

経路の未来情報を利用した車輪型 自立移動ロボットの経路追従制御

橋本幸男*¹ 王碩玉*² 土谷武士*³ 松田敏彦*¹

Path-Tracking of Wheeled Mobile Robot Using Preview and Predictive Control

Yukio Hashimoto*¹, Shouyu Wang*², Takeshi Tsuchiya*³ and Toshihiko Matsuda*¹

This paper describes a method to make a wheeled mobile robot follow a given path just like a driver would decelerate the speed of a car before a corner and accelerate it on the straight course of the road. Two kinds of velocity vectors are introduced to determine an optimal direction and speed of the robot in order to follow the path. The velocity defined as a function of curvature of the path is mainly utilized to change the speed of the robot along the path. The other velocity given as the tangent of a virtual potential function produces an attraction force toward the path. Path tracking control of the robot is formulated with current and future sets of these velocity vectors in a frame of the digital preview and predictive control. By this method, the robust tracking performance is achieved.

Key Words: Path Control, Mobile Robot, Preview Control, Predictive Control, Tracking Control

1. はじめに

良識的なドライバーは、カーブの手前では車のスピードを落とし、カーブの出口に近づいてから徐々にスピードを上げて、安全にカーブを通過しようとする。製品や部品を搬送する移動ロボットの場合でも、慣性力や駆動系の制御性能までを考慮に入れると、コースの曲率が大きく変化する場所では速度を落とした方が荷とロボットに働く加速度の変化を小さく抑えることができ、無理のない走行が実現できると予想される。

人が車を運転するときには、進行方向の道路の様子に合わせて絶えずスピードの加減とハンドル操作を行う。例えば、高速で走行している場合には対応が遅れないように、遠方までの道路の様子を見てスピードとハンドル操作を決めているし、カーブでは車に無理がかからないように、進行位置を予測しながらスピードの調節とハンドルの操作を注意深く行っている。車の運転では、進路先の道路の情報と車の働きの予測から現在の車のスピードとハンドル操作を決めており、軌道計画と制御が同時に行われている点に特徴があるように思われる。

車輪型移動ロボットに車のような加減速走行を行わせる場合にも、軌道の計画と経路からのずれを修正する制御が行われる必要がある。経路の曲率が大きくなるほど減速して進むべき

であるのは明白なので、このうちロボットの進行スピードの計画に関しては、経路の曲率の関数として与えることで解決ができる問題と考えられる。しかしながら、経路にロボットを拘束するための制御については、ロボットの位置と姿勢角が非線形な関係で結ばれるので[1][2]、任意経路に沿わせるロバストなフィードバック制御系を構成するのは簡単ではないという問題がある。また、ロボットの動きを予測しながら加減速制御を同時に実行する問題については、これまで明確な形で議論されたことはなかったように思われる。

このようなことから、本論文では、PWSタイプの移動ロボット[3]に、人が車を運転するときのようなスピードの調節と操舵方法を自立的に行わせるためのひとつの制御方法について検討する。ロボットの進行スピードと姿勢角に対しては線形制御器が設計できるので、駆動系の制御と経路に拘束する制御を階層的に分離して考え、進行スピードと姿勢角に関する漸近安定な下層線形サーボ系を最初に構成する。ロボットを経路上に拘束して可変速走行させる上層制御に関しては、仮想的なポテンシャル関数を導入し、ロボットの進行スピードを経路の曲率から決めた速度ベクトルとこのポテンシャル関数の傾きから求めた速度ベクトルを使って指定する。また、ロボットのとるべき姿勢角もこれらの二つの速度ベクトルから算出する。以上の構成により、滑らかな経路が与えられるとロバストなフィードバック経路追従制御が実現される。次に、車の運転のときのよう経路の情報をもっと利用する手段として、予見制御[4][5][6]の考え方を導入する。オリジナルな予見制御は目標値を時間関数として定式化されているが、経路追従制御の場合は経路上の位置とそこでの速度ベクトルが与えられているだけで

原稿受付 1993年6月30日

*室蘭工業大学

*山形大学工学部

*北海道大学工学部

*Muroran Institute of Technology

*Faculty of Engineering, Yamagata University

*Faculty of Engineering, Hokkaido University

あり、そのままでは適用ができない。このため、現時点のロボットの進行スピードから将来の位置を予測し、その位置でのスピードを未来の目標値として利用する。結果的にロボットの進行スピードが速いほど遠方の経路情報を利用できることになり、車の運転の場合と同じような加減速走行が実現される。

2. 移動ロボットのダイナミクスと制御方針

移動ロボットを経路に沿わせて運行させる制御では、移動機構部分には理想的な速度サーボ系が設計されていることを前提とし、ロボットの力学的なダイナミクスを無視することが多かった[1][2][7]。しかしながら、現実にはサーボ系は固有のダイナミクスをもっており、しかもそのダイナミクスはロボットの質量の変化や路面抵抗の変化などのパラメータ変化の影響を受けて変わるので、ロバストな経路追従制御を実現するためには、駆動系のダイナミクスも考慮に入れる方が望ましい。また、移動ロボットではマイクロ・コンピュータを使い、デジタル的に制御を実施するのが普通であるから、この経路追従制御は離散時間形式の制御として定式化し、議論する方が好ましいと思われる。

以上の点を考慮に入れ、DC サーボモータによって車輪が駆動される「山彦」タイプの車輪型ロボット[3]の離散時間形式の可変速経路追従制御を検討する。そのための準備として、本章では駆動系を含めたロボットの移動ダイナミクスを明らかにし、経路追従制御の方針について述べる。

