

プロアクティブヒューマンインターフェースの研究

- 第2報 ヒューマノイド型コミュニケーションロボット PICO の運動制御 -

大政紘映 倉爪亮 田中俊太郎 長谷川勉

Study on Proactive Human Interface - Motion Control of Humanoid-type Communication Robot, PICO-

*Hiroaki OMASA, Ryo KURAZUME, Shuntaro TANAKA, Tsutomu HASEGAWA

Abstract— We are conducting research on “Embodied Proactive Human Interface”. The aim of this research is to develop a new human-friendly active interface based on a new driving mechanism named “Proactive Interface” and a physical devise using robot technology. This paper introduces the developed humanoid-type communication robot for proactive interface named “PICO” and its motion control system using built-in sensors. Posture control, preview control, ZMP feedback control, and angular acceleration compensation using the ZMP Jacobi matrix are implemented for stable whole body motion control.

Key Words: Active communication, Humanoid, Walking control, Sensor fusion

1. はじめに

我々は「プロアクティブヒューマンインターフェース」の研究を行っている [4]。この研究の目的は，“人間からの詳細なデマンドに寄らない新たな駆動原理（プロアクティブ技術）”に基づき，従来の仮想メディアだけではなく“実体としてのロボット技術”の活用することで，人とコンピュータシステムに存在する垣根を低減化し，様々な人にとって自然で使いやすいヒューマンインタフェースの枠組みを提供しようというものである。

本研究の一環として，我々は新たにヒューマノイド型ロボット HOAP-II をベースとしたアクティブコミュニケーションロボット”PICO”を試作した（図1）。HOAP-II は富士通オートメーション（株）が開発した身長 50cm，体重 7 kg の小型，軽量のオープンアーキテクチャロボットである。このロボットは，内部インタフェース情報が公開されており，ユーザが自由にプログラム可能である。HOAP-II は腕に 4 個ずつ，脚に 6 個ずつ，さらに腰，手の開閉に 1 個ずつ，首に 2 個の計 25 個の関節を有する。また，胴体部に角速度，加速度センサ，足裏に足裏力センサが取り付けられている。

本報告では，これら角速度，加速度，足裏力センサを用いたコミュニケーションロボット PICO の姿勢制御系，安定歩行のための ZMP 制御系などを説明し，姿勢制御実験，安定歩行実験の様子を紹介する。

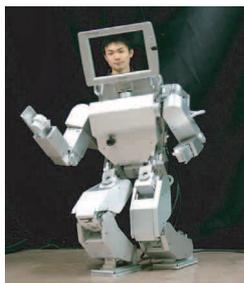


Fig.1 プロアクティブコミュニケーションロボット”PICO”

2. 姿勢制御

PICO に搭載した加速度，角速度センサを用いて姿勢制御を行った。一般に角速度センサのみを用いて，角速度を積分し姿勢を求めた場合には，積分誤差の蓄積が問題となる。また，加速度センサのみを用いてロボットの胴体座標系からみた重力方向をもとに姿勢を求めた場合には，ロボットが加速度運動中は正確な重力方向を示すことができない。そこで，加速度センサ，角速度センサから得られた値をカルマンフィルタを用いて統合し，その出力として姿勢の推定値をもとに姿勢制御を行う。

2-1 カルマンフィルタによる姿勢推定

カルマンフィルタを用いた姿勢推定について説明する。まず，加速度センサを用いてロボットの胴体座標系 Σ_B における単位重力方向ベクトル g_b を求める。次に，慣性座標系 Σ_I における単位重力方向ベクトル g_i と，胴体座標系における単位重力方向ベクトルとの関係が，回転行列を用いて

$$E^{Ri} E^{Pj} E^{Yk} \begin{pmatrix} {}^B g_x \\ {}^B g_y \\ {}^B g_z \end{pmatrix} = {}^I \hat{g} = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \quad (1)$$

と書けるので，この式を変形して

$$\begin{aligned} E^{Yk} \begin{pmatrix} {}^B g_x \\ {}^B g_y \\ {}^B g_z \end{pmatrix} &= E^{-Pj} E^{-Ri} \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \\ &= \begin{pmatrix} \cos R \sin P \\ -\sin R \\ -\cos R \cos P \end{pmatrix} \end{aligned} \quad (2)$$

となる。慣性座標系における単位重力方向ベクトルは既知であり，また Yaw 角は既知であると仮定すると，この式から Roll 角，Pitch 角が得られる。この Roll

角, *Pitch* 角の情報と角速度センサから得た *Roll* 角, *Pitch* 角回りの角速度を内部状態とした状態方程式をもとにカルマンフィルタ [1] を構成し, 角度・角加速度の推定値を得た.

