研究論文

RBFネットワークを用いたむだ時間+非線形プラントに対するモデル追従制御*

石川 陽一**, 石田 義久**

Model-Following Control for Nonlinear Plants with Time Delay Using RBF Neural Networks

Yoichi ISHIKAWA, Yoshihisa ISHIDA

In this paper, the design of a control system for nonlinear plants with a time delay is proposed. It is derived that the control systems of any nonlinear plants with a time delay can be reduced to a simple second-order model. For the identification of nonlinear plants, radial basis function neural networks, which are known for their stable learning capability and fast training, are used. In the simulation study of nonlinear plants with a time delay, it was observed that the steady-state error between the plant output and the reference model output was small. Moreover, in the experimental study of an actual pneumatic cylinder, it is shown that the proposed method yields a stable tracking performance.

Key words: RBF neural networks, Nonlinear plants, Identification, Positioning control

1. はじめに

非線形プラントにおいては、ファジィ制御¹⁾²⁾や階層型ネットワーク³⁾⁴⁾⁵⁾を用いた制御系が提案されている。ファジィ制御の利点は、システムの伝達関数を同定する必要がないこと、入出力画像を簡単に表すことができることなどである。しかし、この方法は、本質的に制御系CADツールが利用できない、メンバーシップ関数の高性能な調整が必要であるなどの不利な点がある。一方、階層型ネットワークを用いた制御系は、非線形な入出力関係を適応的に調整することが可能であるが、一般的にパラメータの学習に長い時間が必要である。また、パラメータの推定がローカルミニマムに陥ることがあり、不安定になる場合がある。本研究では、RBFネットワーク⁶⁾⁷⁾⁸⁾を用いてむだ時間を含んだ非線形プラントを同定し、その結果を用いて制御系を設計する方法を提案する。さらに、提案法を空気圧シリンダの位置決め制御に適用し、その有効性を示す。空気圧シリンダは空気のもつ圧縮性や機器固有の摩擦などの影響により、むだ時間の大きい非線形プラントであり、高精度な位置決め制御が困難とされている。提案法を用いた実験では比較的良好な結果が得られている。

2. 制御系の設計

2.1 むだ時間の補償

いま、次式で表されるような n 次プラントを考える。

$$y_p(k+1) = f[y_p(k), y_p(k-1), \dots, y_p(k-n+1)] + \sum_{j=0}^{n-1} b_j u(k-j-L) \quad (1)$$

ただし、 $u(k)$, $y_p(k)$ はそれぞれプラントの入力、出力を表す。また、 L はむだ時間、 $f[\cdot]$ は非線形関数を表し、 $b_0 \neq 0$ とする。(1)式において、時間を1サンプル進める、すなわち $k \rightarrow k+1$ とすると

$$y_p(k+2) = f[y_p(k+1), y_p(k), \dots, y_p(k-n+2)] + \sum_{j=0}^{n-1} b_j u(k-j-L+1) \quad (2)$$

となる。ここで、 $y_p^{(m)}(k) = y_p(k+m)$, $m = 0, 1, \dots$ とすれば(2)式は、

$$y_p^{(1)}(k+1) = f[y_p^{(1)}(k), y_p^{(1)}(k-1), \dots, y_p^{(1)}(k-n+1)] + \sum_{j=0}^{n-1} b_j u(k-j-L+1) \quad (3)$$

と表すことができる。右辺の第1項鉤括弧内の $y_p^{(1)}(k)$ は、1サンプル先の予測値 $y_p(k+1)$ を表している。これは、(1)式が現在および過去の入出力系列 $\{u(k-L), u(k-L-1), \dots, u(k-L-n+1)\}$, $\{y_p(k), y_p(k-1), \dots, y_p(k-n+1)\}$ から予測値 $y_p(k+1)$ を表す式と考えると、 $y_p^{(1)}(k)$ は

$$y_p^{(1)}(k) = f[y_p(k), y_p(k-1), \dots, y_p(k-n+1)] + \sum_{j=0}^{n-1} b_j u(k-j-L) \quad (4)$$

