

仮想仕事の原理を用いた並列化可能な高速順動力学計算法

○ 山根 克 中村 仁彦 (東京大学大学院工学系研究科 機械情報工学専攻)

Parallelizable Efficient Forward Dynamics Computation Based on the Principle of Virtual Work

*Katsu Yamane and Yoshihiko Nakamura (Dept. of Mechano-Informatics, Univ. of Tokyo)

Abstract— $O(N)$ and parallel algorithms for forward dynamics of kinematic chains have wide range of application such as humanoid robots and hyper-redundant manipulators. This paper presents a new algorithm based on the Principle of Virtual Work with serial $O(N)$ and parallel $O(\log N)$ complexities, which is comparable to the most efficient algorithms known.

Key Words— Forward Dynamics, $O(N)$ Complexity, Parallel Computation, Principle of Virtual Work.

1 はじめに

計算量がリンク数に比例する, すなわち $O(N)$ の順動力学計算法 [1, 2] は特に自由度の高い機構の動力学シミュレーションに有効である. 最近では並列計算を仮定した計算時間が $O(\log N)$ の方法も提案されている [2, 3, 4]. これらの方法は線形代数的手法を用いたもの [2, 4] と Articulated Body Inertia (ABI) [1] を応用したものの [3] に大別できる.

これに対し, 本論文では仮想仕事の原理に基づく $O(N)$ 計算法を提案する. 計算の順序を入れ替えると $O(N)$ 個のプロセスによる並列計算で $O(\log N)$ の計算時間を実現することもできる. この方法の特徴は, 任意の機構に適用できることと, 特殊な線形代数の手法や ABI を用いず, 通常の物理量だけで計算できることである. 本論文では簡単のため枝のないシリアル機構についてシリアル・並列計算法の概略を述べる.

2 計算法の概要

2.1 リンクの追加に必要な計算

Fig.1 のように, $i+1$ 個のリンクが i 個の拘束によって結合されたシリアルリンク系 p の先端リンク i に, 拘束 $i+1$ を介してリンク $i+1$ を結合することを考える.

まずリンク k に関する以下の変数を定義する.

- A_k : 慣性モーメント
- b_k : 速度項・重力項
- f_k : リンクに働く外力
- $\ddot{\theta}_k$: リンクの加速度

拘束 m がリンク $m-1$ とリンク m の相対運動の 6 自由度のうち N_{Cm} 自由度を拘束しているとして, 以下の変数を定義する.

- $\dot{r}_{Cm}$: 拘束点の速度
 - n_m : 拘束力
 - $H_{Cm,k} \triangleq \partial r_{Cm} / \partial \theta_k$ ($k = m-1, m$)
 - $H_{m,k}$: $H_{Cm,k}$ のうち拘束自由度に対応する行
- リンク系 p が独立のときを考える. リンク k ($k = 0, 1, \dots, i$) は拘束 $k, k+1$ から力を受けるので, その運動方程式は

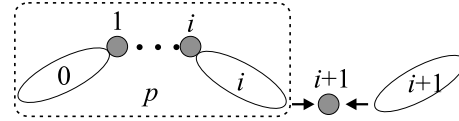


Fig.1: Adding a link

$$A_k \ddot{\theta}_k + b = f_k + H_{k,k}^T n_k + H_{k+1,k}^T n_{k+1} \quad (1)$$

となる. また, 拘束 m ($m = 1, 2, \dots, i$) に関する拘束条件は次式で与えられる.

$$H_{m,m-1} \ddot{\theta}_{m-1} + H_{m,m} \ddot{\theta}_m + \dot{H}_{m,m-1} \dot{\theta}_{m-1} + \dot{H}_{m,m} \dot{\theta}_m = 0 \quad (2)$$

これらをリンク 0 から i についてまとめると

$$\Lambda_i \ddot{\Theta}_i = T_i + \Gamma_i^T N_i \quad (3)$$

$$\Gamma_i \ddot{\Theta}_i = \gamma_i \quad (4)$$

$$T_i \triangleq F_i - B_i$$

$$\gamma_i \triangleq -\dot{\Gamma}_i \dot{\Theta}_i$$

となる. また Γ_i はリンク系 p 全体としての拘束を表す行列で次のように書ける.

$$\Gamma_i = \begin{pmatrix} H_{1,0} & H_{1,1} & O & \cdots & O \\ O & H_{2,1} & H_{2,2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & O \\ O & \cdots & O & H_{i,i-1} & H_{i,i} \end{pmatrix}$$