図 2 に実際の歩行時に測定された姿勢を示す. 黄線は加速度センサより求めた姿勢, 青線は角速度センサより求めた姿勢, 赤線はカルマンフィルタを用いて推定した姿勢である. カルマンフィルタを用いて 2 つの

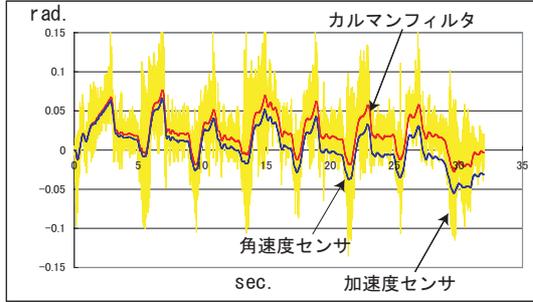


Fig.2 カルマンフィルタによる姿勢推定

センサ情報を統合した結果, 積分誤差の蓄積や歩行時の振動による誤差が生じにくいことが確認できた.

2.2 姿勢制御系

上述した姿勢センサを用いて現在の上半身姿勢を推定し, 目標姿勢との誤差に対する PID 制御系を構築した.

3. ZMP 制御

入力を本体の重心の躍度, 出力を ZMP とし, 式 (3), 式 (4) で表される動的離散時間システムを考える.

$$\begin{pmatrix} x(k+1) \\ \dot{x}(k+1) \\ \ddot{x}(k+1) \end{pmatrix} = \begin{pmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x(k) \\ \dot{x}(k) \\ \ddot{x}(k) \end{pmatrix} + \begin{pmatrix} \frac{\Delta t^3}{6} \\ \frac{\Delta t^2}{2} \\ \Delta t \end{pmatrix} u(k) \quad (3)$$

$$y(k) = \begin{pmatrix} 1 & 0 & -\frac{z(k)}{g} \end{pmatrix} \begin{pmatrix} x(k) \\ \dot{x}(k) \\ \ddot{x}(k) \end{pmatrix} \quad (4)$$

ただし, $x(k), u(k), y(k)$ はそれぞれ, 時刻 k における本体の重心位置, 躍度入力, 単質点を仮定した時の ZMP の位置である.

この動的システムにおいて, 実 ZMP を目標 ZMP 軌道に一致させ, 安定な歩行を実現するために, 躍度予測制御 [2], ZMP フィードバック, ZMP ヤコビ行列を用いたフィードバック制御を組み合わせ, 安定歩行を実現した.

3.1 躍度予測制御

躍度予測制御は梶田らによって提案された ZMP 制御方式である. 本手法では, 一定時間先までの ZMP 目標値を利用して逐次的に運動パターンを生成していく.

制御入力は加速度をさらに一階微分した躍度で与えられ, そのときの入力は次式で決定される [3].

$$u(k) = F_e \sum_{i=0}^k e(i) + F_x x(k) + F_R \begin{bmatrix} R(k+1) \\ R(k+2) \\ \vdots \\ R(k+M_R) \end{bmatrix} \quad (5)$$

ただし, $R(k+1)$ は未来の時刻 $k+1$ における ZMP 目標値である.

3.2 ZMP フィードバック

上述した躍度予測制御は, 単質点モデルを仮定しているため, 多質点である実ロボットでは正確な ZMP は実現できない. そこで多質点モデルでの現在の ZMP 推定値と目標 ZMP との誤差, 及び足裏の力センサから得られる実 ZMP と目標 ZMP との誤差を用いてフィードバック制御を行った. この際の入力は次式で決定される.

$$u(k) = F_{e1} \sum_{j=1}^k e_{calc}(j) + F_{e2} \sum_{j=1}^k e_{sen}(j) + F_{e3} e_{calc}(k) + F_{e4} e_{sen}(k) \quad (6)$$

ただし, $e_{calc}(j)$ は時刻 j における目標 ZMP と多質点モデルでの ZMP との誤差, $e_{sen}(j)$ は目標 ZMP と足裏の力センサから得られる実 ZMP との誤差, $x(k)$ は時刻 k における重心位置である. また, ゲイン $F_{e1}, F_{e2}, F_{e3}, F_{e4}$ はエラーシステムに関する最適レギュレータ問題から求めた.

