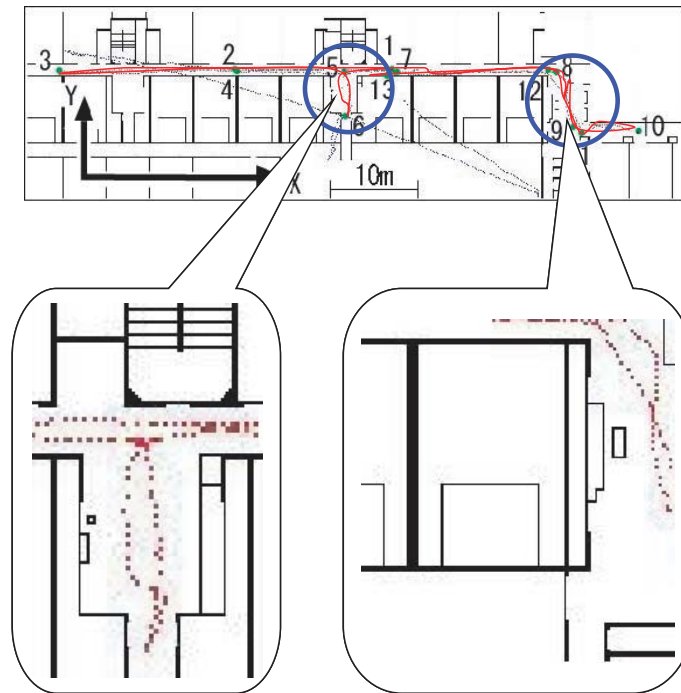


図 28 走行経路の推定結果

このようなポイントで推定経路がぶれる原因として以下のようなことが考えられる。

1. オープンスペースによる環境データの不足

2. ステレオカメラによるアンマッチング
3. 床とのすべりによるオドメトリ誤差
4. 地図に無い障害物の影響

以上の2つの実験により、本手法では距離が進むにつれて誤差が蓄積しないことが確認できた。また、レートを上げる事によって精度の改善も見ることができた。誤差の精度としては精度が比較的良いとされるレーザレンジファインダで行なわれるものと同程度の精度を得る事ができ、実環境でも障害物に衝突することなく走行できることが確認できた。

しかし、これはチェックポイント(手計測を行なった地点)に対しての検知である。Fig:27の推定経路において他の部分を見てみると、この時Fig:28に示すように手計測箇所以外で推定経路に大きくぶれる箇所が存在する。

4.3 車輪のすべりの影響

まず、4.2.3で挙げた、すべりによるオドメトリ誤差の影響について検証する。ロボットを実験経路を走行させる際に、故意にロボットにすべりを発生させる。データはステレオカメラとオドメトリに加え、新たに1軸ファイバ・オプティカル・ジャイロ(以下FOG)によりyaw軸の回転角度を取得する。その後、FOGでの補正の有無における位置同定をオフラインでおこない、その同定結果を検証する。パーティクル数は50点とした。

また、車輪型ロボットにおいて実環境内走行では、段差やフロアタイルによる床と車輪とのすべりが起こる。このエラーは推定誤差に大きく影響を及ぼす。特にすべりによる回転角誤差はその後の位置同定に致命的な影響を与える。曲がり角などではスペースも広がり環境データの取得が困難となることがあり、推定に大きな誤差を持つ可能性がある。そこで、すべりによる回転誤差に対して回転誤差を補正するため、ジャイロを用いる。

4.3.1 車輪のすべり実験

Fig:29 に示す経路を走行する．オレンジで囲まれた部分でロボットに半時計周り方向に回転すべりを断続的に約 90 度与える．a) では直線軌道を保つように走行した．対して b) はロボットに図の下方方向に真っ直ぐ走らせ，回転移動命令は与えない．

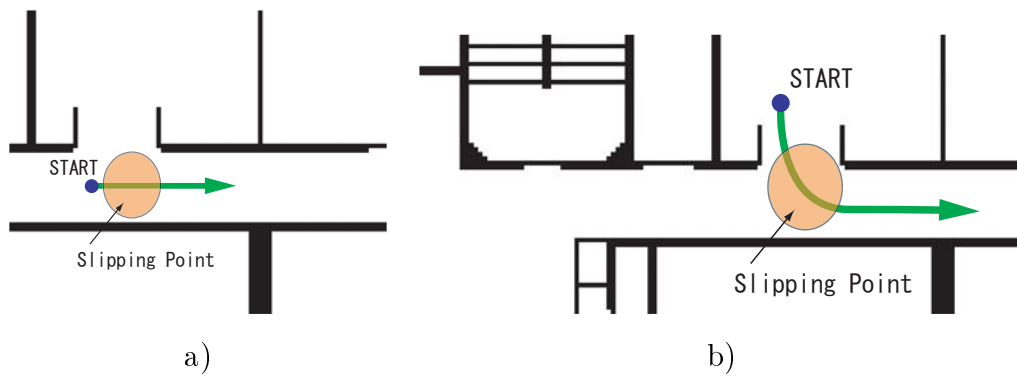


図 29 実験コース

同定結果を Fig:30 に示す．緑線で表される経路が補正を行わないオドメトリによる同定結果であり，赤線で表される経路が FOG を用いた同定結果である．

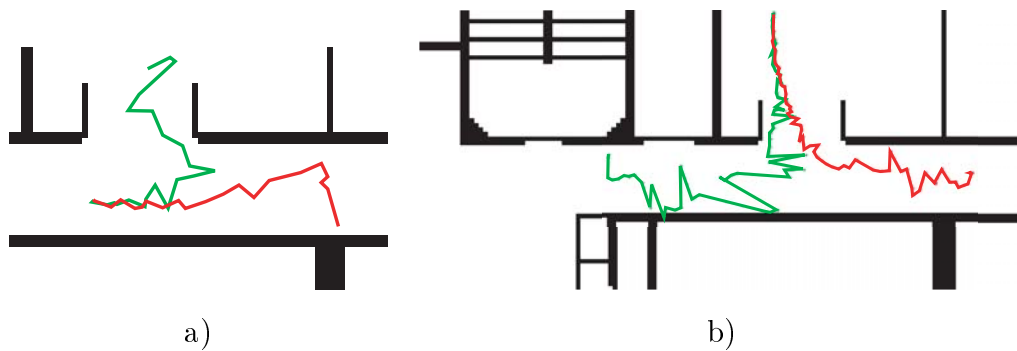


図 30 位置同定結果

双方とも FOG により補正を与えないと自己位置を見失い，誤った方向に進ん

でしまった．一方 FOG により補正を行なえば，本実験のような大きなすべりが生じても同定が可能であることが確認できた．以上により FOG を搭載する事ですべりによる，同定誤差を回避できるようになる．

