

# PID+Q コントローラによる制御系性能向上設計法

## 第1部:ロバスト制御概論\*

松尾 孝美†

2002年10月15日

### 目次

|       |                                  |    |
|-------|----------------------------------|----|
| 1     | まえがき                             | 2  |
| 2     | 状態方程式と伝達関数                       | 4  |
| 2.1   | 理論                               | 4  |
| 2.2   | 例題:電機子制御型直流サーボモータの数式モデル          | 5  |
| 3     | 信号のノルムとシステムのノルム                  | 10 |
| 3.1   | 理論                               | 10 |
| 3.2   | 例題                               | 11 |
| 3.3   | MATLABによる $H_\infty$ ノルム計算       | 12 |
| 4     | システムの不確かさ                        | 13 |
| 4.1   | 理論                               | 13 |
| 4.2   | 例題:構造的な不確かさ:状態方程式                | 14 |
| 4.3   | 例題:構造的な不確かさ:伝達関数                 | 15 |
| 4.4   | 例題:非構造的な不確かさ:寄生要素                | 15 |
| 4.5   | 例題:非構造的な不確かさ:サーボモータ              | 15 |
| 5     | ロバスト制御の背景                        | 17 |
| 5.1   | 感度関数と相補感度関数の意味                   | 17 |
| 5.2   | SIMULINKによるPIDコントローラによる閉ループ系応答計算 | 21 |
| 5.3   | SIMULINKによる感度関数と相補感度関数の計算方法      | 22 |
| 6     | $H_\infty$ 制御の標準問題の定式化           | 23 |
| 6.1   | 一般化プラントと閉ループ系                    | 23 |
| 6.2   | 一般化プラントとロバスト制御の関係                | 24 |
| 6.2.1 | 感度最小化問題                          | 24 |
| 6.2.2 | ロバスト安定性とスモールゲイン定理                | 26 |
| 6.2.3 | ロバスト安定化問題(乗法的な不確かさ)              | 26 |

\*大分大学研究シーズ発表会資料 ver.1

†大分大学工学部福祉環境工学科 Tel)097-554-7804, Fax)097-554-7507, Email)matsuo@cc.oita-u.ac.jp

|          |  |           |
|----------|--|-----------|
| 6.2.4    | ロバスト安定化問題 (加法的不確かさ)  | 27        |
| 6.2.5    | 混合感度問題   | 28        |
| 6.3      | 不確かさを含んだ一般化プラントの閉ループ系  | 28        |
| 6.4      | 線形分数変換 (LFT)   | 29        |
| 6.5      | 連鎖散乱表現と LFT 表現の関係  | 30        |
| 6.6      | 摂動プラントと ULFT 表現  | 32        |
| 6.7      | 閉ループ伝達関数と LLFT 表現  | 33        |
| 6.8      | 摂動プラントの閉ループ伝達関数と LFT 表現  | 34        |
| <b>7</b> | <b><math>H_\infty</math> 制御の標準問題と MATLAB による解法</b>                     | <b>35</b> |
| 7.1      | 問題設定と Glover-Doyle の解  | 35        |
| 7.2      | MATLAB Robust Control Toolbox を用いた $H_\infty$ コントローラの導出                | 37        |
| 7.2.1    | 関数群  | 37        |
| 7.2.2    | 例題   | 37        |
| 7.3      | MATLAB $\mu$ -Analysis and Synthesis Toolbox を用いた $H_\infty$ コントローラの導出 | 45        |
| 7.3.1    | 関数群  | 45        |
| 7.3.2    | 例題 1: Mass-Spring 系  | 47        |
| 7.3.3    | 例題 2: DC サーボモータ  | 50        |
| 7.4      | MATLAB LMI Toolbox を用いた $H_\infty$ コントローラの導出                           | 53        |
| 7.4.1    | 関数群  | 53        |
| 7.4.2    | 例題   | 53        |
| <b>8</b> | <b><math>\mu</math>解析/設計</b>   | <b>54</b> |
| <b>9</b> | <b>LMI による制御系設計</b>  | <b>54</b> |

# 1 まえがき

現代制御理論といえば、状態方程式でシステム解析から設計まですべて行うというイメージがあるが、これは1970年代までの動きであった。その中の代表的解析法が幾何学的手法であり、代表的設計法が最適レギュレータ、オブザーバ、カルマンフィルタである。ここでの中心的な設計仕様は時間的な応答をいかに望まれる波形にするかというものである。特に、出力と入力との2条積分値を最小にするようなフィードバック則が最適レギュレータにより与えられている。また、制御系のロバスト性(robustness)の面では、外乱の出力への影響をまったく除去する制御系がある条件のもとで構成されている。このような制御系設計法はいろいろな分野(航空機、自動車、ボイラ、タービン、ロボット、電力系統など)に応用され、それなりの成果をあげてきている。しかしながら、微分方程式による解析/設計は古くからのラプラス変換を中心とした古典制御理論の利点であった周波数という概念の欠落を招くことになった。つまり、古典制御では制御系設計をBode線図などで外部入力の周波数帯域ごとに視覚的にPIDコントローラの調整により行っているが(周波数整形という)、状態方程式に基づく最適レギュレータとオブザーバからなるコントローラを用いると外部入力の影響をゼロにできないような制御系(非最小位相系)の場合には、外部入力の周波数特性に応じた制御性能を調整できない。ただし、状態方程式に基づくコントローラが全く周波数特性が指定できないかということではなく、最適レギュレータによる状態フィードバックでは感度関数のゲインを1以下にでき、ゲイン余裕は、位相余裕は60°以上あることが指摘されているが、あくまでも状態がすべて観測できる場合のみに有効なものである。

