

格子点配置を用いた自律移動ロボットによる環境掃引経路計画[†]

深澤 佑介*・Trevai CHOMCHANA*・太田 順*
湯浅 秀男**・新井 民夫*・浅間 一***

Region Exploration Path Planning for a Mobile Robot Expressing Working Area By Grid Points[†]

Yusuke FUKAZAWA*, Trevai CHOMCHANA*, Jun OTA*,
Hideo YUASA**, Tamio ARAI* and Hajime ASAMA***

In this paper, region exploration path planning algorithm is proposed. In order for a mobile robot to perform this task, appropriate method to deal with the shape of the working environment, which may be intricate or curved, is necessary. In addition, a robot must be able to re-plan the exploration path as soon as possible when robot encounters unexpected objects. These challenges can be met by approximately expressing the working environment with grid points and regenerating the path using one that was planned beforehand. Simulations are used to demonstrate proposed exploration path planning and re-planning algorithm.

Key Words: region exploration, grid point, mobile robot, path planning

1. 序 論

自律移動ロボットにとって特定の領域を掃引する作業は、非常に重要である。領域内の未知物体の探索、ビルの監視、床面の清掃・塗装などさまざまなアプリケーションを有する。現実世界ではこれらの作業は、曲線など複雑な形状をもつ障害物が存在する環境下で行なわれることが多く、そのような障害物への対応が必要になる。また事前に把握していない未知の障害物に遭遇することによって、掃引作業遂行中にロボットが認識する地図情報が変化してしまうため、計算量、探査効率性の観点から有効な再計画が必要になる。

ロボットのセンシング範囲のようにある一定の領域により作業環境全体を覆い尽くす作業は一般的に領域掃引作業と呼ばれる。領域掃引作業におけるロボットの動作手法は、あらかじめ持っている要素行動から状況に応じて適切な行動

を選択する反射行動型と作業全体を見据えて一連の行動手順を決定する計画行動型の二つに大別される。反射行動型は局所的に適切な動作を選択するが、環境内をくまなく掃引する保証がされないという欠点をもつ。計画行動型は、作業前に経路計画を行ないそれに基づき行動をおこし（以下、オフラインでの計画行動）、未知の障害物に遭遇した場合には再び計画を行ない行動をおこす（以下、オンラインでの計画行動）。後者の設計指針に基づきさまざまな研究が行なわれてきた。

計画行動型の基本手法としては Contour Parallel（沿輪郭）型と Direction Parallel（蛇行）型が挙げられる³⁾。これらは、NC 工作機械や塗装ロボットで一般に用いられる経路形態である。Direction Parallel 型を用いた研究として Cao ら¹⁾、Lumelsky ら²⁾ は単体ロボットに対し、障害物の存在する領域での芝刈り作業を想定した掃引アルゴリズムを提案している。この手法では、ロボットがジグザグに往復する経路をオフラインで計画する。未知の障害物に遭遇した場合には、その周囲をもとの計画経路と重なる点まで回る経路をオンラインで計画する。しかし、再計画された経路が冗長である、オフラインの計画が正方領域のみを対象としており凹形状に対応していないなどといった問題点がある。

倉林ら⁴⁾ は、Contour Parallel 型を用いた掃引経路生成手法を提案している。具体的には、障害物同士からの等距離線（ポロノイ線）と、ロボットのセンシング範囲の大きさだけ障害物からオフセットした線（オフセット線）を生成しそれを結ぶ最短経路を求めることでオフラインでの掃引経路を計画している。未知の障害物に遭遇した場合には、今まで通

[†] 第 20 回ロボット学会学術講演会で一部発表 (2002・10)

* 東京大学大学院工学系研究科 東京都文京区本郷 7-3-1

** 東京大学大学院工学系研究科 東京都文京区本郷/理化学研究所バイオ・ミメティックコントロール研究センター（併任）名古屋市守山区大字下志段味字穴ヶ洞

*** 東京大学人工物工学研究センター 東京都目黒区駒場 4-6-1

* The University of Tokyo, Bunkyo-ku, Tokyo

** Graduate School of Engineering, The University of Tokyo, Hongo, Bunkyo-ku, Tokyo / Bio-mimetic Control Research Center, The Institute of Physical and Chemical Research, Anagahora, Shimo-shidami, Moriyama-ku, Nagoya

*** RACE, Research into Artifacts, Center for Engineering, The University of Tokyo, Meguro-ku, Tokyo

(Received January 6, 2003)

(Revised August 11, 2003)

過していない経路を連結する巡回経路を求めることでオンラインでの経路再計画を行なっている。しかし、曲線形状、凹形状など複雑な形状を有する環境下では、ポロノイ線、オフセット線の総数が増加する。また線分を結ぶ最短経路を求める計算量が線分の総数に対して指数関数的に増加するという問題点がある。

