

フィードバック誤差学習の漸近的収束性と学習速度の改善*

新井史人^{*1}, 柴莉莉^{*2}, 福田敏男^{*1}Asymptotic Convergence of Feedback Error Learning Method
and Improvement of Learning Speed

Fumihito ARAI, Lili RONG and Toshio FUKUDA

This paper deals with the improvement of learning speed based on the analysis of convergence of the feedback error learning method. We derive and obtain the condition for the asymptotic convergence of the feedback error learning method for each trial. This condition is the relationship between the learning rate and the α function, which is calculated from the input-output relationship of the system. Using the α function, we propose a high-speed learning method for a tracking control system. We present the simulation results for the tracking control system of a one-link robot manipulator for two cases as follows: (1) use of the general feedback error learning method and (2) use of the proposed high-speed learning method. The simulation results show the effectiveness of the proposed conditions and learning method.

Key Words: Robotics, Response, Neural Network, Feedback Error Learning, Trajectory Control, Asymptotic Convergence, Learning Speed Improvement

1. はじめに

近年, ニューラルネットワークに関する研究が盛んに行われ, 広い分野でニューラルネットワークが使われている^{(1)~(4)}. ニューラルネットワークを使って, 学習制御を行う方法としては教師信号ありの学習方法が一般的である. この中で, ある目標値を教師信号として, ニューラルネットワークの出力と教師信号との誤差を減少するように, 最急こう配法により学習する方法が主流である⁽⁵⁾. ロボットや, プラントの制御系を学習によって獲得する場合などでは学習すべき対象の教師信号が直接得にくいことが多い. このような場合は順モデルを作成し, 並用する方法も提案されている⁽⁶⁾. 川人らによって提案されたフィードバック誤差学習方法⁽⁷⁾では順モデルを必要としないため, ロボット制御システムへの応用例がたくさん報告されている⁽⁸⁾⁽⁹⁾. いずれの場合でも, ある評価関数が少ない学習回数でできるだけ減少することが望ましい.

図1に示すように, 学習には試行時間に関する横軸

方向の学習と試行回数に関する縦軸方向のものの両方がある. ロボットに動的な軌道を追従させるような場合, 毎試行ごとの学習(毎サンプリングの学習ではない)を通して, 全体の追従誤差を要素とする評価関数はどのように変化するのか, また, その変化は学習係数とどんな関係があるのかがわかれば, 学習速度を改善するような学習係数を操作できる.

従来, フィードバック誤差学習の収束性に関する研究としてはある学習時刻 t に注目して, Lyapunov の方法を利用して証明するものがある⁽¹⁰⁾. この場合, t 時刻のフィードバックトルクの試行時間に対する変化が負であることが証明されるだけで, 一つの試行過程ごとに全部のフィードバックトルクの和がどのように変化するのか, また, 学習係数との関係などが明らか

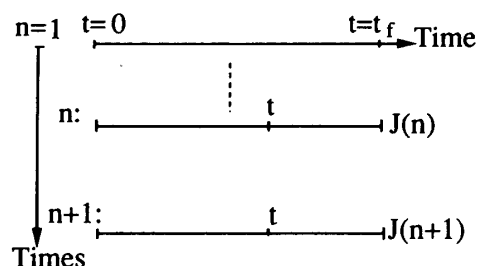


図1 学習の横軸と縦軸の概念図

* 原稿受付 平成4年12月30日.

^{*1} 正員, 名古屋大学工学部 (〒464-01 名古屋市千種区不老町1).^{*2} 名古屋大学工学部.

ではない。そのため、学習速度の改善を目的とするとき、学習係数は試行差誤あるいは経験的に選ぶしかない⁽¹¹⁾。もし学習係数と試行回数ごとの誤差の変化との関係がわかれば、学習速度の改善と学習誤差の減少化が期待できるといえるが、従来、これに関する研究がほとんどない。

本論文では、システムの出力が完全に目標軌道を追従するために、すなわち、制御精度を高くするために、システムの出力と目標軌道とのずれが微小であるという仮定に基づき、学習の漸近的収束性と学習係数との関係を数学的に導くことにより、高速に学習を行うためのアルゴリズムを提案することを目的とする。具体的には、対象物の入出力関係を閉ループで表現することにより、毎回の学習を通して、評価関数が漸的に減少する条件を導き、学習係数と漸近的収束性の関係を与えて、学習速度の改善方法を提案する。最後に、シミュレーションによって、導いた条件式の妥当性と提案する学習方法の有効性を確かめる。ここでいう“漸近的収束性”とはシステムにある動的な軌道を追従させるよう繰返し学習を行う場合、毎試行ごとの学習を通して、全体の誤差を要素とした評価関数が減少していくことである。

2. フィードバック誤差学習則

フィードバック誤差学習のブロック図を図2に示す。

この中で、 G_1 はプラントの入出力関数、 y はプラントの出力、 y_d は目標軌道、 FB はフィードバックコントローラ、 T_{fb} はフィードバックトルクを表す。

