地図の有用性を考慮した地図更新作業における行動計画*

田中 完爾^{*1}, 長谷川 勉^{*2}, 近藤 英二^{*1}

Motion Planning in a Map Updating Task by Considering Utility of a Map

Kanji TANAKA^{*3}, Tsutomu HASEGAWA and Eiji KONDO^{*3} Graduate School of Engineering, Kyushu University,
6-10-1 Hakozaki, Higashi-ku, Fukuoka-shi, Fukuoka, 812-8581 Japan

Many mobile robots use maps for path planning. However, due to changes of obstacle configurations in the environment, utility of the map is often deteriorated. In this paper, we address a problem of map updating task, where a robot automatically explores the environment to update a map. Map updating task is different from well known map making task in that the robot can estimate global structure of the environment by using the initial map. This enables a map updating robot to take into account possible map users, and to estimate utility of a map for them. We propose a method for motion planning in the task, for quickly increasing utility of a map. We also describe an algorithm to execute the planning in a practical computing time. A number of experiments have been performed in various environments and changes, to demonstrate the effectiveness of the method.

Key Words: Mobile Robot, Environment Changes, Map Updating Task, Observation Planning

1. は じ め に

人間の生活環境を動きまわりながら, 床面清掃, 荷物運搬, アミューズメントなどのタスクに従事する自律ロボットの実現が望まれている. ロボットが効率的な移動経路を計画するために, 障害物地図が有用であり, 多くのロボットシステムで用いられてきた⁽⁶⁾⁽⁹⁾⁽¹²⁾. その一方で, 環境内の物体配置が変化すると, 地図の有用性が著しく低下するという問題があった⁽¹⁵⁾.

この問題を解決するために, ロボットが自律的に地図を更新することができれば, 自動化・省設備化の観点から望ましい. その実現には, 2 通りの方法が考えられる.

1. ロボットが物体配置の変化に遭遇するたびに, 地図を更新する⁽¹¹⁾⁽¹⁷⁾.
2. 環境の探索とセンシングをともなう地図更新作業を, 適切な時間間隔で実行する⁽⁸⁾⁽¹⁴⁾.

特に 2. は, 通常の作業の実行を妨げない方法として期待できる.

未知環境における地図作成作業⁽¹⁾⁽⁴⁾⁽¹⁵⁾に関しては, これまでにいくつかの行動計画法が提案されており⁽²⁾⁽³⁾⁽⁷⁾⁽¹⁷⁾, これらは地図更新作業においても, ある程

度, 有効であると考えられる. そのほとんどは, 欲張り戦略に基づいて, 得られる情報の量を最大化するように観測地点を決定する⁽¹⁵⁾⁽¹⁶⁾. Zelinsky は, それぞれの地点で得られる情報の量を考慮に入れた, 距離変換に基づくアルゴリズムを提案した⁽¹⁷⁾. Choset らは, Generalized Voronoi Graph と呼ばれるロードマップをセンサデータを用いて更新しながら, センサベースの探索行動をとる方法を提案した⁽³⁾. Burgard らは, 複数台のロボットの行動を決定して, 効率よく環境を探索するアルゴリズムを提案した⁽²⁾.

地図作成作業では, 作業がほとんど完了するまで, 環境全体の経路構造を知ることができない. 必然的に, 行動計画は経路構造を考慮しないものとなり, “行き止り”や“部屋の隅”のように, ほとんどの地図利用者にとって重要でない領域についても, 必要以上に探索・観測してしまう. これに対し, 地図更新作業では, 過去の地図から, 経路構造の手掛かりを得ることができる. そのため, 適切な観測地点を配置して, 低コストで地図の有用性を向上させることが期待できる. ただし, 現在の環境と地図が部分的に異なっている可能性があることに注意する必要がある.

以上の観点から, 本論文では, 地図利用者(ロボット)にとっての有用性を考慮した, 地図更新作業のための観測行動計画法を提案する. この方法は, 仮想的な複数の地図利用者による多数決により, 検出すべき重要

* 原稿受付 2003 年 4 月 15 日.

