

Java3D を用いた人型二足歩行ロボットの 動作生成システムの設計および実装

酒井 純一[†] 大林 真人[‡] 西山 裕之[†] 溝口 文雄[†]

東京理科大学理工学部経営工学科[†]

東京理科大学情報メディアセンター[‡]

1. はじめに

近年、様々な形状のロボットが登場し、中でも人型二足歩行ロボットの登場は多くの関心を集めている。また以前と比較してロボットが商品化され、人間と同様な動作をすることから、様々な用途への適用が考えられエンドユーザがロボットに触れる機会も増加する傾向にある。しかし人型二足歩行ロボットは人の形状をしていることから多数の関節を持ち、そのため制御は複雑で困難である。また動作プログラミング時には実機動作を予測しながらプログラミングする必要があるためエンドユーザにとって容易にロボット動作プログラミングが行えるシステムが望まれる。

2. 目的

本研究ではロボットの動作生成およびシミュレーションを行う機能を持つシステムを設計及び実装する。このシステムは ZMP (Zero Moment Point) 等の動作力学に基づく動作生成システム [1] やシミュレータ要素の強いユーザインターフェース [2] とは異なり、研究者の視点でロボットを使用するのではなく、エンドユーザの視点でロボットを使用することに重点を置いている。ゆえにエンドユーザにとって直感的かつ容易に二足歩行ロボットの動作プログラミングが可能なシステムを設計する。

3. 設計方針

本研究では、二足歩行ロボットに富士通オートメーション製小型二足歩行ロボット HOAP-1 を使用し、その制御に既存の HOAP-1 制御用ソフトウェアを用いる。またロボット制御に必要な動作データは従来の手動生成を全自動化し、さら

にロボットの動作を視覚的に捕らえながら動作データを作成可能なシステムを設計する。システムは動作データ自動生成システムと 3D ロボットモデル操作システムから成り、動作データ自動生成システムは Java 言語を用いて設計し、3D ロボットモデルは Java3D を用いて表現する。この 3D ロボットモデルはマウスで操作可能であり、その動作を動作データ自動生成システムに反映することで実機ロボットの目標動作状態な動作データを作成する。

4. 実装

4.1 動作データ自動生成システム

動作データ自動生成システムはロボットに必要な動作データファイルを自動生成することでエンドユーザにとって困難となり得る原因のひとつである、多数関節の制御を簡易化する。本研究で使用するロボットは全 20 個のモータをロボット動作の時間経過に沿って制御する必要があるため、動作データファイルには 20 個のモータについて個々にモータ目標値を記述する。また、ロボットが有する姿勢制御センサ・左右足裏の感圧センサの ON・OFF に関する記述をする。動作データに記述する値はモータ固有のパラメータ値に目標回転角度を掛けた値を記述し、10 進数で表す。

モータの回転はモータ負担を軽減しロボットの動作をスムーズに行うため、モータ目標値までの滑らかな回転が求められる。ゆえに動作データファイルには微量なモータ回転を行うべく、モータ回転時間に応じて少しずつ回転するよう記述する。動作データファイル自動生成システムではこれらの計算・作業を自動化することでユーザ負担を軽減する。

4.2 3D ロボットモデル

本研究で開発したユーザインターフェースの特徴として、3D ロボットモデルを操作することでロボットの目的動作を決定する機能が挙げられる。3D ロボットモデルは本研究で使用するロ

Design and Implementation of Humanoid Robot Motion Generation System using Java3D

[†] Junichi Sakai, Hiroyuki Nisiyama, Fumio Mizoguchi
Faculty of Sci. and Tec, Tokyo University of Science

[‡] Makoto Obayashi
Information Media Center, Tokyo University of Science

ボット HOAP-1 の設計図に基づき、ロボットのリンク間衝突の可能性のある稼動部分に関して、またリンク構造については限りなく精密に再現した。

3D ロボットモデルは Java3D を用いて表現した。Java3D を用いることで OpenGL や DirectX といったグラフィックインターフェースとは異なり、OS に対する依存性を無くした。