以上述べた Contour Parallel 型と Direction Parallel 型の掃引経路生成手法は、線分によって環境を掃引する手法である。一方、点によって環境を被覆する手法（以下、点配置型）が提案されている。市川ら⁵⁾は掃引対象領域にロボットが周囲を観測すべき地点（以下観測点）を等間隔に配置し、それらの点群を結ぶ経路を求めることでオフラインでの掃引経路計画を実現している。未知の障害物に遭遇した場合には、観測点の再配置を行ないそれらを結ぶ経路を求めることにより、オンラインでの計画行動を実現している。しかし、障害物付近での観測点の配置に煩雑な手続きが必要であり、形状の複雑な環境下での観測点の配置が困難となる。

以上従来研究の問題点を整理すると

- (1) 複雑な形状を有する環境で生成されるオフライン計画経路の冗長性
- (2) オンライン計画に要する計算量の増加、生成される経路の冗長性

などといった問題点があり、これらを同時に解決する手法は提案されていない。本稿ではつぎの方法により上記の問題点を同時に解決するアルゴリズムの提案を行なう。掃引経路生成の基本手法としては点配置型を選択する。

現実にはロボットが掃引作業を行なう場合、ロボットが認識すべき物体はある一定以上の大きさを有していると考えてよい。このことを利用し本稿では有限個の点（以下、格子点）により環境の近似を行なう。格子点間の距離を物体の最小の大きさに設定することで掃引対象領域内に存在する物体は完全に観測可能となり掃引作業遂行の保証はなされる。観測点のセンシング範囲によってすべての格子点を被覆しそれらを結ぶ最短経路を求めることによりオフラインでの経路計画を行なう。格子点は、環境の形状が複雑になってもその総数は増加しないという利点を有する。これにより、(1)の問題点を解決する。(2)の問題点は、すでに配置されている観測点の位置情報と、それらを結ぶ経路情報を保存し、再計画の際に利用することで解決する。市川ら⁵⁾は、観測点の位置情報の保存のみを行なっており、より計算量が必要となる後者については言及していない。

以下2章では、問題設定を述べる。3章では、オフラインでの環境掃引経路計画について述べる。4章では、未知の障害物に遭遇時のオンラインでの掃引経路計画について述べる。3,4各章において、シミュレーションにより有効性を検証する。

2. 問題設定

本章では、本研究における仮定を述べる。

作業環境に関する仮定

- ・掃引作業の対象となる領域（以後、掃引作業対象領域 W ）は、2次元の有界閉空間とする。
- ・掃引作業対象領域 W の外形は既知とする。
- ・掃引作業対象領域 W 内には複数個の形状・位置・姿勢が未知で大きさが一定値 d 以下の障害物が存在する。以下、本論文ではこのような障害物を未知障害物と呼ぶ。
- ・掃引作業対象領域 W のうち、障害物を除いた掃引可能な領域を掃引可能領域 W_s とする。

ロボットに関する仮定

- ・ロボットは全方向に移動可能であるとする。
 - ・ロボットは自己位置の同定が可能であるとする。
 - ・ロボットの形状は、半径 R_{robot} の円形とする。
 - ・ロボットのセンシング範囲 S_{sen} は、ロボットの中心から半径 r_s の円形の範囲とする。
 - ・ロボットは、センシング範囲に一部が含まれる未知障害物の形状、位置・姿勢情報を得ることができるとする。
- 上記の仮定のもとでロボットは特定の地点で周囲をセンシングしそのセンシング範囲 S_{sen} の情報を得る。想定する掃引可能領域 W_s 全体をセンシング範囲で覆うように観測点を配置し、それを順番に回ることによって掃引作業を遂行する。

3. オフラインでの環境掃引経路計画

本章では、掃引領域に観測点を配置し、それらを結ぶ経路を生成する手法について述べる。シミュレーションにより、提案手法の有効性を検証する。

3.1 経路計画アルゴリズム

前述のように本手法では、経路計画の基本手法として点配置型を採用する。観測点を配置する手法として市川らは、観測点間の距離が一樣になるよう観測点の移動と追加を行なうことにより観測点の配置を実現している⁵⁾。しかし、障害物付近に観測点を配置する場合、障害物の中には観測点が存在せず観測点間の距離を得ることができないため、観測点の配置が困難であるという欠点をもつ。そこで、本稿では各観測点のセンシング範囲が被覆する面積を近似的に求め、その面積が一樣になるよう観測点の移動と追加を行なうことにより観測点の配置を実現する。障害物は付近の観測点の被覆する面積に影響を及ぼすため、障害物付近の観測点においてもほかの観測点と同じ被覆面積をもつ配置を実現できる。提案手法全体の流れを Fig. 1 に示す。配置する観測点の総数は環境の自由空間に存在する格子点を完全に被覆する最少の数を指す。また最少数の観測点数による配置は、もし領域が無限に広い2次元平面ならば各頂点を観測点とする正三角形を敷き詰めた配置となる (Fig. 2)。本稿では、この配置を観測点の六方最密充填配置と呼ぶ。観測点の六方最密充填配置を目指して、観測点の移動と追加を行なうことによりこれらの格子点を被覆する。また観測点は、格子点上のみを離散的に移動する。以下の順で述べる。