^{*1} 正員, 九州大学大学院工学研究院(〒812-8581 福岡市東区箱崎 6-10-1).^{*2} 正員, 九州大学大学院システム情報科学研究院.

E-mail: kanji@mech.kyushu-u.ac.jp

な変化を選ぶ。その結果、多くの地図利用者の移動コストが改善されることが期待される。地図中の局所的な特徴に基づいて考慮すべき重要な変化を絞りこむ方法、および変化の重要度を評価するアルゴリズムを示す。多様な環境と変化を模したシミュレーションの結果を示し、本方法の有効性を検証する。

2. 問題設定

2.1 環境変化 ロボットは、2種類の物体配置変化に遭遇する可能性がある(図1)。1つは、地図に記されていない物体の出現であり、これを物体出現と呼ぶ。もう1つは、地図に記されていた物体の消失であり、物体消失と呼ぶ。両者をまとめて、環境変化と呼ぶことにする。環境内での物体の移動は、元の場所における物体消失と、現在の場所における物体出現の組み合わせであるとして扱う。固定された障害物の場所や形は、予め与えられているものとする。

2.2 地図 2次元の格子地図を用いる。各格子(セル)は、床面上の正方形領域に対応し、以下の3種類のいずれかの状態をとる。

- **FIXED.OBSTACLE**: 固定された障害物(例: 壁、柱)により占有されている
- **MOVABLE.OBSTACLE**: 移動可能な障害物(例: 机、椅子)により占有されている
- **FREE**: 障害物に占有されていない

2.3 行動計画の目的 地図更新作業における課題は、多くの地図利用者にとって“重要な”環境変化を検出することである。そのような環境変化を検出できそうな観測地点を見つけ出すことを、行動計画の主な目的とする。

ある環境変化の重要度は、仮想的な地図利用者(ロボット)を用いて評価する。各地図利用者を、そのスタート地点とゴール地点のペアにより表す。そして、ある環境変化の知識を用いずに計画された最短経路よりも、用いて計画された最短経路の方が、移動コストが改善されるならば、その環境変化は地図利用者にとって重要であるという。

任意のFREEセルのペアを、各地図利用者のスタート地点およびゴール地点とみなし、地図利用者の集合を得る。そして、環境変化の重要度を、その環境変化を重要であるとみなす地図利用者の数と定義する。

重要度は、局所的な空間特徴(環境変化の位置、周囲の物体配置)に強く依存する。例えば、行き止りや部屋の隅などは、ほとんどの地図利用者が通行しない領域であり、そこで生じうる環境変化の重要度は低い。一方、部屋の出入口は、より多くの地図利用者が通行する

(a) original environment (b) modified environment

Fig. 1 Changes in an indoor environment. (Black polygons indicate objects that are changed.)

ため、重要度が高い。つまり、各地点で重要度の高い環境変化が生ずるかどうかが、その地点周囲の地図情報をもとに、ある程度判断することができる(3.1節)。

3. 環境変化の予測および評価

3.1 Critical change 環境変化の候補は無数にあるため、評価すべき重要な環境変化を絞りこんでおく。各地点 P における環境変化が重要であるかどうかを判断するために、図2(a)に示すような P を中心とする局所領域を設定し、局所領域内の任意の(P を含まない)2点間の通行可能性を調べる。ただし、通行可能性とは、一方の地点から他方の地点へ到達する経路が存在するかどうかを表す。

P において環境変化が生じた場合に、もし通行可能性が変化しないと予想されるならば、地図利用者は、局所領域内でのみ経路を変更すればよい。この変更による移動コストの増減は、局所領域内の最長経路と最短経路のコスト差よりも小さく抑えられる。この場合、 P における環境変化は重要でないと判断する。

もし通行可能性が変化すると予想されるならば、局所領域外でも経路を変更する必要が生じ、移動コストが大きく増減する可能性がある。しかも、局所領域内の情報だけでは、その増減の大きさを推定できない。このような環境変化を重要とみなし、Critical changeと定義する。Critical changeが生じうる領域をCritical regionと呼ぶ。

