階層ファジィルールと学習オートマトンを用いた 自律移動ロボットの障害物回避動作計画

青 木 猛^{*1} 堀 内 淳 史^{*2} 鈴 木 達 也^{*3} 大 熊 繁^{*3}

Motion Planning of Obstacle Avoidance for Autonomous Mobile Robot Using Hierarchical Fuzzy Rules and Learning Automata

Takeshi Aoki^{*1}, Atsushi Horiuchi^{*2}, Tatsuya Suzuki^{*3} and Shigeru Okuma^{*3}

We show how an autonomous mobile robot can acquire the optimal action through the interaction with the real world. We propose a new architecture using the hierarchical fuzzy rules, fuzzy evaluation system and learning automaton. By using our proposed method, the robot acquires how to approach the goal avoiding a moving obstacle, using the steering and velocity control inputs, simultaneously. We also show the experimental results to confirm the feasibility of our method.

Key Words: Autonomous Mobile Robot, Hierarchical Fuzzy Rules, Learning Automata, Absolute Expediency, Obstacle Avoidance

1. は じ め に

近年、自律移動ロボットの障害物回避方式が盛んに研究されており、解析的手法ではポテンシャル法、非線形計画法、Gradient法[1]～[3]などが提案されている。一方、数理モデルを持たなくても人間のようにその場の大きな認識・状況判断で障害物回避を行えるようファジィ推論、ニューラルネットワークを用いた手法も数多く研究されている[4]～[11]。これら研究において、制御入力（つまり、計画システムの出力）として操舵制御と速度制御などを併用する複合方式[7]～[11]を採用する手法が提案されている。ファジィ制御においては、複数の制御入力を用いる場合、制御入力が1つの場合は、人間が持っている知識を比較的容易にファジィルールで表現・記述し、調整を行うことができる。しかしながら、制御入力が多数の場合、ルール数が飛躍的に増加すると同時に、制御入力間の干渉を避けるようファジィルールを調整することが困難になるという問題が生じる。また、ニューラルネットを用いる場合には、ニューロ制御器の構造決定や教師信号の質が大きな問題となる。多用される多層型ニューラルネットワークを例にすると、

中間層のユニット数が学習の収束時間や準備する教師信号の数に大きな影響を与えるが、そのユニット数を設計することは困難である。また、局所解に陥らぬよう複雑な環境に対し適切な教師信号を数多く作成せねばならない。これらを解決するためにはオンラインで自律的に学習すると同時に行動を自身で評価する機能を必要とする。

ファジィ制御における複数制御入力問題を解決するため筆者らは、下位レベルで障害物回避を行うための操舵制御量と速度制御量を独立に決定し、上位レベルにおいて両制御入力間の干渉を生じないようにファジィ推論を用いて重み付けして合成する手法を提案した[10]。また、そこでは、各制御量を求めるファジィルールを階層的に構成することにより、制御性能を低下させることなくルール数を削減できることを示した。しかしながら、一般には、複数の制御入力を同時に用いて回避動作を行う場合、その合成ルールはロボットを取りまく状況により異なるため、設計者が全ての状況をあらかじめ想定し、1つのルールで合成方法を表現することは容易なことではない。従って文献[10]のアプローチでは、上位レベルにおいて、ルール・パラメータの調整に試行錯誤を必要とし大きな労力を費やした。また、構築したルールでロボットを制御した場合に、その行動を定量的に評価することが困難であり、ルールの調整、評価に人間が大きく関与せざるを得なかった。例えば、ロボットが移動障害物を回避しながら目標に向かう場合、適切と思われる行動はそのままにして無駄な回避運動と思われる行動のみを押さえるようルールを調整することが困難であった。

原稿受付 1995年5月31日

^{*1}名古屋工業研究所

^{*2}ファナック(株)

^{*3}名古屋大学大学院

^{*1}Nagoya Municipal Industrial Research Institute

^{*2}Fanuc Co., Ltd.

^{*3}Graduate School of Engineering, Nagoya University

以上の点をふまえて、本論文では、各制御入力 of 合成ルールを1つの行動とみなし、それに対して複数の行動選択肢を設定し、学習オートマトンの学習アルゴリズム[12]を用いてロボットが環境に適応した行動選択能力（つまり制御入力の合成ルール）を教師信号なしで自律的にオンラインで獲得する手法を提案する。すなわち、ルール化が容易な下位レベルにおいては、階層的なファジィ推論によるアルゴリズムを用い、ルール化が困難な上位レベルにおいては、環境との相互干渉を通して学習を用いることで環境に対し、より柔軟に対応できるようなシステムを構成する。このようにファジィ推論に確率的探索手法を組み合わせることで、教師信号を必要とするニューラルネットワークやメンバーシップ関数の調整を必要とするファジィ推論のみのシステムに比べ設計・学習が容易になると同時に、未知環境においてある種の数理的特性を保証することが可能となる。