4.3 マウスピッキング

本研究のユーザインターフェースでは 3D ロボットモデルを目的動作状態に決定する手法として、マウスピッキングと数値入力の両手法を採用した。数値入力[3]やジョイスティック[2]を用いた方法は他にも存在するが、両手法を採用することで詳細な動作状態も作成することを可能にする。

4.4 ユーザインターフェース

図 1 に本研究で作成したユーザインターフェースを示す。図 1 の左側が 3D ロボットモデル部で右側が数値入力部及び動作データ作成部である。3D ロボットモデル部では視点の移動やロボットの位置移動が可能であり、数値入力部では動作データ作成時に関節稼動範囲を越えた場合のエラー表示を行う。またメニューバーから作成した動作データのグラフ表示や連続・不連続動作を作成するモードが選択可能である。また、図 2 はユーザインターフェースを用いて動作データを作成する一連の流れを示したものである。

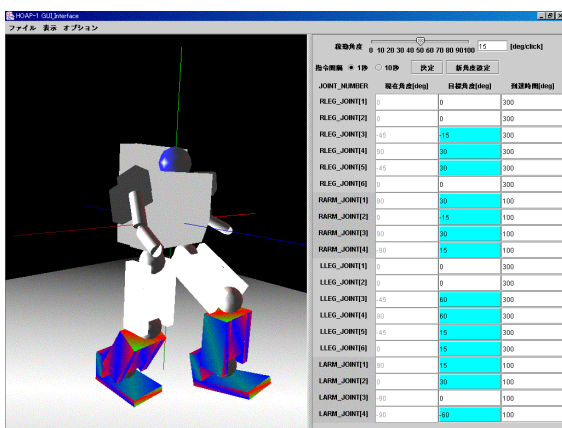


図 1. ユーザインターフェース

3D ロボットモデルの操作、または関節数値入力によって目標動作状態を決定し動作データ自動生成システムによって決定した関節角度のエラー検出を行う。エラーがあった場合には数値表

示部分、またはウィンドウにてエラー関節を示す。エラーが検出されなかったならば動作データが自動生成される。

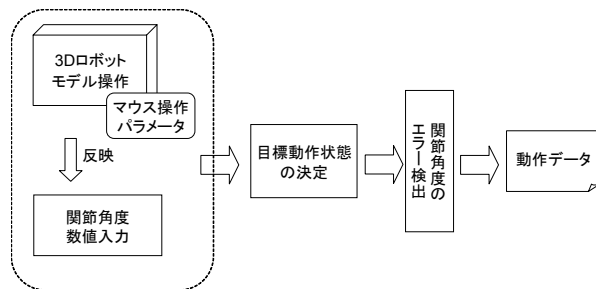


図 2. 動作データ生成までの一連の流れ

5. おわりに

本研究では、エンドユーザが容易に二足歩行ロボットの動作プログラミングが可能なシステムを開発した。一般的にロボット動作プログラミングには機械工学的・制御工学的等の専門知識が必要であるため、ロボット制御を行うのはロボット研究者等に限定されている。しかしながら本研究では容易にロボットの動作生成を可能にし、専門知識を持たないエンドユーザでも動作プログラミング及びロボット制御を可能とするシステムを開発した。

参考文献

[1]Koichi Nishiwaki, Satoshi Kagami, Yasuo Kuniyoshi, Masayuki Inaba, Hirochika Inoue, “Online Generation of Humanoid Walking Motion based on a Fast Generation Method of Motion Pattern that Follows Desired ZMP”, IEEE /RSJ, Intl. Conference on Intelligent Robots and Systems EPFL, Lausanne, Switzerland • October 2002, 2002

[2]Kei oakada, Yasuyuki Kino, Fumio Kanehiro, Yasuo Kuniyoshi, Masayuki Inaba, Hirochika Inoue, “Rapid Development System for Humanoid Vision-based Behaviors with Real-Virtual Common Interface”, IEEE/RSJ, Intl. Conference on Intelligent Robots and Systems EPFL, Lausanne, Switzerland • October 2002, 2002

[3] 原功, 木村陽一, 麻生英樹, 河村進: 「インタラクティブ・ロボット基本ソフトウェアの開発」, 産業技術総合研究所情報処理部門, 株式会社カーネル