4.4 地図に無い障害物の影響

次に 4.2.3 で挙げた，地図に無い障害物に対する位置同定の影響について検証する．ステレオカメラは環境に対して非常にセンシティブなセンサであるため，小さな障害物でも確認することができる．本手法では観測データを環境マップにマッチングさせる事によりロボットの位置を同定している．よって，同定結果は地図に無い障害物に非常に影響を受けることが考えられる．

4.4.1 走行実験

走行実験は Fig:31 に示すような環境を走行する．経路は Fig:31a) のような直進，反転，直進とシンプルな経路となっている．本実験でも推定位置と真値の差を評価し，地図に無い障害物の影響を考察する．経路が非常にシンプルであるため，今回の実験ではオドメトリのみによる推定値を真値とする．赤い円で囲まれた部分には地図にはない障害物を設置した．今回の場合では靴 (Fig:31b)) を用いた．

Fig:32，Fig:33 に推定経路結果とその誤差を示す．赤線が本手法により推定経路であり，青線が走行経路である．Fig:32 から，やはり障害物の影響を受けていることがわかる．本手法により推定した経路は真値よりも障害物側にずれている．これは地図に無い障害物 (靴) と環境マップの壁の部分がマッチングされたために，推定位置が障害物方向に引きずられる結果となったと考えられる．しかし，Fig:32，Fig:33 から障害物に引きずられるものの，観測した障害物との相対距離により自己位置を見つける本手法では壁に衝突することなく走行できた．また，地図に無い障害物がカメラの視野外になると推定値は真値に収束していくことが確認できた．

以上のように，ステレオカメラを用いた本手法では靴のような小さな障害物を確認することが出来るが，その存在に推定の精度が影響を受けてしまう．しかし，

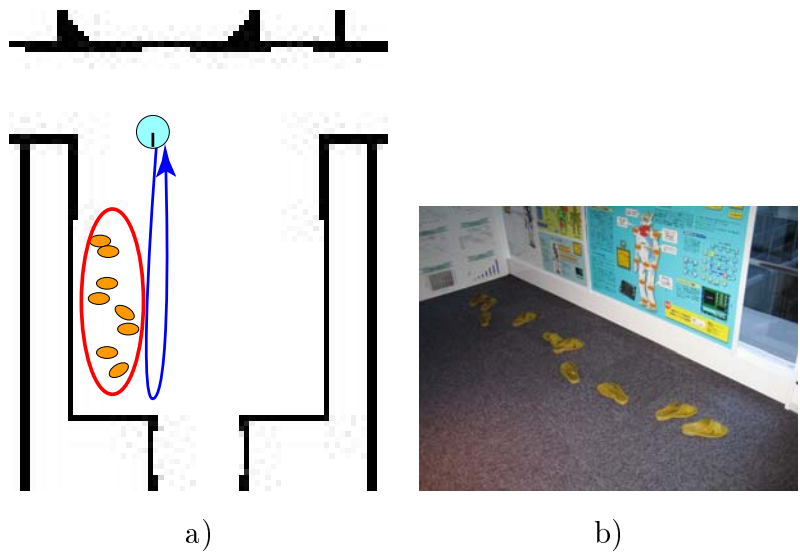


图 31 实验环境

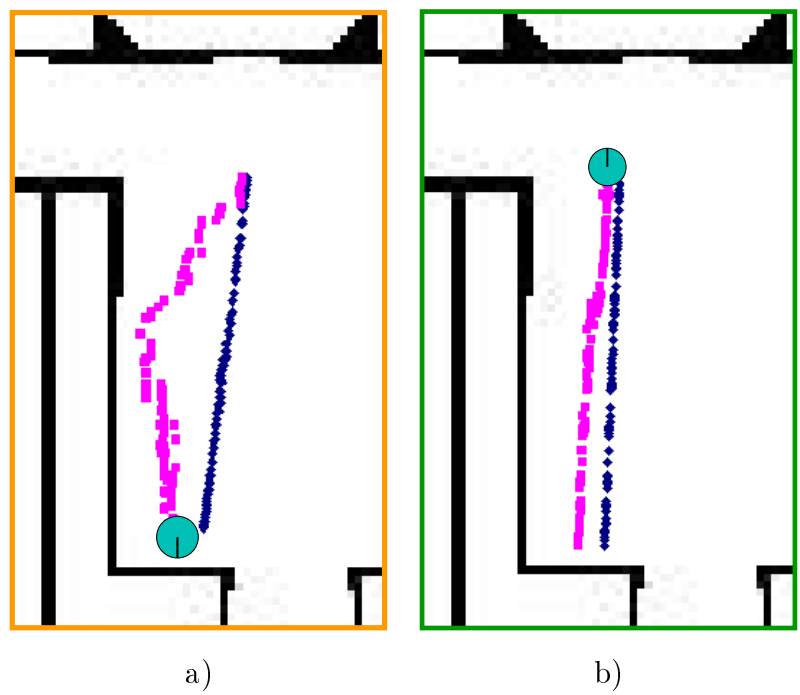


图 32 推定结果：a) 往路, b) 复路

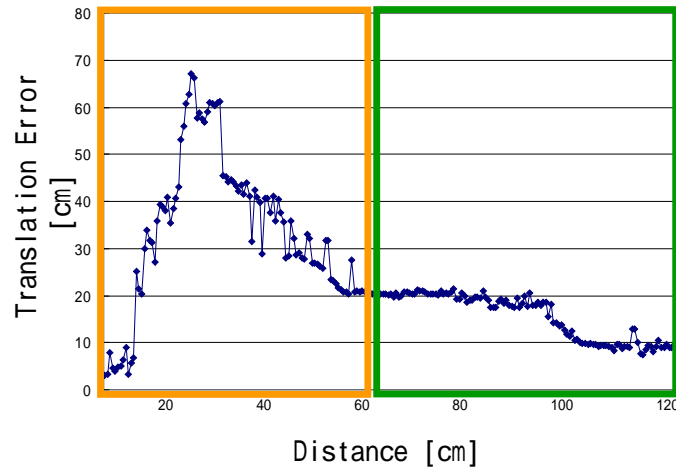


図 33 位置推定誤差

一時的な障害物エリアならば，障害物が存在する地点を通り過ぎるとその誤差を回復することができる．

4.4.2 高さ情報の利用

4.4.1 で述べたように，本手法では小さい障害物を発見することができるが，そのために，自己位置同定の結果に影響を及ぼしてしまう．そこで，高さ情報を利用する事によりこの問題を解決する．

