

重力による弾性変位を考慮したヒューマノイドロボット 動作教示システムの開発

○山野光裕 安部賢人 大沼俊一 金子慎一郎 水戸部和久 那須康雄 (山形大学)

Development of a Motion Teaching System for a Humanoid Robot with Consideration of Its Elastic Deflections Due to Gravity

*Mitsuhiro Yamano, Masato Abe, Syun-ichi Ohnuma, Shin-ichiro Kaneko,
Kazuhisa Mitobe and Yasuo Nasu (Yamagata University.)

Abstract— We discuss the development of a motion teaching system for a biped humanoid robot that have elastic deflection at its legs. We describe the modeling of the humanoid robot with flexible legs and a compensation method of elastic deflections in order to avoid its falling down. We develop the motion teaching system includes 3D computer graphics animation using OpenGL to show the motion of the robot in three dimensions and game controller to input the three dimensional coordinates of the robot. The system also calculates and display the center of gravity and zero moment point. Finally, we present an integration plan of the compensation method and motion teaching system.

Key Words: Biped Humanoid Robot, Elastic Deflection, Game Controller, Computer Graphics

1 はじめに

ヒューマノイドロボットは、腕や脚に多くの自由度を有するため、重量が大きくなりやすく、脚部に大きな力、モーメントが加わりやすい。剛性を高めるために脚部の部材を厚くすることは、さらなる重量増加を招き、モータや減速器の大型化による脚への負荷の増大という悪循環に陥る。ヒューマノイドロボットの脚の柔軟性を許容した動作が実現できれば、より安価で容易な設計が行えると考えられる。そのため、筆者らは、弾性変形の影響を考慮して、脚の弾性変位を考慮したモデルを考え、単脚支持期における軌道の補正法を提案した [1]。

一方、ヒューマノイドロボットの自由度の多さを生かすためには、両腕、両脚や胴体の位置や姿勢、動作の同期等の煩雑な条件設定を効率的に行えることが望ましい。また、生成された動作が転倒せずに実現可能なものか評価する必要がある。そこで、筆者らは、OpenGL とゲームコントローラを用いて、効率的に動作を教示し、動作の評価を視覚的にできるシステムを開発した [2]。

開発した動作教示システムは、脚の柔軟性を考慮していなかったため、脚の柔軟性を無視できないロボットに適用すると転倒評価にずれが生じ得る。そこで、教示システムと起動補正計算を統合したシステムの構築を進めている。本稿では、2 章で静的な動作の単脚支持期における軌道補正法、3 章で動作教示システムを示し、4 章で両者を統合したシステムの構築法について述べる。

2 静的な動作の単脚支持期における重心位置誤差の補償

ヒューマノイドロボットの脚は、股関節に 3 自由度、膝に 1 自由度、足首に 2 自由度が配置される例が多く見られる。股関節と足首は、関節の可動範囲が広くなるよう部品同士の干渉を極力避けつつ、3 軸または 2 軸の関節を集中させることになるため、他の場所に比べ、

剛性を高めにくい。減速器にはハーモニックドライブが用いられることが多いが、ハーモニックドライブは、回転方向に柔軟性を有する。また、関節に位置フィードバックを適用して駆動する場合、大きな負荷の加わる関節を駆動するサーボモータの偏差も機構的な弾性変位と同様な影響を生じる。弾性変位により各リンクの位置が変化し、重心やゼロモーメント点が変わるため、転倒への影響を考慮する必要がある。

2.1 モデリング

脚が機構的な柔軟性を持たないロボットについては、藤本と河村により動力学モデルとそれを用いたシミュレーションが報告されている [3]。機構的な柔軟性については、マニピュレータを対象としたモデリング法が提案されている [4]。単脚支持の状態において支持脚の足底が床面に密着している場合、支持脚足底の拘束されていると見なして、マニピュレータの場合と同様な方法により、次のようにまとめることができる。

$$\tau = M_{\theta\theta}(q)\ddot{\theta} + M_{\theta e}(q)\ddot{e} + c_{\theta}(q, \dot{q}) + g_{\theta}(q) \quad (1)$$

$$0 = M_{e\theta}(q)\ddot{\theta} + M_{ee}(q)\ddot{e} + K_{ee}e + c_e(q, \dot{q}) + g_e(q) \quad (2)$$

ただし、 θ は関節角度ベクトル、 e は弾性変位ベクトル、 q は θ と e からなるベクトル、 M_{ij} ($i, j = \theta, e$) は慣性行列、 K_{ee} は弾性係数行列、 c_i ($i = \theta, e$) は遠心力、コリオリ力項、 g_i ($i = \theta, e$) は重力項である。

2.2 補償アルゴリズム

ロボットが単脚で支持されている状態であるとき、式 (1)、(2) をもとにゼロモーメント点を考慮する必要があるが、静歩行時のように慣性力や遠心力、コリオリ力が無視できるような場合、関節角軌道を補正して、重心位置のずれを補償すればよい。静的な動作の際の、単脚支持期における弾性変位は、式 (2) に $\ddot{q} = \dot{q} = 0$ を代入することにより、

$$e = -K_{ee}^{-1}g_e(q) \quad (3)$$

のように求まる.