3.3 ZMP ヤコビ行列を用いたフィードバック制御

ロボットの各リンクの質量を考慮した多質点モデルにおける ZMP は, 次式で得られる.

$$x_{zmp} = \frac{\sum_{i=1}^N m_i (r_{ix} (\ddot{r}_{iz} + g) - r_{iz} \ddot{r}_{ix}) - I_i \dot{w}_{iy}}{\sum_{i=1}^N m_i (\ddot{r}_{iz} + g)} \quad (7)$$

$$y_{zmp} = \frac{\sum_{i=1}^N m_i (-r_{iy} (\ddot{r}_{iz} + g) + r_{iz} \ddot{r}_{iy}) - I_i \dot{w}_{ix}}{\sum_{i=1}^N m_i (\ddot{r}_{iz} + g)} \quad (8)$$

式 (7), 式 (8) の分母を一定であると仮定し, 加速度項に注目して式を変形する.

$$X_{zmp} = A_{zmp} \ddot{P} + B \quad (9)$$

が得られる. ただし

$$\ddot{P} = \begin{pmatrix} \ddot{r}_{1x} & \ddot{r}_{1y} & \ddot{r}_{1z} & \ddot{r}_{2x} & \dots & \ddot{r}_{Nz} \end{pmatrix}^T \quad (10)$$

また, 接地した足裏が水平に静止していると仮定すると, 関節角度を $(\phi_1 \dots \phi_N)$ とした時, ロボットの各リンクの位置 P は

$$P = f(\phi_1 \dots \phi_N) \quad (11)$$

と表せる. 両辺を微分するとヤコビ行列 J を用いて

$$\dot{P} = J \dot{\Phi} \quad (12)$$

が得られ、さらに微分すると

$$\dot{P} = J\ddot{\Phi} + \dot{J}\dot{\Phi} \quad (13)$$

が得られる。(9),(13)式より ZMP を各関節の加速度で表現することができる。

$$X_{zmp} = A_{zmp}J\ddot{\Phi} + A_{zmp}\dot{J}\dot{\Phi} + B \quad (14)$$

従って上式の第 2 項, 第 3 項の真値との差が微小であると仮定すると, ZMP の誤差を次式で表現できる。

$$\Delta X_{zmp} = A_{zmp}J\Delta\ddot{\Phi} = J_{zmp}\Delta\ddot{\Phi} \quad (15)$$

J_{zmp} を ZMP Jacobi 行列と呼ぶ。両辺に J_{zmp} の擬似逆行列 J_{zmp}^+ をかけると

$$\Delta\ddot{\Phi} = J_{zmp}^+\Delta X_{zmp} \quad (16)$$

$$J_{zmp}^+ = J_{zmp}^T(J_{zmp}J_{zmp}^T)^{-1}$$

が得られる。これより、得られた $\Delta\ddot{\Phi}$ を関節加速度指令値に加えることで ZMP の誤差を低減することができる。

4. 実験結果

4.1 姿勢制御実験

まず、姿勢制御実験を静止時、歩行時に対して行った。静止時における実験では、ロボットを板の上に乗せ、その板の端を持ち上げロボットの接地面を傾斜させ、姿勢制御の有無における上体姿勢を比較した。実験の様子を図 3 に示す。姿勢制御を行わない場合にはロボットは転倒したが、姿勢制御を行った場合には姿勢センサから得られた情報を基に、上体の姿勢を調整して転倒を回避することができた。次に、歩行時における姿

