

順モデルを利用した拡張フィードバック系による非線形システムの制御

大山英明*・館 暲**

Nonlinear System Control by Extended Feedback System with Forward Model

Eimei OYAMA* and Susumu TACHI**

In order to control a nonlinear dynamic system with unknown characteristics, the use of learning elements, i. e. multi-layer neural networks, has been studied.

A method that uses an acquired inverse model of the target system by learning is used by many researchers. However the conventional methods for learning an inverse model of the target system have many drawbacks. A method that uses an acquired forward model of the target system and calculates the input of the target system by using iterative methods, i. e. Newton's method, was proposed. However since iterative methods use only local information, they sometimes converge to local optimal solutions and can not calculate the precise solution for controlling the target system.

In order to solve a static system concerning inverse problems, we proposed a hybrid system that consists of an inverse model and the feedback system extended for the nonlinearity of the target system. The extended feedback system conducts a kind of iterative method from an enough number of initial values generated by random searches until a correct solution is obtained. Usually one of the correct solutions can be obtained by this procedure. However this configuration of the extended feedback system can not be applied for a dynamic system.

By using the acquired forward model of the target system instead of the target system, the proposed extended feedback system can control a nonlinear dynamic system.

The extended feedback system usually takes computation time. As the inverse model becomes accurate by learning, the required number of iterations for calculating the control value decreases and the computation time also decreases. The performance of the proposed system gets better by learning.

The performance of the proposed method is shown by numerical simulations.

Key Words: learning control, nonlinear dynamic system, extended feedback system, forward model, inverse model

1. はじめに

特性が未知の非線形システムの制御を行うために、多層神経回路などの学習要素の利用が研究されている^{1)~3)}。いくつかの手法によって、対象システムの逆モデルを学習し、制御を行う手法が提案されているが、逆モデルの獲得には問題がある^{4),5)}。Fun⁶⁾、北村⁷⁾、岩田⁸⁾らによって、順モデルを利用して、制御指令を計算する手

法が提案されているが、これらの手法は、共役勾配法やニュートン法などの局所的な情報に基づく反復解法に基づいており、局所最適解に収束してしまった場合など、適切な制御指令を計算できない場合がある。

著者らは、静的な逆問題の解法システムとして、ランダムな初期値探索と反復解法を組み合わせた拡張フィードバック系と、学習要素によって構成される逆モデルとを組み合わせた複合逆問題解法システムを提案した^{4),5)}。拡張フィードバック系の出力を逆モデルの教師信号とすることによって、従来法では正確な逆モデルの獲得が困難な対象システムについても、正確な逆モデルの獲得が可能である。

動的なシステムに対しては、静的システムのための拡張フィードバック系をそのまま用いることは難しい。し

* 機械技術研究所 つくば市並木 1-2

** 東京大学工学部 東京都文京区本郷 7-3-1

* Mechanical Engineering Laboratory, Tsukuba

** Faculty of Engineering, University of Tokyo,

Bunkyo-ku, Tokyo

(Received January 9, 1995)

(Revised May 22, 1995)

かしながら、対象システムの順モデルを学習要素によって獲得し、実際に対象システムを動作させる代わりに、学習した順モデルを動作させることによって、拡張フィードバック系を用いて、動的非線形システムを対象とする逆問題を解くことが可能となり、動的非線形システムの制御を行うことができる。

本論文では、順モデルを用いた拡張フィードバック系による動的非線形システムの制御法と逆モデルの利用法を提案する。数値シミュレーションによって提案法の有効性を示す。

2. ニュートン法による 非線形離散システムの制御

本論文では、つぎのような離散非線形システムを考える。

$$\begin{aligned} \mathbf{y}(k+1) &= \mathbf{f}(\boldsymbol{\theta}(k), \mathbf{u}(k)) \\ \boldsymbol{\theta}(k) &= (\mathbf{y}(k)^T, \mathbf{y}(k-1)^T, \mathbf{y}(k-2)^T, \dots, \mathbf{y}(k-i)^T, \\ &\quad \mathbf{u}(k-1)^T, \dots, \mathbf{u}(k-j)^T)^T \quad (k=0, 1, 2, \dots) \end{aligned} \quad (1)$$

\mathbf{y} は n 次元の出力ベクトルであり、 \mathbf{u} は有界な m 次元の入力ベクトルである。 $\boldsymbol{\theta}$ は、 \mathbf{y} と \mathbf{u} の過去の履歴によって構成されるベクトルであり、 i, j は既知とする。 $\mathbf{f}(\boldsymbol{\theta}, \mathbf{u})$ は、大部分の領域で一階微分可能な連続関数とする。本稿では、 $n \leq m$ の場合のみを考える。毎回の出力 $\mathbf{y}(k)$ を正確に観測できるものと仮定して、出力 $\mathbf{y}(k)$ を目標値 $\mathbf{y}_d(k)$ に追従させる制御入力 $\mathbf{u}(k)$ を計算する制御問題を考える。

本論文では、制御系内部に存在する $\mathbf{f}(\boldsymbol{\theta}, \mathbf{u})$ の近似関数を、対象システムの順モデルと呼ぶ。 $\mathbf{f}(\boldsymbol{\theta}, \mathbf{u})$ と特性が等しい関数を真の順モデルと呼ぶ。一方、次式のような関係を満たす関数 $\mathbf{g}(\boldsymbol{\theta}, \mathbf{y})$ の近似関数を逆モデルと呼ぶ。 $\mathbf{g}(\boldsymbol{\theta}, \mathbf{y})$ を真の逆モデルと呼ぶ。

$$\mathbf{y} = \mathbf{f}(\boldsymbol{\theta}, \mathbf{g}(\boldsymbol{\theta}, \mathbf{y})) \quad (2)$$

$\mathbf{f}(\boldsymbol{\theta}, \mathbf{u})$ が既知である場合、目標値 $\mathbf{y}_d(k)$ が与えられたとき、出力誤差のフィードバックを含む一時的目標値

$$\begin{aligned} \mathbf{y}'_d(k+1) &= \mathbf{f}(\boldsymbol{\theta}(k-1), \mathbf{u}(k-1)) + \Delta \mathbf{y}'_d(k) \\ \Delta \mathbf{y}'_d(k) &= \Delta \mathbf{y}_d(k) + K(\mathbf{y}_d(k) - \mathbf{y}(k)) \\ \Delta \mathbf{y}_d(k) &= \mathbf{y}_d(k+1) - \mathbf{y}_d(k) \quad (0 < K < 2) \end{aligned} \quad (3)$$

を定め、

$$\mathbf{y}'_d(k+1) = \mathbf{f}(\boldsymbol{\theta}(k), \mathbf{u}(k)) \quad (4)$$

を満たす制御量 $\mathbf{u}(k)$ の値を計算する逆問題を解くことにより、出力を目標値 $\mathbf{y}_d(k)$ に追従させることが可能である⁹⁾。