Critical changeには、以下の2種類がある。

- 物体が出現して狭い経路を塞いだ。これにより、地図利用者は回り道をしたり、後戻りをしなければならない⁽⁵⁾⁽¹³⁾。
- 狭い経路を塞いでいた物体が消失した。これにより、地図利用者は近道をしたり、これまで行けなかった場所に行けるようになったりする。

以上を要約すると、Critical regionは、通行可能性が変化する狭い経路であるといえる。

Fig. 2 (a) Local region and 17 check points. (b) Local region is covered by circles centered at each check point. (c) Using local region for determining critical regions.

3.2 CR テスト 本節では, Critical region を抽出するためのテスト (以下, CR テスト) について述べる. このテストは, 地図中の各セルに対応する地点 P が, Critical region に属するかどうかを判定する. CR テストの完了後, Critical region 内にあると判定されたセル群の連結した塊 (クラスタ) をすべて求め, 各クラスタを Critical region とみなす.

前節の定義より, ある地点が Critical region に属するかどうかを調べるには, 局所領域内の各地点の通行可能性をチェックすればよい. 通行可能性を調べるためには, 各地点候補に地図利用者がいると仮定して, 障害物と重なるかどうかを調べればよい. しかし, これには, 地図利用者のサイズと地点候補数に比例した計算コストがかかる. そこで, 通行可能性を効率良く調べるために, C-obstacle 領域を導入する⁽¹⁰⁾. ここでは, 地図利用者を, それを囲むような最小の円で近似する. この円の半径を R とする. このとき, C-obstacle 領域は, 障害物中のすべての点を, 半径 R だけ拡張した領域として定義される. C-obstacle 領域は, 簡単なモルフォロジー演

算によって得られ, 地図利用者が侵入できない領域を表す. ある地点の通行可能性を調べるには, その地点が C-obstacle 領域内にあるかどうかを調べればよい. これには, 地点候補数に比例した計算コストしかかからない. 本方法では, 2 種類の c-obstacle, CO1 と CO2 を用いる. これらは, それぞれ FIXED.OBSTACLE 領域および MOVABLE.OBSTACLE 領域を拡張したものである.

局所領域内には, 通常, 数十個 (セルのサイズが $30[cm] \times 30[cm]$, 局所領域のサイズが $4[m^2]$ の場合で約 44 個) のセルがあり, 各セルの状態には, 「CO1 に含まれる」「CO2 に含まれる」「通行可能である」の 3 種類がある. これらの全ての組み合わせは膨大な数となるため, 局所領域内を離散化し, 17 個の点 (以下, チェックポイント) の通行可能性のみをチェックする. ただし, 離散的なチェックポイント V を, 極座標系で以下のように表現する.

$$V = P + (r \cos \phi, r \sin \phi) \quad (1)$$

$$\phi = 0, \frac{\pi}{4}, 2\frac{\pi}{4}, \dots, 7\frac{\pi}{4} \quad r = 0, R_1, R_2 \quad (2)$$

R_1, R_2 の決定方法は後述する.

CR テストでは, Critical region の性質が満たされるかどうかを, 以下の 2 つの条件により判定する.

- 条件 1: P は通行可能性が変化しうる領域内にある

$$f_1 = \bigwedge_{r=0, R_1, R_2} \bigvee_{\phi} (|\phi - \theta| \leq \pi/4) \wedge (V \notin \text{CO1}) \quad (3)$$

- 条件 2: P は狭い経路上にある

$$f_2 = \bigwedge_{r=R_1, R_2} \bigvee_{\phi} (|\phi - \theta| \leq \pi/4) \wedge (V \in \text{CO1} \vee V \in \text{CO2}) \quad (4)$$

ただし θ は経路の方向を表すパラメータである. 判定式は, 以下のように表される.

$$f = \bigvee_{\theta=0, \frac{\pi}{4}, 2\frac{\pi}{4}, 3\frac{\pi}{4}} f_1 f_2 \quad (5)$$

このように, 論理演算により表現することで, 高速な処理を実現することができる.

