

Just-In-Time 手法による非線形 DD アームの追従制御

牛田 俊, 出口 光一郎 (東北大学)

Tracking Control of Nonlinear Direct Drive Arm Based on Just-In-Time Method

*Shun Ushida, Koichiro Deguchi (Tohoku University)

Abstract— This paper aims to establish the experimental validity of the two degrees of freedom adaptive tracking control via Just-In-Time (JIT) methods for a nonlinear two-links direct drive (DD) arm system. This architecture has various advantages for its identification and control of a nonlinear object in contrast with typical global modeling approaches. In this paper, we propose new algorithm of JIT control to track the DD arm to desired trajectory. These results suggest the possibility of applying our method to more complicated nonlinear control systems.

Key Words: Just-In-Time modeling, Nonlinear Identification, Direct Drive Arm

1 はじめに

制御を行うためには、制御対象の性質や動特性を詳しく知る必要がある。すなわち、制御対象の物理的な特性を数学モデルで表現することにより、現代制御理論やロバスト制御理論等といったモデルベースな手法が適用可能となる。しかし、多くの場合において、事前に知ることのできる制御対象の情報は限られており、その構造が複雑になればなるほどモデル化は困難となる。たとえば、数学モデル内に含まれる種々のパラメータを正確に求める問題は、システム同定における重要かつ基本的な問題の一つであるが、非線形システム、時変システムといった複雑なシステムに対する体系的な手法は確立しているとは言い難い。

本論文では、世の中に存在する制御対象は全て何らかの非線形特性をもっていることを考慮し、非線形システムに対するモデリング手法を考える。特に、近年の計算機ハードウェアの急速な発展を利用した Just-In-Time モデリング (以下、JIT モデリング) と呼ばれる新しい方法を取り扱う。

従来のモデリング手法においては、制御対象を詳しく観察し、力学・電気工学・電磁気学・流体工学・熱力学・化学といった自然科学の広い分野の知識をフルに活用して、大域的なモデルが構成される。この場合、制御対象の構造や物理現象に関する十分な事前情報が必要となる。一方、大域的なモデルを求める代わりに、システムの過去の入出力データを大量に蓄積し同定の必要性が生じた場合のみ蓄積されたデータから局所モデルを構成する方法が、JIT モデリングである。この手法では、システムを表現しているのは大域的なモデルではなく、蓄積された入出力データ自身である。このことは、システムの入出力データこそがシステムの大域的に複雑な特性をよりよく表現する、という考えに基づいている。また、先に述べた通り、JIT モデリングの実装には、近年のコンピュータ技術の発展による大量のデータ保持、および高速な検索が可能になったこ

とが大きく寄与している。[1][2] では、JIT モデリングの同定精度を向上させるために、同定に使用する蓄積データの個数の決定方法、その選択方法、局所モデルの構成法を改良した JIT モデリングの様々なバリエーションが提案されている。さらに、[3][4] において、JIT モデリングを用いた追従制御器設計法が提案された。

本稿では、JIT モデリングを用いて、動特性に強い非線形性をもつダイレクトドライブアーム (以下、DD アーム) の出力予測器を構築し、非線形 DD アームの同定と制御に対して JIT 手法が適用可能であるかの検証を行う。さらに、DD アームの手先位置を目標の軌跡に一致させる追従制御を行うために、JIT モデリングを用いた JIT 制御系を実現する。ここでは、[3] で提案されたアルゴリズムを改良することにより、JIT モデリングの予測精度の向上を目指す。

