

非線形系の外乱推定法を用いたマニピュレータの運動制御

MYLANGA MADHURA RANGANA PERERA¹⁾, 羅 志偉³⁾, 渡部 慶二²⁾³⁾, 村松 鋭一²⁾³⁾
遠藤 茂²⁾

Zhi-wei Luo³⁾ Keiji Watanabe²⁾³⁾ Eiichi Muramatsu²⁾³⁾ Shigeru Endo²⁾

¹⁾ 山形大学 大学院 理工学研究科 ²⁾ 山形大学 工学部 ³⁾ 理化学研究所

¹⁾²⁾ Yamagata University ³⁾ Riken

KEY WORDS: 外乱推定法 非線形系ロボットシステム 3リンクマニピュレータ 軌道生成器
状態推定オブザーバ 内外乱(雑音)

連絡先: 〒992-8510 山形県米沢市城南 4-3-16 山形大学工学部応用生命システム工学科渡部研究室

PERERA MADHURA, Tel 090-7332-4311, E-mail tr372@dip.vz.yamagata-u.ac.jp

1. はじめに

ロボットの商品化によって、サーボシステム技術・半導体技術・制御技術などは飛躍的な進歩を遂げ、その成果としてロボティクス技術は躍進的に発展している。

近年、これまで人間が行ってきた複雑な作業を代行するロボットが増えてきている。特に宇宙、深い海底、原子炉内など.....という人間が入り込めないところの作業に使用できる、マニピュレータロボットの研究開発が必要とされている。特に、最近のロボットの性能向上では、高速化と高精度化が顕著であり、その他に動作速度、軌跡精度、信頼性、操作性、動作範囲、という技術的な分野での改善が求められる。また、上記の性能と他に低価格化、ロボットの小型化も求められている。

本研究では3自由度DDロボットを対象により正確かつ迅速な軌道制御を行うことを目標として考えている。

ロボットは各モータの出力軸にリンクがついたもので、モータの回転が思うように制御され

ば結果としてロボットは所望の運動が実現できると考えられる。このような考え方する場合リンク系である上に発生する力学的非線形な干渉力はすべてのモータの出力軸に加わる外乱であると考え、各関節のモータに非線形系の外乱推定法を用いたマニピュレータの運動制御を考える。

これは非線形ロボットシステムに対し、非線形外乱推定法による非線形システムを制御する新しい発想である。この制御方法をADRC(Active Disturbance Rejection Control)制御方法と呼ぶ。ADRCは主に三部分からなり、その部分は入力信号を設計するPG-TD(Profile Generator based on Tracking Differentiator)と慣性行列の誤差を修正するESO(Extended State Observer)部分と一般的なPD制御器であるGPD(Generalized Proportional-Derivative)からなっている。この制御方法を用いて下記の3リンクマニピュレータを制御することを目的としている。

2. システム構成

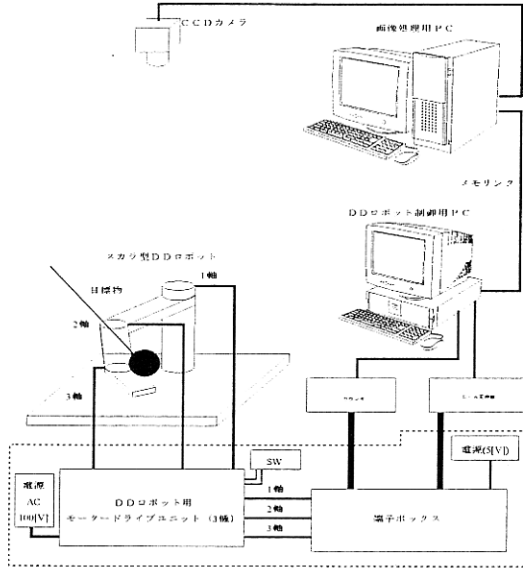


図 1. 実験システム構成

本研究で使用する実験装置の全体を図 2.1 に示す。このマニピュレータロボットシステムの構成図により、天井に取り付けられている CCD カメラから目標物体の画像を読み取り、その画像を画像処理用のコンピュータに送る。そこで、画像処理を行うことで、目標物体の位置情報を取り出す。その位置情報はメモリリンクを介して制御用のコンピュータに送信する。そして、画像処理用の計算機から送信されてきた位置情報の関節角度情報を逆運動学マップから目標関節角度情報を呼び出す。その目標関節角度を利用し、DD ロボットの関節運動制御を行い、目標物体に到達させる。

