技術論文

屋内環境を自律的にナビゲーションする平行二輪倒立振り子型自立移動ロボットの機能分散アーキテクチャによる実現

河 潤 秀^{*1} 油 田 信 一^{*2}

Functionally Distributed Realization of a Wheeled Inverse Pendulum Type Self-Contained Mobile Robot which Navigates Autonomously in Indoor Environment

Yun-Su Ha^{*1} and Shin'ichi Yuta^{*2}

The purpose of this work is to develop a wheeled inverse pendulum type mobile robot which can autonomously navigate in a two dimensional plane while keeping its own balance. In this paper, we report the construction and control algorithm of the wheeled inverse pendulum type self-contained mobile robot "Yamabico Kurara". Yamabico Kurara has a functionally distributed architecture. The navigation program is located in the behavior control level and is implemented on the master module which takes the part of total control. Posture and trajectory tracking control are realized on the locomotion module which is one of the functional modules at the system level. The results of locomotion control experiments and navigation in the indoor environment of the robot are also shown. Discussion on the system integration and feasibility of the application is also included.

Key Words: Wheeled Inverse Pendulum Type Mobile Robot, Functionally Distributed Architecture, System Integration, Indoor Navigation, Autonomous Robot

1. ま え が き

平行二輪型移動体は、それ自体が不安定なシステムである。この種のシステムを安定性を保ちつつ目的地に向かって走行させることは、制御アルゴリズムやコントローラの構成などの面からロボット工学上極めて興味深いテーマである。また、平行二輪車として移動体を制御することは、車輪の揺動をコントロールすることであり、高度な AGV の制御や自動車の乗り心地の向上にもつながる実用的応用可能性を有している。

従来より、山藤、松本らはこの問題について精力的な研究を行い、平行二輪型移動ロボットの直立姿勢安定化および平面走行を実現している [1]~[4]。筆者らも自立移動体が倒立制御しながら、かつ、ある程度複雑な環境内を自律的にナビゲーションするというシステムの開発を目指して、倒立振り子型自立移動体のナビゲーションのための軌跡制御などの研究を進めてきた [5]~[7]。本稿では、筆者らが従来より研究を進めている機能分散アーキテクチャ [8] に従って構築した、平行二輪型自立移動ロボットのシステムについて述べる。さらに、それが大学の実験室のなかなどの環境で自分自身の姿勢の安定性を保ちな

Fig. 1 Inverse pendulum type self-contained mobile robot Yamabico KURARA

がら自律的にナビゲーションするためのアルゴリズムを示し、実験結果を報告する。

原稿受付 1995 年 2 月 12 日

^{*1}筑波大学大学院

^{*2}筑波大学電子情報工学系

^{*1}Graduate School, University of Tsukuba

^{*2}Institute of Information Sciences and Electronics, University of Tsukuba

Table 1 Parameter of the vehicle

Size L × W × H	270 × 340 × 470 [mm]
Weight of body	9.01 [kg]
Weight of wheel and axle	0.51 [kg]
Radius of the wheel	62 [mm]
Distance between left and right wheels	340 [mm]
Reduction ratio of gear	1/39.5

2. 自立型平行二輪移動ロボット「山彦くらら」

2.1 システムの概要

筆者らが開発し、実験を進めている平行二輪型自立移動ロボットを Fig. 1 に示す。このシステムは、筆者らのプロジェクトで開発し実験に用いている「山彦」[10]をベースとしており、「山彦くらら」と名付けられている。「山彦くらら」は重量 10.2 [kg] の小型の移動ロボットであるが、姿勢の安定とナビゲーションに必要な、走行駆動系、センサ、情報処理のためのコンピュータ、バッテリーなどすべての機能を自分自身の筐体内に内蔵している自立型である。メカニズム上の最大の特長は自分自身を支えるための車輪が本体の左右にある 2 個の動輪のみであることで、このシステムは制御なしでは立っていることのできない不安定システム、すなわち、CCV (Control Configured Vehicle) である。これを安定化し、かつ、目的地に向かって自律的に走行させるために、マルチプロセッサによる機能分散コントローラ [8] を用いた。以下本章では、このシステムについて述べる。

2.2 ハードウェア：ボディ・駆動系・センサ

「山彦くらら」のボディは筆者らが開発して自立移動ロボットの研究に利用している「山彦」から前後のキャスタを取り外したものである。したがって、「山彦くらら」はセンサ、情報処理機構、駆動系のほか、バッテリーに至るまで、ナビゲーションに必要なすべての機能を自分自身の筐体内に内蔵した自立型である。