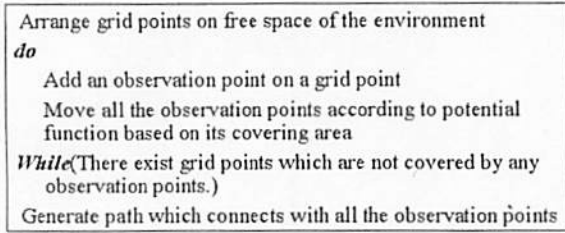


Fig. 1 Flow chart of proposed method

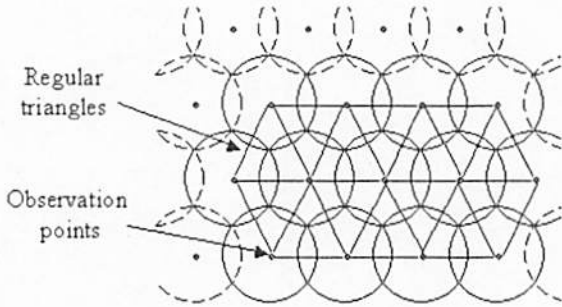


Fig. 2 Hexagonal Closed Packing of observation points

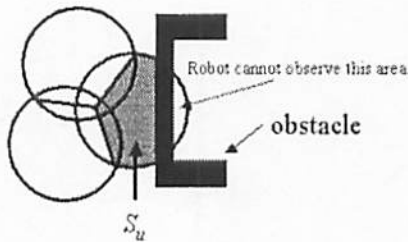


Fig. 3 Area that an observation point covers

- ・ポテンシャル汎関数の設計
- ・観測点の移動と追加
- ・経路生成

3.1.1 ポテンシャル汎関数の設計

観測点の六方最密充填配置を行なった場合、各観測点の被覆する面積は半径 r_s の円に内接する正六角形の面積となる。そこで、各観測点の目標とする被覆面積 S_0 を (1) 式のように設計する。

$$S_0 = \frac{3\sqrt{3}}{2} r_s^2 \tag{1}$$

これを用いて、観測点 u での被覆面積 S_u を評価する関数 $f(u)$ を (2) 式のように設計する。

$$f(u) = |S_u - S_0| \tag{2}$$

このように設計した関数値 $f(u)$ が小さいほど観測点 u の近傍での配置は目標状態に近いといえる。ここで、観測点 u の被覆面積 S_u を定義する。 S_u に含まれる格子点の集合 G を (3) 式を満たす格子点の集合とする。

$$G = \{g | l(g, u) < r_s, l(g, u) < l(g, v), \forall v \neq u\} \tag{3}$$

ここで、 $l(g, u)$ は、格子点 g と観測点 u との間の距離で

ある。ただし Fig. 3 に示すように、観測点のセンシング範囲が障害物をはさんでいる場合、観測不可能な範囲にある格子点を集合 G から除く必要がある。そこで格子点 g と観測点 u との間を結ぶ線分が障害物と交わる場合、その距離 $l(g, u)$ を r_s より大きい値に設定する。また v は、 u との距離が、 $2 \times r_s$ 以下となる観測点である。以下本稿では、 v を u の近傍の観測点と呼ぶ。これらより、 S_u は、 (3) 式を満たす格子点の総数 $n(G)$ と格子点 1 個あたりの面積 d^2 (d は格子点間距離) を用いて、近似的に (4) 式で表わすことができる。

$$S_u \approx n(G) \times d^2 \tag{4}$$

つぎに観測点全体の目標配置を、各観測点での関数値 $f(u)$ が一様に小さい配置とする。“小ささ”の指標として (5) 式を、“一様”であることの指標として、 (6) 式を用いる。観測点全体の配置を評価する汎関数をそれらの和として (7) 式のように設計した。

$$W_0(f) = \sum_{u \in V} f(u)^2 \tag{5}$$

$$W_1(df) = \sum_{u \in V} \sum_{v \in V(u)} |f(v) - f(u)|^2 \tag{6}$$

$$W(f) = W_0(f) + W_1(df) \tag{7}$$

ここで V はすべての観測点の集合であり、 df は、関数 f の余微分である。また $V(u)$ は、観測点 u の近傍の観測点 v の集合である。

3.1.2 観測点の移動と追加

ここでは、観測点の移動方向と移動距離を算出する。観測点 u の移動可能な方向は u から周囲 8 近傍の格子点 $u_k (k = 0, 1, \dots, 7)$ 方向に限定する。周囲 8 近傍の格子点における $f(u_k)$ を求め、 u から $f(u_k)$ が最小となる格子点方向を移動方向とする。 (8) 式を満たす場合は移動を終了する。

