多層ネットワークによる倒立振子の安定化 学習制御*

池田直人**・斎藤真也***・北村新三***

Learning Control for Stabilization of an Inverted Pendulum Using a Multi-layered Neural Network*

Naohito IKEDA**, Shinya SAITO*** and Shinzo KITAMURA***

We propose a learning control scheme using a multi-layered neural network such that the control performance is improved for unknown controlled objects by repeated trials. This system was applied to the stabilization of an inverted pendulum. The system consists of three subsystems; a neurocontroller, a temporary target generator and an error evaluator. The temporary target generator yields reference angle of the inverted pendulum to control the cart position. The error evaluator generates a teaching signal for the neuro controller by comparing the system error with an error-reference model. The backpropagation algorithm was used for the learning of neural network. The learning time could be shortened through the pre-learning process by imbedding *a priori* knowledge in the neural network. We verified the effectiveness of the proposed method by computer simulation.

1. はじめに

制御系を設計する場合、その制御対象に関する既知の知識、要求される制御性能などにより、さまざまな設計方法が考えられる。しかし、従来の制御系設計の主流は、数式モデル（数値モデル）にその基礎を置くものであった。これに対し、近年、コントローラの特性を言語ルールで表現するルールベース制御やファジィ制御が提案されている。さらに、このような言語表現を用いない方法として、ニューラルネットワークの応用も研究されつつある。これらは、特に、制御対象の同定が難しい場合や非線形特性を示す対象の場合の数式モデルを基礎とした制御系設計法の欠陥を補うもので、前者は、人間の持つ経験的知識により、未知部分の多い制御系に対応しようとするものであり、後者は、ニューラルネットワーク

の結合係数上に、試行・学習により、未知の制御対象の入出力間の写像関係を捉え、系をより望ましい状態へ導こうとするものである。

我々のグループでも、ニューラルネットワークをコントローラとして用いる学習制御系をすでに提案した¹。これは、制御対象についてのある程度の先験的知識と、ニューラルネットワークのもつ学習能力、汎化能力に注目したもので、定性的にはある程度の先験的知識があるが定量的には未知部分の多い制御対象に対して試行によってフィードバック系としての制御則を獲得することを目的としている。本論文では、この学習制御系を倒立振子の安定化制御に適用し、そのシミュレーション結果を述べる。

本論文の構成は次の通りである。2. では、倒立振子のモデルとその制御の研究の現状、および本論文で用いる方法について述べ、3. では、ニューラルネットワークを用いた制御系を倒立振子の安定化制御問題に適用する方法を述べる。そして、4. においてシミュレーション結果およびこの制御系の有効性に対する考察を行なう。

2. 倒立振子のモデルおよび安定化制御研究の現状

倒立振子は、Fig.1のようなモデルで表わされる非線

* 原稿受付 1989. 11. 30

** 三菱重工業 株式会社 システム技術部 System Engineering Department, Mitsubishi Heavy Industries, Co. Ltd.; 1-1, Wadasaki-cho, 1-chome, Hyogo-ku, Kobe 652, JAPAN

*** 神戸大学 工学部 Department of Instrumentation Engineering, Kobe University; Rokkodai 1-1, Nada, Kobe 657, JAPAN

Key Words : neural network, inverted pendulum, learning control, prelearning.

Fig. 1 Inverted pendulum system

形で不安定なシステムであり、古くから線形制御理論の研究対象²⁻³や二足歩行ロボットのモデルとして広く対象とされてきた⁴。また、ルールベース制御やファジィ制御の応用も試みられている⁵⁻⁷。一般に、線形制御理論では制御対象のパラメータの正確な値が必要であり、また、振子が大きく傾くと線形モデルが成立しない。一方、ルール制御では、経験則により振子角度のみの制御は簡単に行なえる。しかし、台車位置も制御しようとする振子と台車の間に存在する複雑な力学関係のために適切な操作を if-then 形式の制御則で記述することが現実には困難なようである^{8,9}。よって、上のような数式あるいは経験則を用いるのではなく、ニューラルネットワークによる学習制御が考えられる。これは制御に言語表現を用いずに、試行により制御則を獲得する方法である。ニューラルネットワークを用いて最初に倒立振子制御の研究を行なったのは Widrow¹⁰ である。彼は約 25 年前に ADALINE (Adaptive Linear Neuron) を提案し、量子化した倒立振子の四つの状態変数を ADALINE の入力とし、Bang-Bang 制御の切り替え線を学習によって再現できることを示した。このニューラルネットワークの学習には Widrow-Hoff LMS アルゴリズムを用いている。また、Barto と Sutton¹¹ は教師なし学習能力のある ASE (Associative Search Element) の利用を提案している。このニューラルネットワークの入力は、Widrow らのものと似たものであるが、学習には“あめ”と“むち”をもとにした強化学習 (Reinforcement learning) を用いている。さらに、最近、Anderson¹² は、Barto らの方法を応用し、“Temporal Difference Method”と“Reinforcement learning”を用いて状態変数の値そのものを入力とした多層型ニューラルネッ