いま、Fig. 1 に示すような固定座標系を導入し、記号を Table 1 のように定義する。簡単のため、両駆動輪を結ぶ線分の中央に重心 G があるものとする。 G の速さ V_G (ロボットの進行スピード) は

$$V_G = \frac{r}{2}(\dot{\theta}_1 + \dot{\theta}_2) \quad (1)$$

となるので、幾何学的な関係より

$$\dot{X} = V_G \cos \phi \quad (2)$$

$$\dot{Y} = V_G \sin \phi \quad (3)$$

また、姿勢角の時間的な変化については

$$\dot{\phi} = \frac{r}{L}(\dot{\theta}_1 - \dot{\theta}_2) \quad (4)$$

したがって、式(2)~(4)を使うと、ロボットの移動に伴う運動エネルギーが計算でき、次式のように求められる。

$$K.E. = \frac{M}{2}(\dot{X}^2 + \dot{Y}^2) + \frac{I}{2}\dot{\phi}^2 + \frac{J}{2}(\dot{\theta}_1^2 + \dot{\theta}_2^2) \quad (5)$$

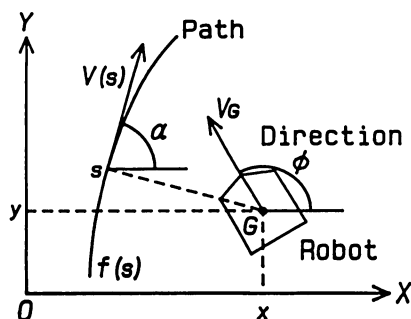


Fig. 1 Coordinate System to Describe Robot Motion

ここで、右辺の第2項目はロボットの旋回に伴う回転運動エネルギー、第3項目は車輪の回転に伴う運動エネルギーをそれぞれ表す。なお、 J は駆動系の全体の慣性モーメントであり、ギアによって車輪が駆動される場合には

$$J = J_w + \mu^2 J_o$$

で与えられる。簡単のため路面抵抗を一定とみなし、式(5)をラグランジュの運動方程式[8]に代入して $\dot{\theta}_1$, $\dot{\theta}_2$ の従う関係式を求めると次式を得る。

$$\left[\frac{Mr^2}{4} + \frac{Ir^2}{L^2} + J \right] \dot{\theta}_1 + \left[\frac{Mr^2}{4} - \frac{Ir^2}{L^2} \right] \dot{\theta}_2 + f = \tau_1 \quad (6)$$

$$\left[\frac{Mr^2}{4} - \frac{Ir^2}{L^2} \right] \dot{\theta}_1 + \left[\frac{Mr^2}{4} + \frac{Ir^2}{L^2} + J \right] \dot{\theta}_2 + f = \tau_2 \quad (7)$$

ただし、 τ_1 , τ_2 はそれぞれ DC サーボモータが発生するトルクを、 f は路面抵抗を表す。通常、このようなロボットでは、機械的な時定数は電気系の時定数に比較して著しく大きくなるので、DC サーボモータ内の電気系のダイナミクスは無視できる。したがって、車輪を駆動するトルクの値は、おのおののモータの電機子電圧を使って次のように表される。

$$\tau_i = \frac{\mu K}{R}(v_i - \mu K \dot{\theta}_i) \quad (8)$$

Table 1 Parameters and variables

(X, Y) : ロボットの重心の座標 (Position of Gravitational Center)
ϕ : ロボットの姿勢角 (X 軸を基準) (Attitude Angle of Robot)
V_G : ロボットの進行する速さ (Proceeding Speed of Robot)
M : ロボットの質量 (Mass of Robot)
I : 重心まわりの慣性モーメント (Rotational Inertia Moment of Robot)
L : 両駆動輪間の距離 (Distance between Driving Wheels)
r : 駆動輪の半径 (Radius of Driving Wheel)
μ : ギア比 (Ratio of Gearing)
J_o : DC サーボモータの慣性モーメント (Inertia Momentum of DC Motor)
J_w : 駆動輪の慣性モーメント (Inertia Momentum of Driving Wheel)
θ_1, θ_2 : 車輪の回転角度 (Angle of Driving Wheel)
K : モータのトルク定数 (Torque Constant of Servo Motor)
R : 電機子の巻線抵抗 (Armature Resistance of Motor)
v_1, v_2 : 電機子電圧 (制御入力) (Armature Voltage of Motor)
s : 経路座標 (Position along Path)
$p(s)$: 経路の方程式 (Path Equation Expressed by Path Length)

$$\tau_2 = \frac{\mu K}{R}(v_2 - \mu K \dot{\theta}_2) \quad (9)$$

以上の関係式を使うことにより、移動ロボットの運動は完全に記述できるが、ロボットの位置も含めた状態方程式を構成すると、式(2)(3)が非線形な関係式になっているために全体として非線形な方程式となり、経路追従制御系を設計するのは難しくなる。ところが、ロボットの姿勢角と進行速度に関する式だけに着目すると、非線形性は現れず制御の設計が容易に行える。そこで線形制御で制御できるところは、厳密に漸近安定性が保証できる制御系を構成しておき、非線形ダイナミクスをもつ位置の制御に対しては、ロボットの運行法として別の上位制御ループを用意して処理する方法が考えられる。

いま、ロボットの姿勢角と進行速度に関する関係式を選び出すと、式(4)(6)(7)(8)(9)から次の線形状態方程式が構成できる。

$$\dot{\mathbf{x}}_c = \mathbf{A}_c \mathbf{x}_c + \mathbf{B}_c \mathbf{u} + \mathbf{d}_c f \quad (10)$$

ただし

$$\begin{aligned} \mathbf{x}_c &= [x_1, x_2, x_3]^T = [\phi, \dot{\theta}, \dot{\theta}_2]^T \\ \mathbf{u} &= [v_1, v_2]^T \\ \mathbf{A}_c &= \begin{bmatrix} 0 & r/L & -r/L \\ 0 & a_{22} & a_{23} \\ 0 & a_{32} & a_{33} \end{bmatrix} \\ \mathbf{B}_c &= \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix}, \quad \mathbf{d}_c = \begin{bmatrix} 0 \\ d_2 \\ d_3 \end{bmatrix} \\ a_{22} = a_{33} &= -\frac{\mu^2 K^2 [4Ir^2 + L^2(Mr^2 + 4J)]}{2R(Mr^2 + 2J)(2Ir^2 + JL^2)} \\ a_{23} + a_{32} &= -\frac{\mu^2 K^2 r^2 (4I - ML^2)}{2R(Mr^2 + 2J)(2Ir^2 + JL^2)} \\ b_{21} = b_{32} &= \frac{\mu K [4Ir^2 + L^2(Mr^2 + 4J)]}{2R(Mr^2 + 2J)(2Ir^2 + JL^2)} \\ b_{22} = b_{31} &= \frac{\mu K r^2 (4I - ML^2)}{2R(Mr^2 + 2J)(2Ir^2 + JL^2)} \\ d_2 = d_3 &= -\frac{2}{Mr^2 + 2J} \end{aligned}$$