このような状況の中、1970年代終りから1980年にかけてロバスト制御の基礎理論が構築され、現代制御理論と古典制御理論の融合の試みが行われるようになった。ロバスト制御とは、不確かなモデル(状態方程式や伝達関数がある誤差をもってしか実際の対象をモデル化できないことを意味する)に対して、誤差がどの程度のものであれば、どれくらいの制御性能を確保できるかという設計指針を与えるものである。ここでの誤差の定量的取扱としては、実プラントとモデル化された伝達関数との誤差の周波数ごとのゲインや最大特異値である。たとえば、実プラントはよく1次遅れ+むだ時間の形でよく表されるが、これをむだ時間を無視して1次遅れのみで近似したときの誤差のゲインは1次の有理関数のゲインにより抑えることができる。さらに設計仕様としては、外乱から制御量までの伝達関数の $H_\infty$ ノルム(単一入出力系ではピークゲイン値のこと、多入出力系では周波数伝達関数の最大特異値のピーク値)を最小にするというものである。外乱から制御量までの伝達関数は外乱や制御量を特定することにより、特別な場合には感度関数あるいは相補感度関数にすることができる(後述する)。感度関数の $H_\infty$ ノルムを小さくすると目標値への追従特性を改善することができ(ノミナル性能,nominal performance)、相補感度関数を小さくするとある大きさ以内の特性変動をもつ制御対象に対しても同じコントローラで閉ループ系の安定性を保持でき(ロバスト安定性,robust stability)、感度関数と相補感度関数を組み合わせた混合感度問題では、ある大きさ以内の特性変動をもつ制御対象に対しても目標値の追従性能をある程度保障することができる(ロバスト性能,robust performance)。このようにしてロバスト制御の概念からある種の伝達関数の $H_\infty$ ノルムを最小にするという設計仕様が生まれてきたわけで、これを $H_\infty$ 最適制御と呼ぶ。また、同じ枠組みの中で、これまでの最適制御も整理され、 $H_2$ 最適制御として一般化されている。

当初、 $H_\infty$ 最適制御の解法は関数解析手法や関数環の補間理論などを用いて行われていたため(もともと $H_\infty$ は $\infty$ ノルムをもつハーディ空間のこと)、難解であり、とでも現場で使えるまでには至らないだろうと思われていた。ところが、1980年代中ごろよりいろいろな $H_\infty$ 制御問題を、1つの標準問題として定式化したことと準最適化問題に単純化したことにより、解法が非常に単純化されることになった。標準問題は外部信号と制御信号を加えた形の4端子型システムとして表される。また、 $H_\infty$ 準最適制御(以後は簡単に $H_\infty$ 制御と呼ぶ)は外部信号から制御信号までの伝達関数の $H_\infty$ ノルムを指定された値以下にするコントローラを求める問題である。指定された値を適当に大きな値からだんだん小さくしていき、もうそれ以上小さくするとコントローラが存在しないとき、 $H_\infty$ 準最適コントローラは $H_\infty$ 最適コントローラとなる。このよ

うな定式化のもので、 $H_\infty$ 準最適コントローラは2本のRiccati型の方程式を解くことにより求められることが1988年に発表された。また、この解法は制御系設計ソフトウェアMATLABによりプログラム化され、標準問題として定式化さえすれば、コントローラを計算できるという便利なものまで現れた。しかし、この解法では、クリアすべき条件が多く、なかなか解を得るまでに、設計条件の調整が必要であった。

一方、積極的に計算機を用いて、制御系設計問題を凸計画法により解くことが1980年代より始まった。特に80年代後半には、安定化補償器のパラメトリゼーションを用いて、いろいろな設計仕様が自由パラメータに凸になることを用いた制御系設計法が提案された。しかし、この問題では探索するパラメータが伝達関数であることから、一般に無限次元の凸計画問題となり、計算効率の面で問題があった。ついで、1990年前後には、ある種の状態フィードバック制御問題が変数変換により線形行列不等式(LMI, Linear Matrix Inequality)に帰着できることが示された。これは探索パラメータが定数行列であるため、有限次元の凸計画問題になるものである。さらに、 $H_\infty$ 制御問題もLMIを用いて解かれるようになった。凸計画問題は内点法(interior-point method)などの代表的計算法が存在し、これはMATLABのLMI toolboxにより提供され、飛躍的に実系への適用を容易にして今日に至っている。

本資料では、基本的な導出は省略し、結果のみを記述し、どのような定式化のもとにどのようにしてMATLABで解くことができるのかを中心に説明する。

## 2 状態方程式と伝達関数

### 2.1 理論

システム (system) とは入力信号を出力信号に変換する作用素 (operator) である。特に動的システム (dynamical system) では、作用素は微積分作用素を含んでおり、具体的には微分方程式により記述される。微分方程式の解がシステムの内部状態であり、これを状態変数という。システムの中を流れたり、出入りする血液のようなものが信号 (signal) である。動的システムは次式のように微分方程式で表される。