トワークの学習により制御を行なっている。ただし、これらの方法は、ニューラルネットワークの出力層セルの値が 2 値 (0 or 1) である。また、Tolat と Widrow¹³ は視覚情報を入力として ADALINE を用いて制御を行なった例を示した。また、制御系の構成は不明であるが、Hecht-Nielsen 社は 1988 年の国際神経回路網学会において実機によるデモを行なっている。さらに、Guez と Selinsky¹⁴ は人間が倒立振子を制御する制御入力を教師信号として、人間の制御方法をニューラルネットワークが学習する制御系を提案している。

一般に、ニューラルネットワークを制御系に応用する際に問題となるのは、教師付き学習を行なう際の (一時的な) 教師信号をどのように生成するかである。教師信号の生成の仕方により、学習制御系の構成が異なり、また、学習効率も変わってくる。この生成機構には、人間の持つ先験的知識をできるだけ埋め込むことが、学習効率を上げることになる。以下では、倒立振子の安定化を対象として、仮想目標値発生部、評価部による一時的な教師信号発生を用いたニューラルネットワークによる学習制御系を提案する¹⁵。また、人間の持つ制御に関する定性的知識を、プレラーニングという形で、前もってニューラルネットワークの結合係数上に与えて置く方法も有効であることを示す。

3. ニューラルネットワークを用いた学習制御系

Fig. 2 に倒立振子の安定化制御の制御系を示す。これは、倒立振子制御の目的が「振子のバランスをとりながら ($\theta = 0$)、台車を原点 ($x = 0$) に移動すること」であるので目標値が一定であること、そして一つの制御入力で台車と振子の二つを制御しなければならないという条件に基づいた構成である。

3.1 倒立振子部

Fig. 1 において台車は入力 F を受け、直線のレール上を移動するとすると、制御対象の倒立振子は次の非線形方程式で表わされる¹⁶。

$$(M_p + M_c)\ddot{x} + M_p L_p \cdot \cos \theta \cdot \ddot{\theta} = -F_c \dot{x} + M_p L_p \cdot \sin \theta \cdot \dot{\theta}^2 + F \quad (1)$$

$$M_p L_p \cdot \cos \theta \cdot \ddot{x} + (I_p + M_p L_p^2) \ddot{\theta} - M_p L_p \cdot \sin \theta \cdot g = -C \cdot \dot{\theta} \quad (2)$$

式の中の諸パラメータは、Table 1 に示すとおりである。

3.2 仮想目標値発生部

ここでは、時刻 $t = nT$ (T : サンプル周期) における台車の位置と速度 ($x(n)$, $\dot{x}(n)$) からこれを原点

Fig. 2 Learning control system for stabilization of an inverted pendulum

(0, 0) に移すためには、その時刻での振子の角度と角速度 ($\theta(n)$, $\dot{\theta}(n)$) がどうなればよいのかという仮想目標値 ($\theta_r(n)$, $\dot{\theta}_r(n)$) を次式で発生する。

$$\theta_r(n) = 2 \cdot \theta_{\max} \cdot (1 / (1 + \exp(a(x(n) + b \cdot \dot{x}(n)))) - 0.5) \quad (3)$$

$$\begin{aligned} \dot{\theta}_r(n) &= \frac{d\theta_r(n)}{dt} \\ &= -2 \cdot \theta_{\max} \cdot a(\dot{x}(n) + b \cdot \ddot{x}(n)) \cdot \frac{\exp(a(x(n) + b \cdot \dot{x}(n)))}{(1 + \exp(a(x(n) + b \cdot \dot{x}(n))))^2} \quad (4) \end{aligned}$$

