センサベースナビゲーションにおいて移動ロボットが 目的地付近まで到達できる誤差の条件†

登尾 啓史*・吉岡 孝**

On the Error Condition that a Mobile Robot Arrives at a Neighborhood of Its Destination in Sensor-Based Navigation

Hiroshi NOBORIO* and Takashi YOSHIOKA**

A mobile robot usually estimates its position and orientation by a dead reckoning system. However under several kinds of uncertainties, e.g. inner/outer sensor errors and/or a slip, a mobile robot unfortunately misunderstands its true position/orientation in a real world as a virtual position/orientation in a computer world. Due to the difference, a mobile robot governed by sensor-based navigation sometimes gets lost in an unknown 2-D environment. On the observation, in this paper, we firstly improve previous metric sensor-based navigation algorithms in order for a mobile robot to deal with such uncertainties. Secondly, we name a class of errors homeomorphic error, where true and virtual positions in real and computer worlds are projected each other by the homeomorphism. Then, we theoretically show that improved metric algorithms never lead a mobile robot with the homeomorphic error to any deadlock (periodic cycle) in an uncertain 2-D world. Finally, we describe simulation results that improved algorithms guide a mobile robot with the homeomorphic error to a neighborhood of its destination surely.

Key Words : autonomous mobile robot, stable navigation, gross sensor-based navigation, position and orientation errors

1. はじめに

移動ロボットを初期地から目的地まで誘導する問題は、“ナビゲーション”として精力的に研究されてきた。ナビゲーションでは、初期地から目的地へ至る経路が存在すれば、移動ロボットは決して無限ループに陥らず目的地へ到達し、そうでなければ移動ロボットが目的地へ到達できないことを認知する機能は重要である。われわれは、この機能を“デッドロックフリー特性”と呼ぶ。一般に、ナビゲーションアルゴリズムには、モデルベースアルゴリズムとセンサベースアルゴリズムがある。モデルベースナビゲーションでは、まず、既知空間—障害物の位置や形状が明らかな空間—をモデル(地図)としてコンピュータへ登録し、次に、A*法などの最適探索法でそれから最適パスを選択し、最後に、移動ロボットはそのパス沿いに走行し目的地へ到達する¹⁾。しかし、この様なオフラインナビゲーションでは、移動ロボットが位置や姿勢を誤認識したり、予期せぬ障害物が置かれたりする

と、本来生じないはずの衝突が起こりロボットは航行不能となる。この結果、アルゴリズムのデッドロックフリー特性は損なわれる。一方、センサベースナビゲーションでは、移動ロボットは未知空間—障害物の位置や形状が明らかでない空間—を行動する。このとき、移動ロボットは、外界センサ(超音波センサ・光センサなど)からの局所情報のみを頼りに障害物を回避し目的地をめざす^{2)~9)}。したがって、環境の地図がなくても(もちろんそれが現実の環境と違っていても)問題はない。また、位置や姿勢の誤差が生じて、移動ロボットはオンラインで行動するので直ちに航行不能となることはないけれど、移動ロボットが目的地の近傍まで到達できるかどうかは保証の限りでない。

一般に、位置誤差が存在すると、移動ロボットは決して目的地に到達できない。そこで本稿では、つぎの様な前提のもと、移動ロボットが目的地の近傍へ到達するには、位置誤差がどのようなクラスに属すればよいかについて考察する。

- (a)位置誤差は連続的に生まれる。
- (b)目的地のまわりに広い円形状の自由領域 F_G が取れる。
- (c)移動ロボットが無限ループに陥らない限り、位置誤差は最後には自由領域 F_G の半径以下に抑えられる。

本稿では、これらの前提条件のもとセンサベースナビゲーションを考察し、

- (1)正常終了条件：初期地から目的地に至る経路—以後デッドロックフリーパス DFP と記す—が存在するとき、移動

† 1995 IEEE Int. Conf. on SMC で一部発表(1995・10)

* 大阪電気通信大学情報工学科 寝屋川市初町 18-8

** Department of Engineering Informatics,

Osaka Electro-Communication University, Neyagawa

** 大阪電気通信大学大学院工学研究科 寝屋川市初町 18-8

**Graduate School of Engineering, Osaka Electro-

Communication University, Neyagawa

(Received June 24, 1996)

(Revised March 5, 1998)

ロボットは無限ループに陥らず自由領域 F_G (目的地近傍)で停止する。

(2)異常終了条件:デッドロックフリーパス DFP が存在しないとき,移動ロボットは必ず無限ループに陥り DFP が存在しないことを認知する。

という2つの条件が維持される位置誤差のクラスを特定する。

さて,従来のセンサベースナビゲーションアルゴリズムは,つぎの条件の組み合わせで“正常終了条件”を保証する¹⁰⁾。

(1)物理条件:目的地へ直進できるかどうかを判別する条件,
(2)距離条件:障害物から離れるところを目的地 G へ近づける条件,(3)線分条件:線分 SG 上にいることを判別する条件(S :初期地, G :目的地),(4)復帰条件:過去に訪れたところを判別しループを認識する条件,(5)位相条件:線分 SG とそれが交差する障害物の周囲が成す位相学的性質(連結性など)にもとづく条件,(6)幾何条件:障害物の幾何学的性質(接線の変化など)を利用する条件であり,物理条件を内包する。ここで“正常終了条件”に注目すると,古典的なアルゴリズム $Bug2$ は条件(1),(2),(3)²⁾,アルゴリズム $Class1 \sim 3$ は条件(1)と(2)で構成されている^{3), 4)}。さらに, $Bug2$ に条件(4)を加えるとアルゴリズム $Alg1$ ⁵⁾, $Class1$ に条件(4)を加えるとアルゴリズム $Alg2$ が誕生する⁶⁾。これらのアルゴリズムのように,条件(2)を含むものは“距離アルゴリズム”と分類される。一方,アルゴリズム $Curv1$ は条件(1),(3),(5)からなり,条件(5)を含むので“位相アルゴリズム”と分類される⁷⁾。最後に,アルゴリズム $GeoI \& II$ は条件(4)と(6)⁸⁾,アルゴリズム $Robust$ は条件(3)と(6)からなり⁹⁾,条件(6)を含むので“幾何アルゴリズム”と分類される。

