

知能化作業支援システム

Intelligent Assisting System—IAS

橋 本 秀 紀*・ブス マーチン*
Hideki HASHIMOTO and Martin BUSS

本解説では、「機械システム（ロボット）は人間とインタラクティブに協調することにより人間の知的能力（思考、推論、判断、認識等）を十分に引き出し高度な作業を行う」ということを前提に機械の知能化を考える。IAS は人間の知的な作業を支援することを目的としており、スキルデータベースを用いて作業の物理的な部分を自動化する能力を持つ。オペレータがセンサグローブを装着し、仮想世界の物体をつかんだり運んだりすることにより高度運動機能（スキル）を計測し解析することが可能となる。解析された高度運動機能は数理モデルで表現されスキルデータベースとして獲得される。さらに、数理表現されたスキルをロボットハンドに実現させるための制御アルゴリズムを提案する。

1. はじめに

著者らは「知能化作業システム」(Intelligent Assisting System—IAS) を提案しており^{1)~6)}、「機械システム（ロボット）は人間とインタラクションすることにより人間の知的能力（思考、推論、判断、認識等）を十分に利用する」ということを前提に機械の知能化を考えている。人間が機械を用いて作業を行うためお互いに協調しなければならず、両者をつなぐヒューマン・マシン・インターフェイスが重要な位置を占め、最も効果的な手法としての人工現実感技術が大きな役割を果たす^{7)~9)}。IAS は人間の知的な作業を支援することを目的としており、作業の物理的な部分（物を運ぶ、動かす等の動作）に関してスキルデータベースを利用することによりその作業を人間と協調して部分的に自動化して行う能力を持つ。

本稿では、最初に「知能化作業支援システム」の基本的な考え方を示す。次に IAS 実現の中心である高度運動機能獲得・移植システム、仮想現実感シミュレータおよび仮想世界の物体との力関係を計算するためのダイナミックフォースシミュレータの実現方法に関して述べる。高度運動機能獲得システムではオペレータがセンサグローブを装着し、仮想世界の物体をつかんだり運んだりすることによりその作業動作が測定可能となる。このとき、測定された複雑な作業動作（高度運動機能）を表現するための数理モデルが必要となる。著者らは力の釣り合いから定義されるグリップ変換行列のダイナミクスが作業スキルの本質であると考え、その数式的な導出を示す。さらに、このモデルによって表現されたスキルを

ロボットハンドに実現させるための制御アルゴリズムを提案する。

2. 知能化作業支援システムの提案

ロボット研究の主流は人間の知的な能力までを代替する知能ロボット研究開発へと進んできた。AI 等に代表される知識情報処理がロボットの複雑なタスク実現のために研究され応用されたが、やがてその限界が見えてきた。大規模なシステムの計画や運行といった多大なデータ処理および静的な組み合わせを対象とするには有効であるが、ロボットのように外界と情報のみならず実時間で動的な力のやりとりをも有する系では、対象として複雑すぎて現段階での AI 技術は難しく、近い将来にブレークスルーを引き起こすかどうか疑問である。

本稿で提案する知能化作業支援システムにおいては、機械システムは人間の一部の機能の増幅、延長でしかない位置づけることにより、いわゆる「知能化」の呪縛から解放される。自動車は人間の「歩く」（移動）という機能の延長線上（歩行ではなく車輪による移動）にある。コンピュータも「記憶する」、「計算する」といった人間の一部の能力を量的に大きく増幅したものと見える。知能化作業支援システムは、人間の行う作業の支援に当たるものであり、外界との情報および力のやりとりを主に担当する。したがって、いわゆる知能ロボットのように人間に取って代わろうとするものではなく、人間の機能の一部を代価するものであり、Fig. 1 に示すような「人間増幅器」の延長線上にあると考えられる。

人間の労働の本質を考えたとき、その労働にどれだけ知的なものが感じられるかがインセンシビリティの大小につながると考えられる。知的なものは人間にとって

*東京大学生産技術研究所 第3部

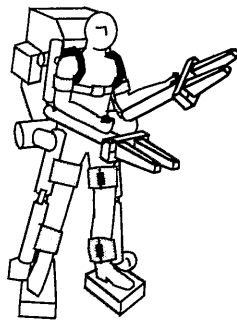


Fig. 1 人間増幅器

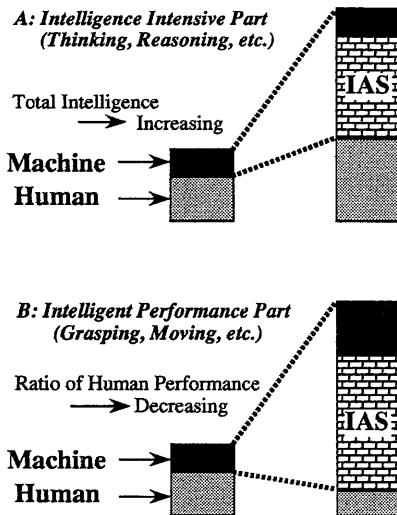


Fig. 2 知能化作業支援システムの概念

楽しいのである。すなわち、労働といった作業の知的な部分はなるべく人間が行い、それ以外の物理的な部分が多くを占める場所は機械システムが支援する。このように、人間がイニシアチブを取りながら知的な部分を受け持ち、機械システムが協調して作業を支援することになる。これは人間から離れていく高度技術の塊である機械システムを道具として位置づけ、しかも使いこなす方向である。以上のことがらを「知能化作業支援システム」として抽象的に表現したのが Fig. 2 である。作業を知的な部分が主である A と物理的な作業を主に含む B に分け IAS を用いることにより全体の作業能率を上げる。A においては人間の比率が高くなり B では低くなる。主張していることはシンプルであるが、意識して議論するかどうかは今後のロボット研究に大きく影響すると思われる。