学習オートマトンは、未知の定常ランダム環境^{*}において行動の学習能力を有する確率オートマトンで、未知の定常ランダム環境に対して絶対適応性と呼ばれる数理的性質が保証されており[12]、しかも、学習アルゴリズムが単純で実時間で実行可能という特長を持つ。学習オートマトンを用いた未知環境中における移動ロボットの研究には文献[13][14]等がある。これらの研究は、環境を格子状に分割し、障害物の代わりに確率的なゲートまたは行動評価器を設定した未知環境下において、出発点から目標点までの最適経路を確率的探索手法を用いて求めている。これに対し本手法は、未知環境において移動障害物の回避手段としてファジィ速度制御を主として用い、この速度操作をサポートするよう操舵操作量の割合を環境に合わせて選択する一種の行動融合器をロボットに自律的に獲得させることを目的としている。

本システムでは、ロボットを取りまく状況を、センサからの情報に基づいて分類し、各状況に対して個別に学習オートマトンを構成する。このように各状況に対して個別に学習オートマトンを構成することにより、各学習オートマトンにとっては、各状況が未知ではあるが定常のランダム環境と見なせることになり、学習の収束性が保証できる。具体的には、状況を表すテーブルを作成し、テーブルの各要素に、その状況に対する行動生起確率ベクトルを記憶させ、この確率を学習アルゴリズムを用いて更新する。これによって、ロボットが計算-実行-評価を実際の環境の中で繰り返すことになり、最終的に、環境に適した行動選択能力を獲得することになる。そして、この環境との相互干渉を通して得られた知識（経験）はテーブルの各要素に蓄えられることになる。また、学習オートマトンを用いる場合、ロボットが起こした行動の評価が必要になるが、本システムでは、行動評価にファジィ推論を用いる。これによって、人間に近い評価が可能となる。

以下、第2章では障害物回避動作計画システムについて述べ、第3章では実験結果について述べる。

2. 障害物回避動作計画システム

2.1 問題設定

本研究に用いる移動ロボットは、移動する障害物を操舵および速度制御で避けながら目標点に達する自律移動型とする。この障害物回避動作計画システムを構築するに際し、次の前提条件を仮定する。

- 1) ロボットは一度に複数の障害物に遭遇しない。
- 2) 障害物は連続的で滑らかな動きをする。
- 3) ロボットは操舵制御および速度制御により障害物を回避する。
- 4) ロボットの移動経路に制約は与えない。
- 5) ファジィ推論は、 $\min \cdot \max \cdot$ 重心法を用いる。

2.2 システム構成

本研究で提案する移動ロボットの障害物回避動作計画システムをFig. 1示す。Fig. 1に示すように、提案するシステムは以下の5つのサブシステムから構成されている。

- 1) センシングシステム：現時点におけるロボットから見た、障害物、目標点の位置、速度ベクトルを検出するサブシステム
 - 2) 階層ファジィシステム：階層ファジィルールを用いてそれぞれの制御入力を計算するサブシステム
 - 3) 行動生成システム：6つの行動選択肢を用意し、選択された行動（合成ルール）に基づき行動を生成するサブシステム
 - 4) 行動学習システム：ロボットが直面している状況と確率表上の要素とを対応づけ、その要素内にある行動生起確率ベクトルに基づいて行動を選択し、行動に対する評価に応じて確率を更新するサブシステム
 - 5) 行動評価システム：実行された行動を評価するサブシステム
- 以下、各サブシステムについて説明する。

2.3 階層ファジィシステム

階層ファジィシステム[10]は、最終的に目標へ向かうための操舵制御入力、移動障害物を回避するための操舵制御入力および回避のための速度制御入力をそれぞれ独立かつ階層的に計算

Fig. 1 Action selection system

^{*}ここでいう環境とは、ロボットを取りまく物理的な環境とは異なる。詳しくは2.5節を参照。

Fig. 2 Hierarchical structure of fuzzy rules

するアルゴリズムである。Fig. 2 に示すように中間パラメータを用意して、階層的にファジィルールを構成することにより、必要なルール数を大幅に減少させ、制御操作量を効率的に計算することができる。ここでの詳しい記述は省略するが、センサ入力から制御入力決定までの信号処理は、Fig. 2 に示すようにボトムアップ的に実行される。その際に、1 次予測部[10] (Pre-Prediction Module) において回避の必要があると判断した場合は、以下に示すように、計算された中間パラメータをもとに回避のための速度制御入力および操舵制御入力を個別に計算する。

