OpenHRP3 における動力学計算サーバ

東京大学 中村 仁彦，山根 克

１．順動力学計算法

（１）はじめに

OpenHRP3 におけるリンク系の物理シミュレーションのための基本的な計算機能を提供する動力学計算サーバには，筆者らが開発した分解・組立法 (Assembly-Disassembly Algorithm, ADA) [1]と呼ばれる順動力学計算法が実装されている．本稿ではその概要について述べる．詳細については学会発表[2][3][4]を参照されたい．

リンク系の順動力学計算法は古くから研究されてきた問題であり，計算量がリンク数に比例する計算法が知られている[5][6][7]．また，並列計算を行うことにより理想的な計算量が O(logN)となる方法が開発されている[8][9]．筆者らの方法もその一つであり，ヒューマノイドなどリンク数の多い複雑な機構に対しても高速な計算が可能である．ただし，OpenHRP3 の動力学計算サーバは並列計算には対応していない．

（２）分解・組立法 (ADA) の概要

ADA は図 1 に示す 2 つのステップからなる．

(a) assembly ステップ
全てのリンクが独立 (関節による拘束がない) 状態からスタートし，関節を 1 つずつ追加していく．追加するときに，その関節で働く拘束力と発生する加速度を計算する．このとき計算される拘束力と加速度は，まだ追加されていない関節の影響を考慮していないので, 一時的な値である. 以上の計算をすべての関節に対して行う．

(b) disassembly ステップ
assembly ステップで最後に追加された関節の拘束力と加速度は，それ以上考慮すべき関節はないので，正しい値である．そこで，最後に追加された関節からスタートし，assembly ステップと逆の順序で全関節の最終的な拘束力と加速度を計算する．

(a)(b)において，各関節の処理に必要な計算量は一定以下である．したがって，全体の計算量は関節数に比例することになる．また，assembly ステップにおいて関節を追加する順序は任意である．ただし，順序を変更すると総計算量が変化する．disassembly ステップにおける順序は，必ず assembly ステップにおける順序の逆にしなければならない．

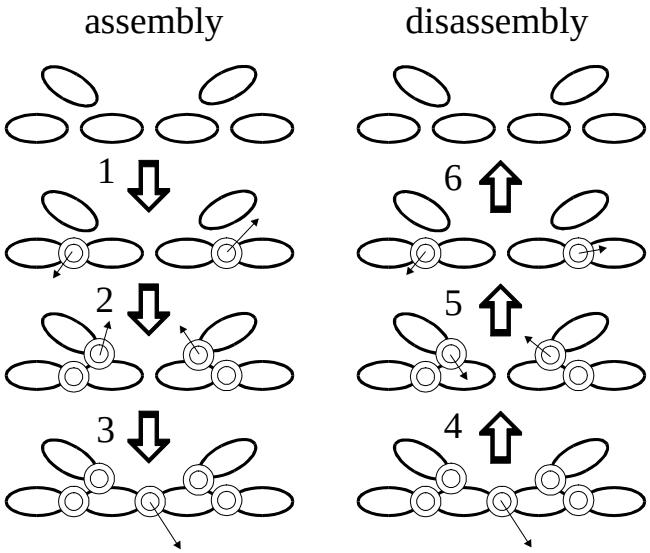

図 1 順動力学計算の 2 つのステップ

この計算法には以下の 3 つの特徴がある.

- 関節を追加していく順序は任意である．また順序を変更することによって数値安定性や並列性を変化させることができる.
- 関節加速度だけでなく，全関節での拘束力が得られる．これは力センサのシミュレーション，接触のモデリングなどに利用可能である.
- 閉リンク機構を含む任意のリンク機構に適用可能である.

閉リンク機構・開リンク機構が切り替わるリンク系 (構造可変リンク系) のシミュレーション例を図 2 に示す.