知能化作業支援システムの一つとして、人間の高度運動機能（作業スキル）を計測し解析することを目的とした研究を進めている（Fig. 3）。人間は未知の環境でも自由に動作の計画を行うが、この時の腕や手や指の具体

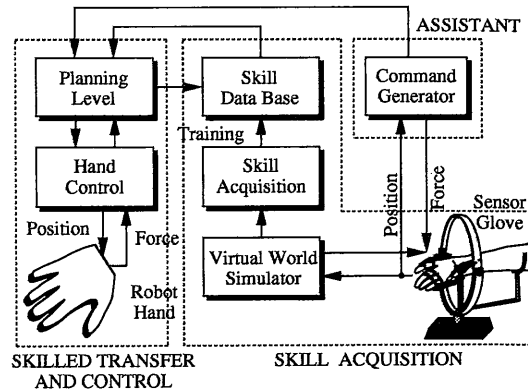


Fig. 3 IASのための高度運動機能獲得システム

的な筋肉の動きは無意識に計画されている。さらに、手をつかんでいる物体の形、大きさ、堅さ、表面摩擦定数等がかなり変化しても目的とする動作をロボットに行うことが容易にできる。この高度運動機能の獲得法を中心に、IAS に関する研究を進めている。

また、高度運動機能の表現のためのモデルができたとなると、従来の学習方法を使ってスキルデータベースに格納することも可能となる。このスキルデータベースを用いることによりオペレータの作業目的をシンボルとして与え、モーションコントロールレベルでの複雑な動作を自動的に行うことが可能となる。すなわち、具体的な動作（シンボル表現）をある程度実現可能な状態（スキル）にした上で、未知なる目的・動作・環境に出合った場合、人間の補助により運動機能の獲得を繰り返しながら、IAS の作業知識（スキル）を増やすことが考えられる。

3. 知能化作業支援システムにおける高度運動機能の実現

人間を支援することが「知能化作業支援システム」の目的であり、利用しやすいヒューマン・マシン・インタフェースで情報伝達および作業に関して人間をサポートする。Fig. 4 に示すように支援は階層化されており、知識（シンボル）、ルール、スキルに分けて考える¹³⁾。上位レベルで、人間が作業および処理の目的をシンボルで与え、システムからアドバイスを受ける。中間レベルでパターンとルールで処理を行う。最下位レベルは作業スキルレベルであり、具体的な高度運動機能を実現する。

スキルレベルが知能化作業支援システムを支える技術として大変重要であり、また人間の持つ高度運動機能の実現が「スキル」の獲得および移植としてロボット研究者に意識されているものである。人間の作業の多くの部分は手によるものである。人間の腕と手は27自由度有ると言われており、機械システムに人間と同様な器用な

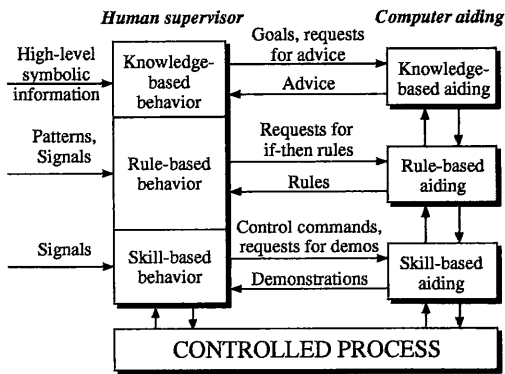


Fig. 4 支援の階層化

作業を行わせるのはハードウェアおよびソフトウェアとも現段階では困難である。しかし、人間の作業を支援することを目的とすると、ハードウェアとしては人間と同じである必要はなく作業目的に応じて必要最小限な構成で十分である。たとえば、建設現場における重量物の運搬を考えるならば、ロボットハンドの自由度としては12自由度（各指4自由度の3指構成）を考えればよい。問題は「物を持つ」という動作をどのように表現して、どのように実現させるかである。人間はまず目で見て、どこを持つかを計画して、手を近づけ接触させて安定にかつ滑らないように力を加減して持つ。また、このとき物体の形、大きさ、硬さ、表面摩擦定数等がかなり変化しても、目的とする動作をロボアストに行うことができる。これを視覚系まで含めてアルゴリズムとして記述するのは困難であり柔軟性も期待できないであろう。

「知能化作業支援システム」では、認識、判断、計画といった部分は人間が主に行い、スキルレベルにおいてロボットハンドは対象物に近づくと安定に把持するように動作する。機械システムの方が人間の知的能力を引き出すために部分的に知能化される（Fig. 3）。たとえば、人間の動作を予感（予測）することにより動作の目的を認識し、ある程度自動的に行わせることが考えられる。機械システムとオペレータの能力を融合させ、物理的作業を行わせ、知的な部分を人間の判断に委ねる形となるであろう。すなわち、人間が制御ループに入り込み、全体の制御システムや作業プロセスの一部となる。

