大域学習と局所学習を併用した B スプラインを用いた 学習制御法とその一自由度弾性アームの軌道制御への適用*

尾崎 弘明^{*1}, William A. GRUVER^{*2}, 久保 壮^{*3}
下川 哲司^{*1}, 林 長 軍^{*1}

Learning Control Using B-Spline Reference Input Curves by Combining Global and Local Learning Processes and Its Application to the Tracking Control of a One-Link Flexible Arm

Hiroaki OZAKI^{*4}, William A. GRUVER, Tsuyosi KUBO,
Tetsuji SHIMOOGAWA and Chang-jun LIN

^{*4} Fukuoka University, Dept. of Mechanical Engineering,
Nanakuma 8-19-1, Jonan-ku, Fukuoka, 814-0180 Japan

A learning control method, which generates feedforward inputs described by B-spline curves, is proposed. The algorithm is composed of two parts: Local Learning (LL) and Global Learning (GL). LL iteratively improves a feedforward input to approach the system output to a specified desired trajectory. GL identifies the controlled system dynamics by trial movements. The method has following features: (1) Describing inputs as B-spline curves has two merits: Input trajectories obtained by LL need relatively less memories to be stored and they are useful for the feedback control because of their differentiability. (2) LL is expressed by a simple iterative rule. (3) The gain matrix of the iterative rule is determined by GL. (4) GL is useful in the meaning that the identification process does not need any knowledge about the number of parameters and the order of dynamics. (5) The inverse dynamics obtained by GL generates the input trajectory which is a good initial trial for LL. The proposed method is effectively applied to the tracking control for a one-link flexible arm.

Key Words: Mechatronics and Robotics, Learning Control, Computer Control, B-Spline, Global Learning, Local Learning, Reference Input, Flexible Arm, Tracking Control

1. 緒 言

ロボットなどのメカトロニクスのもーションコントロールでは、安定化、低感度化、外乱抑制を目的とするフィードバック制御に加えて、追従性を高めるためにフィードフォワード制御を行うことが効果的であることが知られている。対象システムのパラメータが既知であるか精度よく同定できる場合には、これらの値を用いてフィードフォワード入力を算出する方法を用いることができる。一方、何度でも試行動作ができるという特長をもつシステムでは、その入力(フィードフォワード入力)を反復的に修正改善して目標軌道への追従性を高めるという方法が提案されている。すなわち、一つは、同一の初期値から出発して指定された時間の動作を繰返し、偏差が試行の反復ごとに減少するように入力を修正する、繰返し形学習制御法⁽¹⁾⁻⁽³⁾

である。もう一つは、試行の途中で停止せずに、目標軌道を周期的に与えることを考慮して制御系を設計する、繰返し制御⁽⁴⁾⁻⁽⁶⁾と呼ばれるものである。二つの方法では、修正改善後のフィードフォワード入力(繰返し制御では、繰返しコントローラ最終段の値)は、サンプリング周期ごとの時系列となる。

著者らは、評価関数を最適化することで目標軌道を生成することを考え、B スプラインで記述された目標軌道を生成する方法について検討してきた⁽⁷⁾⁻⁽¹⁰⁾。一般に、安定化された(あるいは安定な)制御システムでは、そのローパス特性のために、制御系内の高周波成分は減衰する。また、ある型の二自由度制御系では目標値フィルタによってフィードフォワード入力を生成していることと関連づけて、著者らは、入出力をユニフォーム B スプラインのような滑らかな時間関数で近似することを考えている。そこで本論では、フィードフォワード入力を B スプラインで記述するという方針のもとに、繰返し形学習法の考えによる学習法を提案する。B スプラインで入力軌道が記述されることの長所は、軌道が比較的少数の制御点によって表現さ

* 原稿受付 1998 年 9 月 17 日。

^{*1} 正員, 福岡大学工学部(〒814-0180 福岡市城南区七隈)。

^{*2} Simon Fraser University (B. C. V5A 1S6, CANADA)。

^{*3} 学生員, 福岡大学大学院。

E-mail: ozaki@fukuoka-u.ac.jp

れることから、これを記憶するのに必要なメモリを節約でき、その結果、入力軌道をパターンとして保持することが容易であることにある。また、軌道は連続関数として表現されるため、動作時間内の任意の時刻でその微分値を利用できることも利点である。