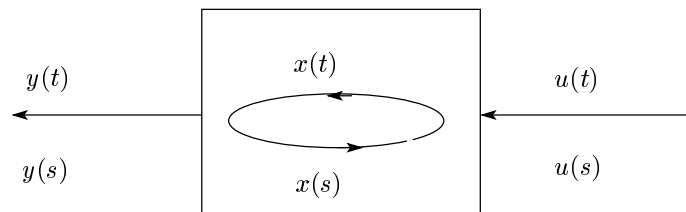


図 1: システムとは

$$\dot{x}(t) = f(x(t), u(t)) \quad (1)$$

$$y(t) = h(x, u) \quad (2)$$

ただし、 $x(t) \in R^n, u(t) \in R^r, y(t) \in R^m$  は時刻  $t$  での状態変数、入力信号、出力信号であり、 $f(x, u) \in R^n, h(x, u) \in R^m$  は  $x, u$  の非線形関数である ( $R^n$  は  $n$  次元実縦ベクトルからなる空間のこと)。これを非線形状態方程式という。

ここで、簡単のため、平衡点を  $(x, u) = (0, 0)$  とする。つまり、 $f(0, 0) = 0$  とすることに相当する。このとき、非線形関数  $f(x, u), h(x, u)$  を平衡点  $(x, u) = (0, 0)$  の周りで線形関数で近似したものは次式のように書くことができる。

$$f(x, u) = Ax + Bu$$

$$h(x, u) = Cx + Du$$

ただし、 $A, B, C, D$  は Taylor 展開から次式のように与えられる。

$$A = \frac{\partial f(x, u)}{\partial x}, B = \frac{\partial f(x, u)}{\partial u}, C = \frac{\partial h(x, u)}{\partial x}, D = \frac{\partial h(x, u)}{\partial u}$$

このとき、システムは次式の線形状態方程式で記述できる。

$$\begin{bmatrix} \dot{x}(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \quad (3)$$

これが現代制御理論の出発点となる状態方程式 (state-space equation) である。さらに、上式をラプラス変換して入力と出力の関係として書いたものが伝達関数 (transfer function) であり、次式のように与えられる。

$$\mathcal{L}\{y(t)\} = \{C(sI - A)^{-1}B + D\}\mathcal{L}\{u(t)\} \quad (4)$$

ただし,  $\mathcal{L}\{y(t)\}, \mathcal{L}\{u(t)\}$  は各々  $y(t), u(t)$  のラプラス変換であり, 複素変数  $s$  の有理関数である  $G(s) = C(sI - A)^{-1}B + D$  を伝達関数と言う. 伝達関数は Doyle 記号を用いてつぎのように書くことが多い.

$$G(s) = \left[ \begin{array}{c|c} A & B \\ \hline C & D \end{array} \right] \quad (5)$$

## 2.2 例題:電機子制御型直流サーボモータの数式モデル

サーボモータのサーボは, 英語の servant (召使) を語源とし, 入力指令に対して正しく追従して動くことを意味する. サーボモータとしては, 最もよく知られている直流 (DC) サーボモータや, DC サーボモータの欠点である機械的整流機構を改良した AC サーボモータなどがある.

DC サーボモータの大部分は他励磁式であり, 電機子制御型と界磁制御型に分かれる. 界磁式制御型はトルク・速度特性が非線形であり, 制御性に難点がある. このため, 一般には電機子制御型 DC サーボモータが用いられる.

この例では電機子制御型 DC サーボモータのみを取り扱う. 図2にモデル図を示す. 各定数は次のように与えられている.

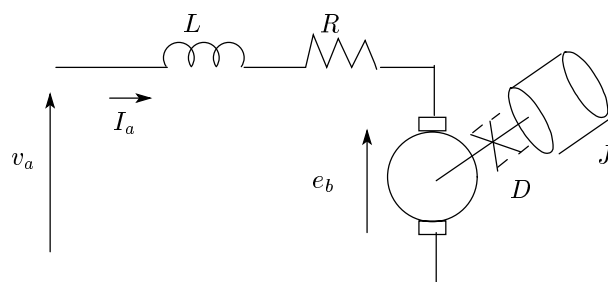


図 2: DC モータ

|             |  |
|-------------|--|
| $J$         | 負荷を含めたモータの慣性モーメント $2 \times 10^{-3} [kgm^2]$ |
| $D$         | 粘性摩擦係数 $0.01 [Nmsec/rad]$                    |
| $K_T$       | トルク定数 $0.5 [Nm/A]$                           |
| $R$         | 電機子抵抗 $1 [\Omega]$                           |
| $L$         | 電機子インダクタンス $0.01 [H]$                        |
| $K_e$       | 逆起電力係数 $0.5 [V/rad/sec]$                     |
| $I_a(t)$    | 時刻 $t$ での電機子電流 $[A]$                         |
| $v_a(t)$    | 時刻 $t$ での電機子電圧 $[V]$                         |
| $e_b(t)$    | 時刻 $t$ での逆起電力 $[V]$                          |
| $\theta(t)$ | 時刻 $t$ での回転角 $[rad]$                         |
| $\omega(t)$ | 時刻 $t$ での回転角速度 $[rad/sec]$                   |
| $T(t)$      | 時刻 $t$ でのモータの発生トルク $[Nm]$                    |

モータの発生トルクを  $T$  とおくと, つぎの運動方程式が成立する.