一般に,位置や姿勢の誤差はこれらの条件に強く影響し,アルゴリズムのグロスデッドロックフリー特性を台無しにすることがある。これは主として,離れてはならないところから誤って障害物を離れたり,離れなければならないところで誤って障害物から離れなかったりし,無限ループ(デッドロック)に陥ることによる。本稿では,位置や姿勢の誤差を処理できるように従来の距離アルゴリズムを修正し,それがグロスデッドロックフリー特性を維持できる誤差のクラスを特定する。

第2章では,まず,移動ロボットやその行動空間を定義したのち,位置や姿勢の誤差の影響で,真の位置・姿勢と計測の位置・姿勢がずれることを説明する。次に,真の位置と計測位置が一对一かつ連続的に写像される誤差を“同相誤差”と定義する。第3章では,それらの誤差を処理できるよう,従来の距離アルゴリズムを修正する。そして,第4章では,修正距離アルゴリズムがグロスデッドロックフリー特性を維持するには,移動ロボットが同相誤差しか生まないことが必要十分条件であることを証明する。第5章では,同相誤差を仮定したとき,修正距離アルゴリズムのグロスデッドロックフリー特性をシミュレーションで確かめる。最後に,第6章で,本研究をまとめたのち,同様に修正した位相・幾何アルゴリズムのグロス

デッドロックフリー特性について言及する。

2. 移動ロボットとその行動空間

この章では,まず,移動ロボットと行動空間を定義する。この空間は,有限でも無限でも良いが2次元とし,そこには多数の未知障害物が存在するものとする。このとき,各々の障害物の形状は自由でよいが,その周囲長や個数は有限とする。

次に,移動ロボットは点で表現され,全方向へ自由に移動できるものとする(Fig.1)。内界センサ(エンコーダやジャイロコンパスなど)がサンプリングタイム毎に計測した走行距離や走行方向を積分すると,移動ロボットは自己の位置 R をオンラインで推測できる。これはデッドレコニング法(Dead Reckoning System (DRS))により達成できる。一般に,エンコーダやジャイロコンパスの計測誤差,積分の離散誤差,タイヤ直径の差,タイヤの同期誤差,タイヤと床のスリップなどが原因で,真の位置 R^* は偽の位置 R として認識される。

本研究では, R^* をロボットの体,それが走行するところを実世界(Real World)と呼ぶ。一方, R をロボットの頭,それが行動するところをコンピュータ世界(Computer World)と呼ぶ。ここで,どちらの世界も同じ絶対座標系で表現される。一般に, R と R^* の差は“位置誤差”と呼ばれる。これより,コンピュータ世界の目的地ベクトル $V_G(=RG)$ は,実世界の目的地ベクトル $V_G'(R^*$ を起点とするベクトル V_G)に対応する。このとき,真の目的地ベクトル R^*G と偽の目的地ベクトル V_G' は,最大で角度 α だけずれるものとする(Fig.1(b))。

Fig.1 (a) A robot body R^* makes a real path P^* faithfully around a real obstacle O^* in a real world (RW). (b) A robot head R generates a virtual path P around a virtual obstacle O using a dead reckoning system (DRS) in a computer world (CW).

さらに,頭 R はコンピュータ世界を V_S 方向へ進み,体 R^* は同期的に実世界を V_S^* 方向へ進む。一般に,この差は“姿勢誤差”と呼ばれる。前述のデッドレコニング法では,走行方向の変化のみ積分して姿勢ベクトル V_S を得るが,それは真の姿勢ベクトル V_S^* と異なり,最大で角度 β だけずれるものとする。ここで,定数 α は走行距離・方向の誤差から決定され,定数 β は走行方向の誤差のみから決定される。一般に,上限値 α と β は強い相関を持つが完全には従属しない。

さて,センサベースナビゲーションでは,ステアリングベ

クトル V_S^* が障害物に妨げられなければ、体 R^* はその方向へ進む。そうでなければ、外界センサ(超音波センサや光センサなど)からの情報をフィードバックし、体 R^* は障害物の周囲を時計回り、または、反時計回りに辿る。ここで、障害物の周囲を追跡したり¹¹⁾、障害物までの距離を複数計測し、その法線ベクトル V_N^* (すなわち接線ベクトル V_T^*) を精度良く算出する研究も精力的になされている¹²⁾。これより、本稿では、体 R^* は障害物の正しい接線ベクトル V_T^* を正確に追えるものとする。このとき、前述の姿勢誤差より、実世界の接線ベクトル V_T^* ($=V_S^*$) はコンピュータ世界の接線ベクトル $V_T (=V_S)$ に移され、コンピュータ世界の目的地ベクトル $V_G (=V_S)$ は実世界の目的地ベクトル $V_G^* (=V_S^*)$ に移される。

一般に、位置の誤差を仮定すると、体 R^* が真の目的地 G に到達できないのは明らかで、ここではそれが無限ループに陥らず目的地の近傍 G^* まで到達できるかどうかについて議論する(偽の目的地 G^* は追って定義する)。

Fig.2 (a)~(d) Four kinds of a true destination G and its virtual destination G^* around an unknown obstacle O^* .