このロボットは本体の左右に直径 124 [mm] の動輪を有し、これらはおおの 10 [W] の DC モータ (MAXON 社製) に減速ギヤ (39.5: 1) を介して接続されている。DC モータには 1 周 500 パルスのエンコーダが取り付けられている。各 DC モータは後述するコントローラで発生させる PWM 信号により制御されるドライバ回路 (スイッチングトランジスタ) で駆動される。「山彦くらら」は、本体の姿勢角を知るために、角速度センサである振動ジャイロを有している。このジャイロにはトキメック製の TFG-160 を利用した。TFG-160 は重量は 130 [g] であるが、帯域 50 [Hz]、最大角速度 200 [deg/s] の性能を有し、この種の倒立振子の姿勢安定化には十分な性能を有している。ナビゲーションのためのセンサとしては前、左右の 3 方向に超音波距離センサを有する。このセンサは村田製作所の 40 [KHz] のトランスデューサを用い、後述するコントローラによって約 3 [m] 以内の反射物体までの距離を測ることができる。「山彦くらら」の外観図を Fig. 2 に、移動体の重量などのパラメータを Table 1 に示す。なお、バッテリーは 12 [V] 7000 [mAh] のシールド鉛バッテリーを用い、これによって約 6 時間連続して働かせることができる。

Fig. 2 Configuration of the Yamabico Kurara

2.3 コントローラの構成

「山彦くらら」のコントローラは「山彦」で従来より用いられている機能分散マルチプロセッサシステムにより実現されている。ここでは、コントローラはシステム全体の動きを統括制御する行動レベルと、モータ制御やセンサ信号処理を行う機能レベルに分けられる。機能レベルにはモータを制御してロボットの姿勢制御をしつつ行動レベルの指示に従って走行軌跡の制御を行う姿勢走行制御部や、超音波センサを用いて環境中のまわりの反射物体までの距離を測定しその情報を行動レベルに伝える超音波レンジファインダ部、さらに、行動レベルからのコマンドに従って音声を発生させる音声合成部がある。これらの各部分は機能ごとにモジュール化されており、おのおの、マスタモジュール (行動レベル) 走行モジュール (機能レベル、以下同じ) 超音波センサモジュール、音声合成モジュールと名付けられている。機能レベルの各モジュールはマスタモジュールと双方向メモリによって接続され、全体としてマスタモジュールを中心とした星型接続のマルチプロセッサシステムを構成している。またこれらの各モジュールは独立の CPU (MC68000) を有し、おのおの 1 枚の回路基板上に実現されている。Fig. 3 に「山彦くらら」のコントローラハードウェア構成図を示す。

コントローラが従来の標準「山彦」とハードウェア的に異なるのは走行モジュールにジャイロインタフェースが付加されたことである。

2.4 ソフトウェアの構成

「山彦くらら」のコントローラには、従来の「山彦」と同じく、各モジュールごとに筆者らのプロジェクトで開発したりリアルタイムマルチタスクのオペレーティングシステム MOSRA が搭載されており、各機能を実現するプログラムはその上の一つのプロセスとして与えられている [10]. 「山彦くらら」の制御プログラムは、全体を統括しナビゲーションレベルの制御を行うマスタモジュール上のプログラムと、各機能モジュールにおかれてそのおのおの機能を実現するプログラムに大別される。ここでマスタモジュール上のプログラムは走行モジュールや音声合成モジュールへ“コマンド”によって指示を行い、また各

Fig. 3 Hardware structure of the controller

機能モジュールが検出したロボットの状態は“状態表示パネル”という通信用の共有メモリによってマスタに伝えられる [8][9].

「山彦くらら」の機能レベルのモジュールのうち超音波センサモジュールと音声合成モジュール上のプログラムは標準の「山彦」と同一のものをを用いているが、走行制御モジュール上のプログラムは姿勢制御を行う機能を追加するため新規に開発された。ソフトウェアの構成を Fig. 4 に示す。

「山彦くらら」のナビゲーションレベルのプログラムは、超音波センサモジュールによって得られる環境情報や走行モジュールが管理するロボットの位置情報を時々刻々監視しつつ、与えられた地図情報を参照して、必要なコマンドを走行モジュールに送ることによりロボット全体の動きを制御する。走行モジュール上のプログラムは5ミリ秒をサンプリングレートとするフィードバックループを基本として作られている。このプログラムではマスタから指示されたコマンドを解析し、それとデッドレコニングによって推定した自分の位置情報に基づいて走行軌跡を決める。また、ジャイロセンサから得られる本体の傾き角速度とその積分値およびエンコーダセンサから得られる車輪の回転角速度を用いてロボットの姿勢を安定化させながら決められた軌跡上を走行させるための制御を行う。

3. 姿勢・走行制御のアルゴリズムとその実現

本章では、ロボットが自分自身の姿勢の安定を保ちつつ、マスタモジュールのナビゲーションプログラムより送られた走行制御コマンドに基づいて二次元平面上を走行させるための制御アルゴリズムと、実現したシステムについて述べる。

