

人間型ロボットの開発

Development of Humanoid Robot

○柳川章全* 那須康雄* 武田謙郎* CapiGenci* 中村友則*
○ShouzenYanakawa, YasudNasu, KenroTakeda, TomonorNakamura
* 山形大学工学部

キーワード：人間型ロボット (HumanoidRobot), CORBA,
ロボット コントロール プラットホーム (RobotControlPlatform)

連絡先：〒992-0038 山形県米沢市城南4-3-16
山形大学大学院理工学研究科 機械システム工学専攻 那須研究室 柳川章全
TEL: (0238)26-3237 Fax: (0238)26-3237
E-mail : yanakawa@mnasu2.yz.yamagata-u.ac.jp

1. はじめに

将来、ロボットは人間の補助的な存在になり、産業の分野だけではなく、災害救助、家庭など、人間の身近な存在になると考えられる。我々は、産業用ロボットとしてではなく、より身近な存在として、人間を補助する人間型ロボットの開発をしている。

人間型ロボットに限らず、ロボットの開発には機械、電気、情報などのさまざまな研究分野が必要とされている。ロボット研究者は、これらの各分野の技術を修得し、自分で開発しているロボットに実装しなければならない。しかしながら、これらの技術は各研究分野において専門的に研究され、ロボット研究者が修得して実装するよりは、各専門分野の研究者が実装した方がはるかに完成度が高く、開発スピードも早い。トータルシステムとして完成度の高いものを作り上げるには、各研究分野の連携をスムーズに行なうような仕組みが必要である。

本研究では、これら各研究分野の技術の連携をスムーズにし、さらに機能の追加を容易にするような、ロボットコントロール・プラットフォームの提案と、ロボットコントロール・プラットフォームの制御対象である人間型ロボットの製作について述べる。

2. 人間型ロボットの開発

2.1 ハードウェア構成

人間は、膨大な筋によって骨格系の活動が生まれている。人間の骨格は複雑で、人体各部にある筋の数を調べてみると、下肢だけでも62対もの筋

があると言われている。1) この自由度すべてにアクチュエータを配置すると、極めて複雑な動的システムが必要となり、制御が困難となる。人間型ロボットの自由度を決定する際に、人間同様のシステムを実現するのは、ほとんど不可能である。よって、人間の機構、関節を正確に再現するのではなく、できるだけ少ない自由度で、人間の動きを実現できるように設計を行った。

一般に、ロボット・マニピュレータは、3次元空間で手先効果器を位置づけるためには少なくとも6関節が必要であることが言われている。2)

