

# 小型ヒューマノイドロボット HOAP-2の遠隔操縦

## Teleoperation of the small size Humanoid Robot HOAP-2

松本剛明\*, 近野敦\*, 内山勝\*

Takaaki Matsumoto\*, Atsushi Konno\*, Masaru Uchiyama\*

\*東北大学

\*Tohoku University

キーワード : ヒューマノイドロボット (Humanoid Robot), マンマシンインタフェース (Man-machine interface)

連絡先 : 〒980-8579 仙台市青葉区荒巻字青葉01 東北大学大学院工学研究科航空宇宙工学専攻 内山研究室  
松本剛明, Tel.: (022)217-6973, Fax: (022)217-6971, E-mail: takaaki@space.mech.tohoku.ac.jp

## 1. はじめに

人間は200以上の自由度を持つとされるが, 人間を模倣したヒューマノイドロボットは20から30程度の自由度を持っている. そのヒューマノイドロボットの操縦において, すべての関節を個々に制御して操縦することは不可能であり, 何らかのインタフェースを用いてより低い自由度で操縦する必要がある. 例えば本田技研工業のASIMOがジョイパッドを用いて操縦されている. ジョイパッドでの操縦は直感的でわかりやすく, 技術者はもちろん非専門家でも簡単に操縦が可能である.

ところがこのようなインタフェースを用いた場合, 操作者の操作指令に応じて, 実時間でヒューマノイドロボットの動作を計画する必要がある. しかし全く無の状態から実時間で動作計画を行うのは困難である.

本研究では, ヒューマノイドロボットの動作を, あるまとまりのあるプリミティブな動作に分割し, あらかじめ用意しておく. これらのプリミティブな

動作を操作指令に応じて結合し, 実時間でヒューマノイドロボットの動作生成を行う.

本論文では実時間動作生成法とジョイパッドを用いたヒューマノイド操縦インタフェースについて報告する. 富士通オートメーション社製のヒューマノイドロボットHOAP-2での実験結果を示す.

## 2. HOAP-2諸元

### 2.1 主な特徴

ヒューマノイドロボットHOAP-2の外観をFig. 1に示す. HOAP-2は身長は500 [mm], 重量は7 [kg]と小型・軽量である. OSとしてLinuxに実時間性を付加したRT-Linuxを採用しており, 構成デバイスのインタフェースも公開されている. 各部を繋ぐ体内LANにはUSB1.0を採用しており, 高速かつ容易にデバイスを扱うことができる. また1 [ms]の周期で制御が可能である上, オプションで無線での制御や頭部カメラの搭載にも対応している.

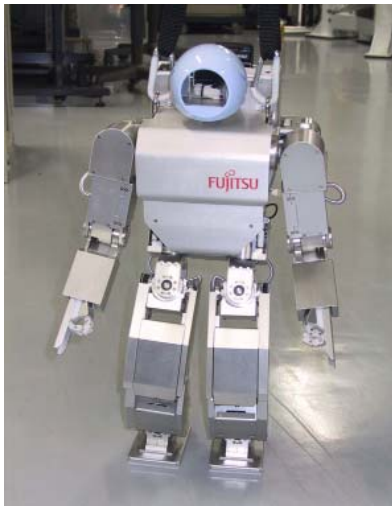


Fig. 1 An overview of HOAP-2.

## 2.2 ハード構成

HOAP-2の関節の配置をFig. 2に示す．その自由度の内訳は，

- 6DOF/脚 × 2
- 4DOF/腕 × 2
- 1DOF/腰 × 1
- 1DOF/手 × 2
- 2DOF/首 × 1

の計25DOFである．

またセンサは，

- FSR圧力センサ × 4 /各足底
- 3軸角速度センサ × 1 /腰
- 3軸加速度センサ × 1 /腰
- 角度センサ × 1 /各関節

を搭載している．

## 3. システム構成

### 3.1 システムの仕様

開発したHOAP-2の遠隔操縦システムは，マンマシンインタフェース用コンピュータ,ジョイパッドおよびロボットからなる．インタフェース用コンピ

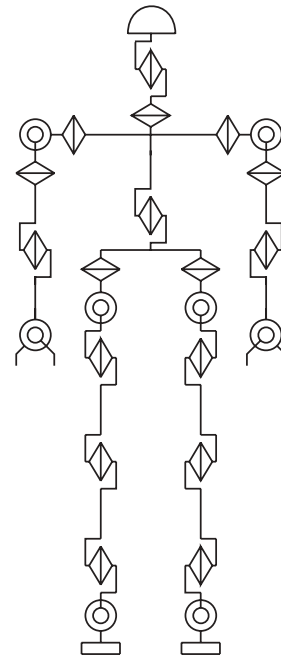


Fig. 2 Arrangement of DOF of the HOAP-2.