静的な動作の軌道を生成する際、ロボットが転倒しないように、重心が支持脚の足底面内の中にある必要があるが、弾性変位により、重心位置が変化するため、関節角度を補正して重心位置誤差を補償する必要がある. 関連する研究として、フレキシブルアームの弾性変位による手先位置のずれを補償する Newton-Raphson 法ベースの計算法が Svinin と Uchiyama によって提案されている [5]. 静的な動作の場合は、遊脚足底の位置に加えて重心位置が重要であるため、同様の方法で遊脚足底の位置と重心位置が補償される方法を考える. 重心位置の床面内の座標からなるベクトルを $\mathbf{p}_c(\boldsymbol{\theta}, \mathbf{e})$, 遊脚足底の位置, 姿勢の中から, 補償する成分を選んだベクトルを $\mathbf{p}_f(\boldsymbol{\theta}, \mathbf{e})$ とし, $\mathbf{p} = [\mathbf{p}_c^T \ \mathbf{p}_f^T]^T$ とする. 補正前の関節角軌道を $\boldsymbol{\theta}_0$ とし, \mathbf{e} の影響を補償し, $|\mathbf{p}(\boldsymbol{\theta}_0, \mathbf{0}) - \mathbf{p}(\boldsymbol{\theta}_k, \mathbf{e}_k)|$ が十分小さくなるように, $\boldsymbol{\theta}$ を次式の収束計算により補正する [1].

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + (\mathbf{J}_{p\theta} + \mathbf{J}_{pe}\mathbf{J}_g)^+ \times (\mathbf{p}(\boldsymbol{\theta}_0, \mathbf{0}) - \mathbf{p}_k(\boldsymbol{\theta}_k, \mathbf{e}_k)) \quad (4)$$

ただし,

$$\mathbf{J}_{p\theta} = \frac{\partial \mathbf{p}}{\partial \boldsymbol{\theta}} \quad (5)$$

$$\mathbf{J}_{pe} = \frac{\partial \mathbf{p}}{\partial \mathbf{e}} \quad (6)$$

$$\mathbf{J}_g = \frac{\partial (-\mathbf{K}_{ee}^{-1} \mathbf{g}_e)}{\partial \boldsymbol{\theta}} \quad (7)$$

とし, 肩字の $+$ は一般化逆行列を表す.

2.3 数値例および実験

高さ 1.25 [m], 重量 30.5 [kg] のヒューマノイドロボット Bonten-Mar II [6] を用いた数値例と実験を示す. ロボットの足底面の形状は, 通常前後よりも左右の方が短く, 単脚支持期においては, ラテラル面内の転倒に対する余裕が小さくなるため, ラテラル面内の運動について考える. Bonten-Mar II の写真とモデルを Fig. 1 に示す. ロボットの質量が, 上半身の質量 m_0 , 右脚, 左脚の質量 m_1, m_2 に集中していると近似し, 支持脚の足首と股関節付近に柔軟性を持つと仮定したモデルとした. 足首関節, 股関節の関節角を θ_a, θ_b , 足首部および股関節部の弾性変位角を ϕ_a, ϕ_b とし, $\boldsymbol{\theta} = [\theta_a \ \theta_b]^T$, $\mathbf{e} = [\phi_a \ \phi_b]^T$ とし, \mathbf{p}_c を重心位置の x 座標, \mathbf{p}_f を遊脚脚先の高さとして, 式 (4) を適用した.

上半身と脚の質量を $m_0 = 14.5$ [kg], $m_1 = m_2 = 8.0$ [kg] とし, 弾性係数行列を $\mathbf{K} = \text{diag}[500, 500]$ と設定した. 補正前の関節角度は, 静歩行の単脚支持期を想定し, θ_a が 5° と 15° の間を往復するように与え, $\theta_b = -\theta_a$ とした. Fig. 2 に足首関節の真下を原点としたロボットの重心の x 座標を示す. Fig. 2 の rigid は柔軟性を無視した剛体モデルにより計算した重心軌道であり, non-compensated は柔軟性を考慮した弾性体モデルで計算した補償前の重心軌道である. compensated は, 式 (4) により補正した関節角度を与えた補償後の重心軌道であり, 破線と同様に弾性体モデルで計算されている.