ここで、ロボットの姿勢角と重心の速さを式(10)の線形系の出力と考えて出力方程式を作ると

$$\mathbf{y} = \mathbf{C} \mathbf{x}_c \quad (11)$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & r/2 & r/2 \end{bmatrix}$$

式(10)式(11)の行列の間に

$$\text{rank} \begin{bmatrix} \mathbf{A}_c & \mathbf{B}_c \\ \mathbf{C} & \mathbf{0} \end{bmatrix} = 5$$

の関係が成立することは直接確認できる。したがって、電機子電圧を変えることによって、ロボットの進行速度と姿勢角を独立に制御するサーボ系が構成可能である[9]。前述したように、ロボットはデジタル的に制御されるのが普通であるから、サーボ系は、式(10)(11)の関係を一定の制御周期 T で離散化して導出した次の系に対して設計するのが適当と思われる。

$$\mathbf{x}(k+1) = \mathbf{A} \mathbf{x}(k) + \mathbf{B} \mathbf{u}(k) + \mathbf{d} f \quad (12)$$

$$\mathbf{y}(k) = \mathbf{C} \mathbf{x}(k) \quad (13)$$

ただし

$$\begin{aligned} \mathbf{A} &= \exp(\mathbf{T} \mathbf{A}_c), \quad \mathbf{B} = \int_0^T \exp(s \mathbf{A}_c) ds \mathbf{B}_c \\ \mathbf{d} &= \int_0^T \exp(s \mathbf{A}_c) ds \mathbf{d}_c \end{aligned}$$

を表す。

駆動サーボ系に対する姿勢角と進行速度の指令値の与え方によってロボットの位置が決定される。ただし経路上からロボットをスタートさせ、経路の向きと経路の曲率から姿勢角と進行速度をそれぞれ求めて駆動サーボ系に与えたとしても、これらの指令値が時間的に変わっていくこととサーボ系の応答の遅れのために、ロボットは経路から離れてしまう。経路上にロボットを拘束した状態で走行させるためには、経路からの偏差を零にする制御が必要になる。理想的には、位置に関する非線形な関係式(2)(3)を考慮に入れて、フィードバック制御系を構成することが望まれるが、離散時間形式で移動ロボットの経路追従制御を定式化し、実現することは現段階では難しいと思われる。そこで、式(2)(3)の非線形な関係式を直接扱うことは避けることとし、ロボット・マニピュレータの経路制御[10]の場合のように、経路の距離できる仮想的なポテンシャル関数を導入し、このポテンシャル関数の傾きを利用してロボットの位置と速度を同時に制御する方法を採用する。

以降章では、ロボットの姿勢角と進行速度を制御するためのデジタルサーボ系の構成と、ロボットを経路上に拘束しながら可変速運行を行わせるためのフィードバック制御系の詳細について記述する。経路の未来情報を利用するための予見制御の導入については、章を改めて4章で述べる。なお、以降の議論では、車輪の回転速度はタコジネレータなどで、またロボットの姿勢角は磁気コンパス[11]などで直接観測ができるものと仮定する。

3. 経路追従サーボ制御系の構成

3.1 姿勢角と進行速度に対する1形の最適状態フィードバックサーボ系

ロボットの目標姿勢角 ϕ_d と目標進行速度 V_{Cd} を成分とする目標値ベクトルを \mathbf{y}_d と表わし、偏差ベクトルを次式で定義する。

$$\mathbf{e}(k) = \mathbf{y}_d - \mathbf{y}(k) \quad (14)$$

ここで、後退差分演算子 Δ を導入して式(11)に作用させると、 \mathbf{y}_d が定値ベクトルであることから、差分はゼロとなり

$$\Delta \mathbf{e}(k+1) = -\Delta \mathbf{y}(k+1) \quad (15)$$

一方、式(13)(12)にも差分演算子を作用させると、路面抵抗を一定と仮定していることより外乱項が消えて

$$\Delta \mathbf{y}(k+1) = \mathbf{C} \Delta \mathbf{x}(k+1) \quad (16)$$

$$\Delta \mathbf{x}(k+1) = \mathbf{A} \Delta \mathbf{x}(k) + \mathbf{B} \Delta \mathbf{u}(k) \quad (17)$$

したがって、これらの関係を式(15)に代入して左辺の差分を元にもどすと、以下のような関係式を導くことができる。

$$\mathbf{e}(k+1) = \mathbf{e}(k) - \mathbf{C} \mathbf{A} \Delta \mathbf{x}(k) - \mathbf{C} \mathbf{B} \Delta \mathbf{u}(k) \quad (18)$$

ここで、式(18)(17)を使って拡大系を構成すると

$$\begin{bmatrix} \mathbf{e}(k+1) \\ \Delta \mathbf{x}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{I} & -\mathbf{C} \mathbf{A} \\ \mathbf{0} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{e}(k) \\ \Delta \mathbf{x}(k) \end{bmatrix} + \begin{bmatrix} -\mathbf{C} \mathbf{B} \\ \mathbf{B} \end{bmatrix} \Delta \mathbf{u}(k)$$

(19)