ュータは米Dell社製Precision M50 (CPU: Pentium4 Mobile 2.5 [GHz], RAM: 1 [GB], HDD: 60 [GB], Video: nVIDIA Quadro4 Go 700), ジョイパッドはLogicool社製WingMan GAMEPAD EXTREMEである．またシステムの構築にあたっては，以下のような仕様を作成した．

- 有線操縦と無線操縦に対応すること
- 両操縦方式において操作方法が統一されていること
- ジョイパッドの十字キーの上方向が押されている間だけ前進歩行が継続可能なこと
- 歩行しながらの旋回が可能こと(十字キーの斜め上方向に対応)
- 足踏みしてその場で転回が可能なこと
- 特定の動作をボタンに割り当てられること

### 3.2 有線操縦時の構成

有線操縦時の構成をFig. 3に示す．有線操縦時には，動作指令コンピュータとロボット本体の通信にUSBを用い，高速かつリアルタイムの通信が可能である．ロボット本体の電源には外部電源(DC

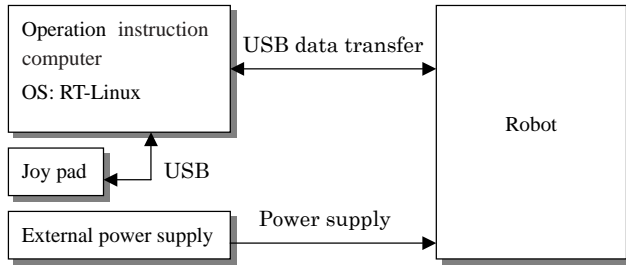


Fig. 3 Configuration in the wired control mode.

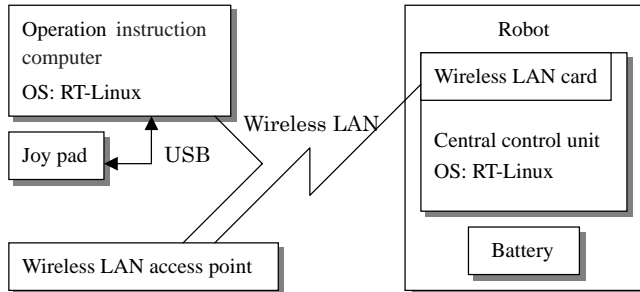


Fig. 4 Configuration in the wireless control mode.

24 [V])を用いる．動作指令コンピュータのOSはRT-Linux version 3.2 pre1である．なおジョイパッドは有線時も無線時もインタフェース用コンピュータに接続される．有線操縦時では，ロボットはUSBデバイス的一种と見なすことができる．

### 3.3 無線操縦時の構成

無線操縦時の構成をFig. 4に示す．無線操縦時では，インタフェース用コンピュータとロボットの中央制御ユニットとの間で無線LANを用いて通信を行う．ロボットの電源には内部バッテリー(DC 24 [V])を用いる．動作指令コンピュータについては，本操縦システムでは有線操縦時のコンピュータを無線時にも用いている．ロボットの中央制御ユニットのOSはRT-Linux version 3.1である．

## 4. インタフェース

本操縦システムはインタフェースと動作生成システムから成る．まずインタフェースについて記す．インタフェースはインタフェースソフトウェア

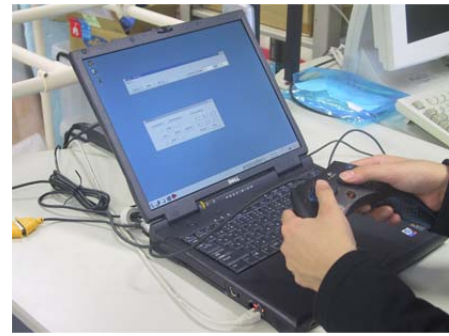


Fig. 5 Man-machine interface.



Fig. 6 Graphical joy pad emulator.

