

Particle Swarm Optimizationを用いた ジブクレーンのゲインスケジュールド制御

高見 勲, 石田 将一

Key Words : PSO , Polytopic, LMI, Gain-Scheduled Control, H₂ Control

1 はじめに

本研究の制御対象であるジブクレーンシステムは、吊荷を吊るしているロープの巻き上げ（下げ）を行うペイロードシステム、滑車の並進運動を行うジブシステムにより、2次元空間内で吊荷を任意の位置へ運搬するシステムである。また、本システムは吊荷のダイナミクスがパラメータに対して非線形に依存するという特徴を持つ。運搬のサイクルタイムを短縮するためにはこの2動作を同時に行うことと、ロープ長が変動した場合でも安定性や制御性の良さを保証する制御器が必要である。本システムのように観測可能な変動パラメータを含むシステムに対する有効な制御方法として、ゲインスケジュールド（以下 GS）制御が挙げられる [1]。また、制御対象のダイナミクスがパラメータに対して非線形に依存する場合は、パラメータ変換により依存性をアフィンにする方法 [2] や二乗和多項式を用いる方法 [3] などが有効である。本研究では本システムに対し、パラメータ変換と状態ベクトルの等価変換を適切に行うことで、安定性や制御性がロープ長変動に影響されない制御系を実現した。また、最適化手法のひとつである Particle Swarm Optimization（以下 PSO）を用いることで、H₂ 制御におけるパラメータチューニングを自動化した。また、その有用性をシミュレーションおよび実験により検証した。

2 制御対象

ジブシステム概略図を Fig.1 に示す。ジブシステムの状態空間表現は式 (1) である。

$$P(l) : \begin{cases} \dot{x} = A(l)x + B(l)u \\ y = C(l)x \end{cases}, x = \begin{bmatrix} x_j \\ \gamma \\ \dot{x}_j \\ \dot{\gamma} \end{bmatrix} \quad (1)$$

$$A(l) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & a & 0 & 0 \\ 0 & \frac{b}{l} & 0 & 0 \end{bmatrix}, B(l) = \begin{bmatrix} 0 \\ 0 \\ c \\ \frac{c}{l} \end{bmatrix} \quad (2)$$

$$C(l) = \begin{bmatrix} 1 & -l & 0 & 0 \end{bmatrix} \quad (3)$$

モデリングに用いた記号は x_j [m] : 滑車の位置, γ [rad] : 吊荷の振れ角, l [m] : ロープの長さ, u [A] : ジブモータへの入力電流, y [m] : 吊荷の水平方向の位置, a, b, c : 定数 である。ただし, $l \in [l_1, l_2]$ である。

3 制御系設計

GS 制御とは制御器の係数を、観測された可変パラメータの値と共に変化させることにより、常に最適な制御性能を実現する制御方法である。本研究では等価変換とパラメータ変換を用いることで、ポリトープ型 GS H₂ 制御器を設計した。以下にその設計法を示す。

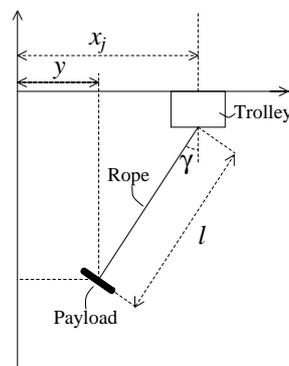


Fig. 1 Jib System Model

3.1 一般化制御対象の構成

応答特性の向上と目標値への追従および制御入力の加減を目的とし、一般化制御対象を Fig.2 のように構成した。その際に用いた記号は w : 目標値信号, W_x : 状態ベクトル x に対する重み, W_e : 偏差の積分 x_e に対する重み, W_u : 制御入力 u に対する重み, z_1, z_2, z_3 : 評価出力 である。Fig.2 より一般化制御対象は式 (4) と与えられる。

$$G(l) : \begin{cases} \dot{x}_G = A_G(l)x_G + B_{G1}w + B_{G2}(l)u \\ z = C_G x_G + D_{G2}u \end{cases} \quad (4)$$

$$x_G = \begin{bmatrix} x & x_e \end{bmatrix}^T, z = \begin{bmatrix} z_1 & z_2 & z_3 \end{bmatrix}^T \quad (5)$$