$\mathbf{f}(\boldsymbol{\theta}, \mathbf{u})$ が大部分の領域において一階連続微分可能な関数である場合、ニュートン法や共役勾配法などの反復解法を利用することによって、(4)式を解くことが可能

な場合が多い⁶⁾⁻⁸⁾。北村、岩田らは、多層神経回路によって構成される順モデルを利用し、局所線形近似に基づく非線形システムの制御法を提案している^{7),8)}。

ニュートン法によって、(4)式を解くことを考える。ニュートン法の i 回目の反復時の $\mathbf{u}(k)$ の値を $\mathbf{u}(k)(i)$ とし、 $\mathbf{y}'_d(k+1)$ と $\mathbf{f}(\boldsymbol{\theta}(k), \mathbf{u}(k)(i))$ の誤差を

$$\Delta \mathbf{y}'_d(k+1)(i) = \mathbf{y}'_d(k+1) - \mathbf{f}(\boldsymbol{\theta}(k), \mathbf{u}(k)(i)) \quad (5)$$

とする。単純なニュートン法では、適当な初期値から、

$$\begin{aligned} \mathbf{u}(k)(i+1) &= \mathbf{u}(k)(i) \\ &\quad + \mathbf{J}_u^*(\boldsymbol{\theta}(k), \mathbf{u}(k)(i)) \Delta \mathbf{y}'_d(k+1)(i) \end{aligned} \quad (6)$$

のような反復改良演算によって、 $\mathbf{u}(k)(i)$ の更新を行う。ただし、 $\mathbf{J}_u^*(\boldsymbol{\theta}, \mathbf{u})$ は(1)式のヤコビ行列

$$\mathbf{J}_u(\boldsymbol{\theta}, \mathbf{u}) = \frac{\partial \mathbf{f}(\boldsymbol{\theta}, \mathbf{u})}{\partial \mathbf{u}} \quad (7)$$

の逆行列または一般化逆行列である。解の近傍においてヤコビ行列がフルランクであり、初期値 $\mathbf{u}(k)(0)$ が適切であれば、 $\mathbf{u}(k)(i)$ は解へ2次収束し⁹⁾、これを用いて制御を行うことができる。 $\mathbf{u}(k)(0)$ としては、

$$\mathbf{u}(k)(0) = \mathbf{u}(k-1) \quad (8)$$

のように、前回の制御量がいられることが多い。

しかしながら、ニュートン法や共役勾配法は、対象システムの局所的な情報に基づいているため、(4)式を満たす制御量 $\mathbf{u}^*(k)$ が存在している場合でも、 $\mathbf{u}^*(k)$ と $\mathbf{u}(k)(i)$ との間に、ヤコビ行列 $\mathbf{J}_u(\boldsymbol{\theta}, \mathbf{u})$ がフルランクでなくなる領域が存在する場合には、 $\mathbf{u}^*(k)$ から遠く離れた値が⁸⁾、しばしば計算され、十分な制御ができない場合がある。たとえば、

$$\begin{aligned} y(k+1) &= f(y(k), u(k)) \\ &= y(k) + \frac{1}{3}u(k)^3 - u(k) \end{aligned} \quad (9)$$

$$|u(k)| < 3$$

という1入力1出力の非線形システムに、正弦波

$$y_d(k) = 10 \sin(\omega k) \quad (10)$$

$$\omega = 2\pi/30$$

を目標値として与えたとき、(3)式の K を1とし、(6)式のニュートン法によって、(4)式を解いた場合の応答は、Fig. 1 のようになる。ただし、(8)式に従って、反復の初期値を生成している。Fig. 1 において、破線が目標値であり、実線が対象システムの出力である。到底十分な制御ができていないとはいいたいが、このような結果は、ニュートン法が対象システムの非線形性に十分対応できないために生じている。共役勾配法を利用した Fun らの手法は、ニュートン法よりも安定であるが、正確な解を得ることはニュートン法と同様に難しい。線形近似に基づく岩田らの手法⁷⁾ はニュートン法よりもさらに非線形

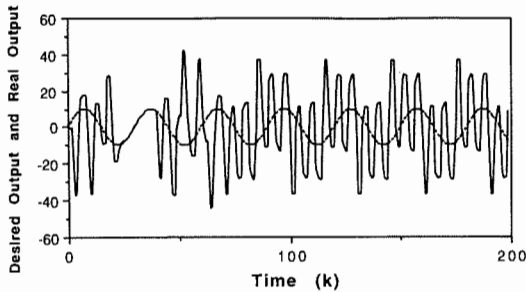


Fig. 1 Simulation result of controller using simple Newton's Method (Dashed line: Desired output, Plain line: Real output)

性への対応が難しいと考えられる。

3. 拡張フィードバック系による動的システムの制御

著者らは、静的な逆問題の解法システムとして、学習要素によって構成される逆モデルと拡張フィードバック系を組み合わせた複合逆問題解法システムを提案している^{4),5)}。非線形方程式の反復解法の研究が始まって以来、対象システムの非線形性に対抗する最も有力な手法は、多数の初期値から反復解法を出発する多点探索と直線探索である。拡張フィードバック系は、このような非線形性への対策を利用して、非線形方程式を解く。また、その解を利用して、対象システムの逆モデルのオンライン学習が可能である⁵⁾。

静的なシステムの場合、複数の入力値をシステムに入力して、出力を観測することにより、入力値の適切さの評価を行い、適切な入力を求めることができる。一方、動的システムの場合、一つの入力値を対象システムに入力すると、対象システムの入出力関係が変化するため、複数の入力値から、適切なものを選択することは困難である。しかしながら、対象システムの順モデルを学習要素によって獲得し、実際に対象システムを動作させる代わりに、順モデルを動作させることによって、複数の入力値の評価、比較することが可能であり、拡張フィードバック系を用いて、動的システムの逆問題の解を見つけることが可能となる。また、学習した逆モデルの出力を反復改良の初期値として利用することにより、少ない反復回数による制御量の計算が可能となる。

3.1 拡張フィードバック系による動的システムの制御

本論文では、以下に述べるような理想的な学習要素を想定する。学習要素への入力 t であるとき、学習要素の出力を $\Phi(t)$ とし、そのときの教師信号を $\Phi'(t)$ とする。学習要素への入力 t であるとき、