$$J \frac{d^2 \theta}{dt^2} + D \frac{d\theta}{dt} = T \quad (6)$$

また，トルクは電機子電流  $I_a$  に比例するので，界磁の強さおよびモータ特性による比例係数を  $K_T$  とすると

$$T = K_T I_a \quad (7)$$

が成立する．逆起電力  $e_b$  は回転角速度に比例し，逆起電力係数を  $K_e$  とすると，次式で表される．

$$e_b = K_e \frac{d\theta}{dt} \quad (8)$$

さらに，キルヒホッフの法則より

$$v_a = L \frac{dI_a}{dt} + R I_a + e_b \quad (9)$$

が成立する．ここで，状態変数  $x(t)$  を

$$x(t) = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ I_a(t) \end{bmatrix}$$

とし，入力を  $v_a(t)$ ，出力を  $\theta(t)$  とすると，次式の状態方程式が得られる．

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{D}{J} & \frac{K_T}{J} \\ 0 & -\frac{K_e}{L} & -\frac{R}{L} \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} v_a(t) \quad (10)$$

$$= \begin{bmatrix} 0 & 1 & 0 \\ 0 & -5 & -250 \\ 0 & -50 & -100 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 100 \end{bmatrix} v_a(t) \quad (11)$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x(t) \quad (12)$$

伝達関数表現を求めると，次式のようになる．

$$\Theta(s) = \frac{K_T}{(Ls + R)(Js^2 + Ds) + K_T K_e s} V_a(s) = \frac{250}{s((1 + 0.01s)(s + 5) + 125)} V_a(s) \quad (13)$$

ただし， $\Theta(s)$ ， $V_a(s)$  は  $\theta(t)$ ， $v_a(t)$  のラプラス変換である．これがモータの線形モデルである．

つぎに，この例題の状態方程式と伝達関数を MATLAB を用いて表現する方法について説明する．

- 状態方程式を Doyle 表現で定義する M ファイル (Control System Toolbox)

プログラム 1 ex2\_2\_1.m

```
% physical parameters
J = 2e-3; D = 0.01; KT = 0.5;
Ke = 0.5; R = 1; L = 0.01;
% state-space matrices
A = [ 0   1   0
      0  -D/J  KT/J
      0  -Ke/L -R/L ];
B = [ 0
      0
      1/L ];
C = [ 1  0  0 ];
```

```

D = 0;
step(A,B,C,D);%step repsonse
[num,den] = ss2tf(A,B,C,D,1)

```

- 状態方程式を Doyle 表現で定義する M ファイル (Robust Control toolbox)

プログラム2 ex2\_2\_2.m

```

% physical parameters
J = 2e-3;D = 0.01;KT = 0.5;
Ke = 0.5;R = 1;L = 0.01;
% state-space matrices
A = [ 0 1 0
      0 -D/J KT/J
      0 -Ke/L -R/L ];
B = [ 0
      0
      1/L];
C = [1 0 0];
D = 0;
% pack form (Doyle form)
ssrep = mksys(A,B,C,D,'ss');
AA = branch(ssrep,'a')

```

- 状態方程式を Doyle 表現で定義する M ファイル ( $\mu$  toolbox)

プログラム3 ex2\_2\_3.m

```

% physical parameters
J = 2e-3;D = 0.01;KT = 0.5;
Ke = 0.5;R = 1;L = 0.01;
% state-space matrices
A = [ 0 1 0
      0 -D/J KT/J
      0 -Ke/L -R/L ];
B = [ 0
      0
      1/L];
C = [1 0 0];
D = 0;
% pack form (Doyle form)
ssrep2 = pck(A,B,C,D)
minfo(ssrep2)
see(ssrep2)
% poles
spoles(ssrep2)
% zeros

```



```

szeros(ssrep2)
% frequency response
omega = logspace(-1,2,200);
freqdata = frsp(ssrep2,omega);
minfo(freqdata)
vplot('bode',freqdata)
% Tustin transform (digitalization)
T = 2*pi/20; % sampling time
dssrep2 = tustin(ssrep2,T);
dfreqdata = frsp(dssrep2,omega,T);
vplot('bode',freqdata,dfreqdata)
% generation of input signal
clf % clear graphical window
timebase = [0:0.01:5];
u = siggen('sin(5*t)',timebase);
y = trsp(ssrep2,u); % transient response
vplot(u,'b--',y,'r') % b->blue line, r->red line
title('Response of DC servo motor to input sin5t')
xlabel('time(seconds)')

```

- 状態方程式を Doyle 表現で定義する M ファイル (LMI toolbox)

プログラム 4 ex2\_2\_4.m

```

% physical parameters
J = 2e-3; D = 0.01; KT = 0.5;
Ke = 0.5; R = 1; L = 0.01;
% state-space matrices
A = [ 0  1  0
      0 -D/J KT/J
      0 -Ke/L -R/L ];
B = [ 0
      0
      1/L ];
C = [1 0 0];
D = 0;
% pack form (Doyle form)
ssrep3 = ltisys(A,B,C,D)
% pack form to (A,B,C,D)
[AA,BB,CC,DD] = ltiss(ssrep3)
% transfer function
[num,den] = ss2tf(A,B,C,D,1)
ssrep4 = ltisys('tf',num,den)
% pack form to transfer function
[numnum,denden] = ltitf(ssrep4)

