

# 自己組織化ネットワークを用いた移動ロボットによる 行動系列からの部屋の認識

○室田 盛道  
大阪大学 基礎工学部

山田 誠二  
大阪大学 産業科学研究所

## Recognizing rooms with a mobile robot's behavior sequence

○Morimichi MUROTA  
Osaka University

Seiji YAMADA  
Osaka University

### 1 はじめに

従来の移動ロボットによる環境認識の研究は、セルラマップに代表されるように、環境の幾何学的構造を高精度で獲得することを目指したものが多く、しかし、精度のよいセンサーを持たない生物でも環境の認識は行っているし、例えば部屋の識別などの重要ではあるが比較的簡単なタスクのためには、必ずしも幾何学的地図が必要ではない。また、工学的にも、精度の低いセンサとアクチュエータを用いることは、コスト面で意義がある。また、当然のことながら、ロボットはある程度の環境変化に対し、適応可能でなければならない。さらに、環境認識において、明示的な教師なしに識別が可能になる学習能力も要求される。

以上のような要求を満たす移動ロボットを実機を用いて構築することが、本研究の目的である。換言すれば、精度の良くないセンサーを用いて、どの程度の環境認識ができるのかを探求することを研究の目的とする。本研究では、頑健な識別能力と教師なし学習を実現するために、行動 (behavior) ベースの移動ロボットにより得られる行動系列から、コホーネンの自己組織化ネットワークを用いて、形状の異なる部屋の識別を行うことを目指す。

### 2 全体の処理の流れ

全体の処理の流れを図 1 に示す。まず、識別すべき  $n$  種類の部屋を行動ベースの移動ロボットが、壁沿い移動 (wall following) を行って、部屋内部を一周し、そのとき実行された行動系列 (記号列) を得る。このとき、それぞれの部屋に対し、少なくとも一回は壁沿い移動が行われるとし、合計  $m (\geq n)$  個の行動系列が得られたとする。得られた行動系列を多次元ベクトルに変換し、それを自己組織化ネットワークに入力して、学習が行われる。このとき、どの入力ベクトルがどの部屋に対応するかという情報を明示的に与える教師は存在しない。また、 $m$  個の入力ベクトルが繰り返して与えられて、学習が進んで行く。学習が終了すると、新たにいずれかの部屋の行動系列を自己組織化ネットワークに入力し、その競合層の勝者ノードを調べることにより、その部屋が以前にたどったどの部屋であるかが認識される。以上の手続きの詳細を、次節以降で説明していく。

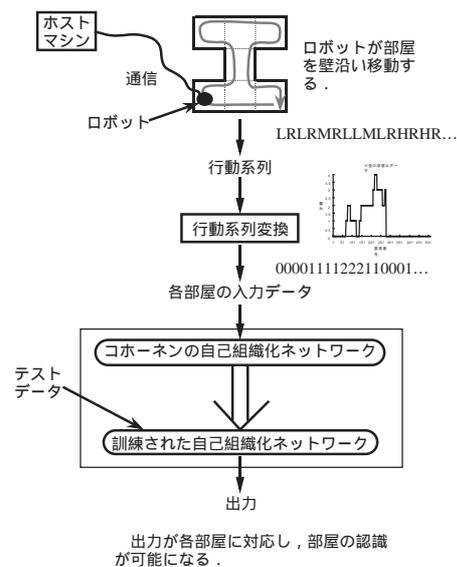


Fig.1 全体の処理の流れ

### 3 行動ベース移動ロボットによる壁沿い移動

本研究で用いる移動ロボット (図 2) は、左右前方 2 個ずつ、正面と斜め前方を向いたの赤外線近接センサを計 4 個もち、さらに慣性型方向センサと移動距離センサも搭載している。赤外線近接センサは壁の認識に、慣性型方向センサはステアリングに、また、移動距離センサは、コーナーを曲がる際の一定距離の移動に用いられる。しかし、赤外線センサの精度は良くなく、またロボットの周囲 20 センチ程の範囲の 4 方向の距離情報という、非常に局所的な情報しか得られない。このような知覚能力の低い移動ロボットを用いることは、我々の研究目的にそったものである。

また、駆動系は、左右に 2 つの PWM で制御された独立したステップモータによる駆動輪、前後に 2 つの操舵輪がある。操舵輪は、一つの DC サーボモータで駆動される。

移動ロボットは、壁沿い移動という単純な行動を行うが、環境変化への適応性を高めるため、多少の障害物に対しても同様の移動が可能でないとならない。そのため、本研究では、Brooks の提案 1 に始まる行動ベースを用いた移動ロボットを構築した。行動ベース

