剛体運動系のモデル獲得と制御*

(第4報, 基底関数ネットワークによる慣性行列と コリオリカベクトルのモデル化)

安 東 隆 志^{*1}, 岩 壺 卓 三^{*2}

Obtaining Models and Control for Rigid Body Motion Systems (4th Report, Modeling of Inertia Matrix and Coriolis's Force Vector by Basis Function Networks)

Takashi ANDO^{*3} and Takuzo IWATSUBO

^{*3} Hyogo Prefectural Institute of Industrial Research,
3-1-12 Yukihiro-cho, Suma-ku, Kobe-shi, Hyogo, 654-0037 Japan

This paper proposes a motion learning method for motion systems using basis function networks. Motion systems have unknown nonlinearities such as variation of inertia, Coriolis's force, nonlinear electromagnetic driving characteristics and gravity that are complicated to formulate. In this study, these nonlinearities are described by basis function networks. The updating rules of the weights in the network are derived by Popov's Hyper Stable Theory. However, there exist modeling errors and unmodeled forces which generate tracking errors. Therefore, H_∞ control is introduced to attenuate the influence of modeling errors. This paper shows an analytical formulation of basis function networks for motion systems and effectiveness of the proposed motion control method by demonstrating an experiment of tracking desired trajectory in 2 types of 2-degree-of-freedom link motion system that have different motion equations.

Key Words: Motion Control, Adaptive Control, Learning, Nonlinear System, Inverse Dynamics, H_∞ Control

1. は じ め に

一般に運動制御では運動方程式を解析的に導き出し, 同定実験によりパラメータを決定することによって制御則を決定する手法が主流である⁽¹⁾⁽²⁾. しかし, この方法では, リンク機構などのように運動系の自由度が増すと運動方程式に含まれる慣性行列とコリオリカベクトルを解析的に導くことが困難となる. また, 対象となる運動系が変わるごとに運動方程式を記述しなおし, 制御則を決定しなおさなければならない. 一方, 人間を含む生体は運動制御をおこなうとき, 学習によって運動モデルを獲得し, 対象となる運動系が変わっても適応することができる^{(3)~(5)}. 工学的にも同定実験を必要とせず, 制御系が自ら制御対象に適応またはオンライン同定できれば実用的である^{(6)~(9)}.

これまで著者らは運動モデルを制御系内に獲得し, 逆ダイナミクスを制御系内に構成することによって

フィードフォワード制御をおこなう手法を提案してきた^{(10)~(12)}. しかし, これまでの手法では, 運動学習則を導くためにあらかじめ解析的に運動方程式を記述しておき, 慣性行列とコリオリカベクトルの構造を確定しなければならなかった. そのために制御する運動系が変わるごとに運動学習則を変更しなければならなかった.

そこで, 本研究では駆動力, 重力, 粘性摩擦力と同様に慣性行列とコリオリカベクトルを基底関数ネットワークにより表現し, これらをモデルとして獲得するためのネットワーク結合重み学習則を提案する. この手法では, 運動系の幾何学的な構造を反映する基底関数ネットワークを予め構成しておき, それに対する学習則を導くことにより, 運動系に変更があっても運動学習則を変更する必要がない. 本研究では提案する手法が運動系の幾何学的な構造と自由度が同じであれば運動方程式に依存しないことを示すため, 2種類の運動系: (a) 回転軸が平行な2自由度リンク運動系, (b) 回転軸が直角なねじれの位置にある2自由度リンク

* 原稿受付 2000年12月4日.

^{*1} 正員, 兵庫県立工業技術センター(〒654-0037 神戸市須磨区行平町3-1-12).

^{*2} 正員, 神戸大学工学部(〒657-8501 神戸市灘区六甲台町1-1).

E-mail: ando@hyogo-kg.go.jp

運動系に適用する。この両者は運動方程式に含まれる慣性行列とコリオリ力ベクトルが異なり、これまで著者らが提案した手法では運動学習則を変更する必要があった。しかし、本研究で提案する手法では運動学習則を変更する必要がない。本論文では、まず二つの運動系における軌道追従実験で提案する手法の有効性を確認する。次に、提案する手法によって発生するモデル獲得誤差による軌道追従誤差を抑制するために H_∞ 制御を組み込むことを提案する。モデル獲得誤差を外乱、軌道追従誤差を出力とする線形系を制御対象として H_∞ 制御系を設計することにより、外乱抑制に効果があることを示す。

2. 運動系の定式化と逆動力学的系の導出

2.1 ラグランジュ方程式による運動系の定式化
まず、運動系のモデルを獲得するため運動方程式を記述し、獲得すべきモデルを設定する。 m 自由度の運動系の運動方程式は次式のラグランジュ方程式で与えられる。

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\mathbf{y}}} \right) - \frac{\partial L}{\partial \mathbf{y}} + \frac{\partial D}{\partial \dot{\mathbf{y}}} = \boldsymbol{\tau} \quad (1)$$

ここに、 L はラグランジュ関数、 D は減衰による損失エネルギーであり、それぞれ次式で与えられる。

$$L = \frac{1}{2} \dot{\mathbf{y}}^T \mathbf{J}(\mathbf{y}) \dot{\mathbf{y}} - U(\mathbf{y}) \quad (2)$$

$$D = \frac{1}{2} \dot{\mathbf{y}}^T \mathbf{D}(\mathbf{y}) \dot{\mathbf{y}} \quad (3)$$