このシステムは原系が可制御で、 $z=1$ に不変零点をもたなければ可制御であることが示されるので[12]、状態フィードバック制御によって自由にシステムの応答の速さを設計できる。パラメータの値には同定誤差が含まれていることを前提とした方がよいと思われるので、この状態フィードバック制御則としては、LQR 問題を經由して決定するのが適当と考えられる。いま、半正定対称行列 $Q \in R^{5 \times 5}$ と正定対称行列 $R \in R^{2 \times 2}$ を与え、次の評価関数

$$P.I. = \sum_{k=0}^{\infty} [z^T(k)Qz(k) + \Delta u^T(k)R\Delta u(k)] \quad (20)$$

$$z(k) = [e^T(k), \Delta x^T(k)]^T$$

を最小にする状態フィードバック則を求めると

$$\begin{aligned} \Delta u(k) &= F_e e(k) + F_x \Delta x(k) \\ &= F_e e(k) + F_x [\Delta x(k) - x(0)] \end{aligned} \quad (21)$$

ただし

$$[F_e, F_x] = -(R + \Gamma_0^T P \Gamma_0)^{-1} \Gamma_0^T P \Phi_0$$

$$\Phi_0 = \begin{bmatrix} I & -CA \\ 0 & A \end{bmatrix}, \quad \Gamma_0 = \begin{bmatrix} -CB \\ B \end{bmatrix}$$

で、 P は次のリカッチ方程式の解である[13]。

$$P = Q + \Phi_0^T P \Phi_0 - \Phi_0^T P \Gamma_0 (R + \Gamma_0^T P \Gamma_0)^{-1} \Gamma_0^T P \Phi_0$$

なお、式(21)の $x(0)$ は初期値補償の意味で導入している。

ここで、式(21)の両辺に

$$\nabla z(k) = \sum_{j=0}^k z(j)$$

で定義される演算子 ∇ を作用させると、差分演算との間に

$$\nabla(\Delta z(k)) = z(k)$$

の関係が成り立つことから

$$u(k) = F_e \sum_{j=0}^k e(j) + F_x [x(k) - x(0)] \quad (22)$$

なる位置形のフィードバック制御則が導かれる。

3.2 経路上における進行スピードと姿勢角

ロボットが経路上を移動しているときに働く遠心力はロボットの移動速度と経路の曲率で決められる。したがって、ロボットに働く遠心力を小さくし経路からの逸脱やスリップを最小限に抑えるためには、経路の曲率の大きなところほど進行スピードを遅くしなければならない。曲率がゼロに近い直線的な経路では高速度で移動しても問題はないから、ロボットの経路に沿った速度指令値は経路の曲率の関数として与えるのが妥当のように思われる。このようなことから、経路上の位置 s におけるロボットの目標進行スピード $V_{//}$ を、ここでは次式を使って与えることを提案する。

$$V_{//}(k) = v_{\max} \exp\{-\kappa [p'(s)]^2\} \quad (23)$$

ただし、 $p'(s)$ は経路の曲率を表し、 v_{\max} 、 κ はそれぞれ正の定数とする。直線経路に適用すると分かるように、 v_{\max} はロボットのとり得る最大の進行スピードを表している。

ロボットが経路上を移動しているときの姿勢角の指令値については、経路の進行方向の接線と固定座標系の X 軸とのなす角 α で与える (Fig. 1 参照)。

3.3 経路上にロボットを拘束する制御

駆動系に対して設計したサーボ系のダイナミクスが無視でき

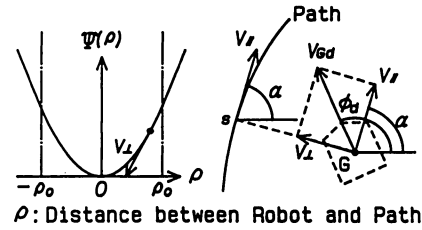


Fig. 2 Potential for Path Tracking

るのであれば、前節に示した指令値の与え方でもロボットは経路に沿って可変速移動を行う。しかしながら、実際にはサーボ系の応答遅れのせいで、ロボットが経路上からスタートしたとしても、位置誤差が累積して経路から離れてしまう。ロボットが経路上にあっても、あるいは経路から離れたところからスタートしても、必ず経路に沿って進むようにするには、位置に関するフィードバック制御系を構成しなければならない。ここではそのための手段として、経路上で極小値をとる仮想的なポテンシャル関数を導入する。

Fig. 2 に示すように、経路の周辺に経路からの距離 ρ で決まる滑らかなポテンシャル関数 $\Psi(\rho)$ を考える。いまポテンシャル関数を

$$\Psi(\rho) = \begin{cases} -2K_{\perp}\rho_0(\rho + \rho_0) + K_{\perp}\rho_0^2 & (\rho < -\rho_0) \\ K_{\perp}\rho^2 & (|\rho| \leq \rho_0) \\ 2K_{\perp}\rho_0(\rho - \rho_0) + K_{\perp}\rho_0^2 & (\rho_0 < \rho) \end{cases}$$

ただし、 K_{\perp} 、 ρ_0 はそれぞれ正定数

で構成するものとし、経路と直交する方向のロボットの速度 V_{\perp} をこの関数の傾きを使い、次式で与えることにする。

$$V_{\perp}(k) = -\gamma_s \frac{\partial \Psi(\rho)}{\partial \rho(k)} \quad (24)$$

ただし、 γ_s は $\gamma_s > 0$ なる定数

このとき、ロボットが $|\rho| > \rho_0$ の位置にあるときにはポテンシャルが一次関数で与えられていることから

$$\Delta \Psi[\rho(k+1)] = \Psi[\rho(k+1)] - \Psi[\rho(k)]$$

$$= \frac{\partial \Psi(\rho)}{\partial \rho(k)} \Delta \rho(k+1)$$

$$= -T\gamma_s \left[\frac{\partial \Psi(\rho)}{\partial \rho(k)} \right]^2$$

$$< 0$$

したがって、ロボットが式(24)にしたがって移動するかぎり、ポテンシャル値は減少を続け、やがて $|\rho| \leq \rho_0$ の領域に達する。 $|\rho| \leq \rho_0$ の領域においては、ポテンシャルが2次関数で与えられていることに注意して

$$\Delta \Psi[\rho(k+1)] = \Psi[\rho(k+1)] - \Psi[\rho(k)]$$

$$= \frac{\partial \Psi(\rho)}{\partial \rho(k)} \Delta \rho(k+1) + \frac{1}{2} \frac{\partial^2 \Psi(\rho)}{\partial \rho(k)^2} \Delta \rho(k+1)^2$$

$$= -T\gamma_s \left[\frac{\partial \Psi(\rho)}{\partial \rho(k)} \right]^2 (1 - T\gamma_s K_{\perp})$$