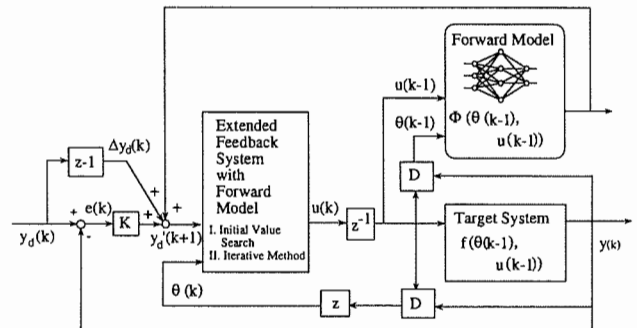


Fig. 2 Configuration of controller using Extended Feedback System

$$\Phi'(t) = s_i \quad (11)$$

のように、教師信号 s_i ($i=1, 2, 3, \dots$)が与えられたとき、理想的な学習要素では、その出力 $\Phi(t)$ が、

$$\Phi(t) \doteq E(s_i) \quad (12)$$

となるように学習が行われるものとする。 $E(s_i)$ はベクトル s_i の平均値である。また、補間能力により、有限個の点の学習で、入力 t の全域に渡る学習が可能と仮定する。現在、用いられている学習要素は、このような性質を近似的に満たしている。

動的なシステムの逆問題を解くための、拡張フィードバック系を利用した制御系は、対象システム $f(\theta, u)$ の順モデルを学習要素 $\Phi(\theta, u)$ によって獲得する。学習の結果、有界な学習要素への入力 (θ, u) について、

$$f(\theta, u) \doteq \Phi(\theta, u) \quad (13)$$

が成立するものと仮定する。順モデルの学習法については、3.2節で示す。

提案する制御系の構成をFig. 2に示す。図中の D は $y(k)$ と $u(k-1)$ から $\theta(k)$ を合成する制御要素である。また、図中の神経回路の模式図は、学習要素を示すものとする。Fig. 2に示すように、提案法では、

$$y_d'(k+1) = \Phi(\theta(k-1), u(k-1)) + \Delta y_d'(k)$$

$$\Delta y_d'(k) = \Delta y_d(k) + K(y_d(k) - y(k))$$

$$\Delta y_d(k) = y_d(k+1) - y_d(k) \quad (14)$$

によって計算される一時的出力目標値 $y_d'(k+1)$ について、

$$y_d'(k+1) = \Phi(\theta(k), u(k)) \quad (15)$$

という方程式を満たす $u(k)$ を、順モデルを利用した拡張フィードバック系により計算する。(15)式の解が存在し、それを計算できる場合、3.6節で示すように、出力の制御を行うことが可能である。そうでない場合、十分な制御が可能となる保証はない。

動的システムの制御のための拡張フィードバック系の内部構造をFig. 3に示す。拡張フィードバック系は、(6)式の代わりに、

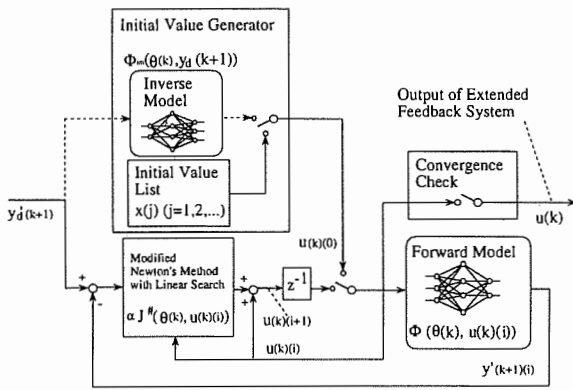


Fig. 3 Configuration of Extended Feedback System

$$u(k)(i+1) = u(k)(i) + \alpha(i) J_{ua}^*(\theta(k), u(k)(i)) \Delta y_d'(k+1)(i) \quad (16)$$

$$\Delta y_d'(k+1)(i) = y_d'(k+1) - \Phi(\theta(k), u(k)(i)) \quad (17)$$

のようなニュートン法に基づく、反復改良を行う。 $\alpha(i)$ は直線探索によって決まる係数である。 $J_{ua}^*(\theta, u)$ は、順モデル $\Phi(\theta, u)$ のヤコビ行列

$$J_{ua}(\theta, u) = \frac{\partial \Phi(\theta, u)}{\partial u} \quad (18)$$

の近似行列 $J_{ua}(\theta, u)$ の逆行列 ($n=m$ の場合) もしくは疑似逆行列 ($n \leq m$ の場合) である。近似ヤコビ行列 $J_{ua}(\theta, u)$ の計算法については、3.3 節で示す。直線探索の係数 $\alpha(i)$ の計算法については、付録 1 で述べる。

Fig. 3 の Initial Value Generator (初期値生成器) は、拡張フィードバック系の反復計算の初期値 $u(k)(0)$ を生成するシステムである。図の破線の経路で示すように、学習要素によって構成される逆モデル $\Phi_{im}(\theta, y)$ が存在すれば、これを利用することによって、逆モデルの学習後は、高速な $u(k)$ の計算が可能となる。逆モデルの学習については 3.5 節で述べる。

反復改良演算を多数の初期値から行うために、初期値発生器においては、十分な個数の初期値のリスト $x(j)$ ($j=0, 1, \dots, j_{\max}$) を用意する。初期値リスト $x(j)$ のうち、 $x(j)$ ($j \geq 1$) は、制御の前に、一様乱数によって生成しておく。 $x(0)$ については、対象システムの逆モデルが存在しない場合、前回の制御出力を利用して、

$$x(0) = u(k-1) \quad (19)$$

とし、逆モデル $\Phi_{im}(\theta, y_d)$ が存在する場合、

$$x(0) = \Phi_{im}(\theta(k), y_d'(k+1)) \quad (20)$$

とするものとする。

(16) 式の反復改良によって、解を得るためには、(15) 式を満たす解が存在し、その近傍でヤコビ行列 $J_{ua}(\theta, u)$ がフルランクであることが必要である。また、十分な数

の初期値から反復改良演算を繰り返し行うことのできる計算能力が必要である。

拡張フィードバック系は、以下のような動作手順によって、(15) 式を満たす $u(k)$ を求める。

(I) $j=0$ とする。

(II) $u(k)$ を計算するための反復改良の初期値を $u(k)(0)$ とし、

$$u(k)(0) = x(j) \quad (21)$$

とする。

(III) 解が見つかる場合など、後述する停止条件が成立するまで、(16) 式のような反復演算を行う。

(IV) 解が得られた場合には、

$$u(k) = u(k)(i) \quad (22)$$

として、制御指令 $u(k)$ を出力する。

(V) 解が得られておらず、計算時間も制限を越えていない場合には、 j の値を一つ増やし、(II) へ戻る。

(VI) 解が得られておらず、計算時間も制限を越えてしまった場合には、(I) の時点から、出力誤差のノルム $|\Delta y_d'(k+1)(i)|$ を最も小さくした $u(k)(i)$ の値を u_{\min} とすると、

$$u(k) = u_{\min} \quad (23)$$

として、制御指令 $u(k)$ を出力する。

手順 (III) の反復停止条件には、

(a) 目標精度 ε を達成する解が見つかる場合

この場合、次式が成立する。

$$|\Delta y_d'(k+1)(i)| < \varepsilon \quad (24)$$

(b) 局所最適解に収束した可能性が高い場合

次式が成立すれば、局所最小解に収束した可能性は高い。

$$|\Delta y(k+1)(i)| < \gamma |\Delta y_d'(k+1)(i)| \quad (25)$$

$$\Delta y(k+1)(i) = \Phi(\theta(k), u(k)(i+1)) - \Phi(\theta(k), u(k)(i)) \quad (26)$$