```

```

% poles
poles = spol(ssrep3)
% frequency response
omega0 = 10;
freqresponse = sresp(ssrep3,omega0)
% bode diagram
wrange = logspace(-1,1,100);
splot(ssrep3,'bo',wrange)
pause
% nyquist plot
splot(ssrep3,'ny')
pause
% line-log nyquist plot
splot(ssrep3,'li')
pause
% step response
trange = 0:0.01:10;
splot(ssrep3,'st',trange)
pause
% impulse response
trange = 0:0.01:10;
splot(ssrep3,'im',trange)

```

- 物理系をそのまま積分器を用いて表現したファイル (Simulink)

プログラム 5-1 DCmotor\DCpara.m

```

% physical parameters
J = 2e-3; D = 0.01; KT = 0.5;
Ke = 0.5; R = 1; L = 0.01; K=0.5

```

ブロック線図 DCmotor\DCmotor.mdl

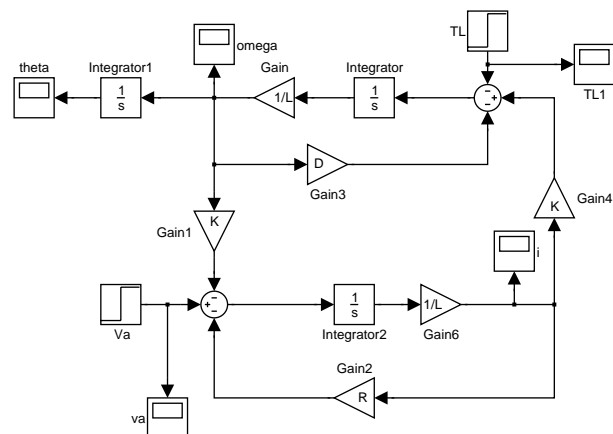


図 3: DC サーボモータ単体のブロック線図

プログラム 5-2 DCmotor\DClinmod.m

```
% physical parameters
J = 2e-3; D = 0.01; KT = 0.5;
Ke = 0.5; R = 1; L = 0.01; K=0.5
[A,B,C,D] = linmod('DCmotor_lin')
[num,den] = ss2tf(A,B,C,D,1)
```

ブロック線図 DCmotor\DCmotor\_lin.mdl

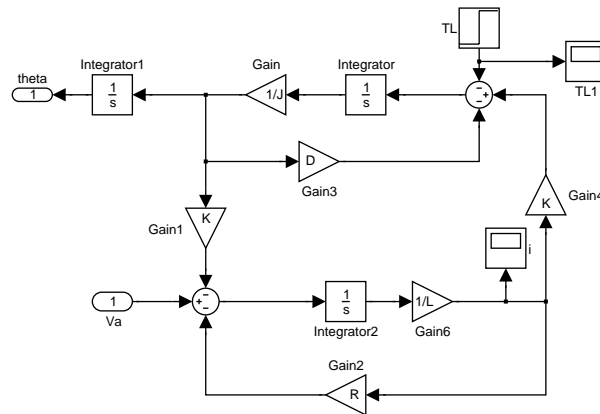


図 4: DC サーボモータ単体のブロック線図 (linmod.m 用)

### 3 信号のノルムとシステムのノルム

#### 3.1 理論

信号は  $t < 0$  のとき、すべてゼロであるもの考える (ゼロでなくてもよいが、その場合には  $L_2(-\infty, \infty)$ ,  $L_2(-\infty, 0]$ ,  $L_2[0, \infty)$ ,  $H_2$ ,  $H_2^\perp$  などが登場して複雑となるので、ここでは述べない)。信号  $u(t) \in R^n, 0 \leq t < \infty$  の大きさ (ゼロ信号からの遠さ) を表すものが信号のノルム (norm) である。特に、次式の信号のエネルギーを意味する 2 乗積分値により与えられるノルム  $\|u\|_2$  を  $L_2[0, \infty)$  ノルムといい、このノルムが有限となるすべての信号からなる空間を  $L_2[0, \infty)$  空間という。

$$\|u(t)\|_2 = \int_0^\infty u^T(t)u(t)dt \quad (14)$$

さらに、信号  $u(t)$  のラプラス変換を  $U(s)$  とすると、この信号のエネルギーを表すノルムは次式のように定義され、このノルムが有限となるすべての信号からなる空間を  $H_2$  空間という。

$$\|U(s)\|_2 = \int_{-\infty}^\infty U(j\omega)^*U(j\omega)d\omega \quad (15)$$

ただし、\*は複素共役転置をとることを意味する。Parseval の定理から時間信号のノルムとラプラス変換した周波数信号のノルムの間で次式が成り立つ。

$$\|u(t)\|_2 = \int_0^\infty u^T(t)u(t)dt = \frac{1}{2\pi} \int_{-\infty}^\infty U(j\omega)^*U(j\omega)d\omega = \frac{1}{2\pi} \|U(s)\|_2$$

したがって、時間関数の  $L_2$  ノルムと周波数関数の  $H_2$  ノルムはスケールが違うだけで、本質的には等価であることがわかる。

つぎの安定な伝達関数を考える．

$$G(s) = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \quad (16)$$

システムは有力信号を出力信号に変換する作用素であり，システムのゲイン（入出力信号の振幅増幅率）をシステムの  $H_\infty$  ノルム (norm) という． $G(s)$  がスカラー伝達関数の場合には，これはピークゲイン値になり，次式のように定義される（図5参照）．

$$\|G(s)\|_\infty = \sup_{\omega} |G(j\omega)| \quad (17)$$

ただし， $\sup$  は上限値の記号であり， $\omega$  は角周波数， $G(j\omega)$  は周波数伝達関数である． $G(s)$  が行列の場合に