2 問題の定式化

2.1 DD アームの運動方程式

Fig. 1 に示す水平面内を動く 2 自由度 DD アームを考える。各軸 ($i = 1, 2$) の角度、トルク、重心の質量、

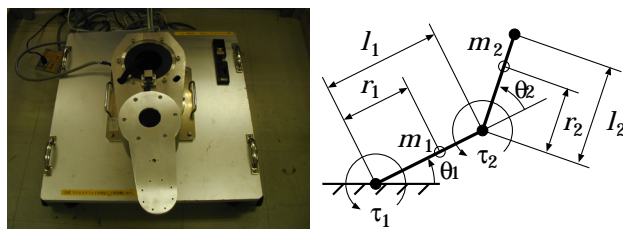


Fig. 1: 2 links manipulator

重心までの距離、各軸の長さ、重心まわりの慣性モーメントをそれぞれ θ_i , τ_i , m_i , r_i , l_i , I_i とすると、DD アームの運動方程式は、次式で表される。

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + B\dot{\theta} + D(\theta) = \tau \quad (1)$$

ただし, $\theta := [\theta_1 \ \theta_2]^T$, $\tau := [\tau_1 \ \tau_2]^T$

$$M(\theta) := \begin{bmatrix} M_1 + 2R \cos \theta_2 & M_2 + R \cos \theta_2 \\ M_2 + R \cos \theta_2 & M_2 \end{bmatrix}$$

$$C(\theta, \dot{\theta}) := \begin{bmatrix} -2R\dot{\theta}_1\dot{\theta}_2 \sin \theta_2 - R\dot{\theta}_2^2 \sin \theta_2 \\ R\dot{\theta}_1^2 \sin \theta_2 \end{bmatrix}$$

$$B := \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix}, \quad D(\dot{\theta}) := \begin{bmatrix} D_1 \text{sgn}(\dot{\theta}_1) \\ D_2 \text{sgn}(\dot{\theta}_2) \end{bmatrix}$$

$$M_1 := I_1 + I_2 + m_1 r_1^2 + m_2 (l_1^2 + r_2^2)$$

$$M_2 := I_2 + m_2 r_2^2, \quad R := m_2 r_2 l_1$$

である。(5)式の左辺の各項はそれぞれ、慣性力 $M(\theta)\ddot{\theta}$ 、遠心力・コリオリ力 $C(\theta, \dot{\theta})$ 、粘性摩擦力 $B\dot{\theta}$ 、動摩擦力 $D(\dot{\theta})$ を表しており、それらがDDアームの発生するトルク τ とつりあっている。

2.2 JIT モデリング

同定対象から新しいデータが得られるたびに同定モデルを更新するオンライン同定法は、一般に Eager Learning と呼ばれている。これに対して、JIT モデリングでは、新たに得られたデータに対しては蓄積のみを行い、同定の必要性が生じた場合に限り、次に述べる手順により蓄積データを利用して局所モデルを構成する。このような同定方法は Lazy Learning と呼ばれている [5]。

次式で表される離散時間非線形システムを考える。

$$y(t) = f(\phi(t-1)) + e(t), \quad t = 0, 1, 2, \dots \quad (2)$$

ここで、 $y(t)$ はシステムの出力、 $f(\cdot)$ は非線形写像、 $e(t)$ は観測雑音であり、 $\phi(t)$ は情報ベクトルと呼ばれている。情報ベクトル $\phi(t)$ は、システムの過去の入出力データを用いて次式で定義される。

$$\phi(t-1) := \begin{bmatrix} y(t-1) & \dots & y(t-n_y) \\ u(t-1) & \dots & u(t-n_u) \end{bmatrix}^T \quad (3)$$

ここで、 $u(t)$ はシステムの入力であり、 n_y 、 n_u はそれぞれシステムの出力と入力の次数であり、既知であると仮定する。JIT モデリングでは、(3) 式の情報ベクトルの形式でデータの蓄積が行われる。