フィードバック誤差学習則は次の式で示される⁽⁷⁾。

$$\frac{dw}{dt} = c \frac{\partial T_n}{\partial w} T_{fb} \quad (1)$$

ここで、 w はニューラルネットワークの重み、 c は学習係数である。また、

$$T_{fb} = k_p(y_d - y) + k_v(\dot{y}_d - \dot{y}) \quad (2)$$

で、 $k_p > 0$ 、 $k_v > 0$ である。プラントの出力をある目標

軌道に追従させることを目的とし、その評価関数を次式のように定義する。

$$J(n) = \frac{1}{2} \sum_{k=1}^N T_{fb}(k, n)^2 \quad (3)$$

この中で、 N は1回学習のサンプリング総数、 n は試行学習回数、 k はサンプリング時刻、 $T_{fb}(k, n)$ は n 回め学習するときの k サンプリング時刻のフィードバックトルクを表す。学習が漸的に収束するということは、次の式が成立するということである。

$$\begin{aligned} \Delta J &= J(n+1) - J(n) \\ &= \frac{1}{2} \sum_{k=1}^N T_{fb}(k, n+1)^2 - \frac{1}{2} \sum_{k=1}^N T_{fb}(k, n)^2 < 0 \end{aligned} \quad (4)$$

3. フィードバック誤差学習の漸近収束性

3.1 システムの等価的表現 図2のシステムを図3で表すことができる。ここで、 G_2 は閉ループの伝達関数である。ここで、フィードバックゲインがシステムの安定性を保証することを仮定する。このとき、システムの出力 y 、 $\dot{y}$ およびニューラルネットワークの出力 T_n は次のように表すことができる。

$$y = y(y_d, T_n) \quad (5)$$

$$\dot{y} = \dot{y}(y_d, T_n) \quad (6)$$

$$T_n = T_n(y_d, w) \quad (7)$$

3.2 学習が漸的に収束するための条件 ここで、フィードバック誤差学習によるシステムの漸近的収束の条件を導く。まず、 ΔJ を計算する。

$$\begin{aligned} T_{fb}(k, n+1) &= k_p\{y_d(k) - y(k, n+1)\} \\ &\quad + k_v\{\dot{y}_d(k) - \dot{y}(k, n+1)\} \\ &= k_p\{y_d(k) - y(k, n) + y(k, n) \\ &\quad - y(k, n+1)\} + k_v\{\dot{y}_d(k) \\ &\quad - \dot{y}(k, n) + \dot{y}(k, n) - \dot{y}(k, n+1)\} \\ &= T_{fb}(k, n) \\ &\quad + k_p\{y(k, n) - y(k, n+1)\} \\ &\quad + k_v\{\dot{y}(k, n) - \dot{y}(k, n+1)\} \end{aligned}$$

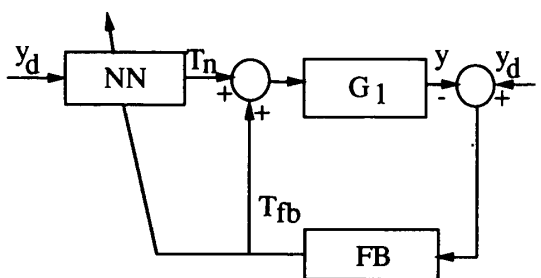


図2 フィードバック誤差学習

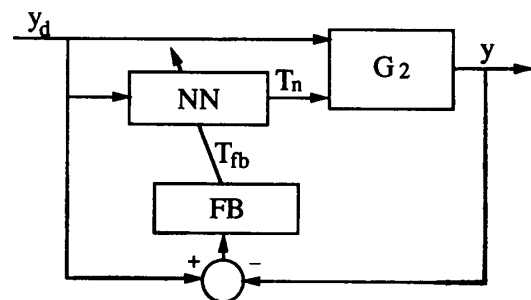


図3 フィードバック誤差学習の等価ブロック線図

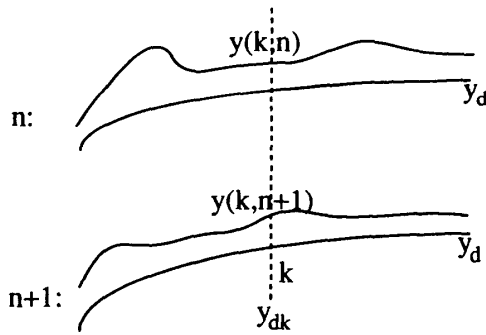


図4 学習の概念図

$$= T_{fb}(k, n) + \Delta T_{fb} \dots\dots\dots (8)$$

すると、 ΔJ を次のように表すことができる。

$$\Delta J = \frac{1}{2} \sum_{k=1}^N (2T_{fb}(k, n) + \Delta T_{fb}) \Delta T_{fb} \dots\dots\dots (9)$$

