〔解 説〕

ロボット制御系の設計*

有 本 卓**

1. はじめに

ロボットの制御系の設計は、オペレータによる何らかの教示を前提とする場合、それほど難しい問題は含まない。しかし、ロボットに自律性を要求するとなると、制御系の設計法が重要な役割を演じることになる。さらに、ロボットに高速かつ高精度の作業を課すとなると、制御系の設計法の良し悪しが作業結果に直接的に影響を与え、ロボットの死命を制することになる。

ここでは、ロボットに与えられた種々の作業環境のもとでそれぞれ適すると思われるロボット制御法について説明し、制御系設計のシンセシス法を述べる。

2. 教示再生方式と PTP 制御

産業用ロボットの制御法は、現在でも、教示再生 (teaching/playback) 方式が主流である。この方法の考え方は、ロボットに課す作業の手順とそれを実現する動作を何らかの方法でオペレータが教え込むことにある。これを教示というが、教示内容にはハンド部および各関節の位置、速度、作業指令、実行順序等がある。また、その実現方法には、オペレータがロボットの先端近くに取り付けた握り (joystick) を手に取って、ロボットを実際の作業順序に従って動かしながら、その時々刻々の各関節角の位置や速度を記憶させる方式がある。これを直接教示方式というが、この方法では通常、位置と作業指令のみを教示し、速度については適正な値を別に計算させる。他方、作業手順を数値化あるいは記号化して、キーボードを通じてロボットに組み込んだコンピュータに入力すると共に、ティーチングボックスのスイッチ操作 (これはコンピュータを通じロボットのコマンド指令となる) によってロボットの運動を生成させる方法があり、これを間接教示方式という。

教示方式に従う場合、いずれにしても、各関節軸の制御は指令値を忠実に実現する方式が望ましいことになり、従って、高速性を要求しない限りパルスインクリメンタ

ル方式のサーボ系が使われやすくなる。しかし、ある程度の高速性が要求され、サーボモーターを使うとき、各関節の連続的な指令値を用意しておくが必要になる。教示の際、普通、ロボットの手先が連続的に動いても、通過した要所の点のデータのみが記憶される。一般に、点から点への (ある姿勢から別の姿勢への) 位置決めを PTP 方式 (Point to Point) と呼び、連続経路 (Continuous path) に追従させる制御を CP 方式という (図 1 参照)。前者では、普通には目標点あるいは目標姿勢に到達することのみが要求され、途中の経路は問題とされない。たとえば、スポット溶接作業は、このような点から点への位置決めおよび溶接ガンに対する姿勢と溶接作業指令の順序づけられた手順によって構成される。他方、アーク溶接や塗装では、連続経路に従って作業が行われるので、経路の所要所の点を適当に取り、これらの点間の連続的経路はコンピュータによる補間計算にゆだねることになる。手先の経路の補間には、直線補間や円弧補間の他、最近ではスプライン補間も出来るプログラムも開発されている (図 2 参照)。しかし、手先に指定した経路を実現するには、それを各関節変数の時間変数に変換する必要がある。手先効果器の位置と姿勢を表す変数を $\mathbf{x} = (x_1, \dots, x_n)^T$ 、関節変数を $\mathbf{q} = (q_1, \dots, q_n)^T$ で表すと、手先効果器を含めたアームの関節変数が決まれば姿勢が決まるので、一般にある関係式

図 1 PTP 制御方式と CP 制御方式の違い

1986 年 6 月 6 日

* Design of Robot Control Systems

** 大阪大学基礎工学部

図 2 手先軌道の補間法

$$x_i = f_i(q_1, \dots, q_n), \quad i=1, \dots, n \quad (1)$$

が成立する。これを順変換といいアームの機構が与えられると比較的に定められるが、逆変換

$$q_j = h_j(x_1, \dots, x_n), \quad j=1, \dots, n \quad (2)$$

の導出は一般的にはかなり面倒である。この導出が最も困難なのは垂直多関節型のマニピュレータであるが、その自由度が5ないし6の場合、既にいくつかの公式が導出されている。なお、PUMA-560のようにオフセットがある場合、位置のみならず速度や加速度まで含めた逆変換の公式はごく最近になってやっと完全にされつつあるようである¹⁾。

後で述べるように、作業座標に基づくCP制御の設計方式には、この逆変換の公式を必要とするものがいくつかあり、従って、完全な公式を求めておくとともに、これを効率よくかつ完全に計算できる技術の開発も重要なテーマであるが、ここでは触れない。