定義1 (グロスデッドロックフリー特性) 正常終了条件: 未知空間にデッドロックフリーパス DFP が存在するなら、体 R^* は決して無限ループに陥らず目的地 G の近傍 G^* へ到達する (Fig.2(a)). 異常終了条件: 未知空間にデッドロックフリーパス DFP が存在しないなら、頭 R は必ず無限ループに陥りそのことを認知できるものとする (Fig.2(d)). ○

“異常終了条件”を維持するには、真の位置 R^* が偽の位置 R へ一対一に写像されることが必要かつ十分である ($R=f(R^*)$). このとき、誤差の連続性から、写像 f は同相写像となり、位置 R から位置 R^* への逆写像 f^{-1} も同相写像 (一対一かつ連続) となる ($R^*=f^{-1}(R)$) (Fig.3). 本稿では、このような写像を引き起こす誤差を同相誤差と呼ぶ。この誤差は、ロボット R がループ L (ロボット R^* がループ L^*) を一周するときに生じる位置誤差の合計がゼロとなるものである。これには、アフィン変換のよ

Fig.3 True and false positions are projected each other by a homeomorphic error.

うな線形誤差 (Linear error), また時系列でお互いに打ち消しあう非線形誤差 (Non-linear error) などが含まれる。最後に、同相誤差と真の目的地 G から偽の目的地 G^* は一意に決定され ($G^*=f^{-1}(G)$), 同相誤差と線分 RG から曲線分 R^*G^* は一意に決定される ($R^*G^*=f^{-1}(RG)$).

前述のように、“異常終了条件”を維持する必要十分条件は、移動ロボットが同相誤差しか生まないことであり、これは自明である。しかし、“正常終了条件”を維持する必要十分条件も、移動ロボットが同相誤差しか生まないことである。これは決して自明でなく本稿で証明する。

さて、真の目的地 G と偽の目的地 G^* が実世界の障害物で分断されると、つぎの様な誤解が生じる。

定義2 未知空間では、初期地 S と目的地 G , そして、初期地 S と目的地 G^* はつぎの4つのパターンを形成する (Fig. 2). (1) 初期地 S と目的地 G , かつ、初期地 S と目的地 G^* の間に DFP が存在する (Fig.2(a)). (2) 初期地 S と目的地 G の間に DFP は存在するが、初期地 S と目的地 G^* の間に DFP は存在しない (Fig. 2 (b)). (3) 初期地 S と目的地 G^* の間に DFP は存在するが、初期地 S と目的地 G の間に DFP は存在しない (Fig.2(c)). (4) 初期地 S と目的地 G , かつ、初期地 S と目的地 G^* の間に DFP は存在しない (Fig.2(d)). ○

本研究では、目的地 G のまわりに自由領域 FG を仮定し、パターン (2) や (3) が生じないようにしている。このとき、偽の目的地 G^* は必ず自由領域 FG の中に存在するものとする。この前提のもと、改良アルゴリズムはグロスデッドロックフリー特性を得る。

3. 修正距離センサベースト

ナビゲーションアルゴリズム

この章では、位置と姿勢の誤差を同時に考慮できるよう、従来の物理条件『 V_G 方向が障害物に妨げられなければ、移動ロボットは V_G を V_S として行動する (Fig.4(a)の○表示)。そうでなければ、妨げられた障害物の V_T —回避方向のもの—を V_S として行動する (Fig.4(a)の×表示)』をつぎのように置き換える。

定義3 移動ロボットが障害物を時計回り [反時計回り] に回避するとき、外積 $V_G \times V_T$ が負またはゼロ [正] ならべ

Fig.4 (a) A previous physical condition in a computer world (CW). (b) An improved physical condition in a computer world (CW).

クトル V_G をベクトル V_S , 正 [負またはゼロ] ならベクトル V_T をベクトル V_S として行動する (Fig.4(b)). ○

新しい物理条件は, 誤差の影響を受けるベクトル V_G と V_T を陽に関係づける.

さて, 修正距離アルゴリズムの概略を説明する. まず, 頭 R は初期地 $S (=L_0)$ を出発したのち, ベクトル V_G を V_S としてその方向へ走行し, それと同期して体 R^* は V_S^* 方向へ走行する. 次に, 体 R^* が衝突点 H_i^* (最近点 C^*) で障害物に出会ったら, 頭 R は現在地を衝突点 H_i (最近点 C) とする. そして, 体 R^* は障害物の接線ベクトル V_T^* を V_S^* としてその方向へ走行し, 頭 R は同期的に接線ベクトル V_T を V_S としてその方向へ走行する. ここで, *Class1* のみ, 体 R^* が障害物を回避中に頭 R が点 C よりも目的地 G へ近づけば, そこを最近点 C とする. そして, *Class1 ~ 3* では, 現在地が各々点 C, H_i, L_{i-1} よりも目的地 G へ接近し, かつ, 外積 $V_G \times V_T$ が負またはゼロであれば, 頭 R はそこを離脱点 L_i とする. そして, 頭 R は再び V_G を V_S としてその方向へ走行し, 体 R^* は同期的に V_S^* 方向へ走行する. ここで, “距離条件” は, ユークリッド距離 $|CG|$, $|HG|$, または $|LG|$ を単調に減少させる (たとえば, 不等式 $|L_i G| > |L_{i-1} G|$ が成立する) ことで定義され, これより頭 R は確実に目的地 G へ到達する. 一方, “物理条件” は外積 $V_G \times V_T$ でチェックされる. 次に, *Class2* に “線分条件”, すなわち, 頭 R は線分 SG 上からしか障害物を離脱できないという条件を加えるとアルゴリズム *Bug2* が誕生する.