ここで、式 (4) の各係数行列の要素に含まれる可変パラメータは l と $1/l$ の 2 つである。

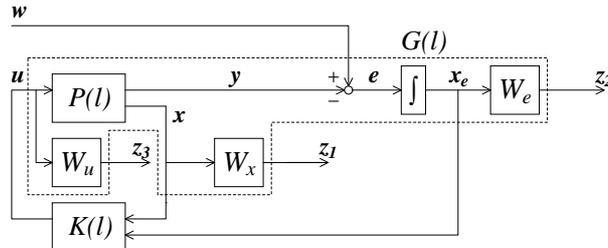


Fig. 2 Control System

3.2 一般化制御対象のポリトープ表現

本研究ではまず、 $T(l) = \text{diag} \begin{bmatrix} 1 & l & 1 & l & 1 & 1 \end{bmatrix}$ とし、新たに状態変数を $\hat{x}_G = T x_G$, 可変パラメータを $\sigma = 1/l$ とおくことで、式 (4) を LPV システムに変換した。次に、そのシステムをポリトープ型で表現した。これらの変換により、線形時不変系に対するロバスト制御理論に基づいての GS 制御器が設計可能となった。変換後のポリトープシステムは式 (6), (7) と与えられる。

$$\dot{\hat{x}}_G = \sum_{i=1}^2 \alpha_i A_i \hat{x}_G + \hat{B}_{G1} w + \hat{B}_{G2} u \quad (6)$$

$$\hat{z} = \sum_{i=1}^2 \alpha_i C_i \hat{x}_G + D_{G2} u \quad (7)$$

ただし、 $\alpha_1 = (\sigma_2 - \sigma)/(\sigma_2 - \sigma_1)$, $\alpha_2 = 1 - \alpha_1$, $\sigma \in [\sigma_2, \sigma_1]$, $\sigma_1 = 1/l_1$, $\sigma_2 = 1/l_2$ である。

3.3 ポリトープシステムに対する H_2 状態フィードバック GS 制御系設計

式 (6),(7) に対する制御器 $\hat{K}(\sigma)$ は次式となる。

$$\hat{K}(\sigma) = \sum_{i=1}^2 \alpha_i \hat{K}_i \quad (8)$$

また、一般化制御対象の閉ループ系は次式となる。

$$\dot{\hat{x}}_G = \sum_{i=1}^2 \alpha_i (A_i + \hat{B}_{G2} \hat{K}_i) \hat{x}_G + \hat{B}_{G1} w \quad (9)$$

式 (9) が H_2 制御仕様を満足するためには次の LMI を解けばよい。ただし $Y_i = \hat{K}_i X$ であり、 Y_i は変数行列である。また、 X は正定対称な変数行列である。

$$\begin{aligned} & \min \gamma \\ & \text{s.t.} \begin{bmatrix} A_i X + \hat{B}_{G2} Y_i + X A_i^T + Y_i^T \hat{B}_{G2}^T & \hat{B}_{G1} \\ & \hat{B}_{G1}^T & -I \end{bmatrix} < 0 \\ & \begin{bmatrix} X & X \hat{C}_i^T + Y_i^T D_{G2}^T \\ \hat{C}_i X + D_{G2} Y_i & W \end{bmatrix} > 0 \\ & \text{trace}(W) < \gamma^2 \end{aligned}$$

以上より，式 (4) に対する状態フィードバック制御器 K は次式で与えられる．

$$K = \hat{K}(\sigma)T \quad (10)$$

$$= \sum_{i=1}^2 \alpha_i Y_i X^{-1} T_i \quad (11)$$

4 PSO とは

粒子群最適化 (PSO) は生物の採餌行動を模擬した最適化手法で，微分可能性などの条件を必要とせず，目的関数値の情報のみで非線形最適化問題を解く事ができる直接探索型のアルゴリズムである [4]．PSO は多次元空間において，位置ベクトルと移動ベクトルを持つ particle(探索点) の群でモデル化される．これらの particle が情報交換をし，それに基づいて自身の位置と速度を調整しながら解空間を探索し，最適な位置を探す．

4.1 PSO の概要

n 次元空間における particle は，

$$\text{位置ベクトル } x_i = (x_{i1}, x_{i2}, \dots, x_{ij}, \dots, x_{in})^T, \text{ 移動ベクトル } v_i = (v_{i1}, v_{i2}, \dots, v_{ij}, \dots, v_{in})^T \quad (12)$$

で表される．ここで， i は particle の番号， j はベクトルの要素番号を表す．さらに，各 particle はそれまでの探索で発見した最良解 $pbest_i = (pbest_{i1}, pbest_{i2}, \dots, pbest_{ij}, \dots, pbest_{in})^T$ と，その評価値である $f(pbest_i)$ を記憶している．群としては，すべての particle がこれまでの探索で発見した最良解 $gbest = (gbest_1, gbest_2, \dots, gbest_j, \dots, gbest_n)^T$ と，その評価値である $f(gbest)$ を記憶している．これらのベクトルを用いて，各 particle は新たな移動ベクトル v_i^{k+1} を生成し，現在の位置 x_i^k から次の位置 x_i^{k+1} へ移動する．ここで， k は反復回数を表す．