時刻 t において、(2) 式のシステムの出力 $y(t)$ の予測を JIT 手法により行うこと考える。このときの $\phi(t)$ は要求点 (Query, 以後この $\phi(t)$ を q で表す) と呼ばれ、この要求点に類似した情報ベクトルを蓄積データから選び出し、近傍として採用する。情報ベクトルの類似度は、情報ベクトル間の距離 d を重みつき \mathcal{L}_1 ノルム

$$d(\phi(i), \phi(j)) := \sum_{l=1}^{n_y+n_u} \left| \frac{\phi_l(i) - \phi_l(j)}{\max_m |\phi_l(m)|} \right| \quad (4)$$

の近さによって測る。ここで、 $\phi_l(t)$, $l = 1, 2, \dots, (n_y + n_u)$ は時刻 t における情報ベクトルの要素を表す。近傍の最も単純な選び方は、要求点からもっとも近い k 個の情報ベクトルを集める方法であり、 k -NN (k Nearest Neighbors) と呼ばれている [5]。 k -NN 以外にも、同定精度を向上させるために k -BN (k Bipartite Neighbors), k -BN2, k -SN (k Surrounding Neighbors), k -SBN (k Surrounding Bipartite Neighbors) という様々なバリエーションが提案されている [1][2]。

要求点 q に対する近傍が得られたとき、局所モデルを構成する代表的な方法として、重み付き局所回帰法 (LWR) と重み付き線形平均法 (LWA) が挙げられる：

$$\text{LWR: } \begin{cases} \hat{y}(t; q) = \phi^T(t; q) \hat{\theta} \\ \hat{\theta} := \arg \min_{\theta} \sum_{i=1}^k w_i |y(i) - \hat{y}(i)|^2 \end{cases} \quad (5)$$

$$\text{LWA: } \hat{y}(t; q) = \sum_{i=1}^k w_i y(i), \quad \sum_{i=1}^k w_i = 1 \quad (6)$$

ただし、 $\hat{y}(i)$ は局所モデルによる出力の推定値であり、 k は近傍のサイズである。各要求点に対する近傍のサイズ k は一定である必要はないが、現実問題に実装する際には、計算量を減らすためにある値に固定しても構わない。近傍のサイズ k により、JIT 手法による同定精度がどのように変化するかを定量的に見積もることは今後の課題として残されている。

JIT モデリングの概念図を Fig. 4 に示す。JIT モデ

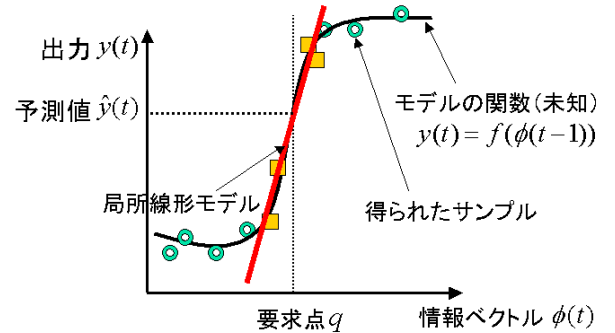


Fig. 2: JIT Modeling

リングの重要なポイントは、選択された近傍がもつ時刻と要求点の時刻 t との非依存性にある。すなわち、近傍の決定は要求点 q との距離のみによって行われるため、データがいつ蓄積されたかということには依存しない。さらに、各要求点 q 毎に局所モデルが構成されるため、 q に対する出力の予測後は局所モデルはその都度破棄されることが、従来の同定法と大きく異なる。

2.3 JIT 制御による DD アームの制御

Fig. 3 のような 2 自由度適応制御系を考える。ただし、 $x_d := [\theta_d \ \dot{\theta}_d]^T \in \mathbb{R}^2$, $x := [\theta \ \dot{\theta}]^T \in \mathbb{R}^2$ は角度、角速度からなる信号であり、添字の d は目標値を意味する。