より求めることができる。(3)式において、さらに $k \rightarrow k+1$ とおくと、

$$\begin{aligned} y_p^{(2)}(k+1) &= f[y_p^{(1)}(k+1), y_p^{(1)}(k), \dots, y_p^{(1)}(k-n+2)] \\ &\quad + \sum_{j=0}^{n-1} b_j u(k-j-L+2) \\ &= f[y_p^{(2)}(k), y_p^{(2)}(k-1), \dots, y_p^{(2)}(k-n+1)] \end{aligned}$$

*平成18年6月1日

**明治大学理工学部

(所在地 〒214-8571 川崎市多摩区東三田1-1-1)

$$+ \sum_{j=0}^{n-1} b_j u(k-j-L+2) \quad (5)$$

となる。さらに同じ動作を繰り返すと、

$$y_p^{(L)}(k+1) = f[y_p^{(L)}(k), y_p^{(L)}(k-1), \dots, y_p^{(L)}(k-n+1)] \\ + \sum_{j=0}^{n-1} b_j u(k-j) \quad (6)$$

となり、むだ時間のない n 次プラントとなる。この式もまた現在および過去の入出力系列 $\{u(k), u(k-1), \dots, u(k-n+1)\}$, $\{y_p(k), y_p(k-1), \dots, y_p(k-n+1)\}$ から求めることができる。

2.2 制御系の設計

次に、モデル追従型制御系を構築する。標準2次系をモデルとすると、モデルの差分方程式は次式のようになる。

$$y_m(k+1) = -a_{m1}y_m(k) - a_{m2}y_m(k-1) \\ + b_{m0}r(k) + b_{m1}r(k-1) \quad (7)$$

ただし、 $r(k)$ は目標値、 $y_m(k)$ はモデルの出力を表す。ここで、追従誤差を $e(k) = y_m(k) - y_p^{(L)}(k)$ とすると、提案する設計法は、Narendra らの方法⁴⁾のように $\lim_{k \rightarrow \infty} e(k) = 0$ とするのではなく、定常状態において、次式が成り立つように制御系を設計する。

$$e(k+1) = e(k) \quad (8)$$

(8)式より

$$y_m(k+1) - y_p^{(L)}(k+1) = y_m(k) - y_p^{(L)}(k) \quad (9)$$

(6)式と(7)式を(9)式に代入して

$$-a_{m1}y_m(k) - a_{m2}y_m(k-1) \\ + b_{m0}r(k) + b_{m1}r(k-1) \\ - f[k+L] - b_0u(k) - \sum_{j=1}^{n-1} b_j u(k-j) \\ = y_m(k) \\ - f[k+L-1] - b_0u(k-1) - \sum_{j=1}^{n-1} b_j u(k-j-1) \quad (10)$$

ただし、 $f[k+L] = f[y_p^{(L)}(k), y_p^{(L)}(k-1), \dots, y_p^{(L)}(k-n+1)]$ である。ここで、(9)式の右辺第1項の $y_m(k)$ が置き換えられていないことに注意されたい。 $y_m(k) \rightarrow y_p^{(L)}(k)$ として、プラントの入力 $u(k)$ を求めると

$$u(k) = u(k-1) \\ + \frac{1}{b_0} \left\{ -(a_{m1}+1)y_p^{(L)}(k) - a_{m2}y_p^{(L)}(k-1) \right. \\ + b_{m0}r(k) + b_{m1}r(k-1) - f[k+L] + f[k+L-1] \\ \left. - \sum_{j=1}^{n-1} b_j [u(k-j) - u(k-j-1)] \right\} \quad (11)$$

となる。非線形関数 $f[\cdot]$ は、RBF ネットワークによってプラントの入出力データからオフラインで同定する。提案する方法は簡単な設計法であるが⁸⁾、Narendra らによる方法⁹⁾と同様な結果が得られている。Narendra らは、様々なシミュレーション実験より有効性を示しているが、実プラントへの適用例を述べていない。さらに、むだ時間に対する対策も講じていない。空気圧シリンダのようにむだ時

