

# 液晶パネル搬送ロボットの制御系設計

長岡技術科学大学 B4 平田研究室 佐藤 祐介

平成 19 年 10 月 4 日

## 1 はじめに

コスト削減や効率化の為に液晶のガラス基盤のサイズは年々増加傾向にあり、2011 年にはガラス基盤の一边が 4m を超えるのではないかと予測が出るほど大きくなってきている。それに伴い搬送するロボットのサイズも大型化しており、従来の制御則を用いて要求された動作精度を満たすことが困難となってきている。そこで本研究は新型の大型搬送ロボットを動作させる制御系を設計することを目的とする。

この資料は機構の関係を考えて 3 軸の片腕を 1 軸まで落とし運動方程式の導出・線形化伝達関数の導出・そして制御系設計を行うという所までと、実機に実装するために合成コントローラの構成と平衡実現、離散化を行った。また最後にサーボアンプの付属ツールより実機のプラントを推測するところまでをあわせて報告する。

## 2 運動方程式の導出

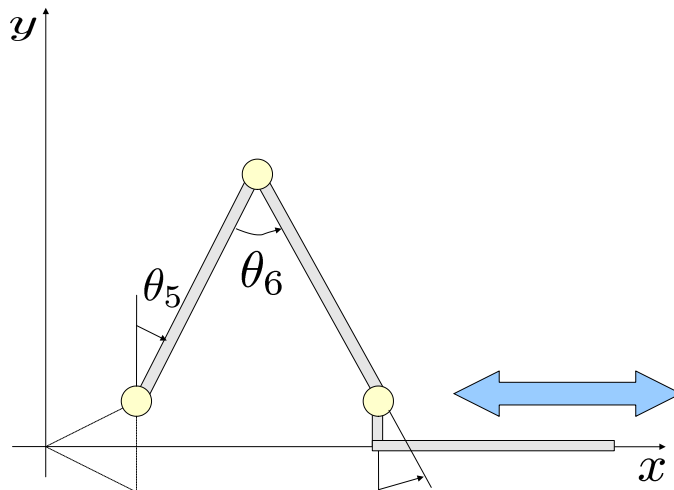


Fig. 1: ロボットのモデル

Fig.1 に示すような片腕のモデルはハンドが  $x$  軸方向に平行に動作することから以下のような機構拘束があると考えられる。

$$\theta_6 = 2\theta_5, \theta_7 = \theta_5 \quad (1)$$

以上のような機構拘束を考慮した上で、ラグランジュ法を用いてモデルの運動方程式を導出する。導出結果は以下のようになる。

$$\tau = \mathbf{M}(\theta_5)\ddot{\theta}_5 + \mathbf{h}(\theta_5, \dot{\theta}_5) \quad (2)$$

この時

$$\mathbf{M}(\theta_5) = (2 Mc1 L^2 + 2 Mb1 LLb1 + 2 Md1 L^2) \cos(2\theta_5) + Lb1^2 Mb1 + Ma1 La1^2 + L^2 Mb1 + Ia1 + Ib1 \quad (3)$$

$$+ 2 Md1 L^2 + 2 Mc1 L^2 \quad (4)$$

$$\mathbf{h}(\theta_5, \dot{\theta}_5) = -2 L \sin(2\theta_5) (Mc1 L + Md1 L + Mb1 Lb1) \dot{\theta}_5^2 \quad (5)$$

のようになっている。

### 3 運動方程式の線形化

#### 3.1 平衡点

この運動方程式を状態空間表現で表すと

$$\frac{d}{dt} \begin{bmatrix} \dot{\theta}_5 \\ \theta_5 \end{bmatrix} = \begin{bmatrix} -M^{-1}(\theta_5)h(\theta_5, \dot{\theta}_5) \\ \dot{\theta}_5 \end{bmatrix} + \begin{bmatrix} M^{-1}(\theta_5)\tau \\ 0 \end{bmatrix} \quad (6)$$

となる．状態  $u$  と入力  $x$  はそれぞれ

$$x = \begin{bmatrix} \dot{\theta}_5 \\ \theta_5 \end{bmatrix} \quad (7)$$

$$u = \tau \quad (8)$$

ここでの  $\tau$  はロボットの  $\theta_5$  の軸にかかるトルクのことを示している．一般的な状態空間における平衡点は右辺が 0 であるので

$$\frac{d}{dt}x = f(x_0, u_0) = 0 \quad (9)$$

が成立する．この時の  $x_0, u_0$  を求める．この結果と式 (2) より

$$\begin{bmatrix} -M^{-1}(\theta_5)h(\theta_5, \dot{\theta}_5) \\ \dot{\theta}_5 \end{bmatrix} + \begin{bmatrix} M^{-1}(\theta_5)\tau \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (10)$$