- 1) 回避操舵制御入力は、前田らの提案した静的危険度、動的危険度を用いて回避テーブル[4]をもとに決定する。
- 2) 回避速度制御入力は、ロボットの走行ライン上で障害物が横切る予測交差点を相対位置・速度ベクトルから幾何学的に求め、ファジィ推論により障害物回避に必要な速度の加減速度（加速度）の決定[10]を行う。

2.4 行動選択と確率表

前節で示した階層ファジィルールが出力する3つの制御入力（目標へ向かうための操舵制御入力、障害物を回避するための操舵制御入力、障害物を回避するための速度制御入力）を合成して回避動作を行う場合、その合成ルールはロボットを取りまく状況により異なる。そのため、設計者が全ての状況をあらかじめ想定し、1つのルールで合成方法を表現することは容易なことではない。この問題を解決するため、本節と次節では、各制御入力の合成ルールを1つの行動とみなし、それに対して複数の行動選択肢を設定し、学習オートマトンの学習アルゴリズムを用いてロボットが環境に適応した行動選択能力を自律的に獲得する手法を提案する。行動選択肢としてあらかじめ準備する6つの合成ルール（Action）は以下の通りである。

Action 0: 目標へ向かうための操舵制御と目標へ向かうための速度制御を行う。

Action 1: 目標へ向かうための操舵制御と障害物回避のための速度制御を行う。

Action 2: 障害物回避のための操舵制御を25%、目標へ向かうための操舵制御を75%と障害物回避のための速度制御を行う。

Action 3: 障害物回避のための操舵制御を50%、目標へ向かうための操舵制御を50%と障害物回避のための速度制御を行う。

Action 4: 障害物回避のための操舵制御を75%、目標へ向かうための操舵制御を25%と障害物回避のための速度制御を行う。

Action 5: 障害物回避のための操舵制御と障害物回避のための速度制御を行う。

なお、Action 0 中の目標へ向かうための速度制御とは、加減速度が零の（つまり、速度による回避は行わない）速度制御のことを示す。本研究では、文献[10]で設計したファジィ融合器において用いたファジィルールをロボットが実世界との相互干渉を通じて自律的に獲得することが目的である。従って、文献[10]と同様に、目標に向かい出来るだけ直線的にロボットを移動させ、どうしても必要な場合だけ操舵を切るという速度ベースの選択肢を設定することにする。また、回避操舵制御入力の割合は25%ずつ増加させているが、ロボットの機能拡張の一貫として、様々な行動選択肢そのものを自己生成し行動選択する能力をロボットに付加することも考えられるが、今後の課題としたい。

これら6つの選択肢にはそれぞれ行動生起確率があり、ルール方式により1つのActionが確率的に選択される。選択されたActionに基づき、階層ファジィルールで求めた3つの制御入力を合成し、ロボットに実際の行動を発生させる。その際、選択されたActionに従い、ロボットの操舵制御入力は以下の式を用いて決定する。

$$S_c = (1-w) \cdot S_g + w \cdot S_a \quad (1)$$

where S_c : Final Steering Control Input

S_g : Steering Control Input for Goal

S_a : Steering Control Input for Obstacle Avoidance

w : weight based on Selected Action

Table 1 は、ロボットを取りまく状況を表現する確率表である。この確率表は動的危険度 β の絶対値および相対方位角 RA の絶対値で構成されており、ロボットの遭遇する状況がこの確率表上に写像される。この確率表に用いている 2 つのパラメータは文献[10]で用いたファジィ融合器のパラメータを用いている。これは、文献[10]で設計したファジィ融合器において用いたファジィルールをロボットが実世界との相互干渉を通じて自動的に獲得させることが目的だからである。その確率表は、回避テーブル[4]の分割に合わせて作成した。

すなわち、動的危険度(文献[4]では 10 分割)は、絶対値に直して 5 分割にしてある。また回避テーブル上で静的危険度と相対させている相対方位角に関しては、静的危険度と同様に 5 分割とした。すなわち回避テーブルの正負を削除し、確率表は同じ分割法をとることにする。確率表の各要素については、例えば、障害物がロボットの正面から接近してくる場合、つまり、動的危険度 β の大きさが 1 で相対方位角 RA の大きさが 0° である場合、この状況は確率表の要素 X に相当する。また、障害物がロボットの真横から接近してくる場合、つまり、動的危険度 β の大きさが 1 で相対方位角 RA の大きさが 90° である場合、この状況は確率表の要素 Y に相当する。表の要素には Action に対する行動生起確率ベクトルが格納されており、ロボットは、直面する状況に相当する確率表の要素が持つ行動生起確率ベクトルを基にルーレット方式で 1 つの Action を選択する。