間の大きいプラントは、十分な対策が必要である。

さて、本設計法の安定性についてであるが、非線形制御系の安定性の解析は一般的に困難である。閉ループ系が安定であるためには、まず制御器が安定である必要がある。本研究における制御器は、以下のように求めることができる。(11)式において、

$$v(k+1) \triangleq -(a_{m1}+1)y_p^{(L)}(k) - a_{m2}y_p^{(L)}(k-1) \\ + b_{m0}r(k) + b_{m1}r(k-1) - f[k+L] + f[k+L-1] \quad (12)$$

とおく。(11)式の Z 変換をとると

$$\frac{U(z)}{V(z)} = \frac{1}{1-z^{-1}} \frac{1}{\sum_{j=0}^{n-1} b_j z^{-j}} \quad (13)$$

となる。(13)式が安定であるためには、離散時間系プラントの全ての零点が単位円上および単位円内にあること、すなわち安定零点でなければならない。一方、非線形項が完全に同定できれば閉ループ系の入出力の関係を表す差分方程式は、(11)式を(6)式に代入して

$$y_p^{(L)}(k+1) = -a_{m1}y_p^{(L)}(k) - a_{m2}y_p^{(L)}(k-1) \\ + b_{m0}r(k) + b_{m1}r(k-1) \quad (14)$$

となり、非線形項が除去されてモデルの応答に一致する。非線形項が除去されない場合の安定性については、理論的解析が複雑である。なお、(14)式より初期誤差があっても、次式が成立する。

$$\lim_{k \rightarrow \infty} e(k) = 0 \quad (15)$$

(8)式では、必ずしも初期誤差を零と仮定していないが、操作量を(11)式のように設定した場合、定常偏差は零となる。このことは、シミュレーション実験からも明らかである。これは、(9)式の右辺第1項の $y_m(k)$ を(7)式によって置換しないことに起因するものと思われるが、理論的証明は安定性の解析と併せて稿を改めて述べたい。

3 非線形プラントの同定

非線形プラントの同定には、誤差逆伝播法を使った階層型ネットワークが広く使われている。しかし、階層型ネットワークを用いると、ウェイトの更新の際にかなりの計算量を必要とすることと、ネットワークパラメータが、ローカルミニマムに陥る可能性があることがよく知られている。本研究では、システム同定の際に非線形項を表す方法として、RBF ネットワークを用いたオフライン同定法を提案する。

3.1 RBFネットワーク

RBF ネットワークの構成を Fig. 1 に示す。階層型ネットワークと同様に、入力層、隠れ層、出力層の3層からなっている。隠れ層には、ガウス応答関数と呼ばれる非線形関数を用いている。

$$h_i = \exp \left(-\frac{\|x - c_i\|^2}{2\sigma^2} \right) \quad (16)$$

ただし、 x は入力ベクトル、 c_i はセントロイドベクトル、

Fig. 1 RBF networks

σ は正規化パラメータ、 $\|\cdot\|^2$ は L_2 ノルムを表している。隠れ層と出力層の間のウェイトを w_i とすると、RBFネットワークの出力 y は、次式で表すことができる。

$$y = \frac{1}{\delta} \sum_{i=1}^{N_h} h_i w_i \quad (17)$$

ただし、 δ は定数である。

3.2 非線形プラント

(1)式 of 非線形関数 $f[k]$ は、RBFネットワークの出力と入力ベクトル $x(k) = [y_p(k), y_p(k-1), \dots, y_p(k-n+1)]^T$ で置き換えることができる。

$$f[k] = f[x^T(k)] = \frac{1}{\delta} \sum_{i=1}^{N_h} h_i(k) w_i \quad (18)$$

ただし、 $h_i(k)$ は(16)式で与えられる。このとき、むだ時間を含んだ非線形プラントの差分方程式は、次のように表すことができる。

$$y_p(k+1) = \frac{1}{\delta} \sum_{i=1}^{N_h} h_i(k) w_i + \sum_{j=0}^{n-1} b_j u(k-j-L) \quad (19)$$