ここに、

$\theta_{\max} (> 0)$: 仮想目標角度の最大値を与える定数

$a (> 0)$: シグモイド関数の傾きの定数

$b (> 0)$: 位置と速度の比率の定数

である。また、(3)、(4) 式の計算時間は十分速いと考え、 $t = nT$ における台車の位置、速度および加速度から同時刻での θ_r と $\dot{\theta}_r$ を求めるものとする。

この式は、「台車の位置が原点から離れるほど、振子の仮想目標角度を鉛直方向から原点側により大きく傾ける」という知識を表わしている。また、台車の速度も考

Table 1 Parameters of inverted pendulum system

Cart Mass	M_c	0.48	kg
Pendulum Mass	M_p	0.16	kg
Length between the Axis and the Center of Gravity	L_p	0.25	m
Moment of Inertia around the Center of Gravity	I_p	0.043	kg · m ²
Acceleration of Gravity	g	9.8	m/s ²
Cart Friction Coefficient	F_c	3.82	kg/s
Pendulum Friction Coefficient	C	0.0022	kg · m ² /s

慮する必要があるため速度成分も含まれている。また、振子の目標角度が大きくなり過ぎないようにするために、シグモイド関数を用いて台車が原点からある一定の距離以上離れても目標角度は一定値以上大きくならないようにしてある。

3.3 評価部

評価部は、コントローラの出力（制御対象の操作量）の良し悪しを評価し、それをもとにニューロコントローラが学習するための教師信号を生成する。

一般にフィードバック制御系ではコントローラの動作

は制御偏差に着目するが、偏差とコントローラ出力の関係は制御対象の特性によって変わるべきものであり、単純には定められない。そこで、ここでは偏差とその微分($e(n)$, $\dot{e}(n)$)が1サンプリング時間後にどうなれば良いかを評価するために規範偏差(e_r , $\dot{e}_r$)を考え、これを用いて操作入力の良さを評価する事にする。ここでは規範偏差を発生するのに次の2次系の方程式を用いる。

$$\ddot{e}_r(t) + 2\zeta\omega_n\dot{e}_r(t) + \omega_n^2 e_r(t) = 0 \quad (5)$$

ここで減衰係数 ζ 、固有角周波数 ω_n は偏差の減少の程度を規定するものであり、設計者によって与えられる。この式において $e_r(t)$, $\dot{e}_r(t)$ の時間変化を位相面に描くとFig. 3の曲線ようになる。この図より実際の偏差とその微分 $e(n)$ と $\dot{e}(n)$ が上の2次系の微分方程式に従って減少するような制御を行えば最終的にはともに零に収束し制御目標が達成できる。よって、ここでは仮想目標発生部で台車の位置より発生された振子の仮想の目標角度と角速度を用いて、 $e_0 (= e(n))$ および $\dot{e}_0 (= \dot{e}(n))$ を、

$$\begin{cases} e_0 = \theta_r(n) - \theta(n) \\ \dot{e}_0 = \dot{\theta}_r(n) - \dot{\theta}(n) \end{cases} \quad (6)$$

$$\begin{cases} e_0 = \theta_r(n) - \theta(n) \\ \dot{e}_0 = \dot{\theta}_r(n) - \dot{\theta}(n) \end{cases} \quad (7)$$

として、(5)式の微分方程式を e_0 , $\dot{e}_0$ を初期値として解き、1サンプリング周期後($t = (n+1)T$)の値($e_r(n+1)$, $\dot{e}_r(n+1)$)を規範偏差とする。しかし、実際には $t = nT$ における操作入力 $F(n)$ によって生ずる $t = (n+1)T$ における偏差とその微分($e(n+1)$, $\dot{e}(n+1)$)は、($e_r(n+1)$, $\dot{e}_r(n+1)$)とは一致しない。そこで、Fig. 3のようにこの両者が近付くように操作量を修正しなければならない。このために、その修正量を次式のように定める。

$$\Delta F = -K_L(\dot{e}_r(n+1) - \dot{e}(n+1)) \quad (8)$$

あるいは、

Fig. 3 Trajectories of error reference model

$$\Delta F = -K_L(e_r(n+1) - e(n+1)) \quad (9)$$

の形でよい。ここで K_L は正定数である。

(8), (9)式は、規範偏差と1サンプリング周期後の偏差との差が、それぞれFig. 3の位相面の縦軸(偏差の微分)方向のみ、あるいは、横軸(偏差)方向のみに対して小さくなるようにするための修正量であり、どちらの式を用いても学習・試行過程に差はあるが、最終的に制御目標を達成することができる。これにより、ニューロコントローラは、各サンプリング時刻において、