3. PD フィードバック制御方式

古典的なサーボ方式による位置決めは、位置と速度の線形フィードバックによって行われている。ロボットは多自由度であり、関節間には動的な干渉があるが、現在の産業用ロボットのほとんどの制御は、各関節に対応し

て独立なサーボループを構成する方式に従っている。言い換えると、目標姿勢 $q^0 = (q_1^0, \dots, q_n^0)$ が与えられたとき、各関節は独立なフィードバック

$$v_i = -\bar{a}_i(q_i - q_i^0) - \bar{b}_i \dot{q}_i, \quad i=1, \dots, n \quad (3)$$

に基づくサーボ系を構成している。これをPDフィードバック制御方式と呼ぶが、この方式は線形かつ各関節で独立であるにもかかわらず、非線形かつ動的干渉のあるロボットの制御に現実に用いられ、ある程度機能していると思われる。そこで、まずその理由を述べておこう。

一般にサーボ系を含めたロボットのダイナミクスは

$$\{J_0 + R(q)\} \ddot{q} + \dot{R}(q) \dot{q} - \frac{1}{2} \frac{\partial}{\partial q} \{\dot{q}^T R(q) \dot{q}\} + B_0 \dot{q} + g(q) = Dv \quad (4)$$

と表される²⁾。ここに $v = (v_1, \dots, v_n)^T$ はサーボモータの入力電圧、 D は対角行列で、各対角線要素はモータ定数やギヤ比に関する定数、 B_0 もダンピング定数から成る対角行列、 J_0 は

$$J_0 = \text{diag}(J_{ii}), \quad J_{ii} = J_i / r_i^2, \quad i=1, \dots, n \quad (5)$$

と表される対角行列である。ここに J_i はモータの内部(シャフト側)慣性モーメントであり、 r_i はギヤの減速比である。また $R(q)$ はロボットの運動学から決まる量で慣性行列と呼ばれる。これは、負荷側の各慣性モーメントを表すが、一般に対称かつ正定値である。しかし、この $R(q)$ は一般に g に依存し、かつ非対角型(すなわち、関節間に動的干渉が存在する)なので、式(4)の第二項や第三項が存在し、式(4)は非線形の運動方程式となる。なお、 $g(q)$ は重力の影響から来る項であり、重力項と呼ぶことにする。

さて、式(3)のPDフィードバックを行うと、式(4)から閉ループ系のダイナミクスは

$$\{J_0 + R(q)\} \ddot{q} + \dot{R}(q) \dot{q} - \frac{1}{2} \frac{\partial}{\partial q} \{\dot{q}^T R(q) \dot{q}\} + B \dot{q} + g(q) + A(q - q^0) = 0 \quad (6)$$

となる。ここに

$$\begin{cases} B = B_0 + D\bar{B}, & A = D\bar{A} \\ \bar{B} = \text{diag}(\bar{b}_i), & \bar{A} = \text{diag}(\bar{a}_i) \end{cases} \quad (7)$$

と置いた。ところで、一般に産業ロボットは暗々裡に次のような設計方針が採られているといわれる。

1) 関節アクチュエータとしてのモータの定格を大きめに選び、ギヤの減速比を大きくとっている。

2) 各サーボの前向き経路の増幅器のゲインを大きくすることにより、結果的に位置と速度のフィードバックの係数が大きくなっている。つまり、高ゲインのフィードバックを構成している。

3) バランスの良い機構設計を行って、重力の影響を避けたり、軽減したりしている。

これら3点から、式(6)では、 J_0 のそれぞれの対角要素は $R(q)$ の対応する対角要素よりドミナントになり、また線形フィードバックの項 $B\dot{q}$ と $A(q-q^0)$ の項も他の非線形項よりドミナントになっているものと思われる。結局、式(6)は線形のダイナミクス

$$J_0\ddot{q} + B\dot{q} + A(q-q^0) = 0 \quad (8)$$

で近似できるものと思われる。式(8)の各係数行列は対角行列であり、従って式(8)は各サーボについて独立なダイナミクスを表すので、古典的な1軸のサーボメカニズム理論に従って、ゲイン定数 $\bar{a}_i, \bar{b}_i$ を各関節で独立に選んでおけばよいことになる。