$T\gamma_s K_{\perp}$ は定数であるから、 $T\gamma_s K_{\perp} < 1$ のように選ばれているかぎり

$$\Delta \Psi[\rho(k+1)] \leq 0$$

となり、常にポテンシャル値が減少を続ける。したがって、最終的にはポテンシャルの谷の位置、すなわち経路に到達して停

止する。現実にはサーボ系の応答遅れのために、式(24)でスピード指令を与えても、1制御周期の間に

$$\Delta\rho(k+1)=TV_{\perp}(k)$$

と等しい距離を移動することは期待できないが、ポテンシャルが減少する方向へは進むので、経路に到達するという最終的な結論には変わりはない。

ロボットは経路に沿って進むことも求められている。したがって、実際にロボットに与える速度指令値は、式(23)で与えられる速度ベクトル $V_{\parallel}(k)$ と式(24)の経路へロボットを拘束するための速度ベクトル $V_{\perp}(k)$ の和として与えることにする。以上のことから、ロボットの進行スピードと姿勢角の指令値はそれぞれ以下のように求められる。

$$V_{ca}(k)=\sqrt{V_{\parallel}(k)^2+V_{\perp}(k)^2} \quad (25)$$

$$\phi_a(k)=\tan^{-1}\left[\frac{V_{\parallel}(k)\sin\alpha(k)+V_{\perp}(k)\cos\alpha(k)}{V_{\parallel}(k)\cos\alpha(k)-V_{\perp}(k)\sin\alpha(k)}\right] \quad (26)$$

なお、 $V_{\parallel}(k)$ 、 $V_{\perp}(k)$ の大きさは、Fig. 2 に示すようにロボットの位置と最も近い経路上の点の位置で計算するものとする。また、ポテンシャル関数の構成の際に定義した定数 ρ_0 は、駆動系の能力を越えた速度指令値が生成されるのを避けるために導入したしきい値であり、式(25)から計算される進行スピードの最大値がロボットの駆動系の定格を越えないように適切に決めるものとする。

4. 予見制御によるロボットの可変速経路追従制御

4.1 経路の未来情報と予見制御

車を運転する場合、スピードを上げるときには遠方までの道路の様子を見きわめた上で行動を起こす。高速に至った後も遠方の様子に注意を払い続け、前方にカーブが見えた場合にはカーブにさしかかる前に減速動作を行う。これは、ドライバー自身の応答の遅れと車のダイナミクスを考慮に入れ、安全な運転を行おうとしているためと考えられる。移動ロボットの経路制御においても、前章で構成したフィードバックサーボ系は有限の応答遅れをもつことから、進路先の経路の情報を前もって利用する方がコースからの逸脱が少なくなると考えられるし、その結果スリップの原因にもなる過大な操作量を与えなくても済むと予想される。

ドライバーが行うような加減速運転をロボットに行わせるためには、ロボットのスピードに合わせた進路先の経路情報として何を利用するかという問題と、その情報をどのようにして制御に反映させるかという二つの問題に対処する必要がある。本論文では、駆動制御系に対して、ロボットの姿勢角と進行スピードの二つの値を指令する立場をとっているため、経路の情報としてもこれらの値を与えるのが自然のように思われる。経路上の各点でロボットがとるべき姿勢角と進行スピードは経路が与えられると事前に決定することができるので、この意味でも自然な未来目標値と考えられる。

未来目標値を制御に反映させる数理的な手法は予見制御として知られている。ロボットの経路追従制御においても予見制御の考え方が取り入れられるならば、車の運転のときと同じような加減速走行が実現できると予想されるが、オリジナルな予見

制御では、目標値は時間関数として与えられることを前提としており、経路上の位置の関数として与えられる前出の目標値とは異なっている。したがって、経路情報を予見制御の未来目標値として直接使用することはできないが、以下のような対応付けを行うと位置で指定した目標値も利用可能になる。

経路上の点を s とすると、目標値は $y_d(s)$ と表される。現時点におけるロボットの経路上の位置を $s(k)$ 、進行スピードを $V_c[s(k)]$ とすると、ロボットのダイナミクスを無視した近似のもとで、1ステップ未来の経路上の位置は

$$s(k+1)=TV_c[s(k)]$$

と求められる。したがって、1ステップ未来の目標値は

$$y_d[s(k+1)]=y_d(k+1)$$

同様に、 N ステップ未来の目標値も定めることが可能であり、未来目標値時系列が構成できる。

$$y_d(k+1), y_d(k-2), \dots$$

以上のような構成法をとると、ロボットが高速で移動しているときほど遠くの経路情報が未来目標値として生成されることになり、合理的である。

オリジナルなデジタル予見制御[6]では、未来目標値は各成分とも同じ未来区間だけ利用するように定式化されている。ロボットの場合では姿勢角とスピードはそれぞれ異なった未来区間で与える方が現実的であると思われるので、ここでは未来目標値を別々に選択的に利用できるように拡張して利用する。目標値が時変時系列として与えられることに注意すると、1形のサーボ系を構成する際に用いた式(15)は、正しくは次式のように書かれなければならない。

$$\Delta e(k+1)=-\Delta y(k+1)+\Delta y_d(k+1) \quad (27)$$

ここで、姿勢角の目標値と進行スピードの目標値を選択的に区別して扱う都合上、目標値ベクトルを次式のように分けて表す。

$$\Delta y_d(k+1)=g\Delta\phi_d(k+1)+h\Delta V_c(k+1) \quad (28)$$

ただし

$$g=[1 \ 0]^T, \quad h=[0 \ 1]^T$$

簡単のため、例えば進行スピードの未来目標値だけを制御に利用する場合には、式(27)を

$$e(k+1)=e(k)-\Delta y(k+1)+h\Delta V_c(k+1)$$

と書き換え、3.1節と同様のプロセスを繰り返して以下の拡大系を作成する。

$$\zeta(k)=[z(k), \Delta V_c(k+1), \Delta V_{ca}(k+2), \dots, \Delta V_c(k+N)]^T \quad (29)$$

$$\zeta(k+1)=\begin{bmatrix} \Phi_o & \Phi_{12} \\ 0 & \Phi_{22} \end{bmatrix} \zeta(k)+\begin{bmatrix} \Gamma_o \\ 0 \end{bmatrix} \Delta u(k) \quad (30)$$

