

ニューラルネットワークを用いた制御系の設計

A Study of the Design for Neural Network Control System

○高橋吉徳*、大久保重範**

○Yoshinori Takahashi*、Shigenori Okubo**

*山形大学

*Yamagata University

キーワード：ニューラルネットワーク(neural network)、ニューラル同定器(neural estimator)

適応制御(adaptive control)、オンライン学習(on-line learning)

連絡先：〒992 米沢市城南 4-3-16 山形大学大学院 工学研究科 大久保研究室、

高橋吉徳、Tel. : (0238)26-3246、Fax. : (0238)24-6445、E-mail : takayosi@mip3470.yz.yamagata-u.ac.jp

1. はじめに

ニューラルネットワーク理論は近年活発な研究分野である。1980年代後半から著しい注目が、適応制御問題を含んだ制御システム設計へのニューラルネットワークの適用に払われた。この内容では、動的システム同定と制御へのニューラルネットワークの適用は全く自然な成り行きで発達した。なぜなら、ネットワークの学習課程の適応能力のためである。

適応制御問題に対処するためにニューラルネットワークの文献で2つの基本的方法が紹介されている¹⁾。第一に、あらかじめプラントの入出力信号を測定し、いくつかの重要な場合のプラント動作をニューラルネットワークで同定するため、オフラインで学習される。そのとき、制御はこのニューラルネットワークを用いて実行される²⁾。第二の研究は、適応学習が実行され、ニューラルネットワークの出力が制御入力としてオンラインで決定される。これらの種類のコントローラは次

のような方法を用いて構成される。1つはいわゆる間接学習法で、これは逆ニューラルモデルをオンラインで学習する方法である。2つ目は直接学習法で、ニューラルコントローラを構成するため通常のモデルパラメータが使用される^{4,5,6,7,8)}。

本研究では上で述べたような直接学習法を用いた適応学習法を採用する。未知のプラントのためのモデルとして2層のニューラルネットワーク(以後ニューラル同定器)を用いる。ニューラル同定器は、プラントの実際の出力信号とネットワークによる予想出力との誤差を最小にするように、Widrow-Hoff デルタルールを用いて、システムの通常のダイナミクスをオンラインで学習する。この学習は、コントローラとして用いられる二番目の2層のニューラルネットワーク(以後ニューラルコントローラ)の結合荷重を調整するために用いられる。さらに、3層のニューラルネットワーク(以後 NN 補償器)を微調整のために平行して用い、学習の際にはニューラル同定器のパラメータを用いて学習を実行する。

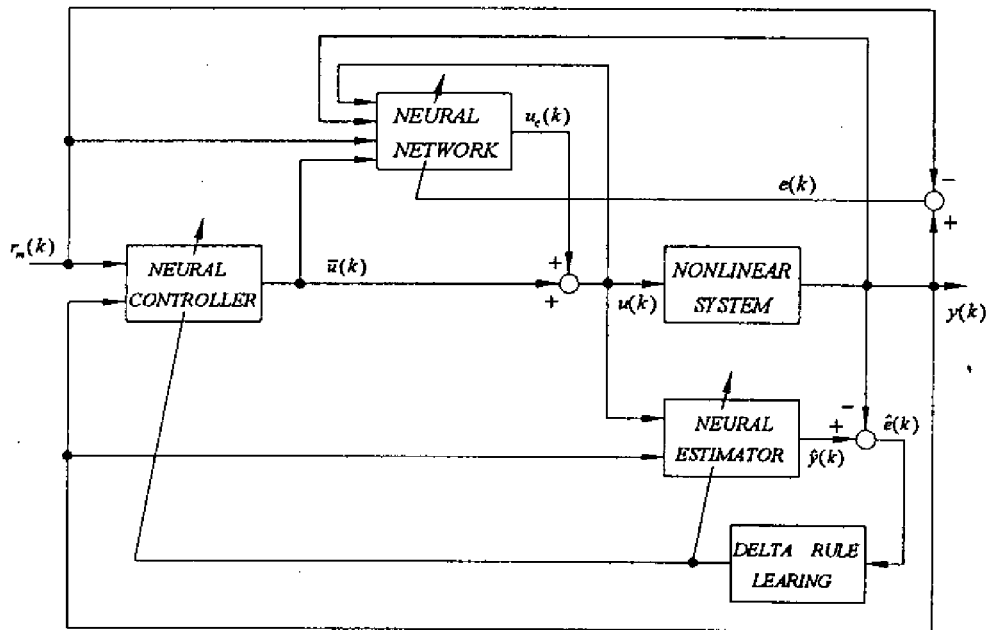


Fig.1 Neural control system

2. プラントモデル

次のような未知線形化プラントを考える。

$$A(q^{-1})y(k) = B(q^{-1})u(k) \quad (1)$$

ここで、 $A(q^{-1})$ と $B(q^{-1})$ は次のように定義された多項式である。

$$\left. \begin{aligned} A(q^{-1}) &= 1 + a_1 q^{-1} + \dots + a_n q^{-n} \\ B(q^{-1}) &= b_1 q^{-1} + \dots + b_m q^{-m} \end{aligned} \right\} \quad (2)$$

q^{-1} は微分オペレータ（例えば、 $q^{-1}y(k) = y(k-1)$ ）で、 $u(k)$ 、 $y(k)$ はそれぞれプラント入力、出力である。