3.1 制御の戦略

ロボットに自分自身の姿勢の安定を保ちつつ、指示に従って二次元平面上を走行させるためには

- (a) ロボットの姿勢（バランス）の制御
- (b) ロボットの並進速度と位置の制御
- (c) ロボットの中心まわりの回転速度と方位の制御

SIMP: State Information Monitoring Panel

Fig. 4 Structure of software for posture and locomotion control

を同時に行う必要がある。従来より、この種のノンホロノミックな移動体の制御方法として (b) と (c) を分離して制御することが行われてきた [11]. 倒立振り子の移動体においては、これらの制御はいずれも (a) のロボットの姿勢（バランス）制御と関連する。しかし、3.3 節に述べるとおり (c) のロボットの方位の制御で大きな角速度を与えない限り、それがロボットの姿勢制御 (a) に与える影響は小さい。したがって、「山彦くらら」では (a) と (b) を組み合わせた姿勢・速度制御と (c) のロボットの方位の制御を分離して行うことにした。この方針で設計した姿勢と走行の制御システムの構成を Fig. 5 に示す。この制御システムでは各サンプル時間ごとに以下の処理を行う。

Fig. 5 Construction of posture and locomotion control system

- (1) 指示された軌跡とロボットの二次元平面上の現在の位置・方位に基づいて、ロボットの走行速度と回転速度のリファレンス値を発生する。(軌跡コントローラ)
- (2) ロボットの並進速度と回転角速度のリファレンス値より左右の動輪の回転速度のリファレンス値を計算する。(PWSコントローラ)
- (3) ロボットの左右の動輪について、姿勢の安定を保ちつつ、おののをリファレンス回転角速度に追従させるために、DCモータに与える電流を制御することによりトルクを制御する。(姿勢・速度コントローラ)

3.2 姿勢・速度制御

ロボットに目標走行速度が与えられたとき、ロボットが姿勢のバランスを保ちながらその速度で走行する制御系を、車輪型倒立振り子の姿勢・速度制御系として設計した[6][7].

ここでは、まず、移動体を一次元の車輪型倒立振り子モデルとして解析したうえ、直立状態近傍で線形近似し、状態方程式(1)を得た。

$$\dot{X} = AX + Bu \quad (1)$$

ここで、

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 38.23 & -0.085 & 0.121 \\ -51.76 & 0.347 & -0.56 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ -13.7 \\ 56.07 \end{pmatrix}$$

$$X = [\phi \ \dot{\phi} \ \ddot{\phi}]^T$$

$\phi, \dot{\phi}$ はジャイロセンサより得られる本体の傾きと角速度を、 $\dot{\theta}$ はロータリエンコーダセンサから得られる車輪の回転角速度を示す。この線形近似モデルに基づいて一次元倒立振り子に姿勢の安定を保ちつつ与えられた目標速度で走行させるトラッキング問題として扱うこととし、最適レギュレータ理論を適用した。なおここで、ジャイロセンサのドリフトやモデリング誤差に対処するため制御系は一つの積分器を持つ最適追従制御系として構成した。その構成を Fig. 6 に示す。ここで τ_s は与えられた車輪のリファレンス回転角速度 ($\dot{\theta}_{ref}$) に対する定常状態でのモータが発生すべきトルクであり、 ϕ_s はそのときロボットが保つべきの姿勢である。 α_1, α_2 はいずれもモデルから求めることができる。「山彦くらら」では車輪の目標回転角速度を階段状で変化させると仮定し、式(1)から $\alpha_1 = 0.000248, \alpha_2 = 0.000623$ を得た。左右の車輪について、おのの独立に制御できるものと仮定すると、一定のリファレンス回転角速度を追従するため

Fig. 6 Posture and velocity control system for one dimensional wheeled inverse pendulum

Fig. 7 PWS control system combined with a posture and velocity control

の左右駆動モータが出すべきトルク $\tau_{rrref}, \tau_{lrref}$ は次のように決められる。

$$\tau_{rrref} = \tau_{rs} - \tilde{k}_{11}(\phi - \phi_{rs}) - \tilde{k}_{12}\dot{\phi} - \tilde{k}_{13}(\dot{\theta}_r - \dot{\theta}_{rrref}) - \tilde{k}_2 \int_0^t (\dot{\theta}_r - \dot{\theta}_{rrref}) dt \quad (2)$$

$$\tau_{lrref} = \tau_{ls} - \tilde{k}_{11}(\phi - \phi_{ls}) - \tilde{k}_{12}\dot{\phi} - \tilde{k}_{13}(\dot{\theta}_l - \dot{\theta}_{lrref}) - \tilde{k}_2 \int_0^t (\dot{\theta}_l - \dot{\theta}_{lrref}) dt \quad (3)$$