Fig. 3 に示す運動制御レベルでの具体的な IAS ではスキルの獲得（Skill Acquisition）、移植および制御（Skill Transfer and Control）人間への支援（アシスティング、Assistant）の3つのモードがある。

高度運動機能（スキル）獲得：オペレータの手にセンサグローブをはめ込み、仮想現実環境の中で手と物体との力の釣り合いをシミュレーションし、オペレータに物体からの反力（関節トルク）をフィードバックする。作業を行う間に、関節角度・トルク・指先の位置・接触

力・物体に働く外力等の作業パラメータがコンピュータの中に取り込まれる。パラメータの特徴量を抽出することにより、高度運動機能をスキルとしてデータベースに格納する。

スキルの移植および制御：スキルデータベースに格納された高度運動機能を引き出し、ロボットハンドの制御を行う。この時、人間の作業動作と同じ意味を持つ動作を異構造ロボットハンドに実行させる。

アシスティングモード：オペレータが行おうとする作業目的を認識し、人間と協調してそれを実現するために必要なスキルをデータベースから引用する。IAS はアシスティングモードでその作業を少なくとも部分的に自動化する能力を持つ。

3.1 センサグローブシステム

人間の手の動作を解析するために3指10自由度を有するグローブを開発した。これはセンサグローブと呼ぶもので、10自由度のうち手首の回転を除く9自由度に対してワイヤー伝達モータ駆動により力をフィードバックすることができる（Fig. 5）。このサーボ系の周波数帯域は6 [Hz]である。各自由度には指の関節角度を計測するためのポテンシオメータおよび力を測るための歪みゲージを用いた線形性の高いトルクセンサが取り付けられている。このシステムは Fig. 6 に示す仮想現実感シミュレータに組み込まれており、計算機の中にあるロボットハンド（23自由度を有する）と結ばれて関節角度、トルクのデータをやり取りする。通信速度はセンサグローブとワークステーション間で38400 [baud]であり毎秒30枚の画面の更新ができる。また、関節角度情報およびトルク情報はそれぞれ8 [bit]、12 [bit] データで12 [sample/sec]でコンピュータに取り込まれる。Fig. 7 にセンサグローブの写真を示す。

オペレータはこのセンサグローブを介して仮想現実感シミュレータ内の物体をつかむことによって反力を感じることができる。これはシミュレータによって物体に働く外力と内力を計算して、反力としてセンサグローブへ返している。オペレータはセンサグローブからの感覚を

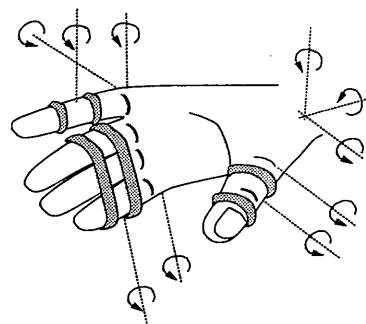


Fig. 5 センサグローブの自由度

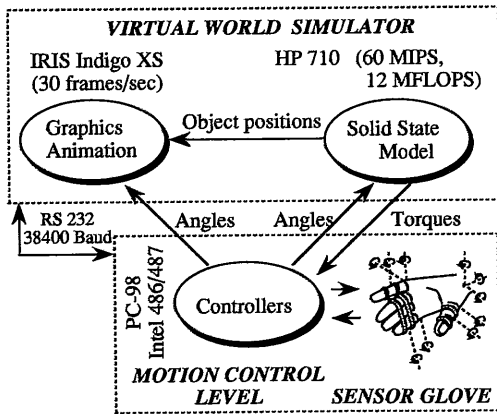


Fig. 6 高度運動機能獲得のための実験システム

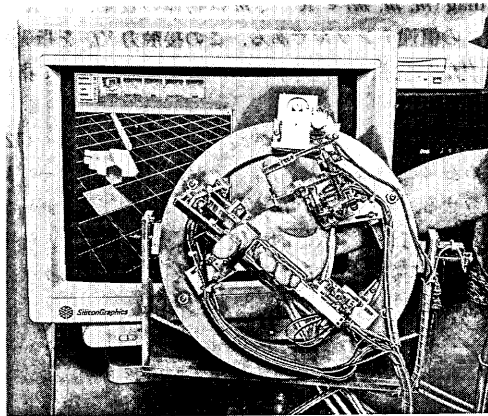
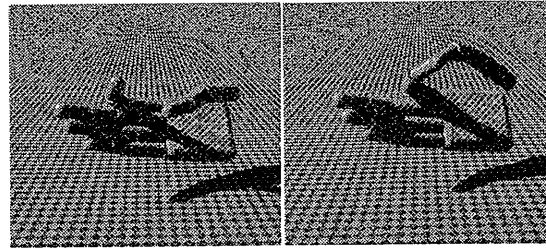


Fig. 7 センサグローブの写真

頼りに作業を行う。このセンサグローブと仮想現実感シミュレータは視覚と力覚の融合に適しており、スキルの解析には強力なツールとなる。現在は、物体把持時の触覚を実現し自由度をさらに増やし両手へ拡張すること、および3次元マウスを併用し高さ方向の自由度を与えることを進めている。

