

# 生体を模倣したロボットアームの制御

○増山樹、村松鋭一、渡部慶二、有我祐一、遠藤茂(山形大学)

Bio-Mimetic Control for Robot Manipulators

\*T.Masuyama, E.Muramatsu, K.Watanabe, Y.Ariga, S.Endo(Yamagata Univ.)

**Abstract**— As a mathematical model for human arm movements, minimum-jerk model has been presented by T. Flash and N. Hogan. For an architecture of brain motor control, feedback error learning has been proposed by M. Kawato and his group. In this paper, these models are applied to robot manipulator control. We present an algorithm for reaching movement of robot hand, and present an adaptive control law based on feedback error learning

**Key Words:** Robot manipulator, Feedback error learning, Adaptive control

## 1. はじめに

人間の腕のような滑らかな運動をロボットアームで実現するためには、人間の運動モデルを模倣した制御が有効である。本稿では、人間の手先を目標物に到達させる運動をロボットに行わせるための、人の運動モデルに基づく制御アルゴリズムの提案を行う。本稿で述べる提案手法は大きくわけて2つある。まず1つは、躍度最小モデル[1]に基づく目標関節角の生成アルゴリズムであり、目標手先位置が与えられたときに、瞬時に目標関節角と目標関節角速度を計算する方法である。もう1つはフィードバック誤差学習[2]に基づくロボットアームの制御である。小脳における学習を模倣し、ロボットの慣性や粘性に関するパラメータが未知であっても適応的にアームの逆モデルを獲得して、目標軌道に沿うように関節を制御する方法について述べる。

## 2. 目標軌道生成アルゴリズム

ある位置から目標位置まで手先を移動させる場合、ヒトはおおむねただ1つの軌道を選択する。それがどのような評価に基づいて決定されているかについてさまざまなモデルが提案されている[3]。本稿ではFlashとHoganによる「躍度最小モデル」に着目し、これに基づく目標関節角の軌道を瞬時に計算する計算アルゴリズムを提案する。

躍度最小モデル[1]においては

$$C_f = \frac{1}{2} \int_0^{t_f} [(d^3x/dt^3)^2 + (d^3y/dt^3)^2] dt \quad (1)$$

を最小にするように手先の軌道が決定される。これを満たす軌道は解析的に求められ、手先の軌道は

$$\begin{cases} x(t) = x_0 + (x_0 - x_f)(15\tau^4 - 6\tau^5 - 10\tau^3) \\ y(t) = y_0 + (y_0 - y_f)(15\tau^4 - 6\tau^5 - 10\tau^3) \end{cases} \quad (2)$$

で与えられる。ただし、 $x_0, y_0$  は運動の始点、 $x_f, y_f$

は終点を表す。2次元平面内において、手先の軌道は直線となり、 $x$ 方向と $y$ 方向の速度の、時間に対するグラフは釣鐘型になる。

手先の目標軌道は上式で計算できるが、後に述べる3リンクロボットアームの制御には3つの関節の目標角度と目標角速度が必要となる。したがって、目標手先軌道から目標関節角・角速度を算出するアルゴリズムが必要となる。ここでその算出アルゴリズムを提案する。

1) 始点と終点の座標が与えられると(2)式の5次関数と、それを微分した $(\dot{x}, \dot{y})$ の軌道を定める4次関数の係数が決定される。

2) 始点の時刻から終点の時刻を4等分した区間に分割する。

3) ある区間に注目し、その開始時間を $t_s$ 、終了時間を $t_e$ とする。区間両端の $x(t), y(t), \dot{x}(t), \dot{y}(t)$ の値