また、 $\mathbf{y} = [y_1 \cdots y_m]^T$ は変位ベクトル、 $\boldsymbol{\tau} = [\tau_1 \cdots \tau_m]^T$ は駆動トルクベクトル、 $\mathbf{J}(\mathbf{y})$ は慣性行列であり、正定対称行列である。 $U(\mathbf{y})$ は重力による位置エネルギー、 $\mathbf{D}(\mathbf{y}) = \text{diag}\{d_1(y_1), \cdots, d_m(y_m)\}$ は粘性減衰係数行列である。駆動トルクは電磁アクチュエータによって発生するものとする、磁力は磁極間の距離に強く依存するので、次式のような非線形性を持つものとする。

$$\boldsymbol{\tau} = \mathbf{K}(\mathbf{y}) \mathbf{v} \quad (4)$$

ここに、 $\mathbf{v} = [v_1 \cdots v_m]^T$ は駆動電圧ベクトル、 $\mathbf{K}(\mathbf{y}) = \text{diag}\{k_1(y_1), \cdots, k_m(y_m)\}$ は駆動電圧から駆動力への変換係数行列であり、変位に依存する関数としている。慣性行列 $\mathbf{J}(\mathbf{y})$ 、減衰係数行列 $\mathbf{D}(\mathbf{y})$ 、位置エネルギー $U(\mathbf{y})$ を用いて次の運動方程式が得られる。

$$\mathbf{J}(\mathbf{y}) \ddot{\mathbf{y}} + \mathbf{c}(\mathbf{y}, \dot{\mathbf{y}}) + \mathbf{f}_G(\mathbf{y}) + \mathbf{D}(\mathbf{y}) \dot{\mathbf{y}} = \mathbf{K}(\mathbf{y}) \mathbf{v} \quad (5)$$

ここに、 $\mathbf{c}(\mathbf{y}, \dot{\mathbf{y}})$ はコリオリ力ベクトル、 $\mathbf{f}_G(\mathbf{y})$ は重力ベクトルであり、それぞれ次式で与えられる。

$$\mathbf{c}(\mathbf{y}, \dot{\mathbf{y}}) = \dot{\mathbf{J}}(\mathbf{y}) \dot{\mathbf{y}} - \frac{1}{2} \frac{\partial}{\partial \mathbf{y}} \left\{ \dot{\mathbf{y}}^T \mathbf{J}(\mathbf{y}) \dot{\mathbf{y}} \right\} \quad (6)$$

$$\mathbf{f}_G(\mathbf{y}) = \frac{\partial U(\mathbf{y})}{\partial \mathbf{y}} \quad (7)$$

運動系を安定化するために駆動電圧に対し、次式のフィードバック則を適用する。

$$\mathbf{v} = -\mathbf{K}_D \dot{\mathbf{y}} - \mathbf{K}_P (\mathbf{y} - \mathbf{u} - \mathbf{u}_B) \quad (8)$$

ここに、 $\mathbf{K}_D$ は微分ゲイン行列、 $\mathbf{K}_P$ は比例ゲイン行列、 $\mathbf{u}$ は目標軌道から生成される制御系への入力ベクトル、 $\mathbf{u}_B$ は重力やその他の外力を補償する入力ベクトルである。本研究では、運動モデルを獲得することにより式 (8) に含まれる $\mathbf{K}_D$ 、 $\mathbf{K}_P$ 、 $\mathbf{u}_B$ を調整し、運動系を線形化する手法を提案する。

2.2 $\mathbf{K}_D$ 、 $\mathbf{K}_P$ 、 $\mathbf{u}_B$ の調整による運動系の線形化と逆動力学的系の導出 非線形な駆動系を大域的に線形化するために次式の調整則によりゲイン行列 $\mathbf{K}_D$ 、 $\mathbf{K}_P$ を調整する。

$$\mathbf{K}_D = \hat{\mathbf{K}}(\mathbf{y})^{-1} \hat{\mathbf{J}}(\mathbf{y}) \mathbf{A} \quad (9)$$

$$\mathbf{K}_P = \hat{\mathbf{K}}(\mathbf{y})^{-1} \hat{\mathbf{J}}(\mathbf{y}) \mathbf{B} \quad (10)$$

ここに、 $\hat{\mathbf{J}}(\mathbf{y})$ は慣性行列 $\mathbf{J}(\mathbf{y})$ の推定モデル、 $\hat{\mathbf{K}}(\mathbf{y})$ は駆動特性行列 $\mathbf{K}(\mathbf{y})$ の推定モデル、 $\mathbf{A} = \text{diag}\{a_1, \cdots, a_m\}$ と $\mathbf{B} = \text{diag}\{b_1, \cdots, b_m\}$ は目標とする粘弾性行列であり、正定な対角行列とする。また、ベクトル $\mathbf{u}_B$ は重力とコリオリ力を補償する項であり、

$$\mathbf{u}_B = \mathbf{B}^{-1} \hat{\mathbf{J}}(\mathbf{y})^{-1} (\hat{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}}) + \hat{\mathbf{D}}(\mathbf{y}) \dot{\mathbf{y}} + \hat{\mathbf{f}}_G(\mathbf{y})) \quad (11)$$

ここに、 $\hat{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}})$ 、 $\hat{\mathbf{f}}_G(\mathbf{y})$ はそれぞれコリオリ力ベクトル $\mathbf{c}(\mathbf{y}, \dot{\mathbf{y}})$ 、重力ベクトル $\mathbf{f}_G(\mathbf{y})$ の推定モデルである。式 (8) に式 (9)~(11) を代入して整理すると次式となる。

$$\begin{aligned} \mathbf{v} = & \hat{\mathbf{K}}(\mathbf{y})^{-1} [\hat{\mathbf{J}}(\mathbf{y}) \{-\mathbf{A} \dot{\mathbf{y}} - \mathbf{B}(\mathbf{y} - \mathbf{u})\} \\ & + \hat{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}}) + \hat{\mathbf{f}}_G(\mathbf{y}) + \hat{\mathbf{D}}(\mathbf{y}) \dot{\mathbf{y}}] \end{aligned} \quad (12)$$

ここで、 $(\hat{\cdot}) = (\cdot)$ 、すなわち運動モデルが制御系内に獲得されれば、式 (5) は次式の線形系に帰着される。

$$\ddot{\mathbf{y}} + \mathbf{A} \dot{\mathbf{y}} + \mathbf{B}(\mathbf{y} - \mathbf{u}) = \mathbf{0} \quad (13)$$