チェックポイントの通行可能性をチェックすれば, その点を中心とする半径 R の領域内に, 障害物があるかどうかを調べることができる. したがって, R_1, R_2 を決定する際は, 17 個のチェックポイントを中心とする半径 R の円によって局所領域が覆われ, かつ, 互いの円の重なりが小さくなるようにすべきである. 本論文では, そのような値として, $R_1 = 1.8R, R_2 = 2.6R$ を選んだ. これらは, 注目点から 8 方向に伸びる直線上にチェックポイントを配置するとき (図 2(a),(b)), 互いの円の重なりが最小となる R_1, R_2 の値である.

図 2(c) では, 地点 1 の $\theta = 0, 2\pi/4, 3\pi/4$ について, 地点 2 の $\theta = \pi/4, 2\pi/4, 3\pi/4$ について, 条件 1, 2 が満たされる。したがって, 地点 1, 2 は, Critical region に属すると判断する。一方, 地点 3 は固定された障害物領域内にあるため, 条件 1 をいずれの θ についても満たさず, 地点 4 は条件 2 をいずれの θ についても満たさない。したがって, 地点 3, 4 は Critical region に属さないと判断する。

3.3 重要度の評価 ある Critical region 内に環境変化が生じた場合に移動コストが大きく増減する, と予想される地図利用者の数を, その Critical region の重要度 N と定義した。そのような地図利用者は, 以下の 2 種類に分類することができる。

- グループ 1 : Critical region が通行可能な場合のみ, ゴール地点に到達できる。
- グループ 2 : Critical region が通行不能であっても, 迂回経路を通してゴール地点に到達できる。

図 3 中の $(P_1, P_3), (P_1, P_2)$ は, それぞれグループ 1, 2 の地図利用者の例である。グループ 1 および 2 中の地図利用者の数, N_1 および N_2 を個別に求め, N_1 と N_2 を合計することにより N を得る。

ある Critical region によって, 複数の FREE 領域 (例: 図 3 の free region 1, 2) が互いに隔てられている状況を想定しよう。これらの FREE 領域の集合を F とすると, 異なる 2 つの FREE 領域 $f, f' \in F$ 内にスタート地点とゴール地点がある地図利用者の集合が, グループ 1 となる。したがって, N_1 は

$$N_1 = \sum_{f, f' \in F} n_f n_{f'} \quad (6)$$

と表される。ただし, $n_f, n_{f'}$ は FREE 領域 f, f' 内のセル数である。 F の要素である孤立した FREE 領域は, 連結した FREE セルに同じラベルを付けるというラベリングの手続きにより求められる。

地図利用者がグループ 2 に属すかどうかを判断するには, Critical region が通行可能である場合と, そうでない場合の移動コストを比較する必要がある。

前者の場合, 最短経路は Critical region を境にして, 2 つに分割することができる。したがって, Critical region を基準とした距離変換⁽⁹⁾を行うことにより, 全ての地図利用者 (P_1, P_2) の移動コスト $D_{P_1 P_2}$ が求まる。

$$D_{P_1 P_2} \simeq D_A(P_1) + D_B(P_2) \quad (7)$$

ただし, $D_A(P), D_B(P)$ は, それぞれ地点 A, B から地点 P までの距離を返す関数である。 A, B は Critical region に隣接したセルであり, $D_A(B)$ の値が最大となるように選ぶ。

Fig. 3 Two kinds of map users $(p_1, p_2), (p_1, p_3)$ for which the Critical region is important.