本研究では，センサにステレオカメラを用いる事により，1度のセンシングで3次元な観測データを取得することができる．よって，各ピクセル毎にレンジプロファイルを作成する際，高さ情報も加味して障害物を判定することによって，地図にない障害物による自己位置同定への影響を回避することができる．以上のような判定を新たにシステムに加え，4.4.1で行なった実験のデータを用いて，再度位置同定を試みた．

Fig:34 に推定結果を示す．青線で示される軌跡が実際の走行経路を示している．これは Fig:32 同様，オドメトリのみで算出した走行経路である．赤線は高さ情報を加味したロボットの位置同定の推定結果を表す．Fig:32 よりも改善が図れてい

るのがわかる．次に走行経路の真値との誤差を Fig:35 に示す．数値的に見ても精度が改善されたことを確認できる．最大誤差は12[cm]であった．

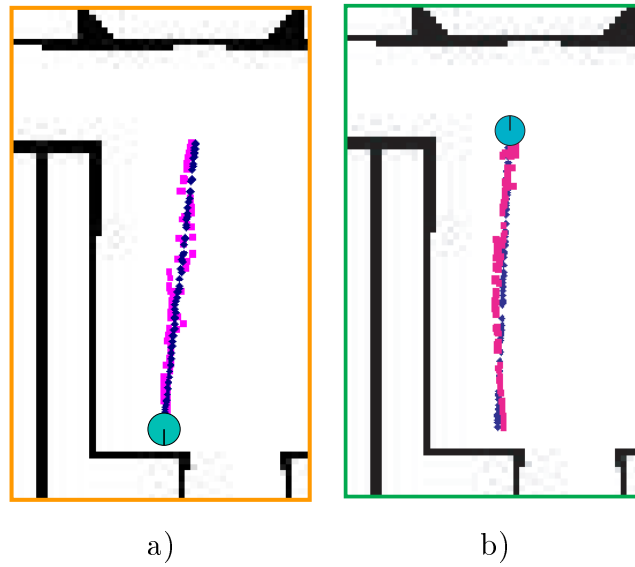


図 34 高さ情報を加味した推定結果：a) 往路, b) 復路

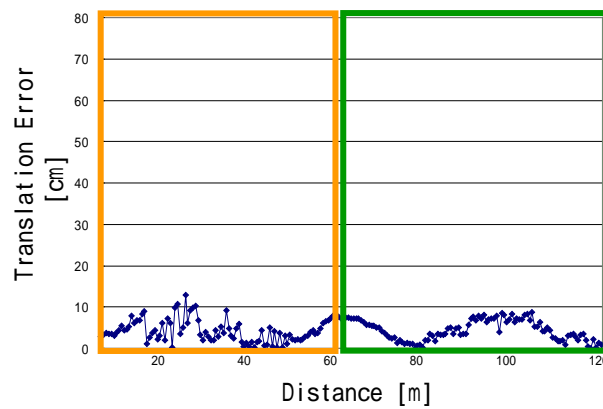


図 35 高さ情報を加味した場合の位置推定誤差

4.5 オンライン自己位置同定

NOMAD200 本体に本手法を実装して、オンラインシステムを構築した。その際、経路計画機能も加えて実装した。実際に実環境を走行し自己位置同定を行う。

4.5.1 実際の走行結果

オフライン実験同様外部の PC からジョイスティックにより操作し、自己位置同定を行なう。システムは全て NOMAD 内に実装しており、推定結果を外部 PC に Xwindow を飛ばす事で表示させている。走行経路は Fig:36 に示すように、直進後右折するような経路を走行し、同時にその位置推定を行なっている。環境は与えた環境マップと同様の環境を与え、地図上に存在しない障害物はない状態で行なう。

Fig:36 にオンライン自己位置同定システムを示す。左側の実機の走行時の様子を示す写真であり、右側がそれに対応する推定結果なのである。各それぞれは位置同定を始めてから 1,27,40,54 フレーム目の状態を示している。Fig:36 に示すように、オンラインでもオフライン実験と同様の精度で自己位置推定をすることができた。

4.5.2 経路探索法

ここで簡単に使用する経路計画について説明をおこなう。使用する経路計画法は Kuffner の最適化経路探索法である [28]。

Fig:37 に示すように、グリッド表現された地図でユークリッド距離で表された環境では、90 度以下の分岐を含むパスは、以下の式で示されるように最短経路となりえない。

$$\|q - r\| \leq \|q - p\| + \|p - r\|$$

Fig:38 の中および右図に示すように、そのノードに別のノードから到達してきた場合に可能な分岐の数は、左図の 8 近傍すべてを探索する場合に比べて平均的に半分に抑えることができるために、全体としても探索時間を半分程度に抑えることができる。