そこで、入力 $\mathbf{u}$ に対して式 (13) の逆動力学を次式のように構成する。

$$\mathbf{u} = \mathbf{B}^{-1} (\ddot{\mathbf{y}}_M + \mathbf{A} \dot{\mathbf{y}}_M + \mathbf{B} \mathbf{y}_M) \quad (14)$$

ここに、 $\mathbf{y}_M$ は目標軌道である。式 (12) に式 (14) を代入して整理すると次式となる。

$$\begin{aligned} \mathbf{v} = & \hat{\mathbf{K}}(\mathbf{y})^{-1} \{ \hat{\mathbf{J}}(\mathbf{y}) (\ddot{\mathbf{e}} + \mathbf{A} \dot{\mathbf{e}} + \mathbf{B} \mathbf{e}) + \\ & \hat{\mathbf{J}}(\mathbf{y}) \ddot{\mathbf{y}} + \hat{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}}) + \hat{\mathbf{f}}_G(\mathbf{y}) + \hat{\mathbf{D}}(\mathbf{y}) \dot{\mathbf{y}} \} \end{aligned} \quad (15)$$

Fig. 1 Basis function network

ここに, $\mathbf{e} = \mathbf{y}_M - \mathbf{y}$ は軌道誤差である. 式 (5) と式 (15) を整理すると,

$$\ddot{\mathbf{e}} + \mathbf{A}\dot{\mathbf{e}} + \mathbf{B}\mathbf{e} = -\mathbf{r} \quad (16)$$

$$\mathbf{r} = \hat{\mathbf{J}}(\mathbf{y})^{-1} \{ -\tilde{\mathbf{K}}(\mathbf{y})\mathbf{v} + \tilde{\mathbf{J}}(\mathbf{y})\dot{\mathbf{y}} + \tilde{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}}) + \tilde{\mathbf{f}}_G(\mathbf{y}) + \tilde{\mathbf{D}}(\mathbf{y})\dot{\mathbf{y}} \} \quad (17)$$

となる. ここに, $(\tilde{\cdot}) = (\hat{\cdot}) - (\cdot)$ はモデル化誤差である. $\mathbf{A}$, $\mathbf{B}$ は対角行列なので, 誤差系 (16) は 1 入力 1 出力となる. 式 (17) はモデル化誤差 $(\tilde{\cdot})$ の総計を表している. そこで, ベクトル $\mathbf{r}$ をモデル獲得誤差と呼ぶ. 式 (16) はモデル獲得誤差 $\mathbf{r}$ が軌道追従誤差 $\mathbf{e}$ に与える影響, すなわち感度を示している. 制御系内に獲得されるモデル $(\hat{\cdot}) = (\cdot)$ となれば $\mathbf{e} = \mathbf{0}$, $\dot{\mathbf{e}} = \mathbf{0}$ となり, 式 (15) は式 (5) に一致する.

3. 運動学習系の導出

3.1 基底関数ネットワークによる非線形性のモデル化 ここでは, 基底関数ネットワークを用いて運動系に含まれる非線形性をモデル化する方法を示す. 図 1 に示すように, 本研究では基底関数 $\phi_k(y)$ ($k = 1 \cdots n$) からの出力に重み w_k をかけて結合するネットワークを基底関数ネットワークと呼ぶ⁽⁶⁾.

まず, 慣性行列 $\mathbf{J}(\mathbf{y})$ を基底関数ネットワークにより表現する. 2 自由度リンク運動系では, 慣性行列は以下の性質を持つ.

1. 2 自由度リンクの慣性モーメントの変化は第 2 関節角 y_2 に依存する.
2. 第 1 関節軸まわりの慣性モーメントは第 1 リンクの慣性モーメント J_1 と第 2 リンクの慣性モーメント J_2 の和 $J_1 + J_2$ を含む.
3. 第 2 関節軸まわりの慣性モーメントは第 2 リンクの慣性モーメント J_2 である.
4. 慣性行列 $\mathbf{J}(\mathbf{y})$ は正定対称行列である.

これらのことを考慮すると, 慣性行列のモデル $\hat{\mathbf{J}}(\mathbf{y})$ は次式で表現できる.

$$\hat{\mathbf{J}}(\mathbf{y}) = \begin{bmatrix} \hat{J}_1 + \hat{J}_2 + \hat{f}_{11}(y_2) & \hat{f}_{12}(y_2) \\ \hat{f}_{12}(y_2) & \hat{J}_2 \end{bmatrix} \quad (18)$$

次に, 基底関数 $\phi_k(y_2)$ を用いて各非線形性のモデル化をおこなう. まず, 重み $\hat{w}_k^{C_1}$, $\hat{w}_k^{C_2}$ を用いて慣性行列モデル $\hat{\mathbf{J}}(\mathbf{y})$ の変動成分 $\hat{f}_{11}(y_2)$, $\hat{f}_{12}(y_2)$ を次式で表現する.

$$\hat{f}_{11}(y_2) = \sum_{k=1}^{n_C} \hat{w}_k^{C_1} \phi_k(y_2) \quad (19)$$

$$\hat{f}_{12}(y_2) = \sum_{k=1}^{n_C} \hat{w}_k^{C_2} \phi_k(y_2) \quad (20)$$

ここに, n_C は慣性モデルにおける基底関数の個数である. 式 (19), (20) を用いると式 (6) によりコリオリ力モデル $\hat{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}})$ は次式となる.

$$\hat{\mathbf{c}}_1(\mathbf{y}, \dot{\mathbf{y}}) = \sum_{k=1}^{n_C} (\hat{w}_k^{C_1} \dot{y}_1 + \hat{w}_k^{C_2} \dot{y}_2) \dot{y}_2 \phi'_k(y_2) \quad (21)$$

$$\hat{\mathbf{c}}_2(\mathbf{y}, \dot{\mathbf{y}}) = \sum_{k=1}^{n_C} \left\{ -\frac{1}{2} \hat{w}_k^{C_1} \dot{y}_1^2 \phi'_k(y_2) \right\} \quad (22)$$