一方, 後者の最短経路には, 全ての地図利用者に共通する経由点は存在しない。理想的には, 全ての地図利用者について, スタート地点を基準とした距離変換を行えば, 移動コストが求まるが, 地図利用者が多数あるため (セルのサイズが $0.3[m] \times 0.3[m]$, 環境のサイズが $90[m^2]$ の場合で 1000), 現実的ではない。これに代わるヒューリスティクスとして, 一定の移動コスト $\frac{1}{2}D_A(B)$ を用いることにする。すなわち, 以下の条件を満たす地図利用者の数を N_2 とする。

$$D_A(P_1) + D_B(P_2) \leq \frac{1}{2}D_A(B) \quad (8)$$

この条件を満たす (P_1, P_2) は,

$$D_A(P_1) + D_{P_1 P_2} + D_B(P_2) \geq D_A(B) \quad (9)$$

という性質から, 地図利用者がグループ 2 に属する十分条件:

$$D_{P_1}(P_2) \geq D_A(P_1) + D_B(P_2) \quad (10)$$

を満たす。

4. 行動決定アルゴリズム

本章では, Critical region の特性および重要度を活用した, 1 つの行動決定アルゴリズムを示す。

地図更新作業では, 以下のステップを繰り返す。

1. 観測を行い地図を更新する。
2. 次のサブゴールを決定する。
3. 次のサブゴールまでの最短経路を, 距離変換により求める。
4. サブゴールに到達するか, 既に観測された領域から出そうになるまで移動する。

ただし, 各 Critical region が通行可能であるかによらず到達可能な, サブゴールのみを選ぶことにした。

ステップ 2 では, 重要度 N が高い critical region の大きな部分を視野内に含める地点を, 次のサブゴールとして選びたい。そこで, 以下の手続きにより, サブゴールの各候補 P を評価する。(1) 環境が地図と同じであ

るという仮定の下で, P における観測範囲を推定する.

(2) 観測範囲内の各セルに, 以下の得点を与える.

$$\begin{cases} v & (\text{Critical region 内のセルである場合}) \\ 1 & (\text{FREE セルである場合}) \\ 0 & (\text{上記以外の場合}) \end{cases}$$

v = Critical region の重要度 N / Critical region の面積

(3) その合計得点を評価値とみなす. そして, 最も評価値の高かった候補をサブゴールとして選ぶ.

5. シミュレーション

シミュレーションでは, 多様な環境について地図更新作業を行った. 環境のサイズを $20[m] \times 40[m]$, セルのサイズを $30[cm] \times 30[cm]$ とした. ロボットは全方位距離センサを搭載しており, このセンサは, 半径 $3[m]$ 内にある物体を正確に計測することができる. パラメータ R は $60[cm]$ に設定した.

5.1 提案法と LS 図 4(a) に, ある環境における作業の様子と結果を示す. この環境には開けた場所と狭い場所の両方がある. 図 5(a) の “initial map” に, 初期地図を示す. 図中で, 長い直線分が固定障害物を表す. ロボットは, 図 6 に示す順序で, 各時点において最も重要な Critical region を優先的に観測した.

提案法との性能比較のために, ロボットによる地図作成や環境探索に有効な Greedy Mapping 法を用いる⁽¹⁶⁾. Greedy Mapping は, 次の観測地点を, 既に観測された領域 (既観測領域) の近くに選ぶ. これにはいくつかの戦略があるが, ここでは FREE セルが最も多く観測できそうな地点を, 既観測領域の中から選ぶことにする. すなわち, 地図が正しく, その地点にロボットがいると仮定したとき, 観測範囲内にある FREE セルの数を評価値とする. この戦略を Local Search (LS) と呼ぶことにする. 図 4(b) は, LS を用いた作業の様子を示す. 図に示されるように, LS は広い範囲を見渡せそうな, 開けた場所を優先的に観測した.

5.2 エラー関数 作業により逐次更新される地図の有用性を評価するために, エラー関数を導入する. エラー関数の値は, 以下のようにして算出する. (1) 地図利用者のサンプルをランダムに 100 個生成する. (2) それぞれのサンプルについて, 評価対象の地図を用いた場合の移動コスト c_1 と, 正しい地図を用いた場合の移動コスト c_2 を計算し, その差異 $\Delta c = c_1 - c_2$ を求める. (3) 全サンプルに対する Δc の平均を求める.

