H_∞ 制御を用いた三慣性システムのロバスト安定化

2005MM028 木村知靖

指導教員：高見勲

1　はじめに

制御系を設計する際，対象が複雑すぎて正確にモデリングできない場合に生じる近似誤差，複雑な対象をモデリングするために行う簡単化の影響，また，使用方法・使用環境・時間経過によって起こる劣化という対象そのものの変化など，制御系のロバスト性を考慮することが大変重要である．本研究では，H_∞ 制御を用いて，ロバスト性・目標値追従性の両方を満足する制御系を設計する．また，実機を用いて理論の有効性を確認する．

2　モデリングと誤差の決定

2.1　モデリング

図 1 中で，$T(t)$ はトルク，$J_n(n=1,2,3)$ は慣性モーメント，$k_n(n=1,2)$ はばね定数，$c_n(n=1,2,3)$ はダンピング係数である．

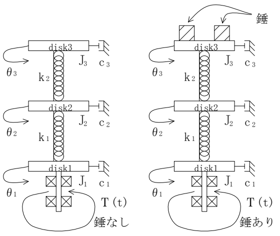

図 1　三慣性システム

このシステムを状態空間表現 $\dot{x}=Ax+Bu, y=Cx$ で表すと A，B，C は以下のようになる．ここで，$x=[\theta_1\ \dot{\theta}_1\ \theta_2\ \dot{\theta}_2\ \theta_3\ \dot{\theta}_3]^T$，$\theta_i$ はディスク i の角度 $(i=1,2,3)$．操作量は $T(t)$，出力はディスク 3 の角度である．

$$A=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{k_1}{J_1} & -\frac{c_1}{J_1} & \frac{k_1}{J_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{k_1}{J_2} & 0 & -\frac{k_1+k_2}{J_2} & -\frac{c_2}{J_2} & \frac{k_2}{J_3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{k_2}{J_2} & 0 & -\frac{k_2}{J_3} & -\frac{c_3}{J_3} \end{bmatrix} \tag{1}$$

$$B=\begin{bmatrix} 0 & \frac{1}{J_1} & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{2}$$

$$C=[\ 0\quad 0\quad 0\quad 0\quad 1\quad 0\ ] \tag{3}$$

2.2　モデル誤差の決定

本研究では，錘なしの状態から，最上部のディスク（ディスク 3）に中心から 0.09[m] の距離に 0.5[kg] の錘を 2 つ置いた状態に変更し，目標値まで追従させる．

2.2.1　錘なしの場合

錘なしの場合，ディスク 3 の慣性モーメントは $J_3=1.8\times10^{-3}$[kgm^2] である．このとき，伝達関数 $P(s)$ は下式となる．

$$P(s)=\frac{b_{11}}{a_{11}s^6+a_{12}s^5+a_{13}s^4+a_{14}s^3+a_{15}s^2+a_{16}s} \tag{4}$$

$a_{11}=1, a_{12}=3.192, a_{13}=5471, a_{14}=1.593\times10^4,$

$a_{15}=5.292\times10^6, a_{16}=7.799\times10^6, b_{11}=5.198\times10^8$

2.2.2　錘ありの場合

錘を置くと，ディスク 3 の慣性モーメントは $J_3=9.9\times10^{-3}$[kgm^2] である．このとき，伝達関数 $\hat{P}(s)$ で表すと下式となる．

$$\hat{P}(s)=\frac{b_{21}}{a_{21}s^6+a_{22}s^5+a_{23}s^4+a_{24}s^3+a_{25}s^2+a_{26}s} \tag{5}$$

$a_{21}=1, a_{22}=3.457, a_{23}=4289, a_{24}=1.013\times10^4,$

$a_{25}=2.238\times10^6, a_{26}=1.418\times10^6, b_{21}=9.455\times10^7$

3　制御系設計

3.1　重みの決定

本研究では，錘なしの伝達関数 $P(s)$ をノミナルプラントとおき，伝達関数 $\hat{P}(s)$ との誤差を考える．$P(s)$，$\hat{P}(s)$ の誤差を Δ_m とおく．乗法的誤差を用いて，Δ_m は

$$\begin{aligned}\hat{P}(s) &= (I+\Delta_m)P(s) \\ \Delta_m &= \frac{\hat{P}(s)}{P(s)}-1\end{aligned} \tag{6}$$

となる．

求めた誤差 Δ_m を考慮して安定なコントローラを設計するため，Δ_m を覆うように W_t を決定する．試行錯誤の結果，$W_t(s)=\frac{(s+0.0005)}{0.3s+1}$ と決めた．誤差 Δ_m と同時に表示したものを図 2 に示す．

次に感度関数 S に対する重み W_s を決定する．W_s は，目標値追従特性が強く要求される低周波領域では大きくし，高周波領域では小さくするため，$W_s(s)=\frac{250(0.15s+1)}{1500s+1}$ と決めた．誤差 Δ_m，重み W_t と同時に表示したものを図 3 に示す．

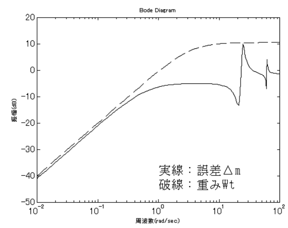

図 2　Δ_m,W_t

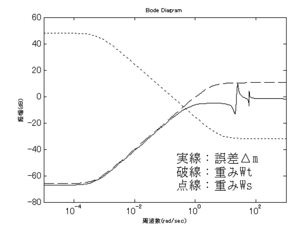

図 3　Δ_m,W_t,W_s

3.2　コントローラの導出

線形行列不等式 (LinearMatrixInequality : LMI) に基づく手法により，コントローラを求めた．

4　シミュレーション

制御系設計で求めたコントローラを用いてシミュレーションをおこなった．目標値は 2000[count](= 45°) とした．結果を図 4 に示し，そのときの操作量を図 5 に示す．