次に ΔT_{fb} を計算する。図4に示すようにあるサンプリング時間 k において、 n 回めの学習と $n+1$ 回めの学習のとき、目標軌道 y_{dk} からのずれを考える。式(5)～(7)を使って、次の式が書ける。

$$\Delta y = \frac{\partial y}{\partial y_d} \Delta y_d + \frac{\partial y}{\partial T_n} \Delta T_n \dots\dots\dots (10)$$

$$\Delta \dot{y} = \frac{\partial \dot{y}}{\partial y_d} \Delta y_d + \frac{\partial \dot{y}}{\partial T_n} \Delta T_n \dots\dots\dots (11)$$

$$\Delta T_n = \frac{\partial T_n}{\partial y_d} \Delta y_d + \frac{\partial T_n}{\partial w} \Delta w \dots\dots\dots (12)$$

k 時刻に、ニューラルネットワークの理想出力を T_{dk} とする。 n 回めの学習のとき、 k 時刻にて次の式が成立する。

$$\begin{aligned} y(k, n) - y_{dk} \\ = \frac{\partial y}{\partial y_{dk}} \Delta y_d + \frac{\partial y}{\partial T_{nk}} \{T(k, n) - T_{dk}\} \dots\dots\dots (13) \end{aligned}$$

同じように、 $n+1$ 回め学習のとき、 k 時刻にて次の式が成立する。

$$\begin{aligned} y(k, n+1) - y_{dk} \\ = \frac{\partial y}{\partial y_{dk}} \Delta y_d + \frac{\partial y}{\partial T_{nk}} \{T(k, n+1) - T_{dk}\} \dots\dots\dots (14) \end{aligned}$$

ただし、式(13)の中の $\partial y / \partial T_{nk}$ と式(14)の中の $\partial y / \partial T_{nk}$ は厳密には同じではないが、ここで、式(13)と(14)の中の $\partial y / \partial T_{nk}$ は同じであると仮定する(付録参照)。

すると、

$$\begin{aligned} y(k, n) - y(k, n+1) \\ = \frac{\partial y}{\partial T_{nk}} \{T(k, n) - T(k, n+1)\} \dots\dots\dots (15) \end{aligned}$$

k 時刻に、ニューラルネットワークの理想重みを w_d とすると、次の式がある。

$$T(k, n) - T_{dk}$$

$$= \frac{\partial T_n}{\partial y_{dk}} \Delta y_d + \frac{\partial T_n}{\partial w_k} \{w(k, n) - w_d\} \dots\dots\dots (16)$$

$$T(k, n+1) - T_{dk}$$

$$= \frac{\partial T_n}{\partial w_k} [w(k, n+1) - w_d] + \frac{\partial T_n}{\partial y_{dk}} \Delta y_d \dots\dots\dots (17)$$

同様に、式(16)と(17)の中の $\partial T_n / \partial w_k$ は同じであると仮定する。

すると、

$$T(k, n) - T(k, n+1)$$

$$= \frac{\partial T_n}{\partial w_k} \{w(k, n) - w(k, n+1)\}$$

$$= -\frac{\partial T_n}{\partial w_k} \sum_{j=k}^{N+k-1} \left(c \frac{\partial T_n}{\partial w} T_{fb} \right)_j \dots\dots\dots (18)$$

となり、この式を式(15)に代入すると、

$$\begin{aligned} y(k, n) - y(k, n+1) \\ = -c \frac{\partial y}{\partial w_k} \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \dots\dots\dots (19) \end{aligned}$$

が求められ、同じように、

$$\begin{aligned} \dot{y}(k, n) - \dot{y}(k, n+1) \\ = -c \frac{\partial \dot{y}}{\partial w_k} \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \dots\dots\dots (20) \end{aligned}$$

が求められる。

上の二つの式を式(9)に代入して整理すると、 ΔJ は次のように求められる。

$$\begin{aligned} \Delta J = c \sum_{k=1}^N \left\{ \left(T_{fb} \frac{\partial T_{fb}}{\partial w} \right)_k \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \right\} \\ + \frac{1}{2} c^2 \sum_{k=1}^N \left\{ \left(\frac{\partial T_{fb}}{\partial w} \right)_k \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \right\}^2 \dots\dots\dots (21) \end{aligned}$$

学習が漸的に収束する、つまり式(4)が成立するための条件は次のようになる。

$$\begin{aligned} c^2 < - \frac{2 \sum_{k=1}^N \left\{ \left(T_{fb} \frac{\partial T_{fb}}{\partial w} \right)_k \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \right\}}{\sum_{k=1}^N \left\{ \left(\frac{\partial T_{fb}}{\partial w} \right)_k \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \right\}^2} c \dots\dots\dots (22) \end{aligned}$$

$$\begin{aligned} \alpha = - \frac{2 \sum_{k=1}^N \left\{ \left(T_{fb} \frac{\partial T_{fb}}{\partial w} \right)_k \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \right\}}{\sum_{k=1}^N \left\{ \left(\frac{\partial T_{fb}}{\partial w} \right)_k \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \right\}^2} \dots\dots\dots (23) \end{aligned}$$