提案する方法は、与えられた一つの目標軌道に対する繰返し学習によって軌道追従性を改善する部分(局所学習と呼ぶ)と制御対象のダイナミクスを推定する部分(大域学習と呼ぶ)とにより構成される。大域的学習では制御対象の次数や内部パラメータを意識する必要がないこと、推定されたダイナミクスを局所学習のゲイン決定に直接利用できることによって、この方法は、極めて利用しやすい構成となっている。以下、局所学習と大域学習の構成について述べ、最後に、一自由度弾性アームの軌道制御実験に適用してその有効性を示す。なお、この問題に関して著者らは、B スプラインを用いて目標軌道を生成することが有効であることを示した⁽¹¹⁾が、その生成には対象のダイナミクスの正確な見積りが必要であった。ここで提案する方法によれば、局所学習の場合には近似のダイナミクスを知ればよいこと、さらに大域学習を併用する場合には、それさえ必要がないことを明らかにする。

2. B スプラインを用いた学習制御法

提案する学習制御法は、繰返し動作可能な安定なシステム(以下では、これをあらためて制御対象と呼ぶ)のフィードフォワード入力を求めるものであり、制御対象は定係数線形なダイナミクスで近似的に表現できるものとする。例えば、減速比の大きな減速器を用いる産業用ロボットを各関節ごとにフィードバック制御すると非干渉でこのようなダイナミクスに近づくことが知られており、この仮定は非現実的であるとはいえない。ただし、ここではその次数は未知であってもよい。また、本学習制御法においては、制御対象の入出力とともに B スプラインで記述できるものとする。

2・1 局所学習 動作の始端時刻を t_0 、終端時刻を t_f としてその動作時間 $[t_0, t_f]$ 内の任意の時刻 t において、制御対象の出力を $\theta_c(t)$ 、その目標値(目標軌道)を $\theta_{cd}(t)$ 、 $\theta_{cd}(t)$ に対する入力を $\theta_{rd}(t)$ 、 $\theta_c(t)$ に対する入力を $\theta_r(t)$ と表す。このとき制御対象のダイナミクスは、次式の定係数線形な微分方程式で近似されるものとする。

$$\sum_{i=0}^e \alpha_i \frac{d^i}{dt^i} \theta_c(t) = \sum_{j=0}^f \beta_j \frac{d^j}{dt^j} \theta_r(t) \quad \cdots \cdots (1)$$

ここに、 α_i, β_j は定数、 e および f は未知の次数とする。いま、動作時間 $[t_0, t_f]$ を N 等分して、各小区間

$[t_i, t_{i+1}]$ ($i=0, 1, \dots, N$) において、例えば $\theta_r(t)$ を n 次ユニフォーム B スプライン(n は奇数とする)で表すと、次式のようになる⁽⁷⁾(B スプラインの詳細については、例えば文献(12)を参照)。

$$\theta_r\{t(s)\} = \sum_{i=0}^n N_i(s) \hat{\theta}_{r, -\frac{n-1}{2}+i}, \quad \cdots \cdots (2)$$

$$t(s) = t_i + \Delta t \cdot s \quad 0 \leq s \leq 1$$

ここに、 $N_i(s)$ は B スプラインの基底関数、 $\Delta t = (t_N - t_0)/N$ (ただし、 $t_N = t_f$)、 $\hat{\theta}_{r, -(n-1)/2+i}$ は B スプラインの制御点である。したがって、入力変数 $\theta_r(t)$ は一組の制御点で表現できることがわかる。他の変数も同様に表現できるので、次のように制御点ベクトルを定義する。

