

ニューラルネットワークを利用した非線形システムの簡易型適応制御の一方法

A Method of Simple Adaptive Control for Nonlinear Systems Using Neural Networks

学 潘 俊賢 (千葉大) 正 呂 建明 (千葉大) 正 谷 萩 隆嗣 (千葉大)

Jiunshian PHUAH, Chiba University, Graduate School of Science and Technology,
Chiba 263-8522, Japan
Jianming LU, Chiba University
Takashi YAHAGI, Chiba University

This paper presents a method of simple adaptive control(SAC) for nonlinear systems using neural networks. The control input is given by the sum of the output of a simple adaptive controller and the output of the neural network. The neural network is used to compensate the nonlinearity of plant dynamics that is not taken into consideration in the usual simple adaptive control. The role of the neural network is to construct a linearized model by minimizing the output error caused by nonlinearities in the control systems.

Key Words: Simple Adaptive Control, Nonlinear System, Neural Network

1 はじめに

確定系の制御対象に対する適応制御としては、モデル規範適応制御 (MRACS) が代表的かつ実用的な手法となっている。近年、制御系の構成がきわめて簡単な簡易型適応制御 (SAC : Simple Adaptive Control) が注目されている。これは、制御系の制御器が制御対象の次数とはほとんど無関係に決定でき、適応則がきわめて簡単で多入出力系に対する実装化もきわめて容易になるという大きな特徴がある。

一方、上述した SAC はあくまでも線形制御対象に対して有効である。上述した SAC は非線形性を持つ制御対象にはうまく対応できない^{(1)~(4)}。そこで、ニューラルネットワーク (NN) が非線形問題を解くのに適した特徴を有しているため、本研究では、NN を利用して非線形性を補償することにより非線形システムの SAC を行う方法を提案する^{(5)~(7)}。また、NN を利用することにより、学習時間が大きいという問題がある。そこで、NN の重みの学習に必要な演算量を減少させるために、NN を多重分割して、NN の学習時間を減らすことのできる方法を述べる。

2 線形 S A C の構成

まず、離散時間 S A C のアルゴリズムを示す⁽³⁾。制御対象が次の可制御・可観測な n_p 次 1 入出力線形系で与えられているとする。

$$\mathbf{x}_p(k+1) = A_p \mathbf{x}_p(k) + \mathbf{b}_p u_p(k) \quad (1)$$

$$y_p(k) = \mathbf{c}_p^T \mathbf{x}_p(k) \quad (2)$$

$$\tilde{y}_p(k) = \mathbf{c}_p^T \mathbf{x}_p(k) + d_p u_p(k) = y_p(k) + d_p u_p(k) \quad (3)$$

ここで、 $A_p, \mathbf{b}_p, \mathbf{c}_p, d_p$ はそれぞれ適切な次数の行列、ベクトル、スカラーであり、 $\mathbf{x}_p(k)$ は $\mathbf{x}_p(k) \in R^{n_p}$ なる状態ベクトルである。 $u_p(k)$ は制御対象の入力であり、また $y_p(k)$ は実際の制御対象の出力、 $\tilde{y}_p(k)$ は前向き要素を付加して得られる拡大系の出力である。

この制御対象は次の仮定を満足しているとする^{(3)~(4)}。
< 仮定 1 >

1. 式 (1)、式 (3) は ASPR である。

2.

$$\begin{bmatrix} A_p & \mathbf{b}_p \\ \mathbf{c}_p^T & d_p \end{bmatrix} \quad (4)$$

は正則である。

また、追従すべき n_m 次 ($n_m \leq n_p$) の漸近安定な規範モデルをつぎのように表す。

$$\mathbf{x}_m(k+1) = A_m \mathbf{x}_m(k) + \mathbf{b}_m u_m(k) \quad (5)$$

$$y_m = \mathbf{c}_m^T \mathbf{x}_m(k) + d_m u_m(k) \quad (6)$$

ここで、 $A_m, \mathbf{b}_m, \mathbf{c}_m, d_m$ はそれぞれ適切な次数の行列、ベクトル、スカラーであり、 $\mathbf{x}_m(k)$ は $\mathbf{x}_m(k) \in R^{n_m}$ なる状態ベクトルである。ここで、 $n_m \ll n_p$ であってもよい。