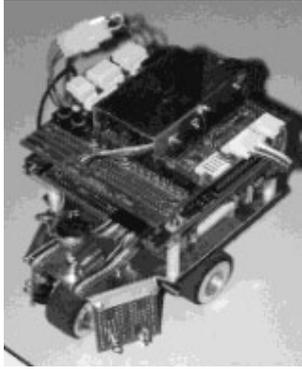


Fig.2 移動ロボット

は、環境変化に対し頑健であり、また比較的精度の低い局所的なセンシングでも、制御が可能である。また、同じ部屋に対する移動軌跡の再現性はそれほどないが、実行された行動系列は再現性が期待できるという好ましい特性がある。

本研究で、移動ロボットは壁沿い移動を行うが、その時に用いられるのが、以下のリアクティブルールである。なお、ステアリングは、正面方向に向かって絶対的であり、ロボットは時計周りに壁沿い移動をする。

- 行動 A (凹コーナーを曲がる) : *if* 前方の壁が 10cm 以内に近づいて、左 10cm 以内に壁がある *then* 右に 40°ステアリングを切る。
- 行動 B (凸コーナーを曲がる) : *if* 左右 5cm、前方 10cm に障害物なし *then* 10cm 前方に進み、左に 40°ステアリングを切る。
- 行動 C (壁沿い 1) : *if* 壁に 5cm 以内に近づいた *then* 右に 13.5°ステアリングを切る。
- 行動 D (壁沿い 2) : *if* 壁から 5cm 以上離れた *then* 右に 13.5°ステアリングを切る。

以上の 4 つの行動を適時実行するだけで、かなりスムーズな壁沿い移動が可能となる。また、上のルールの条件部は、センサデータから簡単に判定可能であり、結論部もアクチュエータに直接命令できるレベルまで詳細化されているため、迅速な行動が可能となる。

#### 4 行動系列の入力ベクトルへの変換

壁沿い移動により各部屋を一周することにより、そのとき実行された前述のリアクティブルールの系列が獲得される。直観的に、この系列は、部屋の形状を反映していると考えられるので、この系列を入力ベクトルに変換して、自己組織化ネットワークで学習されることで、形状の違う部屋の識別が可能になることが期待できる。

行動系列を自己組織化ネットワークに入力するためには、行動系列を多次元ベクトルに変換する必要がある。この際、任意の変換が可能だが、注意すべきは、うまくクラスタリングができるような変換を設定することである。自己組織化ネットワークは、単に入力ベ

クトル間の(ユークリッド)距離の大小関係(位相?)を保持したまま、次元の低い疎な離散ベクトル値に変換するだけなので、元の入力ベクトル空間状態で望ましいクラスタリングができていないとよい結果が得られない。ここで、具体的に考慮しなければならないのは、以下の 2 点である。

- 定性的には、入力の高次元ベクトル空間において 1 つのグループになって欲しい入力ベクトル値間距離の距離は、1 つのグループになって欲しくないベクトル値間の距離よりも小さくなるように変換しなければならない。
- ノイズの影響を吸収できるような変換が望ましい。

そこで、本研究では、部屋の形状が多角形であるという前提の基に、次のような変換を行う。ここでは、入力ベクトルの次元数は、得られた行動系列の長さ  $n$  以上の適当な数  $m$  であり、入力ベクトルは、初期ベクトル値をすべて 0 とする。今、行動系列を  $[r_1, r_2, \dots, r_n]$  ( $r_i \in \{A, B, C, D\}$ )、入力ベクトル  $I = (v_1, v_2, \dots, v_m)$  ( $n \leq m$ ) とすると、 $I$  のベクトル値が  $v_1$  から  $v_n$  の順に、以下の規則に従い決定される。

1.  $r_i = A$  なら、 $v_i = v_{i-1} + 1$ 。
2.  $r_i = B$  なら、 $v_i = v_{i-1} - 1$ 。
3.  $r_i = C$  or  $D$  なら、 $v_i = v_{i-1}$ 。
4.  $v_i = 0$  ( $i > n$ )。

この変換は、前述の条件を満たしていることを説明していく。(これ以降、未完。)

#### 5 自己組織化ネットワークによる学習

本節では、コホーネンにより提案された自己組織化ネットワーク 2 について、簡単に触れる。自己組織化ネットワークは、入力層と競合層からなる 2 層ネットワーク (図 3) であり、入力層のノード (入力ノード) から競合層のノード (競合ノード) へのリンクが、完全結合されている。入力ベクトルが与えられると、入力ノードは入力ベクトルの対応した要素の値をとり、それから競合ノードは入力の加算を行い、唯一の勝者ノードを決定する。そして、その勝者ノードの近傍のノードの重みが更新される。

