

ソフトウェア・サーボによる作業向き自由度の構成

電子技術総合研究所 井上博允・高瀬因克

1 ま え が き

マニピュレータは、当初、放射性物質等を遠隔操作で取扱うことを目的として開発された。マスター・スレイブ方式で、人間が操作する主動端と実際に作業を行なう従動端との空間的対応が実現され、双動型サーボ機構で作業中の力の反射が実現されて、人間の遠隔操作は改善された。1961年 Ernstが発表したマニピュレータの計算機制御の研究は、発展途上にあつた AI のひとつのトピックスとして受けつがれ、Computer Vision の研究の進展と共に、Hand-Eyeシステムへと発展していった。この間、計算機で制御されるマニピュレータにも力の感覚が導入されて、組立作業等にも実行できるようになった。これらのロボットの技術が最も積極的に応用されつつある分野がオートメーションである。記憶装置だけでもつ繰返し形の産業ロボットは、すでに、スポット溶接・プレス作業等に数多く使用され、最近では、簡単な視覚や触覚を備え、ミニコンピュータで制御される高度の産業ロボットの開発例もめづらしくない。機械の腕のもうひとつの分野が義手である。ここでは生体信号を用いて制御することが主題となっている。

ところで、ロボットの応用の場は、オートメーションだけに限定されるわけではない。原子力・宇宙・海洋をはじめとする、人間の直接作業が適さない危険な環境でこそ、人間に代って、人間と同じように器用な作業を行なわせることが期待されている。知的テレオペレータとでも呼ぶべき、この種のロボットは、あるときは自動的に、またあるときは人間との種々のコミュニケーションを通じて、装置の組立て、監視、保守、修理、汚染物質の取扱い、災害復旧作業など、実に多様な仕事の遂行が要求されることとなる。

オートメーションに於ても、知的テレオペレータにしても、バラエティに富む作業をうまく実行するためには、基本的な機能として、精密な位置決めだけでなく、力を制御する能力が必要であるということが、多くの報告で指摘されている。本稿では、マニピュレータの制御問題を、制御変数の一般的な変換としてとらえ、ソフトウェア・サーボによって作業向きを自由度を構成し、腕の運動を分解して制御する方法について述べる。

2 マニピュレータの数式モデル

数式モデルを作る際に対象とした腕は、Fig. 1 に示す b 自由度の関節形のものである。腕は b 個のリンクから構成されており、各関節の回転角を θ_i ($i=1\sim b$) で表す。関節の回転方向を単位ベクトル p_i で表す。 C_i はリンク i の重心に固定された直交座標系である。その z 軸方向は、 $\theta_1\sim\theta_b$ を 0 としたときに、絶対座標 C_0 の方向と一致するようにとる。 C_i の z 軸方向の単位ベクトルを e_{i1} , e_{i2} , e_{i3} で表すと、Fig. 1 の腕では、 e_{i3} は関節 i と関節 $i+1$ とを結ぶ線と方向が一致する。関節の回転方向は、 $p_i = e_{i3}$ ($i=1,4$), $p_i = e_{i2}$ ($i=2,3,5$)
 $p_i = e_{i1}$ ($i=6$) である。

腕の各部の姿勢は、座標系 C_i の3軸方向の単位ベクトルを並べて得られる行列 (e_{i1}, e_{i2}, e_{i3}) で表わされる。隣接する2つの座標系の関係は A_i を座標変換行列として

$(e_{i1}, e_{i2}, e_{i3}) = (e_{i-1,1}, e_{i-1,2}, e_{i-1,3}) \cdot A_i$ となる。したがって i 番目のリンクの姿勢は次式で求められる。

$$(e_{i1}, e_{i2}, e_{i3}) = A_1 \cdot A_2 \cdots A_i \quad (1)$$

関節 j からリンク i の重心へ至る距離ベクトル \mathcal{X}_{ji} は次式で表わされる。

$$\mathcal{X}_{ji} = -\sum_{k=j}^{i-1} l_k \cdot e_{k3} - \sum_{p=1}^3 l_{ip} \cdot e_{ip} \quad (2)$$