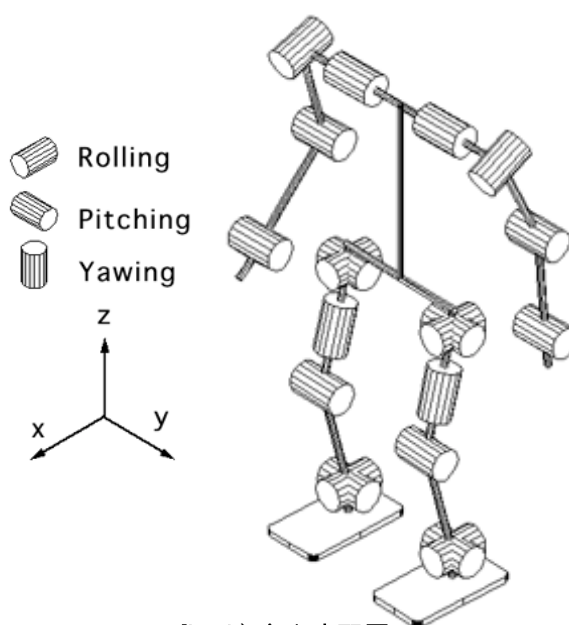


fig.1) 自由度配置

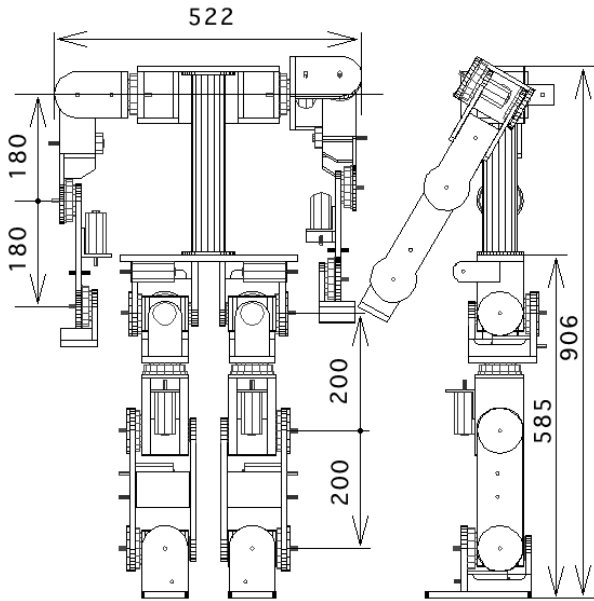


fig.2) 人間型ロボットの寸法

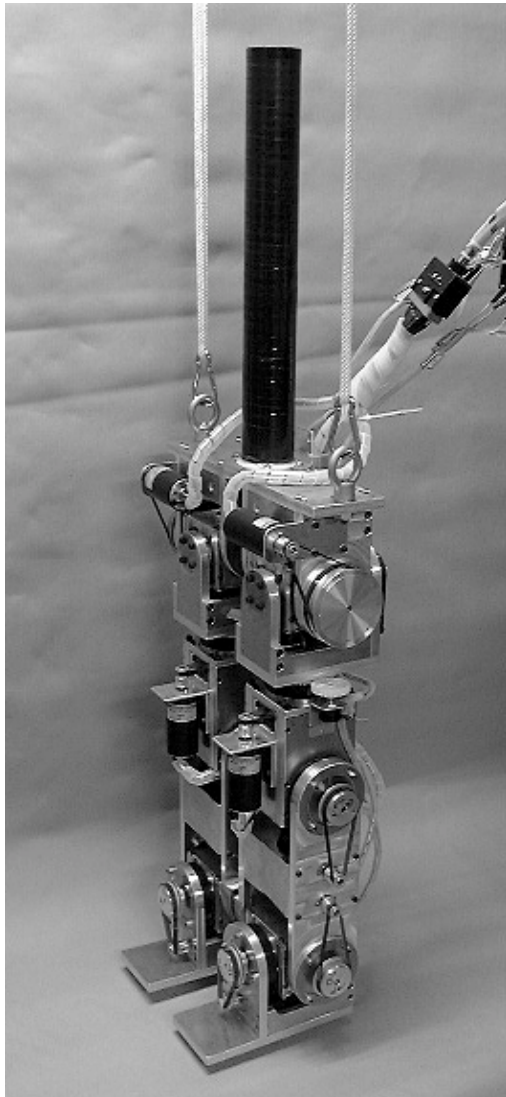


fig.3) 下半身部分の写真

また、2足歩行ロボットの脚部に関して、腰をマニピュレータの土台、足底を手先と見たとき、この系は床に対して固定点を持たない宙に浮いたマニピュレータと考えることができる。3)

これらのことより、人間型ロボットの脚部の自由度配置は、股関節にPitching, Rolling方向の2自由度、大腿部にYawing方向1自由度、膝部にPitching方向1自由度、足首関節にはPitching, Rolling方向の2自由度、片脚6自由度を設け、両脚12自由度とした。fig.1に自由度配置を示す。