とジョイパッドからなる．

インタフェースの外観をFig. 5に示す．インタフェースソフトウェアはジョイパッドより与えられる指令を，動作生成システムへLAN経由で送信することが主な役目である．なお指令値はロボットの軌道ではなく，ジョイパッドの十字キーの方向，各ボタンのON/OFF情報などである．有線時は動作生成システムも同じコンピュータ上で実行されるが，インタフェースソフトウェア側でlocalhostへソケットを接続する．よって有線時と無線時のソフトウェアの変更点は，ソケットの接続先だけである．

インタフェースソフトウェアは米Borland社製のLinux用統合開発環境“Kylux 3”を用いて作成した．GUIを採用しているため，直感的な操作・設定が可能である．さらにジョイパッドがない環境でも操作できるよう，ジョイパッドのGUIエミュレータも搭載している．その外観をFig. 6に示す．

## 5. 動作生成システム

### 5.1 機能と動作

動作生成システムはインタフェースソフトウェアから操作指令を受信し、あらかじめ用意したプリミティブな動作パターンに対応付け、動作軌道を指令する機能を持つ。ここでいう動作軌道とは、あるまとまりを持った動作に対する目標関節角度のデータである。動作生成システムは、実時間で次々と動作軌道を連結していく。あらかじめ用意されている動作軌道はシームレスに繋がるよう配慮して作ってある。連結された動作の例として、

- 1) 右前方へ歩き出す
- 2) そのまま歩き続ける
- 3) 歩き終わる

などがある。

### 5.2 原理と構成

まず本研究にて考案したアルゴリズムの基礎原理について記す。

現在本操縦システムは平坦な床での運用を想定しているが、そのような環境下における移動は任意の距離で行われる。よって移動する動作は任意の距離だけ続かなければならない。しかし例えば前進動作はひとたび歩行を開始すれば、基本的に左右の足を交互に出す動作の繰り返しとなる。その後、歩行終了動作を行って前進は終了すると見なせる。

したがって歩行動作は

- 1) 開始動作 (歩行開始)
- 2) 継続動作  $\times n$  (歩行継続)
- 3) 終了動作 (歩行終了)

という3種類のプリミティブな動作に分解できる。よって指令が出されている間、継続動作を続け

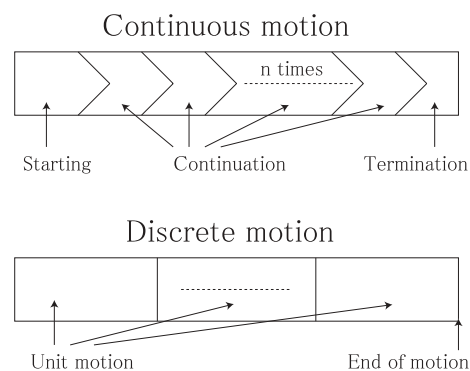


Fig. 7 Concept of the continuous motion and the discrete motion.

ば、任意の時間歩行することが可能であることがわかる。

このように任意の時間継続される部分を持つ動作を連続性動作と呼ぶこととする。厳密に言えばヒューマノイドロボットが任意の距離を移動するためには、歩幅は可変である必要がある。しかしながら本動作生成システムは簡略化のため歩幅を固定し、 $n$ は整数で与えられるものとして製作した。

連続性動作のような、必要な時間だけ継続される動作がある一方で、その動作だけで完結し、それらを結合することで操作指令を実現できる運動がある。これを離散性動作と呼ぶことにする。

連続性動作と離散性動作の概念を図に表すと、Fig. 7のようになる。

例えば前進という動作は動歩行のような連続性動作としても、ステップの連続のような離散性動作としても実現できる。ある動作を連続性動作と離散性動作のどちらでも実現可能であるならば、連続性動作のほうが人間のような自然な動きになるが、動力学を考慮する必要がある。

### 5.3 動作計画

動作計画は連続性動作と離散性動作の両方が実現できるようなものでなければならない。動作計画に際して、各動作への遷移の起点となる初期状態は“膝を曲げた直立姿勢”とした。膝を曲げた直

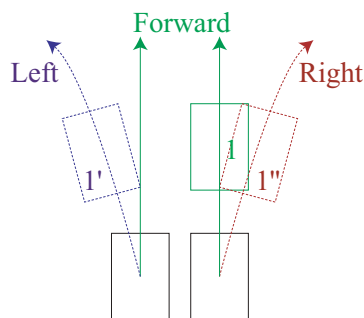


Fig. 8 First step of starting of walk.