ただし、 L はむだ時間、プラントの入力 $u(k)$ は、(11)式で与えられる。

3.3 パラメータの推定

(19)式のパラメータ b_j とウェイト w_i を推定するために、次式のような関係を考える。

$$AW = Y \quad (20)$$

ただし、

$$A = \begin{bmatrix} u(k-L-N+1) & u(k-L-N) & \dots & u(k-L-N-n+2) \\ u(k-L-N+2) & u(k-L-N+1) & \dots & u(k-L-N-n+3) \\ \vdots & \vdots & \ddots & \vdots \\ u(k-L) & u(k-L-N+1) & \dots & u(k-L-n+1) \end{bmatrix}$$

$$W = \begin{bmatrix} \frac{1}{\delta} h_1(k-N+1) & \frac{1}{\delta} h_2(k-N+1) & \dots & \frac{1}{\delta} h_{N_h}(k-N+1) \\ \frac{1}{\delta} h_1(k-N+2) & \frac{1}{\delta} h_2(k-N+2) & \dots & \frac{1}{\delta} h_{N_h}(k-N+2) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{\delta} h_1(k) & \frac{1}{\delta} h_2(k) & \dots & \frac{1}{\delta} h_{N_h}(k) \end{bmatrix}$$

$$W = [b_0, b_1, \dots, b_{n-1}, w_1, w_2, \dots, w_{N_h}]^T$$

$$Y = [y_p(k-N+2), y_p(k-N+3), \dots, y_p(k+1)]^T$$

ただし、 N はデータの数である。また、 $k < 0$ の時は、 $u(k) = 0$ である、一般に、行列 A は正方行列ではないので、最小二乗法を用いると、 W は

$$W = (A^T A)^{-1} A^T Y \quad (21)$$

で求めることができる。しかし、本研究のようにデータ

の数が多の場合、 $(A^T A)^{-1}$ を正確に求めるのが困難である。そこで、本研究では特異値分解(SVD)を用いる。SVDを用いると、 A は次のように書き換えることができる。

$$A = USV^T \quad (22)$$

ただし、 U と V は、それぞれ(22)式を満たすユニタリ行列である。非負の対角要素を降順にもつ対角行列 S は、次式のように表すことができる。

$$S = \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} \quad (23)$$

この時、 W は以下のように求めることができる。

$$W = VS_r U^T Y \quad (24)$$

ただし、

$$S_r = \begin{bmatrix} \Sigma^{-1} & 0 \\ 0 & 0 \end{bmatrix} \quad (25)$$

である。

3.4 操作量の算出

本節では(11)式で与えられた操作量を求めるための計算手順の概要を述べる。まず、RBFネットワークの入力ベクトルとして $x(k) = [y_p(k), y_p(k-1), \dots, y_p(k-n+1)]^T$ を与え、(18)式から $f[k]$ を求める。次に、過去の入力系列 $\{u(k-L), u(k-L-1), \dots, u(k-L-n+1)\}$ を使って(19)式から1サンプル先のプラント出力の予測値 $y_p(k+1) = y_p^{(1)}(k)$ を求める。さらに、 $k \rightarrow k+1$ とおいた(19)式から、RBFネットワークの入力ベクトルと入力系列を1サンプル進めて、それぞれ $x(k+1) = [y_p^{(1)}(k), y_p(k), \dots, y_p(k-n+2)]^T$, $\{u(k-L+1), u(k-L), \dots, u(k-L-n+2)\}$ として2サンプル先の予測値 $y_p(k+2) = y_p^{(2)}(k)$ を求める。以下、同様にして $y_p(k+L) = y_p^{(L)}(k)$ まで求めていく。(11)式における $f[k+L]$ はRBFネットワークの入力ベクトルを $x(k+L) = [y_p^{(L)}(k), y_p^{(L)}(k-1), \dots, y_p^{(L)}(k-n+1)]^T$ として求める。 $f[k+L-1]$ も同様に求め、操作量 $u(k)$ を算出する。

4. シミュレーション実験

モデルの伝達関数は、次式のような標準2次系とする。

$$G_m(s) = \frac{\omega_m^2}{s^2 + 2\zeta_m \omega_m s + \omega_m^2} \quad (26)$$

