

# インターネットを利用した低コストの遠隔制御機械システム

長谷川 辰雄\*、多田 三郎\*

機械制御によく用いられるRS232C通信形式とマルチデータ伝送によく用いられるTCP/IP通信方式を時分割で組み合わせた通信方式を開発した。この方式を用い、ヒューマン・フレンドリなインタフェースを備えた遠隔制御システムを試作し、5自由度をもつマニピュレータを遠隔制御した。マニピュレータ移動速度と通信速度のばらつきに起因するエラー発生などいくつかの問題があることがわかったが、非同期通信などで解決できる見通しを得た。

キーワード：インターネット、遠隔操作、TCP/IP、RS232C、マニピュレータ

## Low Cost Remote Control System for Industrial Machines Based on Internet Technologies

HASEGAWA Tatsuo and TADA Saburo

We developed a remote control system of industrial robot connected to Lan through personal computer. In this system, it makes possible to control processing machines, measuring instruments using RS232C and TCP/IP communication. The controlled machine is a five free joint manipulator, and we also developed user-friendly interfaces to operate this manipulator by remote control. These are JAVA widgets to input numbers and be able to operate three dimensional space sensory. key words : internet, remote control, TCP/IP, RS232C, manipulator

### 1 結 言

電子基板のキズや数値刻印などの検査は、市販の画像処理装置を使って高速に判定可能となってきている。しかし、自社製品に適合した製品検査自動化システムを構築する場合、画像処理装置の他に製造機械や仕分け機械との連携（システム化）が必要となる。このシステム化は、外部委託開発であることが多くコスト高となっている。特に、複数の機械装置を連携する場合、制御プログラムは装置ごとに作成しなければならず、その操作プログラムや通信プログラムが複雑化するため開発コストが高くなる。また、システムの維持も外部委託になるためコスト高の要因となっている。本研究では、自社によるシステム開発を可能とすることで、低コスト化を図ることを目的とする。機械制御はRS232C通信で行い、制御コマンドをインターネットを介した通信で遠隔操作を行う。インターネットはTCP/IP通信により信頼性の高いデータ通信が可能であり、WWWの使用により画面操作

が容易で共通化できる。WWWを操作画面とすることで、ユーザ（クライアント）プログラムの開発が不必要となり開発コスト削減が期待できる。本報では、TCP/IPとRS232Cによる遠隔操作システムの試作及び評価について報告する。

### 2 実験方法

#### 2-1 遠隔操作制御方式

RS232C通信は1対1を基本とするのに対し、TCP/IPは1対多通信が可能である。このことから、機械制御のデータ通信には、RS232Cが一般的に使われているのに対し、ファイル転送やデータベース検索などの複数人がデータを共有化したい場合にはTCP/IPが使われる。本システムの試作では、1対1通信と1対多通信を時間分割で組み合わせることで、機械制御を遠隔で可能とした。TCP/IPはその性質から通信速度は保証されない<sup>1,2,3)</sup>が、通信内容の信頼性は高い。特に広範囲なインターネット

\* 電子機械部

網を利用することは遠隔制御に適している。つまり、応答速度を求めなければ、データ通信の確実性と広範性により遠隔操作にはTCP/IPが適していると判断した。図1はTCP/IPとRS232Cの2つの通信プロトコルを介して、遠隔データ通信を行うためのプログラム構造図である。開発環境はLabWindows/CVIを使用し、WindowsのAPI関数を用いて作成した。クライアント・プログラムは、ユーザのデータをサーバに送信するために、サーバIPアドレスの取得、ポート番号の指定、TCP/IPのSocket通信が行われる。サーバ・プログラムでは、同時複数からの通信を可能とするために、複数の子スレッドが起動できるサーバ機能と、ユーザからのデータをRS232Cで通信し、機械制御を行う機能から構成される。