2.1 仮定

プラントに3つの基本的仮定をする。

A1: 次元 n' 、 m' の上限（それぞれ n 、 m ）は既知である。

A2: $B(q^{-1})$ は安定多項式である。

A3: 係数 $b_1 \neq 0$ である。

3. プラントのニューラル同定器

プラント同定器として線形2層ニューラルネットワークを用いる。このネットワークは Widrow Hoffデルタルールを用いてプラントのダイナミクスを学習する。入力層は $n+m$ エレメントからなる。これらの $n+m$ 入力はプラントの前時刻で測定された入出力信号である。出力層は1つのエレメントからなり、その出力はプラントの予想出力になる。Fig.2はプラントニューラル同定器を示す。

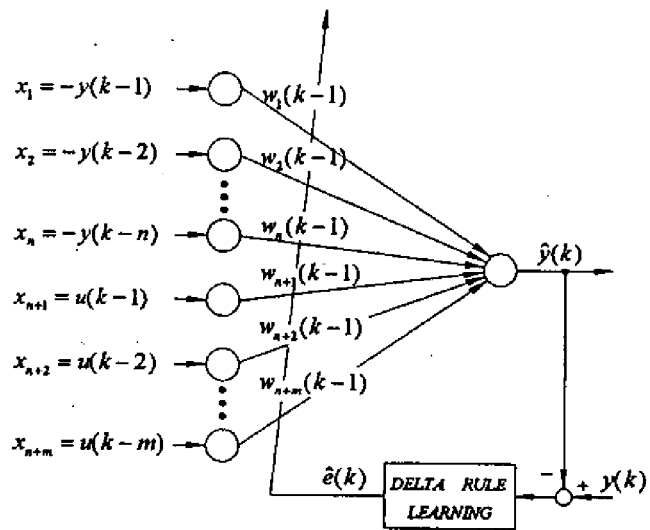


Fig.2 Neural estimator of the plant

k時刻荷重ベクトルは次のように定義される。

$$w(k) = [w_1(k), \dots, w_n(k), w_{n+1}(k), \dots, w_{n+m}(k)]^T \quad (3)$$

ここで、 $x(k)$ はネットワーク入力ベクトルで、この場合次のように定義される。

$$\begin{aligned} x(k) &= [x_1(k), \dots, x_n(k), x_{n+1}(k), \dots, x_{n+m}(k)]^T \\ &= [-y(k), \dots, -y(k-n+1), u(k), \\ &\quad \dots, u(k-m+1)]^T \end{aligned} \quad (4)$$

この予想出力は次のように計算される。

$$\hat{y}(k) = \sum_{i=1}^{n+m} w_i(k-1)x_i(k-1) \quad (5)$$

Fig.1 で示すように同定誤差 $\hat{e}(k)$ は実プラント応答 $y(k)$ とニューラル同定器による予想応答 $\hat{y}(k)$ との誤差である。ここで、Widrow Hoff ルールを用いて荷重を更新する。

$$w(k+1) = w(k) + \frac{\alpha \hat{e}(k+1)x(k)}{\varepsilon + x^T(k)x(k)} \quad (6)$$

α は $0 \leq \alpha \leq 2$ の学習率、定数 ε はゼロに近い値を選択される。(6)式で、もし $x^T(k)x(k) = 0$ のときゼロで割るのを避けるために含まれる。適応アルゴリズム(6)式は誤差を最小にする。

4. ニューラルコントローラ

上で提案したニューラル同定器によるプラントのダイナミクスの学習は制御信号 $\bar{u}(k)$ を生成するニューラルコントローラの結合荷重を調整するために用いられる。プラント入力にこの制御信号が適用されると、プラント出力 $y(k)$ は設定値 $r_m(k)$ に近づく傾向にある。

ニューラルコントローラは入力層に $(n+m)$ のエレメントと出力層に 1 エレメントを持つ 2 層ニ

ューラルネットワークから成る。時刻 k の入力ベクトルは次のように定義される。

$$\begin{aligned} z(k) &= [r_m(k+1), -x_1(k), -x_2(k), \\ &\quad \dots, -x_n(k), -x_{n+2}(k), \dots, -x_{n+m}(k)]^T \end{aligned} \quad (7)$$

結合荷重ベクトル $w'(k)$ は次のようにプラントのニューラル同定器の調節可能荷重 $w_i (i=1, \dots, n+m)$ の関数として定義される。

$$\begin{aligned} w'(k) &= \frac{1}{w_{n+1}(k)} [1, w_1(k), w_2(k), \dots, w_n(k) \\ &\quad w_{n+2}(k), \dots, w_{n+m}(k)]^T \end{aligned} \quad (8)$$