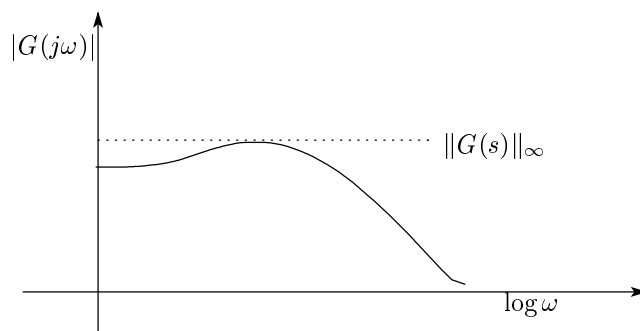


図 5: Bode 線図における  $H_\infty$  ノルム

は，複数の伝達関数要素を持つことから，これらの行列ゲインをとることにより， $H_\infty$  ノルムは，次式のように定義できる．

$$\|G(s)\|_\infty = \sup_{\omega} \sigma_{max}(G(j\omega)) \quad (18)$$

ただし， $\sigma_{max}(G(j\omega))$  は  $G(j\omega)$  の最大特異値であり，最大固有値を  $\lambda_{max}$  とすると，次式のようにしてもとめることができる．

$$\sigma_{max}(G(j\omega)) = \sqrt{\lambda_{max}(G^*(j\omega)G(j\omega))} \quad (19)$$

いま， $Y(s) = G(s)U(s)$  のとき，次の関係が成り立つ．

$$\|G(s)\|_\infty = \sup_{\omega} \frac{\|Y(s)\|_2}{\|U(s)\|_2} \quad (20)$$

これは， $H_\infty$  ノルムが入出力のエネルギー比の最大値（最悪値）を意味すると考えられる．上式を不等式で置換えると，次式のようになる．

$$\|Y(s)\|_2 \leq \|G(s)\|_\infty \|U(s)\|_2 \quad (21)$$

これは， $H_\infty$  ノルムを小さくすると，入力の影響が出力に出にくくなることを意味する．

### 3.2 例題

つぎの伝達関数を考える．

$$G(s) = \frac{as + 1}{bs + 1}, \quad a, b > 0$$

このとき,  $a \geq b$  のとき, 広域通過フィルタになり,  $a < b$  のとき, 低域通過フィルタになる. さらに,  $H_\infty$  ノルムは次式のようになる.

$$\|G(s)\|_\infty = \begin{cases} \frac{a}{b}, & (a \geq b) \\ 1, & (a < b) \end{cases}$$

### 3.3 MATLAB による $H_\infty$ ノルム計算

- $G(s) = \frac{s+1}{2s+1}$  の  $H_\infty$  ノルムを計算する M ファイル (Robust Control Toolbox)

プログラム 6 ex3\_3\_1.m

```
% G(s) = num/den
num = [1 1]; den = [2 1]
ssrep = mksys(num,den,'tf'); % packed form
hinf_norm = normhinf(ssrep)
omega = logspace(-2,2,200);
[mag,pha] = bode(num,den,omega);
subplot(2,1,1)
semilogx(omega,mag),xlabel('angular frequency[rad/sec]'),ylabel('gain')
subplot(2,1,2)
semilogx(omega,pha),xlabel('angular frequency[rad/sec]'),ylabel('phase')
```

- $G(s) = \frac{s+1}{2s+1}$  の  $H_\infty$  ノルムを計算する M ファイル ( $\mu$  Toolbox)

プログラム 7 ex3\_3\_2.m

```
% G(s) = num/den
num = [1 1]; den = [2 1]
ssrep = nd2sys(num,den) % convert transfer function to packed form
[A,B,C,D] = unpck(ssrep);
t = 0:0.01:10;
y = step(A,B,C,D,1,t); % step response
overshoot = max(y) % overshoot of step response
totalvariation = sum(abs(diff(y))) % diff = y(k+1)-y(k)
tol = 1e-5 % tolerance
hout = hinfnorm(ssrep,tol)
% hout = [lowerbound of Hinf_norm, upperbound, frequency at lowerbound]
```

- $G(s) = \frac{s+1}{2s+1}$  の  $H_\infty$  ノルムを計算する M ファイル (LMI Toolbox)

プログラム 8 ex3\_3\_3.m

```
% G(s) = num/den
num = [1 1]; den = [2 1]
ssrep = ltisys('tf',num,den)
hinf_nor = norminf(ssrep,1e-5)
```

- 多入出力系の  $H_\infty$  ノルムを計算する M ファイル (LMI Toolbox)

#### プログラム9 ex3\_3\_4.m

```
% single-input 2-outputs system
A = [ 0  1  0
      0  0  1
      -2 -1 -1 ];
B = [ 0
      0
      1];
C = [1  0  0
      0  1  0];
D = [0
      0];
ssrepmv = ltisys(A,B,C,D)
% system poles
spol(ssrepmv)
% sigma plot
wrange = logspace(-2,2,200);
splot(ssrepmv,'sv',wrange)
% hinf norm
hinf_norm = norminf(ssrepmv,1e-5)
```