人間型ロボットの全長を1.2mと定め、設計を行った。下半身部分の高さは588mm、重量は20kg、頭部、上半身部はまだまだ未完成である。

各部品は構造部材にはすべてアルミを使用し、軽量化のため、細部に至るまで中抜きを行った結果、全体の重量に対する構造部の重量が4割以下となっている。

各関節のリンク間には、ハーモニックドライブを使用し、DCモータからの動力は、プーリ・ベルトを介してハーモニックドライブに伝達する構造になっている。

各関節の有効角度については、人間の動き4)を参考にし、リンク間の干渉が無いように工夫をした。

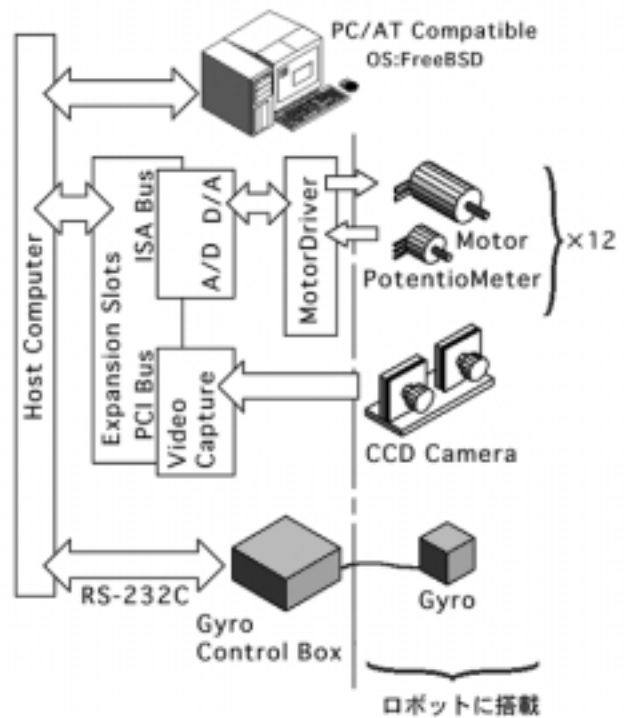


fig.4) システム構成図

現段階では電源、モータドライバ等のコントロールボックスはロボットには搭載していない。

fig.2 に人間型ロボットの各寸法, fig.3 に下半身の写真, fig.4 に人間型ロボットのシステム構成図を示す。

2.2 ソフトウェア構成

各研究分野での技術には、その分野に合ったプログラミング言語が使用されることが多い。それぞれの技術を直接連携させるためには、どちらかのプログラミング言語に合わせなければならない。また、そのプログラムがどのような OS 上で動作するものなのかによって、異なる OS 間のプログラムの移植も必要となってくる。

このことは、各専門分野の技術を統合し、ロボットのシステムとするための障害となるものである。この問題を回避するために、本研究では CORBA (Common Object Request Broker Architecture) を用いた。

CORBA は OMG (Open Management Group) が標準化した仕様であり、プログラミング言語、プラットフォームに非依存である。

CORBA の規格に準拠することで、異なるプログラミング言語で開発されたオブジェクト同士が、CORBA を経由して通信することができる。よって、異なる OS 間でのプログラムの通信が可能となる。

全てのオブジェクトは、ORB (Object Request Broker) を通じて通信が行われる。プログラミング言語、プラットフォームの違いは、この ORB で吸収される。オブジェクトは、直接オブジェクトに対して送るのではなく、ORB を通じて送信するため、必要なオブジェクトの場所を指定する必要がない。この ORB によって、オブジェクトは、場所を全く

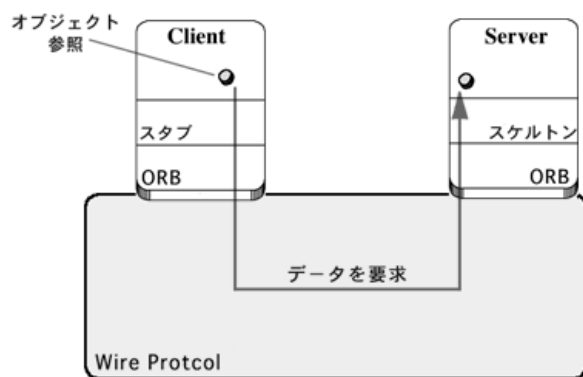


fig.5) ORB

意識することなくオブジェクトを呼び出すことができる。すなわち、PC をネットワークに接続だけで、

ネットワーク上すべての PC のオブジェクトが使用可能となる。

この CORBA ルーチンの追加と、異なるプログラミング言語間、異なるプラットフォーム間の移植作業のどちらが軽度な作業であるかは容易に理解できる。

つまり、各研究分野に都合の良い言語、OS 上で開発されたプログラムに、CORBA の通信ルーチンを追加するだけで、互いの研究成果を利用できることになる。よって、各研究分野の連携をスムーズにし、共有を可能にする。

既存の工業ロボットや、システム (ソフトウェア) に CORBA を用いることによって、分散システムを構築している例が、多数見られ、それに関する研究も盛んに行われている。5) 6) 7) 8) 9)