弾性体モデルによる補償前の重心軌道は, 剛体モデルの重心軌道と比べ, おおよそ原点からより遠ざかる

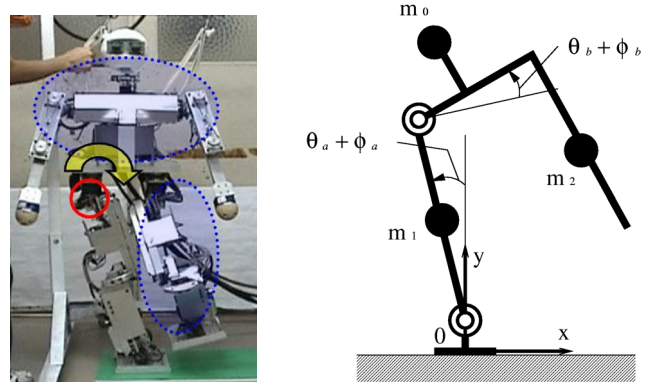


Fig. 1: Photograph and model of the robot.

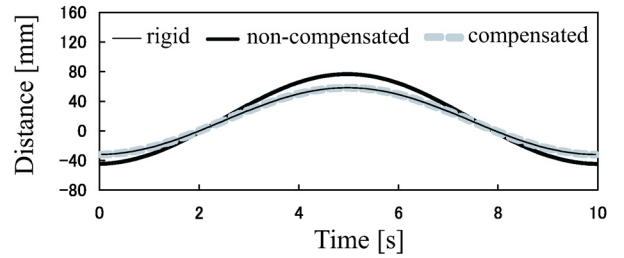
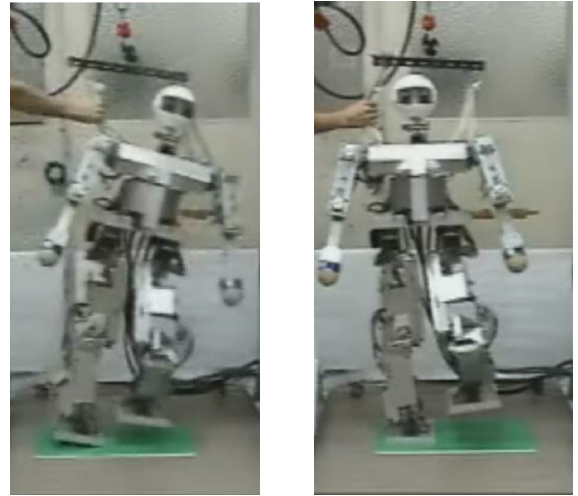


Fig. 2: Center of gravity.



(a) non-compensated (b) compensated

Fig. 3: Experiment.

方向に変化している. 重心が原点から遠ざかるほど, 足底面内から出やすくなるため, 弾性変位の影響で, ロボットが静的な安定性を保ちにくくなっていることを示している. 補正後の重心軌道は, Fig. 2 より, 剛体モデルの重心軌道とほぼ一致していることから, 重心位置の弾性変位による変化を十分補償できていることがわかる. 補正前, 補正後の軌道を Bonten-Mar II に与える実験を行ったところ, 補正前の軌道では前は Fig. 3 (a) のように転倒したが, 補正後の軌道では Fig. 3 (b) のように転倒することなく動作でき, 軌道補正の有用

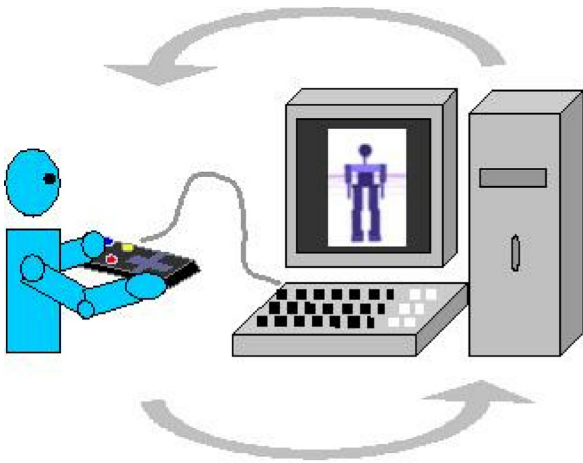


Fig. 4: Motion teaching system.



Fig. 5: Game controller.