γ は十分小さな正の数である。

(c) 反復計算において、 $u(k)(i+1)$ が連続して、 u の定義域から外れる場合のようなものがある。

3.2 順モデルの学習

順モデルの学習においては、学習要素に、 $(\theta(k), u(k))$ を入力し、その出力 $\Phi(\theta(k), u(k))$ が $y(k+1)$ に一致するように学習を行う。学習要素の教師信号を $\Phi'(\theta(k), u(k))$ とすると、

$$\Phi'(\theta(k), u(k)) = y(k+1) \quad (27)$$

とする。

本論文では、順モデルの主たる学習法として、オフライン学習を想定する。対象システムに様々な入力を与え、そのときの入出力の値をサンプリングし、オフライン学

習の教師データを生成する。順モデルが未学習の状態では、対象システムの自由な制御は不可能である。よって、対象システムの状態を、正確な順モデルが学習されれば制御可能となる初期状態に、リセットできることが、順モデルの学習の前提条件となる。

オフライン学習の教師データを得る場合、実際の制御に関係する領域において、データ数が十分でない場合があり、その領域において、順モデルの精度が低下する。制御中に、順モデルのオンライン学習を行うことにより、実際の制御に関係する領域において、順モデルの学習が可能となり、順モデルの精度が向上するにつれ、制御性能も向上する。なお、オンライン学習のみで制御可能になるシステムも存在する。

3.3 反復計算の手法

(16)式の反復計算を行うためには、ヤコビ行列の近似値を計算する必要がある。川人や著者らは、ヤコビ行列やその疑似逆行列を学習要素によって直接学習する手法を提案している^{10),11)}が、本稿では、順モデルを用いたヤコビ行列の計算法について述べる。

一般の学習要素によって構成される順モデルによって、ヤコビ行列を計算する最も簡単な手法は、数値微分である。ヤコビ行列の推定値 $J_{ua}(\theta, u)$ の i 行 j 列成分 J_{uaij} は、適切に選んだ数 d を利用して、

$$J_{uaij} = \frac{\Phi_i(\theta, u + de^{(j)}) - \Phi_i(\theta, u)}{d} \quad (28)$$

のように計算できる。ここで、 $e^{(j)}$ は、その成分 $e_i^{(j)}$ がクロネッカーのデルタ δ_{ij} を利用して、

$$e_i^{(j)} = \delta_{ij} \quad (29)$$

のように定義される m 次元基底ベクトルである。

一般に非線形学習要素の評価には、計算時間がかかる場合が多く、順モデルの評価回数は少ないほうが望ましい。ヤコビ行列を逐次近似する Broyden の準ニュートン法¹²⁾の利用が有効と考えられる。今回は、より基本的な手法である(28)式に基づくヤコビ行列の推定法を用いる。

なお、バックプロパゲーション学習を行う多層神経回路の場合、バックプロパゲーション信号を用いることによって、ヤコビ行列の計算が可能である。北村らが用いている手法もある⁹⁾。

3.4 初期値系列の調整

拡張フィードバック系においては、複数の初期値より反復演算の出発を行うが、初期値の順番を並べ替えることによって、より高速な制御量の計算が可能となる。

初期値 $x(j)$ のそれぞれについて、解への到達回数を記憶する配列 $s(j)$ を用意し、 j 番目の初期値 $x(j)$ が用いられて、解に到達した場合、

$$s'(j) = s(j) + 1 \quad (30)$$

のように、 $s'(j)$ を計算し、 $s'(j)$ を新たな $s(j)$ とする。一定期間の後、 $s(j)$ が大きい順に初期値 $x(j)$ の値を並べ替えることによって、より少ない初期値の変更によって解を得ることができるようになる。 $s(j)$ が小さい $x(j)$ の値は、新たに一様乱数によって変更することも効果的である。

なお、計算能力に余裕がある場合には、解が求められた場合にも計算を停止せず、できる限り多い初期値について、解への到達を調べることによって、より解へ到達しやすい初期値を選ぶことができる。

3.5 複合逆問題解法システムの構成

拡張フィードバック系の初期値系列 $x(j)$ の $x(0)$ とし、逆モデルの出力を利用し、複合逆問題解法システムを構成することによって、逆モデルの学習後は、(15)式の解の計算を高速化できる。

逆モデルの入力を $(\theta(k), y'_a(k+1))$ とし、その出力を $\Phi_{im}(\theta(k), y'_a(k+1))$ とし、拡張フィードバック系の初期値系列 $x(j)$ において、

$$x(0) = \Phi_{im}(\theta(k), y'_a(k+1)) \quad (31)$$

とする。 $x(0)$ を初期値とした反復改良によって、解 $u^*(k)$ が得られた場合には、逆モデル $\Phi_{im}(\theta(k), y'_a(k+1))$ の教師信号を $\Phi_{im}(\theta(k), y'_a(k+1))$ とすると、

$$\Phi_{im}(\theta(k), y'_a(k+1)) = u^*(k) \quad (32)$$

として、逆モデルの学習を行う。

(12)式より、想定する学習要素は教師信号の平均値を出力する。複数個の正しい教師信号の平均値が、正しい教師信号である保証はない。逆モデルへの一つの入力 (θ, y) に対して、正しい教師信号 $\Phi_{im}(\theta, y)$ を唯一つ定めることによって、正確な逆モデルを学習することができる。通常、この条件は成立しているが、前節で述べた初期値系列の調整を行う場合には、調整前と調整後で、 (θ, y) に対応する教師信号が一致する保証はない。よって、調整前は逆モデルの学習を行わず、調整が終わった後に逆モデルの学習を行うものとする。実際に制御を行っている状態で、逆モデルのオンライン学習を行うことができるのは、拡張フィードバック系の利点の一つである。

未学習の逆モデルの出力は、通常、(19)式の初期値よりも、解から遠いため、反復計算に時間がかかる。実際の対象システムを動作させる代わりに、順モデルを動作させて、制御量を計算し、(32)式に基づく学習を行うことによって、逆モデルのオフライン学習が可能である。順モデルを利用したオフライン学習においては、拡張フィードバック系を利用する以外にも、Matyas型ランダム探索法や遺伝的アルゴリズムなど近似的な大域最適化

手法を利用して、逆モデルの学習を行うことができる^{13),14)}。しばしば局所最小解に陥り、正確に教師信号を実現できないという弱点をもつバックプロパゲーション学習¹⁵⁾を行う学習要素には、特に有効と考えられる。通常、順モデルのモデル化誤差の影響が残るので、オンライン学習によって逆モデルの高精度化を図ることが望ましい。近似的大域最適化手法をオンライン学習に用いることは難しく、拡張フィードバック系の利用によるオンライン学習が適切である。