出力誤差および拡張出力誤差をそれぞれ

$$e_y(k) = y_m(k) - y_p(k) \quad (7)$$

$$\tilde{e}_y(k) = y_m(k) - \tilde{y}_p(k) = e_y(k) - d_p u_p(k) \quad (8)$$

と定義すると、制御目的は、

$$\lim_{k \rightarrow \infty} \tilde{e}_y(k) = 0 \quad (9)$$

で表される。

< 定理 1 > 仮定 1 が成立するとき、式 (1)、式 (3) で表されるシステムに対して以下のアルゴリズムに従って入力を計算することにより、式 (9) で表される制御目的が達成される⁽³⁾。離散時間 SAC のアルゴリズムは次のようになる。

$$u_p(k) = k_e(k)e_y(k) + \mathbf{k}_x^T(k)\mathbf{x}_m(k) + k_u(k)u_m(k) \quad (10)$$

$$= K(k)\mathbf{r}(k) \quad (11)$$

$$K(k) = [k_e(k) \quad \mathbf{k}_x^T(k) \quad k_u(k)] \quad (12)$$

$$\mathbf{r}(k) = [e_y(k) \quad \mathbf{x}_m^T(k) \quad u_m(k)]^T \quad (13)$$

$$K(k) = K_p(k) + K_i(k) \quad (14)$$

$$K_p(k) = \tilde{e}_y(k)\mathbf{r}^T(k)T_p \quad (15)$$

$$K_i(k) = K_i(k-1) + \tilde{e}_y(k)\mathbf{r}^T(k)T_i \quad (16)$$

$$(T_p = T_p^T > 0, T_i = T_i^T > 0)$$

また、式 (15)、式 (16) の適応則には $\tilde{e}_y(k)$ が含まれているため、上記のアルゴリズムを直接実装することはできない。そこで式 (8) に式 (11)、式 (14)~式 (16) を代入して整理してみると次式が得られる。

$$\tilde{e}_y(k) = \{1 + d_p(r^T(k)(T_p + T_i)r(k))\}^{-1} \cdot \{e_y(k) - d_p K_i(k-1)r(k)\} \quad (17)$$

$$u_m(k) = d^{-1}(e_m(k) - \tilde{e}_m(k)) \quad (18)$$

図 1

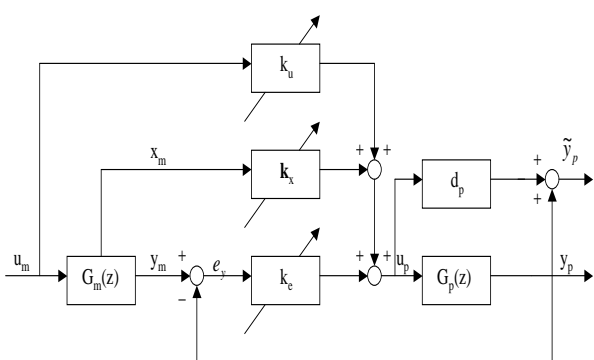


Figure 1: Block diagram of SAC⁽³⁾

3 非線形 SAC の構成

制御対象の入出力特性が非線形有的时候には、式 (1)、式 (3) のように表現することはできない。そこで、非線形関

数 $h(\cdot)$ を用いて、出力 $\tilde{y}_p(k)$ が式 (19) で与えられる場合を考える。

$$\tilde{y}_p(k) = h(\tilde{y}_p(k-1), \dots, \tilde{y}_p(k-n), u_p(k-1), \dots, u_p(k-m)) \quad (19)$$

前章で述べた制御則は、制御対象として線形化されたシステムを前提としている。このため制御対象が非線形性を有する場合には、制御誤差を生じてしまうことが問題となる。そこで、制御対象の非線形性によって生じる制御誤差を NN によって補償する方法を提案する。制御の目的は、システムが非線形特性を有する場合でも、制御対象の出力 $\tilde{y}_p(k)$ が目標値 $y_m(k)$ に追従できるようにすることである。