Fig.3 はニューラルコントローラを示す。

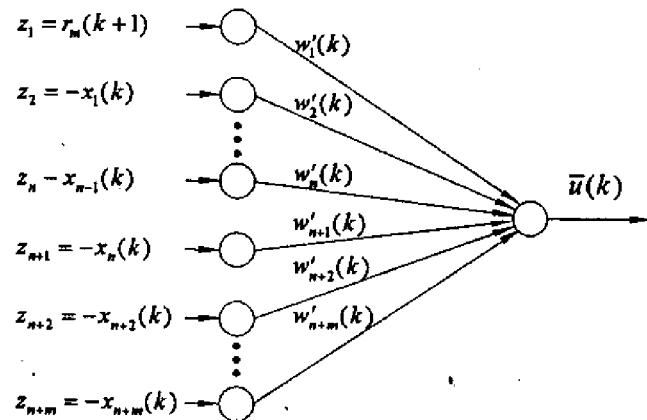


Fig.3 Neural controller

制御信号は新荷重 $w'_i(k)$ を使って入力ベクトル $z(k)$ のエレメントとの線形結合として生成される。

$$\bar{u}(k) = \sum_{i=1}^{n+m} w'_i(k)z_i(k) \quad (9)$$

5. NN補償器の構成

NN 補償器として多入力 1 出力の 3 層構造 $[M, N, 1]$ ニューラルネットワークを用いる。NN 補償器への i 番目の入力を $I_i(k)$ 、入力層 i 番目の

ニューロンと中間層 j 番目のニューロン間の結合荷重を $W_{ji}(k)$ 、中間層 j 番目のニューロンを出力層のニューロン間の結合荷重 $V_j(k)$ としたとき、

NN補償器の出力は、

$$u_c = \sum_{j=1}^N V_j(k) H_j(k) \quad (10)$$

$$H_j(k) = f\left(\sum_{i=1}^M W_{ji}(k) I_i(k)\right)$$

により表現できる。本研究では $[5,20,1]$ 構造の 3 層ニューラルネットワークを用い、入力は $r_m(k), y(k), \bar{u}(k), u(k-1), u(k-2)$ とする。また、 $f(\cdot)$ は応答範囲が $[-1,+1]$ のシグモイド関数を表し、 $f(x) = (1 - \exp(-x)) / (1 + \exp(-x))$ で定義される。

6. NN補償器の学習

NN補償器の学習は、制御対象の出力と目標値との二乗誤差が最小になるように行われる。すなわち、

$$e(k) = \frac{1}{2} (r_m(k) - y(k))^2 \quad (11)$$

を最小にする重みの更新則は、最急降下法を用いて、

$$W_{ji}(k+1) = W_{ji}(k) - \eta \frac{\partial e(k)}{\partial W_{ji}(k)} \quad (12)$$

$$V_j(k+1) = V_j(k) - \eta \frac{\partial e(k)}{\partial V_j(k)}$$

となる。ここで、 η は学習率であり、各勾配は

$$\frac{\partial e(k)}{\partial W_{ji}(k)} = -(r_m(k) - y(k)) \frac{\partial y(k)}{\partial u_c(k)} V_j(k) \times (1 + H_j(k))(1 - H_j(k)) I_i(k)$$

$$\frac{\partial e(k)}{\partial V_j(k)} = -(r_m(k) - y(k)) \frac{\partial y(k)}{\partial u_c(k)} H_j(k)$$

また学習に必要なヤコビアン情報は、(5)式のモデルで入出力関係が表現できるならば、ニューラル同定器のパラメータを用いて、

$$\frac{\partial y(k)}{\partial u(k)} \cong \frac{\partial y(k+1)}{\partial u(k)} = w_{n+1}(k) \quad (13)$$

として、学習を実行する。

7. 学習手順

時刻 k における制御ループの実行順序は次のようになる。

- (a) $r_m(k-1), r_m(k), r_m(k+1), y(k-1), y(k)$ の測定
- (b) 旧荷重 $w_i(k-1)$ を使って予想出力 $\hat{y}(k)$ を計算するためプラントニューラル同定器を使う。
- (c) 同定誤差 $\hat{e}(k)$ を計算し、新荷重 $w_i(k)$ を計算するためデルタルール(6)式を使う。
- (d) (8)式を使ってニューラルコントローラ荷重 $w'_i(k)$ を更新する。
- (e) 制御信号 $\bar{u}(k)$ を生成するためニューラルコントローラを使う。
- (f) 出力誤差 $e(k-1)$ を計算し、新荷重 $W_{ji}(k)$ と $V_j(k)$ を最急降下法(12)式を用いて計算する。
- (g) 新荷重 $W_{ji}(k)$ と $V_j(k)$ での NN 補償器を用いて、補助出力 $u_c(k)$ を生成する。
- (h) 制御信号 $u(k) = \bar{u}(k) + u_c(k)$ をプラントに入力し、 $y(k+1)$ を生成する。

Remark: この制御法は時刻 k において設定信号 $r_m(k+1)$ が既知であることに注意する。これは、制御されるプラントは 1 時刻遅れることから明らかな理由である。望まれたシステム応答は前もって固定されているという実際上の観点からこれは不都合なことではない。