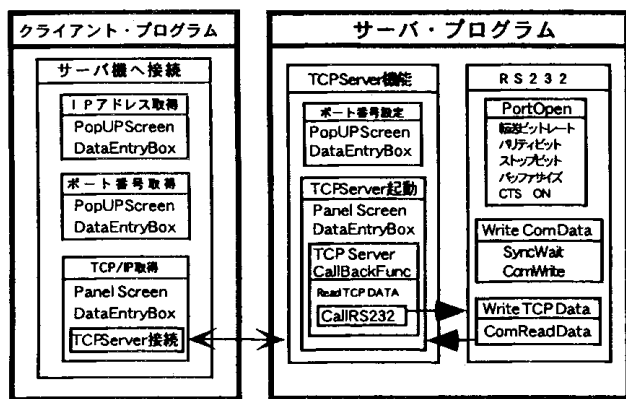


図1 遠隔通信のプログラム構造図

2-2 ロボット制御方式

マニピュレータの軌道制御 (trajectory control) は直交座標系で表現される。その制御は位置制御と力制御がある<sup>5,7)</sup>が、本試作システムでは、先端の位置制御のモデル化を行った。マニピュレータを移動させる場合、各関節の変位から先端 (手先) の位置を求める順問題<sup>8)</sup>と、先端 (手先) の位置から各関節の位置を求める逆問題<sup>9)</sup>がある。直交座標はジョイント座標 (関節の角度) で表現できる。順問題は式 (1) で表される。

$$x_d = f(\theta_d) \quad \dots (1)$$

ここで、変数 $x_d$ は目標位置、関数 $f$ はジョイント角 $\theta_d$ から先端位置への非線形変換となる。これに対し、逆問題は、上記式の逆関数で式 (2) で表される。

$$\theta_d = f^{-1}(x_d) \quad \dots (2)$$

しかし、 $f^{-1}$ は一对多の写像となり、任意に決定できない。つまり、動作条件を与えることで、目標の座標からジョイントの変位 (関節の角度) を求めることができる。本試作システムでは、アームロボットの4つの各関節の角度について、直交座標の関数としてモデル化し

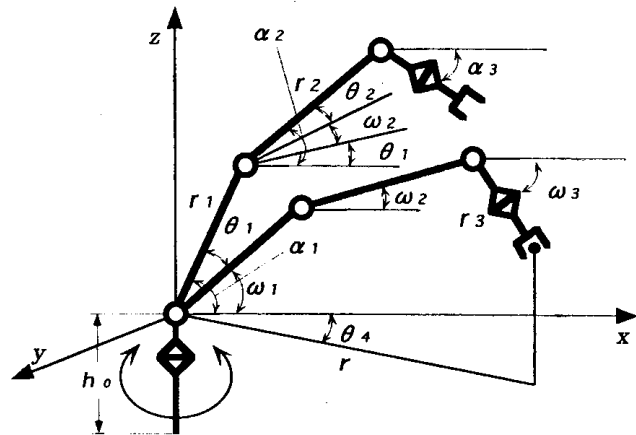


図2 マニピュレータの座標系

た。図2に示すように、腕の長さが $r_1, r_2, r_3$ の5自由度マニピュレータについて、先端座標 $(x, y, z)$ と各関節の変位角 $\theta_1, \theta_2, \theta_3, \theta_4$ との関係を求めた。これを式 (3) ~ 式 (6) に示す。

$$\theta_2 = -\cos^{-1} \frac{r^2 + (z - h_0 + r_3)^2 - r_1^2 - r_2^2}{2r_1 r_2} + (\omega_1 + \omega_2) \quad \dots (3)$$

$$\theta_3 = -\sin^{-1} \frac{r^2 + (z - h_0 + r_3) + r_2^2 - r_1^2}{2r_2 \sqrt{(z - h_0 + r_3)^2 + r^2}} + F \quad \dots (4)$$