2.5 学習アルゴリズム

本システムにおいて、確率表中の行動生起確率ベクトルを何らかの評価に基づいて更新(学習)することで、ロボットは環境に適した行動選択能力を獲得することになり、獲得した能力が確率表中に蓄えられることになる。このための学習法として、本節では学習オートマトンの枠組みの中で提案された強化学習法である Lr-i 法[12]を用い、実世界との相互作用を通して行動(合成ルール)選択の学習を行う。

以下、学習オートマトンの枠組みについて簡単に説明する。学習オートマトンは、最初、Tsetlin らにより導入され、その後、Narendra らにより体系化された[12]。そのシステム概略図を Fig. 3 に示す。

オートマトンは離散化された時刻 n において、行動 $a(n)$ を選択し実行すると、環境はその行動に対する評価値 $b(n)$ をある確率分布に従ってオートマトンに返す(評価が悪ければ悪いほど 0 に近い値を返すことにする)。このように、学習オート

マトンの枠組みにおいては、環境は、単に実世界のみを指すのではなく、行動の評価を含めたサブシステムとしてとらえられている。また、評価値 $b(n)$ の取りうる値によって環境は P (評価値が 0 または 1 の 2 値)、 Q (評価値が 0~1 の離散値)、 S (評価値が 0~1 の連続値) の 3 つのモデルに分かれる。また、学習の収束性を評価するために以下の量を定義する。

$$\begin{aligned} M_{(n)} &\equiv E[b_{(n)}|p_{(n)}] = \sum_{i=1}^r E[b_{(n)}|a_{(n)}=a_i] \cdot p_{i(n)} \\ &= \sum_{i=1}^r S_i \cdot p_{i(n)} \end{aligned} \quad (2)$$

但し、 $S_i = E[b_{(n)}|a_{(n)}=a_i]$ である。

また、 n は時刻、 r は行動の数、 a_i は i 番目の行動選択肢、 $a_{(n)}$ は時刻 n での行動、 $p_{i(n)}$ は時刻 n において i 番目の行動を選択する確率、 $P_{(n)} = [p_{1(n)}, p_{2(n)}, \dots, p_{r(n)}]^T$ は時刻 n での行動生起確率ベクトル、 $b_{(n)}$ は時刻 n での行動評価値を示す。このとき $M_{(n)}$ は、時刻 n での行動生起確率ベクトル $P_{(n)}$ が与えられたときの“行動の良さ”を表す評価値の期待値で表現したものとなる。 $M_{(n)}$ をもとに次式で示される絶対適応性[12]という概念が定義できる。

$$E[M_{(n+1)}] > E[M_{(n)}] \quad (3)$$

これは、学習が進むに従って、“行動の良さ”の期待値表現が単調増加することを意味している。

次に行動生起確率ベクトルの更新則について考える。本節の冒頭で述べたように、ここでは、式(3)で表される絶対適応性が未知のランダム環境に対して保証できる PLr-i 法を用いる。PLr-i 法は、環境を P モデルとして考え、ロボットが選択した行動に対する 2 値評価 {favorable(1), unfavorable(0)} に基づいた確率更新則である。そして、選択した行動の評価が favorable(1)である場合のみ、行動生起確率ベクトルを更新する。そのときの更新則を式(4)に示す。

PLr-i Scheme

$$\begin{cases} p_{i(n+1)} = p_{i(n)} + c\{1 - p_{i(n)}\} & \text{if } a_{(n)} = a_i \text{ and } b_{(n)} = 1 \\ p_{j(n+1)} = (1 - c)p_{j(n)} & \text{for } j \neq i \\ p_{i(n+1)} = p_{i(n)} & \text{for all } i, \text{ if } b_{(n)} = 0 \end{cases} \quad (4)$$

ここで、 $c(0 < c < 1)$ は学習係数を示す。式(4)で表される更新則を用いることによって、未知のランダム環境に対して、絶対適応性が保証されることが知られている[12]。また、このとき、最も望ましい行動を a_d とする場合、

$$E[p_{d(n+1)}] > E[p_{d(n)}] \quad (5)$$

Table 1 Probability table

Degree of Dynamic Danger $ \beta $	Relative Angle RA				
	90°	0°	0°	0°	0°
0					
1