3. ロボットコントロール・プラットフォームの提案

3.1 インターフェース (IDL) の定義

本研究で提案するロボットコントロール・プラットフォームは、fig.6 に示すような多層クライアント/サーバ方式である。

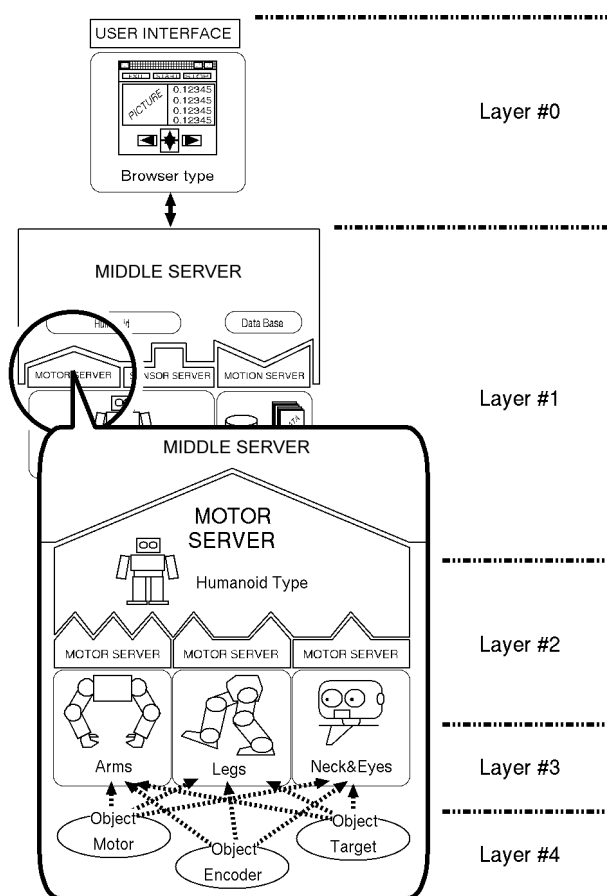


fig.6) Multi layer 構造

下位の階層から、CORBA オブジェクトサーバ (layer#3, #4), 機能をまとめたサーバ (layer#2), 中間サーバ (layer#1), ユーザインタフェース (layer#0) という構造になっている。

各サーバ間のインターフェースは、CORBA IDL (Interface Definition Language) で統一する。

IDLは、分散オブジェクトのインターフェースを記述するための定義言語である。IDLで定義することによって、クライアント/サーバ間のインターフェースが確定でき、クライアントとサーバが独立して開発されたとしても、双方の通信が可能となる。

3. 1. 1 オブジェクトのIDLによる記述

現時点で実装しているサーバは、モータ制御サーバ、センサ監視サーバ、中間サーバである。情報サーバ、ユーザインタフェースについては、開発途中である。

オブジェクトのIDLによる記述の例として、定義したモータ制御サーバのIDLを示す。

```
module MotorModule {  
  
    // Interface of sensor server  
    interface MotorInterface {  
  
        // Get server status  
        short GetServerStatus(in string user_ID);  
  
        // Set angle value of motor  
        short SetMotorData(in string user_ID,  
                           in short motor_no,  
                           in double degree_data);  
  
    };  
};
```

このIDLはロボットのモータの回転角度をクライアントから与えるためのものである。クライアントからサーバに対して、指令を与えるのは、interface MotorInterface で定義されるセクション中の、GetServerStatusと SetMotorDataである。

クライアントに要求されることは、GetServerStatus を用いて、サーバが使用可能か否かを問い合わせ、SetMotorData を用いて、モータの回転角度を指定するだけである。

このように、ロボットを機能で分割し、分割された機能毎に入出力を定義することで、モノリシック

なプログラムに比べて、各機能の関係を明確にでき、また各機能の構築を簡潔にすることができる。他のサーバについても同様なIDLを定義する。

3. 2 各サーバの機能

3. 2. 1 モータ制御サーバ

モータ制御サーバは、モータを管理するサーバである。モータの目標値を中間サーバから受け取り、それぞれのモータを動作させる。モータ制御サーバは他のサーバと違い、使用者が唯一である必要がある。

3. 2. 2 動作記述サーバ

動作記述サーバは、歩行理論、腕による作業に関する理論など、ロボットの動作に関する情報を提供する。1つのサーバとして、独立させることにより、さまざまなアルゴリズムの追加、変更、削除が容易に行える。

3. 2. 3 センサ監視サーバ

センサ監視サーバでは、ロボットに搭載されているさまざまなセンサ類を、常時中間サーバに提供する。このサーバは、クライアント、動作記述サーバなどからの要求に応じ、必要なセンサ情報を提供する。現在、搭載済みのセンサは、CCDカメラ、ジャイロであるが、圧力センサの搭載も予定している。