ただし、 $F = \tan^{-1} \frac{r}{z - h_0 + r_3} - \frac{\pi}{2} + \omega_2 - \omega_3$

$$\alpha_3 = \theta_1 + \theta_2 + \theta_3 + \omega_3$$

$$\theta_1 = -\left(\frac{\pi}{2} + \theta_2 + \theta_3 + \omega_3\right) \quad \dots (5)$$

$$\theta_4 = \tan^{-1} \frac{y}{x} \quad \dots (6)$$

上記の式により、直交座標系からジョイント角度を求めることができる。

2-3 ユーザ・インタフェースの設計

人間が機械制御する場合、その操作パネルや画面の操作性の良さが大きな影響を与える。機械装置のコントロールパネルの数値入力やモード選択は、以前に比べてアイコン化などのグラフィックスを利用したものが主流となっている。本評価システムでは、テキスト型入力とグラフィック型入力の違いについて評価実験を行った。実験は、機械制御のための数値入力の速度について、キーボード装置によるテキスト型入力方式とマウス装置によるグラフィック型のダイヤル入力による比較を行った。テキスト型入力方式は、HTMLの<FORM>タグを

利用し、その処理をCGIプログラムで行った。グラフィック型のダイヤル方式は、JAVAのDIALクラスにより設計した。この他に、機能選択をするときの煩雑な画面スクロールを無くし、TABクラスによるカード選択方式を設計した。一方、機械操作を人間の感覚に近づけるための

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(a)X軸中心回転 (b)Y軸中心回転 (c)Z軸中心回転

行列式1 Affin変換行列式

図3は3次元モデルのVRMLをAffin変換するためのクラス階層図である。VRML Classはマウスの移動量の計算とグラフィックスのVRAM転送を行うメソッドがあり、スレッドプログラムとして実行される。Affin\_Matrix Classはデータとして4×4行列、メソッドとして回転・拡大・縮小の行列式を作成する。ここで、マウスで与える回転変位はX軸方向とY軸方向の2つしかないため、X軸中心の行列式とY軸中心の行列式を組み合わせることでAffin変換行列を求めた。このように、Affin変換行列は複数の回転や移動を一度に計算できる特徴がある。VRML\_Node Classは、ワイヤーフレームで記述されたVRMLのノード値を解釈し、Affin変換するための元データを構築する。

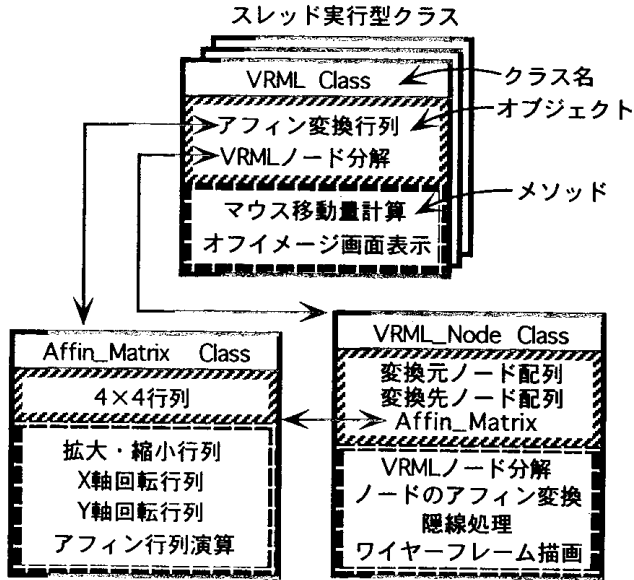


図3 VRMLとAffin変換のクラス階層図

インタフェースとして、3次元CG (VRML) とJAVA<sup>9,10</sup>を利用した画面を設計した。この操作モデルは、VRMLとAffin変換を組み合わせた構造になっており、3次元モデルはデータ量が少量ですむワイヤーフレームで構築した。3次元モデルを移動・回転させる手法として一般的なものがAffin変換であり、行列式1の4×4の行列式で表現される。

3 実験結果及び評価

3-1 遠隔操作制御の評価

遠隔操作はパソコン (Windows95) の操作パネルからLAN接続された産業用ロボット「Move Master II」(三菱電機製) に対し、制御コマンドを送信することで駆動制御し、動作結果を確認した。実験条件を、Move Master IIの動作範囲 (図4) と、遠隔操作実験の構成図