が得られる．従って

$$\begin{cases} \dot{\theta}_5 = 0 \\ -M^{-1}(\theta_5)h(\theta_5, \dot{\theta}_5) + M^{-1}(\theta_5)\tau = 0 \end{cases} \quad (11)$$

となる．

$$\tau = h(\theta_5, \dot{\theta}_5) \quad (12)$$

となる．ここで  $h(\theta_5, \dot{\theta}_5)$  を考える．この項は加速度を含んでおり， $h(\theta_5, 0) \rightarrow 0$  となることが分かる．従って

$$\tau = h(\theta_5, 0) = 0$$

となる．実際に計算した  $h$  の値と比較してみると

$$h = -2L \sin(2\theta_5)(Mc1L + Md1L + Mb1Lb1)\dot{\theta}_5^2 \quad (13)$$

となっており  $h(\theta_5, 0) \rightarrow 0$  となることが確認できる．従って，以下のような平衡点が得られる．

$$\begin{cases} x_0 = \begin{bmatrix} \dot{\theta}_0 \\ \theta_0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\theta}_5 \end{bmatrix} \\ u_0 = \tau_0 = 0 \end{cases} \quad (14)$$

### 3.2 平衡点近傍で線形化

$x = x_0 + \delta x$ ,  $u = u_0 + \delta u$  で表されるような, 平衡点近傍での微小変動を考える. ここで Taylor の定理を適用すると以下ようになる.

$$f(x, u) = f(x_0, u_0) + D_{xf}\delta x + D_{uf}\delta u + R_r \quad (15)$$

平衡点では  $f(x_0, u_0) = 0$ , 平衡点近傍では  $R_r = 0$  であるので

$$f(x, u) \cong D_{xf}\delta x + D_{uf}\delta u \quad (16)$$

この時, ヤコビアンを考えると次のようになる.

$$D_{xf} = \begin{bmatrix} \frac{\partial}{\partial \theta} f_1 & \frac{\partial}{\partial \theta} f_1 \\ \frac{\partial}{\partial \theta} f_2 & \frac{\partial}{\partial \theta} f_2 \end{bmatrix} \quad (17)$$

$$D_{uf} = \begin{bmatrix} \frac{\partial}{\partial \tau} f_1 \\ \frac{\partial}{\partial \tau} f_2 \end{bmatrix} \quad (18)$$

各要素はそれぞれ以下のようにになる.

$$\frac{\partial}{\partial \theta_5} \dot{f}_1 = \frac{\partial}{\partial \theta_5} \dot{\theta}_5 = 0 \quad (19)$$

$$\frac{\partial}{\partial \theta_5} f_1 = \frac{\partial}{\partial \theta_5} \dot{\theta}_5 = 1 \quad (20)$$

$$\frac{\partial}{\partial \theta_5} \dot{f}_2 = \frac{\partial}{\partial \theta_5} M^{-1}(\theta_5) \mathbf{h}(\theta_5, \dot{\theta}_5) + \frac{\partial}{\partial \theta_5} M^{-1}(\theta_5) \boldsymbol{\tau} = \frac{\partial}{\partial \theta_5} M^{-1}(\theta_5) \mathbf{h}(\theta_5, \dot{\theta}_5) \quad (21)$$

$$\frac{\partial}{\partial \theta_5} f_2 = \frac{\partial}{\partial \theta_5} M^{-1}(\theta_5) \mathbf{h}(\theta_5, \dot{\theta}_5) + \frac{\partial}{\partial \theta_5} M^{-1}(\theta_5) \boldsymbol{\tau} \quad (22)$$

$$\frac{\partial}{\partial \tau} \dot{f}_1 = \frac{\partial}{\partial \tau} \dot{\theta}_5 = 0 \quad (23)$$

$$\frac{\partial}{\partial \tau} f_2 = \frac{\partial}{\partial \tau} M^{-1}(\theta_5) \mathbf{h}(\theta_5, \dot{\theta}_5) + \frac{\partial}{\partial \tau} M^{-1}(\theta_5) \boldsymbol{\tau} = M^{-1}(\theta_5) \quad (24)$$