ここで、 l_k はリンク k の長さ、 l_{ip} は C_i における関節 i の座標である。

一般に、リンク i 上の点の位置ベクトルは $\mathcal{X}(x_{i1}, x_{i2}, x_{i3}) = \mathcal{X}_{i1} + \sum_{p=1}^3 x_{ip} \cdot e_{ip} \cdots (3)$ である。ただし x_{ip} は座標系 C_i からみたその点の座標である。

リンク i 上の点の速度は (3) 式を時間で微分することにより得られる。

$$\mathcal{V} = \frac{d\mathcal{X}}{dt} = \frac{d}{dt} \left(\mathcal{X}_{i1} + \sum_{p=1}^3 x_{ip} e_{ip} \right) = \sum_{j=1}^i \frac{\partial}{\partial \theta_j} \left(\mathcal{X}_{i1} + \sum_{p=1}^3 x_{ip} e_{ip} \right) \dot{\theta}_j$$

ここで、関節 j の回転方向を示す単位ベクトルを \mathcal{P}_j とし、 $j \leq i$ のとき、

$\partial e_{ip} / \partial \theta_j = \mathcal{P}_j \times e_{ip}$ 、かつ $j > i$ のとき $\partial e_{ip} / \partial \theta_j = 0$ であることを考慮すれば、

$$\mathcal{V} = \sum_{j=1}^i \mathcal{P}_j \times \left(\mathcal{X}_{ji} + \sum_{p=1}^3 x_{ip} e_{ip} \right) \cdot \dot{\theta}_j \quad \text{----- (4)}$$

となる。加速度 \mathcal{A} は、 \mathcal{V} を同様に微分して求める。

$$\mathcal{A} = \frac{d\mathcal{V}}{dt} = \sum_{k=1}^i \left\{ \mathcal{P}_k \times \left(\mathcal{X}_{ki} + \sum_{p=1}^3 x_{ip} e_{ip} \right) \right\} \ddot{\theta}_k + \sum_{k=1}^i \sum_{j=1}^i u_{kj} \dot{\theta}_k \dot{\theta}_j \quad \text{----- (5)}$$

ただし $k \geq j$ のとき $u_{kj} = \mathcal{P}_j \times \left\{ \mathcal{P}_k \times \left(\mathcal{X}_{ki} + \sum_{p=1}^3 x_{ip} e_{ip} \right) \right\}$ 、 $k < j$ のとき $u_{kj} = u_{jk}$ である。関節 n におけるモーメントの回転方向成分は関節トルク T_n とつりあうから、

$$T_n = \sum_{i=n}^b \left[\sum_{j=1}^i \left\{ (\mathcal{P}_n \times \mathcal{X}_{ni}) + \sum_{q=1}^3 x_{iq} (\mathcal{P}_n \times e_{iq}) \right\} (\mathcal{A} - \mathcal{G}) \cdot d\mathbf{m} \right] - (\mathcal{X}_n \times F_c) \cdot \mathcal{P}_n - M_c \cdot \mathcal{P}_n$$

ただし、 \mathcal{X}_n は関節 n から手の基準点への距離ベクトル、 F_c, M_c は基準点に加わる外力およびモーメント、 \mathcal{G} は重力ベクトルである。上式に \mathcal{A} を代入し積分を求めて式の変形を行えば、運動方程式は次式のように表わされる。

$$T_n = \sum_{k=1}^b a_{nk} \ddot{\theta}_k + \sum_{k=1}^b \sum_{j=1}^b b_{nj,k} \dot{\theta}_j \dot{\theta}_k + T_{gn} + T_{Fn} + T_{Mn} \quad \text{----- (6)}$$