3. 2. 4 情報サーバ

現状のCORBAの構造のみでは、ロボットをCORBAオブジェクトとして扱うには不十分な部分がある「CORBAを用いることで、ネットワーク上すべてのオブジェクトを、場所を全く意識することなく呼び出すことが出来る。」と述べたが、以下に述べる問題点がある。

- ・ロボットを動作させる事ができるのは、一度に1ユーザーである。
 - ・何らかのネットワーク障害によって、通信が切れた時に、現状のCORBAでは復帰を定義できない。
 - ・ロボットに最適な動作を提供するサーバを特定できない。
- などである。

これらの問題点を補うために、必要となる情報を提供するCORBAサーバとして、情報サーバを設ける。

3. 2. 5 中間サーバ

中間サーバは、プラットフォームを利用する際の基盤となるサーバである。

中間サーバの機能を以下に述べる。

ユーザインターフェース・クライアントはユーザが常時起動しているプロセスとは限らない。なんらかの理由で、クライアントと中間サーバの通信が不通になったときに、中間サーバは、ロボットを安全に停止させる等の管理が必要である。また、中間サーバは、ユーザインターフェース・クライアントと各サーバ間に設けるが、ユーザインターフェース・クライアントが存在しない間、各サーバに対するクライアントとして働き、常にロボットを監視する。

3. 3 プロトコルの定義

上述した各サーバがどのような振る舞いをするか述べる。

1. 情報を提供する各サーバは、起動時に情報サーバに、自分が提供できるサーバの情報を登録する。
2. 情報サーバは、クライアントに利用可能なロボット、サーバの情報を伝え、クライアントは選択する。
3. 情報サーバは、クライアントの選択したサーバのアドレスを中間サーバへ、クライアントには中間サーバのアドレスを提供する。
4. クライアントは、情報サーバから指定された中間サーバへアクセスすることで、ロボットを利用できる。

この後、情報サーバは、他のクライアントやサーバからの要求を待つ状態に移行する。

4. 実験

モータ制御サーバ、センサ監視サーバ、中間サーバの実装を行なった。クライアントとしては、歩行動作をさせるためだけの、ごく簡単なプログラムを使用して、提案したロボットコントロール・プラットフォームの検証を行った。

今回は、下半身のみの制御を1台のPC上で、すべてのサーバを実行した。

ロボットコントロール・プラットフォームを”使用した場合”、”使用しない場合”、両者の動作が同様であることを確認した。

よって、サーバ間のデータ通信が問題なく行えていると思われる。

5. まとめ

今回は、

- ・人間型ロボットの脚部を製作した。
- ・ロボットコントロール・プラットフォームの提案、およびモータ制御サーバ、センサ監視サーバ、中間サーバの実装を行なった。

今後の予定として

情報サーバ、ユーザインターフェースの実装をし、ロボットコントロール・プラットフォームを完成させる。また、ロボットコントロール・プラットフォームを複数のPC上で検証をする。

さらに、上半身、頭部の開発、さまざまなサーバの開発をしていく。

参考文献

- 1) Mimir Vukobratovic(著), 加藤一郎 山下忠(訳) : 歩行ロボットと人間の足, 日刊工業新聞社, p9(1975)
- 2) John J. Craig(著), 三浦宏文 下山勲(訳) : ロボティクス - 機構・力学・制御 -, 共立出版, p59(1996)
- 3) 藤本康孝 河村篤男 : 床との衝突および摩擦を考慮した2足歩行ロボットの三次元運動シミュレーション, 日本ロボット学会, vol.15, p857/863(1997)
- 4) 中村隆一 齊藤宏 : 基礎運動学, 医歯薬出版(1976)
- 5) <http://www.ca.sandia.gov/~carmen/DistObjsMon.html>
- 6) George Toye, Mark R. Cutkosky, Larry J. Leifer, J. Marty Tenenbaum, Jay Glicksman, SHARE: A Methodology and Environment for Collaborative Product Development, In Post- Proceedings of the IEEE Infrastructure for Collaborative Enterprises(CDR-TR#19930507)
- 7) Robert A. Whiteside, Carmen M. Pancerella, Paul A. Klevgard, A CORBA- Based Manufacturing Environment, Proceedings of the Hawaii International Conference on System Sciences, 1997
- 8) Steve Vinoski, CORBA: Integrating Diverse Applications Within Distributed Heterogeneous Environments, IEEE Communications Magazine, vol.14, No.2, February, 1997
- 9) 岡野 : CORBAを用いた開放型製造実施システム, 精密工学会誌, vol.65 No.9(1999)