ここへ拘束 $i+1$ を介してリンク $i+1$ を結合すると, 全体の運動方程式と運動学的な拘束式はそれぞれ

$$\begin{pmatrix} \Lambda_i & O \\ O & A_{i+1} \end{pmatrix} \begin{pmatrix} \ddot{\Theta}_i \\ \ddot{\theta}_{i+1} \end{pmatrix} = \begin{pmatrix} \Gamma_i^T & \Gamma_{i+1,i}^T \\ O & H_{i+1,i+1}^T \end{pmatrix} \begin{pmatrix} N_i \\ n_{i+1} \end{pmatrix} + \begin{pmatrix} T_i \\ t_{i+1} \end{pmatrix} \quad (5)$$

$$\begin{pmatrix} \Gamma_i & O \\ \Gamma_{i+1,i} & H_{i+1,i+1} \end{pmatrix} \begin{pmatrix} \ddot{\Theta}_i \\ \ddot{\theta}_{i+1} \end{pmatrix} = \begin{pmatrix} \gamma_i \\ h_{i+1} \end{pmatrix} \quad (6)$$

$$\Gamma_{i+1,i} \triangleq \begin{pmatrix} O & \cdots & O & H_{i+1,i} \end{pmatrix} \quad (7)$$

となる．式(5)を $(\ddot{\mathbf{O}}_i^T \ddot{\theta}_{i+1}^T)^T$ について解き，式(6)に代入すると次式を得る．

$$\begin{pmatrix} \Phi_i & \Phi_{i+1,i}^T \\ \Phi_{i+1,i} & S_{i+1} \end{pmatrix} \begin{pmatrix} N_i \\ n_{i+1} \end{pmatrix} = \begin{pmatrix} \alpha_i \\ a_{i+1} \end{pmatrix} \quad (8)$$

$$\begin{aligned} \Phi_i &\triangleq \Gamma_i \Lambda_i^{-1} \Gamma_i^T \\ \Phi_{i+1,i} &\triangleq \Gamma_{i+1,i} \Lambda_i^{-1} \Gamma_i^T \\ S_{i+1} &\triangleq \Gamma_{i+1,i} \Lambda_i^{-1} \Gamma_{i+1,i}^T + H_{i+1,i+1} A_{i+1}^{-1} H_{i+1,i+1}^T \\ &= H_{i+1,i} A_i^{-1} H_{i+1,i}^T + H_{i+1,i+1} A_{i+1}^{-1} H_{i+1,i+1}^T \\ \alpha_i &\triangleq \gamma_i - \Gamma_i \Lambda_i^{-1} T_i \\ a_{i+1} &\triangleq h_{i+1} - H_{i+1,i+1} A_{i+1}^{-1} t_{i+1} \end{aligned}$$

式(8)の上半分より

$$N_i = N_i^i - \Phi_i^{-1} \Phi_{i+1,i}^T n_{i+1} \quad (9)$$

$$N_i^i \triangleq \Phi_i^{-1} \alpha_i \quad (10)$$

が得られる． N_i^i はリンク系 p が独立であるときの拘束力を表す．これを下半分に代入すると次式が得られる．

$$n_{i+1} = M_{i+1}(a_i - \Phi_{i+1,i} N_i^i) \quad (11)$$

$$M_{i+1} \triangleq (S_{i+1} - \Phi_{i+1,i} \Phi_i^{-1} \Phi_{i+1,i}^T)^{-1} \quad (12)$$

式(7)より $\Phi_{i+1,i}$ の構造を調べると

$$\begin{aligned} \Phi_{i+1,i} &= \Gamma_{i+1,i} \Lambda_i^{-1} \Gamma_i^T \\ &= \begin{pmatrix} O & \dots & O & H_{i+1,i} A_i^{-1} \end{pmatrix} \Gamma_i^T \\ &= \begin{pmatrix} O & \dots & O & S_{i+1,i} \end{pmatrix} \end{aligned} \quad (13)$$