ここに,

$$\phi'_k(y_2) = \frac{d\phi(y_2)}{dy_2} \quad (23)$$

次に, 駆動力特性行列 $\mathbf{K}(\mathbf{y})$, 減衰係数行列 $\mathbf{D}(\mathbf{y})$ は関節軸の角度に依存するのでそれぞれのモデルを重み $\hat{w}_k^{K_i}$, $\hat{w}_k^{D_i}$ を用いて次式で与える.

$$\hat{k}_i(y_i) = \sum_{k=1}^{n_K} \hat{w}_k^{K_i} \phi_k(y_i) \quad (24)$$

$$\hat{d}_i(y_i) = \sum_{k=1}^{n_D} \hat{w}_k^{D_i} \phi_k(y_i) \quad (25)$$

ここに, n_K , n_D はそれぞれ駆動力特性モデル, 減衰モデルにおける基底関数の個数である. 重力 $\mathbf{f}_G(\mathbf{y})$ は姿勢によって変化するので重み $\hat{w}_{kl}^{G_i}$ を用いて重力モデル $\hat{\mathbf{f}}_G(\mathbf{y})$ を次式で与える.

$$\hat{f}_{Gi}(\mathbf{y}) = \sum_{k=1}^{n_G} \sum_{l=1}^{n_G} \hat{w}_{kl}^{G_i} \phi_k(y_1) \phi_l(y_2) \quad (26)$$

ここに, n_G は重力モデルにおける基底関数の個数である.

基底関数は次式で与える⁽¹⁰⁾.

$$\phi_j(y_i) = \left\{ 1 - \frac{1}{1 + e^{\sigma(y_i - \theta_j)}} \right\} \frac{1}{1 + e^{\sigma(y_i - \theta_j)}} \quad (27)$$

式 (18)~(27) のように基底関数ネットワーク表現によって, 運動モデルは重み $\hat{J}_1$, $\hat{w}_k^{C_1}$, $\hat{w}_k^{C_2}$, $\hat{w}_k^{K_i}$, $\hat{w}_k^{D_i}$, $\hat{w}_{kl}^{G_i}$ に対して線形となっている. これらの基底関数ネットワークに含まれる重みを決定できれば制御系の中に運動モデル $\hat{\mathbf{J}}(\mathbf{y})$, $\hat{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}})$, $\hat{\mathbf{D}}(\mathbf{y})$, $\hat{\mathbf{f}}_G(\mathbf{y})$, $\hat{\mathbf{K}}(\mathbf{y})$ を獲得できる. 次節ではこれらを表現するための重みを獲得する方法について述べる.

3.2 運動系のモデル獲得のための学習則 ここ

では, Popov の超安定論⁽¹³⁾ ⁽¹⁴⁾により得られるモデル獲得に必要な $\hat{J}_i$, $\hat{w}_k^{C_i}$, $\hat{w}_k^{K_i}$, $\hat{w}_k^{D_i}$, $\hat{w}_k^{G_i}$ の学習則を以下に与える.

$$\dot{J}_1 = \alpha_J \varepsilon_1 \ddot{y}_{M1} \phi_k(y_2) \quad (28)$$

$$\dot{J}_2 = \alpha_J(\varepsilon_1 \ddot{y}_{M1} + \varepsilon_2 \ddot{y}_{M2})\phi_k(y_2) \quad (29)$$

$$\dot{w}_k^{C_1} = \alpha_C \{ \varepsilon_1 \ddot{y}_{M1} + (\varepsilon_1 - \frac{1}{2} \varepsilon_2) \dot{y}_1^2 \phi'_k(y_2) \} \quad (30)$$

$$\dot{\hat{w}}_k^{C_2} = \alpha_C \{ \varepsilon_1 \ddot{y}_{M2} + \varepsilon_2 \dot{y}_1 \dot{y}_2 \phi'_k(y_2) \} \quad (31)$$

$$\dot{\hat{w}}_k^{K_i} = -\alpha_K \varepsilon_i \phi_k(y_i) v_i \quad (32)$$

$$\dot{\hat{w}}_k^{D_i} = -\alpha_D \varepsilon_i \phi_k(y_i) \dot{y}_i \quad (33)$$

$$\dot{\hat{w}}_{kl}^{G_i} = \alpha_G \varepsilon_i \phi_k(y_1) \phi_l(y_2) \quad (34)$$

ここに、

$$\varepsilon_i = e_i + c\dot{e}_i \quad (35)$$

以上の式の中で、 $\alpha_{(j)}$ は正の定数である。c は Popov の超安定論において強正実条件を満足するために必要な定数である。これらの学習則の導出と強正実条件については付章を参照されたい。

ここでは簡単のため、2 次の単位行列 $\mathbf{I}_2$ を用いて $\mathbf{A} = a\mathbf{I}_2$, $\mathbf{B} = b\mathbf{I}_2$ とおいた場合の強正実条件を与えておく. $\bar{\mu}$ を $\bar{\mu}\mathbf{I}_2 - \hat{\mathbf{J}}(\mathbf{y})^{-1}\mathbf{J}(\mathbf{y})$ が正定となる定数に設定すると強正実条件は次式となる.

$$c > \frac{\bar{\mu}}{a} \quad (36)$$

本研究では、以上のように運動モデルの獲得により目標とする運動を達成することを運動学習と呼ぶ。

また、本研究では式 (28)~(34) の学習則は 2 自由度リンク運動系に対して与えているが、3 自由度以上の運動系に対しても同じ手順に従って学習則を導くことができる。ただし、3 自由度以上では、リンクの幾何学的な構造が異なるものが存在するので、学習則を変更する必要がある。たとえば、3 自由度以上ではリンクが 1 列に配置される場合 (回転軸が平行である必要はない) と、1 つのリンクから 2 つ以上のリンクが分岐する構造が考えられる。そのような場合は慣性行列の回転軸に依存する関係が異なるので、基底関数ネットワークの構成を変更しなければならない。すなわち、同じ自由度であっても運動系の幾何学的な構造が異なる場合は、それを反映するように基底関数ネットワークを構成し直し、学習則を変更する必要がある。