$$\min \{f(u_0), f(u_1), \dots, f(u_7)\} > f(u) \tag{8}$$

つぎに観測点の移動距離 $D(u)$ を算出する。 $D(u)$ は、 $f(u)$ の時間変化がポテンシャル汎関数 $W(f)$ の勾配に比例するように決定する ((9) 式)。

$$\begin{aligned} \frac{\partial f}{\partial t}(u) &= -\frac{\delta W(f)}{\delta f}(u) \\ &= -2f(u) - 4 \sum_{v \in V(u)} (f(u) - f(v)) \end{aligned} \tag{9}$$

この (9) 式は、グラフ位相上での反応拡散方程式⁶⁾とみなすことができる。移動距離 $D(u)$ は、 u における移動方向の大きさを用いて (10) 式のように求めることができる。

$$\begin{aligned} D(u) &= \frac{\partial f}{\partial t}(u) \\ &\times \frac{1}{\|f(u) - \min \{f(u_0), f(u_1), \dots, f(u_7)\}\|} \end{aligned} \tag{10}$$

すべての観測点について上記の移動を行なう。つぎに得られた観測点配置により W_s が完全に被覆されているかどうか

かの判定を行なう。どの観測点の S_{sen} にも覆われていない格子点が存在する場合には、すべての観測点から最も離れた格子点に新たに観測点を追加する。追加方法としてはまず各格子点について、すべての観測点との距離を計算し最小値 L_{min} を求める。この計算をすべての格子点について行ない、 L_{min} が最大になる格子点に観測点を追加する。上記の流れにしたがって移動と被覆判定を繰り返す。

3.1.3 経路生成

そのように配置した観測点のロボットが通過すべき順番を決定する。ここで生成する経路は、各都市を一度だけ訪れる最短経路を求める巡回セールスマン問題 (Traveling Salesman Problem: 以下 TSP と略記)⁷⁾ を解くことにより得られる。TSP の解法は、NP 困難 (nondeterministic polynomial-time hard) であることが証明されており¹⁰⁾ 本稿では焼きなまし法 (Simulated Annealing Method: 以下 SA 法と略記)⁷⁾ を用いて TSP を解き準最適な経路の生成を行なう。障害物を回避する経路を生成するため、障害物と重なる経路の経路長を無限大とした。

3.2 提案手法の評価

本節では、計算機シミュレーションにより提案手法の有効性を評価する。計算機のスペックは、Pentium IV 1.8 [GHz] である。ここでは以下の事項を評価する。

- ・観測点の配置に要する計算量
- ・配置に要する観測点の総数
- ・複雑な形状を有する環境下でのオフライン経路計画

3.2.1 観測点の配置に要する計算量

本節では、観測点の配置に要する計算量を議論する。観測点が追加されるたびに、観測点全体が移動する。観測点全体が落ち着くまでの時間は、観測点の挙動が反応拡散方程式に従うことからわかるように、2次元気体が平衡状態にいたるのに要する時間⁹⁾と同じであると考えられる。その時定数は、波数の2乗に比例することが知られており、観測点数を n とすると収束に要する計算量は $O(n^{0.5})$ 以下である。被覆判定に要する計算量は、格子点総数を M とすると $O(M)$ となる。したがって、観測点数 n 個のとき観測点全体が移動し被覆判定を行なう計算量は最大で $O(Mn^{0.5})$ となる。ここでは、観測点数 0 個から N 個まで観測点の追加、移動、そして被覆判定を行なう。したがって $O(Mn^{0.5})$ を観測点数 n に関して 0 から N まで積分を行なうと観測点の配置に要する計算量は最大で $O(MN^{1.5})$ となる。SA 法による TSP の解を得るのに要する計算量は、 $O(N) < O(MN^{1.5})$ であり、経路生成全体の計算量は最大で $O(MN^{1.5})$ となる。

格子点総数、観測点総数は環境の外形状が複雑になっても増加しない。そのため、提案手法の計算量 $O(MN^{1.5})$ は環境の外形状の影響を受けない。一方、Contour Parallel 型、Direction Parallel 型の掃引経路計画の計算量は、環境の外形状を構成する線分総数の影響を受けており、複雑な外形をもつ環境下では、外形を近似する線分の総数が増加し、同時に経路生成の要する計算量も増加してしまう。市川も点