3.6 モデル化誤差と計算誤差の影響

本節では、制御の全ての時刻で(15)式の解の近似値を得ることができるものと仮定し、提案法による出力誤差の大きさを見積もる。対象システムの真の逆モデルが不安定な場合など、(15)式の解が得られない時刻が存在する場合の研究は今後の課題である。

制御系内の順モデル $\Phi(\theta, u)$ と真の順モデル $f(\theta, u)$ との差を $d(\theta, u)$ と置き、つぎのような関係があるものとする。

$$\Phi(\theta, u) = f(\theta, u) + d(\theta, u) \quad (33)$$

拡張フィードバック系により、(14)式を満たす $u(k)$ によって制御を行う場合、計算誤差のため、

$$y_d(k+1) = \Phi(\theta(k), u(k)) + c(k) \quad (34)$$

となる解が計算されるものとする。丸め誤差などの数値誤差は十分小さいと考えられるため、 $c(k)$ として、問題となるのは反復打ち切りのために生じる誤差であり、3.1節で述べた反復の停止条件(a)より、

$$|c(k)| \leq \epsilon \quad (35)$$

が成立する。十分な計算能力がある場合、 ϵ を非常に小さく取ることができる。出力 $y(k)$ の更新式は

$$\begin{aligned} y(k+1) &= y_d(k+1) - d(\theta(k), u(k)) - c(k) \\ &= y(k) + \Delta y_d(k) + K e(k) \\ &\quad + d(\theta(k-1), u(k-1)) - d(\theta(k), u(k)) \\ &\quad - c(k) \end{aligned} \quad (36)$$

となり、ここで、

$$\lambda(k) = d(\theta(k), u(k)) - d(\theta(k-1), u(k-1)) \quad (37)$$

と置くと、出力誤差 $e(k)$ の更新式は

$$e(k+1) = (1-K)e(k) + \lambda(k) + c(k) \quad (38)$$

となる。順モデルの誤差や計算誤差は、そのまま出力誤差に反映される。

任意の $\theta_1, u_1, \theta_2, u_2$ について、

$$\begin{aligned} h &= \sup (|d(\theta_1, u_1) - d(\theta_2, u_2)|) + \epsilon \\ &\geq |\lambda(k)| + |c(k)| \end{aligned} \quad (39)$$

とする。また、

$$\beta = |1-K| \quad (40)$$

とすると、通常

$$0 \leq \beta < 1 \quad (41)$$

であり、

$$\begin{aligned} |e(k+1)| &\leq |1+K||e(k)| + |\lambda(k)| + |c(k)| \\ &\leq \beta^k |e(0)| + \frac{h}{1-\beta} \end{aligned} \quad (42)$$

を得る。モデル化誤差の関数 $\lambda(k)$ と計算誤差 $c(k)$ をシステムへの入力とみなし、 $e(k)$ を出力とみなすと、(42)式より、入出力安定が成立する¹⁶⁾。

ここで、

$$\Delta x(k) = (\Delta \theta(k)^T, \Delta u(k)^T)^T \quad (43)$$

と置くと、 $\lambda(k)$ は、

$$\begin{aligned} \lambda(k) &= d(\theta(k), u(k)) - d(\theta(k-1), u(k-1)) \\ &= \frac{\partial d(\theta(p), u(p))}{\partial \theta(p)} \Big|_{p=k-1} \Delta \theta(k-1) \\ &\quad + \frac{\partial d(\theta(p), u(p))}{\partial u(p)} \Big|_{p=k-1} \Delta u(k-1) \\ &\quad + O(\Delta x(k-1)^2) \end{aligned} \quad (44)$$

と展開できる。 $\lambda(k)$ は $\Delta x(k-1)$ の1次以上の高次項によって構成されている。よって、計算機内の順モデルと真の順モデルの定数項の誤差は無視できる。目標信号が一定値になる場合、通常、 $|\Delta x(k-1)|$ は十分小さくなり、 $\lambda(k)$ も十分小さいと期待できる。

4. シミュレーション

提案法の有効性を示すために、非線形の対象システムを制御するシミュレーションを行う。本論文で提案した手法は、3章で述べた性質を近似的に満たす一般の学習要素について有効であるが、今回は、学習要素として、バックプロパゲーション学習を行う4層の神経回路を用いる¹⁵⁾。バックプロパゲーション学習には、局所最小解の問題があり、近似的大域最適化手法の利用が提案されているが¹³⁾、本論文では、この点を無視し、学習要素の一例として用いる。中間層を構成する第2層と第3層に、シグモイド関数 ($y = \tanh(x)$) で記述される入出力特性をもった非線形のニューロンを配置し、入力層と出力層は線形ニューロンとする。結合荷重の初期値は $(-0.5, 0.5)$ に分布する一様乱数で与える。

4.1 1入力1出力非線形離散システム

提案法によって、2章の(9)式のシステムを制御するシミュレーションを行う。

用いる順モデルの入力は $(y(k), u(k))$ であり、出力は $y(k+1)$ の近似値である。順モデルを構成する神経回路は入力層より順に、2, 15, 15, 1個のニューロンによって構成される。

拡張フィードバック系の出力誤差の目標精度は

$$\epsilon = 0.05 \quad (45)$$

とする。また、局所最適解への収束の判別パラメータは

$$\gamma=0.001 \quad (46)$$

とする。初期値変更回数の最大値 j_{\max} は 20 回とし、一つの初期値について、行う反復回数の最大値 i_{\max} は 20 とする。反復回数の総和が 200 回を超えると、計算時間が尽きたものとみなす。

4.1.1 オフライン学習とオンライン学習

まず、順モデルのオフライン学習を行う。対象システムの初期状態を

$$y(0)=y(1)=0 \quad (47)$$

とし、

$$u(k)=\sum_{j=0}^M A\left(\frac{1}{p}\right)^j \sin(\omega p^j k) \quad (48)$$

のような疑似ランダム信号を、対象システムに入力する。ここで、 $M=18$, $p=1.25$, $\omega=0.0044$, $A=3.0$ とする。 $\omega p^M=0.244$ である。このときのシステムへの入力 $(y(k), u(k))$ と対応する出力 $y(k+1)$ を記憶し、教師信号とする。ただし、

$$|y(k+1)| > 15 \quad (49)$$

となった場合には、出力の状態をリセットし、

$$y(k)=y(k-1)=0 \quad (50)$$

とするものとする。入出力の組 $(y(k), u(k)) (k=1, 2, \dots, 1000)$ によって、1000 個の教師データを生成し、 $(y(k), u(k)) (k=1001, \dots, 2000)$ をチェック用のデータとする。1000 個の教師データからランダムに選んだ教師信号によって、バックプロパゲーション学習を行い、続いて共役勾配法によって学習を行う。バックプロパゲーション学習において、 i 回目の学習時における神経回路の出力を $\Phi(q_i)$ 、教師信号を p_i とし、誤差評価関数を以下のように定義する。

$$S(i)=|p_i - \Phi(q_i)|^2 \quad (51)$$

神経回路の一つの結線の結合加重を w 、 i 回目の学習時における w の修正量を $\Delta w(i)$ とおくと、

$$\Delta w(i) = -\frac{\eta \partial S(i)}{\partial w} + \xi \Delta w(i-1) \quad (52)$$