4. H_∞ 制御の組み込みによるモデル獲得誤差の低減

基底関数ネットワークにより運動系に含まれる非線形性をモデル化したが、モデル化誤差とモデル化され

Fig. 2 Schematics of proposed control structure

なかった非モデル化力により軌道誤差が現れる。そこで、これらを外乱として、外乱抑制に効果のある H_∞ 制御を組み込む。ここでは概略を述べるが、詳細は文献⁽¹²⁾を参照されたい。

式(8)に H_∞ 制御出力 $\mathbf{u}_\infty$ を次式のように追加する.

$$\mathbf{v} = -\mathbf{K}_D \dot{\mathbf{y}} - \mathbf{K}_P (\mathbf{y} - \mathbf{u} - \mathbf{u}_B - \mathbf{u}_\infty) \quad (37)$$

ここに, $\mathbf{u}_\infty = [u_{\infty 1} \cdots u_{\infty m}]^T$ は H_∞ 制御器からの制御出力である. H_∞ 制御器を組み込むと誤差系 (16) は次式となる.

$$\ddot{e}_i + a_i \dot{e}_i + b_i e_i = -r_i + b_i u_{\infty i} \quad (38)$$

H_∞ 制御器の伝達関数を $K_\infty(s)$ とすると, H_∞ 制御出力 $\mathbf{u}_\infty$ は軌道誤差ベクトル $\mathbf{e}$ を H_∞ 制御器への入力として次式で与える.

$$u_{\infty i}(s) = K_{\infty i}(s)e_i(s) \quad (39)$$

本研究で提案する運動制御手法を図2のブロック線図として示す。この図では運動制御系は大きく分けて、フィードバック系、運動学習系、逆システム、 H_∞ 制御器の4個の要素から構成されている。フィードバック系は運動系を安定化する。運動学習系は運動系のモデルを獲得しながらフィードバック系のゲイン調整をおこない、非線形な運動系を線形化する。逆システムは運動学習系によって得られた線形系への入力を生成する。これらのブロックによって誤差系は図中の点線で囲まれた線形な伝達関数 $\mathbf{P}(s) = \text{diag}\{P_1(s), \dots, P_m(s)\}$ に帰着される。ここに、 $P_i(s) = b_i/(s^2 + a_i s + b_i)$ である。この誤差系に対して式(38)のようにモデル獲得誤差 $\mathbf{r}$ が加わるので、 H_∞ 制御器 $\mathbf{K}_\infty(s)$ によりモデル獲得誤差を抑制する。ここに、 $\mathbf{K}_\infty(s) = \text{diag}\{K_{\infty 1}(s), \dots, K_{\infty m}(s)\}$ である。ただし、 H_∞ 制御器が機能するのは運動学習後である。

5. 運動系のモデル獲得による軌道制御実験

5.1 2種類の2自由度リンク運動系

本研究では、運動系の幾何学的な構造が同じであれば、提案す

Fig. 3 2 types of 2-degree-of-freedom motion system

Fig. 4 Motion systems used in experiment

る手法が運動方程式の構造に依存しないことを示すため、図3に示す2種類の運動系:(a)回転軸が平行な2自由度リンク運動系と、(b)回転軸が直角なねじれの位置にある2自由度リンク運動系に適用する。これらの運動系では第1リンクと第2リンクのなす角が直角のとき、第2関節角 $y_2 = 0$ とする。これら2つの運動系では、リンク間の干渉の仕方が異なるため、慣性行列、コリオリ力ベクトルが異なり、重力の作用が異なる。それぞれの運動方程式に含まれる慣性行列、コリオリ力ベクトル、重力ベクトルは以下の式となる。

運動系 (a) 回転軸が平行な2自由度リンク運動系の場合:

$$J(y) = \begin{bmatrix} J_{A1} + J_{A2} - 2J_{12}\sin y_2 & J_{A2} - J_{12}\sin y_2 \\ J_{A2} - J_{12}\sin y_2 & J_{A2} \end{bmatrix} \quad (40)$$

$$c(y, \dot{y}) = J_{12}\cos y_2 \begin{bmatrix} -2\dot{y}_1\dot{y}_2 - \dot{y}_2^2 \\ \dot{y}_1^2 \end{bmatrix} \quad (41)$$

$$f_G(y) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (42)$$

運動系 (b) 回転軸が直角なねじれの位置にある2自由度リンク運動系の場合:

$$J(y) = \begin{bmatrix} J_{B1} + J_{B2} + 2J_{12}\frac{a_2}{l_1}(\sin y_2 - \frac{a_2}{2l_1}\cos^2 y_2) & 0 \\ 0 & J_{B2} \end{bmatrix} \quad (43)$$

$$c(y, \dot{y}) = J_{12}\frac{a_2}{l_1}\cos y_2(1 + \frac{a_2}{l_1}\sin y_2) \begin{bmatrix} 2\dot{y}_1\dot{y}_2 \\ -\dot{y}_1^2 \end{bmatrix} \quad (44)$$

$$f_G(y) = \begin{bmatrix} 0 \\ -m_2a_2g\sin y_2 \end{bmatrix} \quad (45)$$

以上の式の中で、 J_{Ai} , J_{Bi} はそれぞれ運動系 (a), (b) の場合の第 i リンクの慣性モーメントに相当する定数、 $J_{12} = m_2l_1^2$, l_i は第 i リンクの軸間距離、 m_i は第 i リンクの質量、 a_i は第 i 関節軸から第 i リンクの重心までの距離である。運動系 (a) では重力ベクトル $f_G(y) = 0$ としているが、関節軸を完全に鉛直に立てることはできないので重力が作用し、重力を補償しなければ軌道誤差が生じる。運動系 (b) の第1関節軸も同様である。