4. 重力補償つき PD フィードバック制御

式(3)のPDフィードバック方式は、特に重力の影響が存在しない関節軸には有効である。例えば、SCARA型ロボットの場合、主要な二つの回転軸が鉛直方向にあるので、これらの関節の運動は重力の影響を受けない。従って、残りの鉛直方向のスライド軸のみについて重力の影響を考えておけばよい。しかし、垂直多関節ロボットの場合、重力項が存在し、これによって単なる式(4)のPDフィードバックではオフセット(位置決めの定常偏差)が生ずるのが普通である。しかも、重力項 $g(q)$ は他の非線形項に比べて簡単な形をしており、リアルタイム計算もそれほど困難ではない。そこで、重力項はサンプル時刻ごとに計算して、ソフトウェア・サーボによってフィードフォワード補償しておくとともに、式(4)のフィードバックを行う方式が考えられる³⁾(図3参照)。サンプル周期を十分小さく取れたとすると、このことは連続時間的に入力信号が

$$v = D^{-1}g(q) = \bar{A}(q-q^0) - \bar{B}\dot{q} \quad (9)$$

と近似できていることと解釈できる。これをロボットのダイナミクス(4)に代入すると、

$$\{J_0 + R(q)\}\ddot{q} + \dot{R}(q)\dot{q} - \frac{1}{2} \cdot \frac{\partial}{\partial q} \{\dot{q}^T R(q) \dot{q}\} + B\dot{q} + A(q-q^0) = 0 \quad (10)$$

となる。この中の非線形項が線形項に比して小さければ、このダイナミクスは式(8)で近似できた上に、オフセットが生じないことが証明できる。しかも、この重力補償つきPD制御方式には、次に述べるように、もっと重要な機能が備わっていることが示される。

一般に、ロボットの運動が高速になると、式(8)の第二項や第三項の比重が増し、線形近似が成立しなくなる。その上、ギャ比があまり大きくとれないか、あるいはダイレクト・ドライブ方式のように $r_i=1$ となると、式(10)における J_0 と $R(q)$ のオーダーは同じか、あるいは逆転することになり、ダイナミクスは著しく

図3 重力補償つき PD 制御系

非線形的になる。ところが、このような条件下でも、上述の重力補償つきPD制御によって位置決めが保証できるのである。言い換えると、任意に与えられた初期条件

$$q(0)=q_1, \quad \dot{q}(0)=\dot{q}_1 \quad (11)$$

に対して、式(10)の解 $q(t)$ が $t \rightarrow \infty$ のとき q^0 に漸近的に近づくことが証明できるのである。実際、スカラ関数

$$W(q, \dot{q}) = \frac{1}{2} \dot{q}^T \{J_0 + R(q)\} \dot{q} + \frac{1}{2} (q - q^0)^T A (q - q^0) \quad (12)$$

を定義し、この時間微分を式(10)の解軌道 $q(t)$ に沿って計算すると

$$\dot{W} = -\dot{q}^T B \dot{q} \leq 0 \quad (13)$$

となることが容易に示される^{3)~5)}。つまり、式(10)を状態変数 $(q, \dot{q})$ に関する状態方程式として見たとき、式(12)で定義した $W(q, \dot{q})$ はリヤプノフ関数となっており、従って平衡点 $(q^0, \dot{q}=0)$ は安定になることが証明されている。しかも、 $\dot{W}=0$ は $\dot{q}=0$ を意味するが、そのとき式(10)から $q(t) \equiv q^0$ でない限り、 $\dot{q}(t)$ は0にとどまり得ないので、式(10)の最大不変集合は唯一点 $(q^0, 0)$ となり、こうして Lasalle の定理から

$$q(t) \rightarrow q^0, \quad \dot{q}(t) \rightarrow 0 \text{ as } t \rightarrow \infty$$

となることが証明される^{3), 4)}。

なお、式(8)のような線形2次系の位置決めの収束性は指数関数的であるが、非線形系である式(10)の位置決めの速さも指数関数のオーダーであることが示されている⁵⁾。その他、適当なリヤプノフ関数を選ぶことによって、ゲインの選び方と収束の速さとの関係が論じられるが、ここではこれ以上の詳細には触れない。

5. 人工ポテンシャル法

式(9)の重力補償つき制御法は、運動方程式の中の重力項の代りに偏差項を加えたことに相当する。従って、式(12)は運動方程式(10)に対する全エネルギーに相当し、式(12)の第二項は、重力に起因するエネルギー

