

実/仮想ロボットの統合的制御ソフトウェアシステム Z-REVICHs の開発

Development of Z-REVICHs, Unified System for Real/Virtual Robot Controlling

○学 杉原知道 (東大) 正 中村仁彦 (東大)

*Tomomichi SUGIHARA, Yoshihiko NAKAMURA.

Univ. of Tokyo. 7-3-1, Hongo, Bunkyo-ku, Tokyo

This paper introduces a software system which can be applied for the control of both real and virtual humanoid robots compatibly. This system provides researchers with much more handy and lower risky environment for experiments. The structure of the system and the basic strategy of how to design such a system is described.

Key Words: Humanoid Robot, Software System, Control, Simulation

1. はじめに

実ロボットを用いた実験における危険性・コストを低減するため、シミュレーションを併用することが有効である。その際ロボットの挙動だけでなく、その挙動を誘起する制御ソフトウェアシステムも同時に検証出来ることが望ましい。即ち、実ロボットと仮想ロボットを統合的に制御することを可能としたシステムが要求される¹⁾。

従来、シミュレーションや制御系の開発等には市販のシミュレータや計算エンジンが多く利用されてきたが、それらは i) 実ロボットに実装される制御システムとのデータ可搬性に劣る ii) それ自体が大規模なシステムであり大量のリソースを消費する iii) 一般的に非常に高価であるなどの理由から、このようなシステムに組み込むのは困難である。また、同様の問題意識から開発されたシステムも幾つか発表されているが^{2) 3) 4)}、実ロボットとの可換性や検証環境としての信頼性、使用における簡便さといった点で問題がある。

このような背景から、実/仮想ロボットの統合的制御ソフトウェアシステムである Z-REVICHs (Z's REal/Virtual Compatible Humanoid System, “Z” は開発者名の頭文字) を開発した。本稿では、その構成及び特徴について述べる。

2. 実/仮想ロボットの統合的制御

ロボットのソフトウェアシステムは、ハードウェアシステムと密接な繋がりを持つプロセスから、様々な知的処理を行う高次のプロセスまで包括する大規模なものとなるため、構成が複雑になり、拡張性・保守性の低下を招きやすい。このような問題の解決に当たって重要なのは、できるだけ複雑化を避けるような適切なモジュールの切り分け、及びそれらのモジュール間の整合性を確保することであるが、必要な部分的機能を各々全くデータ処理方式の異なるモジュールを繋げることによって補い合ういわゆる「継ぎ剥ぎ」システムでは、各モジュールの単純化や整合性の確保は難しい。データの処理方式に関する規約をまず定め、それを元にトップダウンに全体を設計することが高い拡張性・保守性、高い開発効率に繋がると考える。

更に、実ロボットと仮想ロボットを統合的に制御対象として扱えるシステム構築のための要件として次の3点が挙げられる。

I) 最小限のモジュール置換による実/仮想ロボット切替

実ロボットの操縦・制御システムは、個々の機能に応じて実装された複数のモジュールやプロセスが適切に連動することで実現される。最小限の置き換えによって実/仮想ロボットの切替を行えるようにするためには、これらのプロセスのうちロボットハードウェアと密に関わるモジュールの独立性を強め、他の高次処理を行うモジュールと完全に切り分けることが必要である。

II) プロセススケジューリング機構

通常の OS ではプロセスのスケジューリングはカーネルに一任され、外部から明示的に制御することは困難である。個々に実装されたプロセス群を適切に連動させるために、これを外部から陽に管理出来る特殊な OS を用いるか、或いはシステム自身が何らかのスケジューリング機能を有することが望ましい。

III) システムと親和性の高い順動力学シミュレータ

実ロボットの制御においては、センサ情報を逐一取り込み、時々刻々変化していくロボットの内部状態を即時的に観測することが必要である。動力学シミュレーションにおいても同様に、ロボットのモデル及びデータが容易に外部プロセスから観測でき、さらに制御にそのまま用いることが可能であるような、高いデータの可搬性が求められる。またデータのみならず、II) に掲げたようなスケジューリング機構の下で他のプロセスと同様に機能するような、システムとの高い親和性を有していなければならない。