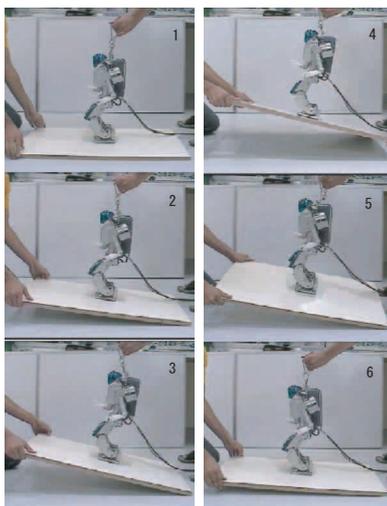


Fig.3 姿勢制御実験

勢安定化実験を行った。歩行周期は 8.0s, 歩幅 0.1m, Duty 比 0.75 であり、水平な台上を歩行させた時の上体姿勢を測定した。胴体座標系の定義を図 4 に、実験結果を図 5, 図 6 に示す。これより x 軸回り, y 軸回りのグラフとともに姿勢制御を行った場合の方が振幅が小さいことから、上体の振動が抑えられていることが

わかる。また、図 6 に注目すると姿勢制御を行わない場合には、バックラッシュの影響で y 軸回り正方向に傾く傾向が見られたが、姿勢制御を行うことによって、その傾向は改善されたことがわかる。

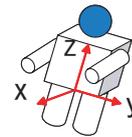


Fig.4 胴体座標系

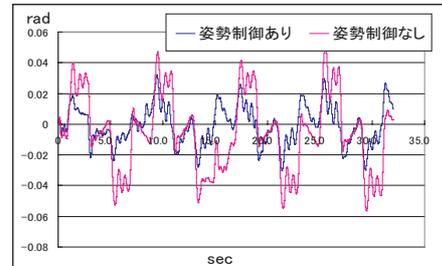


Fig.5 姿勢制御の有無による姿勢の比較 (歩行時, x 軸回り)

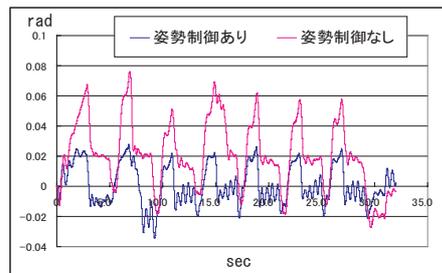


Fig.6 姿勢制御の有無による姿勢の比較 (歩行時, y 軸回り)

4.2 歩行実験

まず、ZMP ヤコビ行列を用いたフィードバック制御のシミュレーション実験を行った。図 7 に予見制御のみを用いた場合と予見制御に ZMP Jacobi 行列を用いたフィードバック制御を加えた場合との比較を示す。図の X 軸方向が進行方向である。これより ZMP Jacobi 行列を用いたフィードバック制御を加えることで目標追従性が向上することが確認された。次に、実際の口

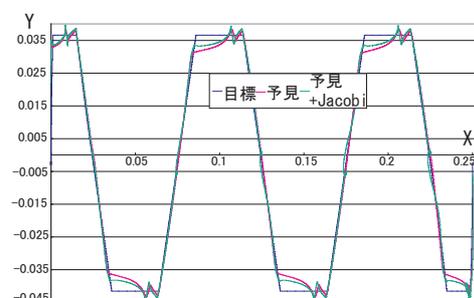


Fig.7 予見制御のみの場合との比較

ポットを用いて、足裏のセンサから実 ZMP を計測し、目標との誤差を用いて式 (6) によりフィードバックを行った。歩行のパラメータは Duty 比 0.75, 歩行周期

8.0s, 歩幅 0.1m である。実験の結果得られた ZMP 軌道を図 8 に示す。これよりフィードバックを行うことで、より目標追従性能が向上することが確認できた。