従って

$$D_{xf} = \begin{bmatrix} 0 & 1 \\ \frac{\partial}{\partial \theta_5} M^{-1}(\theta_5) \mathbf{h}(\theta_5, \dot{\theta}_5) & \frac{\partial}{\partial \theta_5} M^{-1}(\theta_5) \mathbf{h}(\theta_5, \dot{\theta}_5) + \frac{\partial}{\partial \theta_5} M^{-1}(\theta_5) \boldsymbol{\tau} \end{bmatrix} \quad (25)$$

$$D_{uf} = \begin{bmatrix} 0 \\ M^{-1}(\theta_5) \end{bmatrix} \quad (26)$$

これより, 平衡点において  $\mathbf{h}(\theta, 0) = 0, \tau_0 = 0$  であることから

$$D_{xf} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad (27)$$

$$D_{uf} = \begin{bmatrix} 0 \\ M^{-1}(\theta_5) \end{bmatrix} \quad (28)$$

従って

$$\frac{d}{dt} \mathbf{x}(t) = f(x, u) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ M^{-1}(\theta_5) \end{bmatrix} u \quad (29)$$

$$\mathbf{y}(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{x}(t) \quad (30)$$

## 4 伝達関数の導出

### 4.1 伝達関数の求め方

一般的な状態空間表現は

$$\begin{cases} \frac{d}{dt} \mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \end{cases} \quad (31)$$

と表すことが出来る．これらの式をそれぞれラプラス変換すると

$$\begin{cases} s\mathbf{X}(s) - \mathbf{x}_0 = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \\ \mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s) \end{cases} \quad (32)$$

となる．ただし，初期値は

$$\mathbf{x}_0 = \lim_{t \rightarrow 0} \mathbf{x}(t)$$

とする．ラプラス変換された式を変形して  $\mathbf{X}(s)$  について解き， $\mathbf{Y}(s)$  の式に代入すると

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{x}_0 + (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B}\mathbf{U}(s) \quad (33)$$

$$\mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1} \mathbf{x}_0 + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B}\mathbf{U}(s) \quad (34)$$

となる．ここで初期値  $\mathbf{x}_0$  を 0 とし，入力  $\mathbf{U}(s)$  に対しての出力  $\mathbf{Y}(s)$  を示す伝達関数は以下のように定義できる

$$\mathbf{G}(s) = \frac{\mathbf{Y}(s)}{\mathbf{U}(s)} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \quad (35)$$

### 4.2 片腕のモデルの伝達関数の導出

前述の定義に線形化された運動方程式を適用すると

$$\mathbf{G}(s) = \begin{bmatrix} 1 & 0 \end{bmatrix} \left( \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ \mathbf{M}^{-1}(\theta_5) \end{bmatrix} \quad (36)$$

$$= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1/s & 1/s^2 \\ 0 & 1/s \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{M}^{-1}(\theta_5) \end{bmatrix} \quad (37)$$

$$= \begin{bmatrix} 1/s & 1/s^2 \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{M}^{-1}(\theta_5) \end{bmatrix} \quad (38)$$

$$= \frac{\mathbf{M}^{-1}(\theta_5)}{s^2} \quad (39)$$

となる．ここで  $\mathbf{M}^{-1}(\theta_5)$  は実際にパラメータを代入すると

$$\mathbf{M}^{-1}(\theta_5) = \{ (2 M c 1 L^2 + 2 M b 1 L L b 1 + 2 M d 1 L^2) \cos(2 \theta_5) + L b 1^2 M b 1 + M a 1 L a 1^2 + L^2 M b 1 + I a 1 \} \quad (40)$$

$$+ I b 1 + 2 M d 1 L^2 + 2 M c 1 L^2 \}^{-1} \quad (41)$$

$$\doteq \frac{2.5}{2.65 \times 10^2 \cos(2\theta_5) + 9.87 \times 10^2} \quad (42)$$

となる．実際には値が細くなるが，ここでは3桁までを表示している．これらのことより，伝達関数が求められた．この伝達関数は  $\cos$  の項を含んでおり，ロボットの姿勢（ここでは  $\theta_5$ ）によってゲインが変動することがわかる．実際に  $\theta_5$  をロボットの動作範囲内 (-51.4[deg] ~ 84.8[deg]) 内で変化させてゲインの最大値と最小値を示すと以下のようになる．