もし, 位置や姿勢の誤差がなければ ($R=R^*$, $V_G=V_G^*$, $V_S=V_S^*$), 3つの条件はコンピュータ世界 (CW) と実世界 (RW) で完全に一致し, 体 R^* は目的地 G へ到達する. そうでなければ, 線分 RG と R^*G^* , 距離 $|RG|$ と $|R^*G^*|$ ($R=R^*=S$, $R=C$, $R=H_i$, $R=L_i$ など) はコンピュータ世界 (CW) と実世界 (RW) で異なる. したがって, コンピュータ世界と実世界で頭 R と体 R^* が3つの条件 (たとえば, 距離条件では, $|L_i G| > |L_{i-1} G| \Leftrightarrow |L_i G^*| > |L_{i-1} G^*|$) を同時に満足させるところが必ず存在することを示さなければならない. 本稿では, それらが両立する必要十分条件は, 移動ロボットが同相誤差しか生まないことであることを証明する (第4章の定理3).

また, アルゴリズム *Class1* に “復帰条件”, すなわち, 『過去の離脱点や衝突点に戻ったら, (1) 時計回りと反時計回りのルートのうち短い方で直前の衝突点まで戻り, (2) 予め選択されたのと逆方向に障害物を回避し, (3) 障害物から離れたら回避方向を元に戻す』を付け加えると *Alg2* が誕生する. 最後に, アルゴリズム *Class1* に “線分条件” と “復帰条件” を付け加えると *Alg1* が誕生する. 本稿では, “復帰条件” を満足する必要十分条件が, 移動ロボットが同相誤差しか生まないことであることをあわせて証明する (第4章の定理2).

3. 1 修正アルゴリズム *Class1* ~ 3**

この節では, 前述の誤差を取り扱えるよう, 従来のアルゴリズム *Class1 ~ 3* を自然な形で拡張する (Fig.5(a)~(c)).

0. 初期地 S を離脱点 L_0 , $i=1$ として, 1. へ進む.

1. つぎの事象が起こるまで, 頭 R は離脱点 L_{i-1} からベクトル V_G を V_S としてその方向へ進み, 体 R^* は同期的に V_S^* 方向へ進む.

(a) 頭 R が真の目的地 G に到達したら, 体 R^* は偽の目的地 G^* に到達し, アルゴリズムは正常終了する (Fig.2 (a)).

Fig.5 (a), (b), (c) Several robust algorithms *Class1* ~ 3** for a mobile robot with position and orientation errors, which are revised from previous algorithms *Class1 ~ 3*.

(b) 体 R^* が障害物に衝突したら、頭 R の位置を衝突点 H_i 、ならびに、最近点 C として、2.へ進む。

2. つぎの事象が起こるまで、体 R^* は障害物の周囲で V_T^* —時計回り [反時計回り] のもの—を V_S^* としてその方向に辿る。そして、それに同期して、頭 R は V_T を V_S としてその方向に走行する。ここで、 $Class1^*$ のみ、頭 R が点 C より目的地 G に接近したら、そこを最近点 C として記憶する。

(a) 頭 R が真の目的地 G に到達したら、体 R^* は偽の目的地 G^* に到達し、アルゴリズムは正常終了する (Fig.2 (a)).

(b) $Class1^* \sim 3^*$: 頭 R が、各々点 C , 点 H_i , 点 L_{i-1} よりも目的地 G へ接近し、かつ、外積 $V_G \times V_T$ が負またはゼロ [正] のところを離脱点 L_i とし、 $i=i+1$ として1.へ戻る。

(c) 体 R^* が障害物を離脱できず、頭 R が衝突点 H_i に戻ったら、初期地 S と目的地 G を結ぶパスは存在しないので、体 R^* は停止しアルゴリズムは異常終了する (Fig.2(d)).

ここで、アルゴリズム $Class1^*$ を Fig.5(a) で説明する。ここでは話をわかりやすくするため、 V_S (V_G または V_T) に対して V_S^* を常に右へ 7° 誤解した線形誤差を考える。まず、頭 R は初期地 S を出発し、コンピュータ世界を V_S 方向へ進む。これに同期して、体 R^* は実世界を V_S^* 方向へ進む。そして、体 R^* が実世界で障害物に衝突したら、頭 R は現在地を衝突点 H_1 として、体 R^* はその障害物を辿る。そして、距離条件: 頭 R が衝突点 H_1 よりも目的地 G に接近し、かつ、物理条件: 外積 $V_G \times V_T$ が負またはゼロであれば、頭 R は現在地を離脱点 L_1 とする。そして、頭 R はコンピュータ世界を V_S 方向へ進み、体 R^* は同期的に実世界を V_S^* 方向へ進む。そして、体 R^* が再び障害物に衝突すれば、頭 R は現在地を衝突点 H_2 として、体 R^* はその障害物を辿る。そして、距離条件と物理条件が満足される離脱点 L_2 から、頭 R は再び V_S 方向へ進み、体 R^* は同期的に V_S^* 方向へ進む。さらに、体 R^* が障害物に衝突したら、それを辿りながら同様の処理で離脱点 L_3 を発見し、頭 R は V_S 方向、そして体 R^* は V_S^* 方向へ走行する。最後に、頭 R が目的地 G に到達したら、アルゴリズムは終了する。このとき、体 R^* は自由領域 FG の中に到達している。

3. 2 修正アルゴリズム $Bug2^*$

これは、アルゴリズム $Class2^*$ に“線分条件”を付加したものである。ここでは、アルゴリズム $Class2^*$ から変更された手続きのみを記述する (Fig.6(a)).

