

# 1412 ニューラル・ネットによるロボットの運動学習

東京大学工学部 \*柴田克成, 稀葉雅幸, 井上博允

Learning of robot's movement by neural network

The University of Tokyo, Department of Mechanical Engineering

\*Katsunari Shibata, Masayuki Inaba, Hirochika Inoue

## 1. はじめに

ニューラル・ネットは並列処理や学習ができる等の利点を持ち、近年非常に注目を集めている。ロボットの運動制御の面についても、その有効性は川人らによって確かめられている<sup>[1]</sup>。我々は、振り型の自律移動ロボットを対象として、ニューラル・ネットを用いてその動特性を学習し、制御することを目的としている。まず、これを実機へ応用する前段階として、このロボットのモデルを作成して、その運動とモーターへの出力の関係を学習させて、制御することを試みたので報告する。

## 2. 運動学習とバック・プロパゲーション法

様々なニューラル・ネットのモデルが提案されているが、ここでは、以下の理由でバック・プロパゲーション法を用いた。

(ア)バック・プロパゲーション法は、与えられた入力理想出力の組より、出力と理想値との差の自乗の和が最小になるようにロジスティク関数を用いて近似している。このため、複雑な系を学習し、その結果をアナログ値で要求するような問題では、適していると考えた。

(イ)実際に人間の運動を制御している小脳を中心とした部分は可塑性が大きいことが知られており<sup>[2]</sup>、また、“小脳パーセプトロン論”<sup>[2]</sup>も唱えられている。よって、ロボットの運動学習にパーセプトロンの拡張であるバック・プロパゲーション法を用いるということは、人間の様な柔軟な適応学習機能の実現に近づくものと思われる。

## 3. 学習システム

### 3.1 学習の対象

我々の作った二輪の自律移動ロボットを図3-1に示す。このロボットは、車輪部と振り部に分けられ、その間のモーターによって駆動される。その際に大きな慣性を持つ振り部が振動し、制御に影響を及ぼす。ここでは、実際にロボットに適用する前段階と

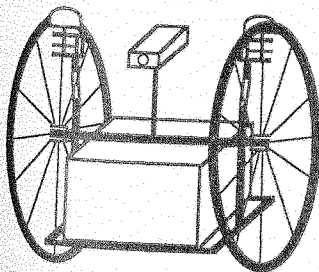


図3-1 自律移動ロボットの外観

して、図3-2のようにこのロボットの2次元モデルを考えて、平面走行時、つまり図3-2で $\alpha = 0$ の時の運動方程式に、モーターへの出力値 $Out$ を与え、サンプリングタイムを62.5msecとして、ルンゲ・クッタ法で数値計算した回転速度 $V$ を実際のロボットの代りとして用い、その $V$ と $Out$ の関係を学習させた。

### 3.2 学習方法

動きの学習、つまり、ある速度で走れという指令からモーターへの出力をどれだけにするかということバック・プロパゲーション法を用いて学習していく方法を述べる。ネットワークへの入力は、データの値を一定時間保持するようなユニットがネットワーク内に存在すると仮定して、サンプリングしていった時の過去3回分のセンサ(速度)のパターン $V_n \sim V_{n-2}$ と過去4回分の実際のモーターへの出力パターン $Out_n \sim Out_{n-3}$ 及び目標速度 $V_{target}$ とし、その時モーターに出力すべき値をネットワークの出力とするようにした。この学習システムでは、複数の速度入力により加速度成分も考慮できる上、実際に経てきたセンサ履歴を入力することによって、ネットワークがフィードバックの動きを担うことができるという利点を持つ。また、ロボットが坂道を登ったり、自重が変化する時にそれを検知して反応できるようにモーターへの出力の履歴をネットワークへの入力に加えたが、その有効性を検証するような実験はまだ行っていない。この学習システムで実際にロボットを動かしながら学習していくには、次のような2つの操作をすれば良い。

- 目標速度を適宜選んで、前述のようにネットワークへ入力し、モーターへの出力を求めて、実際にモーターを駆動する。
- モーターへ出力した後に、目標速度の入力の部分だけを、得られた速度と置き換えて、もう一度ネットワークの出力を得る。そして、(a)で求めた出力を教師信号としてネットワークの補正(学習)を行う。

しかし、上記の方法をリアルタイムで実現することは現状では簡単には行えない上、常にロボットの

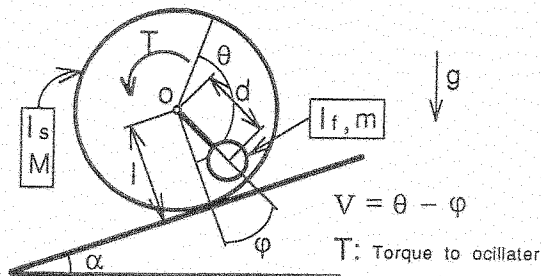


図3-2 自律移動ロボットの2次元モデル

軌道を生成して動かし続けなければならない。そこで、予め生成した有限個のデータのみで、また、前述の(a)を行わなくても学習できるように、まずロボットをネットワークとは無関係に運動させ、その時の速度とモーターへの出力のパターン  $\{V_k, Out_k\} (k=1, 2, \dots)$  を作り、図3-3のように、ネットワークへの入力データをFlow of dataと書いた矢印の方向へデータを一つずつ動かしながら学習を行っていった。この時、有限個のデータの中に様々な情報が盛り込まれるように、最初にロボットを動かす時のモーターへの出力パターンを、乱数成分を組み込んだ式、