3. 外乱推定法

ADRC の制御方法について次のように説明します。二軸ロボットの運動方程式は次式で表わされる。

$$\tau = M(\theta)\ddot{\theta} + V(\theta, \dot{\theta}) + G(\theta) \quad (1)$$

ただし、

$$M(\theta) = \begin{bmatrix} l_2^2 m_2 + 2l_1 l_2 m_2 c_2 + l_1^2 (m_1 + m_2) & l_2^2 m_2 + l_1 l_2 m_2 c_2 \\ l_2^2 m_2 + l_1 l_2 m_2 c_2 & l_2^2 m_2 \end{bmatrix} \quad (2)$$

$$V(\theta, \dot{\theta}) = \begin{bmatrix} -m_2 l_1 l_2 s_2 \dot{\theta}_2^2 - 2m_2 l_1 l_2 s_2 \dot{\theta}_1 \dot{\theta}_2 \\ m_2 l_1 l_2 s_2 \dot{\theta}_1^2 \end{bmatrix} \quad (3)$$

$$G(\theta) = \begin{bmatrix} m_2 l_2 g c_{12} + (m_1 + m_2) l_1 g c_1 \\ m_2 l_2 g c_{12} \end{bmatrix} \quad (4)$$

ここで、コントローラ設計のときの唯一情報 θ を使い、 $M_0(\theta)$ を次のように設計する。

$$M_0(\theta) = \begin{bmatrix} l_{20}^2 m_{20} + 2l_{10} l_{20} m_{20} c_2 + l_{10}^2 (m_{10} + m_{20}) & l_{20}^2 m_{20} + l_{10} l_{20} m_{20} c_2 \\ l_{20}^2 m_{20} + l_{10} l_{20} m_{20} c_2 & l_{20}^2 m_{20} \end{bmatrix}$$

ADRC のシステム構成を図 2 に示す。

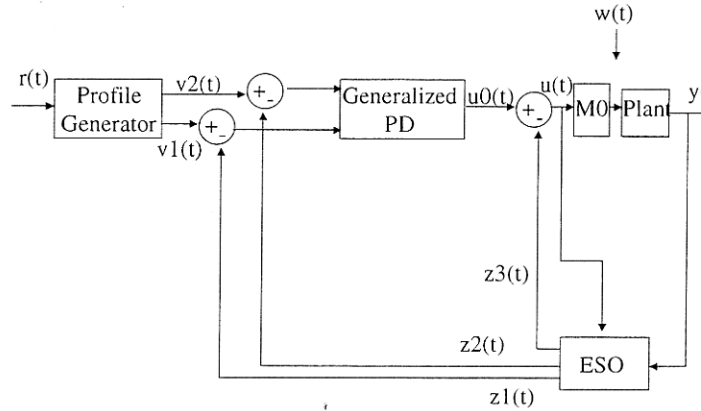


図 2. ADRC のシステム構成

図 2 のシステムにおいて各ブロックは以下の通りである。

Profile Generator : 軌道生成器

profile generator は次の式によって表される。

$$\dot{V}_1 = V_2 \quad (6)$$

$$\dot{V}_2 = -R \text{sign}(V_1(t) - r(t) + \frac{V_2 |V_2|}{2R}) \quad (7)$$

ここでは、唯一パラメータ R を使っている。 R の物理的な意味としては、 R の最大加速で Plant が設計される。 R が大きくなるほどロボットはすばやく目標軌道に追従する。ステップ信号を入力されると、 $V_1(t)$ の早いステップ応答を出すために最も早い道は R の設計である。ここで、setting time は $t_s = 2\sqrt{\frac{A}{R}}$ で A は step value である。また、その上に $V_1(t)$ のオーバーシュートない事である。

2. ESO：状態推定オブザーバ

ESO の運動方程式は次式で表される。

$$\dot{Z}_1 = Z_2 - \beta_{01}(Z_1 - \theta(t)) \quad (8)$$

$$\dot{Z}_2 = Z_3 - \beta_{02}fal(Z_1 - \theta(t), \alpha_1, \delta) + U \quad (9)$$

$$\dot{Z}_3 = -\beta_{03}fal(Z_1 - \theta(t), \alpha_2, \delta) \quad (10)$$

ただし、

$$fal(Z_1 - \theta, \alpha, \delta) = \begin{bmatrix} fal(z_{11} - \theta_1, \alpha, \delta) \\ fal(z_{12} - \theta_2, \alpha, \delta) \end{bmatrix} \quad (11)$$

$$Z_1 = \begin{bmatrix} z_{11} \\ z_{12} \end{bmatrix} \quad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \quad (12)$$