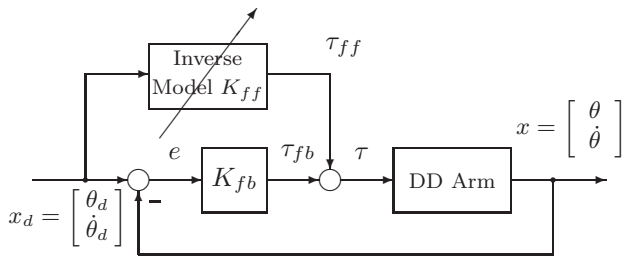


Fig. 3: Tracking Control via FF/FB Controller

Fig. 3の適応制御系は，固定された線形フィードバック制御器 K_{fb} と，適応型フィードフォワード制御器 K_{ff} から構成されている． K_{fb} を含むフィードバックループは，制御系の安定性および制御性能の向上に寄与しているが，それらには制御対象の非線形に起因する限界が生ずる．そこで， K_{ff} を前節で述べた JIT モデリングに基づいて，制御対象の逆モデルとなるように適応的に構築する方法が，[3][4]において提案された JIT 制御である．もし K_{ff} として制御対象の完全な逆モデルが得られたとすると， $x_d = x$ ，すなわち $e = 0$ となり，フィードバック K_{fb} の影響は 0 となる．結果として， K_{ff} によって，目標値 x_d を達成する制御入力 τ が得られる．JIT モデリングにより，フィードフォワード制御器 K_{ff} が対象の逆モデルとして機能するが，その同定精度は過去の動特性を含む情報ベクトルが増えれば増えるほど向上することが期待される．これは，人間の運動における学習機能と非常に類似していると考えられる [6][7]．

3 JIT モデリングによる順モデルの同定結果

2.2 節で述べた JIT モデリングによる同定が，DD アームに適用可能かどうかを検証するために，Fig. 4 (a) のような順方向モデルの同定実験を行う．ここで， $\tau \in \mathbb{R}^2, \theta \in \mathbb{R}^2$ はそれぞれ各軸の入力トルク，観測される角度であり，

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}, \quad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}.$$

で与えられる．2 軸のアームの順モデルは，次式で与

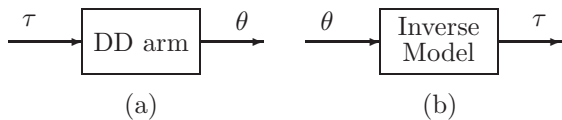


Fig. 4: Model and Inverse Model
えられる．

$$\theta(t) = f(\phi(t-1)) \quad (7)$$

ここで， $n_y = 2, n_u = 1$ とすると，情報ベクトル $\phi(t-1)$ は次のようになる．

$$\phi(t-1) = \begin{bmatrix} \theta_1(t-1) & \theta_2(t-1) \\ \theta_1(t-2) & \theta_2(t-2) & \tau_1(t-1) & \tau_2(t-1) \end{bmatrix}^T \quad (8)$$

実機の 1 軸および 2 軸に，ランダム周期で正負が反転するような矩形形状の入力を加え，1000 個の入出力デー

タを採取する．この 1000 個のデータの中で初めの 900 個に対して (8) 式で定義される情報ベクトルを構成し，蓄積データとして保存する．残りの 100 個の入力データは JIT モデリングの検証用データとして利用した．すなわち，100 個の入力データから新しい要求点が生成され，それらに対して JIT モデリングを行った．近傍のサイズは計算量を減らすために $k = 9$ に固定し，近傍の選択は k -NN とした．局所モデルの構成法として (5) 式の LWR を採用した．上記の設定の下で，実機の出力データと JIT モデリングによる出力の予測値との比較を行い，良好な同定結果を得た (Fig. 5)．このことは，非線形性の強い DD アームの特性が，JIT モデリングによって同定可能であることを示している．