と定義すると、上の式は次のように書ける。

$$c^2 < \alpha c \dots\dots\dots (24)$$

この式は学習の漸近的収束性を保証するものである。つまり、学習により、各試行ごとに評価関数 J が減少するためには、学習係数 c はこの式を満足するように選ばばよい。 r を1より小さい正数として、次の式が満足するように c を選ばば、学習は漸的に収束する。

$$c = r\alpha \dots\dots\dots (25)$$

3.3 漸近的収束条件式に関する一考察 学習の収束の特性は α と密接な関係がある。 α の符号はその分子によって決まるため、ここで、 α の分子の性質を見てみよう。

$$Y = \sum_{k=1}^N \left\{ \left(T_{fb} \frac{\partial T_{fb}}{\partial w} \right)_k \sum_{j=k}^{N+k-1} \left(\frac{\partial T_n}{\partial w} T_{fb} \right)_j \right\} \quad \dots\dots\dots (26)$$

と定義する。また、

$$x = \begin{bmatrix} T_{fb}(1, n) \\ \vdots \\ T_{fb}(N, n) \end{bmatrix}, x_1 = \begin{bmatrix} T_{fb}(1, n+1) \\ \vdots \\ T_{fb}(N, n+1) \end{bmatrix}$$

とすると、 Y は次のように表すことができる。

$$Y = x^T A_1 x + x^T A_2 x_1 \quad \dots\dots\dots (27)$$

また、

$$T_{fb}(k, n+1) = a_{3kk} T_{fb}(k, n) \quad \dots\dots\dots (28)$$

とすると、 x と x_1 は次の関係がある。

$$x_1 = A_3 x \quad \dots\dots\dots (29)$$

A_3 は対角行列である。この式を式(27)に代入すると、 Y は次のように 2 次形式で表現できる。

$$Y = x^T A_1 x + x^T A_2 A_3 x = x^T A x \quad \dots\dots\dots (30)$$

ここで、 A は実対称行列、

$$A = \frac{1}{2} \{ (A_1 + A_2 A_3) + (A_1 + A_2 A_3)^T \},$$

$$a_{ii} = \frac{\partial T_{fb}}{\partial w_i} \frac{\partial T_n}{\partial w_i} = \frac{\partial T_{fb}}{\partial T_n} \left(\frac{\partial T_n}{\partial w_i} \right)^2$$

$$a_{ij} = \frac{1}{2} \left(\frac{\partial T_{fb}}{\partial w_i} \frac{\partial T_n}{\partial w_j} + a_{3ii} \frac{\partial T_{fb}}{\partial w_j} \frac{\partial T_n}{\partial w_{N+i}} \right)$$

$$a_{ij} = a_{ji}, i, j = 1, 2, \dots, N \quad \dots\dots\dots (31)$$

となる。2 次形式の理論によると、 $Y > 0$ あるいは $Y < 0$ の必要かつ十分条件は行列 A は正定あるいは負定である必要がある。行列 A が負定であれば、 α が正であり、 c が α に比べある小さい正の定数で学習を行うと、学習は漸近的に収束する。

2 次形式 Y は変数の適当な直交変換 $x = yT$ (T : 実直交行列)によって、次のような標準形になる。

$$Y = \lambda_1 y_1^2 + \dots + \lambda_N y_N^2 \quad \dots\dots\dots (32)$$

ここで、 $\lambda_1, \dots, \lambda_N$ は行列 A の固有値である。もし、行列 A のすべての固有値が負であれば、 α が正であり、