$$t(n) = F(n) + \Delta F \quad (10)$$

を一時的な教師信号として学習する。

3.4 ニューロコントローラ部

Fig. 2のニューロコントローラは、各サンプリング時刻における振子の角度、角速度、台車の位置および速度($\theta(n)$, $\dot{\theta}(n)$, $x(n)$, $\dot{x}(n)$)を入力とし、これらが仮想目標発生部より与えられる目標値($\theta_r(n)$, $\dot{\theta}_r(n)$, 0, 0)となるように、台車に加えるべき力 $F(n)$ を出力する。したがってニューラルネットワークは入力層4セル、出力層1セルで構成される。また中間層は1層で40セルを用いた(Fig. 4参照)。学習は、上記の評価部で生成される教師信号に基づいて、バックプロパゲーションアルゴリズム¹⁷により行なった。バックプロパゲーションアルゴリズムは、前向き方向の結合を持つ多層型ネットワークの学習アルゴリズムの一つである。多層型ネットワークでは、入力信号はいくつかの入力層ニューロンに入力され、これは、中間層ニューロンを経て、出力層ニューロンに到達する。ここで、出力層ニューロンの出力信号と望ましい出力(教師信号)との偏差が零となるように、ニューロン間の結合係数と各ニューロンのしきい値を変更する。この際、普通に用いられるアルゴリズムは、偏差の2乗値を極小に導く最急降下法である。実際には、結合係数としきい値の変更は、出力層より入力層に向かって成されるので、偏差が逆伝播する意味で、バックプロパゲーションアルゴリズムと呼ばれている。

なお、Fig. 2の制御系においては、正確な教師信号が未知であり、(8)~(10)式を用いて一時的な教師信号 $t(n)$ を生成している。ここで、この $t(n)$ はニューロコントローラの直接の出力ではなく、この出力によって決定される制御対象の出力情報を用いて生成される。この構成の場合、一般に、入出力変数間の感度が動作中に正負に変化するような制御対象に対しては、必ずしもバックプロパゲーションアルゴリズムによる学習が保証されない。しかし、倒立振子のような入出力変数間の感度が常に正であるような制御対象に対しては、学習可能であることが経験的に知られている。よって、前述のように

Fig. 4 The structure of neuro controller

教師信号を生成することが可能となる。

また、試行を行なうに先だって、あらかじめ先験的知識に基づきニューロコントローラがある程度の望ましい操作量を出力するように事前学習を行なっておく（以下プレラーニングと呼ぶ）。これにより、経験的に分かっている情報をコントローラに前もって学習させておくことで、まったく未学習の状態から始めるよりも学習時間が大幅に短縮できる。もちろん、これは、制御対象に関する予備知識によって変わってくるが、多くの場合、定量的にはむずかしくても、定性的には、対象に対する操作方法の知識があると考えてよい。この倒立振子の問題では、ニューラルネットワークの出力が次の形をとるようにプレラーニングを行なう。

$$F = \begin{cases} F_c & (\theta \geq 0.5) \\ 0.6 \cdot F_c & (\theta = 0.4) \\ 0.2 \cdot F_c & (\theta = 0.2) \\ 0.0 & (\theta = 0.0) \\ -0.2 \cdot F_c & (\theta = -0.2) \\ -0.6 \cdot F_c & (\theta = -0.4) \\ -F_c & (\theta \leq -0.5) \end{cases} \quad (11)$$

ここで、 F_c は試行学習を行なう前の、台車に加える力の最大値を与える定数である。この式は、 $\theta, x, \dot{x}$ がい

かなる値の場合でも θ が正か負かにのみに注目して、振子を立てるためには振子の傾いている方向に台車を動かすという先験的知識を表わしたものである。また、Bang・Bang 的出力を避けるため、 $-0.5 < \theta < 0.5$ における出力は F_c と F_c の間の値をとるようにした。ここで、 $\theta = \pm 0.2, \pm 0.4$ のときの出力を上式のように与えて学習しておけば、ニューラルネットワークの汎化能力により $-0.5 < \theta < 0.5$ の範囲の任意の値に対する出

Fig. 5 Output of neuro controller at $(x, \dot{x}) = (0, 0)$ (after prelearning)