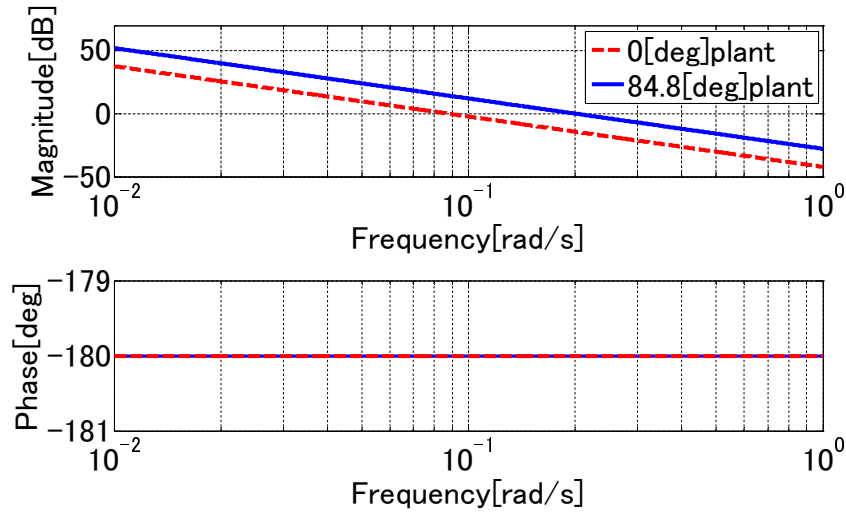


Fig. 2: プラントのボード線図

Fig.2 からわかるように対象となるロボットのプラントは動作範囲内でゲインが上下することが分かった．次にこのゲインが変動する事を考慮したうえで制御系設計を行う．

## 5 計算で求めたプラントに対する二自由度制御系設計

### 5.1 フィードバックコントローラ的设计

二自由度制御系を設計するにあたり，まずはフィードバックコントローラ的设计を行う．コントローラは適切な位相余裕確保と定常偏差を無くす等の目的から PI+位相進み補償を採用した．なぜ PI+位相進み補償を選択したかを述べる．まず位相進み補償に関して，対象となるプラントは角度で変化するものの  $s$  の 2 乗分のゲイン倍といった形となっているので，位相は常に  $-180[\text{deg}]$  である．したがって適切な位相余裕を確保するためには位相を  $-180[\text{deg}]$  から進ませる補償器が必要である．そこで位相進み補償を今回は用いた．次にプラントに積分器がかかっており，シミュレーションの段階では PI 補償を追加する必要は無いと思われるが，運動方程式の段階で摩擦を無視して導いていたので，実機に実装した場合に摩擦の影響で定常偏差が残ってしまうと考えられる．そのため PI 補償も追加した．

設計目標は (この当時は) 非線形項の影響，実機に実装した際のモデル化誤差を考慮して設計を行った．非線形項の影響があるのか，バンド幅が低いとシミュレーションを行った際に不可解な挙動が見られた．(  $K=2000$  で設計を行なった際) そこで何度か行い以下の設計目標位相余裕が  $50[\text{deg}]$  程度以上，バンド幅  $8[\text{rad}/\text{sec}]$  程度以上を目標に  $K=5000$  としてコントローラ的设计を行った．

設計目標を達成するコントローラの各パラメータは以下のようになった．

$$C(s) = K \frac{T_{PI}s + 1}{T_{PI}s} \frac{T_{PD}s + 1}{\alpha T_{PD}s + 1} = \frac{3225s^2 + 16075s + 5000}{0.04631s^2 + 3s} \quad (43)$$

$$K = 5000, T_{PI} = 3, \alpha = 0.0718, T_{PD} = 0.215 \quad (44)$$

以上でコントローラのパラメータが決定した．(このコントローラのパラメータは運動方程式が修正される前のものであり，不安定になることは無いと思うが，開ループ特性がやや変化するかもしれない) プラントとコントローラの開ループ特性を以下に示す．

