

HRP-2における統合型システムソフトウェア環境による 統合行動計画システム

小倉 崇 岡田 慧 稲葉 雅幸 井上 博允 (東京大学)

The United Behavior Planning System for Integrated System Software on HRP-2

*Takashi OGURA, Kei OKADA, Masayuki INABA, Hirochika INOUE
(The University of Tokyo)

Abstract— This paper shows a way to extend the HRP-2 Control Software for advanced behaviors. It is able to develop a robot system integrately and rapidly by using an existing software environment with geometry models. The extended system enables high-level behavior planning. Now HRP-2 can walk to the target position and operate it by only the information of the operation target and the obstacles. Path Planning, Footprint Planning and Joint Angles Planning can be done in the extended system, then HRP-2 Software executes the results of the plans. We show an experiment in an elevator environment.

Key Words: Humanoid, HRP-2, Robot System

1. はじめに

本研究では HRP-2 制御ソフトウェアをそのまま利用しながら、既存のシステムを利用して高度な行動を実現するためのソフトウェア拡張システムの構成法を述べる。本論文で目指す高度な行動とは、ロボットから見たある物体の位置、障害物の情報を与えたときに、障害物を考慮してその目的の物体を操作できる場所へ歩いて行き、その物体に押す、つかむ等の簡単なオペレーションを行うことができるというレベルである。そのためには、物体操作が可能な位置の割り出し、その位置までのパス、具体的な歩容計画、物体オペレーションのための姿勢決定などを行わなくてはならない。行動計画を统一的に既存のシステム上で行ってしまい、その結果を HRP-2 で実行、実現する。最後に行動統合実験としてエレベータ乗り込み実験を行う。

2. HRP-2 制御ソフトウェアの拡張手法

Fig.1 (a) は通常の HRP-2 制御ソフトウェアの構造を簡単にあらわしたものである。ユーザはそれぞれの行動実験を行う際に、歩行のためのパラメータファイルや、ユーザプラグインと呼ばれる、HRP-2 専用のデータやプログラムを用意する必要がある。

HRP-2 の制御ソフトウェアを書くためには、I/O レベルで記述する方法、ユーザプラグインを記述する方法、既存のプラグインを利用したスクリプトを書く方法を挙げることができる。I/O レベルで記述する方法では低レベル処理から自由な記述ができるメリットがあるが、シミュレータへの互換性がなく、また、高度な行動記述を行うには適していない。ユーザプラグインを作成すると、Adaptor という機構を利用できるため、シミュレータへもそのまま適応が可能である。ただし、プラグインは HRP-2 独自のプログラムであり、これをはじめから記述してゆくにはソフトウェアの開発時間がかかってしまう。HRP-2 には既存のプラグインがいくつかあり、これらを利用することで、簡単に行動の記述が可能であるが、これら単体では高度な行動を記述することは困難である。また、既存のシステムと連携させることを考える。既存のシステムを利用することで不要なソフトウェアの開発をする必要

がなくなる。そして、ここで新たに作るソフトウェアも、別なロボットに適用することが可能であるように作ることで、今後新しく作られるハードウェアにもソフトウェア資産を継承することができるような構成となる。

また、既存のシステムを HRP-2 のシステムと統合、連携させるには、HRP-2 制御ソフトウェアとのインタフェースを記述する必要があり、以上に述べたどのレベルと接続させるかを考える必要がある。

センサ値を利用するにはこのインタフェース専用のプラグインを記述する必要があるが、高度な行動計画を実現するという目的を実現するためには特に必要とされない。そこで、本研究ではもっとも簡単に統合が可能で、HRP-2 の高度なシステムを利用可能な HRP-2 のプラグインを利用した方法をとる。行動計画は既存のシステムで行い、HRP-2 の全身関節角度列までを作成し、それをシーケンス機構へ渡してやればよい。

Fig.1 (b) は本研究で作るシステムの全体像である。EusLisp²⁾ 上に作成されたロボットの統合システムソフトウェア上で、パスプランニング、歩容計画、物体オペレーションのための姿勢決定を行い、これを HRP-2 に適用する。