ただし $a_{nk} = \sum_{i=r}^b \left\{ m_i (\mathcal{P}_n \times \mathcal{X}_{ni}) \cdot (\mathcal{P}_k \times \mathcal{X}_{ki}) + \sum_{q=1}^3 \sum_{p=1}^3 I_{ipq} (\mathcal{P}_n \times e_{iq}) \cdot (\mathcal{P}_k \times e_{ip}) \right\}$

$$b_{nj,k} = \sum_{i=r}^b \left[m_i (\mathcal{P}_n \times \mathcal{X}_{ni}) \cdot \left\{ \mathcal{P}_j \times (\mathcal{P}_k \times \mathcal{X}_{ki}) \right\} + \sum_{q=1}^3 \sum_{p=1}^3 I_{ipq} (\mathcal{P}_n \times e_{iq}) \cdot \left\{ \mathcal{P}_j \times (\mathcal{P}_k \times e_{ip}) \right\} \right] \quad (k < j \text{ のとき})$$

$$b_{nj,k} = b_{nkj} \quad (k < j \text{ のとき})$$

$$T_{gn} = -\sum_{i=n}^b (\mathcal{X}_{ni} \times m_i \mathcal{G}) \cdot \mathcal{P}_n, \quad T_{Fn} = -(\mathcal{X}_n \times F_c) \cdot \mathcal{P}_n, \quad T_{Mn} = -M_c \cdot \mathcal{P}_n$$

ここで m_k はリンク k の質量、 $I_{ipq} = \int x_{ip} x_{iq} dm$ はリンク i の慣性乗積、記号 r は $\max(n, k)$ を意味する。

$$(6) \text{ 式をベクトルの形式で表わすと } T = A \ddot{\theta} + \dot{\theta}^T B \dot{\theta} + T_c \quad \text{----- (7)}$$

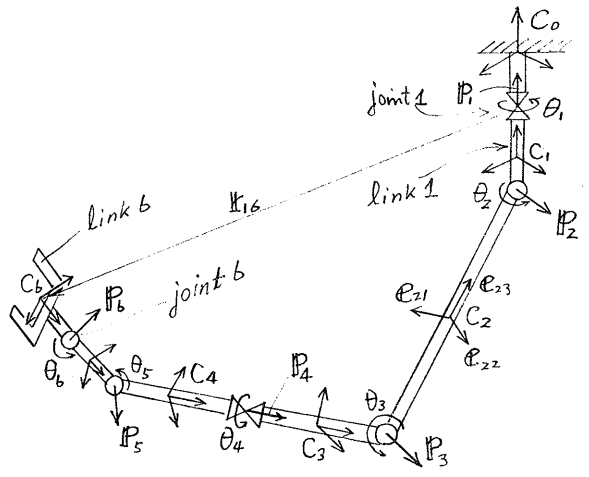


Fig.1 マニピュレータの座標系

となる。ここで $\boldsymbol{\tau}$ と $\boldsymbol{\theta}$ は、それぞれ、関節トルクおよび関節角を示すベクトルで $\boldsymbol{\tau} = (\tau_1, \tau_2, \dots, \tau_6)^T$, $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_6)^T$ である。A は a_{nk} を要素とする行列、B は $b_{ijk} = (b_{1jk}, b_{2jk}, \dots, b_{6jk})^T$ を要素とする行列である。 $\boldsymbol{\tau}_c$ は外力によるトルクで、 $\boldsymbol{\tau}_c = (\tau_{g1} + \tau_{f1} + \tau_{m1}, \tau_{g2} + \tau_{f2} + \tau_{m2}, \dots, \tau_{g6} + \tau_{f6} + \tau_{m6})^T$ である。

3 カベクトルを用いた作業の記述と制御

マニピュレータの役割は、物体を掴んで操作することである。三次元空間内の物体の並進運動と回転運動は、物体に加わる力とモーメントによって定まる。マニピュレータは、目的の作業を行なうために、物体に加える力やモーメントを制御すればよい。本節では、カベクトルを制御してマニピュレータに作業を実行させる方法について述べる。

マニピュレータの手の規準点に、静的な力 \boldsymbol{F} を発生させるためには、各関節に $\tau_{fn} = (\boldsymbol{v}_n \times \boldsymbol{F}) \cdot \boldsymbol{p}_n$ ($n=1 \sim 6$) で定まるトルクを発生させなければならない。また、マニピュレータ自体の重力の影響を相殺するためには、各々の関節トルクに $\tau_{gn} = \sum_{i=1}^n (\boldsymbol{v}_{ni} \times m_i \boldsymbol{g}) \cdot \boldsymbol{p}_n$ ($n=1 \sim 6$) なる重力補償トルクを加えてなければならない。即ち、各関節の駆動トルクを $\tau_n = \tau_{fn} + \tau_{gn}$ となるように制御すれば、手の規準点に力 \boldsymbol{F} が加わったときと同じ効果をもつ。この直接的な方法では、作業の記述を直感的に容易に行なうことができる。以下、いくつかの例を示す。