性を確認できた。

3 動作教示システム

ヒューマノイドロボットに様々な動作を教示するためのシステムとして、3次元グラフィックスを用いた動作の教示と動作の評価支援の機能を有するシステム [2] を用いる。このシステムは、ユーザーがロボットの運動学や力学を知っている必要がなく、直観的に手先や脚先の位置を、PCに接続されたゲームコントローラから与えることができる。動作軌道は、教示に基づいて生成され、生成された軌道によりロボットが転倒するかどうかについての評価指標が提示される。

動作教示システムのハードウェア構成を Fig. 4 に示す。PCは、OpenGLにより、3次元空間でヒューマノイドロボットのアニメーションを提示する。マウスやキーボードの操作により、アニメーションのズームイン、ズームアウトや視点変更が、自由に行える。

ロボットの 手先、脚先および胴体の 3次元の座標を教示する際、マウスやキーボードでは、3次元のデータを入力しにくい。ため、Fig. 5 のようなゲームコントローラを用いる。ゲームコントローラには、X, Y, Z についてそれぞれ正負のボタンを割り当てているため、X, Y, Z の座標を、独立かつ同時に入力することができる。PC 画面上のアニメーションはゲームコントローラに

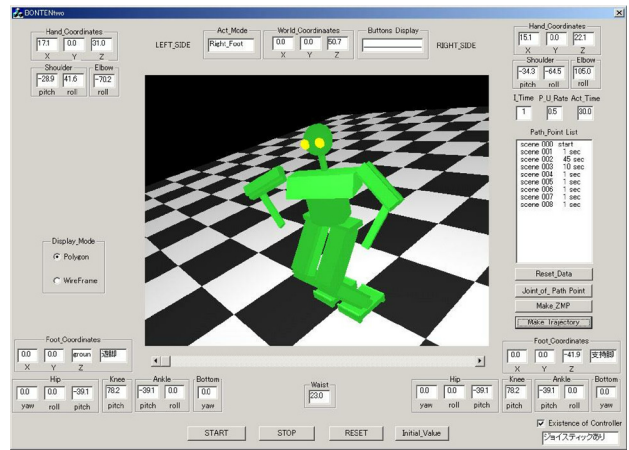


Fig. 6: Motion teaching program.

よる操作によってリアルタイムに動作するため、ユーザーはロボット手先や脚先の位置を直観的に教示することができる。動作教示プログラムのスクリーンショットを Fig.6 に示す。プログラムは、動作教示と動作評価のモードを持つ。それぞれについて 3.1 節、3.2 節で述べる。

3.1 動作の教示

教示プログラム内のモデルは、Bonten-Maru II と同様、各腕 3 自由度と各脚 6 自由度、胴体中央部で腰をひねる自由度を有する。動作をいくつかのシーンに分割し、シーンごとに次のような操作により教示する。

- A) 左右の手先、脚先および胴体の中から教示するものを一つを選択する。
- B) 左右の脚の一方を支持脚、もう一方を遊脚に設定しておく。
- C) 選択された手先、脚先または胴体の 3次元座標を教示するため、ゲームコントローラの X, Y, Z の正または負に対応するボタンを一つだけまたは二つ以上同時に押して、座標を変化させる。A) において胴体を選択された場合は腰をねじる角度を増減させる。手先位置の場合は、胴体上部を基準に操作され、遊脚脚先位置の場合は胴体下部を基準に操作される。支持脚脚先が選択された場合、脚先は床面上を動かさず、胴体が脚先の操作と反対側の方向に動く。
- D) 左右の腕、脚および胴体について A)~C) を繰り返す。
- E) シーンの動作時間を設定する。

分割された各シーンについて、A)~E) を行い、全てのシーンにおける手先、脚先位置、腰の回転角度の教示された後、コンピュータは、シーンごとに逆運動学および多項式補間を用いて、ロボットの関節角軌道を計算し、その結果を結合して動作全体の関節角軌道データを得る。

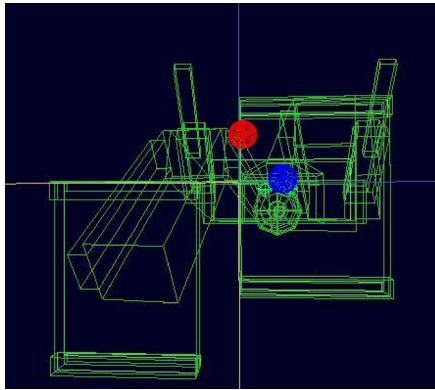


Fig. 7: Motion evaluation.

3.2 動作の評価

教示, 生成された軌道により, ロボットが転倒せずに動作できるか評価を行う必要がある. そのために, 重心およびゼロモーメント点を計算し, 表示する機能が, システムに統合されている. 生成された軌道どおりに各関節を駆動した場合, ゼロモーメント点が接地している足の凸包内にあることが動作時において, ロボットが転倒しない条件となる.