HRP-2 には全身の関節角度を補完して作成する seqplay プラグインや、移動量や、足跡座標を与えると歩行を行うプラグインがある。そのため、これらを用いれば、対象物までの足跡計画と、全身の関節角度計画を得ることで、移動、物体操作がある程度可能である。そこで、本研究では障害物情報と、操作対象物情報を入力として、足跡計画と、全身の関節角度計画を得るプランナを Fig.2 のように設計した。移動目標点プランナ (Move Target Point Planner) が対象を操作するための座標を決定し、それにもとづいて経路プランナ (Global Path Planner) が目標座標までのグローバルな経路を計画する。そして足跡プランナ (Footprint Planner) がグローバルなパスにもとづいて足跡を決定してゆく。姿勢プランナ (Joint Angles Path Planner) はこれらの入力から、動作の元となる姿勢を作る。この二つの出力である足跡計画 (Footprint Plan) と関節角度計画 (Joint Angles Plan) を HRP-2 に渡し、動作

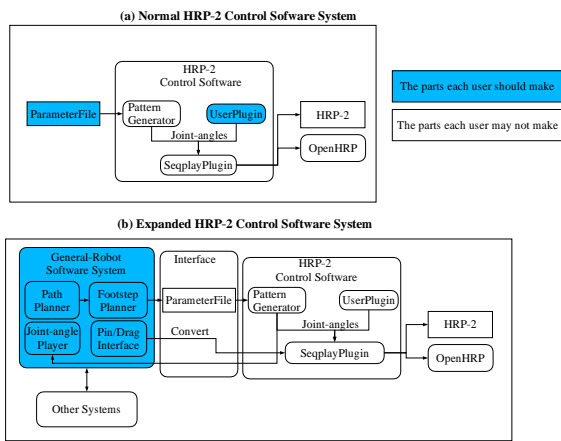


Fig.1 HRP-2 のソフトウェアシステムとその拡張

を実行する。

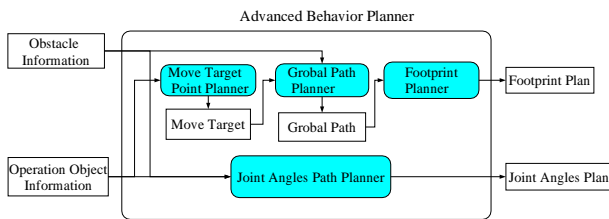


Fig.2 本研究で実現する高次行動計画システム

3. 足跡計画にもとづく HRP-2 の対象物操作可能地点への移動

3-1 操作対象物情報からの操作可能地点位置の決定

移動目標点プランナは操作対象の座標系が与えられたとき、その対象物を扱うための位置を決定する。対象とするロボット（ここでは HRP-2）の腕の長さを元に、腕の長さに係数（= 0.8 程度）を掛ける。対象の中心からの距離をこの長さにし、対象を正面に捉えることができる位置、向きを対象物操作可能地点とする。

3-2 操作位置へのグローバルパスの決定

経路プランナは、障害物情報をもとに障害物を考慮した移動目標点までの経路を決定する。まず、RRT-connect(Rapid Random Tree)³⁾ を用いてロボットの体をバウンディングボックスで近似し、目標地点までの体の向きを含んだパスを決定する。ここでいうパスは、X, Y, Z 座標と、Z 軸周りの回転角度で表現される。バウンディングボックスで近似し、回転を考慮した場合、狭いドアなど、体を回転させると通ることのできる場合、回転させたパスを得ることができ、人間のよう自然で複雑な歩行計画ができる。また、シリンダーで体を近似し、回転を考慮しないパスをもとめた場合、回転などは出てこないが、冗長な動きがなく、安全で確実なパスを得ることができる。

3-3 グローバルな経路にもとづく足跡計画

足跡プランナは目的地までのグローバルな経路をもとに足跡を決めていく。足跡は左右の足をこのパスに

そって配置してゆくことで決定する。あらかじめ最大の歩幅、最小の歩幅、最大の回転角度をあたえ、一步で無理がある場合は歩数を増やし、補完した位置に着地するように決定する。また、パスから左右の足の歩幅を足したものを y 座標に加える。ただし、左右の足の干渉を考慮する必要があるが、一步前の足を障害物に加え、RRT-connecter を用いることで探索することができる。