制御目的は、 $\tilde{y}_p(k)$ を $y_m(k)$ に追従させるための制御入力を次式のように合成することを考える。

$$u_p(k) = v(k) + \bar{v}(k) \quad (20)$$

但し、 $v(k)$ は SAC の制御器の出力、 $\bar{v}(k)$ は NN の出力である。 $v(k)$ と $\bar{v}(k)$ は次式で表せる。

$$v(k) = K(k)\mathbf{r}(k) \quad (21)$$

$$\bar{v}(k) = \hat{h}(v(k-1), y_m(k+1), \tilde{y}_p(k-1), \dots, \dots)$$

図 2

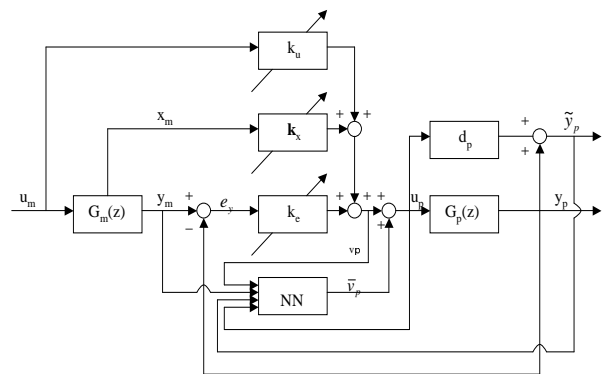


Figure 2: Block diagram of SAC with neural network

4 NN の構成

NN は入力層、中間層、出力層からなる 3 層構造の NN を用いることにする。ここで、入力層の i 番目の素子への入力を $x_i(k)$ 、中間層の j 番目の素子への入力を $p_j(k)$ とし、出力層への入力を $q_1(k)$ で表すことにする。また、入力層と中間層との間の結合係数を $w_{ij}(p)$ 、中間層と出力層との間の結合係数を $w_{j1}(p)$ と定義する。

3 層構造の NN の出力 $\bar{v}(k)$ と SAC の制御器の出力 $v(k)$ の和が、制御入力 $u_p(k)$ として扱われ、それに応じて出力

$\tilde{y}_p(k)$ が得られる. この $\tilde{y}_p(k)$ を目標値 $y_m(k)$ に一致させるように NN の学習が行われる.

本研究では NN への入力 $x(k)$ は, 式 (22) より

$$x(k) \equiv \{v(k-1), y_m(k+1), \tilde{y}_p(k-1), \dots, \tilde{y}_p(k-n), u_p(k-1), \dots, u_p(k-m)\} \quad (23)$$

で与えられる. 従って, NN により多入力 1 出力の非線形関数

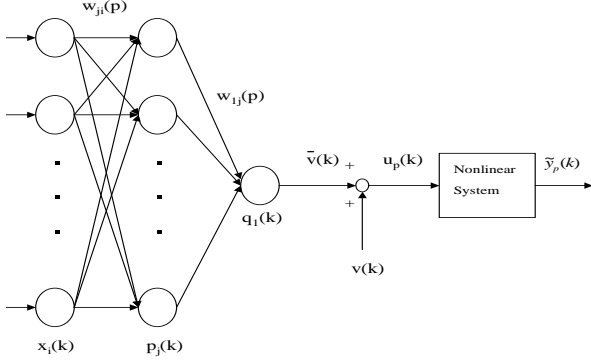


Figure 3: System configuration with neural network

5 NN の学習

シグモイド関数 $f(\cdot)$ を用意すると, 図 3 より

$$p_j(k) = \sum_i w_{ji}(k)x_i(k) \quad (24)$$

$$q_1(k) = \sum_j w_{1j}(k)f(p_j(k)) \quad (25)$$

$$\bar{v}(k) = f(q_1(k)) \quad (26)$$

が得られる.