次節では、Z-REVICHs においてこれらの要件が具体的にどのように実装されているかについて述べる。

3. Z-REVICHs のシステム構成

Z-REVICHs は、Linux 及び RT Linux の上で利用可能な実/仮想ロボットの統合的制御ソフトウェアシステムである。共通のライブラリ Z-DYNAFORM⁵⁾ を基にトップダウンに全体を設計・構築し、モジュール間の整合性を確保した。このため、拡張・保守が比較的容易である。システム全体の構成を Fig.1, Fig.2 に示す。

システムは、次の5つのプロセス及び独自に開発した順動力学シミュレータ (次節で説明) から成る。

- operation interface

ロボット操縦用のインタフェース。システム全体の起動・停止、動作モードの決定等を司る。

- sequencer

オペレータからの指令をシーケンシャルに送信するプロセス。

- motion manager

sequencer から送られてきた指令を、ロボットのバランス等を考慮して適切に変換し、個々のアクチュエータへの指令値を算出するプロセス。

- base/vbase

アクチュエータへの指令値をデバイスドライバ等を与える実ロボット用のプロセス (base), 及び順動力学シミュレータに与える仮想ロボット用プロセス (vbase)。

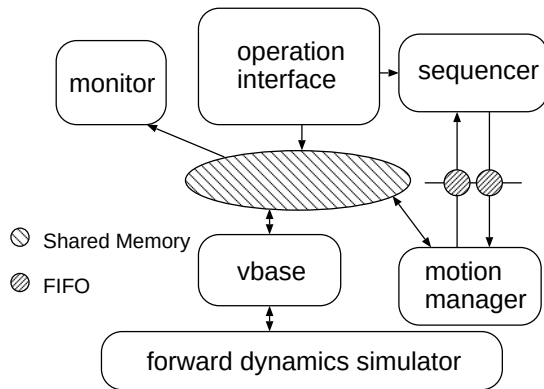


Fig.1 Z-REVICHS for virtual robot

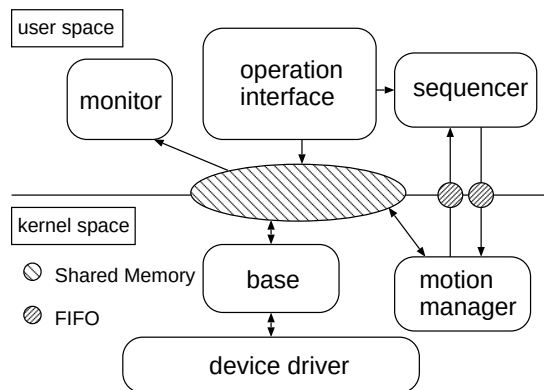


Fig.2 Z-REVICHS for real robot on RT Linux

● monitor

ロボットの内部状態をオペレータが観測するためのプロセス。

プロセス間通信は主に共有メモリ（一部 FIFO）によって行う。共有メモリには、ロボットの内部状態モデル、各アクチュエータに与えられる指令値セット等ロボットの制御及び順動力学シミュレーションに必要なデータ全て、及びオペレーション全体を管理するためのデータ群を格納し、データの可搬性と整合性を確保した。

通常の Linux 上で動作する際には、独自のスケジューリング機構によりプロセス間の同期を図る。これは、共有メモリ上に確保された円環リストに各プロセスの ID 及び実行周期を登録し、周期タスクの実行順序を管理するものである。また RT Linux の上で動作するときは、OS 自体が有する強力なハードリアルタイムスケジューラをそのまま利用することが可能である。

4. 順動力学シミュレータ

データの可搬性、システムとの親和性などを考慮すれば、順動力学シミュレータも、市販のものを採用するのではなく、他のプロセス・モジュール群と同一の枠組を持つことが望ましい。文献³⁾では、演算の高速性を重視しゲーム用の動力学演算エンジンを用いているが、本来ロボットの制御は系のダイナミクスと密接に関連するものであり、シミュレーションにおいても演算結果に十分な妥当性が求められる。

本研究では、単位ベクトル法による順動力学演算法⁶⁾と、環境との接触点に固いばね・ダンパモデルを挿入することによる外力計算法を併用し、順動力学シミュレータを開発した。通常ばね・ダンパモデルによる外力計算は数値的に不安定になりやすく、非常に細かい時間刻み幅を必要とすることが多いが、ここでは文献⁷⁾の方法を応用し、1[msec]程度の時間刻みでも比較的安定な結果を得ている。