3.2 キャリブレーション問題

スキルの解析を行うためにセンサグローブを用いて仮想現実感環境下で各種データ収集するが、オペレータの手の大きさの違い等によりキャリブレーションが必要となる。また、センサグローブの自由度の合計が10であるのに対して、仮想現実感シミュレータ内のロボットハンドのモデルは23自由度ある。したがって、シミュレータ内で自然な動きを実現するためにセンサグローブからデータを補完してロボットハンドへの入力情報を作る必要がある。また、力のフィードバックでは逆向きの補完が必要となる。著者らはこのキャリブレーションおよび補完にニューラルネットワークを用いて、の動作を自然なものとしている (Fig. 8)。



(a) 関節角度の線形結合 (b) ANNを用いた結果

Fig. 8 キャリブレーション実験結果

一般にセンサ機構とオペレータの手の構造を同一にすることは困難である。ここで開発したセンサグローブの場合、人指し指の関節角度 ψ_1, ψ_2 の2自由度だけが測定可能であり、仮想現実感モデルは3自由度 $\theta_1, \theta_2, \theta_3$ を持つ。すなわち、より正確にオペレータの手の状態を画面に表示し現実感を与えるには、キャリブレーションのための非線形写像を求める必要がある。

$$\Theta = f(\Psi) \tag{1}$$

ここで $\Psi = [\psi_1 \psi_2 \dots \psi_m]^T, \Theta = [\theta_1 \theta_2 \dots \theta_n]^T, m$ はセンサによる入力信号数, n は理論上 (画面上) のモデルの自由度である。

式(1)で表す非線形写像は Artificial Neural Network (ANN) を用いて求められる。ここで利用した ANN は階層形3層ネットワークで隠れ層10個であり、パラメータ調整則にバックプロパゲーションを用いた。学習パターンを作るために、まず人指し指と親指が動くおのおのの平面において、指先の位置と指の根元の関節角度を測定した。測定結果を理論上の指モデルの逆キネマティクスに代入する。

$$\theta_2 = \tan^{-1} \frac{y}{x} - \cos^{-1} \frac{x^2 + y^2 + l_2^2 - l_3^2}{2l_2 \sqrt{x^2 + y^2}} \tag{2}$$

$$\theta_3 = \cos^{-1} \frac{x^2 + y^2 - l_2^2 - l_3^2}{2l_2 l_3} \tag{3}$$

ここで l_i は指のおおのの関節長さ, x, y は平面内での指先位置を表す。センサの検出値が ψ_1, ψ_2 のときにオペレータと理論上の指先位置が同じになる $\theta_1, \theta_2, \theta_3$ を決定する。ANN を約500万回学習した後、オペレータがセンサグローブを装着して、人指し指と親指を接触させることを行う。ANN を用いてキャリブレーションを行った場合のグラフィックス表示を Fig. 8(b) に示す (Fig. 8(a) はセンサの検出値 ψ_1, ψ_2 を線形結合することにより得られた結果である)。ANN をキャリブレーションに用いることで、かなりの改善が見られることがわかる。

3.3 ダイナミクスフォースシミュレータ (DFS)

ダイナミックフォースシミュレータの情報フローを

Fig. 9 に示す. センサグローブの関節角度は毎秒100回サンプリングされ, ワークステーションに送られる. このデータはセンサ信号 $\Psi \in R^{10}$ である. 前述の ANN を用いたキャリブレーション法によって式(1)の非線形写像を実現し, 仮想現実感内のハンドモデルの $\Theta \in R^{23}$ を得る. 次に得られた角度ベクトル Θ を IRIS Indigo に送り, グラフィック表示を行う. また, ハンドの順キネマティックにより, Fig. 10 に示すように指先に張られる座標系とワールド基準座標系との関係は同次変換行列 $T_i, i=3, 7, 11, 15, 19$ で表現される. このとき, 手首の自由度を含めて基準座標系から指の付け根関節への変換行列を $B_i (i=1: 親指 \dots i=5: 小指)$ として, おのおの指先への変換行列 T_i は

$$T_{19} = B_1 A_{16} A_{17} A_{18} A_{19} \quad (4)$$

$$T_3 = B_2 A_4 A_1 A_2 A_3 \quad (5)$$

$$T_7 = B_3 A_4 A_5 A_6 A_7 \quad (6)$$

$$T_{11} = B_4 A_4 A_9 A_{10} A_{11} \quad (7)$$

$$T_{15} = B_5 A_4 A_{13} A_{14} A_{15} \quad (8)$$

となる. ただし,

$$A_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i & 0 & a_i \sin \theta_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

$$A_i^* = \begin{bmatrix} \cos \theta_i & 0 & -\sin \theta_i & 0 \\ \sin \theta_i & 0 & \cos \theta_i & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

ここで, a_i は関節 i の長さを表す. 次に, 指先座標系の原点から仮想世界の物体への距離を計算することにより, ハンドと物体との接触点を求める. 仮想世界の物体の数, おおのこの物体の質量, 位置, 姿勢, 速度および加速度はデータベースにある. 接触点で指先座標系から物体座標系への変換行列, 物体の加速度および角加速度を用いて物体に対するニュートンオイラー方程式

$${}^0f_c = \begin{bmatrix} \dot{m} & 0 \\ 0 & J \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \omega \times \dot{m} v \\ \omega \times J \omega \end{bmatrix} \quad (11)$$

を得て, 指先に働く仮想接触力 0f_c を計算する. ただし, $\dot{m} = \text{diag}(m, m, m) \in R^{3 \times 3}$, m は物体の質量, $J \in R^{3 \times 3}$ は物体の慣性テンソルである. この接触力 0f_c を指先座標系 H_{f_c} に変換し, ハンドヤコビアン J によって指関節トルクを

$$\tau = J^T H_{f_c} \quad (12)$$

と求める. このトルクを目標値としてセンサグローブに実現させる. オペレータは物体の反力を感じる. 同時に物体の微分方程式を数値的に積分し, 物体の位置, 姿勢を Indigo に送り, 仮想世界の状態をグラフィック表示する.