の代りに新たに人為的に加えられた偏差項によるポテンシャルエネルギーと見なされる。また、式(13)は運動方程式(10)に対する散逸エネルギーに相当する。こうして、位置の偏差の線形フィードバックは2次形式の人工ポテンシャルエネルギーを導入したことになるが、2次形式とは異なるポテンシャル関数を導入することにより、非線形のフィードバック則を導くこともできる。たとえば、次のようなポテンシャル関数を導入してみる^{9),7)}

$$U(q) = \begin{cases} h(q) - \frac{\sigma}{2}, & h > \sigma \\ h^2(q)/2\sigma, & h \leq \sigma \end{cases} \quad (14)$$

ここに

$$h(q) = \sqrt{\sum_{i=1}^n r_i (q_i - q_i^0)^2} \quad (15)$$

とおいた。もし $r_i > 0$ ($i=1, \dots, n$) ならば、 $U(q)$ は正定値関数になり、そのプロファイルは自由度2のとき図4のようになる。これを円錐形ポテンシャルと呼ぶことにするが、これから導びかれるフィードバック則は、たとえば、 $r_i = \alpha^2$ ($i=1, \dots, n$) のとき Lagrange の運動方程式から

$$Dv = -\frac{\partial U(q)}{\partial q} = \begin{cases} -\alpha \|q - q^0\|^{-1}(q - q^0), & \|q - q^0\| > \frac{\sigma}{\alpha} \\ -\frac{\alpha^2}{\sigma}(q - q^0), & \|q - q^0\| \leq \frac{\sigma}{\alpha} \end{cases} \quad (16)$$

となる。すなわち、円錐形人工ポテンシャルを用いると、位置のフィードバックの大きさは各関節とも α 以上にはならず、位置のフィードバック飽和特性を持たしたこと

図4 円錐形ポテンシャル関数のプロファイル

になるのである。その他、ある種の人工ポテンシャルを想定することによって、Bang-Bang型に近いフィードバック制御法も得られるが、これらの特徴は何よりも位置決め時の漸近安定性が保証されることである。

6. CP制御と逆ダイナミックス法

CP制御の目標は、ある時間区間 $[0, T]$ にわたってロボットの姿勢の連続的変化 $q_d(t)$ が与えられたとき、これにロボットの実際の軌道 $q(t)$ が追従するように制御系を設計させることにある。軌道の速度成分 $\dot{q}_d(t)$ があまり大きくないときは、重力補償つきPDフィードバック制御

$$v = D^{-1}g(q) - \bar{A}(q - q_d) - \bar{B}\dot{q} \quad (17)$$

を用いても、実際の軌道 $q(t)$ は $q_d(t)$ から大きく離れることはなく、近似的に追従するものと思われる。また、与えられた軌道の速度成分が大きいときは、制御則

$$v = D^{-1}g(q) - \bar{A}(q - q_d) - \bar{B}(\dot{q} - \dot{q}_d) \quad (18)$$

も考えられる。しかし、いずれにしても、速い動作にはこれらのPDフィードバック則でも精度の高い追従性は得られない。このことは、2次系としての線形サーボ系でも見られるのであって、非線形性が直接的原因ではない。しかし、ここに追従誤差を小さくするための何らかの工夫が必要であることが見てとれる。そのために考えられた一つの方法が逆ダイナミックス法（あるいはトルク計算法）であり、他の一つが学習制御法である。後者については次節で述べることにして、前者の方法の考え方を述べておこう。

目標軌道 $q_d(t)$ が与えられたと仮定して、マニピュレータの運動方程式(式(4))の左辺のすべての項を目標軌道 $q_d(t)$ に沿って計算し、これらの和に見合う駆動入力（トルク）を各関節サーボ系に与えることができれば、理論的には軌道追従が実現できるはずである。この考え方は昔からあったが^{8)~10)}、ロボットの運動方程式は現実には複雑であるため、その計算が実時間では不可能であると考えられていた。ところが近年、マイクロプロセッサ技術の進展と高速アルゴリズムの発見によって、近い将来にはこの方式の実現が見通し出来るようになってきている。