ここで、添字 r, l は右と左モータに関係するパラメータであることを示す。 $\tilde{k}_{11}, \tilde{k}_{12}, \tilde{k}_{13}, \tilde{k}_2$ は最適レギュレータ理論を用いて決定したフィードバック係数である。「山彦くらら」では $[\tilde{k}_{11}, \tilde{k}_{12}, \tilde{k}_{13}, \tilde{k}_2] = [-382.9, -67.23, -5.23, -4.89]$ と決めた。

「山彦くらら」の車輪の駆動用 DC モータの発生トルクはモータに流れる電流に比例するのでモータに流れる電流で制御できる。ここで、モータの電流制御は [12] のフィードフォワード方式を用いた。

3.3 操舵制御

平行二輪倒立振り型移動ロボットが旋回運動する際、厳密には一次元の倒立振り子モデルとして扱うことはできない。しかし、

Fig. 10 Experimental results of posture and velocity control

Fig. 12 Experimental results on the sloped rough-ground

Fig. 11 Experimental scene on the sloped rough-ground

で与える。ここで k_η , k_α はこの系を安定化させるためのフィードバック係数であり、「山彦くらら」では $k_\eta=0.004577$, $k_\alpha=1$ としている。

3.6 デッドレコニング

前節の軌跡制御を実現するためには、ロボットは常時自分の位置が分かっている必要がある。そのために「山彦くらら」の姿勢走行制御部には、左右の車輪の回転から自分の位置と方位を常時推定しているデッドレコニング機能を実装している。このデッドレコニングによる推定位置情報は状態表示パネルによって走行軌跡制御にのみでなく、マスタモジュールのナビゲーションレベルのプログラムでも使われる。

一般にデッドレコニングによる位置推定は累積誤差が避けられない。しかも「山彦くらら」の姿勢走行制御部にはそれを修正する機能はない。したがって、累積誤差の修正はナビゲー

ションレベルのプログラムが、正確に位置情報を得たときに走行制御の推定位置をリセットすることによって行う。

3.7 姿勢の安定化と走行制御の評価実験とその結果

本節では上に述べた姿勢制御と走行制御の安定性と性能を示すために「山彦くらら」上で行った評価実験の結果を示す。

(1) 目標速度を急激に変化させたときの姿勢の安定性と速度追従性能実験

姿勢制御の安定性と走行制御の性能を確かめるため、姿勢のバランスを保ちながら停止している「山彦くらら」の両輪の姿勢速度制御部に直接ステップ状の目標速度 20 [cm/s] を与えたときの振舞いを Fig. 10 に示す。本体の角度は一瞬は 3° 程度まで達するが 2 秒程度で安定していることが分かる。また速度は若干オーバーシュートするが、これも 3 秒程度で目標速度で安定している。

(2) 傾斜のある不整地での姿勢の安定性と走行性能

地面の状態の影響を確かめるため、大学内の 10% の傾きを有するタイル貼りの通路を走行させた。その環境と実験場面を Fig. 11 に、そのときの振舞いを Fig. 12 に示す。ここでは、両輪の目標速度を 20 [cm/s] とし、時刻 0 で止めている状態からスタートさせた。実験結果で走行速度は振動が見られるのは「山彦くらら」のギアのバックラッシュの影響か、ボディの共振によると考えられるが、全体としては地面の凸凹の影響をうけつつも安定して走行できた。Fig. 12 より地面の傾きの中で安定化するため、定常時にも本体が自動的に傾いているのが分かる。

(3) 走行経路に小さな段差がある場合の姿勢安定化と走行性能

定常走行中、地面に小さな段差があった場合の影響を知るため、Fig. 13 に示すように約 8 [mm] の段差を作り「山彦くらら」が 25 [cm/s] で走行中にそれに直接当たって登るときの振舞いを記録した。この実験は必ずしも成功するとは限らなかったが、成功した一例の結果を Fig. 14 に示す。このデータはロボットの車輪直径の 5% を越える段差でも姿勢の安定を保ったまま越え得ることを示している。

Fig. 13 An experiment to climb up a small step

Fig. 14 The results of experiment to climb up a small step

(4) 目標直線軌跡追従性能

Fig. 15, Fig. 16 は「山彦くらら」による軌跡追従制御の実験結果である。Fig. 15 は x - y 二次元平面上の原点を出発し、 x 軸上を 25 [cm/s] で走行しているロボットに x - y 座標上の位置 (70, 0) と (100, 0) に到達したときに、太い実線で示す、それまでの直線に直交する目標直線を与えたときの結果を示す。実線で示したのが (100, 0) 目標直線を与えたときの追従軌跡であり、破線で示したのが (70, 0) で目標直線を与えたときのロボットの軌跡である。