ダイナミックフォースシミュレータの1サイクルは世界の複雑さで決まる. 本稿では対象とする物体の数が少ないのできわめて短い時間で計算が終り, 1サイクルのほとんどの部分がグラフィック表示のために費やされる. IRIS Indigo XS では1秒に30枚のレートでグラフィック表示できるので, リアルタイム性は十分であると考えられる.

4. 高度運動機能モデル

センサグローブを用いて仮想世界で作業をする時に, その作業を高度運動機能としてモデル化することによってスキルの獲得が可能となる. この獲得されたスキルをロボットハンドの知的制御に適応することにより高度運動機能を実現する.

4.1 高度運動機能のモデル

物体に働く一般化外力と, その力を実現する一般化接触力 (レンチ) とはグリッパ変換行列で結ばれる. 著者らは, このグリッパ変換行列のダイナミクスを高度運動機能の抽出モデル (スキル) として考える.

4.1.1 グリッパ変換行列

ある物体を m 本指でつまんだ時に, 一般化接触力 $c_i(t)$ と一般化外力 $f_{r,i}(t)$ との釣り合い関係は

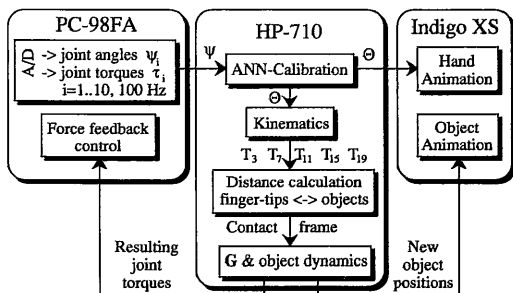


Fig. 9 シミュレータのフローチャート

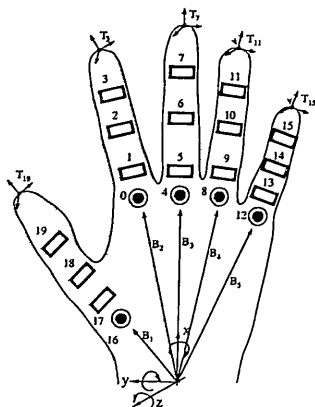


Fig. 10 ハンドのキネマティックモデル

$f_{ext}(t) = W(t)c(t)$ (13)
 となる。ここで一般化外力 $f_{ext}(t) \in R^6$, $c(t) \in R^{n(t)}$ は各接触力の大きさを表し, $W(t) \in R^{6 \times n(t)}$ はその接触力の方向を表す行列とする^{12), 14)}. 行列 $W(t)$, 把握力の大きさ $c(t)$, 外力 $f_{ext}(t)$ は一般に時変であるが, 以下の展開ではすべての時間に渡って $rank(W) = 6$ と仮定する. 式(13)の解であるベクトル $c(t)$ は特解 $c_p(t)$ と斉次解 $c_h(t)$ に分解できる.

$$c(t) = c_p(t) + c_h(t) \quad (14)$$

ここで, $c_p(t) \perp c_h(t)$, $c_h(t) \subset \text{null}(W(t))$ である. $c_p(t)$ は $W(t)$ の右一般化逆行列を用いて

$$c_p(t) = -W^+(t)f(t) \quad (15)$$

と計算できる ($W^+ = W^T[WW^T]^{-1}$). $c_h(t)$ が内力の大きさに対応し, $W(t)$ の $n(t) - 6$ 次元の零化空間を張る $n(t) - 6$ 個の単位直交基底ベクトル $c_{i,h}(t)$ を用いて, 行列 $N(t) \in R^{n(t) \times (n(t)-6)}$ を新たに定義し, 外力および内力をベクトル $F(t)$ とし, 物体の一般速度および内速度 (トウイストが張る空間の零化空間) をベクトル $V(t)$ とする.

$$N(t) = [c_{1,h}(t) \dots c_{n(t)-6,h}(t)] \quad (16)$$

$$F(t) = \begin{bmatrix} f_{ext}(t) \\ f_{int}(t) \end{bmatrix} \quad (17)$$

$$V(t) = \begin{bmatrix} v_{ext}(t) \\ v_{int}(t) \end{bmatrix} \quad (18)$$

$$v_{ext}(t) = [v_x v_y v_z \omega_x \omega_y \omega_z]^T = \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (19)$$

$F(t)$ と接触力との関係は

$$F = (G^T)^{-1}c = \begin{bmatrix} W \\ N^T \end{bmatrix} c \quad (20)$$

$$c = G^T F = [W^+ N] F \quad (21)$$

となり, 行列 $G(t)$ はグリップ変換行列である. 仮想仕事法を用いて物体速度 $V(t)$ と接触点速度 (トウイスト) との関係は

$$d = G^{-1}V = [W^T N] V \quad (22)$$

$$V = Gd = \begin{bmatrix} [W^T]^+ \\ N^T \end{bmatrix} d \quad (23)$$

となる. 詳細は文献^{4), 12)}にある.