ところで、ダイナミックス計算のアルゴリズムにはNewton-Euler法¹¹⁾とLagrange¹²⁾法の二つがある。前者は、基底リンクから手先までの位置、速度、加速度（角速度あるいは角加速度）を求め、次いで逆順に手先から基底リンクまでの外力と外部モーメントを求め、そしてリンクに加えるべき駆動力を求めるものである。後者はこのような考え方をLagrangeの運動方程式に基づいて展開したものになっており、いずれにしても、これらのアルゴリズムの計算法は1サンプル当りの計算数が自由

図 5 逆ダイナミクス法の実時間制御系

度 n に比例することが示されている。この考え方は、最近では力制御や位置と力のハイブリッド制御にまで適用するように拡張されており¹³⁾、将来のメカトロニクスの制御技術の一方法として定着する日も来るように思われる。しかし、現在の産業用ロボットへの応用となると、解決すべきいくつかの困難さが残っており、未だ大学や研究所の一つの試みの段階にとどまっている。ここではその一例を図 5 に示す¹⁴⁾。これはカーネギー・メロン大学 (CMU) で設計された DD ロボットの制御系として構築されたもので、研究計画の遂行の便宜から CPU に VAX-11/750 を用いているが、逆ダイナミクスの計算は MC 68000 が受け持っている。つまり、ロボットの制御系は直接的には MC 68000 が支配している。また、各関節には分散制御装置として TI 社の信号処理プロセッサ TMS-320 が装備されている。各サンプル周期毎に MC 68000 は各関節の角度と角速度のデータを吸い上げ、これらを Marincos アレイプロセッサに転送し、逆ダイナミクスの計算を Marincos アレイプロセッサ上で走らす。つまり、後者は MC 68000 の算術演算ユニットとして働くよう設計されている。なお、図 5 のシステムでは自由度 6 のロボットに対して Newton-Euler 法が実装されたが、その時の 1 サンプル周期の計算時間は表 1 であったことが報告されている。

7. CP 制御と学習制御法

一般にロボットの各要素や要素間の物理定数を正確に求めるのは困難であり、伝達機構の摩擦や flexibility、部材の elasticity、等々の式で表現できない現象があり、逆ダイナミクス法といえども与えられた軌道にロボットを完全に追従させるのは至難である。ところで、ロ

ットに教示する手間と時間を考えると、ロボット自身に練習を繰り返させ、与えられた軌道を何回かの試行のうちに自動的に覚えさせる学習の方法が有効のように思われる。

この最初のアイデアは内山¹⁵⁾によって提案されたが、永い間、等閑視されていた。筆者らは、誤差微分を修正項として使う新しい方法で試行ごとに軌道追従が必ず改善できることを示し、ここに始めて学習制御法としての枠組みが可能なることを示し¹⁶⁾、以後、種々の改良や新しい提案が見られるに至っている^{17)~25)}。現時点で最も実地的な学習制御法は、ダイナミクスの既知の部分のトルク成分を理想の軌道まわりで計算して補償しておき、残りの部分を開ループ系として学習系を組む方法であろう。その中の一例として、重力項を補償するとともに PD フィードバックによりある程度軌道追従させた上で残りの項を学習系によって繰り返し近似させる方法を図 6 に示す。なお、3 節で述べたように、ロボットのダイナミクスが式 (8) の線形系で近似できるなら、この係数 J_0, B, A のおおよその値は既知と仮定できるので、これらを用いて学習系を図 7 のように組むことが考えられる。なお、この方式の物理的背景とともに、初期設定

表 1 Marincos processor による 1 サンプル周期当たりの演算実行時間

Operation (Floating point)	Number	Execution time (μs)	Total execution time (ms)
Multiplication	303	2.0	0.606
Addition	226	2.0	0.552
Negation	80	0.125	0.010
Sine-cosine	6	15	0.09
Total			1.258

図 6 ロボットの学習制御系

の誤差や試行中に入る外乱の影響を論じた最近の研究結果は近年筆者等によって発表される²⁶⁾。

8. 作業座標に基づく制御法

ロボットの自律制御ではロボットに行わせる作業の記述の問題も重要である。ここでは、作業が関節座標系よりも手先の位置および手先効果器の姿勢から成る作業座標系で記述された場合の制御法を述べる。

さて、PTP 制御の目標点が作業座標系 x° で与えられたとき、もし作業座標系から関節座標系への逆変換が容易に計算できるなら、式 (2) より $q^\circ = h(x^\circ)$ が求まり、これを改めて目標姿勢として、たとえば PD フィードバック方式が使える。しかし一般に x° が正確に与えられても、ロボットの物理定数の不正確さから逆変換によって決められた $q^\circ = h(x^\circ)$ を用いても正しい位置決