2.(b) 頭 R が線分 SG 上にあり、点 H_i よりも目的地 G に接近し、かつ、外積 $V_G \times V_T$ が負またはゼロ [正] のところを離脱点 L_i とし、 $i=i+1$ として1.へ戻る。

ここで、アルゴリズム $Bug2^*$ を Fig.6(a) で説明する。 $Bug2^*$ と $Class2^*$ は線分条件の有無が異なる。すなわち、前述の距離条件と物理条件に線分条件を加える。線分条件: 頭 R が線分 SG 上に位置するとき、頭 R は離脱点 L_1, L_2 , そして L_3 から V_S 方向へ進み、体 R^* は同期的に V_S^* 方向へ進む。

3. 3 修正アルゴリズム $Alg2^*$

これは、アルゴリズム $Class1^*$ に“復帰条件”を付加したものである。ここでは、アルゴリズム $Class1^*$ に付加した手続きのみを記述する (Fig.6(b)).

2.(d) 頭 R が過去の離脱点 L_k や衝突点 H_k ($k < i$) に戻ったら、時計回りまたは反時計回りのルートのうち短い方—どちらもすでに通過し距離は計算されている—で直前の衝突点 H_i まで戻る。そして、ベクトル V_T^* —反時計回り [時計回り] のもの—をベクトル V_S^* として体 R^* は走行し、それに同期して頭 R はベクトル V_S へ走行する。このとき、頭 R が点 C よりも目的地 G に接近したら、そこを新たに点 C とする。さらに、外積 $V_G \times V_T$ が正 [負またはゼロ] であれば、現在地を離脱点 L_i とし、 $i=i+1$ として1.へ戻る。

ここで、アルゴリズム $Alg2^*$ を Fig.6(b) で説明する。 $Alg2^*$ と $Class1^*$ の違いは復帰条件の有無である。すなわち、頭 R がコンピュータ世界で衝突点 H_1 に戻ったら、時計回りまたは反時計回りのルートのうち短い方で直前の衝突点 H_3 まで戻る。Fig.6(b) では、時計回りのルートが選択され、離脱点 L_1 が L_3 ,

Fig.6 (a), (b), (c) Several robust algorithms $Bug2^*$, $Alg2^*$, and $Alg1^*$ for a mobile robot with position and orientation errors, which are revised from the algorithms $Bug2$, $Alg2$, and $Alg1$.

衝突点 H_2 が H_4 , 離脱点 L_2 が L_4 , 衝突点 H_3 が H_5 となる. そして, 衝突点 H_5 で, 前回の回避方向 (衝突点 H_3 の時計回り) の逆を選ぶ. これらが復帰条件である. Fig.6(b) では, 反時計回りが選択され, 体 R^* は実世界で障害物を通り, 頭 R はコンピュータ世界を同期的に進む. そして, 頭 R は距離条件と物理条件を同時に満足する離脱点 L_5 を発見する. そして, 頭 R は V_s 方向へ進み, 体 R^* は実世界を同期的に V_{s^*} 方向へ進む. 最後に, 頭 R が目的地 G に到達したらアルゴリズムは終了する. このとき, 体 R^* は自由領域 F_G の中に到達している.

3. 4 修正アルゴリズム Alg1*

これは, アルゴリズム *Class1** に “線分条件” と “復帰条件” を付加したものである. ここでは, アルゴリズム *Class1** で変更された手続きや, それに付加した手続きのみを記述する (Fig.6(c)).

- 2.(b) 頭 R が線分 SG 上にあり, かつ点 C よりも目的地 G に接近すれば, そこを新しく点 C とする. さらに, そこで外積 $V_G \times V_T$ が負またはゼロ [正] であれば, 現在地を離脱点 L_i とし, $i=i+1$ として 1. へ戻る.
- 2.(d) 頭 R が過去の離脱点 L_k や衝突点 H_k ($k < i$) に戻ったら, 時計回りまたは反時計回りのルートのうち短い方—どちらもすでに通過し距離は計算されている—で直前の衝突点 H_i まで戻る. そして, ベクトル V_{T^*} —反時計回り [時計回り] のもの—をベクトル V_{s^*} として体 R^* は走行し, それに同期して頭 R はベクトル V_s へ走行する. そして, 頭 R は線分 SG 上で点 C よりも目的地 G へ接近したらそこを新たに点 C とする. さらに, 外積 $V_G \times V_T$ が正 [負またはゼロ] であれば, 現在地を離脱点 L_i とし, $i=i+1$ として 1. へ戻る.

4. 修正距離アルゴリズムのグロス

デッドロックフリー特性

修正距離アルゴリズムがグロスデッドロックフリー特性 (“正常終了条件” と “異常終了条件”) を維持するための必要十分条件は, 移動ロボットが同相誤差しか生まないことである.

補助定理 1 移動ロボットが同相誤差を生むとき, 位置 R^* は常に同じ位置 R に対応する. $\square$

証明 移動ロボットが同相誤差を生むとき, 真の位置 R^* と偽の位置 R は一対一に対応し, 題意は満足される. $\blacksquare$

補助定理 2 移動ロボットが同相以外の誤差を生むとき, 位置 R^* は必ずしも同じ位置 R に対応しない. $\square$

証明 移動ロボットが同相以外の誤差を生むとき, 真の位置 R^* と偽の位置 R は必ずしも一対一に対応しない. したがって, 題意は満足される. $\blacksquare$

定理 1 実世界に DFP が存在しないとき, 修正距離アルゴリズムが “異常終了条件” を維持するための必要十分条件は, 移動ロボットが同相誤差しか生まないことである. $\square$

証明 実世界に DFP が存在しないなら, 初期地 S と目的地

Fig.7 (a) A real loop L^* without any leave point is detected by checking if its virtual loop L without it is detected. (b) A real loop L^* with several leave points is detected by checking if its virtual loop L with them is detected.