のように修正量を決定する。慣性率 ξ は 0.5 で固定とする。順モデルのための学習係数 η は 0.0001 とする。バックプロパゲーション学習の回数は 100 万回である。

順モデルの出力の誤差を

$$e_{fm}(k) = y(k+1) - \Phi(y(k), u(k)) \quad (53)$$

とする。共役勾配法を用いた学習の結果、チェック用データに関する出力の RMS 誤差

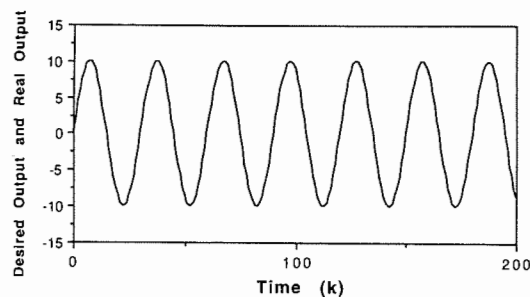
$$r = \sqrt{E(|e_{fm}(k)|^2)} \quad (54)$$

は 0.173 となった。

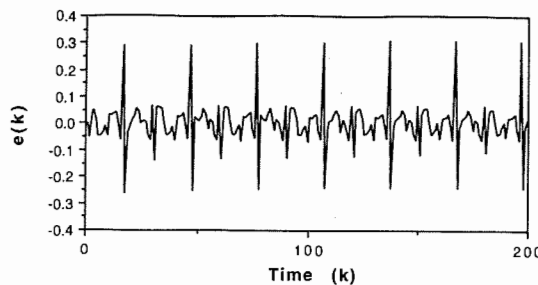
順モデルが学習された後、2章の制御と同様

$$y_d(k) = 10 \sin(\omega \Delta t k) \quad (55)$$

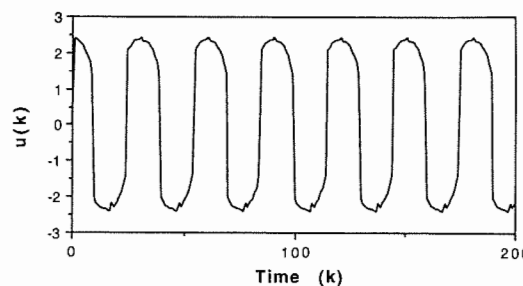
$$\omega \Delta t = 2\pi/30$$



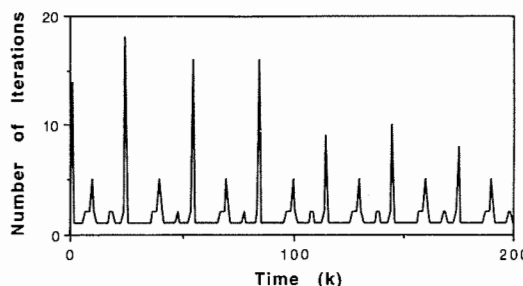
(a) Desired output and real output



(b) Output error



(c) Control value



(d) Number of iterations

Fig. 4 Simulation results ($k=0\sim 200$, $\epsilon=0.05$)

によって生成される目標値に対して、追従制御を行う。拡張フィードバック系の初期値系列は、3.4節で述べた一括型的手法で、制御計算を 100 回行うごとに調節するものとする。

提案法による追従の様子を Fig. 4 に示す。単純なニュートン法では、追従は不可能であるが、提案法によって追従が可能である。Fig. 4 の (a) は、追従の様子であり、(b) は出力誤差 $e(k)$ である。(c) は、制御量 $u(k)$ であり、(d) は、ニュートン法の反復回数である。 $k=100$

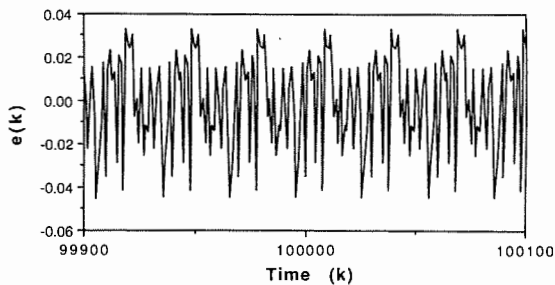
の時点で、それまでの初期値の解への到達回数記録を用いて、3.4節で述べた一括型の初期値系列の並び替えを行っている。また、3.4節の $s(j)$ の値が、 $s(j)$ の総和の1割に満たない初期値は捨てて、新たな初期値を一様乱数を用いて生成している。(d)に示すように、初期値系列の調整を行わない時点においては、反復演算の回数は18回に達している。調整後は、反復回数は、最大9回となっており、調整によって、計算量が減少することがわかる。

続いて、オンライン学習による順モデルの高精度化の様子を示す。制御量計算のたびに、順モデルの学習を行い、99900回の順モデルの学習を行った時点での制御の結果を Fig. 5 示す。Fig. 4の(a)と同様、目標出力と実際の出力とは、肉眼では区別がつかない。Fig. 5の(a)より、順モデルの精度の向上によって、追従誤差が0.04以下に、減少していることがわかる。

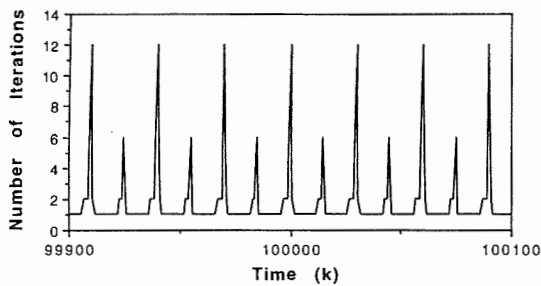
4.1.2 逆モデルの利用

拡張フィードバック系による制御系の特徴の一つは、逆モデルのオンライン学習が可能なことである。逆モデルを利用して制御を行いつつ、逆モデルのオンライン学習を行うシミュレーションを行った。

逆モデルを構成する神経回路は入力層より順に、2, 15, 15, 1個のニューロンによって構成される。逆モデルのバックプロパゲーション学習のための学習係数 η は0.0001とする。シミュレーションの結果、精度については4.1.1項と同様の結果を得た。Fig. 6に、反復回数を示す。 $k=100$ の時点でのみ初期値系列の調整を行っている。