Fig. 3 Learning automata system

が成り立つことも保証できる。つまり、最適な行動の生起確率の期待値¹が単調増加することになる。この単調増加する理由については、参考文献[12]を参照されたい。

本システムでは、確率表の各要素内に、各状況に対応する行動生起確率ベクトルが格納されており、その初期値は $(1/6, \dots, 1/6)^T$ とする。ロボットは各状況に応じて行動生起確率ベクトルを取り出し、その確率に従ってルーレット方式で行動選択を行った後、式(4)に基づいて行動生起確率ベクトルを更新する。

最後に、本システムにおける環境の定常性について述べておく。本節で述べたように、ここでいう環境とは、ロボットを取りまく実世界と行動の評価とから成っている。従って、ロボットが移動を続ける限り、通常は環境の定常性は仮定できない。しかしながら、本システムでは、状況を分類してテーブル(確率表)を構成し、各状況ごとに学習オートマトンを構成しているため、各学習オートマトンにとっては、未知ではあるが、ほぼ定常的な環境下で動作すると見なすことができる。これは状況の分類を細かくすればするほど、より顕著に言える。

2.6 ファジィ行動評価器

学習オートマトンの枠組みの中で学習を行う際、評価をどのように行うかが重要な問題点の1つとして挙げられる。

一般に、動的な状況の中でロボットが選択した行動を数式で評価することは容易ではなく、また、例えできたとしても数式が複雑になる傾向がある。そのため、本論文では、ファジィ推論を用いて、人間に近い評価を行える評価システムを構築することで、この問題の解決を図る。

ファジィ推論を用いて評価を行う利点は以下の通りである。

- 1) 設計者の経験および直感的な評価方法をファジィルールで表現することができる。
- 2) 計算が簡易である。
- 3) ファジィ出力により、選択した行動の評価を定量的に表すことができ、そのまま学習オートマトンへの入力とすることができる。

ロボットの行動を評価するためのファジィ入力として以下の2つのパラメータを用いる。

- ① $(n-1)$ 試行目と n 試行目における動的危険度の差、 $|\beta(n)| - |\beta(n-1)|$ 、または n 試行目における動的危険度の絶対値 $|\beta(n)|$ で、その絶対値が、より大きな値

$$\max(|\beta(n)|, ||\beta(n)| - |\beta(n-1)||) \quad (6)$$
- ② $(n-1)$ 試行目と n 試行目におけるロボットから目標までの距離の変化

①の入力パラメータを2つの選択肢に分けたのは、より大きな値のほうが、より意義のあるパラメータになるからである。すなわち、行動の結果、動的危険度が少し小さくなくても危険度そのものが高いう場合は、高い危険度に注目すべきである。一方、行動の結果、動的危険度が大きく増加した場合、小さな動的危険度よりも危険度の大きな変化に注目し評価すべきであると考えたからである。上記の入力パラメータをもとに、min-max-重心法を用いて評価値をファジィ推論する。推論の際に用いるファジィ評価ルールを Table 2, 3 に示す。

¹生起確率そのものが確率変数になるため、期待値をとって収束性を議論する必要がある。

Table 2 Fuzzy rule for $|\beta(n)|$

Parameter ① Degree of Dynamic Danger	Parameter ② Change of Distance between Robot & Goal			
		Positive	Zero	Negative
	Big	Negative Big	Negative Medium	Negative Small
	Medium	Negative Medium	Zero	Positive Medium
	Small	Positive Small	Positive Medium	Positive Big

Table 3 Fuzzy rule for $|\beta(n)| - |\beta(n-1)|$

Parameter ① Change of Degree of Dynamic Danger	Parameter ② Change of Distance between Robot & Goal			
		Positive	Zero	Negative
	Positive	Negative Big	Negative Medium	Negative Small
	Zero	Negative Medium	Zero	Positive Medium
	Negative	Positive Small	Positive Medium	Positive Big

3. 実 験

3.1 実験結果

提案するアルゴリズムの有効性を実機で確認するため、Fig. 4 に示すシステムで実験を行った。

移動ロボットとしては比較的小さい形状(20×10×5[cm])の移動体を用いた。このロボットの胴体は2つの前輪(ゴムタイヤ)と1つの後輪(ボール)で支えられている。前輪はそれぞれステッピングモータ(5×5×1[cm], 20[g], 7.2[degree/pulse])に直結してあり、前後方向に回転できるようになっている。このモータはケーブルを通して直流電源およびコンピュータ(NEC 9801 BA: Intel 486 DX 2 40[MHz])に接続されており、コンピュータからの指令に基づいて駆動する。