三次元空間内に任意の平面を与え、手が常にこの平面に乗るように制御することを考える。平面は、点 \boldsymbol{P}_a と法線ベクトル \boldsymbol{N}_a を用いて表わされる。手の基準点 \boldsymbol{P}_h と平面との距離に比例する拘束力 $\boldsymbol{F}_a = -k \cdot (\boldsymbol{P}_h - \boldsymbol{P}_a) \cdot \boldsymbol{N}_a$ を \boldsymbol{P}_h に作用させれば、手は与えられた平面に拘束され、常にこの平面内に保たれるようにフィードバック制御される。二平面を用いて手を拘束制御すれば、手は二平面の交線上に拘束され、三平面による拘束で手は一点に位置決めされる。黒板を消す場合、サンドペーパーをかける場合の如く、一定力 f_p で対象物表面に押しつけながら手を動かすことが要請される場合には、対象物表面の法線ベクトルを \boldsymbol{N}_0 として、 $\boldsymbol{F} = -f_p \boldsymbol{N}_0$ なる力を加えることで、これらの拘束は実現される。同様に、直定規で線をひく動作は、紙面および定規端面の法線ベクトルを $\boldsymbol{N}_1, \boldsymbol{N}_2$ として、 $\boldsymbol{F}_1 = -f_1 \boldsymbol{N}_1, \boldsymbol{F}_2 = -f_2 \boldsymbol{N}_2$ の合成が拘束条件を満足させる。

対象物から受ける拘束に従いつつそれを駆動する動作の代表的な例としてクランクの回転を考える。クランクを回すためには、常に、クランクが定める円軌道の接線方向に力を加える必要がある。 f_t を接線力の大きさ、 \boldsymbol{R} を回転方向を示す単位ベクトル、 \boldsymbol{P}_c をクランクの中心位置として、クランクを回すための力 \boldsymbol{F}_t は次式で与えられる。

$$\boldsymbol{F}_t = f_t \cdot \boldsymbol{R} \times (\boldsymbol{P}_h - \boldsymbol{P}_c) / |\boldsymbol{P}_h - \boldsymbol{P}_c|$$

マニピュレータ関節角の読みから \boldsymbol{P}_h が計算され、上式により、

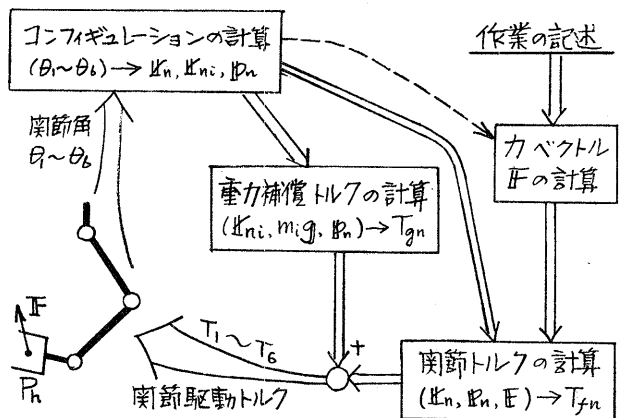


Fig. 2 カベクトル制御の概略ブロック図

クランクを回転させるための接線力 F_t が計算される。 F_t は各関節のトルクに分解され、重カ補償トルクと加算されてマニピュレータを駆動し、クランクをスムーズに回転する。

我々が実験に使用しているマニピュレータは、アクチュエータとして磁気粉体クラッチを用いたもので、全ての関節駆動トルクは、D/A変換器を介して計算機で制御される。Fig. 2 に本節で述べたカベクトル制御方式の概略ブロック線図を示す。なお作業例については映画で紹介する予定である