図 7 線形部分のみ逆ダイナミクス近似した学習制御系

めは得られず、いくつかの誤差が伴うのが普通である。従って、逆変換せずに、直接作業座標 x° を実現する制御方式が重要になる。そのためには、作業座標 $x(t)$ を直接測定することが必要になる。一般に、作業座標系をロボットの外部から直接的に（すなわち、ベースに固定した慣性座標で）測ることのできるセンサーを外界センサーと呼び、ロボットに取り付けた関節の位置や速度を測るセンサーを内界センサーと呼ぶ。ここでは、内界センサーによって関節座標系 $q, \dot{q}$ が測定でき、また外界センサーによって作業座標系 x も測定できる場合を考える。そのとき、これらの測定データに基づく次の重力補償つきフィードバック制御が考えられる（図 8 参照）。

$$v = D^{-1}g(q) - \bar{B}\dot{q} - D^{-1}J^T(q)K(x - x^\circ) \quad (19)$$

ここに、 $J(q)$ はヤコビアン行列で、次のように定義さ

図 8 作業座標に基づく重力補償つき PD 制御系

図 9 ロボットの MRAC 型適応制御系

れる。

$$J(q) = (J_{ij}), \quad J_{ij} = \partial h_i(q) / \partial q_j \quad (20)$$

また、 K は適当な正値対角行列である。

式 (19) の制御法によると、任意の初期条件のもとで、次のような作業座標上の位置決め漸近安定性が示される。

$$x(t) \rightarrow x^0 \text{ \& } q^0(t) \rightarrow 0 \text{ as } t \rightarrow \infty \quad (21)$$

なお、式 (19) では逆座標変換の代りにヤコビアン行列の計算を必要とするが、一般に後者の計算の方が簡単であり実装が容易である。なお、逆ダイナミクス法の場合には、逆変換を実時間でまともに計算せねばならず、ここにも何らかの高速化が必要である。

9. 適 応 制 御 法

ロボットのある種の作業では、作業中の物理量の変動が避けられない場合がある。たとえば、ペイロードの変動がそうである。このような変動に追従できる制御法として適応制御の考え方を応用することが考えられる。この方向は Dubowsky²⁷⁾ によって始められた。彼等は参照システムとして 2 次のアンダーダンモデルを選び、実システムの出力と参照システムの出力の差の 2 乗和を最小にする最急降下法によって局所的な PD サーボの可変ゲインを調節した。この方法は、図 9 に示すようないわゆる MRAC (Model Reference Adaptive Control) の考え方に基づくものであるが、本質的には実ロボットシステムの非線形項、干渉項を無視した 1 自由度の単入力単出力システムに対する設計であった。しかも、実システムに適用した場合の調整則の安定性も保証されていなかった。

この考え方とは別に、ロボットの非線形性および各関節間の干渉を考慮し、CP 制御に適した簡単な制御法が竹垣等²⁸⁾によって提案されている (図 10 参照)。この方式では、一般の MRAC のような参照モデルは介在しない。

図 10 ロボットの適応制御系のブロック図

いが、これは、システムへの入力に目標軌道の加速度項が含まれ、フィードフォワード部分に適応機構を設けていないためである。そこでは、適応調整則の収束性がリヤプノフ法の安定論によって軌道まわりで線形化したシステムのもとで示されている。

また、Balestrino 等²⁹⁾ は、ロボットのダイナミクスに最も直接的に MRAC を構成し、Popov の超安定性の理論を用いて調整則の収束性を証明している。彼等の方法では、ロボットシステムおよび参照モデルの状態変数をそれぞれ x_P, x_M とすると、サーボ入力は $u_P = \Phi(v, x_P, t) - K_P x_P + \Psi(v, u_M, t) u_M + K_u u_M$ (22) と定められる (図 9 参照)。ここに、 $-K_P x_P$ でフィードバック部分を、 $K_u u_M$ (u_M は参照モデルへの入力) でフィードフォワード部分を構成している。また、 Φ と Ψ は次のように調整される。

$$\begin{cases} \Phi = q \frac{v}{\|v\|} (\text{sgn } x_P)^T \\ \Psi = p \frac{v}{\|v\|} (\text{sgn } u_M)^T \end{cases} \quad (23)$$