ロボットを取りまく状況を認識するためのセンサとして、小型 CCD カメラを用いる。カメラは、地上約 2.5[m]の高さに固定し、実験場(2.5×3[m])を見渡せるようにしてある。但し、ワールド座標系での演算は行わず、ロボットと障害物、あるいは目標との相対的な関係のみを情報として用いる。このカメラもケーブルを通してコンピュータに接続されており、取り込んだ画像から、ロボットと障害物に関するパラメータ(位置、速度ベクトル)を計算する。画像取込、画像処理、制御入力計算、ステッピングモータ制御はすべて、このコンピュータ1台で集中管理できるようオンライン設定してある。

画像処理において認識するロボットおよび障害物の大きさは、CCD カメラから取り込んだ CRT 上の各物体を包含する、最小の円をロボットや障害物の大きさとする。よって相対距離はロボットおよび障害物を包む円の重心間距離から各円の半径を引いた値とし、この値を基に各パラメータの計算を行う。ま

Fig. 4 Experimental set up

Fig. 5 Experimental result before learning

Fig. 6 Experimental result on learning

た移動速度およびサンプリング周期については、以下の通りである。

- ・ロボット初期速度は、約 20[cm/s].
- ・障害物の移動速度は、約 10[cm/s].
- ・ロボットの速度範囲は初期速度の 1%~200%.
- ・画像処理時間の制約上、1 制御周期は約 0.2 秒.

実験では、白色ビータイル床面上でロボットや障害物が認識されやすいよう各物体の頭部を黒布で覆ってある。CCD カメラからの画像情報は、メモリ節約と処理速度アップのため分解能を 1/4 に落として用いる。そして 2 値化処理を行い、各物体の重心を計算して、その時刻における各物体の位置とする。また、速度ベクトルはこの位置の差分を近似速度ベクトルとして用いる。

実験結果を Fig. 5~7 に示す。Fig. 5 は学習を行う前のロボ

Fig. 7 Experimental result after learning

Table 4 Action probability table

Degree of Dynamic Danger $ \beta $	Relative Angle RA			
	90°			0°
0				
1				

ットと障害物の行動履歴を示す。初期状態は、障害物がロボット前方正面からロボットに対して向かってくる場合で、Table 4 中の要素 X で表される状況における動作である。初期段階ではロボットの行動生起確率ベクトルが調整されていないため不適切な行動が頻発し、目標へ向かう中間地点あたりで移動障害物と衝突している。Fig. 6 は数回走行し学習した後のロボットと障害物の行動履歴を示している。行動生起確率ベクトルが環境に対し調整され、ロボットが障害物をスムーズに回避しながら、目標地点に到達していることが分かる。Fig. 7 は、学習後のロボットと障害物の行動履歴を示している。Fig. 5 と似た状況である Fig. 7 の状況でロボットの学習効果を示す。ロボットが障害物をスムーズに回避しながら目標地点に到達しており、学習により行動生起確率ベクトルが環境に適応していることを示している。

また、Fig. 8 は、この状況下での各行動生起確率の収束状況（学習履歴）を示している。但し、この時の学習係数 c は 0.1 である。縦軸は確率を示し、横軸は学習回数を示す。Fig. 8 に示すように、学習に伴い Action 5 の行動選択確率が 1 へ、その他の Action は 0 へ向かい、初期段階で急速に学習が進み、その後、ゆるやかに収束している。この結果、障害物が比較的正面から接近してくる場合は、速度制御だけで障害物回避を行うのではなく、操舵制御を十分に用いて回避する行動選択能力をロボットが獲得したことを示している。なお、似た状況での実験を何度も繰り返して行い、Fig. 8 の縦軸を生起確率の期待値とすると、2.5 節で述べたように $E[P_{5(n)}]$ は単調増加になる。Fig. 9 は、Table 4 に示す確率表上の要素 Z における行動生起確率の収束状況（学習履歴）を示している。要素 Z に相当する状況は、障害物がロボットの比較的正面に存在しているが、動的危険度（近づいてくる危険）はない場合に相当する。但し、この時の学習係数 c は 0.1 である。Fig. 9 に示すよう

Fig. 8 Action probability history of element X using PLr-i

Fig. 9 Action probability history of element Z using PLr-i

に、Action 0 の行動選択確率が 1 へ、その他の Action は 0 へ向かい、初期段階で急速に学習が進み、その後、ゆるやかに収束している。この結果、障害物をよける必要がない場合は、回避のための速度・操舵制御を行わず、目標へまっすぐに進む行動選択能力をロボットが獲得したことを示している。なお、この場合も Fig. 8 と同様に、何度も実験を行い、行動生起確率をプロットすれば $E[P_{0(n)}]$ は単調増加となる。