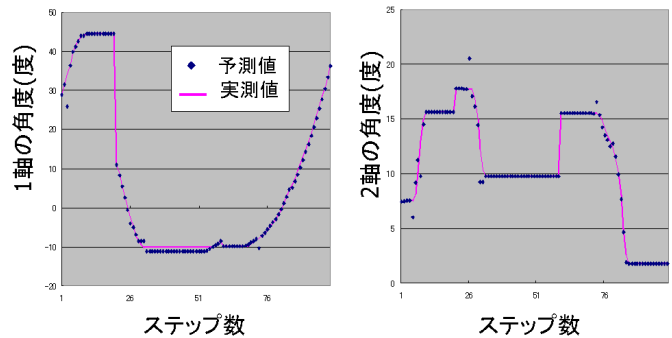


Fig. 5: JIT Modeling of DD Arm

4 JIT 手法による 2 自由度制御系の設計

2.3 節で述べた JIT 制御による DD アームの追従制御を行う．Fig. 3 に示す 2 自由度制御系のアーキテクチャに対して，DD アームの逆モデル K_{ff} を JIT モデリングにより構成する．すなわち，Fig. 4 (b) のような逆方向モデルを考えると，逆モデルは次式で与えられる．

$$\tau(t) = f^{-1}(\bar{\phi}(t+1)) \quad (9)$$

情報ベクトル $\bar{\phi}(t+1)$ は，(7), (8) 式より未来の角度出力 $\theta_1(t+1), \theta_2(t+1)$ を含むが，JIT 制御では目標角度 $\theta_{d1}(t+1), \theta_{d2}(t+1)$ を代入することにより， $\bar{\phi}(t+1)$ が要求点の役割を果たす．したがって，情報ベクトル $\bar{\phi}(t+1)$ は次式で定義される．

$$\bar{\phi}(t+1) := \begin{bmatrix} \theta_{d1}(t+1) & \theta_{d2}(t+1) \\ \theta_1(t) & \theta_2(t) & \theta_1(t-1) & \theta_2(t-1) \end{bmatrix}^T \quad (10)$$

本論文では，(10) 式を拡張し，JIT 制御系内の逆モデルを構成する際に，次の情報ベクトルを用いる．

$$\bar{\phi}(t+1) := \begin{bmatrix} \theta_{1d}(t+1) \\ \theta_{2d}(t+1) \\ \theta_{1d}(t+1) - \theta_1(t) \\ \theta_{2d}(t+1) - \theta_2(t) \\ \theta_{1d}(t+1) - 2\theta_1(t) + \theta_1(t-1) \\ \theta_{2d}(t+1) - 2\theta_2(t) + \theta_2(t-1) \end{bmatrix}^T \quad (11)$$

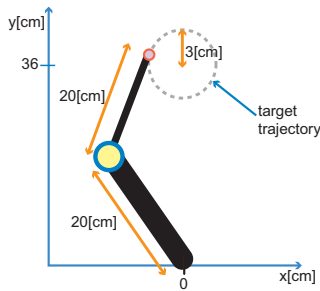


Fig. 6: Target trajectory of tracking control

(11) 式の入力 $\tau(t)$ が、目標角度を達成するトルクとして DD アームに入力される。(11) 式は、DD アームの角度の速度情報と加速度情報を利用していることに相当し、これによって要求点に対する近傍としてより適切なデータを選択することが可能になる。

以下では、実機に対する JIT 制御を行う代わりに、(1) 式の DD アームの運動方程式を差分近似により表現したシミュレータに対する JIT 制御の追従制御の実験を行った。フィードバック制御器 K_{fb} としては比例制御 P を採用し、そのゲインを $P = 10$ に設定した。 $P = 0$ の場合はフィードバックループは切断され、純粋なフィードフォワード制御機構となる。手先の目標軌跡は半径 3[cm] の円である (Fig. 6)。JIT モデリングでは、過去の蓄積データがない場合には原理的に同定を行うことができないため、あらかじめ事前情報として 10 個の蓄積データが与えられていると仮定した。近傍の個数 k は一定値 ($k = 30$) とする。上記の設定の下で、追従制御を行った結果を Fig. 7 に示す。