4.2 高度運動機能の例

鉛筆を4本の指でつかみ, おおのこの指は滑べりながら鉛筆の位置を変えるタスクを簡単な高度運動機能の例題として示す. Fig. 11 に示すように, 各指は鉛筆におおのこの3つの一般化接触力 (レンチ) w_i ($i=1, 2, \dots, 12$) を与える. 左の指から順番に一本ずつ左方向に鉛筆の表面を離れずに動かす. 指 i を時間 $t_{i-1} \leq t < t_i$ の間に動かす. ここで, 接触位置, 接触力および内力の大きさが変わる. この時の $W(t)$ は

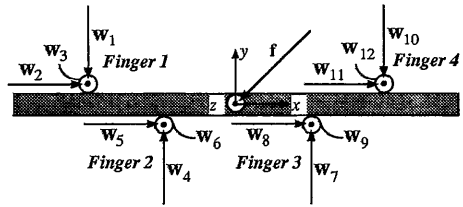


Fig. 11 4本指で掴んだ鉛筆の例

$$W(t) = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & r & 0 & 0 & -r & 0 & 0 & -r & 0 & 0 & r \\ 0 & 0 & \phi_1 & 0 & 0 & \phi_2 & 0 & 0 & -\phi_3 & 0 & 0 & -\phi_4 \\ \phi_1 - r & 0 & -\phi_2 & r & 0 & \phi_3 & r & 0 & -\phi_4 - r & 0 & 0 & 0 \end{bmatrix} \quad (24)$$

となる. たたし,

$$\phi_i(t) = \begin{cases} l_i & 0 \leq t \leq t_{i-1} \\ l_i + \alpha_i(t - t_{i-1}) & t_{i-1} \leq t \leq t_i \\ l_i + \alpha_i(t_i - t_{i-1}) & t_i \leq t \end{cases} \quad (25)$$

$i = 1 \dots 4$, l_i は指の接触点と鉛筆の重心との距離の初期値であり, r は鉛筆の半径, $\alpha_i \geq 0$ は指の滑べる速度を表す. 外力 $f_{ext}(t)$ は鉛筆に働く重力になる.

$$f_{ext} = [0 \ -mg \ 0 \ 0 \ 0 \ 0]^T \quad (26)$$

内力の大きさは Kerr, Roth により摩擦条件を満たした上で求める方法が提案されている¹⁰⁾. 指と鉛筆が離れないための条件は, 接触力の大きさが次の不等式を満たすことである.

$$c_j \geq f_0 \quad i=1, 4, 7, 10 \quad (27)$$

最小力 f_0 より大きければ良い. 内力の大きさを決めるために線形化したクローン摩擦条件は

$$c_{j+1} + \mu c_j < 0 \quad k=i \quad (28)$$

$$c_{j+1} + \mu c_j \geq 0 \quad k \neq i \quad (29)$$

$$c_{j+1} - \mu c_j \leq 0 \quad k \neq i \quad (30)$$

$$c_{j+2} + \mu c_j \geq 0 \quad k \neq i \quad (31)$$

$$c_{j+2} - \mu c_j \leq 0 \quad k \neq i \quad (32)$$

である ($j=3k-2$, $k=1, \dots, 4$). c_i は各レンチ $w_i, i=1, \dots, 12$ の大きさを表す.

式 (27-32) の線形計画問題をシンプレックス法によって解き, 内力の大きさを求めることができる. $G(t)$ 行列および接触レンチの大きさ $c_i(t)$ の軌跡を求めた上で, それを実現する指の関節空間での制御アルゴリズムを次の章で示す.

4.3 高度運動機能の実現

前節で得られたグリップ変換行列 $G(t)$ に基づくロボットハンドを制御するアルゴリズムを与える.

4.3.1 制御問題

物体の目標軌跡を基準座標系で表すと

$$\dot{p}^d(t) = Trans(r^d(t)) RPY(\varphi^d(t)), \quad (33)$$

となる。ここで、 $r(t)$ は基準座標系原点から物体中心へのベクトルであり、 $RPY(\varphi)$ は物体の姿勢をロール、ピッチ、ヨー角度 $\varphi = [\phi_1 \ \phi_2 \ \phi_3]^T$ として決定する。物体の速度を r, φ の関数 $\Psi(r, \varphi)$ を用いて

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \Psi(r, \varphi) \begin{bmatrix} \dot{r} \\ \dot{\varphi} \end{bmatrix} \quad (34)$$

と表すことができる。

物体を動かす間は安定な把握を保つために内力の大きさを摩擦条件を満たした軌跡 $d_h^d(t)$ に追従させる必要がある。物体の位置および姿勢誤差 $e_p(t) \in R^6$ ならびに内力誤差 $e_f(t) \in R^{n-6}$ を次のように定義する。

$$e_p(t) = \begin{bmatrix} r(t) \\ \varphi(t) \end{bmatrix} - \begin{bmatrix} r^d(t) \\ \varphi^d(t) \end{bmatrix} \quad (35)$$