ここに

$$\Phi_o=\begin{bmatrix} I & -CA \\ 0 & A \end{bmatrix} \in R^{5 \times 5}, \quad \Phi_{12}=\begin{bmatrix} h & 0 \cdots 0 \\ 0 & 0 \cdots 0 \end{bmatrix} \in R^{5 \times N}$$

$$\Phi_{22}=\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & 0 & 1 \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix} \in R^{N \times N}, \quad \Gamma_o=\begin{bmatrix} -CB \\ B \end{bmatrix} \in R^{5 \times 2}$$

ここで、3.1節と同じ対称行列 Q 、 R を与え、次の2次形式

評価関数を最小にする意味で最適な状態フィードバック制御則を求めると

$$P.I. = \sum_{k=-N+1}^{\infty} \left\{ \xi^T(k) \begin{pmatrix} Q & 0 \\ 0 & 0 \end{pmatrix} \xi(k) + \Delta u^T(k) R \Delta u(k) \right\}$$

この関数の最小化問題は、式(30)に対する高次のリカッチ方程式を解かなくてもよく、3.1節で与えた部分系に対するリカッチ方程式の解 P を用いて、拡大系式(31)に対する最適な状態フィードバック入力に次式のように求められる[6].

$$\Delta u(k) = F_e e(k) + F_x \Delta x(k) + \sum_{j=1}^N f_{vj} \Delta V_{Cd}(k+j) \quad (31)$$

ここに

$$f_{vj} = -(R + \Gamma_o^T P \Gamma_o)^{-1} \Gamma_o^T \xi^{j-1} P \begin{bmatrix} h \\ 0 \end{bmatrix}$$

$$\xi = [I - \Gamma_o^T P \Gamma_o]^{-1} \Gamma_o^T P \Phi_o$$

を表わし、状態フィードバック係数 F_e , F_x は3.1節で与えたものと同行列となる。

ロボットの姿勢角に対する未来目標値を利用する場合にも上述したものと同様に議論が適用できるから、拡大系に対する最適な状態フィードバック入力が以下のように求められる。

$$\Delta u(k) = F_e e(k) + F_x(k) \Delta x(k) + \sum_{j=1}^M f_{\phi j} \Delta \phi_d(k+j) \quad (32)$$

ここに

$$f_{\phi j} = -(R + \Gamma_o^T P \Gamma_o)^{-1} \Gamma_o^T \xi^{j-1} P \begin{bmatrix} g \\ 0 \end{bmatrix}$$

以上の結果から、姿勢角について M ステップ未来の目標値までを、また進行速度に関して N ステップ未来までをそれぞれ考慮に入れた場合の制御入力は

$$\Delta u(k) = F_e e(k) + F_x \Delta x(k) + \sum_{j=1}^M f_{\phi j} \Delta \phi_d(k+j) + \sum_{j=1}^N f_{vj} \Delta V_{Cd}(k+j) \quad (33)$$

で構成してよいことがわかる。したがって、式(33)の両辺に演算子 ∇ を作用させ、初期値の補償も考慮に入れた位置形式の制御入力を求めると次式が得られる。

$$u(k) = F_e e(j) + F_x [x(k) - x(0)] + \sum_{j=1}^M f_{\phi j} [\phi_d(k+j) - \phi_j] + \sum_{j=1}^N f_{vj} [V_{Cd}(k+j) - V_{Cj}] \quad (34)$$

ここに、 ϕ_j , V_{Cj} は不定定数であり、目的に合わせて値を決める必要がある。ここでは、理論の厳密性を崩すことになるが、現在の状態に対する未来目標値の変化が強調されるように、これらの値にロボットの現在の姿勢角と進行速度の値を使うことにする。また、少ない予見ステップ数でより早く加減速走行が行えるように、1ステップに進む距離を制御周期そのもので算出するのではなく、適当に長い時間を設定して算出し、制御入力を以下のように構成する。

$$u(k) = F_e \sum_{j=0}^k e(j) + F_x [x(k) - x(0)] + \sum_{j=1}^M f_{\phi j} [\phi_d(k+j) - \phi(k)] + \sum_{j=1}^N f_{vj} [V_{Cd}(k+j) - V_{Cj}(k)] \quad (35)$$

4.2 ロボットの停止制御

これまでの制御では、ロボットを任意経路に沿わせて加減速移動を行わせることは可能であっても、目標とする位置で停止させることはできない。ロボットの運行法としては、同一のアルゴリズムを用いてスタートから停止までを一貫して制御するのが望ましいと考えられる。本文で提案した方法は、経路に沿った速度目標値を生成するために N 制御周期先までの経路の長さを常に監視している。したがって、予見動作を行っている最中に、この経路上に停止点が見いだされた場合には、前もって停止制御にロボットの制御の流れを移すことが可能である。

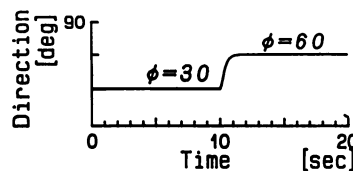
3.3節に示したように仮想的なポテンシャル関数を導入すると、速度指令を与えながら位置の制御も同時に行うことができるので、ロボットの停止制御の場合にも経路上の停止点を極小とするような新たなポテンシャル関数を定義し、経路に沿った速度指令値をこのポテンシャル関数を利用する停止方法が考えられる。その際、予見動作も同時に行うものとするれば、経路に沿った速度ベクトルの計算アルゴリズムを修正するだけでよく、姿勢角とスピードの指令アルゴリズムとしては、式(25)(26)をそのまま用いることができ便利である。