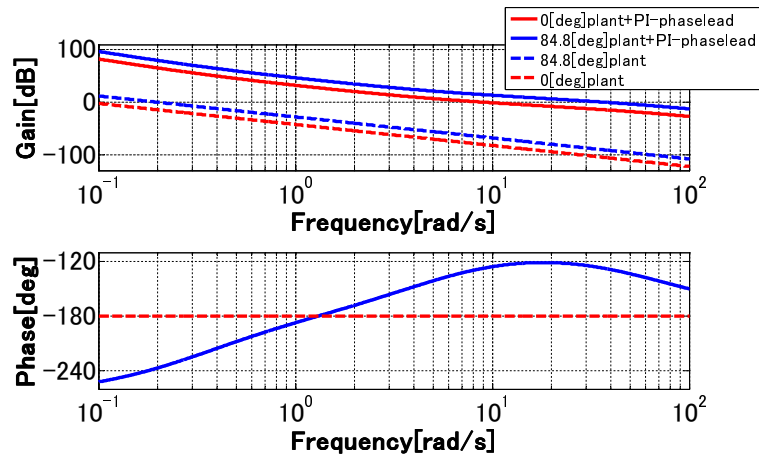


Fig. 3: プラントとコントローラのボード線図

アームの動作範囲内で適切に位相余裕が取れていることが分かる．このコントローラを二自由度制御系のフィードバックコントローラとして用いる．

## 5.2 リファレンスモデル

次にリファレンスモデルを求める．設計当初では無謀にも 1.5[s] で動作をするという，実際に見学に行った際に見た動作速度と同じリファレンスモデルを設定していた．しかし，そうすると定格トルク内に収めることが出来ないと考え，(この時点では定格トルク以上に入カトルクが入っても良い事は知らなかった)  $\omega_n$  を調整し，およそ 2[s] で動作をするリファレンスモデルを作成した．

$$T_{ref} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n + \omega_n^2} \cdot \frac{\omega_n}{s + \omega_n} = \frac{110.6}{s^3 + 13.44s^2 + 64.51s + 110.6} \quad (45)$$

$$\zeta = 0.9, \omega_n = 4.8 \quad (46)$$

## 6 二自由度制御系の構成

導出したフィードバックコントローラとリファレンスモデルを用いて，二自由度制御系を構築した．具体的な Simulink モデルは以下ようになる．

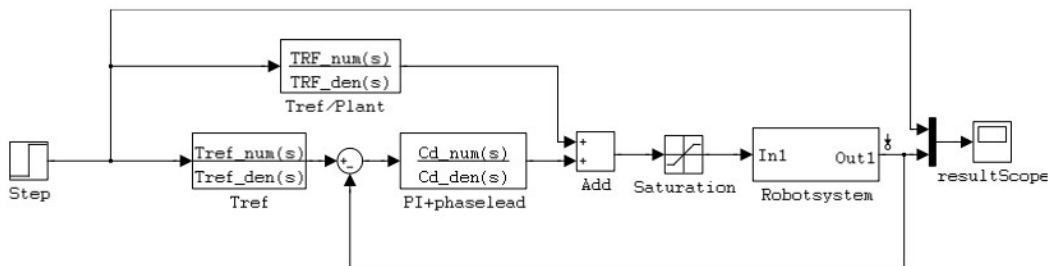


Fig. 4: 二自由度制御系の Simulink モデル

また，subsystem 内は以下のような構成である．

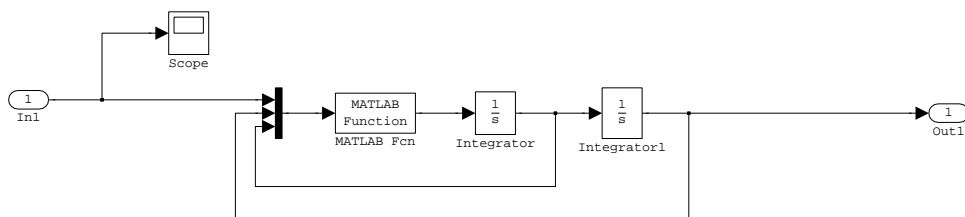


Fig. 5: 二自由度制御系の Simulink モデル (subsystem)

制御対象を非線形の運動方程式の状態空間モデルとして、シミュレーションを行う。

## 7 非線形の運動方程式に実装してシミュレーション

シミュレーション内容は原点復帰姿勢 (-51.4[deg]) を初期位置として、5[s] 後に最大伸展角度 (84.8[deg]) までのステップ入力为目标値としてシミュレーションを行った。