$$e_f(t) = c_h(t) - d_h^d(t). \quad (36)$$

物体位置姿勢誤差 $e_p(t)$ と、内力誤差 $e_f(t)$ をゼロに収束させることが制御目的である¹¹⁾。

4.3.2 制御アルゴリズム

ここで制御アルゴリズムを定理として示す。

定理 1 物体に対する式 (35) の位置姿勢誤差 e_p および式 (36) の内力誤差 e_f は次の制御アルゴリズムによって両方ともゼロに収束する。

$$\begin{aligned} \tau = & M^* \Psi(r, \varphi) \begin{bmatrix} \dot{r} \\ \dot{\varphi} \end{bmatrix} + H(\theta, \theta) \\ & + M^* \Psi(r, \varphi) \left\{ \begin{bmatrix} \ddot{r}^d \\ \ddot{\varphi}^d \end{bmatrix} - K_v e_p - K_p e_p \right\} \\ & + J^T \left\{ c_h^d - \frac{1}{\delta} \dot{e}_f - \frac{1}{\delta} W^+ \dot{W} e_f + W^+ \begin{bmatrix} \omega \times \dot{m} v \\ \omega \times J \omega \end{bmatrix} \right\} \\ & + M J^+ \left\{ \dot{W}^T \begin{bmatrix} v \\ \omega \end{bmatrix} - \dot{J} \dot{\theta} \right\} \end{aligned} \quad (37)$$

ただし、

$$M^* = M J^+ W^T + J^T W^+ \begin{bmatrix} \dot{m} & 0 \\ 0 & J \end{bmatrix} \quad (38)$$

ここで、 $\delta > 0$ は内力誤差 $e_f(t)$ をゼロに収束させるように十分大きく取る。すなわち、次のような微分方程式が安定になるように $\delta > 0$ を選ばば良い。

$$\dot{e}_f = -(\delta I + W^+ W) e_f. \quad (39)$$

定理 1 の証明は文献 4) にある。

5. 実験・シミュレーション

5.1 トルク制御実験

トルクセンサの特性について調べた。センサの出力信号と関節トルクとの関係は Fig. 12 に示すように十分線形である。また、トルク制御の周波数応答のボード線図を Fig. 13 に示す。6 Hz までの帯域があり、人間の動作を解析するには適当な周波数まで測定できる。

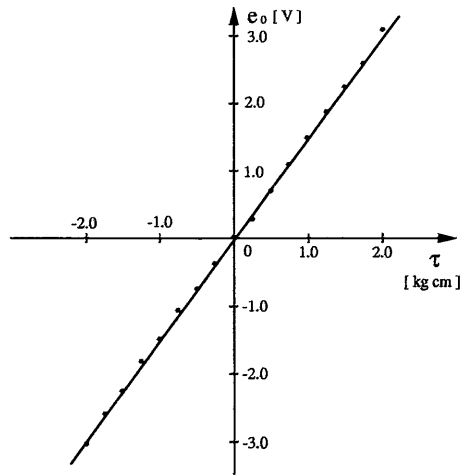


Fig. 12 トルクセンサの実験結果

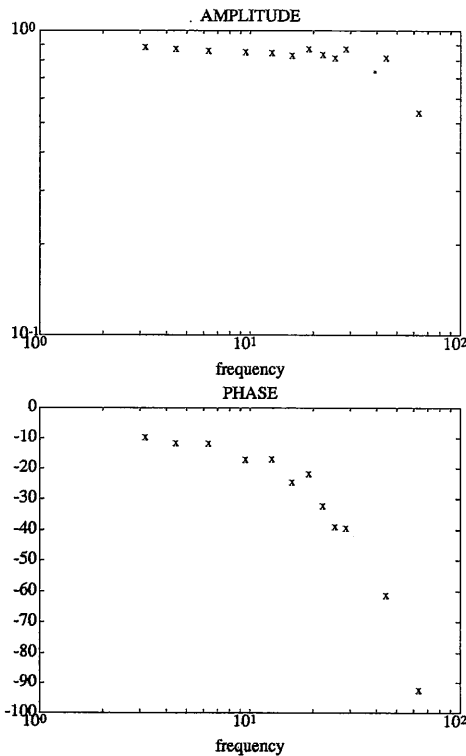


Fig. 13 トルク制御の周波数応答のボード線図

5.2 高度運動機能のシミュレーション

以上の高度運動機能の例を次の条件でシミュレーションした。クローン摩擦定数： $\mu = 0.25$ ，鉛筆半径，長さ，質量： $r = 1 \text{ cm}$ ， $l = 10 \text{ cm}$ ， $mg = 1 \text{ N}$ ，指の重心からの初期値距離，滑べる速度： $l_1 = 3 \text{ cm}$ ， $l_2 = 2 \text{ cm}$ ， $l_3 = 2 \text{ cm}$ ， $l_4 = 3 \text{ cm}$ ， $\alpha_i = 1 \text{ cm sec}^{-1}$ ，時間パラメータ： $t_0 = 0 \text{ sec}$ ， $t_1 = 1 \text{ sec}$ ， $t_1 = 1 \text{ sec}$ ， $t_2 = 2 \text{ sec}$ ， $t_3 = 3 \text{ sec}$ ， $t_4 = 4 \text{ sec}$ 。シ