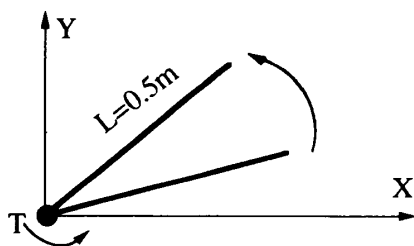


図 5 1 リンク剛体モデル

c が α に比べ、ある小さい正の定数として学習を行うと、学習は漸近的に収束する。

行列 A の要素はシステムの入出力関係を依存する。言い替えれば、システムの入出力関係を変えることによって、行列 A の性質が決められる。

4. シミュレーション

4.1 シミュレーションモデル ここで、シミュレーションによって、学習係数 c に対する α と学習収束特性の関係を確かめ、学習速度を改善する方法を

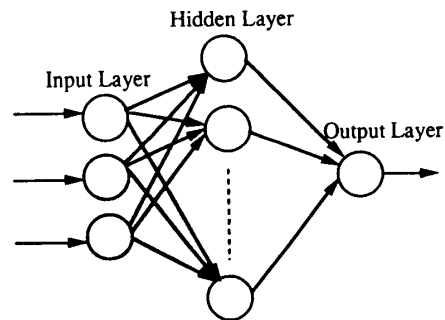


図 6 ネットワークの構造図

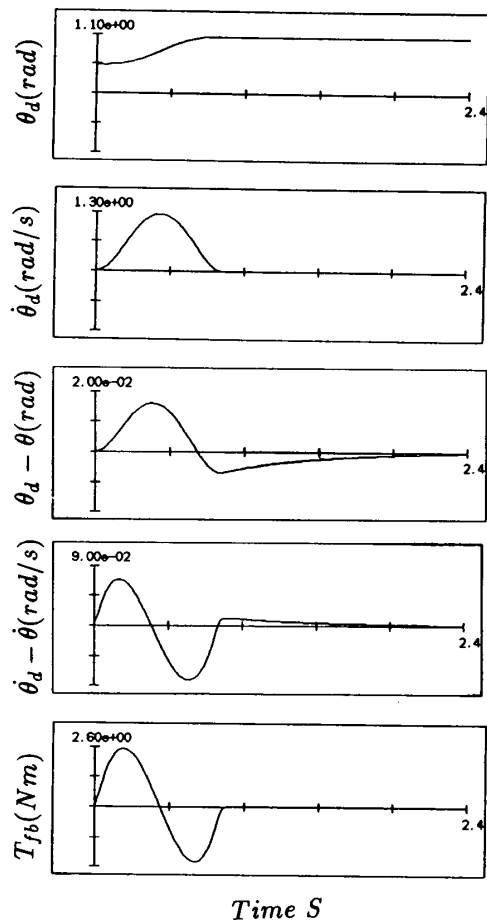


図 7 学習なしのシステムの応答

提案する。

シミュレーションモデルは1リンクの剛体アームで、図5に示す。リンクの長さは0.5 mで、関節は初期値の30度から目標値の60度まで動くとする。角度の目標軌道は時間に関する5次多項式で与える。

$$\theta_d(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \cdots (33)$$

境界条件は

$$\theta_0 = 30, \theta_{t_f} = 60,$$

$$\dot{\theta}_0 = \dot{\theta}_{t_f} = \ddot{\theta}_0 = \ddot{\theta}_{t_f} = 0 \cdots (34)$$

である。整定時間 $t_f = 0.8$ 秒であり、サンプリング時間は0.004秒である。

本研究で用いたニューラルネットワークは、3入力1出力で、中間層に8個のユニットを有する3層のものである。中間層のユニットの応答関数には式(35)に示すシグモイド関数を用いて、出力層のユニットの応答関数には式(36)に示すシグモイド関数を用いた。

$$f(x) = \frac{1}{1 + e^{-x}} \cdots (35)$$

$$f(x) = \frac{1}{1 + e^{-x}} - 0.5 \cdots (36)$$

ネットワークの構造を図6に示す。ネットワークの重みの初期値は乱数によって $-0.25 \sim 0.25$ の範囲で与えた。

4.2 学習係数を一定とした場合のシミュレーション結果 図7に示すのは学習なしの場合の角度、角速度の目標値、誤差値、およびフィードバックトルクである。

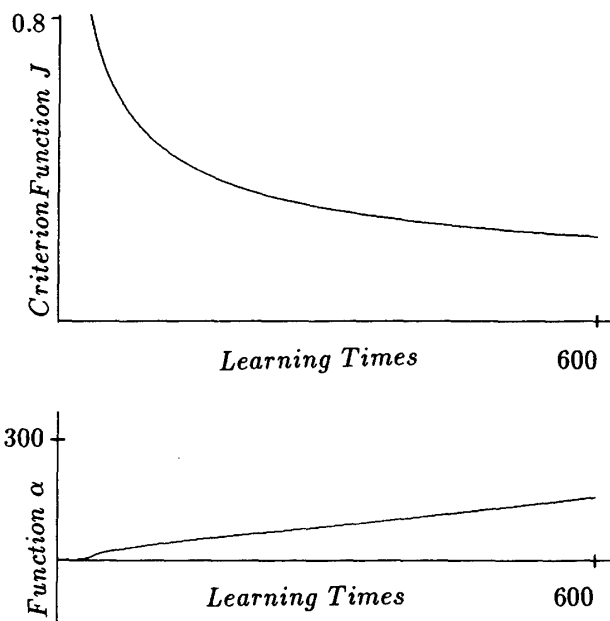


図8 $c=0.05$ のとき学習による応答

図8に示すのは学習係数 $c=0.05$ のとき600回学習した後の評価関数 J および α である。このとき、 α はずっと正で、 J も漸近的に減少しつづけていくことがわかる。