これまで著者が提案してきた手法⁽¹¹⁾⁽¹²⁾ではこれらの2つの運動系に対して式(40), (41), (43), (44)のように解析的に慣性行列とコリオリ力ベクトルの構造を確定しなければならなかったため、慣性力、コリオリ力をモデルとして獲得する式(28)~(31)を別々に表現する必要があった。しかし、本研究では慣性行列とコリオリ力ベクトルの学習則を別々に記述する必要はない。

5.2 実験装置と実験条件の設定 ここでは、軌道制御実験において運動モデルが獲得され、軌道追従性能が向上することを示す。実験に用いた2種類の2自由度リンク運動系を図4に示す。この2自由度リンク運動系は、小型で高トルクを発生できるが強い非線形性を持つ電磁アクチュエータ(新電元製ロータリソレノイド)によってダイレクト駆動され、関節角は12bitのアブソリュート型ロータリエンコーダ(マイクロテックラボラトリ製)を用いて検出する。ロータリエンコーダの分解能は $7.66 \times 10^{-4}[\text{rad}] (= 3.14/4096[\text{rad}])$ である。制御系はフィードバック系と調整系に階層化されている。フィードバック式(8)をDSP(MTT製 HERON; TMS320C40, clock 60MHzを内臓)により実行し、サンプリング周波数は1[kHz]である。また、フィードバックゲイン行列 K_D , K_P , ベクトル u_B の調整と学習による $\hat{J}_i$, $\hat{w}_k^{Ci}$, $\hat{w}_k^{Ki}$, $\hat{w}_k^{Di}$, $\hat{w}_k^{Gij}$ の推定をホストコンピュータ(MTT製 Wing; Pentium, clock 266[MHz]を内臓)により実行し、サンプリング周波数は100[Hz]である。また、定数 $n_C = 10$, $n_K = n_D = n_G = 6$, $a = \sqrt{2}\omega$, $b = \omega^2$, $\omega = 18.9[\text{rad/s}](3[\text{Hz}])$ とする。

5.3 H_∞ 制御器の設計 H_∞ 制御において一般的な混合感度問題として H_∞ 制御器を設計する。感度重み関数 W_S と相補感度重み関数 W_T の交点を $18.9[\text{rad/s}](3[\text{Hz}])$, W_S^{-1} の低域感度と W_T^{-1} の高域感度を $-40[\text{dB}](1/100)$ に設定し、感度、相補感度に対する重み伝達関数を次式で設定する。

$$W_S(s) = \frac{18.9}{s + 0.189} \quad W_T(s) = \frac{100s}{s + 1890}$$

Fig. 5 Gain of sensitive function $S(s)$ and complementary sensitive function $T(s)$

Fig. 6 Objective rectangular trajectories of angles

以上の設計仕様に対して得られた H_∞ 制御器の状態表現は以下となる。

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}_\infty \mathbf{x} + \mathbf{B}_\infty e \\ u_\infty &= \mathbf{C}_\infty \mathbf{x}\end{aligned}$$

行列 $\mathbf{A}_\infty$, $\mathbf{B}_\infty$, $\mathbf{C}_\infty$ は文献⁽¹²⁾を参照されたい。また, H_∞ 制御器の伝達関数は行列 $\mathbf{A}_\infty$, $\mathbf{B}_\infty$, $\mathbf{C}_\infty$ を用いて次式となる。

$$K_\infty(s) = \mathbf{C}_\infty (s\mathbf{I} - \mathbf{A}_\infty)^{-1} \mathbf{B}_\infty \quad (46)$$

H_∞ 制御器により得られる感度関数 $S(s)$ と相補感度関数 $T(s)$ を図5に示す。この図から, 外乱の振動数が $1[\text{rad/s}]$ 以下では, 感度関数 $S(s)$ のゲインが $-10[\text{dB}]$ となっており, $1[\text{rad/s}]$ 以下の外乱ではその影響が $1/10$ 以下に抑えられることがわかる。

5.4 目標軌道の設定 目標軌道を図6に示す。この目標軌道は運動系 (a) において空間座標における第2リンクの先端軸が出発点から一辺が 40mm の正方形を描いて出発点に戻る軌道となっている⁽¹²⁾。同じ目標関節角軌道を運動系 (b) にも適用する。

5.5 実験結果と考察 図7に運動系 (a) における軌道誤差を示す。まず, 運動学習前 (Before M.L.) を見ると, 静止状態 ($0 \sim 0.25[\text{s}]$ と $4 \sim 5[\text{s}]$) において軌道誤差が存在することから, 関節軸の傾きによる重力が作用することが確認できる。重力による軌道誤差が大きいので, 慣性力やコリオリ力, フィードバックによる遅れなどによる軌道誤差の挙動を見ることはできない。次に目標軌道に100回 (約10分) 追従させ, 運動学習をおこなった場合 (After M.L.) の軌道誤差を見

Fig. 7 Errors of angular trajectories in motion system(a)

Fig. 8 Errors of angular trajectories in motion system(b)

ると, 静止状態 ($0 \sim 0.25[\text{s}]$ と $4 \sim 5[\text{s}]$) での軌道誤差が非常に小さいことから, 重力による軌道誤差は重力モデルの獲得によりほとんど消滅していることがわかる。また, 第1リンク, 第2リンクのいずれかが運動している状態 ($1 \sim 2[\text{s}]$ と $3 \sim 4[\text{s}]$) においても誤差は小さい。これは一方のリンクの運動が他方のリンクの運動に影響を与えていないこと, すなわちリンク間の干渉が抑制されていることを示しており, 慣性行列モデルの獲得が有効に働いていることを示している。しかし, 2つのリンクが同時に運動している状態 ($0.25 \sim 0.75[\text{s}]$ と $2.25 \sim 3[\text{s}]$), 特に第1リンクが静止状態から運動を開始する状態において軌道誤差が他の状態よりも顕著に現われている。これは運動を開始するときの静止摩擦力とコリオリ力などの速度に依存する力が相乗して現われるためと考えられる。すなわち, 静止摩擦などモデル化されていない力と, 速度に依存するためにモデル化誤差の影響が大きいコリオリ力などが発生する状態であり, モデル獲得だけではこれ以上軌道誤差を抑制することが困難であると考えられる。また, 第1関節軸の軌道誤差と比較して第2関節軸での軌道誤差が大きいのは, 第2リンクの慣性モーメントは第1リ