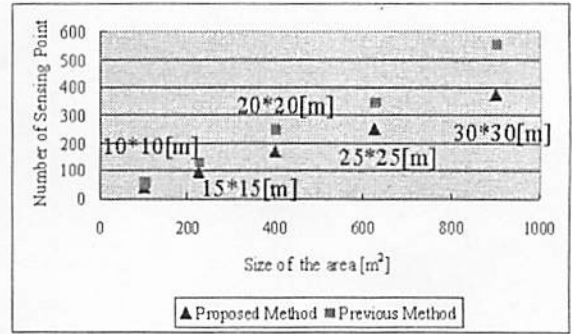


Fig. 4 Comparison of the number of observation points

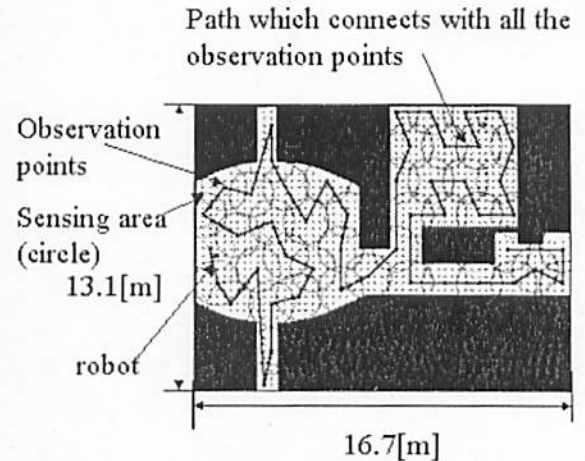


Fig. 5 Off-line planning under corridor and one room environment

配置手法を用いているが、障害物付近の観測点配置に障害物を構成している線分を利用しており外形形状の影響を受けている。以上提案手法の計算量は環境の外形状の影響をうけないことを示した。

3.2.2 配置に要する観測点総数

本節では、配置に要する観測点数の評価を行なう。ここでは、市川⁵⁾の連続系で観測点を配置する手法と比較する。Fig. 4 に環境の面積に対する配置完了までに要する観測点の総数を示す。30 × 30[m] = 900[m²] の環境では、被覆に要する観測点総数は、提案手法では 378 個、比較対照手法では 557 個、その差は 179 個であり、環境が広大になるに従い差が顕著になっている。これは、格子点を被覆することで、近似的に環境を被覆しているためである。

3.2.3 複雑な形状を有する環境下での経路生成

つぎに提案手法を複雑な形状を有する環境へ適用し観測点の配置、経路生成が適切に行なわれているか検証する。ここでは、実環境下を模した環境として以下の環境へ適用する。

- ・曲線形状の廊下と一つの部屋が存在する環境
- ・会議机、複数の椅子、棚が存在する部屋
- ・4つの部屋が存在する環境

それぞれの環境下において格子点間距離を 0.3[m]、セン

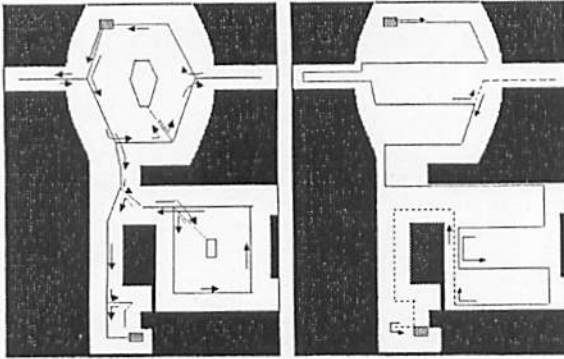


Fig. 6 Comparison of off-line planning: Contour Parallel(left), Direction Parallel(right)

シング範囲を 1.0[m] としてシミュレーションを行なった。観測点と格子点を配置し経路を生成したようすをそれぞれ、Fig. 5, Fig. 7, Fig. 8 に示す。ここで、掃引領域に等間隔に存在する点が格子点であり、それらを被覆している円の中心が観測点である。観測点のセンシング範囲によって掃引対象領域に存在する格子点がすべて被覆されていることを確認していただきたい。