図9に示すのは学習係数を $c=0.5$ と大きくしたと

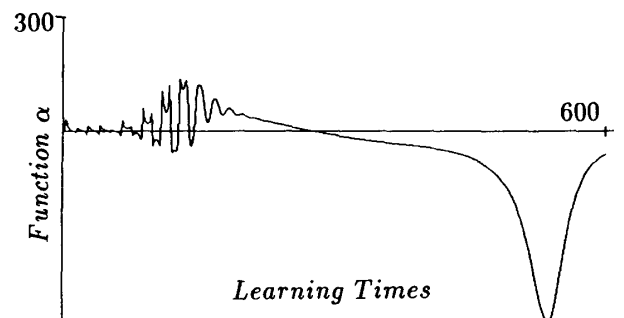
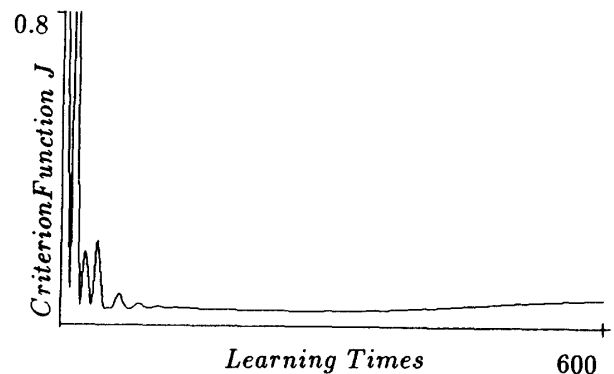
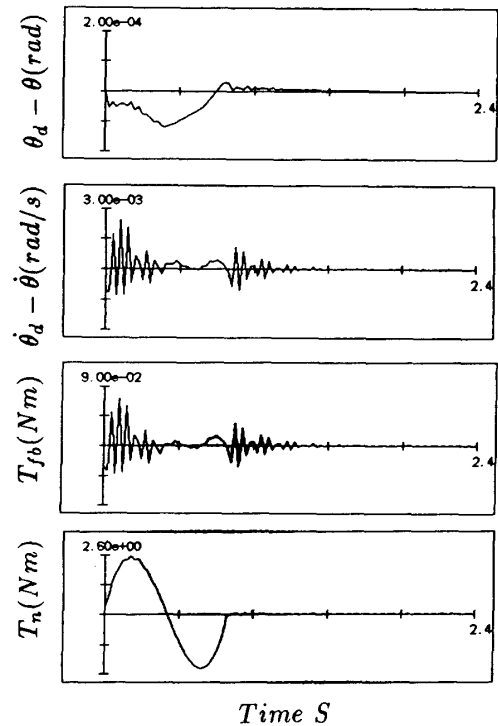
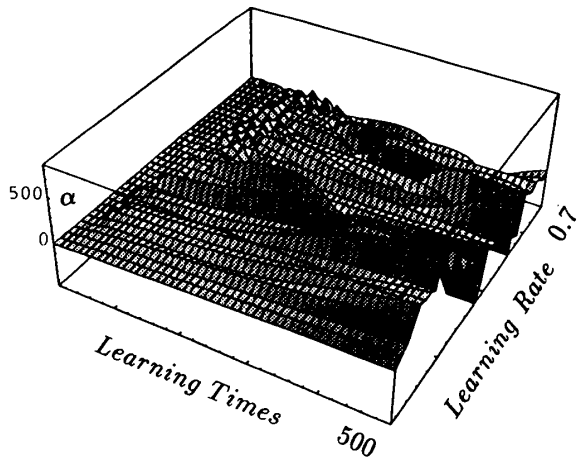
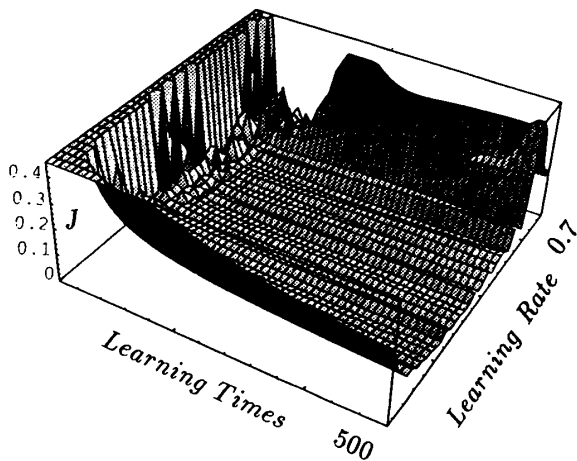


図9 $c=0.5$ のとき学習による応答

図 10 学習係数 c と α の関係図図 11 学習係数 c と評価関数 J の関係図

き、600 回学習した後の角度と角速度の誤差値、フィードバックトルク、ニューラルネットワークの出力トルク、評価関数 J および α である。この図から、 α は正の場合もあるし、負の場合もあり、 α が正のとき J は減少し、 α が負のとき J は増加していくことがわかる。

図 10 に示すのは学習試行回数と α と c の関係で、図 11 に示すのは学習試行回数と J と c の関係である。学習回数は 500 回で、 c は 0.04 から 0.7 まで 0.02 ずつ変化させた。

この二つの図から、学習係数 c がある程度大きくなると、 J は素速くある小さい値まで下がるが、学習を続けると、 α は正になったり、負になったりして、 J はかえって大きくなってしまふ。逆に、学習係数 c が小さすぎると、 J は減少していくが、収束に時間がかかるということがわかった。