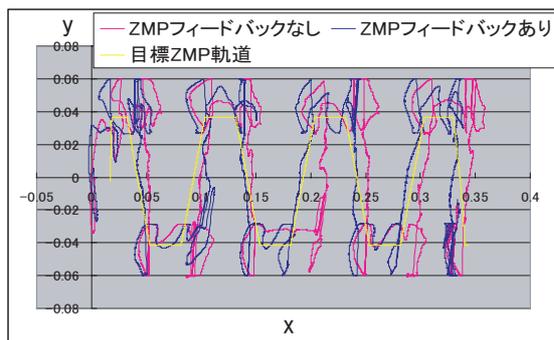


Fig.8 ZMP フィードバックの有無による ZMP 軌道の比較

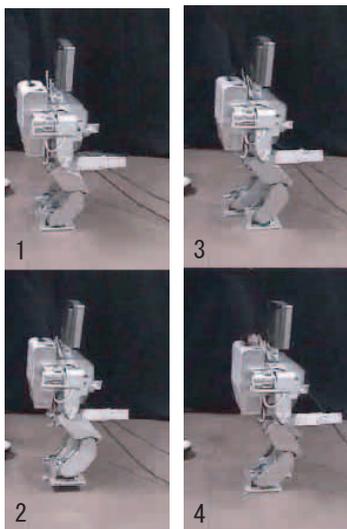


Fig.9 歩行実験 1

次に、ZMP フィードバックを用いた上体姿勢や腕位置

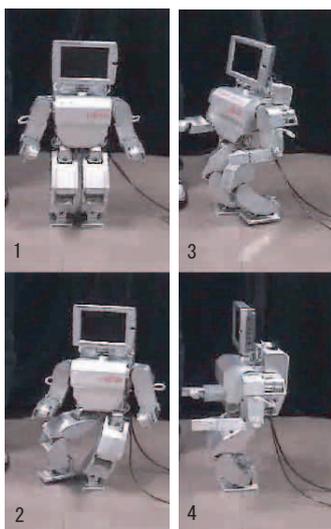


Fig.10 歩行実験 2

変更時の直立動作実験を行った。図 11 にお辞儀動作を、図 12 に人間の腕の動作を再現している様子を示す。ZMP フィードバックを行うことで上体姿勢が大きく変化する場合や腕を急激に動かした場合でも、安定

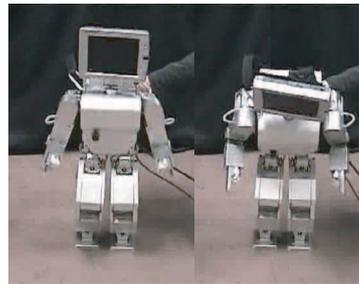


Fig.11 お辞儀動作

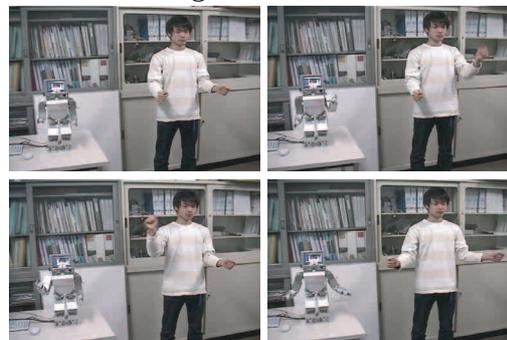


Fig.12 人間動作の模倣実験

な直立動作を実現することができた。また、足裏の力センサを用いた段差踏み越え実験を行った。結果を図 13 に示す。力センサによって段差を検出することで、厚さ 1cm 程度の段差を踏み越えることができた。



Fig.13 段差踏み越え実験

5. まとめ

本報告では開発したプロアクティブコミュニケーションロボット”PICO”に対し、内蔵された加速度、角速度、足裏力センサを用いた姿勢制御、ZMP フィードバック制御の実装法を示した。また、実験によってそれぞれの方法が安定動作を実現する上で有効であることを確認した。

参考文献

- [1] カルマン・フィルター： 有本 卓 著 産業図書
- [2] 梶田, 金広, 比留川： 予見制御理論を応用した 2 足歩行パターン生成, 日本ロボット学会創立 20 周年記念学術講演会予稿集
- [3] デジタル予見制御： 土谷 武士・江上 正 共著 産業図書
- [4] 倉爪, 内田, 長谷川, 谷口：プロアクティブヒューマンインターフェースの研究 - 第 1 報 人間型アクティブインターフェースの開発 -, 日本機械学会ロボティクス・メカトロニクス講演会'04 講演論文集, (2004).
- [5] 倉爪, 米田, 田中, 玉木, 太田, 長谷川：2 足歩行機械の 3D 揺動歩容, 日本ロボット学会誌, Vol.21, No.7, pp.811-818,(2003).