$$\begin{aligned} \hat{\theta}_c &= (\hat{\theta}_{c, -\frac{n-1}{2}}, \hat{\theta}_{c, -\frac{n-1}{2}+1}, \dots, \hat{\theta}_{c, N+\frac{n-1}{2}})^T, \\ \hat{\theta}_{cd} &= (\hat{\theta}_{cd, -\frac{n-1}{2}}, \hat{\theta}_{cd, -\frac{n-1}{2}+1}, \dots, \hat{\theta}_{cd, N+\frac{n-1}{2}})^T, \\ \hat{\theta}_r &= (\hat{\theta}_{r, -\frac{n-1}{2}}, \hat{\theta}_{r, -\frac{n-1}{2}+1}, \dots, \hat{\theta}_{r, N+\frac{n-1}{2}})^T, \\ \hat{\theta}_{rd} &= (\hat{\theta}_{rd, -\frac{n-1}{2}}, \hat{\theta}_{rd, -\frac{n-1}{2}+1}, \dots, \hat{\theta}_{rd, N+\frac{n-1}{2}})^T \end{aligned} \quad \cdots \cdots (3)$$

ここに T は転置を表す。

式(2)を式(1)に代入すると、第 i 区間 $[t_i, t_{i+1}]$ の $s = s_h$ ($h=1, 2, \dots, H$) におけるダイナミクスは次の関係で表せる。

$$\begin{aligned} & \frac{1}{(\Delta t)^e} \sum_{i=0}^n \sum_{l=0}^e \alpha_i (\Delta t)^e N_i^{(l)}(s_h) \hat{\theta}_{c, i-\frac{n-1}{2}+l} \\ &= \frac{1}{(\Delta t)^f} \sum_{i=0}^n \sum_{j=0}^f \beta_j (\Delta t)^f N_i^{(j)}(s_h) \hat{\theta}_{r, i-\frac{n-1}{2}+j} \end{aligned} \quad \cdots \cdots (4)$$

ここに、

$$N_i^{(l)}(s_h) = d^l N_i(s) / ds^l |_{s=s_h}.$$

式(4)は、式(3)で定義された制御点ベクトルを用いると次式で表現できる。

$$A \hat{\theta}_c = B \hat{\theta}_r \quad \cdots \cdots (5)$$

ここに、 A と B は $(N \times H + 1) \times (N + n)$ の定数行列である。目標軌道 $\hat{\theta}_{cd}$ とそれに対応する入力 $\hat{\theta}_{rd}$ の間にも同様な次式の関係が成立する。

$$A \hat{\theta}_{cd} = B \hat{\theta}_{rd} \quad \cdots \cdots (6)$$

ただし、以下では第 k 回めの試行の入力を $\hat{\theta}^k$ と表し、同様な意味で $\hat{\theta}_c^k$ を用いる。

第 k 回めの試行において動作時間内で出力を m 箇所測定し、それらの測定点における目標軌道との差を m 次元誤差ベクトル e^k で表すと次式となる。

$$e^k = Q(\hat{\theta}_{cd} - \hat{\theta}_c^k) \quad \cdots \cdots (7)$$

ここに、 Q は $m \times (N + n)$ の定数行列である。 $Q \hat{\theta}_c^k$ が出力の m 点の測定値を表すベクトル(測定値ベクトル)となることから、 Q を測定行列と呼ぶ。また、次の偏差ベクトル ϵ^k を定義する。

$$\epsilon^k = A(\hat{\theta}_{cd} - \hat{\theta}_c^k) \quad \dots\dots\dots (8)$$

そこで、次の学習ゲイン行列 Γ および G を用いて $k+1$ 回目の試行入力を求めるものとする。

$$\hat{\theta}_r^{k+1} = \hat{\theta}_r^k + \Gamma e^k \quad \dots\dots\dots (9)$$

および

$$\hat{\theta}_r^{k+1} = \hat{\theta}_r^k + G \epsilon^k \quad \dots\dots\dots (10)$$

式(5)～(9)を用いると次の関係を得ることができる。

$$\begin{aligned} \epsilon^{k+1} &= A(\hat{\theta}_{cd} - \hat{\theta}_c^{k+1}) \\ &= B(\hat{\theta}_{rd} - \hat{\theta}_r^{k+1}) \\ &= B\{\hat{\theta}_{rd} - (\hat{\theta}_r^k + G \epsilon^k)\} \\ &= (I - BG) \epsilon^k \quad \dots\dots\dots (11) \end{aligned}$$