$$x(t_s), y(t_s), x(t_e), y(t_e)$$

$$\dot{x}(t_s), \dot{y}(t_s), \dot{x}(t_e), \dot{y}(t_e)$$

を上で述べた5次関数と4次関数から求める。

4) それらの値から逆運動学とヤコビ行列を用いて、区間両端において満たすべき目標関節角と目標関節角速度の値

$$q_i(t_s), q_i(t_e), \dot{q}_i(t_s), \dot{q}_i(t_e) \quad i = 1, 2, 3$$

を計算する。

5) こうして得られた値を境界条件として、目標関節角の軌道を決める。

$$q_i(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

という3次関数で近似するとして、4)で得られた4つの境界条件から、4つの係数 $a_0, a_1, a_2, a_3$ を定める。目標関節角速度の軌道は

$$\dot{q}_i(t) = a_1 + 2a_2 t + 3a_3 t^2$$

として決まる。

6) 上の方法でもとめたそれぞれの区間の $q(t)$ と

$\dot{q}(t)$ をつなぎ合わせて目標関節角と角速度の軌道とする。

例えば、 $(x_0, y_0) = (0.34, -0.08), (x_f, y_f) = (0.2,$

$-0.2)$ で $t_f = 1.5[\text{sec}]$ のとき、上で述べた目標関節角と目標関節角速度はつぎのようなグラフになる。

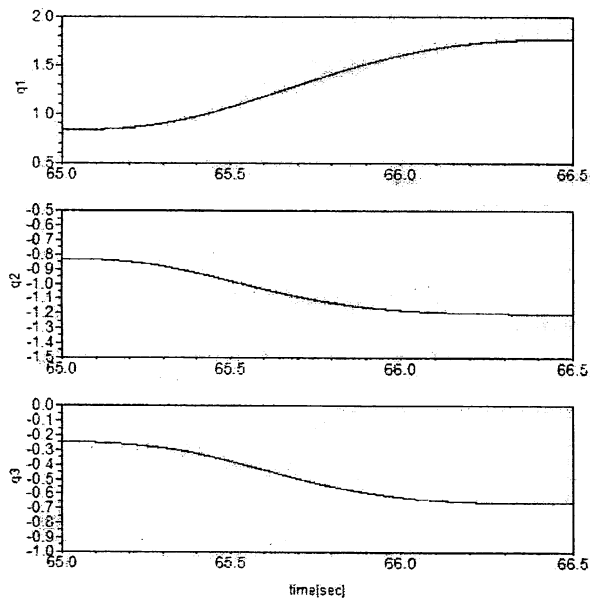


Fig.1 目標角度の変化

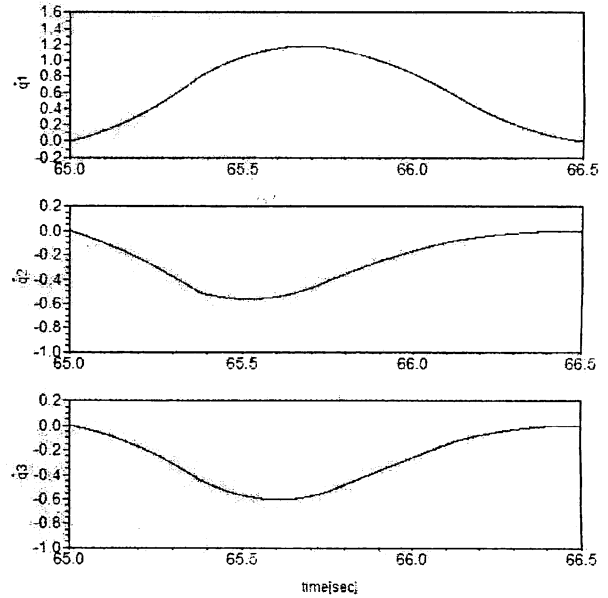


Fig.2 目標角速度の変化

関節がこのような軌道を描いたときの手先の軌道はつぎのようなグラフになり、 $x, y$ 平面で直線的、速度の時間的変化は釣鐘型であることが確認できる。

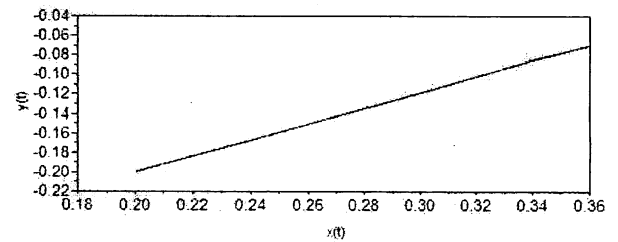


Fig.3  $x, y$ 平面における手先の軌道

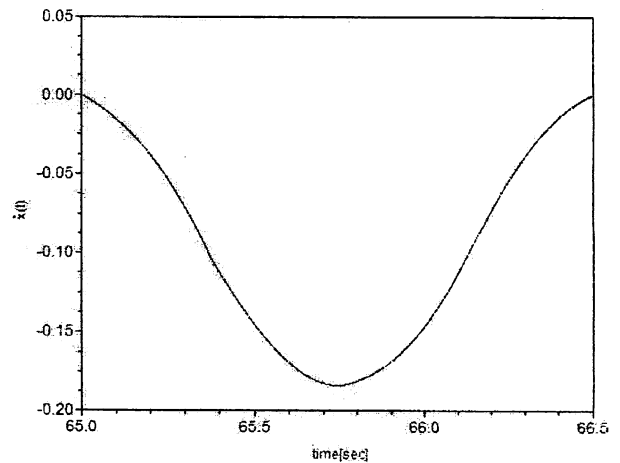


Fig.4  $x$ 軸方向の手先の速度

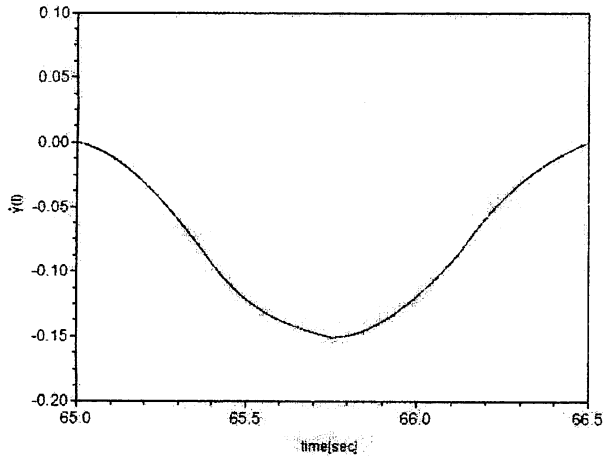


Fig.5 y 軸方向の手先の速度

### 3. フィードバック誤差学習によるロボットアームの制御

前節の方法で目標関節角と目標関節角速度の関数が与えられたとし、その軌道に沿うようなロボット関節を制御することを考える。目標軌道に一致させるアームの制御には逆モデルを用いたフィードフォワード制御が有効であるが、そのためにはロボットのパラメータが既知で逆力学モデルが得られていなければならない。しかし、実際のロボット制御においては、それらを正確に得るのが困難な場合が多い。