3-4 移動計画の HRP-2 への適応

HRP-2 制御ソフトウェアには online, offline 歩行パターンジェネレータが付属している。このソフトウェアは歩容に関する様々なパラメータを与えることで、HRP-2 の歩行の全身の関節角度列軌道を生成することができる。ここでは足裏接地地点の座標を与えることで歩行を実現する。パラメータ USE_SZ を指定することで sx-sy-dsy-dz-th という 5 つのパラメータを設定可能になり、このとき足裏の Z 座標を入力することで階段の昇降なども可能になる強力なものである。

そこで、足裏接地地点座標を別なシステムで生成し、これを HRP-2 歩行パターンジェネレータが読み込むことが可能な形式のパラメータファイルに変換することで段差乗り越えを考慮したパスを HRP-2 において行うことが可能である。

足跡計画からパラメータファイルを作成する。足跡計画は座標系で表現され、これをもとに sx-sy-dsy-dz-th という 5 つのパラメータを決定する。sx, sy, dz は次の一步への座標移動量、th は方向の変化量、dsy は平行移動量をあらわす。このパラメータファイルを歩行パターンジェネレータに与えることで、計画された歩容を HRP-2 上で実現することができる。

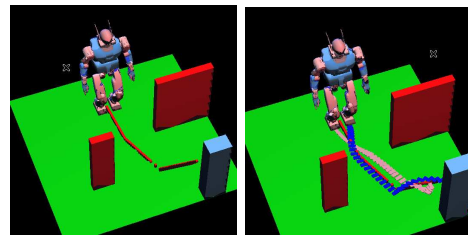


Fig.3 RRT により得られたパスと Footprint

4. 動作目的のための関節角度計画

4-1 姿勢決定法

HRP-2 制御ソフトウェアには腕の逆運動学を解くソフトウェアが付属しているものの、全身の自由度を生かした運動を逆運動学で求めることはできず、自分で用意する必要がある。また、自己干渉、環境との干渉を考慮することも姿勢決定には重要である。本研究では、拘束条件下での全身の逆運動学計算法で、任意のリンクを環境に固定する条件を追加し、任意のリンクにピンを設定できる逆運動学計算法⁴⁾ ピン/ドラッグで全身の関節角度を決定する。また、環境との干渉を考慮した姿勢変化動作を行う必要があり、これには RRT-connect が有効である⁵⁾。

HRP-2 の VRML ファイルをもとに幾何モデルを用意し、このモデルに対しピン/ドラッグを適用し、目的の姿勢を作成する。固定する部位を指定してやり、動かす部位に目標座標系を与えることで重心位置を考慮した全身の自由度を生かした姿勢を作る。ある物体の

座標を与え、これに手を添えるように指定し、両脚先の座標を固定して、ピン/ドラッグで逆運動学を解いた結果の例を Fig.4 に示す。

4.2 HRP-2 制御ソフトウェアへの姿勢列データの受け渡し

次にここで作られた関節角度列を HRP-2 で実行しなければならない。HRP-2 制御ソフトウェアでは seqplay プラグインに全身の関節角度列を与えることでロボットを制御する。その際ファイル形式を読み込むことが可能である。そこで以上のようにして得られたモデルの関節角度列を HRP-2 用のファイル形式で保存する。この形式では最初に記述した関節角度へ以降するまでの時間が設定できない、という問題点がある。そのため、スクリプトファイルには十分な時間を記述し、こちらで入力する最初の姿勢は初期姿勢などをいれるようにする。

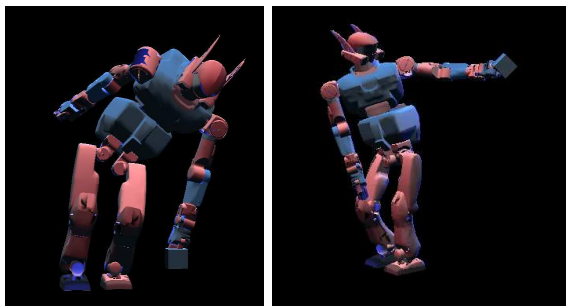


Fig.4 EusLisp モデルにピン/ドラッグによる全身逆運動学を適応した結果

5. エレベータ乗り込み動作実験

エレベータのボタンを押すことを指示されたことをロボットが理解したら、そこまで歩行し、ボタンを押す、というのが本実験のロボットのタスクである。この実験を通して、これまでに述べてきたシステムにおいて、対象物の座標、障害物情報のみによって、歩行、対象操作の計画を行い、実行することができることを示し、このシステムの高度な行動計画の有効性を示す。これまでに述べた高次の行動プランナにより、目標地点までの移動、ボタン押しを実現する。環境は EusLisp で構築したものを VRML 形式に変換して構築した。対象物や障害物の情報は視覚などを通して得るべきものであるが、今回の実験は行動計画が目的であるため既知とした。