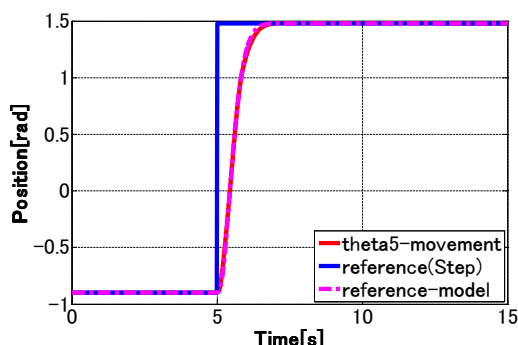


Fig. 6:  $\theta_5$  の動き

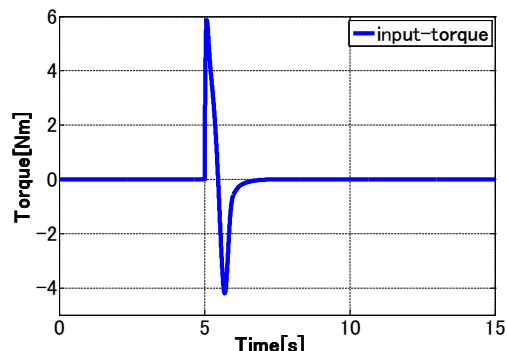


Fig. 7: モータの入カトルク

## 8 合成コントローラの作成

実機に実装するにあたり、Fig.4 のようなモデルから Fig.8 に示すように一つの状態方程式にまとめた合成コントローラを作成する必要がある。

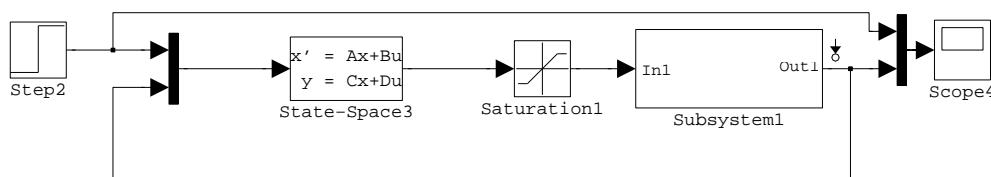


Fig. 8: 二自由度制御系の Simulink モデル (subsystem)

subsystem 内は Fig.5 と同様である。このような合成コントローラの作成は手計算でも可能であるが、手間や間違いが発生してくるので、MATLAB の `sysic` という関数を用いて計算を行う。以下に実際に `sysic` を用いた例を示して解説を行う。合成コントローラを作成する前のモデルは以下ようになる。

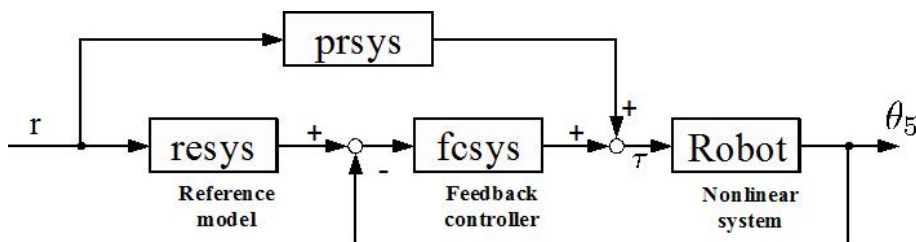


Fig. 9: 二自由度制御系

この制御系に `sysic` を用いると以下ようになる。

```
systemnames = 'fcsys resys prsys'
inputvar = '[r(1);y(1)]'
outputvar = '[fcsys+prsys]'
input_to_fcsys = '[resys-y]'
input_to_resys = '[r]'
input_to_prsys = '[r]'
```

```
sysoutname = 'csys'  
cleansysic = 'yes'  
sysic
```

行ごとに解説を行うと

- 1行目:合成しようとするシステム行列を列挙する (今回は fcsys, resys, prsys)
- 2行目:合成コントローラの入力を入力する (今回は目標値  $r$  と出力  $y$ )
- 3行目:合成コントローラの入力を入力する
- 4~6行目:各 sys の入力を順次入れる
- 7行目:合成コントローラの名前を決める

以上の行程を行い、合成コントローラを作成する。

## 9 平衡実現・離散化

平衡実現に関してはうまく説明することが出来ないかもしれないが、平衡実現を用いることによって、数値的に扱いやすい状態空間モデルが得られ、内部状態が入出力関係にどれくらい寄与しているかを見通しよく計ることが出来るようになる。従って入出力への寄与が大きい状態とそのダイナミクスだけを取り出すことによってモデルの低次元化が行える。