1 フレーム目



27 フレーム目



40 フレーム目



54 フレーム目

図 36 オンライン自己位置同定

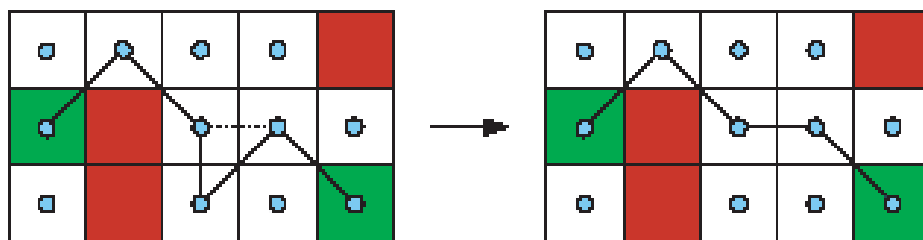


図 37 経路計画法説明図 1

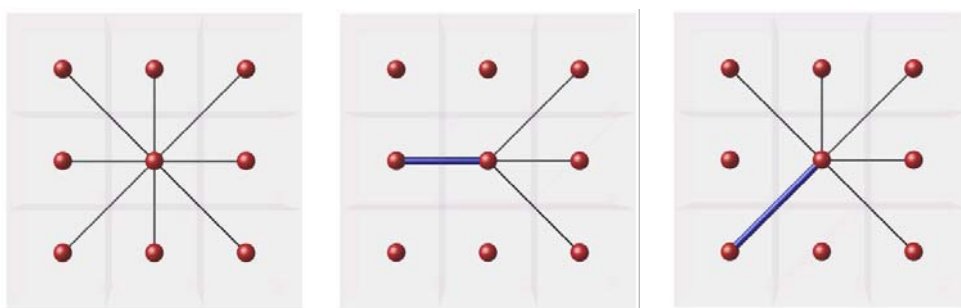


図 38 経路計画法説明図 2

以上のような経路計画法を NOMAD に実装した。Fig:39 に本システムに実装し、経路計画を行なったものを示す。Fig:39 は産業技術総合研究所 3F 廊下から、研究室までの経路をプランニングしたものである。GUI によりゴール地点をクリックすれば、現在のロボット位置からゴール地点までの経路を算出できる。

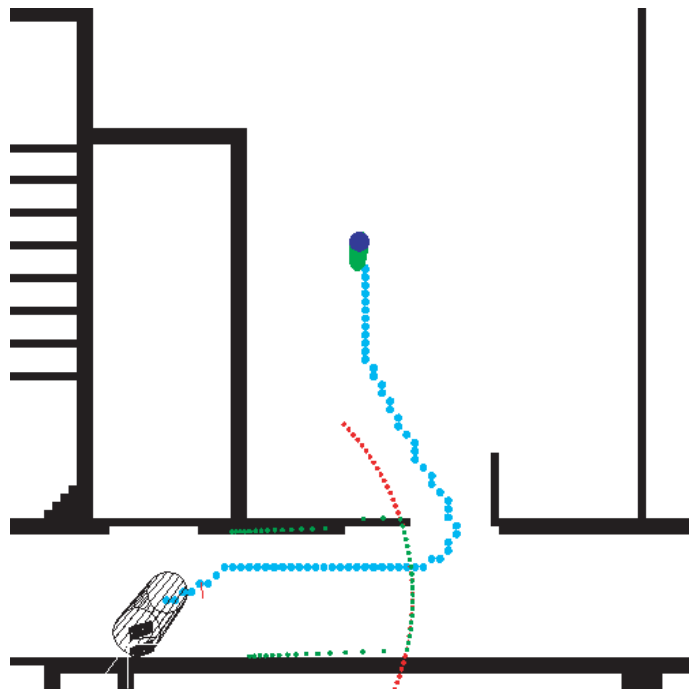


図 39 経路計画

5. 結論

5.1 ステレオ視を用いた自己位置同定

本研究ではステレオ視を利用した移動ロボットにおける自己位置同定法を提案した．移動ロボットにおいて求められる基本機能として移動が挙げられるが，自己位置同定は移動をするために最も重要な機能であり，難しい問題である．従来の手法の多くは平面センシングベースのセンサを用いるものが多く，実環境を真に観測できたと言い難い．そこで，本研究では

- 環境を立体的に観測できる事
- エラーに強く長い距離を移動できる事
- 動的環境に強い事

を満足するシステムの構築を目指した．

まず，位置同定問題を定式化し，それを解決する手法としてパーティクルフィルタのアルゴリズムについて2章で述べた．実際に位置同定を行なうためにはロボットの状態遷移ダイナミクスとセンサの観測特性をモデル化する事によって，位置の推定を行なうことができる．一般にセンサには多くの観測ノイズを含んでおり，これは位置同定に大きく影響してくる．また，位置同定を行なうために，観測データから必要な情報を抽出する必要がある．

3章では実際に本研究の提案手法を実装する際の問題点を挙げその対処法を述べた．距離画像におけるノイズは位置推定に大きな影響を与える．本研究では画像中に存在するノイズを2種類に分け，その除去法を述べた．また，距離画像の3次元データを2次元の環境マップとマッチングさせるための各データの加工プロセスについても説明した．

4章では実際に本手法を搭載するロボットと，ステレオカメラシステムなどの，ハードウェア構成と，本手法のソフトウェア構成および実環境での走行実験を説明した．した．実環境約1フロア距離にして220[m]を走行し，同定を行なった結果ロボットの自己位置を見失うことなく最大位置誤差6[cm], 最大角度誤差6[deg]

という誤差で推定することができることを確認した。この精度は外界観測センサとして、比較的高精度として知られるレーザセンサと同等の精度である。本手法ではオドメトリのような距離により積分誤差が蓄積することなく、誤差を回復しながら同定を繰り返すことができるため、ロボットは自己位置を見失うことなく推定を行なうことができる。またステレオカメラを用いることにより環境を3次元に観測できる本手法は、環境中の小物体も観測でき、ロボットが実環境で行動するために、より優位性のある手法であることを示した。また、実環境の走行ではホイールと床とのすべりにより、微小な誤差が徐々に蓄積してしまうエラーに対しても、yaw 軸光ジャイロを用いることにより、オドメトリの回転方向の移動量を補正することによりエラーに強いシステムを構築した。次に動的に変化する環境についての影響の検証について、地図にない障害物に自己位置同定がどのように影響するかを実験した。結果として地図に無いものを感知すれば同定結果がその障害物の影響を受けることを示した。特に直線上に並んでいるようなものはマッチングの際壁と認識してしまい、同定結果に大きなずれが生じる。しかし、本手法では環境データを三次元で取得できるため、その高さ情報を加味することにより、地図にない障害物を発見しても、それを地図上の障害物として認識する事を回避できる。結果は、障害物に引かれるときの誤差は最大約 $60[cm]$ に対し、高さ情報を加味した同定では最大誤差約 $10[cm]$ でロボットの位置を推定できることを示した。最後に本手法をオンライン移動ロボットシステムとして移動ロボット本体に実装した。この際、Kuffner が提案する経路探索アルゴリズムも搭載しシステムを構築し、実環境を走行しながら位置同定をすることに成功した。