開発したシミュレータ上にて 20 自由度のヒューマノイドを転倒させた。この動作は約 5 秒のもので、シミュレーションには時間刻み幅を 1[msec] として約 12 秒を要した (CPU PentiumIII 650MHz)。シミュレーションの結果のスナップショットを Fig.3 に示す。

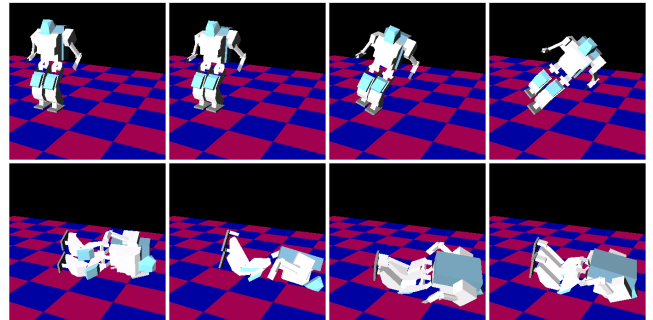


Fig.3 A Snapshot of a falling-down simulation

外力計算法として他に文献⁸⁾の方法も実装し、種々の動作について検証・比較したところ、両者共ほぼ同じ挙動を示した。前者は比較の実装が容易であり、演算量も少ない。更に、任意の剛性を持った物体を環境モデルに組み込むことができる。

5. おわりに

実ロボットと仮想ロボットを統合的に制御することを可能としたソフトウェアシステム Z-REVICHS を開発した。トップダウンな設計に基づいて全体の整合性を確保し、共有メモリを中核としたプロセス間通信による高いデータ可搬性と外部から陽に管理出来るプロセススケジューリング機構、システム全体との親和性の高い順動力学シミュレータ等を有する点が特徴である。

なお、本研究は科学事業振興事業団戦略的基礎研究推進事業 (CREST) 領域「脳を創る」の補助を受けた。

参考文献

- 1) 金広文男, 稲葉雅幸, 井上博允. 実ボディとのブレイン共有が可能なシミュレーション環境の構築. ロボティクス・メカトロニクス講演会'99 講演論文予稿, pp. 2P1-70-101, 1999.
- 2) 平野毅, 藤本康孝, 河村篤男. ロボット制御シミュレータ (ROCOS) の開発と人間型 2 足ロボットの跳躍動作. 電気学会全国大会講演論文集, pp. 169-170, 1998.
- 3) 金広文男. 高速シミュレーション環境 FAST を用いたヒューマノイドシミュレーション. 第 19 回日本ロボット学会学術講演会予稿集, pp. 931-932, 2001.
- 4) 中村仁彦, 比留川博久, 山根克, 梶田秀司, 横井一仁, 藤江正克, 高西淳夫, 藤原清司, 永嶋 史朗, 村瀬有一, 稲葉雅幸, 井上博允. 仮想ロボットプラットフォーム. 日本ロボット学会誌, Vol. 19, No. 1, pp. 28-36, 2001.
- 5) 杉原知道, 西脇光一, 稲葉雅幸, 井上博允. 汎用多リンク系動力学演算ライブラリ「Z-DYNAFORM」の開発. 第 18 回日本ロボット学会学術講演会予稿集, pp. 1139-1140, 2000.
- 6) D.E.Orin M.W.Walker. Efficient dynamic computer simulation of robotic mechanisms. *Transactions of the ASME, Journal of Dynamic Systems, Measurement, and Control*, Vol. 104, pp. 205-211, 1982.
- 7) 山根克, 中村仁彦. $O(N)$ 順動力学計算法と陰積分による衝突・接触の高速シミュレーション. ロボティクス・メカトロニクス講演会'02 講演論文集, 2002.
- 8) 藤本康孝, 河村篤男. 床との衝突及び摩擦を考慮した 2 足歩行ロボットの三次元運動シミュレーション. 日本ロボット学会誌, Vol. 15, No. 6, pp. 857-863, 1997.