今、入力ベクトルと、入力ノード全体から一つの競合ノード  $u_i$  へのリンクの重みをそれぞれ次の  $E, U_i$  とする。ここで、 $i$  は競合層のノードである。

$$E = [e_1, e_2, \dots, e_n]$$

$$U_i = [u_{i1}, u_{i2}, \dots, u_{in}]$$

自己組織化ネットワークは、まず入力ベクトルに対する各競合層のノードの一致度を計算する。この一致度は、下式で計算される、重みと入力ベクトルのユークリッド距離である。

$$\|E - U_i\| = \sqrt{\sum_j (e_j - u_{ij})^2}$$

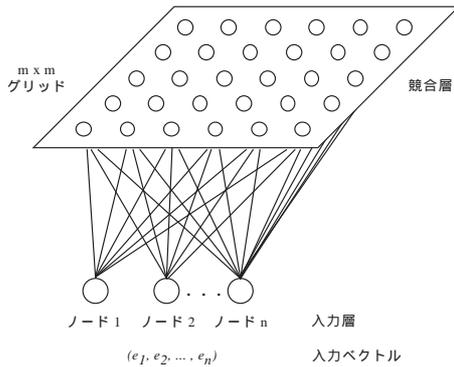


Fig.3 コホーネンの自己組織化ネットワーク

一致値の最も低い(最も一致する)ノードを勝者ノードと呼ぶ。勝者ノードを  $c$  とすると、 $c$  は以下の式を満たす。

$$\|E - U_c\| = \min_i \{\|E - U_i\|\}$$

ここで、最小値はすべての競合ノードから選択される。複数のノードが最小の一致値を持つなら、最小の  $i$  のノードが選ばれる。

勝者ノードが決まると、その近傍の競合ノードの重みが更新される。近傍は、競合層の次元に応じて適当に決められる。例えば、競合層が2次元なら、勝者ノードを重心にするある大きさの正方形がよく用いられる。近傍のノードの重みの更新式は下式である。なお、 $\alpha$  は、学習率である。

$$u_{ij}^{\text{new}} = u_{ij}^{\text{old}} + \Delta u_{ij}$$

$$\Delta u_{ij} = \begin{cases} \alpha(e_j - u_{ij}) & : \text{ノード } i \text{ が近傍にある} \\ 0 & : \text{それ以外} \end{cases}$$

以上の処理を入力ベクトルが与えられる毎に繰り返す。また、学習率と近傍のサイズは、学習が進むにつれて減少させる場合が多い。

本実験では、入力層は、部屋を壁沿い移動で一周して得られる行動系列の長さより大きな適当なサイズが採られる。そして、競合層のノードは、1次元配列としている。ただし、競合層の次元の設定に確たる根拠はない。

## 6 実機による実験

前述の移動ロボットを用いて、実機による実験を行った。計算機は、Intel DX4(100MHz, RAM16M)のIBM-PC/AT 互換機を用い、Microsoft社のVisualC++を用いて実装をした。計算機と移動ロボットは、シリアルRS232Cを介して通信する。

### 6.1 各種設定

白いプラスチックの板で作った図4に示すような7種類の部屋を、移動ロボットにそれぞれ10周づつ壁沿い移動させ、行動系列を獲得させた。ただし、今回の実験では、部屋を一周したことは人間が教えており、さらに、壁沿い移動開始地点は部屋の種類によって固

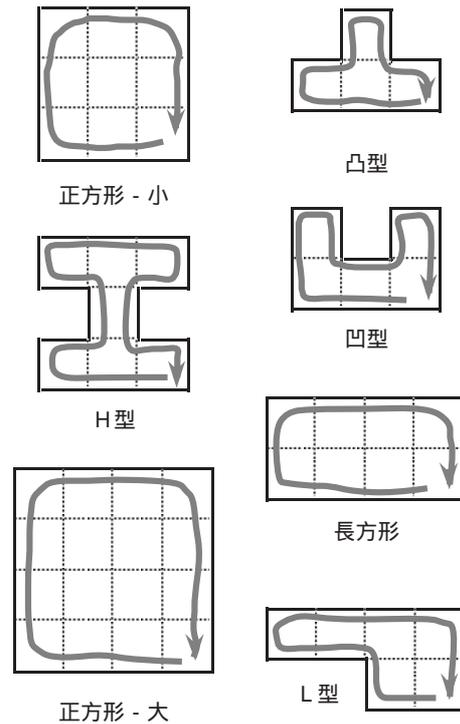


Fig.4 7種類の部屋

定している。この点は、移動距離センサと方向センサにより一周したことを認識させ、最も長い壁で行動系列をソートすることにより、任意の開始点を許すように今後改良予定である。