5.2 今後の課題

本研究で提案した手法は改良余地が残されている。まず本システムではオンラインシステムとして、自己位置同定と経路計画が可能であるため、自動走行機能を追加ことが考えられる。また、地図にない障害物、特に小さな障害物でも観測が可能のため、障害物回避機能も追加することができる。この時問題となるのが、どれだけカメラからの観測データが信用できるかということになってくる。ノイズ処理や、閾値をもうけてある距離以上の観測データを使用しないなど、信用度

の高い観測データを用いるようなシステムとなっているが、環境によってはやはり誤認識が存在する。しかし、数フレームを通して一定の場所に障害物を観測すれば、その誤認識を経路計画に加味しないようにすることが可能であると考えられる。以上のようなことをロボットに実装し、自動走行機能を搭載していきたい。

初期位置の問題も挙げられる。現在初期値は、手計測した値をロボットの持つマップ座標に変換しロボットに与えている。実環境でナビゲーション等のタスクを行なう移動ロボットにおいて、初期位置を自分で探索し推定する能力は大変重要であると考えられる。この機能を応用すれば、異常にマッチングが外れた場合など自己位置を見失ったとき、再探索により再び初期位置としてリセットすることが可能となる。本研究で提案したシステムは長距離走行しても誤差を回復できる特色を持っている。しかし、センサのレンジ以上のオープンスペース等で、環境に対する特徴を認識しづらい場所ではやはり、推定結果に誤差が生まれてしまう。自己探索機能を搭載すれば、このような問題にも対処でき、よりエラーに強いシステムを構築できる。友納はスキャンテンプレートマッチングと大域スキャンマッチングを用いて、ロボットの初期位置をロボット自身が探索し推定することに成功している [29][30]。

以上のような機能を付加し、より実環境に対してロバストな移動ロボットシステムを提案していきたい。

謝辞

本研究を遂行するにあたり，研究生活において貴重かつ的確な御指導を賜わり研究を進めるにあたって最高の環境を与えてくださいました産業技術総合研究所デジタルヒューマン学連携講座金出武雄教授に対し，心から感謝の意を表します。

研究方針及び論文執筆にあたり，丁寧な御検討，御教示いただきました本研究科木戸出正繼教授に深く御礼申し上げます。

本研究の方針決定，遂行，論文執筆に至るまで様々な点において幅広い専門知識による適切な御助言を頂きました産業技術総合研究所デジタルヒューマン研究センターの加賀美聡助教授に深く感謝致します。

奈良先端科学技術大学院大学入学から産業技術総合研究所へ移るまでの1年間奈良先端大において私の学生生活における様々なご指導，またその後においてもお忙しい中相談に応じていただいた本研究科小笠原司教授に深く感謝いたします。

また同じく，奈良先端科学技術大学院大学入学から産業技術総合研究所へ移るまでの1年間奈良先端大において学生生活におけるご指導，また私生活において様々な事にご教授いただいた本研究科松本吉央助教授に深く感謝いたします。

本研究の共同研究者として方針決定，遂行，多くのプログラムの提供に至るまで，さまざまな点において適切な御助言を頂いた上，私の拙い英語力にも熱心に何度も聞き指導くださり，英語論文執筆の添削，指導してくださいました。産業技術総合研究所デジタルヒューマン研究センターの Tompson Simon 氏に心より感謝致します。

本研究の方針決定，遂行，論文執筆に至るまで様々な点において幅広い専門知識による適切な御助言を頂きました産業技術総合研究所デジタルヒューマン研究センターの西脇光一氏に心より感謝致します。

産技術総合研究所デジタルヒューマン研究センターの菅原麻衣子秘書の多数の迅速な手続により本研究のための東京への引っ越しに伴う様々な手続きや，他大学との交流など円滑に研究を進めることができました。私の研究を支えて頂き心より感謝いたします。

有限会社アールラボの長嶋功一氏には，生活環境や研究環境において多くの支援等をいただきました。私が東京での単身生活ができたのもこれらの支援のおかげ

げであります。心より感謝いたします。

本研究講座の高岡豊氏とは、一期生として本学を離れ右も左もわからず、手探りで学生生活を歩きました。共に支えあい励ましあうことでこの1年間を乗り越えることができた事を心より感謝いたします。

研究方針について適切な助言や、ロボットのOSの設定、プログラムのコーディングにおける相談、ご指導をいただいた産業技術総合研究所デジタルヒューマン研究センターの立山義祐氏に深く感謝致します。

本研究遂行時のプログラムのコーディングにおける多数の質問や相談、ご指導をいただいた産業技術総合研究所デジタルヒューマン研究センターの川地克明氏に深く感謝致します。

本研究を進めるにあたり物の考え方や、実験の進め方、論文執筆におけるご指導をいただいた産業技術総合研究所デジタルヒューマン研究センターの中田亨氏に深く感謝致します。

ファイバオプティカルジャイロをロボットに設置する際のシリアル通信プログラミングにおけるご指導、相談に応じていただいた産業技術総合研究所デジタルヒューマン研究センターの多田充徳氏に深く感謝致します。

プログラムのコーディングにおける質問や相談、ご指導、シリアルプログラミングの際のデバックにおいて数々の助言を頂いた産業技術総合研究所デジタルヒューマン研究センターの宮田なつき氏に深く感謝致します。

プログラムのコーディングやLinuxにおける多数の質問や相談、ご指導をいただいた産業技術総合研究所デジタルヒューマン研究センターの木村誠氏に深く感謝致します。

プログラムのコーディングにおける質問や行き詰った際の相談、ご指導をいただいた産業技術総合研究所デジタルヒューマン研究センターの山崎俊太郎氏に深く感謝致します。

研究の様々な面で助言、手助けして頂いた 奈良先端科学技術大学院大学ロボティクス講座スタッフの皆様に深く感謝致します。

そして、研究の様々な面で助言、手助けして頂いたロボティクス講座インタラクション所属の皆様、特に足立佳久氏、小枝正直氏、竹村憲太郎氏、KHIAT

Abdelaziz 氏に深く感謝致します。

研究に関係有る無しに関わらず，私の相談に応じて頂い産業技術総合研究所デジタルヒューマン研究センターで研究に従事されている他大学の学生の皆様，特に東京理科大学の学生の皆様は同期，後輩に関わらず本当にお世話になりました。