力値も連続に得ることができる。プレラーニング終了状態のニューロコントローラの出力の状態と $(\theta, \dot{\theta})$ の関係を図に示すと Fig. 5 のようになる。

4. シミュレーション結果と考察

4.1 シミュレーション結果

前節まで述べてきた制御系構成により、倒立振子の安定化制御のシミュレーションを行なった。制御対象としては、パラメータが Table 1 のような倒立振子を用いる。なお、シミュレーションは、ワークステーション上で行ない、倒立振子、及び偏差参照モデルにおける2次微分方程式の解法としては、ルンゲクッタ法による数値計算を用いた。また、(4)式のパラメータは $\theta_{\max} = 0.2[\text{rad}]$, $a = 5.0$, $b = 1.5$ とし、(5)式のパラメータは $\omega_n = 20.0$, $\zeta = 1.0$ とした。

振子の傾きと台車の原点からの位置の適当な値を初期状態として原点に静止することを目標に試行する。1回の試行時間は10秒間、サンプリング周期は10ミリ秒である。また、試行途中で振子が倒れてしまったり ($\theta < -90^\circ$, $90^\circ < \theta$ の場合)、台車の位置が、原点よりある程度離れると ($x < -2.0[\text{m}]$, $2.0[\text{m}] < x$ の場合)、試行を中止し、次の試行に移る。

ニューラルネットワークは、まず $F_c = 5[\text{N}]$ として (11) 式のようにプレラーニングを行なう。試行中は、各サンプリング時間ごとに学習を行ない、また、試行後に試行中の各時刻における学習データ1000組 ($\theta(n)$, $\dot{\theta}(n)$, $x(n)$, $\dot{x}(n)$) とそれに対する教師信号 $t(n)$: $n = 1, 2, \dots, 1000$ をランダムな順序で並びかえ、さらに1回ずつ学習した。すなわち、1試行当たり1000個の学習データを2回ずつ、2000回の学習を行なう。教師信号の修正量は (8) 式を用いて計算し、式中の学習定数 K_L は、試行中の学習では学習が進みすぎて振子が不安定になるのを防ぐために小さな値 (0.1) とし、試行後の学習においては1.0とした。

振子の振れについての初期値 $\theta(0) = 0.5[\text{rad}]$, $\dot{\theta}(0) = 0$ と台車についての初期値 $x(0) = 0.5[\text{m}]$, $\dot{x}(0) = 0$ の場合の学習制御の結果を Fig. 6 (a), (b) に示す。すなわち、2回目の試行では安定化制御は失敗し、7回の試行を重ねることにより台車、振子とも原点に近づくようになるがまだ時間がかかる。そして、15回目の試行で約5秒後に目標状態を達成するようになっている。他の初期値の場合についても数例のシミュレーションを行なったが、振子の角度の初期値が $\pm 1.0[\text{rad}]$ の範囲であれば、台車の初期位置を適当にとることにより $-2.0 < x < 2.0[\text{m}]$ の範囲を逸脱せずに目標状態を達成した。また、初期値が $-1.2 < \theta < 1.2[\text{rad}]$ の場合は、台車の動作

(a) Response of cart position

(b) Response of pendulum angle

Fig. 6

範囲を大きくすることにより安定化が可能であるがかなりの試行回数を要した。さらに、振子の傾きを大きくした場合は、50回の試行では安定化できなかった。

4.2 考察

4.2.1 学習による制御則の獲得

前節の結果より、ここで提案した方法を用いて、倒立振子を原点に、安定化制御できることがわかった。Fig. 7に15回目の試行後のニューロコントローラの入力に対する出力 F の状態を示す。Fig. 7 (a) は台車が原点 ($x = 0$, $\dot{x} = 0$) にある場合、振子の角度と角速度 $(\theta, \dot{\theta})$ に対するニューロコントローラの出力 F の様子を描いたもので、Fig. 7 (b) は ($x = 0.5$, $\dot{x} = 0$) の場合である。プレラーニングの状態 (Fig. 5) に較べて、制御則が変更されていることがわかる。すなわち、Fig. 7 では、出力の正負の切り替え線が変化しており、また、学習前とくらべて振子の角度が大きいところでは、大きな操作量が出力されるようになっている。また、Fig. 7 (a) では出力の正負の切り替え線が原点を通っているのに対

(a) Output of neurocontroller at $(x, \dot{x}) = (0, 0)$ (after 15th trial) (b) Output of neurocontroller at $(x, \dot{x}) = (0.5, 0)$ (after 15th trial)