ただし、固有振動数 $\omega_m = 2.0$ [rad/sec], $\zeta_m = 0.7$ である。モデルの差分方程式は、零次ホールド法¹⁰⁾を用いて離散化することによって求めることができる。ここで、サンプリング周期は20[msec]で、モデルのパラメータは $a_{m1} = -1.944$, $a_{m2} = 9.455 \times 10^{-1}$, $b_{m0} = 7.852 \times 10^{-4}$, $b_{m1} = 7.707 \times 10^{-4}$ となる。シミュレーションに用いたセントロイドの数 N_h は20、同定に用いたサンプル数は1000、(16)式の σ と(19)式の δ は、それぞれ1.0、20とした。便宜上、これらの値は実験的に求めた。

シミュレーション実験では、むだ時間を含んだ非線形プラントの差分方程式を

Fig. 2 Simulation result

$$y_p(k+1) = -a_1 y_p(k) - a_2 y_p(k-1) - \frac{y_p(k)}{1 + y_p^2(k)}$$

$$+ b_0 u(k-L) + b_1 u(k-1-L) \quad (27)$$

とした時のシミュレーション実験を行った。ただし、 $a_1 = -1.762$, $a_2 = 7.623 \times 10^{-1}$, $b_0 = 1.309 \times 10^{-2}$, $b_1 = 1.195 \times 10^{-2}$, $L = 2$ である。なお、(27)式の非線形項 $\frac{y_p(k)}{1 + y_p^2(k)}$ は、ニューロ制御の分野でよく使われるため、シミュレーション実験の一例として用いた。この時のシミュレーション結果を Fig. 2 に示す。

シミュレーション結果からわかるように、比較的良好な追従特性が得られている。シミュレーション実験では、問題を単純にするために2次系のプラントに限定した。同様の設計法を行えば、高次系のプラントも容易に設計することができる。提案法は、任意の非線形項を近似するためにRBFネットワークを用いている。従来、非線形項の近似には階層型ネットワークがよく用いられているが、本研究では安定な学習性能と高速なトレーニングのためにRBFネットワークを使用した。

5. 実機実験

実機実験では、提案法を用いて空気圧シリンダの位置決め制御を行った。空気圧シリンダは、比較的大きな出力が得られるので、産業界で広く使われている。しかし、非線形性やむだ時間が大きいため、現在もPID制御を使用しているのが実情である。そこで、精度の高い位置決め制御ができれば、様々なところで応用できるものと考えた。

5.1 空気圧シリンダ

Fig. 3 は、実験に用いた制御装置の構成図である。本装置は、演算部の32ビットパーソナルコンピュータ（ターゲットPC）、12ビットA/D・D/A変換器、24ビットカウンタ回路、パワーアンプ（電圧-電流変換型サーボアンプ）、電空比例制御弁などにより構成されている。空気圧シリンダは、チューブ内径 $\phi 20$ [mm]、ストローク長 100

Fig. 3 Structure of the pneumatic servo system

[mm] の片ロッド形複動シリンダである。ピストンロッド先端の荷台には0.5[kg]の負荷を積載し、電空比例制御弁には3ポート形の電磁比例流量制御弁（サーボ弁）を使用している、また、位置検出用のセンサには分解能 10 [$\mu\text{m}/\text{pulse}$] の磁気式インクリメンタル型エンコーダを使用している。このエンコーダは、磁性目盛付ロッドと磁気センサによりピストン変位を検出し、A 相/B 相の 90° 位相差方式でパルス信号を出力する。このパルス信号をカウンタ回路を用いてカウントすることによって現在位置を検出することができる。空気圧シリンダ内の圧力は、ヘッド側、ロッド側それぞれに圧力センサが取り付けられていて、このセンサにより電圧値で出力され、A/D 変換器を介して検出される。なお、使用圧力は 0.5[MPa] である。検出された全ての信号はターゲット PC に取り込まれ、ソフトウェア上で操作量を計算し、D/A 変換器に出力される。この信号は、サーボアンプ、サーボ弁に入力され、空気圧シリンダを駆動させる。サーボアンプには、制御弁の静止摩擦低減のため、ディザ信号（周波数：100[Hz]、振幅：0.1[V]）を加えている。