移動ロボットは、各部屋を1周するのに約1400の行動を実行するが、これをそのまま入力ベクトルとするとサイズが大き過ぎるので、行動C、Dの壁面を移動している系列の比を保持し、かつ行動A、Bを削除しないように、行動系列を1/4に間引いた。

学習における総訓練回数は、4200である。行動系列は、各部屋から一つずつ選んだ7個をそれぞれ600回づつ順不同で与えた。自己組織化ネットワークの入力ノード数を520(次元)、競合層を1次元で32ノードとし、前述の変換方法で、行動系列を入力ベクトルに変換した。また、 $t$ 回目の学習における、学習率 $\alpha$ と近傍の幅 $d$ を以下の式を用いて、学習進行とともに変化させた。また、競合層のノードの重みは、 $1.5 \pm 0.15$ でランダムに設定した。

$$\alpha_t = 0.2 \left(1 - \frac{t}{4200}\right) \quad d_t = 11 \left(1 - \frac{t}{4200}\right)$$

### 6.2 実験結果

実験結果を図5に示す。このグラフで、 $x$ 座標1のデータは、実際の訓練で使用された各部屋の行動系列のデータの勝者ノードを表し、2~10は、学習には使われなかった残り9つの行動系列を入力した場合の勝者ノードを示している。多少ばらついて見えるが、実は勝者ノードの近傍ノードの重みはほとんど同じであり、よって一致値もほぼおなじである。よっ

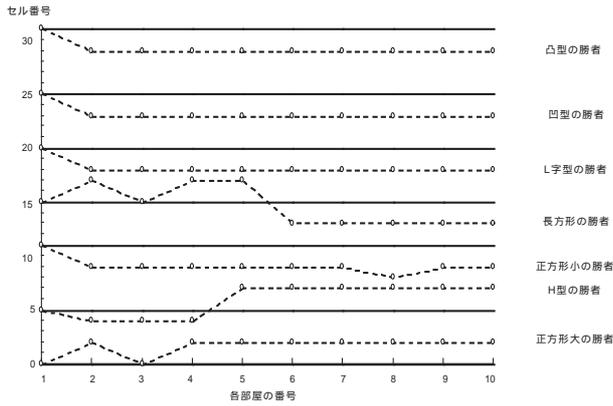


Fig.5 実験結果

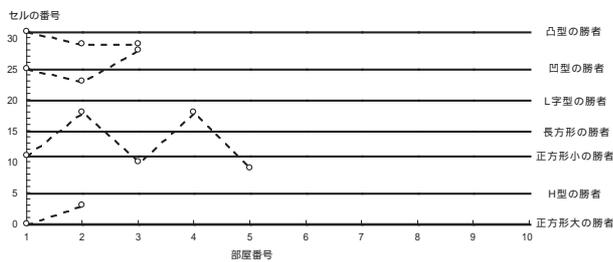


Fig.6 実験結果 (障害物がある場合)

て、部屋の認識が実現されたと考える。また、さらに部屋の壁面に障害物を置いた場合の結果を、図6に示す。その結果、9つのうち4つが正しく認識できなかった。障害物が複数散らばっている場合や、障害物が大きい場合がうまくいかなかった。

## 7 検討課題

## 8 関連研究

## 9 まとめ

## 参考文献

- 1) R.A. Brooks: "A Robust Layered Control System for a Mobile Robot", IEEE Robotics and Automation, vol. 2, no. 1, pp. 14-23, 1986.
- 2) T. Kohonen: "Self-Organization and Associative Memory", Springer-Verlag, 1989. 中谷和夫 (訳), "自己組織化と連想記憶", シュプリンガー・フェアラーク東京, 1993.
- 3) U. Nehmzow and T. Smithers: "Mapbuilding using Self-Organising Networks in Really Useful Robots", Proc. of the 1st Int. Conf. on Simulation of Adaptive Behavior, pp. 152-159, 1990.
- 4) K. Berns, R. Dillmann and U. Zachmann: "Reinforcement-Learning for the Control of an Autonomous Mobile Robot", Proc. of IROS-92, pp. 1808-1815, 1992.