4. 作業自由度の分解と制御

前節では、力の静的平衡だけを考えて、手の並進運動に関するカベクトルの制御法を示した。本節では、並進運動および回転運動に対し、腕の動的特性をも考慮に入れて、運動自由度の分解とその制御法について一般的に考察する。各自由度成分に対応する制御変数をH変数と呼ぶことにし、 $ld = (h_1, h_2, \dots, h_6)^T$ で表わす。自由度の構造は固定して考えず、作業毎にその幾何学的構造に適合するものを適宜選ぶこととする。

4.1 運動の分解と制御変数 並進運動においては、直交座標、円筒座標、極座標などが、動作の幾何学的構造と適合する場合が多い。例として、ある直交座標 Cx を選ぶ。 Cx からみた手の基準点の座標をH変数 $(h_1, h_2, h_3)^T$ として選んでみよう。 d_i を絶対座標における Cx の原点の位置ベクトル、 $M = (e_{t1}, e_{t2}, e_{t3})^T$ 、 x_i を関節 i から手の基準点への距離ベクトルとすれば、

$$\begin{aligned} (h_1, h_2, h_3)^T &= M \cdot (x_i - d_i) \\ (\dot{h}_1, \dot{h}_2, \dot{h}_3)^T &= M \cdot \left\{ \sum_{j=1}^3 (P_j \times x_j) \cdot \dot{\theta}_j \right\} \triangleq J_1 \dot{\theta} \\ (\ddot{h}_1, \ddot{h}_2, \ddot{h}_3)^T &= J_1 \ddot{\theta} + \dot{\theta}^T C_1 \dot{\theta} \end{aligned} \quad \text{----- (8)}$$

と表わすことができる。ここで C_1 は $C_{1ij} = M \{ P_i \times (P_j \times x_j) \}$ ($i \leq j$) を要素とする 6 次の対称行列である。

回転運動に対しては、運動の全ての範囲で回転を人間の直感とうまく合うようにわかりやすく表現しうるH変数を見つけることはむづかしい。しかし、実際の作業では、(i) 主軸の方向と主軸まわりの回転が同時に変化するが変化量が小さい回転運動。(ii) 主軸まわりの大きい回転運動、の2つの場合を考えれば十分であろう。(i) の場合には、手に主軸と側軸を表わす単位ベクトル O_p, O_s をとりつけて考え、回転を測る際の基準となる単位ベクトル a_1, a_2, a_3 を作業系に仮定して、H変数を $h_4 = O_p \cdot a_1, h_5 = O_p \cdot a_2, h_6 = O_s \cdot a_3$ のように選ぶは、

$$(\dot{h}_4, \dot{h}_5, \dot{h}_6)^T = J_2 \cdot \dot{\theta}, \quad (\ddot{h}_4, \ddot{h}_5, \ddot{h}_6)^T = J_2 \cdot \ddot{\theta} + \dot{\theta}^T C_2 \dot{\theta} \quad \text{----- (9)}$$

の形で表わすことができる。

$$\text{ただし } J_2 = \begin{pmatrix} (P_1 \times O_p) \cdot a_1, \dots, (P_6 \times O_p) \cdot a_1 \\ (P_1 \times O_p) \cdot a_2, \dots, (P_6 \times O_p) \cdot a_2 \\ (P_1 \times O_s) \cdot a_3, \dots, (P_6 \times O_s) \cdot a_3 \end{pmatrix} \quad C_2 \text{ は } C_{2ij} = \begin{pmatrix} \{ P_i \times (P_j \times O_p) \} \cdot a_1 \\ \{ P_i \times (P_j \times O_p) \} \cdot a_2 \\ \{ P_i \times (P_j \times O_s) \} \cdot a_3 \end{pmatrix}$$

を要素とする 6 x 6 の行列である。(ii) の場合にも、 $h_4 = O_p \cdot a_1, h_5 = O_p \cdot a_2, h_6 = \tan^{-1}(O_s \cdot a_2 / O_s \cdot a_1)$ のように選ぶは、(i) と同様な表現を得る。

4.2 サーボ H変数と関節角との関係は (8), (9) 式をまとめることにより次式のように表わすことができる。 $\ddot{ld} = J \ddot{\theta} + \dot{\theta}^T C \dot{\theta}$ ----- (10)