つぎに生成された経路の経路長の評価を行なう。ここではオフラインでの掃引経路生成の基本手法である Contour Parallel 型と Direction Parallel 型との経路長との比較をする。Fig. 5 の環境における Contour Parallel 型の経路を Fig. 6 左図 (Fig. 5 と比べて 90° 回転していることに注意されたい) に、Direction Parallel 型の経路を同右図に示しており、ロボットは生成された経路を矢印の方向にたどる。ただし、Direction Parallel 型では、オフラインでの計画において凹形状の物体に対応していないという問題点を有するため、ここでは、凹領域にそれぞれジグザグ状の走行経路を用意しそれらを結ぶ経路を人為的に加えることで対処した。Direction Parallel 型の人為的に追加した経路を破線で示している。比較のため、ロボットの初期位置とゴール位置は、提案手法のそれと一致させている。シミュレーションの結果 3 つの環境において生成された経路長を Table 1 に示す。提案手法は Contour Parallel 型と比較して、Fig. 5 (Corridor) では 18%、Fig. 7 (Conference Room) では 19% 短くなっている。Direction Parallel 型に対しては Corridor の環境では有意な差は見られなかったが、Conference Room では 13% 短くなっている。これは Contour Parallel 型、Direction Parallel 型では、Corridor、Conference Room においていずれも重複して通過する経路が生成されているが、一方提案手法では、そのような経路は存在しないためである。ただし Direction Parallel 型では、Conference Room においてその重複する部分が短く提案手法と同等の経路長となっている。Fig. 8 (Four Rooms) のように、単純な形状のみで構成されている環境の場合には、三手法の間に有意な差は生じなかった。これらより、環境の外形形状に凹領域が含まれていたり、複雑な形状を有する場合、Contour Parallel 型、

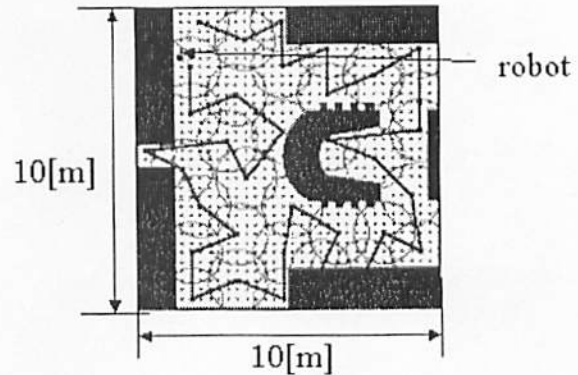


Fig. 7 Off-line planning under conference room environment

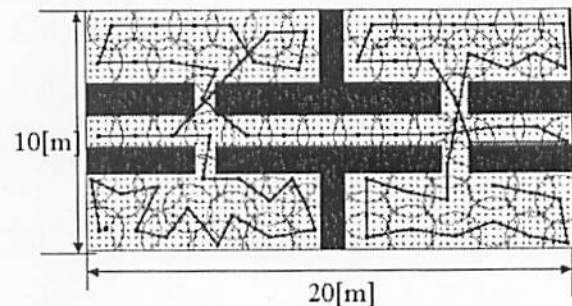


Fig. 8 Off-line planning under four rooms environment

Table 1 Comparison of the off-line path length [m]

	Corridor	Conference Room	Four Rooms
Proposed	86.6	53.6	123.8
Contour Parallel	93.4	66.0	120.1
Direction Parallel	86.2	61.1	124.3

Direction Parallel 型では、経路の冗長性が増すことがわかる。また、これらの環境において市川ら⁵⁾の方法で経路生成を行なう場合、3.1 節で述べたように障害物付近の観測点の配置が困難であることから、ここで対象としている複雑な形状を有する環境下において適切な経路を生成を行なうことはできないと考えられる。以上より提案手法が複雑な外形形状を有する環境下においてオフラインで適切な経路計画が可能であることを示した。

4. オンラインでの環境掃引経路計画

本章では、未知障害物に遭遇した場合のオンラインでの経路再計画手法について述べる。

4.1 経路再計画アルゴリズム

未知障害物発見により、ロボットのもつ地図情報が変化するため、掃引対象領域を再定義する必要がある。新たな掃引対象領域を掃引作業開始時における掃引領域から以下の 2 つの領域を除いたものとする。

- (a) すでに掃引した領域
- (b) 新たに発見された障害物が占める領域

ロボットは、生成された経路にしたがって移動し、観測点に

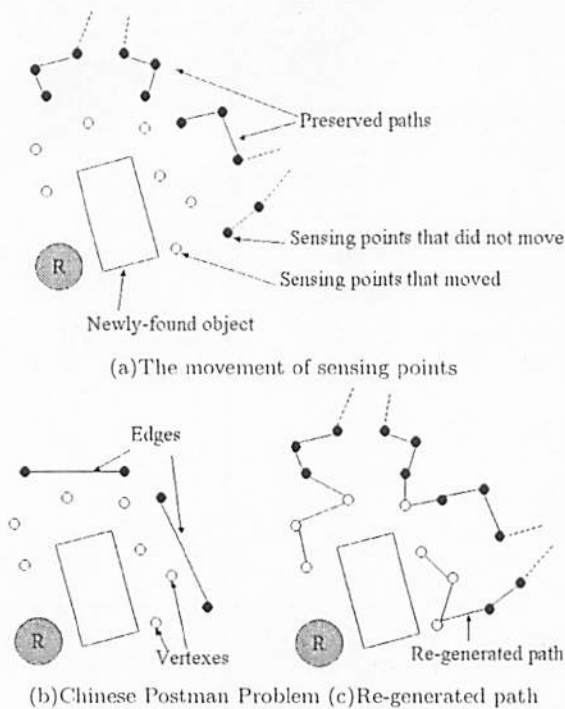


Fig. 9 Flow of re-generation of path

において観測を行なうたびにそのセンシング範囲内の格子点を除去する。これにより、(a)の領域を除く。また、ロボットは2章の仮定より新たに発見した物体の位置、姿勢、形状がわかるので、これらより障害物の内側の格子点を除去する。これにより、(b)の領域を除く。