立姿勢は静的に安定であり，外乱に対してロバストである．

ここでは前進歩行，旋回歩行を連続性動作として，その場での転回，バイバイ，お辞儀などを離散的動作として実装する．その場で転回する動作に関しては，一定角度の転回を必要に応じて繰り返すことによって実現する．連続性動作として実装する歩行は，歩行開始，歩行継続，歩行終了の三つに分離し，それぞれの動作パターンを用意する．歩行開始動作と歩行継続動作は，前進・右旋回・左旋回の三つずつを用意した．これにより前方はもちろんのこと，左右に向けて歩き出すことも，歩きながら旋回することも可能となる．なお歩行終了動作は旋回を行わないものとした．

さらに歩きながら旋回する場合，遊脚時に遊脚を前へ出して前進を継続するか，遊脚側に出して旋回を行うか判断するものとした．人間は遊脚を支持脚側に出しての旋回ができるが，小型のヒューマノイドロボットには構造的に難しいためこのような対処を行っている．

これらを図で表すと，歩行開始時の一步目の動作はFig. 8，歩行継続時の動作はFig. 9 のようになる．例えばFig. 9 で左旋回歩行は1 → 2' → 3'，右旋回歩行は1 → 2 → 3'' → 4''となる．

#### 5.4 システムへの実装

以上をまとめると，移動に関しては次の九つのプリミティブ動作を用意し，これらを組み合わせ

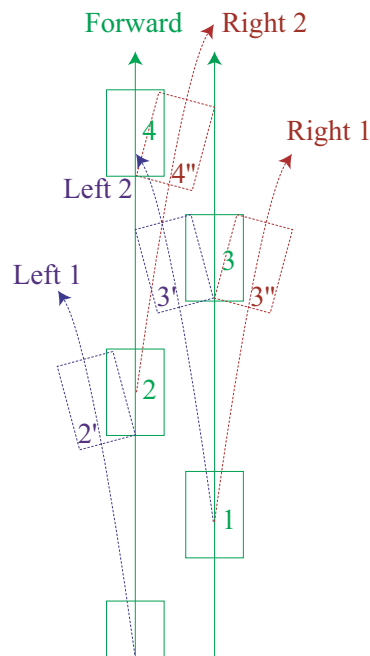


Fig. 9 Motion of walk continuation.

ることにより，操作指令に沿うように動作を生成する．なおいずれの動作軌道も右足と左足を一回ずつ出して一つの動作とする．

- 連続性動作

- 1) 前進歩行開始
- 2) 右旋回歩行開始(15[deg])
- 3) 左旋回歩行開始(15[deg])
- 4) 前進継続
- 5) 右旋回歩行(15[deg])
- 6) 左旋回歩行(15[deg])
- 7) 歩行終了

- 離散性動作

- 1) その場右転回(22.5[deg])
- 2) その場左転回(22.5[deg])

また移動以外の動作に関しては，お辞儀とバイバイを実装している．

## 6. システム運用

構築した操縦システムによってHOAP-2を操縦している写真をFig. 10 からFig. 15 に記す．

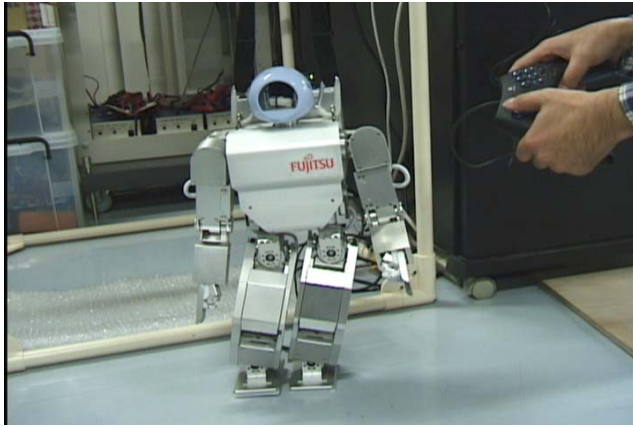


Fig. 10 Starting of walk (front view).

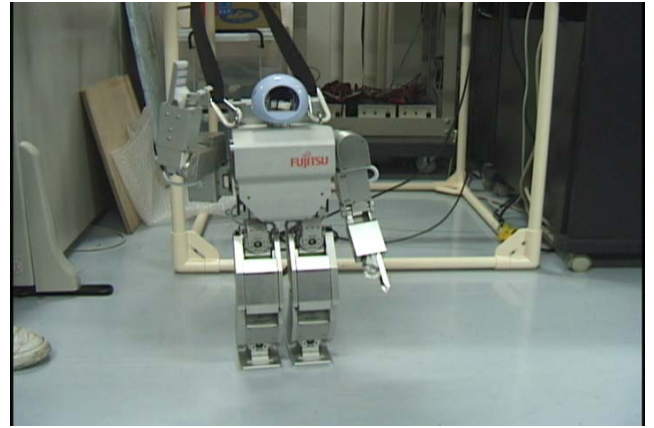


Fig. 13 Bye-bye.