5. 数値シミュレーション実験

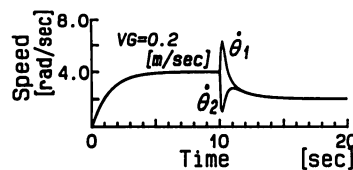
以上の可変速経路追従制御方式の有効性を確認するために行った数値制御実験の結果を以下に述べる。実験で想定したロボットのパラメータは Table 2 に示すようなものである。

Table 2 Parameters of Mobile Robot

$M : 10.0$	[Kg]	$R : 10$	[Ω]
$L : 0.24$	[m]	$K : 0.05$	[Nm/A]
$r : 0.05$	[m]	$J_o : 0.000001$	[Kgm ²]
$I : 0.2$	[Kgm ²]	$J_w : 0.002$	[Kgm ²]
$\mu : 30$			



(a) Step Response of Direction



(b) Step Response of Speed

Fig. 3 Step Response of Servo System

ロボットの動きは位置に関する関係式も考慮に入れた非線形な状態方程式を、時間刻み幅を0.01[sec]にとった4次のルンゲ・クッタ法で計算した。離散時間制御の制御周期 T は0.1[sec]とした。

駆動系に対して設計した姿勢角と進行速度に関する線形サーボ系のステップ応答を Fig. 3 に示す。姿勢角に関しては応答を比較的速く設定したが、進行速度についてはサーボ系の応答遅れが目だつように5秒程度で制定するように設定した。

Fig. 4 は経路に沿わせることを目的として導入した仮想ポテンシャルの有効性を確認するために行った数値実験の結果で、ロボットに対する進行速度の指令値は0.3[m/sec]と一定とし、姿勢角の指令値を式(26)を用いて計算した。仮想ポテンシャル $\psi(\rho)$ のパラメータの値は

$$K_L=0.5, \rho_0=0.2$$

式(24)中の γ_s の値は $\gamma_s=1$ とした。

図1には1秒ごとのロボットの動きを示してあるが、3.3節で明らかにしたように

$$T\gamma_s K_L (=0.01) < 1$$

の条件が満たされているので、経路から離れた位置からロボットをスタートさせても経路に到達し、経路の曲率の大きさなどこゝでも経路に沿って走行できるのが分かる。

Fig. 5 は同じポテンシャル関数を用い、ロボットの経路に沿った速度ベクトル式(23)を

$$V_v(s) = 0.4 \exp\{-2[p''(s)]^2\}$$

で与えたときの走行の様子を描いたものである。スピードに関するサーボ系の遅れのために、車が急カーブに不用意に突入したときのように減速のタイミングが遅れ、カーブの後半で最低スピードになっている。

Fig. 6 は式(35)の予見制御則を用いて走行させたときの結果で、カーブの手前で減速し、カーブの後半で加速するという当初の目的が実現できている。この走行の例では3秒を1ステップあたりの予見の時間幅とし、スピードに関しては3ステップ未来までの目標値を利用している。この時間幅とステップ数は

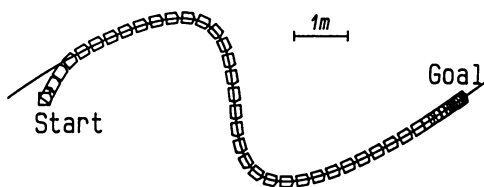


Fig. 4 Constant Speed Path-Tracking Using Feedback Controller

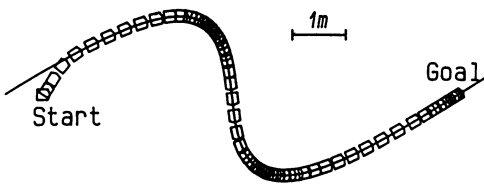


Fig. 5 Variable Speed Path-Tracking Using Feedback Controller

ロボットの走行の状態を観測しながら試行錯誤的に決めたものである。顕著な改善が見られなかったので、姿勢角については予見情報を利用していない。走行中の経路からのずれに関しては Fig. 4, Fig. 5, Fig. 6 の場合とも、最大でも1[cm]程度に納まり、大きな相違は見られなかった。

Fig. 7 は、経路上の各点で、予見情報を利用した走行(Fig. 6 参照)時とフィードバック制御だけによる走行(Fig. 5 参照)時のロボットの進行速度を比較したものである。図中の破線は経路の曲率から求めたスピードの目標値式(36)を表している。予見を行った場合は事前にロボットの進行速度が変化していることが確認できる。

Fig. 8 は走行中のロボットの速度ベクトルの時間変化

$$[V_G(k) - V_G(k-1)]/T$$

ただし、 $|V_G| = V_G$ 、 V_G の方向は姿勢角 α から計算した加速度の大きさを Fig. 4, Fig. 5, Fig. 6 に示した走行について比較したものであり、予見を行わない場合に比べて予見を導入する方が減速時の加速度変化が小さく抑えられている。

以上のように、車を運転するときのような加減速走行を移動ロボットに行わせるという意味では、予見制御を利用した式

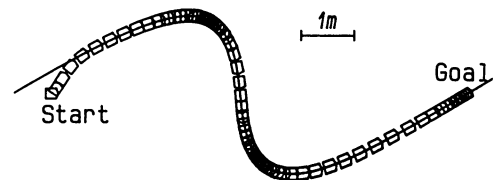


Fig. 6 Variable Speed Path-Tracking Using Proposed Controller

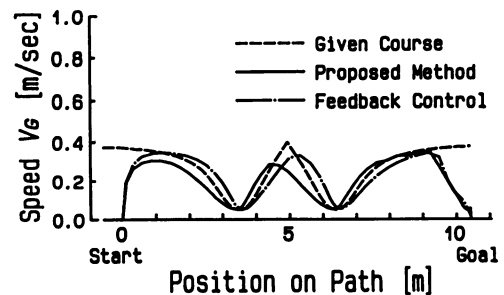


Fig. 7 Speed Profile of Robot along Path

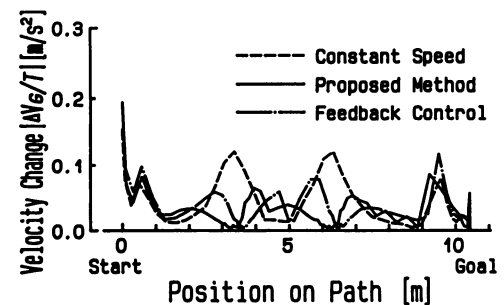


Fig. 8 Magnitude of Acceleration along Path

(35)の制御が有効なことが分かった。しかしながら、ロボットに働く加速度変化の大きさがある範囲に制限するという目的には式(35)の制御則だけでは不完全なことがFig.8から分かり、加速時にはフィードバック制御に切り換える等の工夫が求められる。