また，研究に関係有る無しに関わらず私の相談に応じて頂いたロボティクス講座の諸先輩方，同期の皆様には本当にお世話になりました。

最後に，私の長年に渡る学生生活を支えて頂いた両親，家族，そして友人に心から感謝致します。

参考文献

- [1] C. Martin, H.J. Bohme and H.M. Gross. “Conception and Realization of a Multi-Sensory Interactive Mobile Office Guide”. *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pp. 5368–5373, 2004.
- [2] W. Burgard, D. Fox, D. Hahnel, G. Lakemeyer, D. Schulz, W. Steiner, S. Thrun, and A.B. Cremers. “Real Robots for the Real World -The RHINO Museum Tour-Guide Project”. *Proceedings of the AAAI Spring Symposium on Integrating Robotics Research*, 1998.
- [3] 齊藤裕昭. “スキャンデータ列の記憶とマッチングに基づくナビゲーション”. Master’s thesis, Nara Institute of Science and Technology, 2002.
- [4] M. Zobel, J. Denzler, B. Heigl, E. Noth, D. Paulus, J. Schmidt and G. Stemmer. “MOBSY: Integration of Vision and Dialogue in Service Robots”. *Computer Vision Systems, Proceedings Second International Workshop*, pp. 50–62, 2001.
- [5] M. Montemerlo and S. Thrun. “Large-Scale Robotic 3-D Mapping of Urban Structures”. *The 9th International Symposium on Experimental Robotics*, 2004.
- [6] 加賀美聡, 岡田慧, 稲葉雅幸, 井上博充. “ロボット搭載用実時間視差画像生成システムの構成法”. 第4回ロボティクスシンポジウム予稿集, pp. 177–182, 1999.
- [7] C.A. Brarlow D.L. Stone and T.L. Conrwin. *Bayesian Multiple Target Tracking*. Artech House Publishers, Boston, MA.
- [8] M. Isard and A. Blake. “Condensation-conditional density propagation for visual tracking”. *International Journal of Computer Vision*, Vol. 28, pp. 5–28, 1998.

- [9] N. Ichimura. “Stochastic filtering for motion trajectory in image sequences using a Monte Carlo filter with estimation hyper-parameters”. *Proceedings of International Conference on Pattern Recognition*, Vol. 4, pp. 68–73, 2002.
- [10] S. Thompson, K. Kagami. “Stereo Vision and Sonar Sensor Based View Registration for 2.5 Dimensional Map Generation”. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and System*, pp. 3444–3449, 2004.
- [11] S. Thompson, 加賀美聡. “Multi-Sensor Localization for 2.5D Map Construction”. 日本機械学会ロボティクス・メカトロニクス講演会, 2004.
- [12] 増田健司, Simon Thompson, 加賀美聡, 金出武雄. “ステレオ視を用いた確立的位置同定法の精度検証”. 第22回日本ロボット学会学術講演会予稿集, p. 1B16, 2004.
- [13] K. Masuda, S. Thompson, K. Kagami, T. Kanade. “Verification of Stereo Vision Based Localization System”. *IEEE International Conference on Systems, Man and Cybernetics*, pp. 5435–5440, 2004.
- [14] W. Burgard, D. Fox, S. Thrun and F. Dellaert. *Sequential Monte Carlo methods in Practice*, chapter Particle filters for mobile robot localization, pp. 401–428. Springer-Verlag, 2001.
- [15] S. Nakamura, P. Heracleous, T. Yamada and K. Shikano. “Simultaneous recognition of multiple sound sources based on 3-D N-best search using microphone array”. *Proceedings of 6th European Conference on Speech Communication and Technology*, pp. 69–72, 1999.
- [16] 麻生英樹, 本村陽一, 吉村隆, 山本潔, 市村直行, 緒方淳, 原功, 浅野太. “パーティクルフィルタを用いた複数話者の位置と発話状態の追跡”. ベイジアンネットワークセミナー, 2003.

- [17] 麻生英樹, 小玉智志, K. ABDELAZIZ, 松本泰明, 本村陽一, 原功, 浅野太, 小笠原司, 新田恒雄, 柿倉正義. “確率的推論を利用したマルチモーダル対話制御”. 第 17 回人工知能学会全国大会論文集, pp. 1C1-04, 2003.
- [18] P. Cheeseman R. Smith, M. Self. *Estimating uncertain spatial relationships in robotics*, pp. 167–193. Springer-Verlag, 1990.
- [19] G. Dissanayake, H. Durrant-Whyte, and T. Bailey. “A computationally efficient solution to the simultaneous localization and map building(SLAM) problem”. *Working notes of ICRA’2000 Workshop on Mobile Robot Navigation and Mapping*, 2000.
- [20] J.J. Leonard and H.J.S. Feder. “A computationally efficient method for large-scale concurrent mapping and localization”. *Proceedings of the Ninth International Symposium on Robotics Research*, pp. 169–176, 1999.
- [21] F. Lu and E.Milios. *Globally consistent range scan alignment for environment mapping*, pp. 4:333–349. 1997.
- [22] C. Martin and S. Thrun. “Online acquisition of compact volumetric maps with mobile robots”. *IEEE International Conference on Robotics and Automation*, 2002.
- [23] D. Avots, E. Lim, R. Thibaux, and S.Thrun. “A probabilistic technique for simultaneous localization and door state estimation with mobile robots in dynamic environments”. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and System*, Vol. vol.1, pp. 521–526, 2004.
- [24] 脇田淳一, 近藤敏之, 伊藤宏司. “全方位視覚を用いた移動ロボットの自己位置同定”. 第 22 回日本ロボット学会学術講演会予稿集, p. 1B14, 2004.
- [25] C. Jennings, D. Murray, J.J Little. “Cooperative Robot Localization with Vision-based Mapping”. *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2659–2665, 1999.