1 軸、2 軸ともに、事前情報として用意する蓄積データが少ないにもかかわらず、良好な追従制御結果が得られている。このことは、フィードフォワード制御器において、JIT モデリングによる逆モデルの同定アルゴリズムが十分に精度よく機能していることを意味している。JIT 手法の原理的な欠点として、蓄積データの少ない初期段階では同定誤差が大きくなり、結果として、追従制御の精度が低下することが挙げられる。実機に適用する際には、この誤差は大きな障害と考えられるが、実機への入力トルクにリミッタを設けることで、危険な状態を回避することが可能である。また、

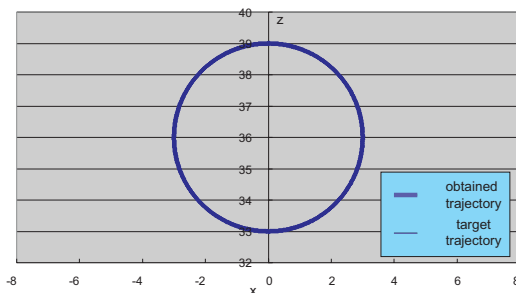


Fig. 7: JIT control considering angular velocity

定量的に誤差を評価するための理論解析が難しいことも、今後の解決すべき大きな問題のひとつである。これは、FAN がもつ問題点と本質的に同じである。

5 おわりに

強い非線形ダイナミクスが含まれる DD アームに対して、JIT 制御による追従制御の実現可能性を検証した。JIT 手法による DD アームの順モデルおよび逆モデルの同定結果は、JIT 制御が十分な性能を達成可能であることを示唆している。さらに、情報ベクトルの取り方に同定精度に大きな影響を及ぼすことも明らかになった。

本稿では、DD アームのシミュレータに対する実験を行ったが、JIT 手法の性能を評価するためには、実機による検証も必要不可欠である。DD アームについては、その動特性の大域モデルを利用した方法が数多く提案されている SICE97 が、それらの手法との比較も行う必要がある。DD アームは正確なパラメータ同定の難しいため、事前情報を必要としない JIT 制御系の優位性が期待される。

参考文献

- [1] 鄭 秋宝, 木村 英紀: Just In Time モデリングの新しい手法とその圧延セットアップモデルへの応用, 計測自動制御学会論文集, Vol. 37, No. 7, (2001)
- [2] Q. Zheng and H. Kimura: Just-In-Time Modeling for Function Prediction and its Applications, *Asian J. Control.* Vol. 3, No. 1, 35/44 (2001)
- [3] 牛田 俊, 河村 大輔, 鄭 秋宝, 木村 英紀: Just-In-Time 手法に基づく 2 自由度制御系の設計法, 第 2 回適応学習制御シンポジウム, pp. 23-26, 名古屋, 1 月 31 日, (2002)
- [4] Q. Zheng and H. Kimura: Direct Just-In-Time Methods for Nonlinear Control Design, *Proc. of 16th Korea Automatic Control Conference* (2001)
- [5] D. W. Aha: *Lazy Learning*, Boston, London: Kluwer Academic Publishers (1997)
- [6] A. Miyamura and H. Kimura: Feedback Error Learning Method with Time Delay. Synthesis Aspects of Cerebellum Motor Control, *Proc. of Int. Sym. Mathematical Theory of Networks and Systems*, Perpignan, (2000)
- [7] 牛田 俊, 岸本 渉, 木村 英紀: フィードバック誤差学習法による非線形 DD アームの適応制御, 計測自動制御学会 第 30 回制御理論シンポジウム予稿集, 大分, 85/88 (2001)
- [8] J. E. Slotine and W. Li: Composite Adaptive Control of Robot Manipulators, *Automatica*, Vol. 25, No. 4, 509/519 (1989)
- [9] 小林, 増田, 他: ロボット制御の実際, 計測自動制御学会編, コロナ社 (1997)