ここに $v = D(x_M - x_P)$ であり、パラメータ q, p は適

図 11 ロボットのセルフチューニング型適応制御系

応メカニズムの収束性を考慮して決められる。また、行列 D を適当に選ぶことにより、適応スピードが調整できる。

適応制御の考え方には、MRAC の他、セルフチューニング法がある。この考え方は図 11 に示すように、ロボットのダイナミクスを表現する離散時間モデルの各パラメータの推定機構と、その結果を用いて制御ゲインあるいは制御入力を決定する部分から成る。Koivo 等³⁰⁾ は、次のような入出力数がロボットの自由度と同じである ARMA モデルを用いて、セルフチューニング型の適応制御法をロボットに適用している。

$$y(k) = A(z^{-1})y(k) + (z^{-1})u(k-d) + h + e(k) \quad (24)$$

ここに $y(\cdot), u(\cdot), e(\cdot) \in R^n$, k はサンプル時刻を表し, $k=0, 1, \dots$ である。また, d はサンプル周期の整数倍の時間遅れを表す。定数ベクトル h は強制項であり、重力などの影響を含む。誤差ベクトル $e(\cdot)$ はガウス性白色雑音と仮定する。式 (24) の各パラメータは recursive な最小 2 乗法で決められる。一方、モデルパラメータの推定終了後の制御入力を決定するアルゴリズムに関しては種々提案されているが、ここではこれ以上の詳細には立ち入らない。

10. お わ り に

上で述べた方式以外にもロボット制御の設計論には多くの提案があり、その中の重要と思われるものも割愛せざるを得なかった。実際、力制御や位置と力のハイブリッド制御も重要な課題であるが、この方向の制御系設計論は別の機会にまとめてみたい。また、ロボットの伝達機構の flexibility を考慮した制御系設計³¹⁾ も重要であるが、この方向の基礎的考察はあまり進んでいないようである。

参 考 文 献

- 1) S. Elgazzar, "Efficient Kinematic Transformations for the PUMA 560 Robot", IEEE J. of Robotics and Automation, Vol. RA-1, No. 3, pp. 142-151, 1985
- 2) 有本卓, "サーボ理論とロボットの制御", 日本ロボット学会誌, Vol. 2, No. 5, pp. 480-486, 1984
- 3) M. Takegaki and S. Arimoto, "A New Feedback Method for Dynamic Control of Manipulators", Trans. of ASME, J. of DSMC, Vol. 103, pp. 119-125, 1981
- 4) S. Arimoto and F. Miyazaki, "Stability and Robustness of PID Feedback Control for Robot Manipulators of Sensory Capability", in M. Brady & R. P. Paul (ed.), Robotics Research: First International Symposium, MIT Press, pp. 783-799, 1984
- 5) S. Arimoto and F. Miyazaki, "Asymptotic Stability of Feedback Control Laws for Robot Manipulators", Proc. of IFAC Symp. on Robot Control, Barcelona, Spain, 1985
- 6) F. Miyazaki, S. Arimoto, M. Takegaki, and Y. Maeda, "Sensory Feedback Based on the Artificial Potential for Robot Manipulators", Proc. of 9th IFAC, Budapest, Hungary, 1984
- 7) F. Miyazaki and S. Arimoto, "Sensory Feedback for Robot Manipulators", J. of Robotic Systems, Vol. 2, No. 1, pp. 53-71, 1985
- 8) 高瀬国克, "マニピュレータの運動成分の一般的分解とその制御", 計測自動制御学会論文集, Vol. 12, No. 3, pp. 300-306, 1976
- 9) R. P. Paul, "Robot Manipulators", The MIT Press, 1981
- 10) P. Vukobratovic, "Scientific Fundamentals of Robotics 1 and 2", Springer-Verlag, 1982
- 11) J. Y. S. Luh, M. W. Walker and R. P. Paul, "On-line Computational Scheme for Mechanical Manipulators", of ASME, J. of DSMC, Vol. 102, pp. 69-76, 1980
- 12) J. M. Hollerbach, "A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity, IEEE Trans. on Syst. Man Cybern., Vol. SMC-10, pp. 730-736, 1980
- 13) J. Y. S. Luh and Y. F. Zheng, "Computation of Input Generalized Forces for Robots with Closed Kinematic Chain Mechanisms", IEEE J. of Robotics and Automation, Vol. RA-1, No. 2, pp. 95-103, 1985
- 14) T. Kanade, P. K. Khosla, and N. Tanaka, "Real Time Control of CMU Direct-Drive Arm II Using Customized Inverse Dynamics", Proc. of 23rd IEEE Conference on Decision and Control, Las Vegas, NV, pp. 1345-1352, 1984
- 15) 内山勝, "試行による人工の手の高速パターン形成", 計測自動制御学会論文集, Vol. 14, No. 6, pp. 94-100, 1978
- 16) S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering Operation of Robots by Learning", J. of Robotics Systems, Vol. 1, No. 2, pp. 123-140, 1984
- 17) ibid., "Can Mechanical Robots Learn by Themselves?", In H. Hanafusa and H. Inoue (ed.), "Robotics Research: The Second International Symposium", MIT Press, pp. 127-134, 1985
- 18) ibid., "Bettering Operation of Dynamic Systems by Learning: A New Control Theory for Servomechanism or Mechatronics Systems", Proc. of 23rd IEEE Conference on Decision and Control, Las Vegas, pp. 1064-1069, 1984
- 19) S. Kawamura, F. Miyazaki, and S. Arimoto, "Iterative Learning Control for Robotic Systems", Proc. of IECON '84, Tokyo, pp. 393-398, 1984
- 20) ibid., "Hybrid Position/Force Control of Manipulators Based on Learning Method", Proc. of ICAR, '84, Tokyo, pp. 235-242, 1985
- 21) ibid., "Applications of Learning Methods for Dynamic Controls of Robot Manipulators", Proc. of 24th IEEE CDC, Fort Lauderdale pp. 1381-1386, 1985
- 22) S. Arimoto, S. Kawamura, F. Miyazaki, and S. Tamaki, "Learning Control Theory for Dynamical Systems", ibid., pp. 1375-1380, 1985
- 23) M. Togai and O. Yamano, "Analysis and Design of an Optimal Learning Control Scheme for Industrial Robots: A Discrete System Approach", ibid., pp.