ここで, 地図利用者のサンプルは, 以下の 2 つの条件を満たすように選んだ. (1) 初期地図と環境の両方において, スタート地点からゴール地点へ到達する経路が

Fig. 4 Tasks with (a) the proposed method, or (b) LS. (obs.” is an abbreviation of “observations”.)

存在する. (2) 初期地図を用いた場合, 正しい地図を用いた場合よりも, 最短経路のコストが高い.

また, コスト c_1, c_2 は, 地図利用者の移動を模擬することにより求める. この移動中に, もし地図利用者が未知の障害物に遭遇したならば, その地点で再計画を行うことになる. この場合, その地点に到達する前のコストと後のコストを合計する.

5.3 結果と考察 図 5(a) に, ある作業中の各地点について, エラー関数をプロットしたグラフを示す. 異なる 3 つのスタート地点について, 両計画法の結果が示されている. “environment” 図は, 環境を示す. グ

Fig. 5 Performances in four different environments. The environments in (a),(b) have both narrow regions and an open space. The environment in (c) has only narrow regions. The environment in (d) has only an open space.

ラフの横軸は観測回数を示し、縦軸はその時点におけるエラー値を示す。“start location #2”は図 4(a),(b)の作業を表している。提案法では、地点#13 と#22 において、地図が大きく改善されている。これは、その地点において Critical change が検出されたためである。

図 5(a),(b),(c),(d) は、4 つの異なる実験環境における結果を示している。(1) 環境中に開けた場所と狭い領域の両方がある場合 (図 5(a),(b)), 提案法のみが Critical changes を検出でき、LS よりも高い性能を示した。(2) 狭い領域か広い場所の一方しかない場合 (図 5(c),(d)), 提案法と LS の性能に有意な差がみられなかった。こ

れは、ほとんど全ての領域が Critical region となるか、Critical region が全くないことから、Critical region を考慮する効果が現れなかったためである。(3) スタート地点が Critical change に近い場合 (図 5(a) の start location #3), LS も結果として Critical change を検出できているため、やはり有意な差がみられなかった。いずれの場合も、経路構造、物体の密度、出入口の数などの条件に依存せずに提案法は高い性能を示した。

ここまでは、ロボットが地図利用者の半径 R を正確に推定できていると仮定していた。実際には、この仮定は必ずしも成り立たない。そこで、 R とその推定値 $\hat{R}$

Fig. 6 The order of observing each critical region. (Numbers refer to ID of each observation.)

が異なる状況を想定して、提案法の有効性を検証した。 $\hat{R} = 0.4R, 0.6R, 0.8R, 1.0R, 1.2R, 1.4R, 1.6R$ の各場合を想定した。環境と地図は Fig.5(a) に示すものを用いた。そして、各 $\hat{R}$ を用いて、各観測地点候補を評価した。図 7 のグラフに、評価値 (横軸) と、その地点で実際に観測を行ったときのエラー値の減少 (縦軸) との関係を示す。グラフ中で左から右にゆくほど、評価値が低くなる。

図 7(a),(b),(c) より、 $\hat{R} < R$ の場合には、評価値とエラー値の減少との相関が低い。すなわち、評価値が高い観測地点を選んだ方が、エラー値が減少しやすいとはかぎらない。この原因として、実際よりも小さい地図利用者を想定しているため、多くの Critical region を見落してしまうことが考えられる。これに対し、図 7(d),(e),(f),(g) より、 $\hat{R} \geq R$ の場合には、評価値が高い観測地点を選んだ方が、エラー値の減少を期待できることが分かる。以上より、 $\hat{R} < R$ とならないように注意すれば、 $\hat{R}$ に誤差があっても、提案法は有効に機能するといえる。