(a) Output error



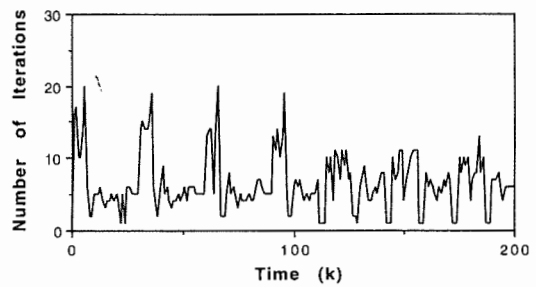
(b) Number of iterations

Fig. 5 Simulation Results ($k=99900\sim 100100$, $\varepsilon=0.05$)

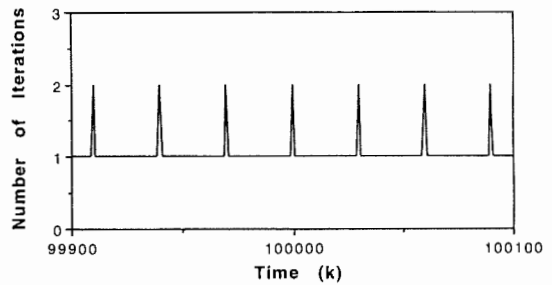
Fig. 6(a)は $k=0\sim 200$ の反復回数を示し、Fig. 6(b)は $k=99900\sim 100100$ の反復回数を示す。逆モデルが未学習の状態では、(19)式の $x(0)$ を利用したほうが、必要な反復回数は少ない。しかしながら、逆モデルの学習が進むと、最大2回の反復で、制御量の計算が可能となり、計算時間が大幅に短縮される。

Fig. 6(a)に示すように、逆モデルが未学習の状態では、逆モデルの利用は得策ではない。3.5節で述べたように、あらかじめ逆モデルの学習をオフラインで行っておくことによって、初期値としては十分正確な逆モデルを得ることができる。対象システムの代わりに順モデルを用いて、Fig. 6の場合と同様の逆モデルの学習を99900回行い、それを逆モデルとして利用した場合のシミュレーション結果を Fig. 7 に示す。

正確な逆モデルが学習された場合には、単純なニュートン法の反復を1回行っただけで、十分な制御が可能と



(a) $k=0\sim 200$



(b) $k=99900\sim 10100$

Fig. 6 Number of iterations

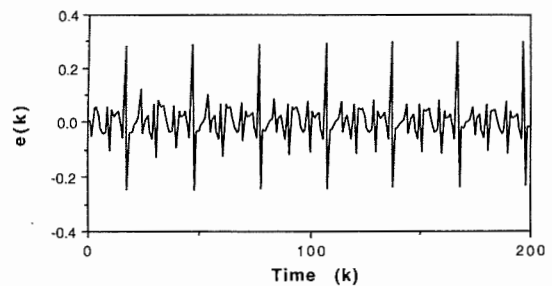


Fig. 7 Output Error of Extended Feedback System with inverse model ($k=0\sim 200$, $\varepsilon=0.05$)

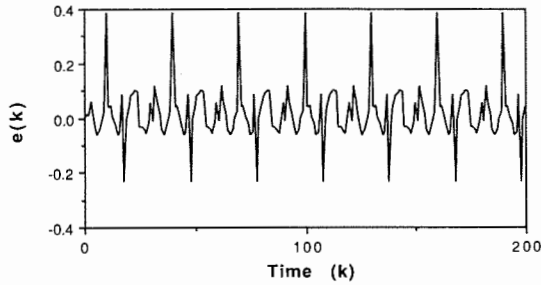


Fig. 8 Output error of simple Newton's method with inverse model ($k=0\sim 200$, $i_{\max}=1$)

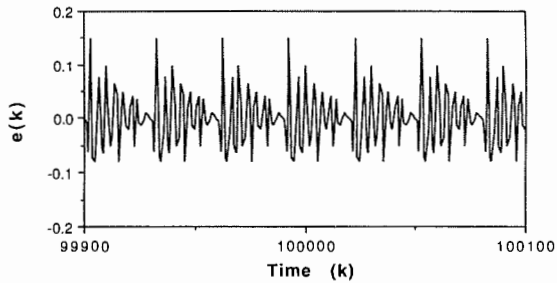


Fig. 9 Output error of Extended Feedback System with inverse model ($k=99900\sim 100100$, $\varepsilon=0.05$)

なる。Fig. 8に、その出力誤差を示す。拡張フィードバック系と比べて、精度はわずかながら低下しているが、十分な制御が可能である。

4.1.3 順モデルの疑似オンライン学習による制御

対象システムを何度でも、初期状態にリセットできる場合には、順モデルの擬似的なオンライン学習が可能となる。

$$|y(k)| > 20 \quad (56)$$

となった場合には、状態をリセットし、

$$y(k) = 0 \quad (57)$$

とするという条件の基に、対象システムの制御を行いつつ、順モデルのオンライン学習を行う。初期の段階では、全く制御できていないが、順モデルの学習が進行するにつれて、制御が可能となる。逆モデルを利用した場合の $k=99900$ 前後のシミュレーション結果を Fig. 9 に示す。反復回数は最大2回である。

4.2 3入力2出力非線形離散システム

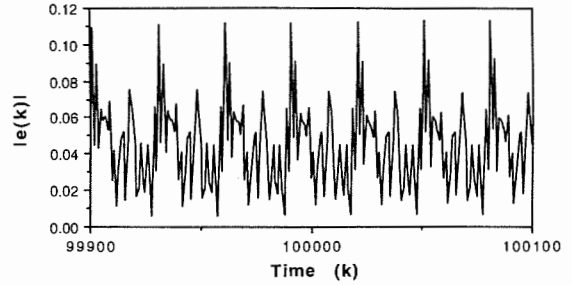
より自由度の大きい系の例として、つぎのような対象システムの制御問題を考える。

$$y_0(k+1) = y_0(k) - 0.01(y_0^2(k-1) - 1)y_0(k) + u_0(k)(1 - u_1(k)u_2(k)) + 0.01u_0^2(k-1)$$

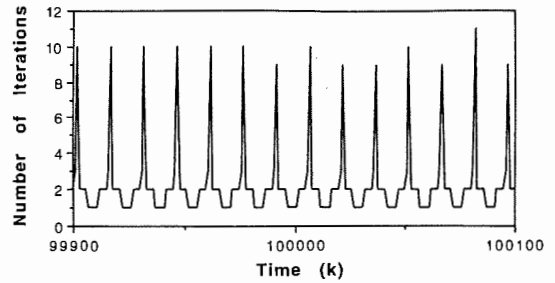
$$y_1(k+1) = y_1(k) - y_0(k) + \frac{1}{3}u_1^3(k) - u_1(k)u_2^2(k)$$

$$+ 0.01u_0(k-1)u_0(k)$$

$$|u_i(k)| < 3 \quad (58)$$



(a) Output error



(b) Number of iterations

Fig. 10 Simulation results ($k=99900\sim 100100$, $\varepsilon=0.05$)