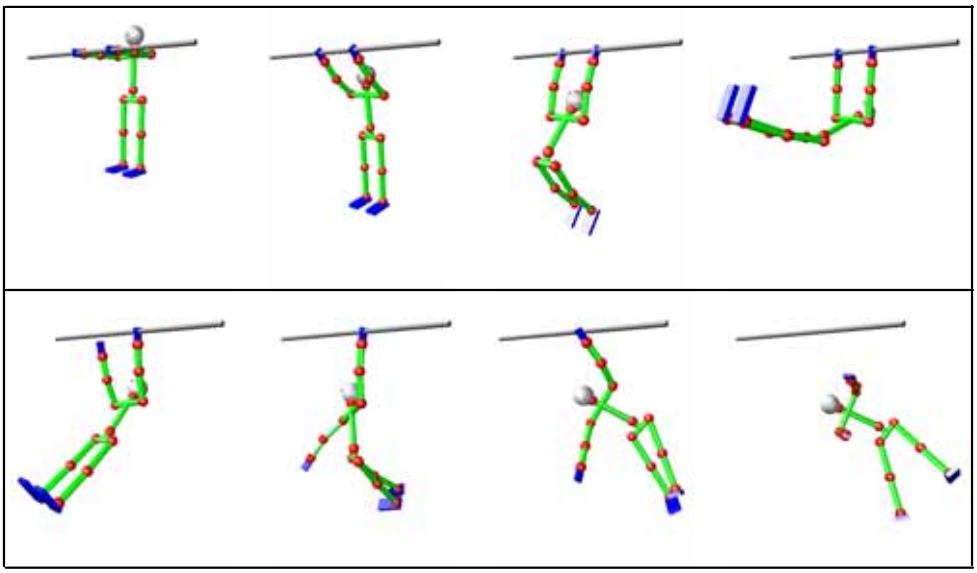

図 2　構造可変リンク系のシミュレーション例

（3）他の計算法との比較

筆者らの ADA と他の計算法の速度を比較する．比較対象は，最初に提案された O(N)計算法である Articulated-Body Algorithm (ABA)[5]，ADA と同様に並列計算が可能な Divide-and-Conquer Algorithm (DCA)[9]である．ABA は並列計算ができず，閉リンク機構にも適用できない．ADA・DCA はいずれも並列計算が可能で，閉リンク機構にも適用可能である．

さまざまな関節数のシリアル機構に対する計算時間を図 3 に示す．このように，ADA の計算速度は ABA には劣るものの，閉リンク機構が扱えるアルゴリズムの中では高速であることがわかる．また，2 プロセッサによる並列計算により ADA の計算時間は 2/3 以下になることがわかっており[10]，並列計算をサポートするようになれば ABA よりも高速になることが期待できる．

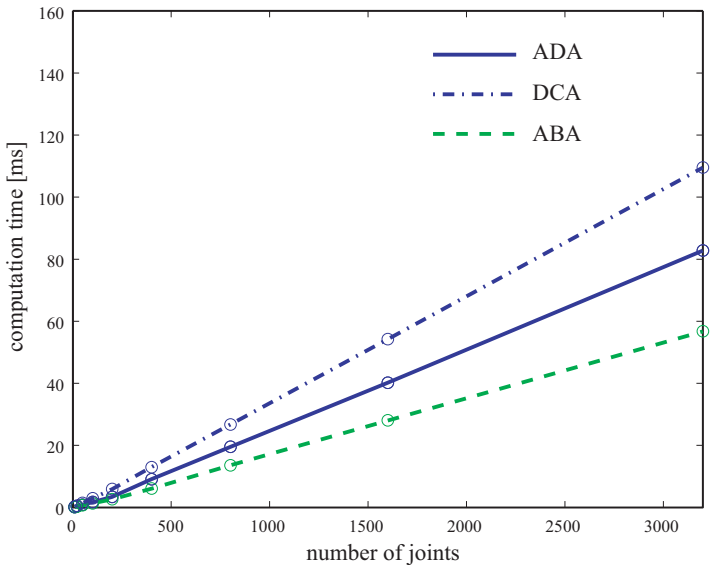

図 3　シリアル機構の順動力学計算に要する時間 (Xeon 3.8GHz)

2．接触力計算

OpenHRP3 の動力学計算サーバでは，剛体接触モデルによる接触力計算を行っている．具体的には，Lloyd の方法[11]と同様に接触問題を線形相補性問題 (Linear Complementarity Problem, LCP) として定式化し，Lemke 法に基づく解法[12]を適用して接触力を求めている．ただし，剛体リンク系の接触問題では単純に Lemke 法を適用しても解が得られないことが多いため，独自の工夫を加えて安定に解を得る方法を開発している．

ヒューマノイドの運動に対するシミュレーション結果と実機実験結果の比較を図 4 に，またそのときの鉛直方向の反力の比較を図 5 に示す．定性的に類似したシミュレーション

結果が得られていることがわかる．鉛直方向の反力は，シミュレーションの方が一般に衝突の数・大きさとも大きい結果となった．これは，シミュレーションでは構造や関節の弾性が考慮されていないためと考えられる．