- [26] G. Kitagawa. “Monte Carlo filter and smoother for non-Gaussian nonlinear state space models”. *Journal of Computational and Graphical Statistics*, Vol. 5, pp. 1–25, 1996.
- [27] N. Gordon A. Doucet, N. Freitas. *Sequential Monte Carlo methods in Practice*, chapter An introduction to sequential Monte Carlo method, pp. 3–14. Springer-Verlag, 2001.
- [28] J.J. Kuffner. “Efficient Optimal Search of Uniform-Cost Grids and Lattices”. *Proceedings of International Conference on Intelligent Robots and System*, pp. FA2–G4, 2004.
- [29] 友納正裕. “大域スキャンマッチングと複数仮説追跡を用いたロバストな自己位置推定”. 第 22 回日本ロボット学会学術講演会予稿集, p. 1B12, 2004.
- [30] 友納正裕. “スキャンテンプレートマッチングによる移動ロボット場所認識”. 第 22 回日本ロボット学会学術講演会予稿集, p. 1B13, 2004.

付録

A. ベイズアプローチ

A.1 隠れ状態推定

ある直接観測不可能な内部状態 (隠れ状態) を持つシステムについて, 一連の観測データの系列 $y(1), \dots, y(T)$ から, 隠れ状態の系列 $\Theta(1), \dots, \Theta(T)$ (の一部) を推定する問題において, ベイズ的なアプローチでは $y(1), \dots, y(T)$ および $\Theta(1), \dots, \Theta(T)$ を確率変数として, それらの間の関係を確率密度関数によって捉えることで様々な推論を行なう. 以下では簡単のために $y(1), \dots, y(T)$ を $y_{1:t}$ などと書く.

最も基本的な推論は隠れ状態の系列の事後確率分布 $p(\Theta_{1:t} | y_{1:t})$ を求めることであるが, この他にも, 事後確率を最大にする $\Theta(t)$ の系列 (MAP 解) を求めることや, これまでの観測結果から現時点の隠れ状態の事後確率分布 $p(\Theta(t) | y_{1:t})$ を求めること (フィルタ), 未来の隠れ状態や未来の観測値の事後確率分布を求めること (予測), 過去に遡って隠れ状態の事後確率分布を求めること (円滑化), などが行われる. これらの計算をおおざっぱにまとめてフィルタリングと呼ぶこともある.

A.2 マルコフ性の仮定

$y(t)$ と $\Theta(t)$ の確率的な関係をモデル化する場合に, 順過程 (隠れ状態から観測値への生成モデル) のほうがモデル化しやすいことが多い. モデルの複雑さを実際に扱える程度にするためにいろいろな仮定が使われるが, 最も典型的なものは以下のような, マルコフ性の仮定を用いるものである. 隠れ状態と観測値の同時分布 $p(\Theta_{1:t}, y_{1:t})$ は,

$$p(\Theta_{1:T}, y_{1:T}) = \prod_{t=1}^T p(\Theta(t), y(t) | \Theta_{1:t-1}, y_{1:t-1})$$

と分解できる. さらに, 各要素の $p(\Theta(t), y(t) | \Theta_{1:t-1}, y_{1:t-1})$ は,

$$p(y(t) | \Theta_{1:t-1}, y_{1:t-1}) p(\Theta(t) | \Theta_{1:t-1}, y_{1:t-1})$$

と分解できる．

第1項 $p(y(t) | \Theta_{1:t-1}, y_{1:t-1})$ にマルコフ性を仮定して，

$$p(y(t) | \Theta(t)) = p(y(t) | \Theta_{1:t-1}, y_{1:t-1})$$

が成り立つとする．さらに，第2項 $p(\Theta(t) | \Theta_{1:t-1}, y_{1:t-1})$ にもマルコフ性を仮定して，

$$p(\Theta(t) | \Theta(t-1)) = p(\Theta(t) | \Theta_{1:t-1}, y_{1:t-1})$$

とする． $p(\Theta(t) | \Theta(t-1))$ および $p(y(t) | \Theta(t))$ は，それぞれ，状態遷移確率 (状態遷移ダイナミクス)，観測確率 (センサモデル) などと呼ばれる．

以上をまとめると，マルコフ性の仮定の下では同時分布を，状態遷移確率と観測確率を使って

$$p(\Theta_{1:T}, y_{1:T}) = \prod_{t=1}^T p(y(t) | \Theta(t)) p(\Theta(t) | \Theta(t-1))$$

と分解できる．これはまた， $\Theta_{1:T}, y_{1:T}$ の関係が Fig:40 のような単純な構造な dynamic probabilistic network (dynamic belief network, dynamic Bayesian network:DBN) によって表現できるということでもある．

$y(t), \Theta(t)$ が連続値を取り，状態遷移ダイナミクスや観測過程が線形変換とガウシアンノイズによってモデル化できる場合には，これはカルマンフィルタ (Kalman filter) が仮定している時系列モデルになる．また， $\Theta(t)$ が離散値を取る場合には，音声認識などに用いられている隠れマルコフモデル (hidden Markov model: HMM) になる．

ロボットの位置における仮定を行なうのにガウス分布に従うカルマンフィルタと違って，本研究では，環境マップ上のロボットの位置を表す任意の確率密度関数の予測近似を行なう．任意の確率密度関数を予測近似する事によってノイズが乗るセンサデータにおいてもマルチモーダルなロボットの位置仮説を行なうことができる．つまり，データのエラーが存在してもよりロバストな予測近似を実現することができる．

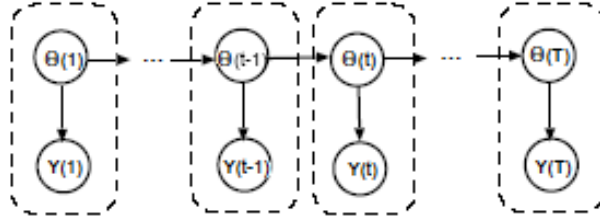


図 40 隠れ状態推定のための動定期ベイジアンネット

A.3 事後確率の漸化式

A.2のようなマルコフ性の仮定の下では，効率の良い確率計算に役に立ついろいろな漸化式が成り立つことが知られている．たとえば， $p(y_{1:t}, \Theta(t))$

$$p(y_{1:t}, \Theta(t)) = p(y(t) | \Theta(t)) \times \sum_{\Theta(t-1)} p(\Theta(t) | \Theta(t-1)) p(y_{1:t-1}, \Theta(t-1))$$