まず、EusLisp 上ですべての行動のプランニングを行う。ボタン位置から、移動位置を決定し、パスの探索、足跡を決定した。このデータは歩行のパラメータファイルとして保存され、パターンジェネレータに渡される。そして今回の対象物に対する操作はボタンを押す操作である。この動作の達成のため、ボタン位置の手前の座標を経由し、ボタン位置まで手を伸ばす軌道をこれまでに述べた手法により作った。この全身関節角度列は seqplay プラグイン用のファイルとして保存される。さらに、パターンジェネレータによって作られた歩行軌道を再生することで、エレベータの中へと移動した。次にこれらのファイルを HRP-2 制御ソフトウェアを用いて再生、実行した。OpenHRP で実行した結果を Fig.5 に示す。この実験により、これまで述べ

たシステムによって、簡単な対象物の指示により複雑な行動が可能なシステムの作成ができたことを示した。

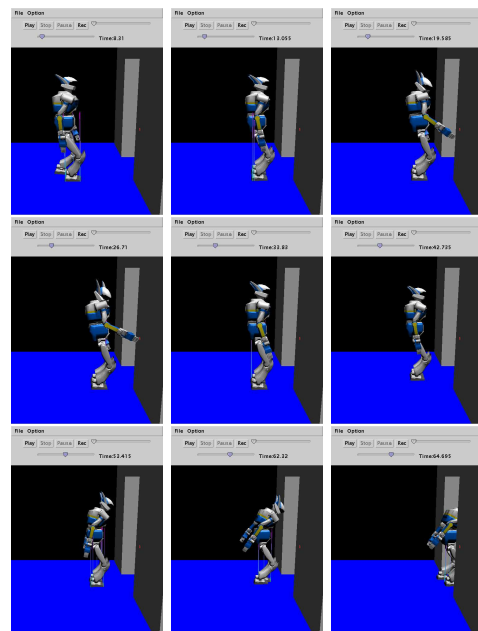


Fig.5 OpenHRP でのエレベータ乗り込み実験シミュレーション

6. おわりに

HRP-2 のシステムを拡張、統合し、全身の逆運動学、障害物を考慮した歩行パスプランニングを HRP-2 において実現することができた。また、これらを統合し、対象物の座標を与えることで対象物に対する操作が可能な地点までの経路と足跡を計画し、移動することができた。

課題としてセンサ情報を統合環境でフィードバックすることができていないことや、リアルタイムな操作の実現がある。ソケット通信を利用した専用プラグインを作成し、オンラインでの動作指令をすることでさらに動的で柔軟な動作の実現が可能である。

参考文献

- 1) 金広文男, 宮田なつき, 梶田秀司, 藤原清司, 比留川博久, 中村仁彦, 山根克, 山海嘉之, 川村祐一郎, 小原一太郎, 川村進: ヒューマノイドの実時間制御ソフトウェア開発環境: OpenHRP, 第 19 回日本ロボット学会講演会 (2002).
- 2) MATSUI TOSHIHIRO, INABA MASAYUKI: EusLisp: An Object-Based Implementation of Lisp, Journal of Information Processing Contents Vol.13 No.03.
- 3) J.J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba and H. Inoue: Footstep planning among obstacles for biped robots, In Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'01).
- 4) 山根 克, 中村仁彦: ヒューマンフィギュアの全身運動生成のための協応構造化インタフェース, 日本ロボット学会誌, 20, 3, pp.113-121 (2002).
- 5) 中井博之, 稲葉雅幸, 井上博允: RRT-Connect Planner を用いたロボットの全身行動生成, ロボティクス・メカトロニクス講演会'03 予稿集.