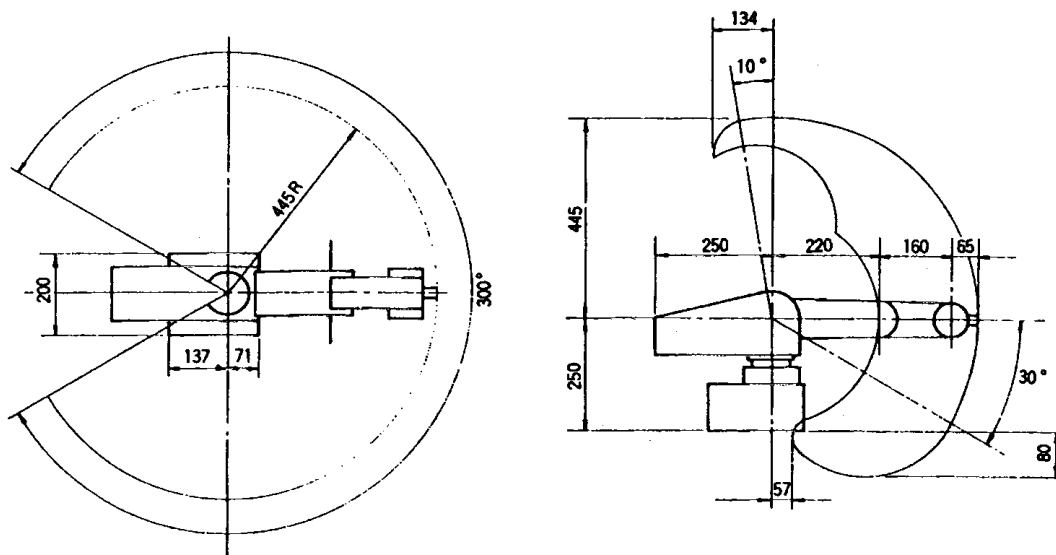


図4 Move Master IIの動作範囲

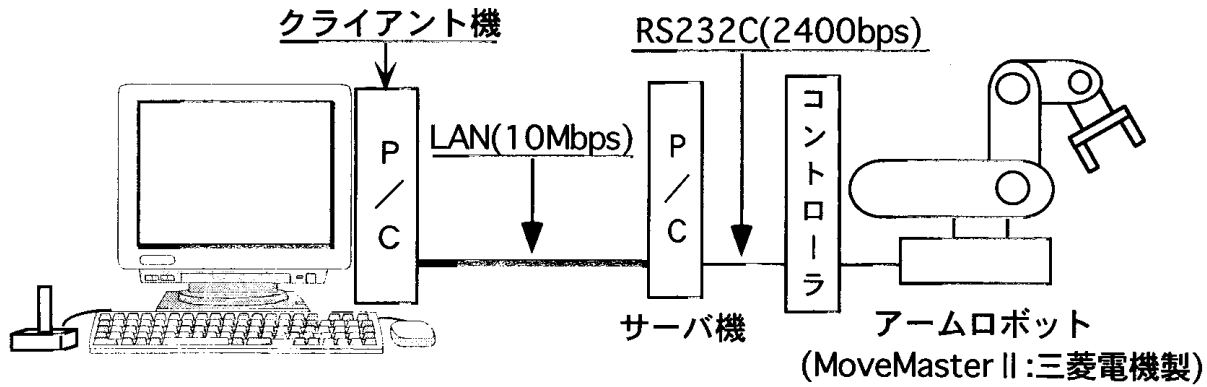


図5 遠隔操作実験構成図

(図5)及びRS232C通信パラメータ(表1)に示す。MoveMaster IIの制御コマンドは38種類あるが、このうちRS232Cで制御可能な7種類の制御コマンドについて、クライアントからの送信開始からMoveMaster IIの動作開始までの平均応答時間を計測した。その測定結果を表2に示す。制御コマンドのNT(原点出し)やHO(ホームポジション)などは、パラメータ引数がないため、2バイトの通信になるのに対し、MIやPSは各関節の駆動パラメータを必要とするため、送信時間を要している。TCP/IP通信では最大10Mbps、RS232C通信が最大2400bpsの仕様速度で応答時間を計算すると0.01秒程度になり、比較するとかなり遅い。この原因は、TCP/