という前向き漸化式が成り立つ．同様に， $p(y_{t+1:T} | \Theta(t))$ については，

$$p(y_{t+1:T} | \Theta(t)) = \sum_{\Theta(t+1)} p(\Theta(t+1) | \Theta(t)) \times p(y(t+1) | \Theta(t+1)) p(y_{t+2:T} | \Theta(t+1))$$

という後ろ向き漸化式が成り立つ．これらの漸化式は，隠れマルコフモデルにおいて信号出力確率を求めるために用いられる前向きアルゴリズム (forward algorithm) や後ろ向きアルゴリズム (backward algorithm) の基礎になるものであり，また，Fig:40のようなDBNに対して $p(\Theta(t), y_{1:T})$ などを計算するための確率伝播アルゴリズムとも関係が深い．

最初の漸化式から，事後確率 $p(\Theta(t), y_{1:t})$ について，

$$p(\Theta(t) | y_{1:t}) = \frac{1}{Z} p(y(t) | \Theta(t)) \times \sum_{\Theta(t-1)} p(\Theta(t) | \Theta(t-1)) p(\Theta(t-1) | y_{1:t-1})$$

という漸化式が成り立つ．ここで Z は $\Theta(t)$ によらない正規化の定数である．

$\Theta(t)$ が離散値を取るときには，観測地 $y_{1:t}$ が得られたときの， $p(\Theta(t) | y_{1:t})$ を $\Theta(t)$ の離散値の順に並べた確率ベクトルを $p(t)$ ，観測確率 $p(y(t) | \Theta(t))$ を $\Theta(t)$

の値の順に並べたベクトルを $p_y(t)$, $p(\Theta(t) | \Theta(t-1))$ を成分とする状態遷移確率行列を $T(t, t-1)$ とすれば ,

$$p(t) \propto p_y(t) \times T(t, t-1) p(t-1)$$

と書ける . ここで \times はベクトルの成分ごとの積である .

事後確率 $p(\Theta(t) | y_{1:t})$ は , 状態空間が連続で , 状態遷移ダイナミクスや観測変換が線形変換であり , ノイズが正規分布とみなせれば , カルマンフィルタのアルゴリズムで効率よく計算できる . しかし , それ以外の場合 , すなわち , 離散状態で状態数がとても多い場合や , 連続状態で線形性や正規性が成り立たない場合には , $\Theta(t-1)$ の取り得る値についての総和を取る計算が実際的に困難になることが多い . 本研究の問題においては , 推定対象であるロボットの位置や姿勢の数 N が増えるにつれて , 隠れ状態 $\Theta(t)$ の取り得る値の数は C^N の形で増えてゆく . 従ってベイズアプローチによる計算を厳密に実行できるのは N が小さい範囲だけである .

B. NOMAD200 の主な仕様

表 1 NOMAD200 主な仕様一覧

直径	本体	18 inch
	バンパー	21 inch
高さ		30 inch
重量 (バッテリー込み)		59 kg
最大加速度	並進	$2.5 \text{ inch}/s^2$
	回転	$2 \text{ rad}/s^2$
最大速度	並進	20 inch/s
	回転	1 rad/s
走行システム	走行タイプ	3 ホイール, ノンホロノミック
	回転挙動	上部独立可動
ロータリーエンコーダ解像度	並進	1854 counts/cms
	上部回転	1510 counts/degree
	下部回転	1510 counts/degree
バッテリー消費電力		432 W/h(PC 用:288 W/h, センサ用:144 W/h)
CPU		Intel Pentium 800MHz
タッチセンサ	数量	20 channel
	感度	8 once
ソナーセンサ	数量	16 channel
	測定距離	17-255 inch
	測定精度	1 %
	照射角	25 °
近接センサ	数量	16 channel
	測定距離	~ 24 inch
無線 LAN		IEEE802.11a(NEC AtermWL54TE)

C. ステレオカメラ主な仕様

表 2 MEGA-DCS の主な仕様一覧

撮像素子	1/2 インチ CMOS カラー/白黒プログレッシブスキャン
フォーマット	1280 × 960, 640 × 480, 320 × 240 8bit 白黒又は Bayer color pattern
フレームレート	3.75, 7.5, 15, 30Hz 1280 × 960 での最大フレーム数 7.5Hz
ゲイン	0 ~ 16dB
SN 比	55dB 未満
電源	1W 未満
同期	60 μ S 未満
レンズ焦点距離	4.8mm C マウント

D. ファイバオプティカルジャイロの主な仕様

表 3 ファイバオプティカルジャイロの主な仕様一覧

項目	単位	アナログ出力	デジタル出力
角度計測範囲	°	± 180	± 180
角加速度計測範囲	°/s	± 200	± 200
レンジ	°	± 10,20,45,90,180	± 180
	°	± 10,20,50,100,200	± 200
分解能	°	0.01 以下 (± 10 °)*	0.01 以下
	°/s	0.01 以下 (± 10 °/s)*	0.01 以下*
直線性	%FS	± 0.1 以下	
バイアス安定性	°/h	± 3 以下 (全作動温度域にて)	
角度ドリフト	°/h	± 0.5 以下 (バイアス補正後温度安定下)	
周波数応答	Hz	20	送信レートによる
出力電圧	VDC	± 5()	—
ミスアライメント	mrاد	7 以下	

* ローパスフィルタ (10Hz 以下) を接続した場合上記角度・角速度の読み取りが可能。

ここで用語の定義を簡単に説明する。

- 直線性 : 入力加速度または角度に対する出力に対する直線性で、スケールファクタの非直線性、非対称性及び温度安定性を含む。
- バイアス安定性 : 全作動温度域におけるバイアス (角度ドリフト) 安定性。ただし、温度変化率が 0.5 /min 以下とする。
- 角度ドリフト : バイアス補正後の温度安定状態における角度ドリフト。
- ミスアライメント : 取付基準面に対する入力軸の傾き。