Fig. 9 Objective circular trajectories of angles

Fig. 10 Errors of angular trajectories in motion system(a) using circular trajectories as objective trajectories

リンクの慣性モーメントよりも小さいので、慣性力に対する摩擦力の影響が大きく、速度の小さい領域で摩擦力にくい止められるためと考えられる。

次に、運動学習をおこなった後に H_∞ 制御を組み込んだ場合 (After M.L. + H_∞ control) の軌道誤差を見ると、全体的に軌道誤差が抑制されていることがわかる。特に 1[rad] 以下の外乱ではその影響が 1/10 以下に抑えられるように設計されているので、静摩擦による軌道誤差の低減に効果があり、2つのリンクが同時に運動し、軌道誤差が最も発生し易い状態でも軌道誤差が低減されている。

さらに、運動系 (b) において同様な軌道追従実験をおこなった場合の軌道誤差を図 8 に示す。運動学習後の軌道誤差の挙動は運動系 (a) での挙動とほぼ同じであることから、提案する手法が運動方程式が異なる運動系に対しても有効であることが確認できる。

以上から本研究で提案する、運動系に含まれる非線形性を基底関数ネットワークにより獲得し、さらに H_∞ 制御によりモデル獲得誤差と非モデル化力の影響を低減する手法の有効性が確認できた。

5.6 検討 今回の実験では xy 平面上での正方形軌道を選んで運動学習をおこなったが、正方形軌道で運動学習をおこなった後に他の軌道、たとえば円軌道で軌道追従制御をおこなったとき、追従性能が改善

されているかという疑問が残る。そこで、ここでは正方形軌道での運動学習をおこなった後に円軌道での軌道追従性能を見ることにする。運動系 (a) において半径 22[mm] の円軌道を 2 周描く円軌道⁽¹¹⁾に対する関節軸の角度軌道を図 9 に示す。また、図 10 に正方形軌道で 100 回の運動学習をおこなった後に円軌道で軌道追従制御をおこなった場合の軌道誤差と、正方形軌道で運動学習をおこなった後に円軌道で 50 回の運動学習をおこなった場合の軌道誤差を示す。まず、正方形軌道で 100 回の運動学習をおこなった後に円軌道で軌道追従制御をおこなった場合 (After M.L. on Rec.) では、軌道追従性能はあまり向上していないことがわかる。その理由としては重力が関節角 y_1 と y_2 の両方に依存するので、重力モデルの重み $\hat{w}_{kl}^{G_i}$ の中で、正方形の場合の目標軌道からはずれたものが修正されないことと、加速度の強く作用する角度が正方形軌道と円軌道で異なることが考えられる。次に円軌道に変更した後に 50 回 (約 5 分) の運動学習をおこなった場合 (After M.L. on Rec. & Circ.) を見ると、第 1 関節軸の軌道誤差はほとんど消滅しているが、第 2 関節軸では第 1 関節軸と比較すると大きな軌道誤差が存在する。その理由としては、円軌道の場合は 2つのリンクが同時に動く状態が多いため、モデル化誤差の影響が出やすいコリオリ力が発生することと、第 2 リンクの慣性モーメントは第 1 リンクのものよりも小さいので、50 回では十分に慣性モーメントがモデル化されないことなどが考えられる。しかし、正方形軌道など一つの軌道でいったん運動学習をおこなっておけば、その後に軌道変更をおこなった場合でも始めから運動学習をやり直すよりも少ない時間で学習を済ませることができるが、実用上軌道変更を余儀なくされる場合は、あらかじめそれらの軌道で運動学習をおこなっておくことが望ましい。

学習を効果的にこなうためには、非線形要素が分離して現れる軌道を用いることが好ましいと考えられる。たとえば、慣性行列とコリオリ力の影響を抑えるために遅い動きで重力をモデル化し、その後、早い動きで慣性行列をモデル化する。また、アームの先端に近い第 2 リンクだけを動かし、次に第 1 リンクのみを動かすことによって、より効果的に慣性行列をモデル化することができる。

6. おわりに

本研究では基底関数ネットワークを用いて慣性行列を表現し、ラグランジュ方程式によりコリオリ力を与え、ネットワーク結合重みを推定するための学習則を

提案した。この手法では、異なる運動系であっても幾何学的な構造が同じであれば、運動方程式に含まれる慣性行列とコリオリカベクトルの学習則を運動系ごとに導き出す必要がない。本研究では提案する手法を2種類の運動系：(a) 回転軸が平行な2自由度リンク運動系と (b) 回転軸が直角なねじれの位置にある2自由度リンク運動系における軌道追従実験で提案する手法の有効性を確認した。次に、提案する手法によって発生するモデル獲得誤差による軌道追従誤差を抑制するために H_∞ 制御を組み込むことを提案し、高性能な軌道追従が可能となることを示した。

以上の結果から本研究で提案する運動制御法が有効であると結論できる。

付 章

学習則の導出 式 (5) に式 (15) を代入して整理することにより、次の誤差システムが得られる。

$$\hat{\mathbf{J}}(\mathbf{y})\{\mathbf{J}^*(\mathbf{y})\ddot{\mathbf{e}} + \mathbf{A}\dot{\mathbf{e}} + \mathbf{B}\mathbf{e}\} = -\mathbf{r}' \quad (47)$$

$$\begin{aligned} \mathbf{r}' = & -\tilde{\mathbf{K}}(\mathbf{y})\mathbf{v} + \tilde{\mathbf{J}}(\mathbf{y})\ddot{\mathbf{y}}_M + \tilde{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}}) \\ & + \tilde{\mathbf{f}}_G(\mathbf{y}) + \tilde{\mathbf{D}}(\mathbf{y})\dot{\mathbf{y}} \end{aligned} \quad (48)$$