G^* は障害物 O^* で完全に分断される (Fig.2(d)). 修正距離アルゴリズムでは, 頭 R が真の目的地 G をめざし, 体 R^* は偽の目的地 G^* をめざす. これより, 体 R^* は障害物 O^* と衝突したのちそれを一周しループ L^* をなす. このとき,

(十分性) 補助定理 1 より, 移動ロボットが同相誤差を生むなら, 実世界のループ L^* はコンピュータ世界のループ L に写像され, 頭 R はそれを認識できる (Fig.7(a)).

(必要性) 補助定理 2 より, 移動ロボットが同相以外の誤差を生むなら, 実世界のループ L^* は必ずしもコンピュータ世界のループ L に写像されず, 頭 R はそれを認識できないことから, 題意は満足される. $\blacksquare$

定理 2 修正距離アルゴリズムが “復帰条件” を維持するための必要十分条件は, 移動ロボットが同相誤差しか生まないことである. $\square$

証明 修正距離アルゴリズムが “復帰条件” を維持するためには, 体 R^* が点 H_k^* または L_k^* ($k < i$) へ戻った (ループ L^* が生じた) とき, 頭 R が点 H_k または L_k へ戻った (ループ L が生じた) ことを認識できればよい. このとき,

(十分性) 補助定理 1 より, 移動ロボットが同相誤差を生むなら, 点 H_k^* または L_k^* (実世界のループ L^*) は点 H_k または L_k (コンピュータ世界のループ L) へ写像され, 頭 R はそれを認識できる (Fig.7(b)).

(必要性) 補助定理 2 より, 移動ロボットが同相以外の誤差を生むなら, 点 H_k^* または L_k^* (ループ L^*) は点 H_k または L_k (ループ L) へ写像されないことから, 頭 R はそれを認識できず, 題意は満足される. $\blacksquare$

補助定理 3 実世界にDFPが存在するとき、頭R(そして体R*)が回避中の障害物O(そして障害物O*)から離脱できる十分条件は、移動ロボットが同相誤差しか生まないことである。□

証明 頭Rが点Cよりも目的地Gに近い点へ到達できる限り、それは点 H_i や L_i よりも目的地Gに近い点を必ず通過できる。したがって、ここでは、点C(H_i)で距離条件を定義し証明を与える。

さて、移動ロボットが同相誤差を生じて、位相的な性質は保存される。これより、線分 $C^*G^*(H_i^*G^*)$ と障害物 O^* の交差、そして線分 $CG(H_iG)$ と障害物Oの交差は共に維持される。一般に、線分 $C^*G^*(H_i^*G^*)$ や $CG(H_iG)$ において、点 $C^*(H_i^*)$ や $C(H_i)$ に最も近い交点 I^* や I は、少なくとも離脱点 L^* や L_i となりうる。なぜなら、まず、線分 $C^*G^*(H_i^*G^*)$ や $CG(H_iG)$ は交点 I^* と I で障害物 O^* やOから出ていくので、外積 $V_G^* \times V_{T^*}$ と $V_G \times V_T$ の符号は一致し物理条件は両立する(Fig.8(a),(b))。次に、交点 I^* と I は線分 $C^*G^*(H_i^*G^*)$ と $CG(H_iG)$ 上にあるので線分条件が両立する。さらに、交点 I^* と I は点 $C^*(H_i^*)$ と $C(H_i)$ よりも目的地 G^* とGに近いので距離条件も両立する($|C^*G^*(=H_i^*G^*)| > |I^*G^*| \Leftrightarrow |CG(=H_iG)| > |IG|$) (Fig.8(a),(b))。最後に、実世界にDFPが存在する限り、体 R^* や頭Rは障害物 O^* やOを一周できる。このことから、体 R^* や頭Rは少なくとも交点 I^* や I で3つの条件を同時に満足し、そこを離脱点 L_i^* や L_i とする。もちろん、3つの条件が満足されれば、点 I^* や I 以外の交点を点 L_i^* や L_i とできる。以上のことから、題意は満足される。■

補助定理 4 実世界にDFPが存在するとき、頭R(そして体R*)が回避中の障害物O(そして障害物O*)から離脱できる必要条件は、移動ロボットが同相誤差しか生まないことである。□

証明 移動ロボットが同相以外の誤差を生むとき、位相的な性質は保存されない。たとえば、体 R^* が位置 R_1^* から R_2^* まで(頭Rが位置 R_1 から R_2 まで)走行する間に、線分 $C^*G^*(H_i^*G^*)$ と障害物 O^* の交差、そして線分 $CG(H_iG)$ と障害物Oの交差が解消されることが起こりうる(Fig.8(c),(d))。これより、線分条件は必ずしも維持されない。また、目的地までの距離の大小も保存されないの、たとえば、距離 $|R_1^*G^*| > |R_2^*G^*| \Leftrightarrow |R_1G| < |R_2G|$ が生じたら、距離条件も両立しない(Fig.8(e))。以上のことから、題意は満足される。■

定理 3 実世界にDFPが存在するとき、修正距離アルゴリズムが“正常終了条件”を維持するための必要十分条件は、移動ロボットが同相誤差しか生まないことである。□

証明 ベクトル V_G^* が実際の障害物 O^* に妨げられない—線分 $L_{i-1}^*G^*$ が障害物 O^* と交差しない—なら、ベクトル V_G も架空の障害物Oに妨げられない—線分 $L_{i-1}G$ は障害物Oと交差しない。このとき、頭Rは真の目的地Gに到達し、体 R^* も偽の目的地 G^* に到達する。これより、正常終了条件は満足される。一方、ベクトル V_G^* が点 $H_i^*(=C^*)$ で障害物 O^* に妨