本システムでは、速度ベクトルに関しては、位置差分を近似値として用いており真の速度ベクトルに対し誤差を含んでいる。また、ハードウェアおよび処理速度の制約からサンプリング周期は約 0.2 秒になり、画像処理に約 0.15 秒の時間を要する。そのため制御に遅れが生じる。従って、センサノイズや画像の低分解能を考慮すると大まかな位置・速度情報しかロボットには与えられていない。しかしながら実験結果で示したように、本手法では大まかな情報でもロボットは不必要に大きく操舵をきることなく障害物を避けながら目標へ向かうよう学習していることが分かる。また、行動選択肢は、25%, 50%, 75%, 100% を含む 6 選択肢を準備したが、結果的には、確率表において 25%, 75% の選択肢を選ぶ要素は少なく、50%, 100% を含む 4 つの選択肢で十分対応できると考えられる。これは、収束時間のさらなる短縮へとつながる。

3.2 考察

提案手法を用いる場合、提案するシステムの確率表におけるすべての要素に対し、ロボットが均等かつ十分な学習機会を得

Fig. 10 Simulation result only using steering control input based on steering table for obstacle avoidance

Fig. 11 Simulation result using our proposed method

ることは容易ではない。また、学習が進むに従い、危険な状況に陥らぬようロボット自身が行動するため、ますます学習機会の不均衡を生じる。結果として、各要素における学習の収束速度に差が生じてくる。Fig. 8 および Fig. 9 に示した行動生起確率の学習履歴は、遭遇する状況が多く得られたため、比較的速やかに収束している。しかしながら、その他の状況は、遭遇する機会が少ないために学習しにくくなっている。

実験結果は示していないが確率表のすべての要素で 1 つの行動選択確率が収束するわけではなく、取りあえず不適切と思われる行動の選択確率が高い確率で 0 に収束し、2 つ以上の行動確率が 0 以外の値を持つような要素も存在する。これは、学習回数の不足からくる要素もあるが、残ったどの Action でもファジィ評価器から、大きさは異なるが報酬をもらうことができるということを示していると考えられる。このことは、同じ状況でも異なる行動をすることを示し、評価の許す範囲内で行動の多様性を表現していると考えられる。学習という概念に相反することではあるが、この意味において、確定的に行う行動制御に対して、デッドロックに陥りにくいと考えられる。

本実験では、学習アルゴリズムに PLr-i 法を用いている。この実験に先立ちシミュレーションで P モデルと Q モデルによる収束性の比較を行った。結果は省略するが、収束性に関しモデルの違いによる大きな差はみられず、むしろ学習パラメータが大きく影響していた。

また、本実験では環境を P モデルとして学習する PLr-i 法を用いているが、収束速度の観点から、選択した行動が良くない場合でも行動生起確率ベクトルを更新する PLr-p 法や、環

境をSモデルとして扱い、離散値の行動評価値に応じて確率を更新するSLr-i法、SLr-p法[12]を用いる改良策も考えられる。これらの手法への拡張は、ファジィ評価器を用いているため、容易に行うことができる。但し、評価が良い場合のみ行動生起確率ベクトルを更新するPLr-i法、SLr-i法とは異なり、PLr-p法やSLr-p法は未知のランダム環境における絶対適応性は保証されていない。そのため、PLr-i・PLr-p法やSLr-i・SLr-p法などのハイブリッド学習法が有効であると考えられる。

最後に、Fig. 10に回避テーブルのみで回避操作を行った場合を示し、Fig. 11に本手法で回避を行った場合を示す。Fig. 10はFig. 11と比較して、大きく回避していることが分かる。しかし、この状況を人間が修正しようすると他の状況に置いて、この修正が原因による負の影響が出る可能性があり、人間が設計するには多大な労力を必要とする。本手法は、ロボットが経験を積み重ねながら行動選択能力を獲得していくことに主眼をおいている。従って衝突しないための行動禁止機能は付加していない。今後、禁止行動監視モジュール等を付加する必要があると考える。

4. ま と め

本論文では、速度と操舵の両方を制御可能な自律移動ロボットを対象とし、階層ファジィルールと学習オートマトンを用いた障害物回避動作計画手法を提案した。提案した手法において、まず、階層ファジィルールを用いて、速度制御入力と操舵制御入力を個別に求め、次に、その合成ルールを行動と見なし、学習オートマトンの学習アルゴリズムを用いて、行動学習を行った。また、試作した実機により、提案した手法の有効性を確認した。このようにファジィ推論に確率的探索手法を組み合わせることで、教師信号を必要とするニューラルネットワークやメンバーシップ関数の調整を必要とするファジィ推論のみのシステムに比べ設計・学習が容易になると同時に、未知環境においてある種の数理的特性を保証することが可能となる。