Fig. 7

し、Fig. 7 (b) では、 $\theta = -0.2[\text{rad}] = -\theta_{\max}$ ((3) 式参照)、 $\dot{\theta} = 0$ 付近を通っている。つまり、台車の位置が原点より離れているときは、振子が完全な倒立状態 $((\theta, \dot{\theta}) = (0, 0))$ にあっても、これを原点方向に傾けるようにニューラルネットワークは出力している。以上のように、ニューロコントローラは、試行を通して制御目標を達成するための近似的逆システムとしての制御則を操作入力を表わす関数の形として獲得していることがわかる。

4.2.2 ニューラルネットワークの学習

ここで用いた多層型ニューラルネットワークのバックプロパゲーションによる学習については、学習誤差（出力信号と教師信号の差）が極小値へ収束する保証はあるが、それが最小値となるかどうかはわからない。これがこの方法の欠点となる。実際には、局所的な極小値に収

束すると、学習が十分でなく、制御結果が良くないことが予想される。この状況は、ニューラルネットワーク結合係数の初期値に依存してくるので、通常は、ランダムに初期値をとって学習を繰り返すことにより、誤差の収束状況を調べる必要がある。ここでは、結合係数の初期値のみ異なったニューラルネットワーク 4 種類を用いてシミュレーションを行なったが、どの場合も前節で述べたものと同じような結果が得られた。

また、ニューラルネットワークの中間層セルの数も一つの未定値である。この数が少ないと、学習が十分でない状況が生じ、また、多すぎると冗長となり計算量も増加する。ここでのシミュレーションでは、40 個のセルを設定したが、シミュレーションの結果 10 セル以上なら良好な制御結果が得られたが、5 セルになると振子が $x = 0$ の近くで振動し続け静止させることができなかった。

また、シミュレーションでは諸定数、(6) 式の ζ 、 ω_n 、(9) 式の K_L および (11) 式の F_c は、試行的に決定した。経験的に K_L の値は大きいほど学習が速く振子は数回の試行で倒れないようになるが、大きくとりすぎると振子が原点付近で振動的になりやすい。逆に小さすぎると学習が遅い。この値は、何度かの試行により最適なものを選ぶ必要がある。また、 F_c の値は、望ましい値（たとえば、Fig. 7 における F の最大値）に対して大きすぎると、試行学習開始後、振子が大きく振動し短時間で倒れてしまうことが予想される。よって、1 試行当たりの学習データが非常に少なくなるため、ここでは小さい値を用いた。

4.2.3 学習における入力データの取り方

学習は、試行中と試行後の 2 度行なったが、試行後の学習は、人間の場合でいうと復習に当たるものである。この際、試行中のデータをランダムな順序で並べ替えて学習を行なっているが、この並べ替えを行わずにデータを学習すると、サンプリング時間前後での入力パターンの各変数の値がほぼ等しく、

$$\begin{aligned} & (x(n), \dot{x}(n), \theta(n), \dot{\theta}(n)) \\ & \approx (x(n+1), \dot{x}(n+1), \theta(n+1), \dot{\theta}(n+1)) \end{aligned} \quad (11)$$

が成り立ち、このために時刻 $t = nT$ 付近での修正量 ΔF がほとんど同じとなり、似ているパターンでの学習が集約的に行なわれることになる。この結果ニューラルネットワーク上の知識の一般化が十分でない。また、別の時刻でも同様に一部の知識による学習が行なわれるので、以前に学習した結果は徐々に忘れられていく。このように、部分ごとでの学習結果は良いが、全体としては一般性がなく悪い学習となってしまう、制御性能の悪化を招

く、よって、ニューラルネットワーク上の知識の一般化を行なうためには、時系列データをランダムに並べ変えて学習する必要がある。

4.2.4 計算量について

ここのシミュレーションでは、通常の逐次処理コンピュータを用いており、中間層の数が増えるほど処理時間が増える。また、本方法のようにサンプリング周期ごとに制御結果を評価し教師信号を生成する場合、この部分の計算に要する時間も無視できない。実際、ワークステーション (CPU 68020) を用いたシミュレーションでは、(1), (2) 式の非線形微分方程式の求解も含めて、中間層 40 個の場合、1 試行 (実時間で 10 [sec]) に約 60 [sec] の計算時間がかかっている。よって、現段階では、中間層の数は必要最小限にするとともに、計算機プログラムの効率化も必要となろう。