行列 A および B がともに列フルランクであれば、式(5)の両辺の誤差ノルムを最小にする意味で、 $\hat{\theta}_c^{k+1}$ は $\hat{\theta}_r^{k+1}$ と一意に対応し、 $\epsilon^{k+1} = 0$ のとき、 $\hat{\theta}_c^{k+1} = \hat{\theta}_{cd}$ および $\hat{\theta}_r^{k+1} = \hat{\theta}_{rd}$ が成り立つ。行列 A が列フルランクであるとして式(11)の両辺に左から $A^+ [(*)^+]$ は $*$ の疑似逆行列を表す] を乗ずると、次式を得る。

$$e^{k+1} = QA^+(I - BG) \epsilon^k \quad \dots\dots\dots (12)$$

そこで、 G を ΓQA^+ に等しくすると、第 $k+1$ 回目の試行の誤差ベクトルは次式となる。

$$e^{k+1} = (I - QA^+ B \Gamma) e^k \quad \dots\dots\dots (13)$$

したがって、 $I - QA^+ B \Gamma$ の最大特異値が 1 より小さければ、試行回数 k の値が増すにつれて、 e^k のユークリッドノルムは指数関数的に零に収束することがわかる。学習ゲイン行列 Γ は行列 A および B のおよその値が見積もれれば、これらを用いて次のように定めればよい。

$$\Gamma = (QA^+ B)^+ \quad \dots\dots\dots (14)$$

いま、 $p (=N \times H + 1)$ は式(5)の行の数を表す指定点数とし、 $q (=N + n)$ は制御点ベクトルの次元を表すものとする。 $I - QA^+ B \Gamma$ の最大特異値を 1 以下にできるためには、行列 Γ の大きさは $m \leq q$ の関係を満たさなければならない。さらに、 A と B が列フルランクであることも考慮すると、局所学習が収束するためには次の関係が成立しなければならないことがわかる。

$$m \leq q \leq p \quad \dots\dots\dots (15)$$

2.2 大域学習

局所学習では、与えられた目標軌道に対する学習の収束は得られるが、目標軌道が異なるたびに試行を繰返さなければならない。有限回の試行で制御対象の動特性が推定できれば、新たに与えられた目標軌道に対して、局所学習の初期の試行から十分追従性のよい入力を推定できる。また、式(14)によって適切な学習ゲイン Γ も決定できる。

制御対象のダイナミクスをその入出力から推定する

方法として、マルコフパラメータを用いた方法⁽¹³⁾ やシステム同定法⁽¹⁴⁾ などがある。前者は、ここで取扱う定係数線形系に対して、LQG 離散形フィードバックコントローラを直接設計できるなど興味深い、フィードフォワード入力を生成して追従性を高めるという本論の考え方とは異なったものである。一方、後者の方法を用いると、式(1)の係数 a_i, β_j を推定することができ、行列 A, B その結果として学習ゲイン Γ を算出することができる。しかし、式(1)の次数、あるいは、その係数の個数をどのように決定すればよいかという問題がある。以下では、式(14)の行列 $QA^+ B$ がダイナミクスの一つの表現であることに着目する。これを推定することで次数の問題が避けられ、学習ゲインも直接算出されることを以下に示す。

大域学習において、 j 番めの入力 $\theta_{rj}(t)$ を表す B スプラインの制御点ベクトル $\hat{\theta}_{rj}$ と、それに対応する出力 $\theta_{cj}(t)$ の m 個の測定点からなる測定値ベクトル θ_{cj} との関係は次のように表せるものとする。

$$\theta_{cj} = D \hat{\theta}_{rj} \quad \dots\dots\dots (16)$$

ここに、 D は制御対象のダイナミクスを表す $m \times q$ の定数行列である。そこで、式(16)の D を推定するために、 z 回の異なる試行を行うと、それぞれの $\hat{\theta}_{rj}$ に対して対応する測定値ベクトル θ_{cj} が得られるので、これらの試行をまとめると次式が成立する。

$$[\theta_{c1} \ \theta_{c2} \ \dots \ \theta_{cz}] = D[\hat{\theta}_{r1} \ \hat{\theta}_{r2} \ \dots \ \hat{\theta}_{rz}] \quad \dots\dots\dots (17)$$