本手法は、ロボットが遭遇する状況をあらかじめ分類し、その状況ごとに学習オートマトンを構成するという立場をとっているため、行動学習の絶対適応性が保証できるという特長を持っている。今後の課題としては、複数移動障害物への拡張やロボット自身による行動選択肢生成や状況分類などの機能拡張が挙げられる。

参 考 文 献

- [1] 小森谷, 小谷内: “移動ロボットの知能化”, 日本ロボット学会誌, vol. 9, no. 1, pp. 100-111, 1991.
- [2] 斉藤, 津村: “移動ロボット群における衝突回避”, システム制御情報学会論文誌, vol. 3, no. 8, pp. 252-260, 1990.
- [3] K. Fujimura and H. Samet: “Path Planning among Moving Obstacles Using Spatial Indexing,” Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1662-1667, 1988.
- [4] 前田, 竹垣: “ファジィ推論を用いた移動ロボットの動的障害物回避運動”, 日本ロボット学会誌, vol. 6, no. 6, pp. 518-522, 1988.
- [5] 小山, 前田: “ファジィプロダクションシステムによる移動ロボットの動的障害物回避問題の検討”, 日本ロボット学会誌, vol. 9, no. 1, pp. 75-78, 1991.
- [6] 竹内: “自律型ファジィ移動ロボット”, 日本ロボット学会誌, vol. 6, no. 6, pp. 549-556, 1988.
- [7] 永田, 土谷: “移動ロボットの障害物回避に関する知的制御”, 日本ロボット学会誌, vol. 11, no. 8, pp. 1203-1211, 1993.
- [8] 久保田, 橋本, 原島: “移動障害物に対する障害物回避行動形式の一手法”, 第8回日本ロボット学会学術講演会, pp. 181-182, 1990.
- [9] 石川: “ファジィ制御を用いた自律型移動ロボットの誘導方式の検討”, 日本ロボット学会誌, vol. 9, no. 2, pp. 149-161, 1991.
- [10] 青木, 他: “階層ファジィルールを用いた自律移動ロボットの障害物回避動作計画”, 電気学会C論文誌, vol. 113, no. 7, pp. 535-542, 1993.
- [11] P. Deplanques, et. al: “Sensor-Based Control Versus Neural Network Technology for the Control of Fast Mobile Robots Behaviors in Unstructured Environments,” Proc. IECON, vol. 1, pp. 694-699, 1992.
- [12] K. Narendra, 他: LEARNING AUTOMATA, Prentice-Hall, Inc., 1989.
- [13] 馬場: “未知環境中における移動ロボットの知的動作”, 計測自動制御学会論文集, vol. 23, no. 8, 1987.
- [14] 馬場, 最上, 白石: “未知環境中における移動ロボットの知的動作に関する基礎的研究”, 計測自動制御学会論文集, vol. 24, no. 9, 1988.

青木 猛 (Takeshi Aoki)

1962年3月21日生。1984年3月, 名古屋大学工学部航空学科卒業。1986年3月, 同大学大学院修士課程修了。同年4月, 名古屋市経済局工業研究所入庁。知能ロボット, メディカルエレクトロニクスなどの研究に従事。計測自動制御学会, 日本ファジィ学会, 日本人工臓器学会の会員。

(日本ロボット学会正会員)

鈴木達也 (Tatsuya Suzuki)

1964年1月3日生。1991年3月, 名古屋大学大学院博士課程修了。工学博士。同年4月, 名古屋大学工学部助手。1995年同講師, 現在に至る。ロボ制御, 学習制御を用いたモーションコントロール, 離散事象システム理論を用いた知的システムに関する研究に従事。1994年システム制御情報学会奨励賞, 1995年電気学会論文賞受賞。電気学会, 計測自動制御学会, システム制御情報学会, IEEEの会員。

(日本ロボット学会正会員)

堀内淳史 (Atsushi Horiuchi)

1970年8月7日生。1993年3月, 名古屋大学工学部電子機械工学科卒業。1995年3月, 同大学大学院修士課程修了。同年4月, ファナック(株)に勤務。

大熊 繁 (Shigeru Okuma)

1948年2月16日生。1970年3月, 名古屋大学工学部電気学科卒業。1972年3月, 同大学大学院修士課程修了。1974年ケース・ウェスタン・リザーブ大学大学院修士課程システム工学専攻修了, 1977年名古屋大学大学院博士課程中退。同年, 名古屋大学工学部助手。講師を経て, 1984年, 同助教授。1990年同教授, 現在に至る。1978年, 工学博士。この間, ロボティクスとパワーエレクトロニクスの研究に従事。電気学会, 計測自動制御学会, 日本機械学会, IEEEの会員。

(日本ロボット学会正会員)