げられる—線分 $H_i^*G^*(=C^*G^*)$ が障害物 O^* と交差する—なら、ベクトル V_G も点 $H_i(=C)$ で障害物Oに妨げられる—線分 $H_iG(=CG)$ も障害物Oと交差する(Fig.8(a),(b))。このとき、頭Rが回避中の障害物Oから離脱できる必要十分条件は、移動ロボットが同相誤差しか生まないことである(補助定理3と4)。

さて、頭Rが回避中の障害物Oから離脱できる限り、距離条件より頭Rは目的地Gへ単調に接近し、最後にそこへ到達する。このとき、点 L_i , H_i , Cの点Gへの収束は、点 L_i^* , H_i^* ,

Fig.8 (a), (b) R^* and R simultaneously find intersection points I^* and I as leave points L_i^* and L_i around obstacles O^* and O . (c) R^* and R simultaneously collide with O^* and O at hit points H_i^* and H_i . (d) After R^* and R follow O^* and O from R_1^* and R_1 to R_2^* and R_2 , respectively, R^* and R cannot find leave points L_i^* and L_i because R cannot keep the segment condition. (e) R^* cannot leave O^* since R cannot maintain the metric condition around O .

Fig.9 (a) R converges to G in CW and simultaneously R^* with the homeomorphic error converges to G^* in RW. (b) R^* without the homeomorphic error does not converge to G^* in RW since R does not converge to G in CW.

C^* の点 G^* への収束に対応する。すなわち、 $L_i, H_i, C \rightarrow G (i \rightarrow \infty) \Leftrightarrow f^l(L_i)=L_i^*, f^l(H_i)=H_i^*, f^l(C)=C^* \rightarrow f^l(G)=G^* (i \rightarrow \infty)$ が成り立つ (Fig.9(a))。以上のことから、題意は満足される。■

この同相写像では、たとえば、 $|H_1^*L_1^*| < |H_2^*L_2^*| \Leftrightarrow |H_1L_1| > |H_2L_2|$ のように、各点間の距離の大小は保存されないが、 $|H_1^*G^*| > |H_2^*G^*| \Leftrightarrow |H_1G| > |H_2G|$ のように、各点から目的地までの距離の大小は保存される。すなわち、“近さ”の概念は、距離ではなく点列の収束で保証される。この性質より、距離条件が頭 R を真の目的地 G へ収束させれば、体 R^* は確実に偽の目的地 G^* へ収束する。

一方、実世界とコンピュータ世界が同相でなければ、たとえば、 $|H_2^*G^*| > |I^*G^*| \Leftrightarrow |H_2G| < |IG|$ のように、各点から目的地までの距離の大小さえ保存されない (Fig.9(b))。これより、コンピュータ世界で頭 R が距離条件を満足できず、実世界で体 R^* が無限ループ(デッドロック)に陥ることがある。さらに、コンピュータ世界で頭 R が線分条件を満足できず、実世界で体 R^* が無限ループに陥ることもある。

5. シミュレーション結果

ここでは、線形誤差を同相誤差として用いたとき、修正距

Fig.10 (a) A robot head R arrives at a destination G in CW by the modified algorithm $Class1^*$. (b) A robot body R^* simultaneously arrives at a virtual destination G^* near the destination G in RW.

Fig.11 (a) A robot head R arrives at a destination G in CW by the modified algorithm $Alg2^*$. (b) A robot body R^* simultaneously arrives at a virtual destination G^* near the destination G in RW.

Fig.12 (a) A robot head R arrives at a destination G in CW by the modified algorithm $Bug2^*$. (b) A robot body R^* simultaneously arrives at a virtual destination G^* near the destination G in RW.

Fig.13 (a) A robot head R arrives at a destination G in CW by the modified algorithm $Alg1^*$. (b) A robot body R^* simultaneously arrives at a virtual destination G^* near the destination G in RW.

離アルゴリズム $Bug2^*$, $Class1^* \sim 3^*$, $Alg1^* \& 2^*$ がグロスデッドロックフリー特性を満足することを2種類の未知空間で確認する。どちらの空間も、真の目的地 G の周囲に自由領域 F_G を

持つので、 G と G^* は決して障害物に分断されない。

まず、ある未知空間で、修正距離アルゴリズム $Class1^*$ と $Alg2^*$ の頭 R と体 R^* の状態を各々Fig.10, Fig.11に示す。ここでは、誤差の影響より、ベクトル V_s に対してベクトル V_s^* は常に 5° 右へ見積もられ、かつ、走行距離は 5% 過小に見積もられる。次に、別の未知空間で、修正距離アルゴリズム $Bug2^*$ と $Alg1^*$ の頭 R と体 R^* の状態を各々Fig.12, Fig.13に示す。ここでは、同じく誤差の影響より、ベクトル V_s に対してベクトル V_s^* は常に 3° 右へ見積もられ、かつ、走行距離は 6% 過大に見積もられる。

これらの結果より、移動ロボットが同相誤差を生む限り、修正距離アルゴリズムは、頭 R を真の目的地 G に収束させ、体 R^* を偽の目的地 G^* に収束させることが確認された。

6. おわりに

本稿では、誤差を処理できるように従来の距離アルゴリズムを修正したのち、それらがグロスデッドロックフリー特性を維持できる誤差の条件について考察した。その結果、移動ロボットが同相誤差しか生み出さないことがその条件であることがわかった。ここで、同相誤差は、スリップなどで走行距離を一定の割合で誤解したり、移動機構やジャイロなどの初期誤差で走行方向を一定量だけ誤解するときなどに生じる線形誤差(たとえば、アフィン変換で写像される誤差)を完全に含むが、それ以外にも閉ループで生じる位置誤差の合計がゼロとなる、たとえば、時系列で相殺されるランダムな誤差も含む。