6. お わ り に

本論文では、地図更新作業における地図の有用性を考慮した行動計画法を提案した。提案法は、地図利用者の移動コストに大きな影響を与えそうな環境変化を、重要なものととらえ、重要な環境変化が生じうる地点を優先的に観測する。地図中の局所的な特徴のみをもとに、重要な環境変化が生じうる領域、Critical region を素早く絞りこむ方法について述べた。また、各 critical region における環境変化の重要度を、効率良く評価するアルゴリズムを示した。実際の環境を模したシミュレーションにおいて、経路構造、物体の密度、出入口の数などの条件に依存しない提案法の有効性が示された。

Fig. 7 Results of viewpoint planning with different estimation $\hat{R}$ of R : (a) $\hat{R} = 0.4R$, (b) $\hat{R} = 0.6R$, (c) $\hat{R} = 0.8R$, (d) $\hat{R} = 1.0R$, (e) $\hat{R} = 1.2R$, (f) $\hat{R} = 1.4R$, (g) $\hat{R} = 1.6R$. (horizontal axis: ID of location sorted according to the evaluation value. vertical axis: reduction of error [%].)

文 献

- (1) Borthswick S. and Durrant-Whyte H., Simultaneous Localization and Map Building for Autonomous Guided Vehicles, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, (1994), 761-768.
- (2) Burgard W., Collaborative Multi-Robot Exploration, *Proc. of IEEE Int. Conf. Robotics and Automation*, (2000).
- (3) Choset H. and Burdick J., Sensor Based Exploration, The Hierarchical Generalized Voronoi Graph, *Int. Journal of Robotics Research*, 19(2000), No.2, 96-125.
- (4) Dissanayake G., Durrant-Whyte H. and Bailey T., A Computationally Efficient Solution to the Simultaneous Localization and Map Building (SLAM) Problem, *Proc. IEEE/RSJ Int. Conf. Robotics and Automation*, (2000), 1009-1014.
- (5) Fabrizi E. and Saffiotti A., Extracting Topology-Based Maps from Gridmaps, *Proc. IEEE/RSJ Int. Conf. Robotics and Automation*, (2000), 2972-2978.

- (6) Hofner C. and Schmidt G., Path Planning And Guidance Techniques For An Autonomous Mobile Cleaning Robot, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, (1994), 610-617.
- (7) Iijima J. and Yuta S., Searching Unknown 2-D Environment by a Mobile Robot with a Range Sensor, *Journal of Computers Elect. Engng*, 18(1992), No.1, 83-98.
- (8) Jennings J., Kirkwood-Watts C. and Tanis C., Distributed Map-making and Navigation in Dynamic Environments, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, (1998), 1695-1701.
- (9) Lee D., *The Map-Building and Exploration Strategies of a Simple Sonar-Equipped Mobile Robot*, Cambridge University Press, (1996).
- (10) Lozano-Pérez T., Spatial Planning: A Configuration Space Approach, *IEEE Trans. Computers*, C32(1983), No.2, 108-119.
- (11) Magin G., Ruß A., Burschka D. and Färber G., A Dynamic 3D Environmental Model with Real-Time Access Functions for Use in Autonomous Mobile Robots, *J. Robotics and Autonomous Systems*, 14(1995), 119-131.
- (12) Shieh J. S. and Calvert T. W., View and Route Planning for Patrol and Exploring Robots, *Advanced Robotics*, 6(1992), No.4, 399-430.
- (13) Tanaka K., Zha H. and Hasegawa T., Viewpoint Planning in Map Updating Task for Improving Utility of a Map, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, (2003), 729-734.
- (14) Tanaka K., Tokushima S., Kimuro Y., Zha H. and Hasegawa T., Map Generation and Renewing for a Mobile Office Robot, *Proc. 32nd International Symposium on Robotics*, (2001).
- (15) Thrun S., Robotic Mapping: A Survey, (2002), CMU-CS-02-111.
- (16) Tovey C. and Koenig S., Improved Analysis of Greedy Mapping, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, (2003), 3251-3257.
- (17) Zelinsky A., A Mobile Robot Exploration Algorithm, *Trans. IEEE Robotics and Automation*, 8(1992), No.6, 707-717.