4.2 PSO のアルゴリズム

以下に一般的な PSO のアルゴリズムを示す [6]．ここで， $i = 1, 2, \dots, m$ ， $j = 1, 2, \dots, n$ とする．ただし， m は particle の数である．

Step 0 : 準備

m ，および最大反復回数 T_{max} を与え， $k = 0$ とおく．

Step 1 : 初期化

各 particle の初期位置 x_i^0 と初期速度 v_i^0 をランダムに与える．

$$pbest_i^0 = x_i^0$$

$$gbest^0 = pbest_{i_g}^0$$

ただし， $i_g = \arg \min_i f(pbest_i^0)$

Step 2 : 位置と速度の更新

particle の位置と速度を以下の式に従って更新する．ただし， $\text{rand}_1()_{ij}, \text{rand}_2()_{ij}$ は 0 から 1 の間に分布する一様乱数であり， w, c_1, c_2 は各項目に対する重み係数である．

$$v_{ij}^{k+1} = w \cdot v_{ij}^k + c_1 \cdot \text{rand}_1()_{ij} \cdot (pbest_{ij}^k - x_{ij}^k)$$

$$+ c_2 \cdot \text{rand}_2()_{ij} \cdot (gbest_j^k - x_{ij}^k)$$

$$x_{ij}^{k+1} = x_{ij}^k + v_{ij}^{k+1}$$

Step 3 : $pbest$ と $gbest$ の更新

$$I = \{i \mid f(x_i^{k+1}) < f(pbest_i^k)\}$$

$$pbest_i^{k+1} = x_i^{k+1}, i \in I$$

$$pbest_i^{k+1} = pbest_i^k, i \notin I$$

$$gbest^{k+1} = pbest_{i_g}^{k+1}$$

とおく．ただし， $i_g = \arg \min_i f(pbest_i^{k+1})$

Step 4 : 終了判定

$k = k + 1$ として Step 2 へ戻る．もし $k = T_{max}$ であるならば，最適解を $gbest^{k+1}$ ，最適値を $f(gbest^{k+1})$ として終了する．

5 PSO の制御系設計への適用

PSO は制御モデルの線形性，評価関数の微分可能性も要求しないので，どのような対象にも適用でき，非線形な対象についても問題なく適用できる．本章では制御系のパラメータ設定に PSO を適用する観点から，PSO に修正を加え，従来よりも効率的なパラメータチューニングの実現を目指す．

本研究では，状態フィードバックで制御する際のフィードバックゲインを *particle* の位置として考えて PSO を適用する．また，PSO を制御系に適用するために必要な，評価関数の設定を制御の観点から行い，その手法を以下に示す．

5.1 評価関数 J の設定

良い制御応答とは，安定であり，過渡応答が良いことを指す．これを評価するための基準として，評価関数 J を導入して，シミュレーションと PSO を連動させて制御性能を評価し，最適化する．評価関数 J は式 (13) で与えられ，図 3 に示すようなオーバーシュート量 (Z_{OS})，誤差面積 (Z_{error})，整定時間 (Z_{Ts}) を用いて設定している [5]．ここで，誤差面積は制御偏差の絶対値の積分であり，整定時間は目標値からの誤差 $\pm 2\%$ として設定した．また， a_1, a_2, a_3 は各項目に対する重み係数であり，この J を最小とする一般化制御対象の重み W_x, W_e, W_u を求めていくことになる．

$$J = a_1 \cdot Z_{OS} + a_2 \cdot Z_{error} + a_3 \cdot Z_{Ts} \quad (13)$$

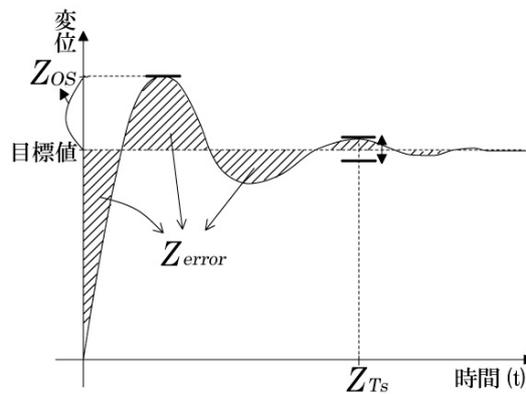


Fig. 3 評価関数の設定

5.2 評価関数 J の修正：操作量の抑制

制御での問題点として，操作量が過大になってしまい，非現実的な例がしばしば見受けられる．この問題を防止する為に，操作量の変動 (Z_{input}) とピーク値 (Z_{peak}) の指標を加えた式 (14) を評価関数 J に加え，操作量の過大を抑制する．ピーク値を入れる事で非現実的なシミュレーション結果を排除でき，操作量変動を入れる事でより少ない操作量での制御が実現できる．

$$J_1 = a_4 \cdot Z_{input} + Z_{peak} \quad (14)$$

$$Z_{peak} = \begin{cases} \infty & \text{if } u_{peak} \geq u_{lim} \\ 0 & \text{otherwise} \end{cases}$$