このように再定義された掃引対象領域に観測点の再配置を行なう。再配置に要する計算量を低減するためすでにある観測点を利用する。観測点は、格子点上のみを離散的に移動し、またその移動は、反応拡散方程式にしたがっているため、掃引対象領域の変動の起きた(a),(b)の境界付近の観測点のみが移動し、ほかの観測点は移動しない。(Fig. 9(a))。そのような移動のなかった観測点群に対してTSPを再び解き経路を求めた場合、再計画前と同じ観測点どうしを結ぶと考えると差し支えない。そのためすでに生成されている経路情報を保存し利用することで経路生成に要する計算量を低減可能であると思われる。以下提案手法を述べる。

まず、移動の生じた観測点と生じなかった観測点を分け、移動の生じた観測点につながっている経路成分を切り離す(Fig. 9(a))。保存された経路成分の両端の2点がいくつかの観測点を経由して結ばれループを生成してしまうことを防ぐため、両端を結び仮想の辺とみなす(Fig. 9(b))。つぎに、これらの辺と、移動の生じた観測点をすべて通過する最短経路を求める。このような問題は、一般化中国人郵便配達夫問題(Rural Chinese Postman Problem, RPP)⁸⁾として定式化されている。この問題は、NP困難であることが証明されており¹⁰⁾、本研究では実時間で準最適解を求めることにする。形成すべき経路は各観測点をたかだか一回通過するも

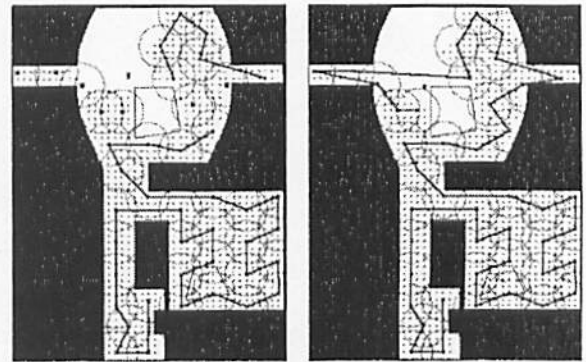


Fig. 10 Re-generation of the path

Table 2 Comparison of the computation time and path length of re-generation of new path

	Average Computation Time [s]	Average Path Length [m]
Proposed Method	7.3	86.9
Compared Method	73.0	88.0

のとする。Fig. 9(b)における辺の重み(距離)を0として辺の両端の観測点を結ぶことによりTSPへと帰着させることができる。ここではSA法を応用することによりTSPの近似解を求める。SA法によって解を求めた後、保存された経路を戻す(Fig. 9(c))。ここでSA法の計算に要する観測点数は、環境を被覆している観測点の総数と比べて少ないので、SA法に要する計算量は必ず低減される。

4.2 計算機シミュレーション

Fig. 10左図に未知障害物(黒枠の障害物)に遭遇し、観測点の再配置を行なったようすを、同右図に再生成された経路を示す。すでに観測した領域の格子点と障害物内の格子点が除去されていることを確認していただきたい。ここで、移動のあった観測点を小正方形で、移動のない観測点で構成される掃引作業開始時の経路を線分で示している。

1個の障害物の位置をランダムに配置した10種類の環境下において、再計画後の経路長、再計画に要する計算時間を計測した。ここでは、経路を保存せずに経路生成を行なう手法(文献⁵⁾で採用)と比較することで、経路を保存することによる計算量の低減効果、生成された経路の悪化の有無を評価する。いずれの手法においても観測点の再配置に要する計算時間は再計画直前の観測点の情報を利用しているため3秒程度で完了している。提案手法と、比較対照手法の経路生成に要する計算時間の平均値と、生成された経路の経路長をTable 2に示す。提案手法では経路を保存しない手法に対して1/10の計算時間で経路を生成している。これは、移動のない観測点で構成される経路を保存することで、SA法によってTSPを解く観測点数を低減しているためである。提案手法の経路長と、経路を保存しない比較対照手法の経路長は同程度であり経路を保存することによる経路の質の悪化はないといえる。また、序論でも述べたようにDirection Parallel型では障害物の周囲を周回する冗長な経路が生成さ

れる. Contour Parallel 型では経路を求める計算量が線分の総数に対して指数関数的に増加するという問題点がある. 以上, 提案手法が未知障害物に遭遇した場合, 計算量を抑えてかつ準最適な経路を導出可能であることを示した.