ここで、 J は 3×6 の行列 J_1, J_2 を重ねて作った 6×6 の行列で $J = \begin{pmatrix} J_1 \\ J_2 \end{pmatrix}$ である。 C は 6 次のベクトルを要素とする 6×6 の対称行列 $[C_{ij}]$ で $C_{ij} = \begin{pmatrix} C_{1ij} \\ C_{2ij} \end{pmatrix}$ である。(10)式と(7)

式から $\ddot{\theta}$ を消去すれば、

$$\tau = AJ^{-1}\ddot{d} + \dot{\theta}^T D \dot{\theta} + \tau_c \quad \dots(11)$$

となる。ここで D は 6×6 の対称行列 $[d_{ij}]$ で $d_{ij} = -AJ^{-1}C_{ij} + B_{ij}$ である。

腕の関節角および角速度を測定し $\dot{\theta}^T D \dot{\theta}$ を計算して各関節トルクに加算する補償ループを導入すれば、(11)式は、 $\tau = AJ^{-1}\ddot{d} + \tau_c$ となる。重力等の外力を補償すれば、更に簡単な式 $\tau = AJ^{-1}\ddot{d}$ で表わされる。

H変数に加える制御を見通しのためにより形にするために $\tau = AJ^{-1}\mathcal{F}$ で定義される操作量 \mathcal{F} を導入すれば、 $AJ^{-1}\mathcal{F} = AJ^{-1}\ddot{d} \rightarrow \mathcal{F} = \ddot{d}$ ----(12) となる。 \mathcal{F} は H変数に関する正規化された加速度としての意味をもつ。(12)式は各成分にわけて表わせば、 $f_i = \ddot{h}_i$ ($i=1 \sim 6$) となる。上述のように変換された制御変数に対するサーボのブロック線図を Fig. 3 に示す。

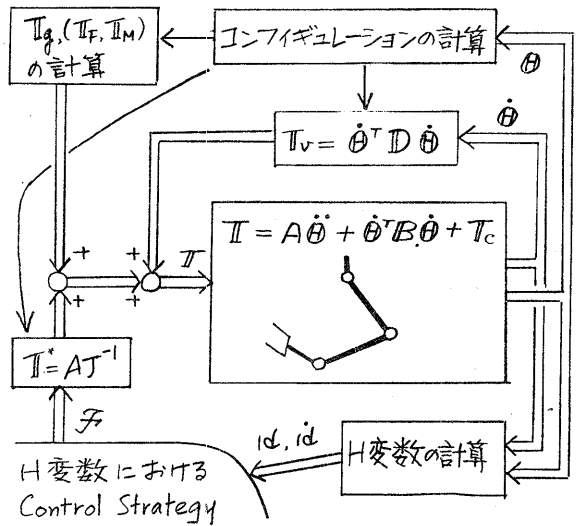


Fig. 3 サーボ系の構成

4.3 変換された自由度の制御方策

正規化された加速度 \mathcal{F} を決めれば、 $\tau = AJ^{-1}\mathcal{F}$ により関節駆動トルクが求められ、手が制御される。(12)式は、成分毎の関係式で表わせば、 $f_i = \ddot{h}_i$ であるから、各自由度毎に制御を考えればよい。制御方策としては作業の性質にあわせて色々なものが考えられるが、以下によく使用される代表的な制御方策をあげよう。

(a) 束縛制御 i 番目の自由度が、常に一定値 h_{ir} となるように制御するためには、例之ば、次のような操作力 f_i を加えればよい。

$$f_i = -k_v \dot{h}_i - k_e (h_i - h_{ir}) \quad \text{ここで、} h_i \text{ および } \dot{h}_i \text{ は測定量 } \theta, \dot{\theta} \text{ から計算でき、したがって } f_i \text{ も計算で求まる。}$$

$f_i = \ddot{h}_i$ だから、この自由度の制御は $\ddot{h}_i + k_v \dot{h}_i + k_e h_i = k_e h_{ir}$ となる。このように h_i に定値制御を加えることは、 i 番目の自由度に束縛を加える効果をもつ。並進運動の H変数として直交座標が選ばれている場合、1自由度の定値制御は平面への束縛を意味する。2自由度に定値制御を施せば、2つの束縛面の交線上に束縛されることになる。