表3 移動距離精度

移動距離(mm)	5	10	15	20	25	30	35	40	45	50
実測距離(mm)	5	9.5	14.5	19.5	24	29	34	39	44	49
誤差(mm)	0	0.5	0.5	0.5	1	1	1	1	1	1

### 3-2 ロボット制御の評価

マニピュレータ制御モデルの確証を、レーザー変位計を用いた位置精度と移動距離精度を測定した。図6の位置精度測定グラフは、原点(0, 0, 0)から位置(300, 50, 120)の2点間の座標を繰り返し移動し、各軸方向ごとに10回の精度測定を行った。表3の移動距離精度は位置(300, 50, 120)からZ軸方向の移動距離を5mmごと増加させ、その移動距離を測定した。この実験結果から、位置の標準偏差を求めると、X軸:0.019mm、Y軸:0.057mm、Z軸:0.113mmであり、移動距離の精度が分かった。このことから、マニピュレータの制御モデルが正当であることが確認できた。

表1 RS232C通信パラメータ

通信速度(bps)	パリティ	データ(bit)	ストップビット(bit)	ハードウェアフロー制御
2400	奇数	8	2	ON

表2 遠隔操作応答時間 単位:秒

制御コマンド	NT	HO	OG	MI	PS	GO	GC
平均応答時間	2.55	2.61	2.55	2.84	2.83	2.63	2.65

IP通信の実通信速度が低いことや、ロボットがコマンドを受けて実行するまでの動作時間、プログラム実行時間などが考えられる。また、本ロボットのRS232Cの最大速度(9600bps)での通信ができていない問題がある。

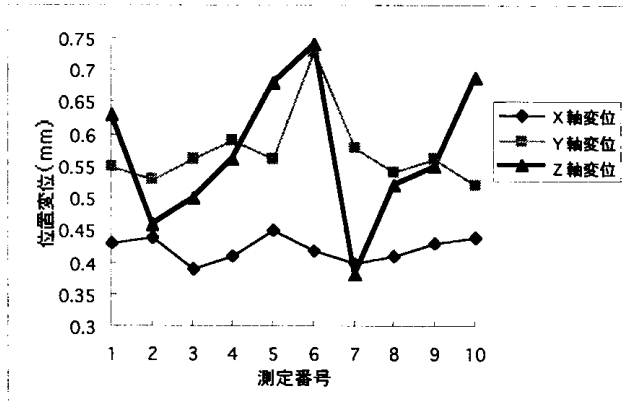


図6 位置精度グラフ

### 3-3 ユーザ・インタフェースの評価

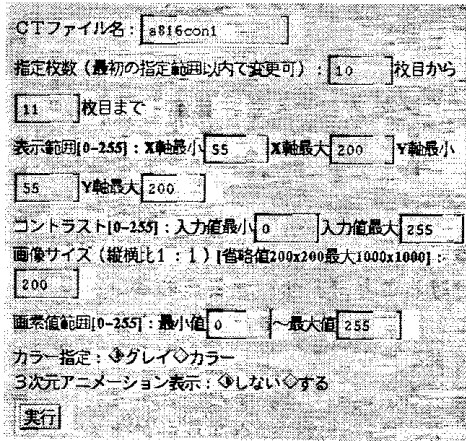
ユーザ・インタフェースの良し悪しは、人の嗜好に影響されることが多く、その評価は困難である。この試作システムでは、数値の入力速度と3次元パラメータの入力方式について検討した。

#### 3-3-1 テキスト型入力とダイヤル型入力

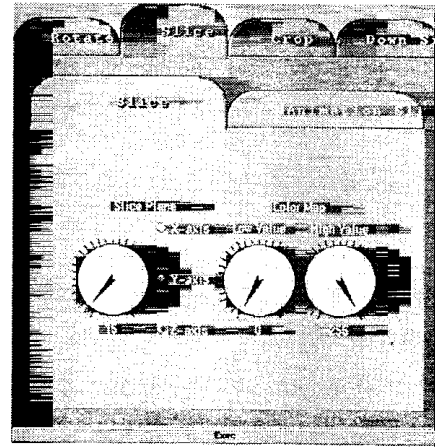
テキスト型とグラフィック型の数値入力速度の比較を図7の入力画面を作成し実験した。測定は、3桁のランダムな整数値を2回入力し、そのときの入力平均時間を