$$S_{i+1,i} \triangleq H_{i+1,i} A_i^{-1} H_{i,i}^T \quad (14)$$

となっていることがわかる．これを用いて式(11),(12)を計算し直すと

$$n_{i+1} = M_{i+1}(a_{i+1} - S_{i+1,i} n_i^i) \quad (15)$$

$$M_{i+1} = (S_{i+1} - S_{i+1,i} M_i^{-1} S_{i+1,i}^T)^{-1} \quad (16)$$

となる．以上から，どの i に対しても同じ計算量で n_{i+1} が計算できることが確認された．

次にこの結果をリンク i から0に与え，全拘束力を確定する．式(13)などを用いて式(9)から拘束 i に関する部分を抜き出すと

$$n_i = n_i^i - M_i S_{i+1,i}^T n_{i+1} \quad (17)$$

が得られ， n_{i+1} から n_i を計算することができる．同様の計算を順に拘束1まで行くと全拘束力が計算される．全ての拘束力が求まると，式(1)よりリンクの加速度が計算される．

2.2 再帰的 $O(N)$ 計算法

1. リンクに関する forward path : $i = 0 \rightarrow N$ について A_i, A_i^{-1}, b_i, t_i を計算．
2. 拘束に関する forward path : $i = 1 \rightarrow N$ について $H_{i,i-1}, H_{i,i}, h_i$ と以下を計算．

$$\begin{aligned} a_i &= h_i - H_{i,i-1} A_{i-1}^T t_{i-1} - H_{i,i} A_i^T t_i \\ S_i &= H_{i,i-1} A_{i-1}^{-1} H_{i,i-1}^T + H_{i,i} A_i^{-1} H_{i,i}^T \\ S_{i,i-1} &= H_{i,i-1} A_{i-1}^{-1} H_{i-1,i-1}^T \\ M_i &= (S_i - S_{i,i-1} M_{i-1} S_{i,i-1}^T)^{-1} \\ n_i^i &= M_i(a_i - S_{i,i-1} n_{i-1}^i) \end{aligned}$$

3. 拘束に関する backward path : $i = N \rightarrow 1$ について初期条件を $n_N = n_N^N$ として以下を計算．

$$n_i = n_i^i - M_i S_{i+1,i}^T n_{i+1} \quad (18)$$

4. リンクに関する forward path : $i = 0 \rightarrow N$ について以下を計算．

$$\ddot{\theta}_i = A_i^{-1}(t_i + H_{i,i}^T n_i + H_{i+1,i}^T n_{i+1}) \quad (19)$$

2.3 並列 $O(\log N)$ 計算法

前節の方法では拘束を順に1個ずつ追加していくため並列計算はできない．しかし，なるべく遠い拘束から追加していくと，途中で現れるリンク機構が独立になるため並列計算が可能になる．この場合には，もともと複数のリンクからなるリンク系どうしを新しい拘束で接続するため， M_i, n_i^i 以外の成分が次のステップで必要となり，全体の計算量は前節の方法よりも多い．しかし，1回の接続に必要な計算量はもとの2つのリンク系の大きさによらないので，リンク数が 2^M 個のとき $O(M)$ 個のプロセスで並列計算を行うと計算時間は $O(M)$ となる．したがって，リンク数 N に対する計算量は $O(\log N)$ である．

3 おわりに

本論文では仮想仕事の原理に基づいたリンク系の効率的な順動力学計算法を提案した．この方法は任意の機構に適用でき，線形代数の手法やABIを使わずにシリアル計算で $O(N)$ ，並列計算で $O(\log N)$ の効率を達成している．

本研究は科学技術振興事業団の戦略的基礎研究推進事業「自律的行動単位の力学的結合による脳型情報処理機械の開発」の支援を受けた．また第一著者は日本学術振興会の特別研究員である．

参考文献

- [1] R. Featherstone. *Robot Dynamics Algorithm*. Kluwer Academic Publishers, Boston, MA, 1987.
- [2] A. Fijany, I. Sharf, and G.M.T. D'Eleuterio. Parallel $O(\log N)$ Algorithms for Computation of Manipulator Forward Dynamics. *IEEE Transactions on Robotics and Automation*, 11(3):389–400, 1995.
- [3] R. Featherstone. A Divide-and-Conquer Articulated-Body Algorithm for Parallel $O(\log(n))$ Calculation of Rigid-Body Dynamics. *International Journal of Robotics Research*, 18(9):867–892, September 1999.
- [4] K.S. Anderson and S. Duan. Highly parallelizable low order algorithm for the dynamics of complex multi-rigid-body systems. *AIAA Journal on Guidance, Control and Dynamics*, 2000.