ここで，操作量の変動 Z_{input} とは平衡状態からの，操作量偏差の絶対値の積分である．また， a_4 は重み係数， u_{peak} は操作量の変動ピーク， u_{lim} は操作量変化の限界である．

6 設計結果

本研究で設計した制御器の有用性を示すために，次の 3 つの場合に分けてシミュレーションおよび実験を行った．

- Case1：設計点ではない $l = (l_1 + l_2)/2$ における実験．
- Case2： l_2 から l_1 まで巻き上げながらの実験．
- Case3： l_1 から l_2 まで巻き下げながらの実験．

目標値は $0.3[m]$ である．シミュレーション結果を Fig.4 に，実験結果を Fig.5 に示す．いずれの実験結果も立ち上がり方や収束時間が同程度であることから，パラメータに変動があっても安定性や制御性を損なわない制御器を設計できたといえる．

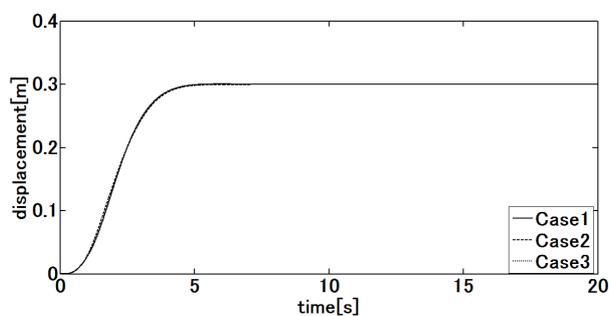


Fig. 4 Simulation Results

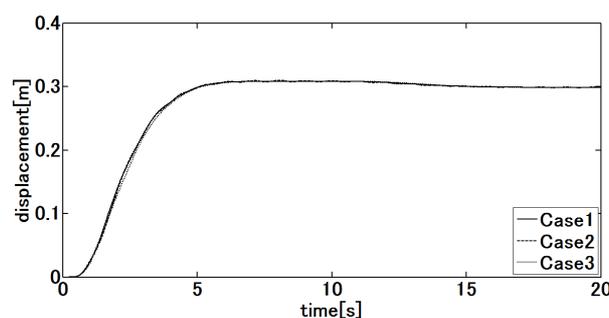


Fig. 5 Experimental Results

7 おわりに

ダイナミクスがロープ長に対して非線形に依存するジブシステムに対して、パラメータ変換と等価変換を用いることで、ポリトープ型GS制御器を設計できることを示した。また、制御の観点から評価関数に修正を加えることで、実用的な解を得られるようになった。そして、実験を通して安定性や制御性がロープ長に影響されない制御系を設計できたことを確認した。

謝辞

本研究は2010年度南山大学パツへ研究奨励金 I-A-2 の助成を受けた。

参考文献

- [1] D. J. Leith, W. E. Leithead: Survey of gain-scheduling analysis and design, *International Journal of Control*, **73**-11, 1001/1025 (2000)
- [2] P. Apkarian, P. Gahinet, G. Becker: Self-scheduled H^∞ control of linear parameter-varying systems: A design example, *Automatica*, **31**, 1251/1261 (1995)
- [3] 青木, 高見, 大石: 2乗和多項式に基づくクレーンのゲインスケジュールド制御, *計測自動制御学会論文集*, **45**-4, 208/214 (2009)
- [4] J. Kennedy, R. Eberhart: Particle Swarm Optimization, *IEEE Int. Conf. on Neural Networks*, Vol. 4, pp1942-1948, 1995
- [5] 安藤貴彦, 高見勲, 制御知識を取り入れた Particle Swarm Optimization による PID 制御系設計, *高速信号処理応用技術学会誌*, Vol.11/No.2, 59-64, 2008
- [6] 安田恵一郎, 石亀篤司, 非線形計画アルゴリズム-実用的観点から-, *システム/制御/情報* 第50巻第9号, システム制御情報学会, 2006