上式を次のように表しておく。

$$\Theta_c = D \hat{\Theta}_r \quad \dots\dots\dots (18)$$

試行回数 z を増すことにより、 $\text{rank} \hat{\Theta}_r = q$ とすることができれば、 D は、最小二乗近似解の意味で、 $\hat{\Theta}_r$ の疑似逆行列 $\hat{\Theta}_r^+$ を用いて次のように推定される。

$$\hat{D} = \Theta_c \hat{\Theta}_r^+ \quad \dots\dots\dots (19)$$

新たに与えられた目標軌道 θ_{cd*} のための局所学習の初期試行入力は次式で求められる。

$$\hat{\theta}_{r*}^0 = \hat{D}^+ \theta_{cd*} \quad \dots\dots\dots (20)$$

また、局所学習の学習ゲイン Γ として $\hat{D}$ の疑似逆行列 $\hat{D}^+$ を用いればよいので、次の局所学習則を得る。

$$\hat{\theta}_r^{k+1} = \hat{\theta}_r^k + \hat{D}^+ e^k \quad \dots\dots\dots (21)$$

以上により、局所学習と大域学習とを組合せた全体の学習制御法の構成は、図 1 のようになる。図中の外側のループが式(7)および(9)による局所学習部を示す。 z^{-1} は局所学習の 1 回の試行遅れを意味している。また、中央部の破線で囲んだ部分が大域学習部を表わしている。なお、大域学習を用いる場合には、式(15)中の $q \leq p$ の条件は不要となる。

推定ダイナミクス $\hat{D}$ を式(19)に基づいて直接計算しようとする、試行結果をすべて保持しておく必要

Fig. 1 Structure of learning control combining global and local learning processes

があるため、大容量のメモリが必要となる。そこで、 $\hat{D}$ を以下のアルゴリズムによって逐次的に更新することにすれば、メモリを節約することができる。まず、式(17)の転置をとると次式ようになる。

$$\begin{bmatrix} \theta_{c1}^T \\ \theta_{c2}^T \\ \vdots \\ \theta_{cz}^T \end{bmatrix} = \begin{bmatrix} \hat{\theta}_{r1}^T \\ \hat{\theta}_{r2}^T \\ \vdots \\ \hat{\theta}_{rz}^T \end{bmatrix} [d_{T1} \ d_{T2} \ \cdots \ d_{Tm}] \cdots \cdots (22)$$

ここに、 d_{Tj} は $\hat{D}$ の第 j 行の転置を表す q 次元ベクトルである。ここで、式(22)の両辺の第 j 列に着目すると、次式が得られる。

$$\{\theta_{cj}^T\} = \hat{\theta}_{rj}^T d_{Tj} \cdots \cdots (23)$$

ただし、 $\{*\}_j$ は行列 $*$ の第 j 列を取出すことを意味している。このとき d_{Tj} は以下の手順に従って逐次更新できる。

Step 1 初期値 $d_{Tj}^0 = 0$, $P_j^0 = \alpha I_q$ とする。このときの試行は 1 回目 ($z=1$) である。ただし、 I_q は $(q \times q)$ 次元の単位行列であり、 α は十分に大きい正数とする。

Step 2

$$c_j^z = P_j^{z-1} \hat{\theta}_{rz} / (1 + \hat{\theta}_{rz}^T P_j^{z-1} \hat{\theta}_{rz}) \cdots \cdots (24)$$

を計算する。ただし、添字 z は、その値が z 回目の更新値であることを意味しているものとする。

Step 3

$$d_{Tj}^z = (I_q - c_j^z \hat{\theta}_{rz}^T) d_{Tj}^{z-1} + c_j^z [\theta_{cz}^T]_j \cdots \cdots (25)$$

を計算する。ただし、 $[*]_j$ は行列ベクトル $*$ の第 j 要素の値を意味している。

Step 4

$$P_j^z = P_j^{z-1} - P_j^{z-1} \hat{\theta}_{rz} \hat{\theta}_{rz}^T P_j^{z-1} / (1 + \hat{\theta}_{rz}^T P_j^{z-1} \hat{\theta}_{rz}) \cdots \cdots (26)$$

を計算する。

Step 5 すべての $j=1, \dots, m$ について (Step 1)