(b) 加速度制御 i 番目の自由度に一定の力 f_c を加えると $f_c = \ddot{h}_i$ からこの自由度は等加速度運動を行なうことがわかる。次に $-a\dot{h}_i$ の速度抵抗を追加し、 $f_i = f_c - a\dot{h}_i$ として制御すれば、この自由度は速度が一次遅れのカーブを描いて f_c/a に収束する運動を行なう。この様な加速度制御は、力に基づいて運動を制御するから、対象物から加えられる現実の束縛に対する適応性もち、物体による案内作用を利用して手の位置決めを行わせる際に効果的である。

(c) トラジエクトリ制御 H変数の目標値が隣りの関節で与えられる場合、即ち、 i 番目の自由度を $h_{ir}(t)$ に追従させる制御を考える。 $h_{ir}(t)$ がわかっている

る場合には、 $f_i = \ddot{h}_{ir}(t) - k_v(\dot{h}_i - \dot{h}_{ir}(t)) - k_e(h_i - h_{ir}(t))$ という操作力を加えればよい。ただし $\ddot{h}_{ir}(t)$, $\dot{h}_{ir}(t)$ は $h_{ir}(t)$ から前以って計算で求めておくものとする。この操作力により、良好な追従制御が実現される。

以上、各自由度成分の制御について代表的な方策を示したが、実際に手に作業を行なわせる際には、作業に適したH変数を選び、各自由度毎に適切な制御方策を組み合わせて用いる。4.2節で述べたソフトウェアサーボによって、腕は見かけ上、H変数と対応する自由度構成をもっている様に制御される。即ち、手の機械的構造から解放されて、目的とする作業向きに腕の自由度を適宜組替えることができるわけである。

5. お ち ぎ

本稿では、マニピュレータの数式モデル、カベクトル制御方式、作業向き自由度の構成とその制御について述べた。

マニピュレータに実行させたい作業は多岐にわたる。ひとつひとつの小作業は、割合単純な幾何学的構造をもっているが、作業によって効果的な自由度の構成は異なる。作業に適合した自由度構成を選べば制御もまた容易になる。しかし、マニピュレータの機械的構造は一定であり、作業の構造と適合させ得ない場合がある。そこで、本稿では、マニピュレータの自由度構成を、ソフトウェアサーボで見かけ上組み替えてしまうことを試みた。

3節で述べたカベクトル制御法は、マニピュレータの機械的構造を意識せずに、運動をカベクトルで規定しようという考えに基づいている。手は任意のカベクトルを等価的に発生することができるから、カベクトルの重ね合わせで容易に作業を記述し、実行させることができる。この方法は、対象物から受ける物理的拘束に適応しつづ手を動かすことを極めて容易にする。4節では、手の並進および回転運動を、何々の作業向きに選んだ座標系を用いて手の自由度を規定し、各自由度成分ごとに、適当な制御方策を組合わせて作業を実行させる方法について述べた。この方法を厳密に実施するためには、大量の数値計算が必要であるが、LSIなどの進歩を考え合わせると十分な実施可能性をもつと考えられる。なお、実行すべき作業が予め計画できる場合には、計算の一部を前以って行なっておくことにより実時間計算を大いに減少させることができる。

参 考 文 献

- 1) R. Paul: Modelling, Trajectory Calculation and Servoing of a Computer Controlled Arm, Stanford Univ. A.I. Lab. AIM-177 (1972)
- 2) Nevins et. al.: Report on Advanced Automation, C.S. Draper Lab. R-764 (1973)
- 3) R. Finkel, et. al.: AL, A Programming System for Automation, Stanford Univ. AIM-243 (1974)
- 4) 黒野: 一人の人工の手の協調制御, 日本機械学会誌, Vol.78 No.682 (1975-9)
- 5) 高瀬, 井上, 他: トルク制御機能をもつマニピュレータ, 電総研彙報, Vol.37, No.3 (1973-3)
- 6) 高瀬, 井上: 環境より拘束を受ける系におけるロボット腕の制御, 電気学会全国大会 #1632 (1974)
- 7) 高瀬, 井上: マニピュレータの力制御, 第14回 SIC 工学術講演会 #157 (1975)