なお本研究では, 環境を格子点により近似しているため格子点間距離 d 以下の物体を無視した経路生成を行なう可能性がある. 経路上に d 以下の物体が存在し走行の障害となる場合には, 局所的なセンサ (超音波センサなど) を用いて, 一時的に障害物を回避することで対処可能であると考えられる.

5. 結 論

本稿では, 格子点配置を用いた自律移動ロボットによる環境掃引経路計画手法の提案を行なった. 環境を離散化することにより曲線などの一般的な形状を持つ障害物の存在する環境下においてオフラインでの環境掃引経路計画を行なうことができた. また未知障害物に遭遇した場合, すでにある経路を有効に利用することで計算量を抑えてかつ準最適な経路を導出することができた. 今後の展望としては, 実環境への対応, 群ロボット系への拡張が挙げられる.

参 考 文 献

- 1) Z. L. Cao, Y. Huang, E. L. Hall: Region Filling Operations with Random Obstacle Avoidance for Mobile Robots, *Journal of Robotic Systems*, 5-2, 87/102 (1988)
- 2) V. Lumelsky, S. Mukhopadhyay and K. Sun: Sensor-Based Terrain Acquisition - a Seed Spreader Strategy, *IEEE/RSJ Int. Workshop on Intelligent Robots and System*, 62/67 (1989)
- 3) M. Held: On the Computational Geometry of Pocket Machining, *Lecture Notes in Computer Science* 500, Springer-Verlag (1991)
- 4) 倉林, 太田, 新井, 吉田: 掃引作業における移動ロボット群の動作計画, *日本ロボット学会誌*, 16-2, 181/188 (1998)
- 5) 市川, Trevai, 太田, 湯浅, 深澤, 新井, 浅間: グラフ上の反応拡散方程式を用いた環境探索経路計画, *計測自動制御学会論文集*, 38-12, 996/1002 (2002)
- 6) 湯浅, 伊藤: グラフ上の反応拡散方程式と自律分散システム, *計測自動制御学会論文集*, 35-11, 1447/1453 (1999)
- 7) E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan and D. B. Shmoys: *The Traveling Salesman Problem - A Guided Tour of Combinatorial Optimization*, A Wiley-Interscience Publication (1985)
- 8) M. Kwan: Graphic Programming using ODD or EVEN Points, *Chinese Math*, 1, 273/277 (1962)
- 9) L. E. Reichl, 鈴木増雄 監訳: 現代統計物理 (上), 丸善, 158/160 (1983)
- 10) J. K. Lenstra and A. H. G. Rinnooy Kan.: On general routing problems. *Networks*, 6, 273/280 (1976)

[著 者 紹 介]

深 澤 佑 介



2002年東京大学工学部精密機械工学科卒業. 同年東京大学大学院工学系研究科修士課程入学. 現在に至る. ロボットの学習機構に興味を持つ.

Trevai CHOMCHANA



2001年電気通信大学電気通信学部機械制御工学科卒業. 2003年東京大学大学院工学系研究科修士課程修了. 同年同博士課程入学. 現在に至る. 群ロボットの研究に興味を持つ.

太 田 順 (正会員)



1989年東京大学大学院工学系研究科修士課程修了. 同年新日本製鐵(株)入社. 91年東京大学工学部助手. 94年同講師. 96年より東京大学工学系研究科助教授. この間, 96年~97年スタンフォード大学客員研究員. 群知能ロボット等の研究に従事. 博士(工学).

故 湯 浅 秀 男 (正会員)



1986年名古屋大学院博士前期課程修了. 助手を経て, 92年名古屋大学工学部講師. 93年理化学研究所フロンティア研究員を併任. 99年東京大学工学系研究科助教授. 2002年9月18日没. システム理論, ロボット工学の研究に従事. 博士(工学).

新 井 民 夫 (正会員)



1970年東京大学工学部精密機械工学科卒業. 77年同博士課程修了. 工学博士. 87年東京大学工学部精密機械工学科教授. 2000年東京大学人工物工学研究センターセンター長(併任). 自動組立, 複数移動ロボットなどの研究に従事. IEEE, CIRPなどの正会員.

浅 間 一 (正会員)



1984年東京大学大学院工学系研究科修士課程修了. 86年より理化学研究所研究員補. 同所工学基盤研究部副主任研究員などを経て, 2002年より東京大学人工物工学研究センター教授. 自律分散型ロボットシステムにおける協調・適応技術, 知的データキャリアとその応用技術の開発等に従事. 工学博士.