また、 fal 関数を次のように定義する。

$$fal(x, \alpha, \delta) = \begin{cases} |x|^\alpha sign(x), & |x| > \delta \\ x / \delta^{1-\alpha}, & |x| \leq \delta \end{cases} \quad (13)$$

また ESO からの出力は次式で表される。

$$Z_1(t) \Rightarrow \theta(t) \quad (14)$$

$$Z_2(t) \Rightarrow \dot{\theta}(t) \quad (15)$$

$$Z_3(t) \Rightarrow -M^{-1}(\theta)V(\theta, \dot{\theta}) - M^{-1}(\theta)G_1(\theta) \quad (16)$$

ESO の優れている点は、システムの内乱と外乱をひとまとめにした雑音として扱って除去することができる点である。

3. Generalized PD：

ここでの制御法は

$$U = -Z_3 + k_1(V_1 - \theta) + k_2(V_2 - \dot{\theta}) \quad (17)$$

$$\tau = M_0(\theta)U \quad (18)$$

である。

4. シミュレーション

シミュレーションは MATLAB6.1 の SIMULINK を用いて、各部分を生じ、結合させた。

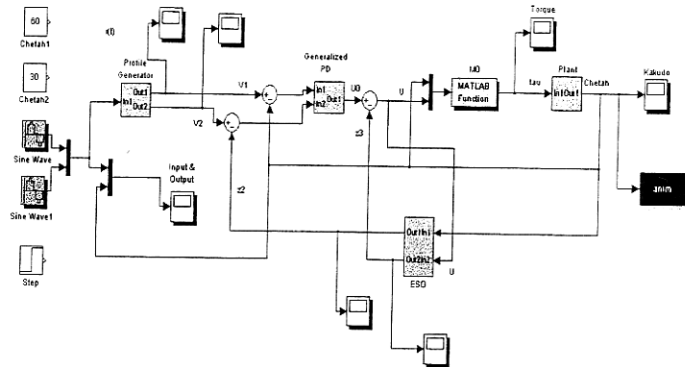


図3. 完成モデル

完成させたモデルを用いて次のようにシミュレーションを行った。

シミュレーション I：

入力信号には次のようなステップ信号を与えた。

$$\begin{aligned} \theta_{1d} &= 60^\circ \\ \theta_{2d} &= 30^\circ \end{aligned} \quad (19)$$

ここでは入力信号がサイン波を与えました。ただし、リンク I に対してリンク II に 90 度の位相を与えた。

$$\begin{aligned} \theta_{1d} &= \sin t \\ \theta_{2d} &= \sin(t + 90) \end{aligned} \quad (20)$$

シミュレーション II：

ただし、上記に雑音(外乱)を加えたときのシミュレーションを行う。

リンク I の条件；

$$\begin{aligned} l_1 &= l_{10}(1 + 0.2\sin(3t)) \\ m_1 &= m_{10}(1 + 0.2\cos(-3t)) \end{aligned} \quad (21)$$

リンク II の条件；

$$\begin{aligned} l_2 &= l_{20}(1 + 0.2\sin(6t)) \\ m_2 &= m_{20}(1 + 0.2) \end{aligned} \quad (22)$$

5. シミュレーション結果：

結果Ⅰ.

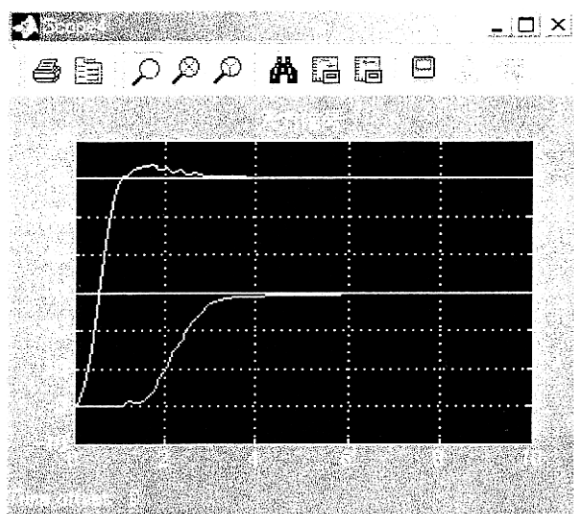


図4. シミュレーションⅠの結果

結果Ⅱ.