- 1399-1404, 1985
- 24) T.Mita and E.Kato, "Iterative Control and Its Application to Motion Control of Robot Arm—A Direct Approach to Servo-Problems", *ibid.*, pp. 1393-1398, 1985
 - 25) C.G. Atkeson and J.McIntyre, "Robot Trajectory Learning Through Practice", *Proc. of IEEE Conference on Robotics and Automation*, San Francisco, pp.1737-1742, 1985
 - 26) S. Arimoto, S. Kawamura, and F. Miyazaki, "Convergence, Stability, and Robustness of Learning Control Scheme for Robot Manipulators", to be published in *Proc. of Int. Symp. on Robot Manipulators: Modeling, Control and Education*, Albuquerque, NM, Nov. 1986
 - 27) S. Dubowsky and D. T. Desforges, "The Application of Model Reference Adaptive Control to Robotic Manipulators", *Trans. ASME, J. of DSMC*, Vol. 101, pp. 193-200, 1979
 - 28) M. Takegaki, and S. Arimoto, "An Adaptive Trajectory Control of Manipulators", *International J. of Control*, Vol.34, No.7, pp. 219-230, 1981
 - 29) A. Balestrino, G. De Maria, and L. Sciavicco, "An Adaptive Model Following Control for Robot Manipulators", *Trans. ASME, J. of DSMC* vol. 105, pp.143-151, 1983
 - 30) A. J. Koivo and T. H. Guo, "Adaptive Linear Controller for Robotic Manipulators", *IEEE Trans. on Automatic Control*, Vol.1. AC-28, No.2, pp. 162-171, 1983
 - 31) F. Miyazaki and S. Arimoto, "Stability of PD Feedback Control for Robot Manipulators", *Proc. of International Sessions in SICE '86*, Tokyo, July 1986

有本 卓 (Suguru ARIMOTO)

昭和 11 年 8 月 3 日生れ。34 年京大理学部卒。沖電気工業 K. K., 東大工学部助手, 講師を経て 43 年大阪大学助教授, 基礎工学部機械工学科に勤務。48 年同教授。工学博士。ロボティックス, デジタル信号処理, 制御理論等の研究教育に従事。日本機械学会, 計測自動制御学会, 情報処理学会, 等の各会員。IEEE の Fellow 会員。
(日本ロボット学会正会員)