4.3 学習速度の改善とシミュレーション結果

ここで、4.2 節の結果を参考に、学習速度を改善する

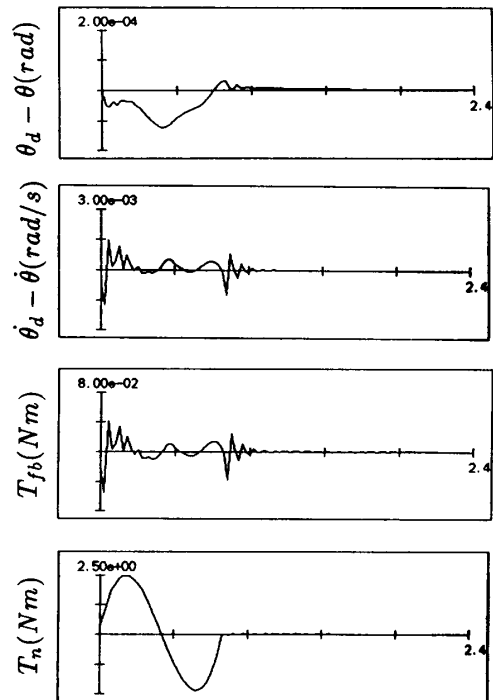
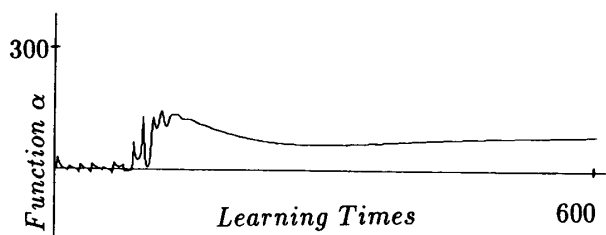
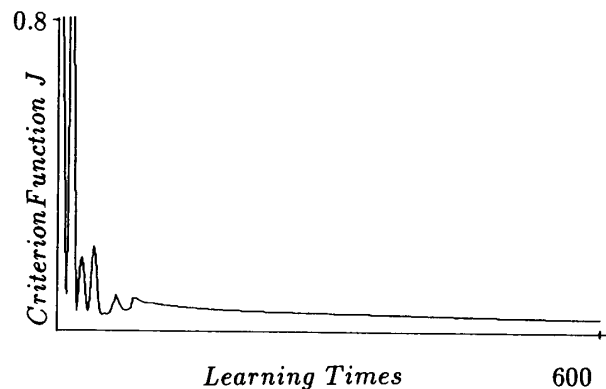
Time S 

図 12 提案した学習方法による応答

ための学習アルゴリズムを提案する。

始めに、学習係数 c を比較的大きく選んで、 J をある小さい値まで下げる。 α が負になる場合は、学習係数 c を小さくして、学習を続ける。このように漸近的な収束条件式を監視しながら c を適宜設定し直すことにより、 J は漸的に小さくなり、学習速度が改善できる。

図 12 に示すのは以上の学習アルゴリズムを利用して 600 回学習した後の角度と角速度の誤差値、フィードバックトルク、ニューラルネットワークの出力トルク、評価関数 J および α の曲線である。ここでは始め、学習係数 c を 0.5 とし、 α の変化に応じて、式(25)を使って c を変更する。ここで、 $r=0.8$ とした。 α の変化は激しいときもあるし、緩やかなときもあるため、いつ、どのタイミングで c を変えるかは検討の余地があるが、今回は α が連続 5 回以上 -1 より小さくなると、 c を変えるアルゴリズムを採用した。ただし、 c の変化の回数があり多くなると、 c が小さ過ぎて、収束の速度が遅くなる可能性があることに注意を要する。

図 12 と 9 を比べてみると、本論文で提案した学習方法を用い、学習係数を適宜変えることによって、評価関数 J を漸近的に減少することができ、また、学習速度も速くなることがわかる。

3 章で述べたように、 α はシステムの入出力関係と密接な関係がある。学習が漸近的に収束することを保証する条件式(24)を成立するためには学習係数 c と α の変化関係を調べる必要がある。問題はどのように c を調節して条件式(24)を成立させるかにあるが、 α がシステムの入出力関係と密接な関係にある以上、未知の制御対象に対して各試行ごとに c を厳密に決定することは困難である。シミュレーション結果から今回提案した手法の有効性が得られたが、学習係数のより適切な変化方法に関しては検討の余地があり、今後の検討課題と考えられる。

5. お わ り に

本論文において、フィードバック誤差学習の各試行ごとの漸近的収束性を検討することによって、フィードバック誤差学習の学習速度を改善する方法を提案した。対象物の入出力関係を閉ループで表現することにより、毎回の学習を通して、ある評価関数が漸近的に減少するための条件を導いた。これによって、学習速度の改善のための学習アルゴリズムを提案した。そして、シミュレーションによって、導いた条件式の妥当性を確かめて、提案した学習方法の有効性を確認した。

付 録

式(13)から式(14)を引くと次の式が書ける。

$$y(k, n) - y(k, n+1) = \left(\frac{\partial y}{\partial T_{nk}} \right)_n T(k, n) - \left(\frac{\partial y}{\partial T_{nk}} \right)_{n+1} T(k, n+1)$$