Fig. 2 Appearance of flexible arm control system

～(Step 4)を行ったのち、大域学習の試行回数に達するまで $z=z+1$ として (Step 2) へ戻る。

以上の手順によって d_{Tj} を逐次更新することで、すべての試行結果を保持しておく必要はなくなり、メモリを節約することができる。

3. 一自由度弾性アームの軌道制御への適用実験

図2に示すダイレクトドライブモータ (DDM) で駆動される一自由度弾性アームの軌道制御に、2章の学習制御アルゴリズムを適用する。アーム基部が固定されている DDM をサーボドライバ指令入力電圧 u で駆動したときの回転角およびその角速度を θ および $\dot{\theta}$ 、アーム先端のたわみおよびその速度を δ および $\dot{\delta}$ とする。 θ , $\dot{\theta}$ は DDM のエンコーダおよび FV 変換器の出力によって検出したものをそれぞれカウンタおよび A/D 変換器を通じてホストコンピュータ (NECPC-9821 V 20) に入力している。 δ はアーム先端に装着した発光ダイオードと DDM 上部に取付けた

光学センサとを用いて検出(浜松ホトニクス C 2399-00)し, A/D 変換器により同様に入力している。

まず, 軌道制御に先だって制振制御系を構成するために, 質量ばねモデルの考えに基づいて最適レギュレータを構成する⁽¹⁵⁾。ここでは, アームのたわみ速度 $\dot{\delta}$ は最小次元オブザーバを用いて推定することにし, 状態変数を $x=[x_1 \ x_2 \ x_3 \ x_4]^T=[\theta \ \dot{\theta} \ \delta \ \dot{\delta}]^T$, また, たわみ δ はアーム長 l に対して十分小さいと考え, アーム先端の角度 θ_c を出力 y とすると, 近似モデルは次のようになる。

$$\dot{x}=A_s x+b_s u, y=c_s x \quad \cdots \cdots (27)$$

ここに,

$$A_s = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{C_\theta+k_v k_f}{J} & \frac{k l}{J} & \frac{C_a l}{J} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{(C_\theta+k_v k_f) l}{J} & -k\left(\frac{1}{m}+\frac{l^2}{J}\right) & -C_a\left(\frac{1}{m}+\frac{l^2}{J}\right) \end{bmatrix} \quad \cdots \cdots (28)$$

$$b_s = \begin{bmatrix} 0 \\ \frac{k_v}{J} \\ 0 \\ -\frac{k_v l}{J} \end{bmatrix}, c_s = [1 \ 0 \ 1/l \ 0] \quad \cdots \cdots (29)$$

であり, A_s, b_s および c_s 中の各パラメータの実測値を表 1 に示す。

評価関数の重み $Q=diag(15 \ 0.1 \ 300 \ 2), R=1.0$

Table 1 Parameters of flexible arm system

DDM	Torque constant	k_v	19.9	(Nm/v)
	Velocity feedback gain	k_f	0.399	(V/(rad/s))
	Damping coefficient	C_θ	1.0	(Nm/(rad/s))
	Moment of inertia	J	0.037	(kgm ²)
Arm	Length of arm	l	0.44	(m)
	Spring constant	k	25.0	(N/m)
	Mass of arm head	m	0.137	(kg)
	Damping coefficient	C_a	0.0094	(N/(m/s))

Fig. 3 Comparison of two step responses
(with and without suppression of vibration)

とし, フィードバックゲイン $k=[3.87 \ 0.471 \ -24.1 \ 0.101]$ を得た。このとき最適レギュレータは

$$u=-k(x-x_d). \quad \cdots \cdots (30)$$

ここに, $x_d=[\theta_r \ 0 \ 0 \ 0]^T$ であり, θ_r が制振制御系の入力となる。制振制御系のステップ応答を調べると, 図 3 のようであった。同図中にはアーム部を剛体とみなして制御したステップ応答も示してあり, 質量ばねの近似モデルに基づいた制振制御が効果的であったことを示す。なお, 本制御のサンプリング時間は 1 ms である。