さて、位相アルゴリズム $Curv1$ がグロスデッドロックフリー特性を維持できる必要十分条件も、移動ロボットが同相誤差しか生まないことである。位相アルゴリズム $Curv1$ は、線分 SG とそれが交差する障害物周囲のなす連結グラフ上で機能する。したがって、誤差のクラスが同相のときのみ、連結グラフの位相が保たれ、位相アルゴリズム $Curv1$ はグロスデッドロックフリー特性を維持できる。次に、幾何アルゴリズム $Robust$ は、線分 SG 上のみで幾何条件(物理条件を含む)を用いるが、誤差が同相であるときのみ、線分 SG と障害物周囲の交差、ならびに、その交点での幾何条件(物理条件を含む)は保存される。したがって、幾何アルゴリズム $Robust$ がグロスデッドロックフリー特性を維持できる必要十分条件も、移動ロボットが同相誤差しか生まないことである。

最後に、センサ誤差が存在すると正確なベクトル V_{T^*} が得られなくなり、外積 $V_G \times V_T$ と $V_G^* \times V_{T^*}$ の符号は必ずしも一致しなくなる。したがって、アルゴリズムのグロスデッドロックフリー特性は維持できない。このことから、センサベースナビゲーションが通常の誤差を克服する手だてを現在検討している。

なお、本研究の一部は、平成6年度文部省科学研究費補助金(課題番号06750266)の援助によるものであることを記し、謝意に代える。

参考文献

- 1) たとえば, T. Lozano-Perez and M. Wesley: An Algorithm for Planning Collision Free Paths among Polyhedral Obstacles, *Commun. ACM*, 22-10, 560/570 (1979).
- 2) V. Lumelsky and A. Stepanov: Path Planning Strategies for a Point Mobile Robot Moving amidst Unknown Obstacles of Arbitrary Shape, *Algorithmica*, 2, 403/430 (1987).
- 3) H. Noborio: A Path-Planning Algorithm for Generation of an Intuitively Reasonable Path in an Unknown 2D Workspace, *Proc. of the Japan-USA Symposium on Flexible Automation*, 2, 477/480 (1990).
- 4) 登尾啓史: 未知空間におけるパスプランニングアルゴリズムを設計するための十分条件とそれにもとづくアルゴリズムの評価, 情報処理学会論文誌, 33-8, 1013/1021 (1992)
- 5) A. Sankaranarayanan and M. Vidyasager: A New Path Planning Algorithm for Moving a Point Object amidst Unknown Obstacles in a Plane, *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 1930/1936 (1990).
- 6) A. Sankaranarayanan and M. Vidyasager: Path Planning for Moving a Point Object amidst Unknown Obstacles in a Plane: A New Algorithm and a General Theory for Algorithm Development, *Proc. of the 29th CDC*, 1111/1119 (1990).
- 7) A. Sankaranarayanan, et al.: A New Algorithm for Robot Curve-Following amidst Unknown Obstacles, and a Generalization of Maze-Searching, *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 3, 2487/2494, (1992).
- 8) 吉岡孝, 登尾啓史, "障害物の幾何学的性質にもとづくセンサベースデッドロックフリーナビゲーションアルゴリズム," 計測自動制御学会論文集, 33-4, 280/288, (1997).
- 9) 吉岡孝, 登尾啓史, 富永昌治 "障害物が密集する未知環境で位置や姿勢の誤差を克服するセンサベースナビゲーション," 計測自動制御学会論文集, 34-2, 157/165, (1998).
- 10) H. Noborio: On a Sensor-Based Navigation for a Mobile Robot, *J. Robotics and Mechatronics*, 8-1, 2-14 (1996)
- 11) Y. Ando and S. Yuta: Following a Wall by an Autonomous Mobile Robot with a Sonar-Ring, *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2599/2605 (1995).
- 12) 大矢晃久, 永島良昭, 油田信一: 超音波による壁面の法線方向の高速測定, 日本ロボット学会誌, 13-5, 700/703 (1995).
- 13) T. Yoshioka and H. Noborio: On Deadlock-Free Characteristics in Previous Sensor-Based Path-Planning Algorithms by a Dead Reckoning Error, *Proc. of the Japan-USA Symp. Flexible Automation*, 645/648 (1994).
- 14) H. Noborio, T. Yoshioka, and T. Hamaguchi: On-Line Deadlock-Free Path-Planning Algorithms in the Presence of a Dead Reckoning Error, *Proc. of the IEEE Int. Conf. on SMC*, 483/488 (1995).

「著者紹介」

登尾 啓 史 (正会員)

1987年, 大阪大学大学院基礎工学研究科物理系専攻博士後期課程修了。同年同大学基礎工学部機械工学科助手, 88年大阪電気通信大学工学部精密工学科専任講師。現在, 同大学情報工学部情報工学科教授。96年ミュンヘン工科大学客員研究員(フンボルト財団招聘)。ロボットの知能化, 人間機械系の研究に従事。日本ロボット学会, 情報処理学会, IEEEなどの会員。(工学博士)

吉 岡 孝

1993年, 大阪電気通信大学工学部精密工学科卒業。98年同大学大学院工学研究科情報工学専攻博士後期課程修了。現在, 同大学情報工学部情報工学科客員研究員。移動ロボットのナビゲーション, ロボットビジョンの研究に従事。日本ロボット学会の会員。(博士(工学))