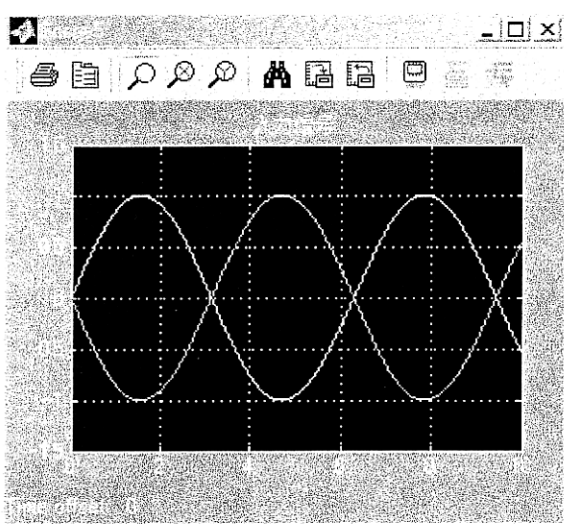


図5. シミュレーションⅡの入力波形

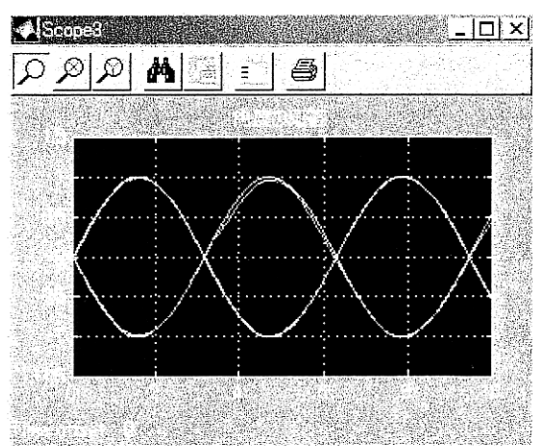


図3. シミュレーションⅡの入出力の比較波形

ADRC 制御法と比較するため同じ雑音を与えて PD 制御でシミュレーションを行った結果を図4に示す。

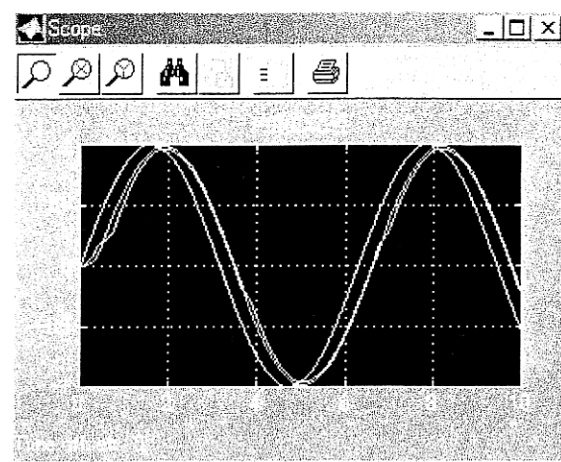


図6. PD 制御による入出力の比較波形

このような結果を考慮すると ADRC 制御法での外乱は非常に良く除去できている事がわかる。

5. おわりに：

本研究では非線形系の外乱推定法を用いた制御法で行ったシミュレーション結果からは外乱除去されて非常になめらかなかつ入力波形と非常に近い応答が得られました。ステップ入力の出力応答では少しオーバーシュートが発生し、振動的になっていますが、この点はパラメータ修正によって、少なくできるはずである。

今後の課題としては、現在この制御法を用いて VISUAL C++でプログラムを作成し、実験を行っている段階である。

6. 参考文献：

I. Active Disturbance Rejection Control of Robot Systems Based on Extended State Observer
論文 Yi Huang

Ⅱ. 制御系設計 西村秀和・平田光男共著 東京電気大学出版局

Ⅲ. 生体とロボットにおける運動制御 伊藤宏司・伊藤正美 共著 計測自動制御学会

Ⅳ. MATLAB ハンドブック 小林一行 著 株式会社秀和システム