表4 3桁整数値入力時間 単位:秒

入力装置	キーボード	マウス
厳密入力平均時間	6.4	12.1
冗長入力平均時間	5.7	3.8



(a) テキスト入力



(b) ダイアル入力

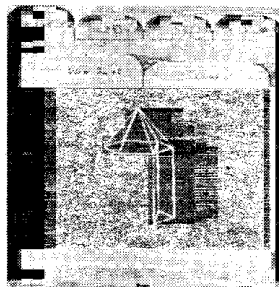
図7 テキスト型入力(a)とダイアル型入力(b)

測定した。表4は3桁数値を厳密に入力した場合と、例えばx:100~120、y:200~220のような冗長な数値入力をする場合の平均入力時間を比較した結果である。この結果から、厳密な数値入力はキーボード入力の方が早く、冗長性のある入力はマウスが早いことがわかった。ダイヤル式の冗長入力では、数値箇所が視覚的に瞬時に判断できるため、テキスト入力よりも早く入力できる。厳密入力方でマウスが劣っている原因は、マウスイベント処理が遅いことや、ダイヤルパラメータは円周上にメモリを割り振るため、数値範囲が大きくなれば、マウスによる移動角度が小さくなるため、厳密数値選択が困難となるためである。例えば、ダイヤル設定の数値幅が360を超えるような場合は、1メモリを設定するために1度以下の回転がマウス移動に要求され、微妙な設定ができなくなる。

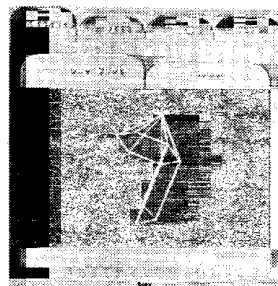
### 3-3-2 3次元座標の感覚的入力方式の検討

グラフィック型のユーザインタフェースを考えた場合、グラフィックは画像データであるために多用すればするほど、処理に時間を要する。インターネットでは、特に通信速度に大きく影響するため、少グラフィックス化が提案されている。以前のインターネット環境でのシ

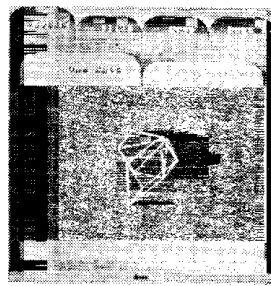
ステム開発では、グラフィックス開発は複雑であるために困難であった。操作性を重視するか、処理速度を重視するか、どちらか一方を重視した設計であった。この双方を満たそうとしたのがJAVA開発環境<sup>9,10</sup>である。JAVAは、マウスイベントやグラフィック処理をクライアント機で行うことで、グラフィックス処理速度を向上させた。インターネット通信の負荷は、サーバ機への最初の1回目のアクセス時に、グラフィックスデータを送信する時だけであり、その後のグラフィックスデータの処理は全てクライアント側で行う。この新しいネットワーク開発方式は、コンピュータOS自体のインタフェースに近くなってきており、境界がなくなりつつある。本試作システムでは、上記技術を応用して、JAVAとVRMLにより3次元モデルを利用した数値入力について検討した。図8は、ワイヤーフレームで構築した3次元モデルをマウスで対話的に回転(Affin変換)させるインタフェースであり、3次元モデルが回転することで、人間に3次元座標を感覚的に理解させようとしたモデルである。マウスの移動量(dx, dy)によってX軸方向回転、Y軸方向回転のパラメータを与えた。実験結果により、3次元感覚は、この2方向の回転で十分与えることがで