Fig. 16 は x 軸上を 25 [cm/s] の速度で走行しているロボットが (50, 0) の位置に来たとき、それまでの直線に平行な目標直線を与えたときの実験結果を示す。これらの実験結果は「山彦くらら」がどのような位置からでも自動的に操舵を行って、与えられた直線に追従して行くことを示しており、二次元平面上で安定して軌跡が制御されていることが分かる。

3.8 走行制御コマンド

移動ロボットのナビゲーションを行うためには、ナビゲーションプログラムから姿勢走行制御部に上記に述べた目標軌跡追従制御や走行速度や停止指示を与える必要がある。「山彦くらら」の姿勢走行制御部にも、その動きを指示するための言語としてのコマンド系を定義し、実装した。「山彦くらら」のコマンド系は筆者らのグループですでに 4 輪のロボットのために設計した Spur [11] にわずかな修正を加えたもので HYS コマンド系と名付けられている。

Fig. 15 Experimental results of linear trajectory tracking control

Fig. 16 Experimental results of linear trajectory tracking control

ここでは、その代表的なコマンドについて、ナビゲーションレベルのプログラムがそのコマンドを発行するための関数について若干の説明を加える。

(1) 直線軌跡指示コマンド：HYS.line.LC(x, y, ψ)

X - Y 平面上で (x, y) を通り x 軸 ψ の傾きを有する直線を追従するよう走行制御を行う。

(2) 停止コマンド：HYS.stop()

この関数が発行されると指示された減速度でなるべく速やかに停止する。

(3) 推定位置修正コマンド: $HYS_adj_pos_LC(x, y, \psi)$

姿勢走行制御部は常時デッドレコニングによって自律的に現在の位置を推定しているが、その推定位置と方位を (x, y, ψ) におきかえる。この位置修正コマンドはそれ自体が走行軌跡を与えるものではない。しかし、直線軌跡指示コマンドに従って走行中にこのコマンドが発行されると、ロボットが推定している位置がコマンド発行前後で変わるため、他の直線への乗り換えたと同様な走行軌跡を描くことになる。

(4) 標準走行速度、回転角速度の設定コマンド:

$HYS_set_vel(v)$, $HYS_set_angv(\psi)$

標準走行速度と標準回転角速度を設定する。

4. ナビゲーションのアルゴリズムと実現

本章では「山彦くらら」上に実現した屋内ナビゲーションの方法を述べる。このナビゲーションレベルのプログラムはコントローラのマスタモジュール上に搭載されている。

4.1 ナビゲーションの戦略

「山彦くらら」の屋内ナビゲーションはデッドレコニングと超音波センサによる現在位置推定に基づく位置ベーストな方式で実現した [13][15]。

いまロボットには、あらかじめプランニングして走行すべき経路を直線列で与えることとする。「山彦くらら」はデッドレコニング機能を有するので、その推定値が正しい限り、上述の直線軌跡追従アルゴリズムによって、指示された経路を走行することができる。したがって実環境内ナビゲーションのために必要な技術は

- (1) 経路上に存在する予定にない障害物への対処
 - (2) デッドレコニングによる推定位置に含まれる累積誤差の解消
- の二つである。

(1) については、ロボットはナビゲーション中常時前方を監視し、もし、予定経路上の前方 70 [cm] 以内に障害物があったら停止してそれが撤去されるまで待つことにする。一方、(2) については、超音波センサによる走行路の側面の壁までの距離を環境地図との照合によってデッドレコニングによる推定の誤差を修正することとした。また、経路上で実際の環境のなかで壁までの距離を測定して地図との誤差を求める修正計算を行う場所はあらかじめプランニング時点で決め、その地点は地図に記載しておくこととした。

4.2 地図の表現と現在位置情報の修正方法

ロボットが、例えば、Fig. 17 の環境内で S 点にあり、G 点まで経路 P を通って走行させるとする。そのとき経路 P は、各直線区間の長さとし、各曲点における曲り角度の列によって表現できる。

さて、ロボットの走行距離そのものは車輪の回転からある程度の精度で求めることができる。しかし、走行のとき重要なのはロボットの向いている方向の測定である。そこで、経路中の所々でロボットはその進行方向の左か右の壁までの距離を測定することとし、その測定値と比較すべき値を地図から求めてその差によりロボットの進行横方向推定現在位置を修正する。さらに、同じ直線経路上で 2 回以上の修正を行うときは、横方向

Fig. 17 An example of the environment

```
(START)
(200, TURN_LEFT, 90)
((150, MEASURE_LEFT, 100)(300, MEASURE_LEFT, 100)(400, TURN_RIGHT, 90))
((200, MEASURE_RIGHT, 60)(400, MEASURE_RIGHT, 60)(600, TURN_LEFT, 90))
(200, STOP)
```

Fig. 18 An example of map representation

の位置のみでなく、方位についても修正する。この修正ではロボットの進行方向についての推定位置の修正はできないが、この方向の位置誤差はロボットがやがて方向を変えた後に横方向の位置として修正されることになる。