5. 結論

本論文では多層型ニューラルネットワークをコントローラとして用いる学習制御系を倒立振子の安定化制御に適用し、そのシミュレーション例を示した。シミュレーションの結果、コントローラは試行を重ねて学習的に倒立振子を安定に立てながら台車を原点に停止させる制御則をフィードバック系として獲得できることを確認した。

ここで提案した方法は、学習制御系の構成に、先験的知識を用いて、仮想目標値発生部と評価部を設計することにより、ニューラルネットワークの学習のための一時的な教師信号をうまく生成するものである。これにより、非線形系などの制御対象に関する知識が少ない場合にも学習制御系の構成を可能とするものである。

また、ここで提案した方法と類似の方法により実際の倒立振子の制御に関して、田中ら¹⁸⁾により多層型ネットワークを用いて振子の角度のみの制御を行なった例が報告されている。この例では、演算時間を短くするために中間層セルを 3 個としており、また、学習を速めるために実際の倒立振子での実験の前に計算機上でのシミュレーションで学習を行なっている。

参考文献

- 1) 池田, 北村: 多層型ネットワークを用いた学習制御系; 第 33 回システム制御情報研究発表講演会, pp. 289~290 (1989)
- 2) 森, 西原, 古田: 倒立振子制御用ハイブリッド制御系; 計測自動制御学会論文誌, 12 巻, 4 号, pp. 116~121 (1976)
- 3) たとえば, 石田, 伊波, 崎間, 長堂: パーソナルコンピュータによる倒立振子の安定化制御; 琉球大学紀要, Vol. 32, pp. 101~110 (1986)
- 4) たとえば, 呉松, 北村, 中井: 二足歩行ロボットの軌道計画—倒立振子とニューラルネットワークの組み合わせ—; システム制御情報学会論文誌, 2 巻, 4 号, pp. 118~127 (1989)

を参照

- 5) 市川, 適応プロダクションシステムによる汎用制御系; 計測自動制御学会論文誌, 19 巻, 1 号, pp. 88~90 (1983)
- 6) 寺野, 増井, 菅谷: 不安定系のファジィ制御; 第 4 回ファジィ・システム・シンポジウム, pp. 139~144 (1987)
- 7) 市橋, 田中: PID と FUZZY のハイブリッド形コントローラ; 第 4 回ファジィ・システム・シンポジウム, pp. 97~102 (1988)
- 8) 荒木: ファジィ制御装置の数理モデルと倒立振子の制御; システム制御情報学会信号処理とディジタル制御研究分科会資料, 14 巻, pp. 1~8 (1989)
- 9) 川路, 野口: ファジィ推論による倒立振子の安定化制御; 第 5 回ファジィ・システム・シンポジウム, pp. 107~113 (1989)
- 10) B. Widrow: The Original Adaptive Broom Balancer; IEEE Conference on Circuits and Systems, pp. 351~357 (1987)
- 11) A. G. Barto, R. S. Sutton, & C. W. Anderson: Neuron-like Adaptive Elements That Can Solve Difficult Learning Control Problems; IEEE Transaction on System, Man and Cybernetics, Vol. SMC-13, pp. 834~846 (1983)
- 12) C. W. Anderson: Learning to Control an Inverted Pendulum Using Neural Networks; IEEE Control Systems Magazine, Vol. 9, No. 3, pp. 31~37 (1989)
- 13) V. V. Tolat & B. Widrow: An Adaptive "Broom Balancer" with Visual Inputs; IEEE International Conference on Neural Networks, II, pp. 641~647 (1988)
- 14) A. Guez & J. Selinsky: A Neuromorphic Controller with a Human Teacher; IEEE International Conference on Neural Networks, II, pp. 595~602 (1987)
- 15) 池田, 北村: ニューラルネットワークによる倒立振子の安定化学習制御; 第 31 回自動制御連合講演会論文誌, pp. 257~258 (1988)
- 16) 古田, 川路, 美多, 原: メカニカルシステム制御, オーム社, pp. 197~199 (1984)
- 17) D. E. Rumelhart, J. L. McClelland & the PDP Research Group: Parallel Distributed Processing, Vol. I, MIT Press (1986)
- 18) 田中, 小坪, 矢加部: ニューロコントローラによる倒立振子制御; 第 32 回自動制御連合講演会, pp. 301~302 (1989)