また、今までのシミュレーションは連続時間系であったので、実機に実装するに当たり離散化する必要がある。この2つの操作は

```
\matlab_single_arm\one_axis_cont\syc_splhold.m
```

のファイル内で行っており、平衡実現は 27~33 行目で、離散化は 36~38 行目で行っている。

## 10 実機で実装した結果

実機に今まで設計してきた二自由度制御系を実装し、動作検証を行った。結果はメールでも送られて来た通りオーバーシュートしてしまう結果となった。運動方程式より求めていたプラントの伝達関数が実機とは大きく異なっているのではないかと、ワインドアップがあったのではないかと、零点があったのではないかと等、様々な原因を考えてきたが明らかとはなっていない。

推測では、入力トルクの上がり方が急すぎたために、実際のモータはそこまでトルクを急激に出力できずにリファレンスモデルより大きく出遅れ、その後オーバーシュートをしてしまったのではないかとということと、モデル誤差が大きすぎたためにバンド幅等が足りずにオーバーシュートしてしまったということが考えられる。

## 11 サーボアンプ付属ツールによって求められたボード線図より伝達関数の推定

まずはじめに、この付属ツールによって求められたボード線図の信用性はあまり無いものであるを先に述べる。

サーボアンプから求められた、ロボットの姿勢が  $0[\text{deg}]$ ,  $-30[\text{deg}]$ ,  $60[\text{deg}]$  の場合のトルク指令 (モータ) から各速度までの伝達関数のボード線図を重ねて示す



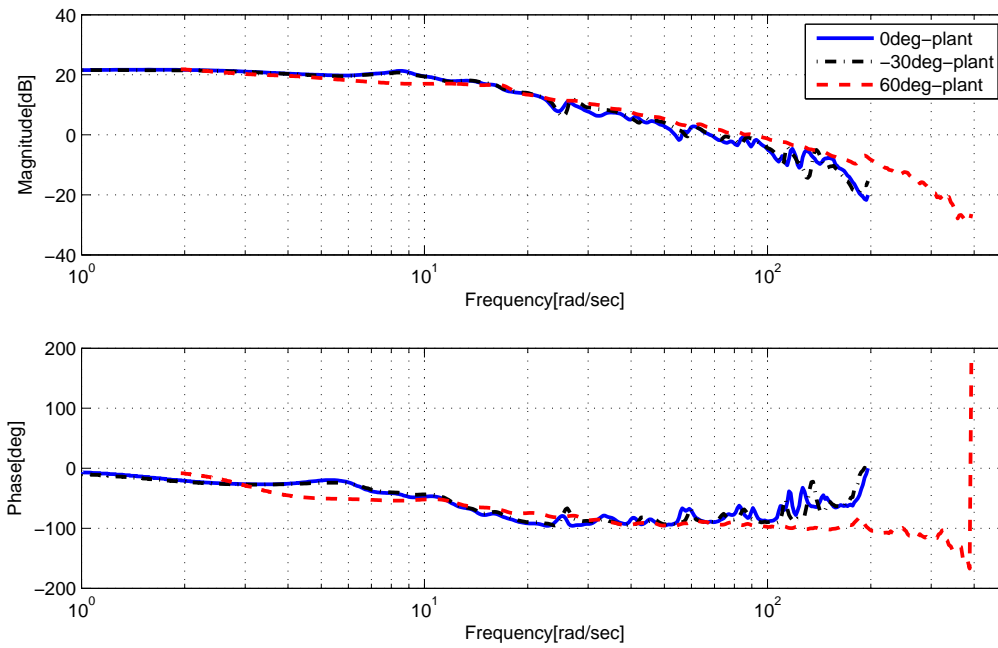
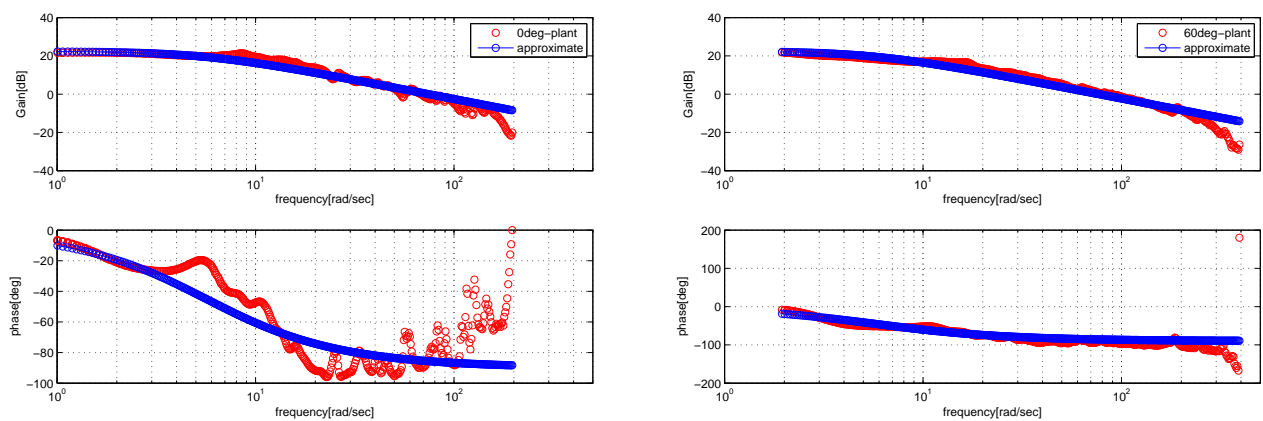