生成された軌道について, コンピュータにより, 動作の最初から最後まで各時刻のゼロモーメント点が計算される. 動作評価の際は, Fig.6 中央の教示用アニメーションが Fig.7 のような評価支援用のアニメーションに切り替えられる. Fig.7 は, ヒューマノイドロボットをワイヤフレームで表示し, 視点を足底の下に設定したものである. 軌道評価支援のため, 生成された軌道がアニメーションで再生されると同時に, 重心とゼロモーメント点が床面上に表示される. ゼロモーメント点が接地している足の凸包内にあるか目視で確認することにより, 転倒せずに実行できる動作であるかについてユーザーが評価することができる.

4 動作教示と位置誤差補償と統合

現在構築中のシステムの構成を Fig. 8 に示す. このシステムでは動作教示と位置誤差補償を次のように統合して用いる.

腕, 脚や胴体のすべての姿勢が決まらなければロボットの各部に加わる重力が求められないため, まず, 柔軟性を考慮しない剛体モデルで動作教示を行う. 本稿では, 慣性力, 円心力, コリオリ力の影響が無視できるほど遅い動作の単脚支持期について考える. ゼロモーメント点を評価する際, 遅い動作であれば, ゼロモーメント点と重心はほぼ一致するため重心で代替できる. その後, 教示され生成された軌道の各時刻の重力による位置誤差の補償のため, 軌道を補正する.

軌道教示のプログラムは, Microsoft Visual C++ で開発し, 位置誤差補償の計算は, MathWorks 社の MATLAB で行う. 両者を移植することなく, 統合するため, MATLAB Engine を用いて, データを受け渡しながら, 計算するシステムを作成中である. Fig. 8 の矢印の順に, ユーザーは, まず手先や脚先の軌道の教示を行い, 重心位置の動きをアニメーション表示させて

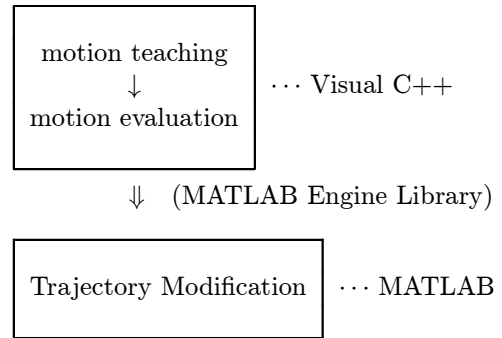


Fig. 8: System of motion teaching and compensation.

確認する. そこで生成された軌道データは MATLAB Engine を介して, MATLAB へ送られ, 弾性変位による重心位置を補償するための軌道補正の計算が行われる.

5 おわりに

本稿では, ヒューマノイドロボットの重力による弾性変形の影響を考慮した軌道補正法と, ヒューマノイドロボットの動作を簡単に教示, 評価できるシステムについて示し, 両者の統合法について述べた. 今後, 統合されたシステムを構築して, 脚の柔軟性を考慮しながら, さまざまな動作を容易に生成できるようにし, さらに, 慣性力の影響が無視できない速さの運動や両脚支持期にも対応できるよう拡張していく予定である.

参考文献

- [1] 山野光裕, 水戸部和久, 那須康雄, 金子 慎一郎: “ヒューマノイドロボットの脚の柔軟性を考慮した軌道補正” 第 22 回日本ロボット学会学術講演会 CD-ROM, 2L13, 2004.
- [2] M. Yamano, Y. Nasu, M. Abe, S. Ohnuma, S. Kaneko and K. Mitobe: “A Motion-Generation System for a Humanoid Robot Using Computer Graphics,” Eighth Int. Conf.on Manufacturing and Management, 2004 (to be presented).
- [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”, Vol.15, No.6, pp. 857-863, 1997.
- [4] 近野敦, 内山勝: “ホルツァ法に基づくフレキシブルマニピュレータの動力学モデリング”, 日本ロボット学会誌, Vo. 12, No. 7, pp. 1021-1028, 1994.
- [5] M. SVININ and M. UCHIYAMA: “Contribution to Inverse Kinematics of Flexible Robot Arms,” JSME Int. Journal, Series C, vol. 37, no. 4 , pp. 755-764, 1994.
- [6] 山野光裕, 那須康雄, 金子慎一郎, 水戸部和久: “ヒューマノイドロボットの脚と腕を用いた障害物乗り越え動作”, 日本ロボット学会創立 20 周年記念学術講演会予稿集 (CD-ROM), 1C23, 2002.