上記の制振制御系に対して, 提案する学習制御法を適用する。まず, 局所学習の有効性を示すために, 近似モデルの A_s, b_s, c_s および k を用いて式(5)の A, B を計算し, 式(14)により学習ゲイン行列 Γ を算出した。この場合, A, B すなわち Γ は実際の弾性アームの高次モードを省略して計算されていることになる。この Γ を用いて局所学習を行った例を図 4 に示す。ここでは, 学習区間 2.7(s) ($t_0=0.2$ (s), $t_f=2.9$ (s)) を 27 分割 ($N=27$) し, $H=2$ として指定点数 $p=55$, 測定点数 $m=28$, B スプラインの次数 $n=5$ すなわち $q=32$ とし, アーム先端の目標軌道 $\theta_{cd}(t)$ を B

Fig. 4 Learning process by local learning

スプラインで適当に与えて実験を行った結果である。初期の入力軌道 $\theta^0(t)$ は、出力の目標軌道 $\theta_{cd}(t)$ と一致させて与えている。この初期試行では、フィードフォワード入力 $\theta^0(t)$ が適当でないため、目標軌道 $\theta_{cd}(t)$ と出力 $\theta^0(t)$ とのずれが大きいことがわかる。しかし、試行を重ねるにつれて出力は目標軌道に漸近し、30 回めの試行では両者の差は十分に小さくなる。図 5 は、この変化を誤差ベクトル e^k のノルムで表したものであり、式 (13) に従って指数関数的に減少するようすを示している。

図 6 は、同上の測定点数 m と制御点の次元 q を用いて、大域学習と局所学習とを併用した学習過程の実験例を示している。ここでは、まず 37 本 ($z=37$) の異なるモータ入力 $\theta_{rj}(t)$ (B スプライン制御点ベクトル $\hat{\theta}_{rj}$ により生成) を用いて 2・2 節の大域学習のアルゴリズムにより D の行列を推定している。これを用いると、図 4 と同じ目標軌道 $\theta_{cd}(t)$ に対し、軌道追従

性のよい出力 $\theta^0(t)$ を生ずる初期入力軌道 $\theta^0(t)$ が得られていることがわかる。また、学習ゲインとして $\hat{D}^+$ を用いることで、極めて速やかに出力が目標軌道に近づくことがわかる。ここで注目すべきことは、図 4, 5 の例では局所学習のための学習ゲイン行列 Γ の計算に制振制御系モデル (A_s, b_s, c_s および k) を必要としたのに対し、大域学習を併用する場合にはモデルを必要としていないことである。この点は、適用面での重要なメリットである。また大域学習では、実際の制振制御系の応答から学習ゲイン行列を直接算出できるので、モデルと実機との差の影響を受けにくいものと考えられ、局所学習の収束も速い、このことは、図 7 の誤差ベクトルのノルムの変化からも理解できる。図 8 は、異なる目標軌道を与えたときにも優れた初期軌道が得られることを示している。図 9 は、急速に変化する約 2 Hz 程度の振動成分を含む目標軌道に対する実験結果を示している。上段の応答は目標軌道 $\theta_{cd}(t)$ に一致させた入力 $\theta^0(t)$ を用いたものであり、出力軌道 $\theta^0(t)$ が目標軌道から大きくずれている。大域学習と局所学習とを併用した結果を中、下段に示している。この例では、局所学習を繰返しても、1 回めの試行と同程度以上には追従性は向上しなかった。ただし同図からわかるように、この場合のアーム先端のたわみ δ が最大約 9 cm 程度にもなっていることを考えれば、学習の有効性は明らかである。図 10 および 11 は制御点の次元数 $q=15$ 、測定点数 $m=11$ とした例を示している。測定点においては良好な学習結果を示すことがわかる。なお、以上の実験で用いた制振制御、

Fig. 5 Improvement of error vector norm $\|e^k\|_2$ in local learning

Fig. 6 Learning process by combining global and local learning when $N=27$

Fig. 7 Improvement of error vector norm $\|e^k\|_2$ in combined learning process when $N=27$

Fig. 8 Initial input trajectory obtained by inverse dynamics estimated in global learning when $N=27$

Fig. 9 Learning process for quickly changing desired trajectory when $N=27$

大域学習および局所学習のプログラムはすべて C 言語で作成している。

4. 結 言

本論では、繰返し動作の可能な安定なシステムに対して、その目標軌道への追従性を高める学習制御法を検討した。提案した制御法は、繰返し動作によって学習ゲインを見出す大域学習と与えられた目標軌道への追従性を逐次的に改善する局所学習とによって構成されており、以下の特長がある。