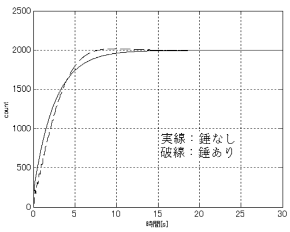

図 4 シミュレーション結果

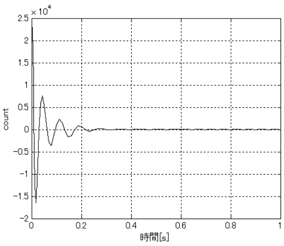

図 5 操作量

5 実験

5.1 アプローチ

コントローラを用いてプログラムを作成し実験を行った．その際，以下のことを改善しながら進めていった．

1. プログラム中の離散時間系の状態空間表現の式を考え直す．
2. コントローラからの操作量が大きく変動するため，実機がすぐに停止してしまう．そこで，操作量を制限するために制御入力に重みを加える．
3. 定常ゲインを大きくするために重みを調整する．

5.2 解決策

操作量を抑えたことにより実機が停止することは無くなった．しかし，どうしても定常ゲインを大きくすることができない．そこで，出力フィードバック制御から，状態フィードバック制御へ変更することにした．その際，以下のことを含めた一般化制御対象 G を定義する．

1. 出力の部分に重み W_t を，制御入力に重み W_n を加える．
2. 実験を進める中で摩擦が影響しているということが分かったので，偏差の部分に積分器 W_e を加える．このとき，積分器 W_e は下式である．

$$W_e = \begin{bmatrix} A_e & B_e \\ C_e & D_e \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{7}$$

5.3 状態フィードバック制御

プラント W_p，積分器 W_e，重み W_t に対する状態方程式は下式となる．

$$\begin{array}{rcl} \dot{x} & = & Ax + B_1 w + B_2 u \\ z & = & C_1 x + D_{11} w + D_{12} u \\ e & = & C_2 x + D_{21} w + D_{22} u \end{array} \tag{8}$$

また，

$$W_p = \begin{bmatrix} A_p & B_p \\ C_p & D_p \end{bmatrix},\ W_t = \begin{bmatrix} A_t & B_t \\ C_t & D_t \end{bmatrix} \tag{9}$$

である．このとき，

$$A = \begin{bmatrix} A_p & 0 & 0 \\ -C_p & 0 & 0 \\ C_p B_t & 0 & A_t \end{bmatrix},\ B_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix},\ B_1 = \begin{bmatrix} B_p \\ 0 \\ 0 \end{bmatrix} \tag{10}$$

$$C_1 = \begin{bmatrix} 0 & 1 & 0 \\ C_p D_t & 0 & Ct \end{bmatrix},\ C_2 = \begin{bmatrix} -c_p & 0 & 0 \end{bmatrix} \tag{11}$$

$$D_{11} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, D_{12} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, D_{21} = \begin{bmatrix} 0 \end{bmatrix}, D_{22} = \begin{bmatrix} 0 \end{bmatrix} \tag{12}$$

となる．よって一般化制御対象 G は,

$$G = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right] = \left[\begin{array}{ccc|cc} A_p & 0 & 0 & 0 & B_p \\ -C_p & 0 & 0 & 1 & 0 \\ C_p B_t & 0 & A_t & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 \\ C_p D_t & 0 & Ct & 0 & 0 \\ -c_p & 0 & 0 & 0 & 0 \end{array}\right] \tag{13}$$

である．ここで，重み W_n を静的ゲインとして追加する．また，状態フィードバック制御なので，$C_2 = I_n$，$D_{21} = 0$ とおく．よって，重み W_n を含めた一般化制御対象 G は以下のようになる．

$$G = \left[\begin{array}{ccc|cc} A_p & 0 & 0 & 0 & B_p \\ -C_p & A_e & 0 & 1 & 0 \\ C_p B_t & 0 & A_t & 0 & 0 \\ \hline 0 & C_e & 0 & 0 & 0 \\ C_p D_t & 0 & Ct & 0 & 0 \\ 0 & 0 & 0 & 0 & W_n \\ I_n & 0 & 0 & 0 & 0 \end{array}\right] \tag{14}$$

この一般化制御対象 G を用いてフィードバックゲイン F を求めた．

5.4 実験

フィードバックゲインを用いてプログラムを作成し，実験を行った．錘なしの場合の実験結果とシミュレーションを同時に表示したものを図 6 に示す．錘ありの場合を図 7 に示す．

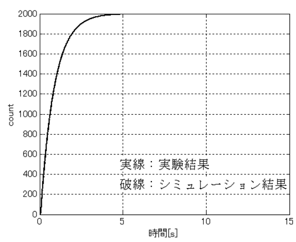

図 6 実験結果：錘なし

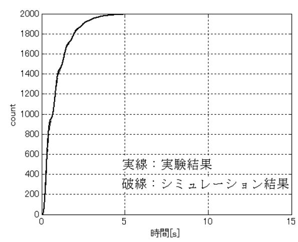

図 7 実験結果：錘あり

実験結果より，錘なしの場合も，錘ありの場合もシミュレーション結果と一致していることがわかる．

6 本研究の成果

三慣性システムのディスク 3 に錘を置かない場合と，置いた場合の両方の状態に対して，偏差を抑えるための積分器 W_e，ロバスト性を考慮するための重み W_t，操作量を制限するための重み W_n を含めた一般化制御対象を定義してコントローラを設計し，実験を行うことにより理論の有効性を確認することができた．

参考文献

[1] 野波健蔵, 西村秀和, 平田光男:”MATLAB による制御系設計”, 東京電気大学出版局,(1998).