## 4 システムの不確かさ

### 4.1 理論

実際の制御対象を伝達関数や状態方程式を用いてモデル化する際には、数式化する際の近似誤差やパラメータ測定の際の誤差が存在する。このため、実際の制御対象とそれをモデル化したシステムには誤差が存在することになる。さらに、モデル化したシステムが複雑であり、コントローラを設計するのに向かない場合には、このモデルからさらに簡略化した設計用モデルを作る必要がある。この設計用モデルをノミナルプラント (nominal plant) という。また、測定信号にはいろいろなノイズが外乱が混入することが多い。このような誤差をモデルの不確かさ (model uncertainties) といい、つぎのように分類できる。

- 信号の不確かさ：信号に加わる外乱やノイズ
  - 加法的ノイズ:線形システム理論の通常ノイズや外乱を意味する。1960年代から1970年代にかけて盛んであったもので、白色雑音をもつ線形状態方程式に対する推定器であるカルマンフィルタ、最適レギュレータ、およびLQGコントローラなどが有名である。
  - 乗法的ノイズ:基本的には非線形要素であり、ほとんど考慮されていないが、係数が確率的に変わる微分方程式は確率微分方程式を用いて解を求めることができる。
- システムの不確かさ：状態方程式および伝達関数の不確かさ
  - 構造的な不確かさ (structured uncertainties):伝達関数や状態方程式のパラメータの測定誤差、変動を意味する。これらのシステムに対するロバスト制御系の解法には、2次安定化法、 $\mu$ 解析/設計法、Interval System に対する Kharitonov の定理を用いたコントローラ設計などがある。

- 非構造的な不確かさ (unstructured uncertainties): 偏微分方程式で表されるシステムは有限要素法や有限次数のフーリエ級数展開などにより有限次元の状態方程式で近似した後にコントローラを設計することが多い。このような場合には、要素分割や打ちきりによる誤差が存在するが、これは近似モデルの係数変化で表すことはできず、近似モデルに付加された寄生的な動的システムとして表すことができる。
- 非線形性による不確かさ: 制御対象の非線形性 (たとえば、非線形摩擦、バックラッシュ、むだ時間、飽和などや、ダイナミクスなど) の線形近似による誤差である。

したがって、設計用モデルに基づいて設計したコントローラを実際の制御対象に適用した場合に、うまく稼動するかどうかの保障が必要である。同じコントローラで不確かさ影響をあまり受けずに制御対象を制御できるものをロバストコントローラ (robust controller) という。

不確かさを含んだプラントを摂動プラント (perturbed plant) という。特に、非構造的な不確かさを含んだ摂動プラントの伝達関数による表現方法にはつぎのようなものがある。ただし、 $\tilde{P}(s)$  を摂動プラントの伝達関数、 $P(s)$  をノミナルプラントの伝達関数とする。

- 乗法的な不確かさ (multiplicative uncertainty)

$$\tilde{P}(s) = (I + \Delta_m(s))P(s) \quad (22)$$

ただし、 $\Delta_m(s)$  を乗法的な不確かさという。不確かさの大きさは、次式のように  $H_\infty$  ノルムか、周波数伝達関数の最大特異値の上限を用いて規定されている。

$$\|\Delta_m\|_\infty < \epsilon_m \quad (23)$$

$$\bar{\sigma}(\Delta_m(j\omega)) < l_m(j\omega) \text{ for all } \omega \quad (24)$$

- 加法的な不確かさ (additive uncertainty)

$$\tilde{P}(s) = P(s) + \Delta_a(s) \quad (25)$$

ただし、 $\Delta_a(s)$  を加法的な不確かさという。不確かさの大きさは、次式のように  $H_\infty$  ノルムか、周波数伝達関数の最大特異値の上限を用いて規定されている。

$$\|\Delta_a\|_\infty < \epsilon_a \quad (26)$$

$$\bar{\sigma}(\Delta_a(j\omega)) < l_a(j\omega) \text{ for all } \omega \quad (27)$$

- 既約分解的な不確かさ (coprime factor uncertainty): これについては、既約分解の説明をしないといけないので、省略する。

ノミナルプラントは不確かさ  $\Delta_m(s)$  あるいは  $\Delta_a(s)$  は安定になるように選ばなければ、プラントをある程度近似しているとは言えないことに注意しよう。

## 4.2 例題: 構造的な不確かさ: 状態方程式

次式のような状態方程式は構造的な不確かさをもっている。

$$\dot{x}_1(t) = (1 + q_1)x_1(t) + (q_1 + q_2 + 4)x_2(t) \quad (28)$$

$$\dot{x}_2(t) = (3 + q_2)x_1(t) + (q_1 - q_2 + 3)x_2(t) + (2 + q_1)u(t) \quad (29)$$

ただし、不確かさはつぎのように規定されている。

$$|q_1| \leq 1, \quad |q_2| \leq 1$$

### 4.3 例題:構造的不確かさ：伝達関数

次式のような伝達関数は構造的な不確かさをもっている．

$$P(s, q) = \frac{0.5q_1q_2s + 10^{-5}q_1}{10^{-5} + 0.005q_2s^2 + (0.00102 + 0.5q_2)s + (2 \times 10^{-5} + 0.5q_1^2)} \quad (30)$$