ここに、 $\mathbf{J}^*(\mathbf{y}) = \hat{\mathbf{J}}(\mathbf{y})^{-1}\mathbf{J}(\mathbf{y})$ である。Popov の超安定論^{(13),(14)}によれば、 $\mathbf{r}'$ から ε への伝達関数が強正実となり、次の積分不等式

$$\int_0^t \varepsilon^T \mathbf{r}' dt \geq -\delta^2 \quad (49)$$

が成立すれば、誤差システム (47) は漸近収束する。ここに、 ε^T は ε の転置ベクトル、 δ^2 は時間に依存しない正の定数である。学習則 (28)~(34) は積分不等式 (49) を満足する。ここでは、慣性行列とコリオリカベクトルについて示す。式 (48) を式 (49) に代入し、慣性力項とコリオリ力項を取り出し、学習則 (28)~(31) を用いると、

$$\begin{aligned} & \int_0^t \varepsilon^T \{\tilde{\mathbf{J}}(\mathbf{y})\ddot{\mathbf{y}}_M + \tilde{\mathbf{c}}(\mathbf{y}, \dot{\mathbf{y}})\} = \\ & \int_0^t \underbrace{\{\hat{J}_1 \varepsilon_1 \ddot{y}_{M1} + \hat{J}_2 (\varepsilon_1 \ddot{y}_{M1} + \varepsilon_2 \ddot{y}_{M2})\}}_{\hat{J}_1} dt + \\ & \int_0^t \sum_{k=1}^n \underbrace{w_k^{C1} \{\varepsilon_1 \ddot{y}_{M1} \phi_k(y_2) + (\varepsilon_1 \dot{y}_1 \dot{y}_2 - \frac{1}{2} \dot{y}_1^2) \phi'_k(y_2)\}}_{\dot{w}_2^{C1}} dt \\ & + \int_0^t \sum_{k=1}^n \underbrace{w_k^{C2} \{\varepsilon_1 \ddot{y}_{M2} + \varepsilon_2 \ddot{y}_{M1}\} \phi_k(y_2) + \dot{y}_2^2 \phi'_k(y_2)}_{\dot{w}_2^{C2}} dt \\ & \geq -\frac{1}{2} [\hat{J}_1^2 + \hat{J}_2^2 + (\dot{w}_2^{C1})^2 + (\dot{w}_2^{C2})^2]_{t=0} \equiv -\delta_J^2 \end{aligned} \quad (50)$$

他の項についても同様に証明できるが、詳細は文献^{(10) (11)}を参照されたい。

強正実条件 式 (47) の両辺に $\hat{\mathbf{J}}(\mathbf{y})^{-1}$ をかけ、次式のように変形する。

$$\mathbf{J}^*(\mathbf{y})\ddot{\mathbf{e}} + \mathbf{A}\dot{\mathbf{e}} + \mathbf{B}\mathbf{e} = -\mathbf{r}^* \quad (51)$$

ここに、 $\mathbf{r}^* = \hat{\mathbf{J}}(\mathbf{y})^{-1}\mathbf{r}'$ である。

$\mathbf{A} = a\mathbf{I}$, $\mathbf{B} = b\mathbf{I}$, $\mu_i (i = 1, 2)$ を $\hat{\mathbf{J}}(\mathbf{y})$ の固有値とすると、 $\mathbf{r}^*$ から ε の伝達関数が強正実となる条件は

$$c > \max_{i=1,2} \left(\frac{\mu_i}{a} \right) \quad (52)$$

である。この証明は文献⁽¹¹⁾を参照されたい。 $\bar{\mu}$ を $\bar{\mu} > \mu_i$ とすると、強正実条件は式 (36) となる。

References

- (1) 細江他 4 名, 制御系設計, システム制御情報学会, (1994)
- (2) Bong Soo Kang et al, *Robust Tracking Control of a Direct Drive Robot*, J. of Dyn. Sys., Meas. and Cont., Trans. of ASME, JUNE, (1999), Vol.121, 261/269
- (3) 川人, 小脳の内部モデルと運動学習, 計測と制御, Vol.33, No.4, (1994), 296/303
- (4) 河野, 設楽, 感覚から運動への変換, 計測と制御, Vol.33, No.4, (1994), 274/281
- (5) 赤澤, 運動制御と筋特性, 計測と制御, Vol.33, No.4, (1994), 304/309
- (6) R.M.Sanner and J.J.E.Slotine, *Gaussian Networks for Direct Adaptive Control*, IEEE Trans. on Neural Network, 3-6, (1992), 837/863
- (7) Kawato et al, *A Hierarchical Neural-Network Model for Control and Learning of Voluntary Movement*, Biological Cybernetics, 57, (1987), 169/185
- (8) 大須賀, 非線形メカニカルシステムの適応制御, 計測自動制御学会論文集, (1986), Vol.22, No.7, 38/44
- (9) 川崎, オンラインパラメータ同定によるマニピュレータの軌道制御, 計測自動制御学会論文集, (1984), Vol.20, No.9, 8/15
- (10) 安東, 岩壺, 剛体運動系のモデル獲得と制御 (第 1 報), 日本機械学会論文集 (C), (1998), Vol.64, No.625, 3391/3397
- (11) 安東, 岩壺, 剛体運動系のモデル獲得と制御 (第 2 報), 日本機械学会論文集 (C), (2000), Vol.66, No.649, 2988/2995
- (12) 安東, 岩壺, 剛体運動系のモデル獲得と制御 (第 3 報), 日本機械学会論文集 (C), (2001), Vol.67, No.662, 3160/3167
- (13) V.M.Popov, *The Solution of A New Stability problem for Controlled System*, Automation and Remote Control, (1963), Vol.24, 1/23
- (14) 富塚, 適応制御システムの理論と実際, オーム社 (1981)