いま、Fig. 17 の環境中ではロボットは走行中に壁までの距離を測定でき、それによって、進行方向に垂直な方向の現在位置の推定値を修正することができる。そこで、経路の計画時に測定点を決めて、例えば Fig. 17 中の M1 と M2 点付近で左側の壁までの距離を測定し、M3 と M4 点付近で右側の壁までの距離を測定させることにする。このとき経路の情報としての地図には曲り角の情報のほか測定点の情報を付け加えて記述することにする。この表現を経路地図と呼ぶ。Fig. 17 における経路地図を Fig. 18 に示す。

この地図はロボットが走行するとき、START また TURN 後の走行距離が括弧の左側の値に達したら、その後に記述されている動作を行うことを求めている。すなわち、その動作が MEASURE であれば測定を行い、測定結果を右側の値と比較して自分の推定位置を修正させる。また動作が TURN であるときはロボットが追従すべき軌跡を変更させる。このように表現させた地図によってロボットは自己位置を修正しつつ与えられた経路に従って目的地に向かうことができる。

4.3 ナビゲーションプログラム

「山彦くらら」では直線軌跡追従コマンドにより走行すべき直線を指示することができるほか、位置修正コマンドにより現在位置推定誤差を正すことができる。「山彦くらら」のナビゲーションレベルのアルゴリズムはこれらのコマンドを用いること

Fig. 19 Planned path and environment model

を前提として次のように決めた。

- (1) START
- (2) 進行すべき方向に向けて直線軌跡追従コマンドを発行する。
- (3) 経路地図より次の一本の直線経路を取り出す。
 - (a) 常時前方を監視し、もし障害物があったら停止し、障害物がなくなるのを待って走行を再開する。
 - (b) その直線経路上の位置修正点に来たら超音波センサの値と地図の値を比較し、そのずれを位置修正コマンドで走行系に指示する。
- (4) その直線経路のおわり近くに来たら（直線経路の長さだけ走行したら）、停止コマンドを発行して終了、または、地図に従って右か左に向かう直線軌跡指示コマンドを発行し、その方向に直進させ (3) に戻る。

このナビゲーションアルゴリズムを筆者らの開発した言語 ROBOL/0 [10][14] で記述し、「山彦くらら」のマスタモジュールに搭載した。

4.4 ナビゲーション実験

筆者らの大学の研究室内にいくつかの走行経路を決め、それに基づく経路地図を作成し、上記のナビゲーションプログラムに与えることにより「山彦くらら」に自律的にナビゲーション行わせた。その中の一つはさらに音声合成を組み込み、来客に研究室を案内する走行距離 63.2 メートル、実行時間約 5 分のデモンストレーションとした。

Fig. 19 は筆者らの研究室の地図であり、研究室内デモンストレーションプログラムの経路と修正点を示す。

Fig. 20 は「山彦くらら」が実際に研究室内を走行している様子である。今までの経験では、ロボットの姿勢安定性は常に確保され、99%以上の成功率で転倒することなく常に与えられた経路を走行することができている。この「山彦くらら」のデモ

Fig. 20 Experiment of indoor navigation

ンストレーションプログラムはいつでも働かせることが可能であり、大学へ来客があったときなどには常に活躍している。

5. 評価と検討

5.1 システムのインテグレーションについて

—機能分散アーキテクチャの効用—

本研究の特長的な成果は、システムインテグレーションにある。本稿で報告した平行二輪型自律移動ロボットのハードウェアおよびアルゴリズムなど個々部分は、すでに筆者らのプロジェクトで開発されたものや、ほかから発表されているものが少なくない。すなわち、車輪型倒立振子の安定化制御方式にはすでにいくつもの報告があり、また、軌跡追従のための制御方式と走行制御コマンドシステムについては原則的に筆者らのグループがすでに車輪型移動ロボットについて発表している方式を準用している。さらに、屋内での超音波センサと地図を用いる位置情報に基づくナビゲーションも、筆者らを含むいくつかのグループがすでに開発し、発表している。しかし、一般に、それらの機能のすべてをまとめて一台の自立型のロボットの中に集約することは容易ではない。これを比較的容易にして、システム全体のインテグレーションを可能にしたのは、第2章に述べたコントローラの機能分散型モジュール化アーキテクチャである。このコントローラのモジュール化は筆者らが従来より主張してきたものであるが、本研究によってその有効性がさらに検証されたと考えている。

本システムは、基本的には、筆者らがすでに有していた標準的「山彦」システムのうち、走行制御系の中の比較的下位レベルの部分である、走行速度制御レベルとモータ制御レベルの間に、姿勢安定化制御機能を組み込むことによって実現された。これは「山彦」の機能分散型アーキテクチャのフレキシビリティを示したものである。ここでは、「山彦」の超音波センサモジュールや、音声合成モジュール、あるいは、オペレータとのインタフェースなどは、単にユーザとして利用した。これらの機能は本研究においても極めて有用であった。なお、本アーキ