また, NN の学習のための誤差関数 $E(k)$ を式 (27) で与える.

$$E(k) = \frac{1}{2}[y_m(k) - \tilde{y}_p(k)]^2 \quad (27)$$

以上の関係式から, バックプロパゲーションアルゴリズムを利用して NN の結合係数の調整を行う. そして, 結合係数の調整方法は一括慣性修正法を用いる.

従って, NN の結合係数は

$$\Delta w_{1j}(k) = -\eta \frac{\partial E(k)}{\partial w_{1j}(k)} + \Delta w_{1j}(k-1) \cdot \alpha(k) \quad (28)$$

$$\Delta w_{ji}(k) = -\eta \frac{\partial E(k)}{\partial w_{ji}(k)} + \Delta w_{ji}(k-1) \cdot \alpha(k) \quad (29)$$

により修正する. 但し, η は学習係数, $\alpha(k)$ は慣性係数を表し, 1 ステップ前の値を $\alpha(k-1)$, 増分を $\Delta\alpha$ として, 一定値 A まで $\alpha(k) = \alpha(k-1) + \Delta\alpha$ のように, 学習回数とともに変化させる. $\frac{\partial E(k)}{\partial w_{1j}(k)}$, $\frac{\partial E(k)}{\partial w_{ji}(k)}$ は次のように計算する.

$$\frac{\partial E(k)}{\partial w_{1j}(k)} = \frac{\partial E(k)}{\partial \tilde{y}_p(k)} \cdot \frac{\partial \tilde{y}_p(k)}{\partial u_p(k-1)} \cdot \frac{\partial u_p(k-1)}{\partial f(q_1(k))}$$

$$\frac{\partial f(q_1(k))}{\partial q_1(k)} \cdot \frac{\partial q_1(k)}{\partial w_{1j}(k)} \quad (30)$$

$$\begin{aligned} \frac{\partial E(k)}{\partial w_{ji}(k)} &= \frac{\partial E(k)}{\partial \tilde{y}_p(k)} \cdot \frac{\partial \tilde{y}_p(k)}{\partial u_p(k-1)} \cdot \frac{\partial u_p(k-1)}{\partial f(q_1(k))} \\ &\quad \cdot \frac{\partial f(q_1(k))}{\partial q_1(k)} \cdot \frac{\partial q_1(k)}{\partial f(p_j(k))} \cdot \frac{\partial f(p_j(k))}{\partial p_j(k)} \\ &\quad \cdot \frac{\partial p_j(k)}{\partial w_{ji}(k)} \end{aligned} \quad (31)$$

式 (30) と式 (31) の計算においては制御対象の微分情報すなわち $\frac{\partial \tilde{y}_p(k)}{\partial u_p(k-1)}$ が必要となるので, 次式のように求められる.

$$\begin{aligned} \frac{\partial \tilde{y}_p(k)}{\partial u_p(k-1)} &= \frac{\partial \tilde{y}_p(k)}{\partial v(k-1)} \cdot \frac{\partial v(k-1)}{\partial u_p(k-1)} \\ &= \frac{\partial \tilde{y}_p(k)}{\partial v(k-1)} \Big/ \frac{\partial u_p(k-1)}{\partial v(k-1)} \end{aligned} \quad (32)$$

ここで, 図 3 の構造から, 制御入力 $u_p(k-1)$ は

$$u_p(k-1) = v(k-1) + \bar{v}(k-1) \quad (33)$$

で与えられるので, 式 (32) の分母は

$$\frac{\partial u_p(k-1)}{\partial v(k-1)} = 1 + \frac{\partial \bar{v}(k-1)}{\partial v(k-1)} \quad (34)$$

となる. 式 (34) で, $\partial \bar{v}(k-1)/\partial v(k-1)$ は NN の出力を入力で微分したものであるから, バックプロパゲーションを行った場合と同様に求めることができるため, 次のようになる.

$$\begin{aligned} \frac{\partial \bar{v}(k-1)}{\partial v(k-1)} &= \frac{\partial f(q_1(k))}{\partial x_1(k)} \\ &= \frac{\partial f(q_1(k))}{\partial q_1(k)} \cdot \frac{\partial q_1(k)}{\partial f(p_j(k))} \cdot \frac{\partial f(p_j(k))}{\partial p_j(k)} \cdot \frac{\partial p_j(k)}{\partial x_1(k)} \end{aligned} \quad (35)$$

6 NN の多重分割

本章では, NN を多重分割し, 小規模な NN を並列接続して利用する方法を検討する.