一方、人間が行っている滑らかで高制度な腕の運動制御においては、筋骨格系の逆モデルを学習によって得るフィードバック誤差学習が用いられているとされている[2],[3]。線形1入力1出力システムに対するフィードバック誤差学習に基づく制御理論としては文献[5],[6]が、線形多入力多出力システムに対する制御理論としては文献[7],[8]がある。本稿では、ロボットアームの適応制御理論[4]とフィードバック誤差学習を融合させた新しい制御則を提案する。

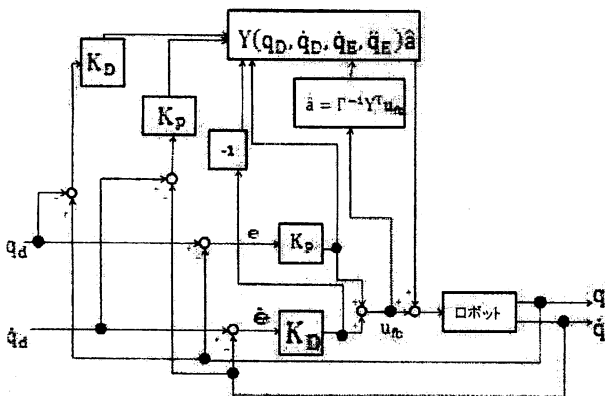


Fig.6 フィードバック誤差学習のブロック線図

上の図において、ロボットはつぎの運動方程式によって記述されるとする。

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} + D\dot{q} = Y(q_D, \dot{q}_D, \dot{q}_E, \ddot{q}_E)a = \tau \quad (3)$$

これに対して

$$Y(q_D, \dot{q}_D, \dot{q}_E, \ddot{q}_E)\hat{a} \quad (4)$$

の部分に逆モデルを表す。ただし、

$$\begin{aligned} q_D &= -K_D e \\ \dot{q}_E &= K_P e \end{aligned} \quad (5)$$

である。ただし、 $K_D, K_P$  は正定行列である。また、 $\hat{a}$  は可調整パラメータベクトルであり、

$$\dot{\hat{a}} = \Gamma^{-1} Y^T u_{fb} \quad (6)$$

によって適応的に調整する。ただし、 $\Gamma$  は正定行列である。

上記の適応則により、時間の経過とともに  $e(t) \rightarrow 0$  となり、目標に沿った関節軌道が得られる。

(証明の概略)

$$V = \frac{1}{2} [u_{fb}^T H(q_D) u_{fb} + \Delta a^T \tau \Delta a]$$

とする。ただし、

$$\Delta a = \hat{a} - a$$

である。 $V$  の微分を計算し、(3)~(6)式を利用すると、

$$\dot{V} = -u_{fb}^T u_{fb} \leq 0$$

となり、これと

$u_{fb} = K_P e + K_D \dot{e}$  より、 $e(t) \rightarrow 0$  が得られる。

### 4. シミュレーション

第2節の目標軌道生成法と第3節の制御則を用いたシミュレーションを示す。ロボットアームのモデルには当研究室の3リンクロボットアームの運動方程式を用いる。シミュレーションでは時刻0秒から30秒までは、(6)式による適応的なパラメータ調整の時間とし、アームの目標角を正弦波として運動させる。30秒経過したところで、目標手先位置を与えて、第2節の目標軌道の生成を行い、目標位置までの到達運動をさせる。到達までの時間は1.5秒とする。