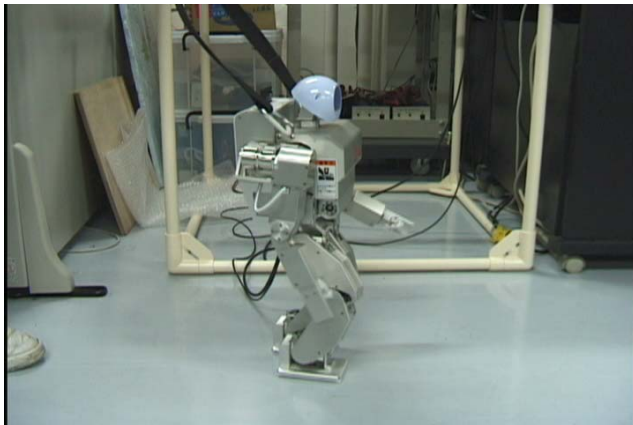


Fig. 11 Starting of walk (side view).

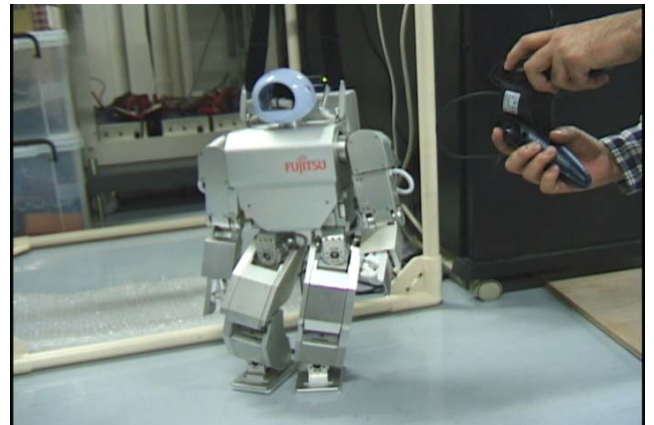


Fig. 14 Rotation to right.

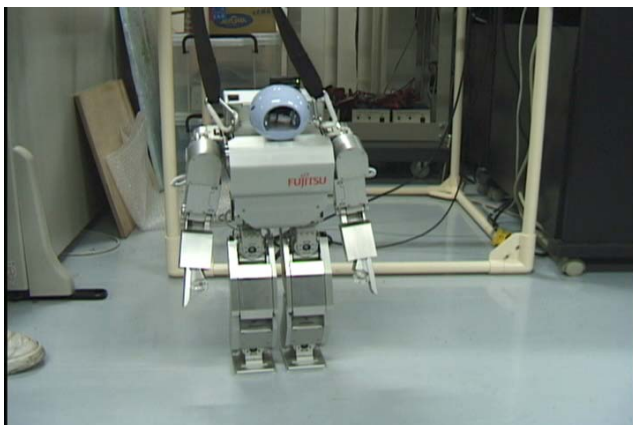


Fig. 12 Bow.

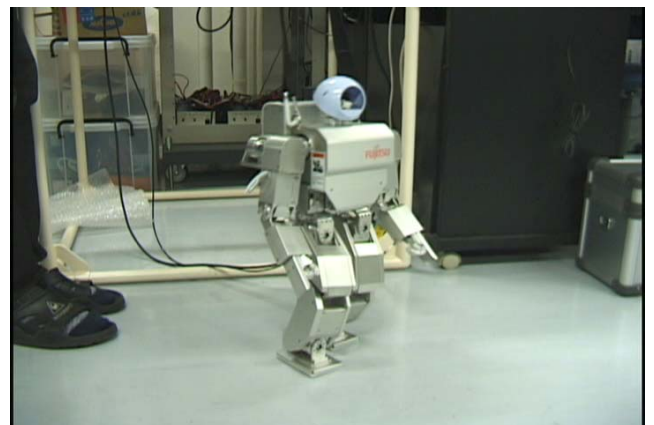


Fig. 15 Walk turning to left.

## 7. おわりに

本論文では、小型ヒューマノイドロボットHOAP-2の操縦システムと、実時間動作生成システムの原理および開発について述べた。更なる高機能化のために、次のような機能拡張を予定している。

- ZMP，加速度，速度情報を利用した制御
- 動的安定化
- 動力学シミュレータの開発
- ロボットの3Dモデルの描画
- アーム，ハンドの操作システムの開発
- 自律行動プログラムの作成

## 参考文献

- 1) “小型ヒューマノイドロボット「HOAP-2」取扱説明書”，富士通オートメーション株式会社，2003．