5.2 実験結果

実機実験において、サンプリング周期を 5 [msec] で離散化したとき、空気圧シリンダの差分方程式は、次式で表されると仮定した。

$$y_p(k+1) = f[y_p(k), y_p(k-1)] + b_0 u(k-L) + b_1 u(k-1-L) \quad (28)$$

ただし、関数 $f[k]$ は、実際の状況を考慮して非線形であると仮定した。これは、空気圧シリンダが空気のもつ圧

Fig. 4 Experimental result

縮性や摺動部における摩擦力などによる非線形性が強いので、RBF ネットワークにより近似することにした。セントロイドの数 N_h は 20、同定に用いたサンプル数は 5000、 σ と δ は、それぞれ 10.0、20 とした。むだ時間については、15[msec] とした。また、モデルを決定する ω_m 、 ζ_m については、それぞれ 20.0[rad/sec]、1.0 とした。Fig. 4 に実験結果を示す。Fig. 4 より、学習を重ねるにつれてオーバーシュート量が増えているものの（この後ほぼ一定となる）、比較的良好な結果が得られている。シミュレーション実験から明らかなように、非線形項の同定が良好であれば、オーバーシュート量が減少していくはずである。今後は、同定精度の改善に努めたい、一方、整定時における位置決め精度は、実測によれば最大分解能（10[μ m]）程度であった。

6. ま と め

本研究では、RBF ネットワークを用いてむだ時間を含んだ非線形プラントを同定し、その結果を用いて制御系を設計する方法を提案した。提案した設計法は、Narendra らの方法と比べて極めて容易な方法であるが、シミュレーション実験においては、ほぼ同様な結果が得られている。しかし、実機実験においては、空気圧シリンダの動特性、とりわけ非線形項の同定は難しく、実機実験では RBF ネットワークの導入による効果が明確に現れていない。今後は、同定精度の改善に努め、提案法の有効性を高めていきたい。

謝 辞

本研究にあたり、機材面でご援助頂いた SMC 株式会社

に厚く謝意を申し上げます。また、有益なご助言を頂いた査読者の方々に感謝の意を表します。

参 考 文 献

- 1) E. H. Mamdani: Applications of fuzzy algorithms for control of simple dynamic plant, Proc. IEEE, vol. 121, no. 12, 1585/1588 (1975)
- 2) T. Okayasu, T. Matsuoka and Y. Ishida: Model reference I-PD controller using fuzzy reasoning, Proc. FUZZY-IEEE'94, vol. 1, 535/538 (1994)
- 3) S. Haykin: Neural Networks-A Comprehensive, Macmillan, New York (1994)
- 4) K. S. Narendra and K. Parthasarathy: Identification and control of dynamical systems using neural networks, IEEE trans. Neural Networks, vol. 1, no. 1, 4/27 (1990)
- 5) 松熊・宋・石田・能登山：ニューラルネットワークを用いた空気圧シリンダの PID 制御，日本フルードパワーシステム学会論文誌，第 30 巻，第 1 号，28 / 33 (1999)
- 6) D. F. Specht: Probabilistic neural networks, Neural networks, vol. 3, 109/118 (1990)
- 7) Chen Xiaohong, Gao Feng, and Qian Jixin: Adaptive control based on RBF networks, Proc. the 35th IEEE. on Decision and Control, vol. 4, 3810/3815 (1996)
- 8) Xue-Song, Wang, Yu-Hu Cheng, and Wei Sum: Constructive RBF network based iterative learning controller for manipulators, Proc. 2005 Int. Conf. on a Machine Learning and Cybernetics, vol. 8, 4714/4719 (2005)
- 9) S. Mu khopadhyay and K. S. Narendra: Disturbance rejection in nonlinear system using neural networks, IEEE Trans. Neural Networks, vol. 4, no. 1, 63/72 (1993)
- 10) G. F. Franklin, J. D. Powell, and M. L. Workman: Digital Control of Dynamic Systems, Addison-Wesley, New York (1990)