なお、ここに示した一連の実験ではロボットの停止制御にも4.2節に示したように仮想的なポテンシャル関数を導入したが、予見動作を導入する場合には未来目標値の与え方をポテンシャル関数が定義されている区間に入るように工夫すると、停止点の近くにポテンシャル関数を定義するだけでも離れた位置から減速がはじまり停止点でロボットは停止する。この意味でも予見動作は有用であると考えられる。

6. おわりに

車輪型の移動ロボットに対し、予見動作を導入して、人が車を運転するときのような加減速運行を行わせる方法について検討し、数値実験を通して本文で提案した方法が有効であることを確認した。

本手法では、車輪の駆動系のダイナミクスも考慮に入れて経路追従制御系を構成しているため、自動搬送車の場合のように、積載する荷の質量が変化したとしてもサーボ系の安定性と上位の位置制御系の安定性が許す範囲内で経路への追従が保証される。マイクロ・コンピュータでの実現を想定して離散時間形式で制御の定式化を行ったので、連続時間制御で近似的なデジタル制御を実現するときのような誤差の影響は受けない。また経路についても、滑らかに変化するものであれば直線や円にとらわれることなく、任意の形状であっても追従が可能である。ただし、経路に関する情報が前もって与えられていない場合には予見が行えないのが弱点であり、この欠点を補うために

は、経路の自動生成法の確立やセンサと情報処理の発展に期待しなければならない。

なお、本手法は基本的には運動場の速度ベクトルが与えられればロボットの指令値が生成できるので、ポテンシャル関数と結び付けた障害物回避[14]と相性がよいと考えられる。これらの分野への応用についても今後検討されるべきと思われる。

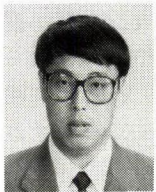
参考文献

- [1] 三平, 伊藤, “非線形制御理論を用いた車両の切り返し車庫入れ制御”, 第6回知能移動ロボットシンポジウム予稿集, 153/158, 1992.
- [2] 中村, 油田, “トレーラ型移動ロボットの軌跡制御”, 日本ロボット学会第6回知能移動ロボットシンポジウム予稿集, 57/61, 1992.
- [3] 飯田, 油田, “独立二動輪操舵型移動車の走行制御”, 日本ロボット学会第8回学術講演会予稿集, 959/962, 1990.
- [4] 早勢, 市川, “目標値の未来値を最適に利用する追従制御”, 計測自動制御学会論文集, vol. 5, no. 1, 86/94, 1969.
- [5] 愛田, 北森, “最適予見サーボ系の設計”, 計測自動制御学会論文集, vol. 22, no. 5, 527/534, 1986.
- [6] 土谷, 江上, “デジタル予見制御”, 1993, 産業図書.
- [7] 飯田, 油田, “車輪型移動ロボットのための走行制御コマンド系と軌跡制御方式”, 日本ロボット学会第1回ロボットシンポジウム予稿集, 85/90, 1991.
- [8] A. Sommerfeld, “ゾンマーフェルト理論物理学講座I力学(高橋訳)”, 190/216, 1972, 講談社.
- [9] 古田, 原, “サーボ技術と現代制御論”, 計測と制御, vol. 19, 953/961, 1980.
- [10] F. Miyazaki, S. Arimoro, “Sensory feedback for robot manipulators,” J. of Robotic Systems, vol. 2, 53/71, 1985.
- [11] 浅沼, “電子コンパスの製作”, トランジスタ技術 SPECIAL no. 34, 134/147, 1992, CQ 出版.
- [12] 文献[6]21/24.
- [13] 美多, “デジタル制御理論”, 第7章, 昭晃堂, 1984.
- [14] 佐藤, “ラプラスポテンシャル場を利用したロボットの動作計画”, 第6回知能移動ロボットシンポジウム予稿集, 117/122, 1992.



橋本幸男 (Yukio Hashimoto)

1953年3月8日生。1979年室蘭工業大学大学院修士課程電気工学専攻修了。高校の非常勤講師等を経て83年室蘭工業大学電子工学科助手。90年より同大学電気電子工学科助手。制御工学, メカトロニクスの研究に従事。計測自動制御学会の会員。(日本ロボット学会正会員)



王 碩玉 (Wang Shou Yu)

1962年2月19日生。1983年中国沈陽工業大学電気工学科卒業。85年同大学大学院自動制御専攻修了。89年来日, 93年北海道大学大学院博士課程修了。93年4月より山形大学電子情報工学科助手。ロボットマニピュレータの制御に関心をもつ。工学博士。(日本ロボット学会正会員)



土谷武士 (Takeshi Tsuchiya)

1941年2月7日生。1965年北海道大学大学院修士課程電気工学専攻修了。北海道大学工学部電気工学科講師, 助教授を経て現在教授。制御工学, ロボット工学, パワーエレクトロニクスなどの研究に従事。工学博士。計測自動制御学会, システム制御情報学会, 日本機械学会, 日本ファジィ学会, IEEEの会員。(日本ロボット学会正会員)



松田敏彦 (Toshihiko Matsuda)

1934年1月20日生。1956年室蘭工業大学電気工学科卒業。北海道大学工学部助手, 室蘭工業大学講師, 助教授を経て78年同大学電気工学科教授。現在同大学電気電子工学科教授。工学博士, 電気機器, 電動機の制御に関する研究に従事。電気学会などの会員。(日本ロボット学会正会員)