Fig. 10: プラントのボード線図

プラントは角度によってゲインがあまり変化しない事，1次遅れ系のような特性があることが分かる．次に伝達関数を1次遅れ系と仮定して近似を行う．

## 12 伝達関数の近似

前述のトルク指令に対する速度応答の伝達関数のボード線図より，1次遅れ系と似た形でプロットされることが分かった．そこで伝達関数を1次遅れ系と仮定して相関法により伝達関数の近似を行なった．以下に各角度におけるプラントのボード線図と近似したプラントのボード線図を重ねてプロットしたものを示す．



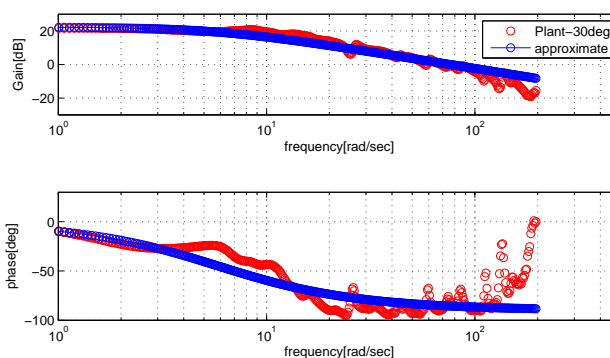


Fig. 11: 各角度におけるプラントのボード線図と近似したプラントのボード線図

近似した伝達関数は各角度ごとに以下ようになった。

$$G_{p0} = \frac{74.54}{s + 5.641} \quad (0[\text{deg}] \text{ 時}) \quad (47)$$

$$G_{p-30} = \frac{76.31}{s + 5.869} \quad (-30[\text{deg}] \text{ 時}) \quad (48)$$

$$G_{p60} = \frac{76.93}{s + 5.717} \quad (60[\text{deg}] \text{ 時}) \quad (49)$$

このプラントと運動方程式より求めた伝達関数との比較は前に行ったので省略する。この伝達関数は信頼度がどこまであるのか分からず、完全に信用して良いものではないということをもう一度述べておく。今後はステップ応答実験を行っていただけるので、伝達関数の比較、制御系の設計はそちらのデータを中心に行っていく。

### 13 今後の課題

- ステップ応答実験結果より求めたプラントとの比較
- ステップ応答実験結果より求めたプラントの制御系設計

### 参考文献

- [1] 杉江 俊治, 藤田 政之 著, “ フィードバック制御入門 ”, コロナ社, 2006
- [2] 今井 弘之, 竹口 和男, 能勢 和夫 著, “ やさしく学べる制御工学 ”, 森北出版株式会社, 2002
- [3] 増淵 正美, 川田 誠一 著, “ システムのモデリングと非線形制御 ”, コロナ社, 1996
- [4] 木下 暢, “ パンタリンクロボットの制御系設計に関する研究～実験環境の構築と2自由度制御系による実験～ ”, 大阪大学 修士論文, 2006
- [5] 藤本 健治, “ 非線形系の特異値解析と平衡実現 ”, 計測と制御