$$\begin{aligned} & + \left\{ \left(\frac{\partial y}{\partial T_{nk}} \right)_{n+1} - \left(\frac{\partial y}{\partial T_{nk}} \right)_n \right\} T_{dk} \\ & = \left(T \frac{\partial y}{\partial T_{nk}} \right)_n - \left(T \frac{\partial y}{\partial T_{nk}} \right)_{n+1} \\ & + \left\{ \left(\frac{\partial y}{\partial T_{nk}} \right)_{n+1} - \left(\frac{\partial y}{\partial T_{nk}} \right)_n \right\} T_{dk} \\ & = - \left\{ \frac{\partial}{\partial n} \left(T \frac{\partial y}{\partial T_{nk}} \right) \right\}_n \Delta n \\ & + T_{dk} \left[\frac{\partial}{\partial n} \left(\frac{\partial y}{\partial T_{nk}} \right) \right]_n \Delta n \\ & = - \left\{ \frac{\partial T}{\partial n} \left(\frac{\partial y}{\partial T_{nk}} \right) \right\}_n \Delta n - \left\{ T \frac{\partial}{\partial n} \left(\frac{\partial y}{\partial T_{nk}} \right) \right\}_n \Delta n \\ & + T_{dk} \left\{ \frac{\partial}{\partial n} \left(\frac{\partial y}{\partial T_{nk}} \right) \right\}_n \Delta n \\ & = \left[- \left\{ \frac{\partial T}{\partial n} \left(\frac{\partial y}{\partial T_{nk}} \right) \right\}_n \right. \\ & \quad \left. - (T - T_{dk}) \left\{ \frac{\partial}{\partial n} \left(\frac{\partial y}{\partial T_{nk}} \right) \right\}_n \right] \Delta n \\ & = - \left(\frac{\partial T}{\partial n} \right)_n \left(\frac{\partial y}{\partial T_{nk}} \right)_n \Delta n \\ & \quad - (T - T_{dk}) \left\{ \left(\frac{\partial y}{\partial T_{nk}} \right)_{n+1} - \left(\frac{\partial y}{\partial T_{nk}} \right)_n \right\} \end{aligned}$$

目標軌道からの微小変化を考えて、 $T - T_{dk}$ と $(\partial y / \partial T_{nk})_{n+1} - (\partial y / \partial T_{nk})_n$ は微小で、その積を無視することができると仮定する。このとき、上の式は次のようになる。

$$\begin{aligned} & y(k, n) - y(k, n+1) \\ & = - \left(\frac{\partial T}{\partial n} \right)_n \left(\frac{\partial y}{\partial T_{nk}} \right)_n \Delta n \\ & = - \left\{ \left(\frac{\partial T}{\partial n} \right)_n \Delta n \right\} \left(\frac{\partial y}{\partial T_{nk}} \right)_n \\ & = - \frac{\partial y}{\partial T_{nk}} \{ T(k, n) - T(k, n+1) \} \end{aligned}$$

式(16)と(17)の中の $\partial T_n / \partial w_k$ についても同じように導けるが、ここでは省略する。

文 献

- (1) Kitamura, S., Neural Networks and Inverse Problems in Measurement and Control, *Proc. of the IMACS/SICE Int. symp. on Robotics, Mechatronics and Manufacturing Systems '92 Kobe, Japan, Sep. 16-20 (1992)*, 205-210.
- (2) Cetinkunt, S. and Chiu, H. T., A Study of Learning Controllers For Tip Position Control of A Flexible Arm Using Artificial Neural Networks, *DSC-Vol. 31, Modeling and Control of Compliant and Rigid Motion Systems, ASME*, (1991), 15-19.
- (3) 山田・藪田, ニューラルネットワークを使用したフィードフォワードフィードバックコントローラ, 第 3 回電磁力関連のダイナミックスシンポジウム講演論文集, 91-6 (1991-6), 330-333.
- (4) 吉田・小林, 多自由度系に対するアクティブ動吸振器のニューラルネットワークコントローラの設計, 第 4 回シンポジウム電磁力関連のダイナミックス講演論文集, (1992), 359-364.
- (5) Rumelhart, D. E., Hinton, G. E. and Williams, R. J.,

- Learning Representations by Error Propagation, In D. E. Rumelhart, J. L. McClelland and the PDP Research Group, Vol. 1, 318/362(1986), MA : MIT Press, Parallel Distributed Processing, Cambridge.
- (6) Jordan, M. I., *Genetic Constraints on Underspecified Target Trajectories*, Proc. IJCNN, Vol. 1 (1989), 217-255.
- (7) 川人・宇野・鈴木, 随意運動における適応と学習 2, 日本ロボット学会誌, 6-3(1988), 222-230.
- (8) 関口・長田, ニューラルネットワークによるロボットマニピュレータの逆ダイナミクス制御, 第5回生体, 生理工学シンポジウム, (1990-11), 57-60.
- (9) Arai, F., Fukuda, T., Tanaka, T. and Shibata, T., Hierarchical Control System for Flexible Materials Handling Robots Using Neural Networks, IROS'92, 1 (1992), 534-541.
- (10) Gomi, H. and Kawato, M., Learning Control for a Closed Loop System using Feedback-Error-Learning, Proc. of the 29th Conference on Decision and Control, (1990), 3289-3294, Honolulu, Hawaii,
- (11) 福田・柴田・ほか2名, 神経回路網によるロボット・マニピュレータの位置・力制御の研究(第2報, Time Delay 型 N. N. の位置と力のハイブリット制御の応用), 機論, 57-535, C(1991), 220-227.
-