テクチャが有効であるのは、本研究のように新しいプロトタイプを作ろうとする場合である。機能やそのためのアルゴリズムが完全に確定した後は、単一の CPU にするコントローラとすることも容易であり、その方がコスト的には有利になる。

5.2 制御の応答速度または時定数と制御系の階層化

倒立振子型の移動ロボットに実環境内をナビゲーションさせるためには、モータの制御部から与えられた環境内で目的地に向かって歩を進めて行く部分までの、いくつものタスクが並列に実行されねばならない。それらはおのおの、以下のとおり、要求される応答速度、あるいは、タスクの時定数ともいべき数値を有している。

- (1) DC モータの電流制御：0.5 ミリ秒のオーダー
- (2) 姿勢制御：50 ミリ秒のオーダー
- (3) デッドレコニングによる現在位置推定：0.1 秒のオーダー
- (4) 走行速度制御：0.5 秒のオーダー
- (5) 障害物に対応する動作：0.5 秒のオーダー
- (6) 地図に従って道を選んでいく制御：数十秒のオーダー
- (7) 経路計画：数分のオーダー

本システムでは、これらのうち (1) はハードウェアにより実現され、(2)～(4) は走行制御モジュール、(5)～(6) はマスタモジュール上のプログラムにより実現した。(7) は本システム上では今のところ実現されていない。

これらの応答速度（時定数）を実現するために、(2)～(4) のタスクは走行制御モジュール上ですべて 5 ミリ秒をサンプリング時間とする制御系で実現した。また (5)～(6) のタスクは、マスタモジュール上での状態監視によるイベントドリブンで実行されるものとした。すなわち、本システムは、制御系として見ると、連続時間制御として行うタスクを走行モジュール上に、イベントドリブン制御を行うタスクをマスタモジュール上に置いた、2 段階の階層制御系となっている。本アーキテクチャでは、これらの 2 種の制御系の間の概念的通信方法として走行経路などを指示するコマンド（マスタ → 走行モジュール）と走行系の状態変数を示す状態表示パネル（走行モジュール → マスタ）を定義している。ここで、ナビゲーション実行時には、コマンドは数十秒に一度程度発行させるだけであるが、状態表示パネルは連続時間として扱うためのサンプリングレートで書き込まれ、ほぼ同時レートでモニタさせている。

このように、多くのタスクをコントローラ上で 2 階層に分けて実現し、その間の通信の形式を方向を分けてあらかじめ定義していることもシステムインテグレーションを実現しやすくなった理由と考えている。

5.3 車輪型倒立振子ロボットの応用可能性と問題点

本システムは、すでに、筆者らの研究室のみではなく他の何ヶ所かへも持参して、デモンストレーションを行っている。今までの実験とデモンストレーションの結果、倒立振子の姿勢を安定化することに関する技術的問題点は、ジャイロのドリフトの問題や地面の多少の凸凹への対応なども含めて、ほとんど解決したと考える。

さらに、「山彦くらら」は、倒立振子として制御することにより従来のキャスト付きの 4 輪車に比べ、むしろ、直線走行時のふらつきが減少し走行性能が向上した。それは、キャストに加

わる力によって方向制御が外乱を受けることがなく、また姿勢そのものを積極的に制御することによって加減速による本体の揺れが減少したためであろう。この性質については無人搬送車をはじめとする平面上で動く多くの移動体に対しても期待できるものであり、今後さらに詳しく検討を加えたい。

一方、倒立振子ロボットが制御中に転ぶなどの事故が発生するのは、ナビゲーション中に経路からはずれて道を踏み外したり、障害物に衝突した時などに発生する。これはいわゆるナビゲーションにおける問題点に起因するものであり、今後、よりロバストなナビゲーションの技術の確立と、衝突などの事故が発生した際に被害を最小限に食い止めるような制御系の検討が重要と考えられる。

6. む す び

本稿では、筆者らが開発した平行二輪倒立振子型自立移動ロボットについて、その制御アルゴリズムからそれに実環境内をナビゲーションさせるための方式までを述べ、インプリメンテーションと実験結果を示した。本システムのコントローラは筆者らが従来より採用している機能分散アーキテクチャに基づいて構成されており、比較的容易にシステムのインテグレーションが実現された。