図 4 は NN を含めたシステムの入出力関係の概略図を示したものである. NN には入力層, 中間層, 出力層からなる 3 層構造のものを用いることにして, 小規模の NN を複数個並列に接続する. 但し, NN の大きさはすべて同じものとする.

ここで, $m(m = 1, 2, \dots, n)$ 番目の NN の入力層の i 番目の素子への入力を $x_i^{(m)}(k)$, 中間層の j 番目の素子への入力を $p_j^{(m)}(k)$ とし, 出力層への入力を $q_1^{(m)}(k)$ で表すことにする. また, 入力層と中間層との間の結合係数を $w_{ji}^{(m)}(p)$, 中間層を出力層との間の結合係数を $w_{1j}^{(m)}(p)$ で表す.

3 層構造の NN の出力の総和 $\sum_{m=1}^n \bar{v}^{(m)}(k)$ と SAC の制御器の出力 $v(k)$ の和が, 制御入力 $u_p(k)$ として扱われ, それに応じて出力 $\tilde{y}_p(k)$ が得られる. この $\tilde{y}_p(k)$ を目標値 $y_m(k)$ に一致させるように NN の学習が行われる. すな

わち、教師信号として $u_m(k)$ を利用し、出力誤差 $\tilde{e}_n(k) =$

y_f
す

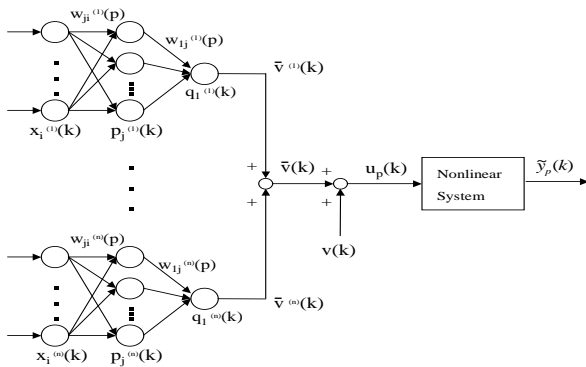


Figure 4: System configuration with neural network

7 むすび

本研究では、ニューラルネットワークを利用して非線形性を補償することにより非線形システムの適応制御を行う方法を提案した。また、NNの結合係数の学習に必要な演算量を減少させるために、NNを多重分割して、NNの学習時間を減らすことのできる方法を提案した。

References

- [1] I. Bar-Kana and H. Kaufman: "Global stability and performance of a simplified adaptive algorithm", Int. J. Control, Vol.42, No.6, pp.1491-1505, 1985.
- [2] I. Bar-Kana and H. Kaufman: "Simple adaptive control of uncertain systems", Int. J. Adaptive Control and Signal Processing, Vol.2, pp.133-143, 1988.
- [3] 柴田, 紅林: "簡易型適応制御に対する離散時間アルゴリズム", 計測自動制御学会論文集, Vol.31, No.2, pp.177-184, 1995.
- [4] 岩井, 大友, 水本: "構造の簡単なロバスト適応制御系", 計測自動制御学会論文集, Vol.27, No.3, pp.306-313, 1991.
- [5] J. Lu and T. Yahagi: "Adaptive control for nonlinear system using neural networks", Trans. IEE of Japan, Vol.119-C, No.3, pp.350-357, 1999.
- [6] J. Lu, J. Phuah and T. Yahagi: "A method of model reference adaptive control for MIMO nonlinear systems using neural networks", IEICE Trans. Fundamentals, Vol.E84-A, No.8, pp.1933-1941, 2001.

- [7] K. S. Narendra and K. Parthasarathy: "Identification and control of dynamical systems using neural networks", IEEE Trans. Neural Networks, Vol.1, No.1, pp.4-27, 1990.