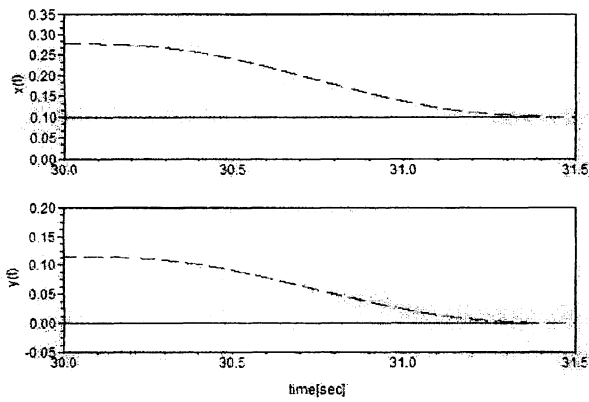


Fig.7 手先の軌道

実線は目標手先位置,破線は手先の軌道を表す. 図に示すように, 目標軌道に沿うようにロボットが運動していることが確認できる.

また, 各関節角は下の図のようになった.

実線: 目標軌道  
破線: 現実の軌道

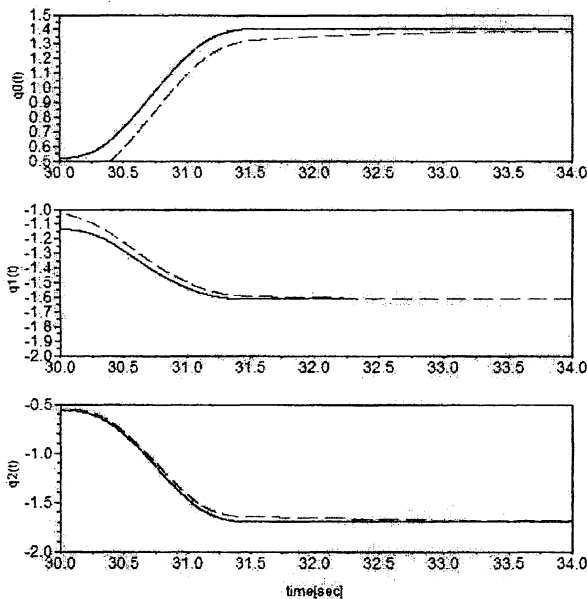


Fig.8 関節角の軌道

## 5. おわりに

人間の腕のような滑らかな運動をロボットアームで実現するため, 躍度最小モデルに基づく目標関節角の生成アルゴリズムと, フィードバック誤差学習に基づく制御法を提案した. ロボットのパラメータが未知であっても適応的な学習によって目標通りの軌道を描くことをシミュレーションによって確認した. 今回提案したフィードバック誤差学習に基づくロボットアームの適応制御法は, 文献[4]の適

応制御と比較した場合, 目標角加速度の信号が必要ないという利点がある.

## 参考文献

- [1] T. Flash and N. Hogan: The Coordination of Arm Movements: An Experimentally Confirmed Mathematical Model, The Journal of Neuroscience, Vol. 5, No. 7. pp.1688-1703, 1985.
- [2] M. Kawato, K. Furukawa, and R. Suzuki: A Hierarchical Neural-Network Model for Control and Learning of Voluntary Movement, Biological Cybernetics, Vol. 57, pp.169-185, 1987.
- [3] 川人光男: 脳の計算理論, 産業図書, 1996.
- [4] J. J. E. Slotine and W. Li: Adaptive Manipulator Control: A Case Study, IEEE Transactions on Automatic Control, Vol. 33, No. 11, 1988.
- [5] A. Miyamura and H. Kimura: Stability of Feedback Error Learning Scheme, Systems and Control Letters, Vol. 45, pp. 303-316, 2002.
- [6] E. Muramatsu and K. Watanabe: Feedback Error Learning Control without Recourse to Positive Realness, IEEE Transactions on Automatic Control, Vol. 49, No. 10, pp. 1762-1767, 2004.
- [7] B. Alali, K. Sugimoto, and K. Hirata: Generalization of Feedback Error Learning (FEL) to MIMO systems, Transactions of the Society of Instrument and Control Engineers, Vol. 43, No. 4, pp. 293-302, 2007.
- [8] B. Alali, K. Hirata, and K. Sugimoto: Feedback Error Learning for Writing One Stroke Characters by Two-Link Manipulator, システム制御情報学会論文誌, Vol. 20, No. 12, pp. 475-481, 2007.