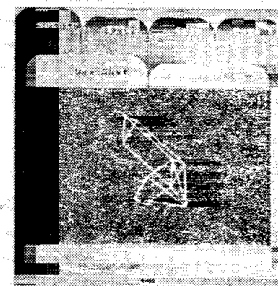
(a) dx=0, dy=0



(b) dx=14.14  
dy=-35.18



(c) dx=39.85  
dy=-22.09



(d) dx=75.85  
dy=-9.0

図8 3次元座標の感覚的入力方式

きた。

#### 4 結 言

##### 4-1 遠隔操作制御

遠隔操作の実験では、TCP/IPとRS232Cの両プロトコル通信が可能なプログラムを構築することができた。しかし、MoveMaster IIは、その動作中のコマンド受信はエラーとなる。連続するコマンドを実行する場合は、MoveMaster IIがコマンドを受け付ける状態（コマンド受信可能状態）を常に監視しながら通信を行わなければならない（データフロー制御）。MoveMaster IIのRS232Cのデータフロー制御は、ハードウェアフロー制御<sup>6)</sup>を使用し、送信要求のRTS(Request To Send)と送信許可のCTS(Clear To Send)の信号線によって行った。この手法はポーリングと呼ばれ、制御機械の通信では一般的となっており、本試作システムでも適用している。しかし、9600bpsでの通信実験では、接続するが、コマンドを送信するとエラーとなる。このような通信速度による送受信のばらつきがあることから、今後は通信プログラムの信頼性や非同期通信による効率化の強化を行う予定である。また、インターネットを使用する場合、セキュリティに対する対応が必須とされているが、今回の試作システムでは対応していないため今後の課題とする。

##### 4-2 ロボット制御

ロボット制御は、直交座標系から関節角度を計算するマニピュレータ・モデルを構築し検証した。位置精度や移動精度を求めることで本モデルを実証した。今回の実験では、本モデルの検証を目的としたため、十数回程度の繰り返し試験しかしていない。そのため、ロボット自体の性能を評価する試験とはなっていない。しかし、マニピュレータ制御は、ブドウやリンゴなどの果実収穫ロボットや、製造ラインにおける電子部品の移動など、多数の応用が考えられる。今後は、マニピュレータ・モデルと視覚センサ（CCDカメラ）や距離センサ、力覚センサなどを組み合わせ、不良解析が遠隔で実施できるシステムを検討する。

##### 4-3 ユーザ・インタフェース

テキスト入力型とダイヤル入力型の比較実験から、それぞれ長所短所があることがわかった。つまり、ユーザ・インタフェースを設計する場合の入力方式について、テキスト入力とダイヤル入力についての特徴を抽出することができた。ユーザの嗜好による影響が強いユーザ・インタフェースの設計には、多種多様な部品を準備することで選択肢を増やし、その特徴をわかりやすく体系付けることで、ユーザ要求のインタフェースを設計することができる。単に部品を増やすだけでなく、入力スピードや視覚的な分かり易さなどの品質を考慮しながら設計する予定である。また、3次元情報を感覚的に与える手法の検討では、マウス移動による3次元モデルの回転は実現できた。しかし、回転角度からの座標算出など、3次元視覚モデルからの3次元座標への変換モデルの作成が今後の課題となる。

#### 文 献

- 1) Kevin Washburn, Jim Evans: TCP/IPバイブル, ソフトバンク (1994).
- 2) W.Richard Stevens: Unixネットワークプログラミング, トッパン (1992). PP235-236
- 3) Bill Rieken, Lyle Weiman: UNIX network applications programming, HBJ出版 (1994) pp.5-7.
- 4) 奥村晴彦: C言語によるアルゴリズム辞典, 日経印刷 (1995). pp245-248
- 5) 近藤文治, 前田和夫, 岩貞継夫, 坪根治広: 基礎制御工学, 森北出版 (1986) pp.4
- 6) 横田秀次: 32ビット通信プログラム作成法, CQ出版 (1997). pp.33-47
- 7) Chae H.An, Christopher G.Atkeson, John M.Hollerbach: モデルに基づくロボットマニピュレータの制御, 実教出版 (1991) pp.4-9.
- 8) 岡本嗣男, 白井良明, 藤浦建史, 近藤直: 生物にやさしい知能ロボット工学, 実教出版 (1992). pp.19-22
- 9) 青柳龍也: Java API プログラミングガイド, 工学図書 (1996).
- 10) 有賀妙子, 竹岡尚三: Java 1.1プログラミング, ソフト