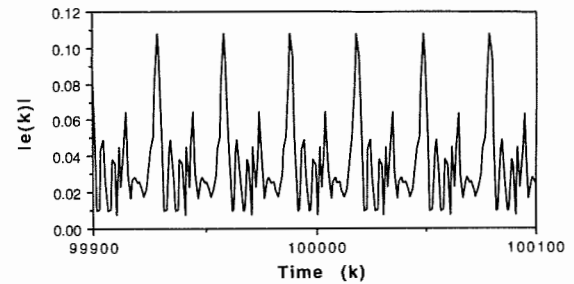


Fig. 11 Output error of Extended Feedback System with inverse model ($k=99900\sim 100100$, $\varepsilon=0.05$)

このような対象システムを制御するシミュレーションを行う。4.1節と同様に、単純なニュートン法に基づく手法では、十分な制御は不可能である。順モデルの入力は $(y_0(k), y_1(k), y_0(k-1), u_0(k-1), u_0(k), u_1(k), u_2(k))$ である。順モデルを構成する神経回路は、入力層から順に、7個、50個、50個、2個のニューロンによって構成されている。5000個の教師データと5000個のチェック用データを、対象システムに一樣乱数によって生成した入力を与えて生成する。共役勾配法による学習の結果教師データに対する順モデルのRMS誤差は0.375となった。

目標値として、

$$y_{d0}(k) = 10 \sin(\omega k)$$

$$y_{d1}(k) = 10 \cos(\omega k) \quad (59)$$

を与え、制御を行いつつ、順モデルのオンライン学習を

行う。順モデルの学習係数 η は 0.001 とする。99900 回の順モデルの学習を行った時点における追従誤差 $e(k)$ のノルム $|e(k)|$ と $\epsilon=0.05$ を達成するための、ニュートン法の反復回数を Fig. 10 に示す。

続いて、逆モデルを利用した場合の結果について示す。逆モデルの入力は $(y_{a0}(k), y_{a1}(k), y_0(k), y_1(k), y_0(k-1), u_0(k-1))$ である。順モデルを構成する神経回路は、入力層から順に、6 個、40 個、40 個、3 個のニューロンによって構成されている。逆モデルの学習係数 η は 0.01 とした。Fig. 11 に、逆モデルを利用した拡張フィードバック系による制御の出力誤差を示す。必要な反復回数は 1 であり、高速な制御量の計算が可能であることが確認された。

5. おわりに

特性が未知の離散非線形システムの制御法として、対象システムの順モデルを利用する拡張フィードバック系による制御法と逆モデルの利用法を提案し、シミュレーションにより、その有効性を示した。

逆モデルをもっていない場合や、逆モデルが未学習の場合、拡張フィードバック系による制御信号の計算においては、ランダムな初期値探索と反復解法を繰り返すため、計算量が非常に大きくなるという問題点があるが、計算機の計算能力の向上により、この弱点は回避可能になると考えられる。

真の逆モデルが不安定な場合を含めて、(15)式の解が存在しない場合への対応は今後の課題である。

4章のシミュレーションにおいては、工業技術院情報計算センターの計算機群を利用した。計算機環境の整備に努めておられる計算センターの諸氏に深謝する。

参考文献

- 1) M. I. Jordan: Supervised Learning and Systems with Excess Degrees of Freedom, COINS Technical Report, 88-27, 1/41(1988)
- 2) M. Kawato, K. Furukawa and R. Suzuki: A Hierarchical Neural-network Model for Control and Learning of Voluntary Movement, Biol. Cybern., 57, 169/185 (1987)
- 3) 山田, 藪田: ニューラルネットワークを用いた適応型フィードフォワードフィードバックコントローラ, 計測自動制御学会論文集, 30-10, 1234/1241(1994)
- 4) 大山, 館: 多対一対応システムの逆モデル学習の一手法, 第3回インテリジェントシステム・シンポジウム, 305/310(1993)
- 5) 大山, 館: 拡張フィードバック系による逆モデル学習, 電子情報通信学会論文誌 D-II, J78-D-II-2, 349/357(1995)
- 6) W. Fun and M. I. Jordan: The Moving Basin: Effective Action-search in Adaptive Control, Technical Report, # 9204, Dept. of Brain & Cognitive Sciences, MIT (1992)
- 7) 北村新三: 計測制御における逆問題とニューラルネット, 計測自動制御学会第2回自律分散シンポジウム, 51/56 (1991)
- 8) 岩田, 北村: ニューラルネットによる順方向モデルを用いた制御と入力推定系の構成, 計測自動制御学会論文集, 30-3, 303/309(1994)
- 9) D. G. Luenberger: Optimization by vector space methods, John-Wiley & Sons, New York (1969) 日本語訳増淵, 嘉納訳: 関数解析による最適理論, コロナ社, p. 304(1973)
- 10) 大山, 館: 非線形ゲインを学習する逆問題解法システム, 電子情報通信学会論文誌 D-II, J 77-D-II-7, 1345/1354(1994)
- 11) M. Kawato: A Optimization and Learning in Neural Networks for Formation and Control of Coordinated Movement, ATR Technical Report, TR-A-0086(1990)
- 12) C. G. Broyden: A Class of Methods for Solving Non-linear Simultaneous Equations, Math. Comp., 19(1965)
- 13) N. Baba: A New Approach for Finding the Global Minimum of Error Function of Neural Networks, Neural Networks, 2, 367/373(1989)
- 14) X. Yao: Evolutionary Artificial Neural Networks, Int. J. Neural Systems, 4, 203/222(1993)
- 15) D. E. Rumelhart, G. E. Hinton and R. J. Williams: Learning Internal Representation by Error Propagation, Parallel Distributed Processing, D. E. Rumelhart and J. L. MacClelland and the PDP Resarch Group ed., 318/326, MIT Press (1986)
- 16) C. A. Desoer and M. Vidyasagar: Feedback Systems: Input Output Properties, Academic Press (1975)

【付録1 単純な直線探索法】

反復改良の係数 $a(i)$ の決定法として、本稿では、非常に簡単な手法を用いる。 p を非負の整数として、

$$a(i) = \left(\frac{1}{2}\right)^p \quad (A 2.1)$$

とおき、 p を非線形性の度合いに応じて調節し、反復計算において、 $\Delta y'_d(k+1)(i)$ のノルムが減少することを保証する。すなわち、

$$|\Delta y'_d(k+1)(i+1)| \leq |\Delta y'_d(k+1)(i)| \quad (A 2.2)$$

が成立するまで、 p を増大させ、 $a(k)$ を $1/2$ にしていく。反復の次のステップでは、 $a(k)$ が $1/2$ 以下であれば、 p を一つだけ減らし、 $a(i)$ が 1 であれば、 $a(i+1)$ も 1 とする。無限ループに陥らないように、 p には上限を設ける。なお、 p の初期値は 0 とする。

[著者紹介]

大山 英明 (正会員)
(Vol. 31, No. 3 参照)

館 暉 (正会員)
(Vol. 31, No. 3 参照)