今後さらに、システムのロバスト性と信頼性を追求し、また転倒などの事故が生じたときの対処についても検討して行きたい。

謝 辞

本システムは、筆者らのプロジェクトの長年の研究成果の上に構築されたものである。今までプロジェクトに参加された各位、特に走行制御系の実現について、飯田重喜氏（現陸上自衛隊）と小柳栄次氏（神奈川県立横須賀工業高校）、制御系全体のアーキテクチャについて、飯島純一氏（明星大学）と木元克美氏（筑波大学大学院）、ナビゲーションレベルのプログラムについて飯田氏のほか、鈴木昭二氏（大阪大学）、永谷圭司氏（筑波大学大学院）、音声合成について前山祥一氏（筑波大学大学院）に感謝の意を表す。また、倒立振子の制御のために振動ジャイロを貸与頂いている（株）トキメックに感謝する。

参 考 文 献

- [1] 山藤、河村：“同軸二輪車の姿勢制御”，JRSC vol.7 no.4, pp.74-79, 1989.
- [2] 山藤、宮川、河村：“平行二輪車の同調および操舵制御”，日本機械学会論文集（C 編）vol.55 no.513, pp.1229-1234, 1989.
- [3] 山藤、越山、三沢、奥田：“可変構造型平行二輪車の姿勢および走行制御”，日本機械学会論文集（C 編）vol.59 no.564, pp.2368-2375, 1993.
- [4] 松本治、梶田秀司、谷和男：“適応オブザーバを用いた車輪型倒立振子の姿勢推定”，第 9 回日本ロボット学会学術講演会予稿集，pp.909-910, 1991.
- [5] 小柳、飯田、油田：“車輪型倒立振子タイプの自立移動ロボット”，第 6 回知能移動ロボットシンポジウム予稿集，pp.51-56, 1992.
- [6] 河潤秀、油田信一：“平行二輪車の 2 次元平面走行制御”，第 11 回日本ロボット学会学術講演会予稿集，no.2, pp.603-606, 1993.
- [7] Y. Ha and S. Yuta: “Trajectory Tracking Control for Navigation of Self-Contained Mobile Inverse Pendulum,” Proc. of the 1994 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp.1875-1882, 1994.

- [8] S. Yuta and J. Iijima: "State information panel for inter-processor communication in an autonomous mobile robot controller," Proc. of the 1990 IEEE/RSJ Int. Workshop on Intelligent Robots and Systems, 1516, 1990.
- [9] Sho'ji Suzuki, Jun'ichi Iijima and Shin'ichi Yuta: "Design and implementation of an architecture of autonomous mobile robots for experimental researches," Proc. of the 1993 ICAR Int. Conf. on Advanced Robots, pp.423-428, 1993.
- [10] Shin'ichi Yuta, Sho'ji Suzuki and Shigeki Iida: "Implementation of a small size experimental self-contained autonomous robot-sensors, vehicle control, and description of sensor based behavior," Proc. of the 1993 ISER (Int. Symposium on Experimental Robots), pp.344-359, Springer-Verlag, 1993.
- [11] S. Iida and S. Yuta: "Vehicle command system and trajectory control for autonomous mobile robot," Proc. of the 1991 IEEE/RSJ Int. Workshop on Intelligent Robots and Systems, pp.212-217, 1991.
- [12] S. Iida and S. Yuta: "Feedforward current control method using two-dimensional table for DC servo motor software servo system," Proc. of the 1988 IEEE Int. Conf. of Industrial Electronics, pp.466-471, 1988.
- [13] 飯田重喜, 油田信一: "移動ロボットのナビゲーションにおける環境の表現法及びセンサ情報に基づく進路補正", 第5回知能移動ロボットシンポジウム予稿集, pp.153-158, 1990.
- [14] S. Suzuki and S. Yuta: "Analysis and Description of Sensor Based Behavior Program of Autonomous Robot Using Action Mode Representation and ROBOL/0 Language," Proc. of the 1991 IEEE/RSJ Int. Workshop on Intelligent Robots and Systems, pp.1497-1502, 1991.
- [15] J.J. Leonard and H.F. Durrant-Whyte: "Directed Sonar Sensing for Mobile Robot Navigation," Klawer Academic Press, 1992.

河潤秀 (Yun-Su Ha)

1962年4月23日生。1986年2月韓国海洋大学機関学科卒業。1989年3月同大学助手。1990年2月同大学院機関学科制御工学専攻修士課程修了。現在筑波大学大学院工学研究科博士課程在学中。自律移動ロボットの走行制御システムの開発を行っている。韓国舶用機関学会会員。

(日本ロボット学会学生会員)

油田信一 (Shin'ichi Yuta)

1948年3月23日生。1975年慶應義塾大学大学院(電気工学科)修了。東京農工大電子工学科を経て現在筑波大学電子情報工学系教授。知能移動ロボット、ロボットセンサ、ロボット用コントローラ等の研究に従事。自動制御や信号処理の理論と応用にも興味を持つ。工学博士、IEEE、計測自動制御学会の会員。

(日本ロボット学会正会員)