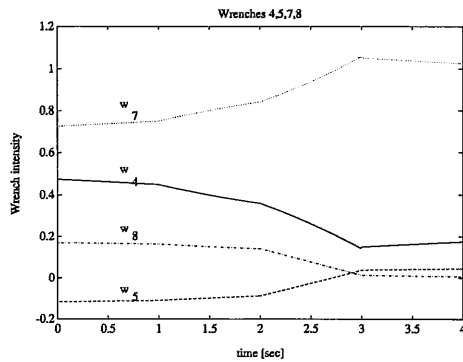


Fig. 14 各指のレンチの目標軌跡

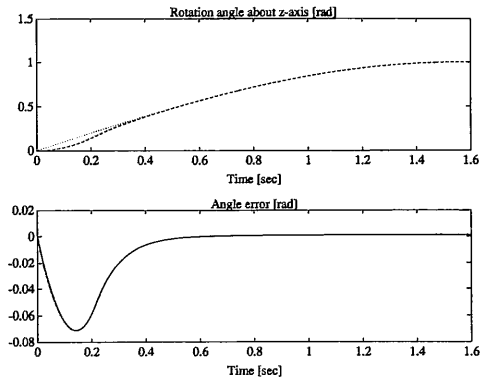


Fig. 15 鉛筆を回転させる制御の例

シミュレーションにより得られた内力および接触力の目標軌跡を Fig. 14 に示す。ただし、 $c_1(t) = 0.1N$, $c_2(t) = -0.025N$, $c_3(t) = -0.025N$, $c_6(t) = 0.04N$, $c_9(t) = -0.04N$, $c_{10}(t) = -0.1N$, $c_{11}(t) = -0.025N$, $c_{12}(t) = 0.025N$ のレンチは一定である。

5.3 制御方法のシミュレーション結果

定理 1 の制御アルゴリズムを用いて Fig. 11 に示すような鉛筆を z 軸で回転させることを考える。この時、鉛筆の位置が $p^d(t) = [0 \ 0 \ 0]^T$ になるように、姿勢が $\varphi^d(t) = [\sin \omega t \ 0 \ 0]^T$ (ω は回転数である) になるように制御する必要がある。Fig. 15 に示す様に位置姿勢誤差はゼロに収束する。

6. おわりに

本稿では知能化作業支援システムの基本的な考え方を示した。高度運動機能獲得、移植のための10自由度のセンサグループを利用する解析実験システムを提案し、仮想現実感およびダイナミックフォースシミュレータの実現法に関して述べた。センサグループのキャリブレーションを行うために ANN を用いた方法の有効性を示し、

高度運動機能の表現する数値モデルを導入した。また、簡単なスキルのシミュレーションを行い、獲得された高度運動機能をロボットハンドに実現させる制御アルゴリズムを提案した。

本センサグループシステムの開発に関して本所試作工場の岡本伸英、谷田貝悦男両氏に感謝する。

(1992年9月30日受理)

参考文献

- 1) 橋本, 「人工現実感とロボット」, 日本ロボット学会誌, 第10巻第7号, 1992
- 2) 橋本, ブス, 「知能化作業支援システムに関する研究」, 日本ロボット学会, 第2回ロボットシンポジウム予稿集, pp. 187-192, 1992.
- 3) 橋本, ブス, 向井, 原島, 「知能化作業支援システムのためのセンサハンドの開発」, 日本機械学会ロボティクス・メカトロニクス講演会講演集 pp. 445-450, 1992.
- 4) 橋本, ブス, 「IASにおけるスキルの解析」, 日本ロボット学会, 第10回学術講演会予稿集, 1992.
- 5) Hashimoto, H., Buss, M., Skill Acquisition for the Intelligent Assisting System Using Virtual Reality Simulator, *The 2nd International Conference on Artificial Reality and Tele-Existence ICAT'92*, pp. 37-46, 1992.
- 6) Hashimoto, H., Buss, M., Analysis and Synthesis of Human Skill for Intelligent Assisting System, *IEEE International Workshop on Robot and Human Communication ROMAN'92*, pp. 177-182, 1992.
- 7) 岩田, 他, “人工現実感を利用した遠隔共同作業システムの試作”, 1990年, 第6回ヒューマン・インターフェース・シンポジウム論文集, No. 1446, pp. 59-64.
- 8) 西田, 他, “ヒューマン・コンピュータ交流技術”, 1991前, オーム社.
- 9) 岩田, “インターフェイス指向の認知工学”, 1990年, 計測と自動制御学会誌, Vol. 31, No.1, pp. 250-450, 1992.
- 10) Kerr, J. and Roth, B., 1986. Analysis of Multifingered Hands. *The International Journal of Robotics Research*, Vol. 4, No.4, Winter 1986, pp. 3-17.
- 11) Li, Z., Canny, Sastry, S.S., 1989. On Motion Planning for Dexterous Manipulation, Part I: The Problem Formulation. *Proceedings of the 1989 IEEE International Conference on Robotics and Automation*, pp. 775-780.
- 12) Mason, M.T., Salisbury, J.K., 1985. Robot Hands and the Mechanics of Manipulation. *MIT Press*, Cambridge, Massachusetts, 1985.
- 13) Rasmussen, J., 1983. Skills, Rules, and Knowledge; Signs, and Symbols, and Other Distinctions in Human Performance Models. *IEEE Transaction on Systems, Man and Cybernetics*, Vol. 13, No. 3, May/June 1983, pp. 257-266.
- 14) Salisbury, J.K. and Craig, J.J., 1982. Articulated Hands: Force Control and Kinematic Issues. *The International Journal of Robotics Research*, Vol. 1, No. 1, Spring 1982, pp. 4-17.