$$Out_{k+1} = (3 \times Out_k + random) / 4$$

で決められた1000個のデータ列とした。また、学習ネットワークは次のように構成した。

- (1) 入力層には前述のような8個の入力をそれぞれ0~1の値に線形変換して与えた。
- (2) 中間層は二層設け、各層に10個のユニットを配した。
- (3) 出力層は1ユニットとし、教師信号は0.05~0.95に線形変換して与えた。
- (4) 各層は1つ上の層にのみ全ユニット結合し、同一層間の結合は行わないものとした。
- (5) 結合の重みの初期値は0~1のランダムな値とした。

#### 4. 学習結果

学習を行った結果、エラーは図4-1のように変化した。但し、エラーは次式で与えられる。

$$Err = (\text{ネットワークの出力} - \text{教師信号})^2 / 2$$

ほぼ計算による丸め誤差の範囲内まで収束している。この時のネットワークを使って、学習に用いた目標速度パターンを与えた場合には良く追従した。また、学習に用いていない目標速度パターンも色々を与えてみたが、非常に良く追従した。その一例を図4-2に示す。また、これと同じパターンに対しPID制御を行ったものを図4-3に示す。これらを見てわかるように、学習したネットワークはかなり正確にモーターへの出力を予測できることがわかる。また、学習が未熟な時、特に学習回数200回未満の時には学習ネットワークだけでは十分に対処できないため、学習ネットワークの出力にさらにフィードバック制御を加えることによって改善することができた。

#### 5. 考察と今後

バック・プロバゲーション法によって、運動制御が比較的容易に実現できることを示した。また、これによる制御は、フィードフォワード+フィードバック制御の要素を持つことがわかった。今後これを発展させていくには、系の構造が複雑で式として表せないようなものの場合に有効であることや、環境の変化に対しても容易に適応し、また、環境を認識するセンサがあれば、環境と自分の動きとの関係を学習し、これを動作に反映させられるかを検証することが課題となる。また、ニューラル・ネットは、並列処理による高速化が非常に有効である画像処理をはじめとして、様々な方面に 응용が可能であると考える。

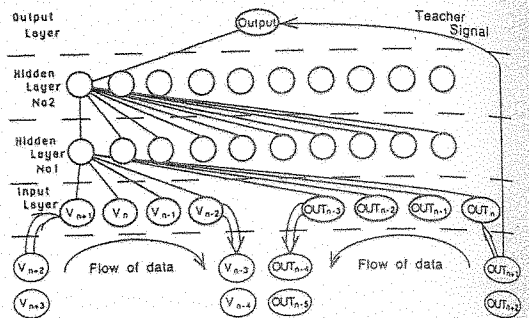


図3-3 学習ネットワークの構成と入力方法

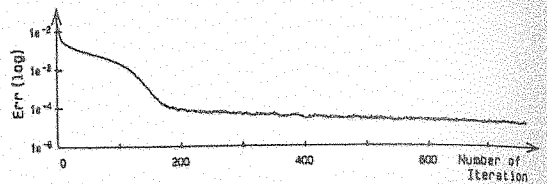


図4-1 学習時のエラーの推移

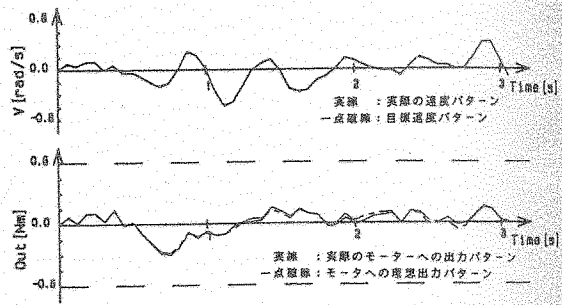


図4-2 学習後の目標速度パターンへの追従とモーターへの出力

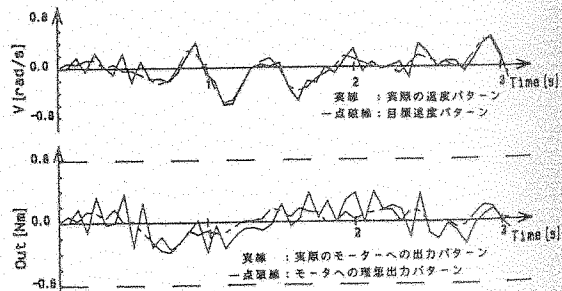


図4-3 PID制御による目標速度パターンへの追従とモーターへの出力

#### 参考文献

- [1] 川入光男, 宇野洋二, 鈴木良次: "随意運動制御における適応と学習II", 日本ロボット学会誌, Vol. 6, No. 3, pp.222-230(1988)
- [2] Rumelhart, McClelland, and the PDP Research Group: "PARALLEL DISTRIBUTED PROCESSING", Vol.1, MIT Press (1986)
- [3] 伊藤正男: "小脳の学習とパーセプトロンモデル", 数理科学, No229, pp.30-36(1982)