(1) 入力を B スプラインで表現するため、次の特長をもつ。すなわち、学習収束後の入力が B スプラインの制御ベクトルで表されているので、入力を目標軌道とともにパターンとして記憶しやすいこと。入力軌道は微分可能であるので、制御の面から扱いやすいこと。

(2) 局所学習の学習則が極めて簡明な表現で表せる。

(3) 大域学習により、局所学習の適切な学習ゲインが見積もれる。このゲインにより、局所学習は速やかに収束する。

(4) 大域学習では、ダイナミクスの次数、内部パラメータは未知でよいので利用しやすい。

(5) 大域学習によって得られた逆ダイナミクスに

Fig. 10 Learning process by combining global and local learning when $N=10$

Fig. 11 Improvement of error vector norm $\|e^k\|_2$ in combined learning process when $N=10$

よって、任意の目標軌道に対する入力軌道を見積もれるので、局所学習にとって優れた初期軌道が得られる。

なお 2・2 節で述べたように、大域学習の一部においてシステム同定法⁽¹⁴⁾を利用することも可能であるが、その場合は上記の(4)の特長が失われる。

提案した方法を、制振制御された一自由度の弾性アーム先端の軌道制御に適用しその有効性を確認した。とくに、大域学習では実システムの応答から直接学習ゲインを決定できること〔上記の(3)および(4)項〕と目標軌道の変更に対して有用であること〔上記の(5)〕が実験によって明らかになった。

今後の課題として以下の点が挙げられる。提案した学習法は、(1) B スプラインを用いてダイナミクスの入出力を記述できること(2) 制御対象が定係数線形なダイナミクスで近似できることを前提として導かれている。本論で適用した実験では十分な有効性が確認さ

れているが、より複雑な制御対象に対して提案する手法がどの程度有効であるかについては今後明らかにしていきたい。また、前提条件(1)については、B スプラインが十分記述能力を持つことが明らかになりつつある⁽¹⁶⁾が、条件(2)については今後これを緩和することを検討したいと考えている。

文 献

- (1) Arimoto, S., Kawamura, S. and Miyazaki, F., *J. Robotics Systems*, **1**-2(1984), 123-140.
- (2) 川村貞夫, 日本ロボット学会誌, **11**-4(1993), 509-514.
- (3) 南條義人・有本卓, 日本ロボット学会誌, **11**-7(1993), 1047-1055.
- (4) 井上恵・中野道雄・久保忠志・松本啓・馬場育, 電気学会論文集, C 100(1980), 234-240.
- (5) 美多勉・加藤英治・青木康史, 計測自動制御学会論文集, **21**-11(1985), 1223-1230.
- (6) 中野道雄・井上恵・山本裕・原辰次, 繰返し制御, (1989), 計測自動制御学会.
- (7) 尾崎弘明・山本元司・毛利彰, 計測自動制御学会論文集, **23**-11(1987), 1199-1205.
- (8) 尾崎弘明・丘華, 機論, **58**-556, C(1992), 3675-3680.
- (9) 尾崎弘明・丘華・林長軍, 日本ロボット学会誌, **14**-4(1996), 560-566.
- (10) 尾崎弘明・林長軍, 日本ロボット学会誌, **15**-1(1997), 139-144.
- (11) 池尻太一・尾崎弘明・丘華, ロボティクス・メカトロニクス講演会, (1993), 15-18.
- (12) 山口富士夫, コンピュータディスプレイによる形状処理工学[II], (1988), 69-89, 日刊工業新聞社.
- (13) Furuta, K. and Wongsaisuwan, M., *Automatica*, **31**-9(1995), 1317-1324.
- (14) 足立修一, MATLAB による制御のためのシステム同定, (1996), 東京電機大学出版局.
- (15) 吉川恒夫・村上弘記・細田耕, 日本ロボット学会誌, **9**-1(1991), 1-10.
- (16) 林長軍・尾崎弘明・下川哲司, 計測自動制御学会論文集, **35**-4(1999), 583-585.