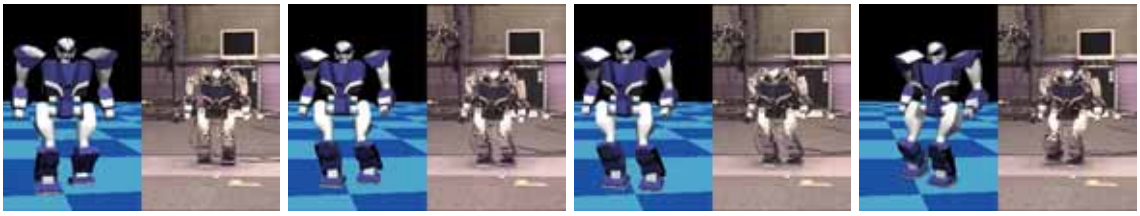

図 4　比較実験における運動の様子

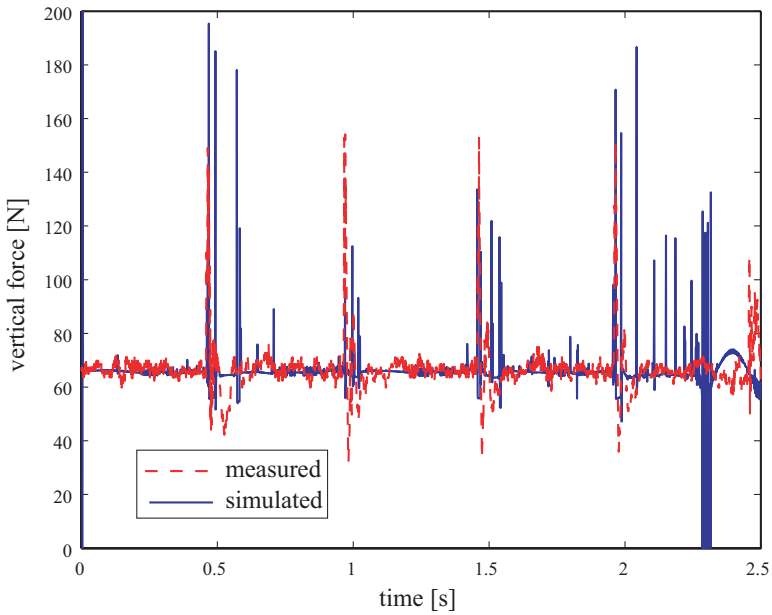

図 5　シミュレータによる垂直反力の計算結果 (青実線) と実測値 (赤破線)

[1] K. Yamane. and Y. Nakamura: “Efficient Parallel Dynamics Computation of Human Figures,” Proceedings of the IEEE International Conference on Robotics and Automation, pp. 530-537, 2002.

[2] K. Yamane and Y. Nakamura: “Parallel O(logN) Algorithm for Dynamics Simulation of Humanoid Robots,” Proceedings of IEEE-RAS International Conference on Humanoid Robotics, pp. 554-559, 2006.

[3] 山根，中村: “分散コンポーネント型ロボットシミュレータにおける順動力学エンジンの計算効率,” 第 24 回日本ロボット学会学術講演会, 2N23, 2006.

[4] 山根，中村: “ピボット法に基づく LCP の解法とその衝突・接触シミュレーションへの応用,” 第 25 回日本ロボット学会学術講演会, 2007 (掲載予定).

[5] R. Featherstone, Robot Dynamics Algorithms, Kluwer Academic Publishers, 1987.

[6] D.E. Rosenthal: “An Order n Formulation for Robotic Systems,” the Journal of the Astronautical Sciences, vol. 38, no. 4, pp.511-529, 1990.

[7] D. Baraff: “Linear-Time Dynamics Using Lagrange Multipliers,” Proceedings of SIGGRAPH '96, pp. 137-146, 1996.

[8] A. Fijany, I. Sharf, and G.M.T. D'Eleuterio: “Parallel O(logN) Algorithms for Computation of Manipulator Forward Dynamics,” IEEE Transactions on Robotics and Automation, vol. 11, no. 3, pp. 389-400, 1995.

[9] R. Featherstone: “A Divide-and-Conquer Articulated-Body Algorithm for Parallel O(log n) Calculation of Rigid-Body Dynamics. Part1: Basic Algorithm,” International Journal of Robotics Research, vol. 18, no. 9, pp. 867-875, 1999.

[10] K. Yamane and Y. Nakamura: “Automatic Scheduling for Parallel Forward Dynamics Computation of Open Kinematic Chains,” Robotics: Science and Systems, 2007.

[11] J. Lloyd: “Fast Implementation of Lemke's Algorithm for Rigid Body Contact Simulation,” IEEE International Conference on Robotics and Automation, pp. 4538-4543, 2005.

[12] K.G. Murty, Linear Complementarity: Linear and Nonlinear Programming, Heldermann-Verlag, 1988.