ただし，不確かさはつぎのように規定されている．

$$0.2 \leq q_1 \leq 0.6, \quad 10^{-5} \leq q_2 \leq 3 \times 10^{-5}$$

### 4.4 例題:非構造的な不確かさ：寄生要素

つぎのノミナルプラント  $P(s)$  と摂動プラント  $\tilde{P}(s)$  を次式のようにおく．

$$P(s) = \frac{1}{s+1}, \quad \tilde{P}(s) = \frac{s+10}{(s+1)(s+9)}$$

このとき，不確かさはつぎのようになる．

- 乗法的な不確かさ

$$\Delta_m(s) = \tilde{P}P^{-1} - 1 = \frac{1}{s+9}$$

- 加法的な不確かさ

$$\Delta_a(s) = \tilde{P} - P = \frac{1}{(s+1)(s+9)}$$

### 4.5 例題:非構造的な不確かさ：サーボモータ

前述のサーボモータを考える．ここで，状態変数や伝達関数は3次であるので，設計を簡単にするために，通常，つぎのような近似が行われる．

インダクタンス  $L = 0.01$  は十分小さく，回路の時定数  $\tau = \frac{L}{R} = 0.01$  は無視できると仮定する．

このとき，(9) 式は次式で近似できる．

$$i_a = \frac{v_a - e_b}{R} \quad (31)$$

これより，つぎの2階常微分方程式を得られる．

$$\frac{d^2\theta}{dt^2} + \frac{1}{T_m} \frac{d\theta}{dt} = \frac{K_m}{T_m} v_a \quad \text{あるいは数値を代入して} \quad \frac{d^2\theta}{dt^2} + 130 \frac{d\theta}{dt} = 250 v_a \quad (32)$$

ただし，

$$T_m = \frac{RJ}{RD + K_T K_e} = \frac{0.002}{0.26}, \quad K_m = \frac{K_T}{RD + K_T K_e} = \frac{0.5}{0.26}$$

とする．2次に近似した設計用モデルの状態変数を

$$x_d(t) = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \end{bmatrix}$$

とし，入力を  $v_a(t)$ ，出力を  $\theta(t)$  とすると，次式の状態方程式が得られる．

$$\dot{x}_d(t) = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{T_m} \end{bmatrix} x_d(t) + \begin{bmatrix} 0 \\ \frac{K_m}{T_m} \end{bmatrix} v_a(t) \quad (33)$$

$$= \begin{bmatrix} 0 & 1 \\ 0 & -130 \end{bmatrix} x_d(t) + \begin{bmatrix} 0 \\ 250 \end{bmatrix} v_a(t) \quad (34)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x_d(t) \quad (35)$$



伝達関数は次式のようになる .

$$\Theta(s) = \frac{K_m}{s(T_m s + 1)} V_a(s) = \frac{250}{s(s + 130)} V_a(s) \quad (36)$$

このとき , 乗法的不確かさは次式のようになる .

$$\Delta_m(s) = \frac{-0.01s(s + 5)}{(1 + 0.01s)(s + 5) + 125} \quad (37)$$

乗法的不確かさのゲインは , つぎのような伝達関数のゲインを上界値として持つ .

$$l_m(s) = 10^{-6}(s + 10)(s + 100) \quad (38)$$

これらの Bode 線図を描く M ファイルはつぎのようになる .

プログラム 10 ex4\_5\_1.m

```
% physical parameters
J = 2e-3; D = 0.01; KT = 0.5;
Ke = 0.5; R = 1; L = 0.01;
% state-space matrices of perturbed plant
A = [ 0   1   0
      0 -D/J KT/J
      0 -Ke/L -R/L ];
B = [ 0
      0
      1/L ];
C = [1 0 0];
D = 0;
% transfer function of perturbed plant
[numper,denper] = ss2tf(A,B,C,D)
% state-space matrices of nominal plant
An = [0 1
      0 -130];
Bn = [0
      250];
Cn = [1 0];
Dn = 0;
% transfer function of nominal plant
[numn,denn] = ss2tf(An,Bn,Cn,Dn)
% Bode magnitude plot
wrange = logspace(-3,12,5000);
[magper,phaper] = bode(numper,denper,wrange);
[magnom,phanom] = bode(numn,denn,wrange);
semilogx(wrange,20*log10(magper), 'b', wrange,20*log10(magnom), 'r--')
grid,xlabel('angular freq[rad/sec]'),ylabel('gain[dB]')
pause
```

```

% multiplicative uncertainty delta = num_delta/den_delta
den_delta = conv([0.01 1],[1 5])+125;
num_delta = -0.01*[1 5 0];
% l_m(s) upper bound of delta
% |delta(j\omega)| < |l_m(j\omega)|
num_lm = 1e-6*conv([1 10],[1 100]);
den_lm = [1];
wrange = logspace(-3,5,1000);
[magdel,phadel] = bode(num_delta,den_delta,wrange);
[maglm,phalm] = bode(num_lm,den_lm,wrange);
semilogx(wrange,20*log10(magdel),'b',wrange,20*log10(maglm),'r--')
grid,xlabel('angular freq[rad/sec]'),ylabel('gain[dB]')

```

## 5 ロバスト制御の背景

### 5.1 感度関数と相補感度関数の意味

ここでは、 $H_\infty$ 制御が出てくるに至った背景を述べる．簡単のため1入出力系を対象とする．つぎのような単一フィードバック系を考える．ただし、 $P(s)$ はプラントのノミナル値、 $K(s)$ は設計すべきコントロ

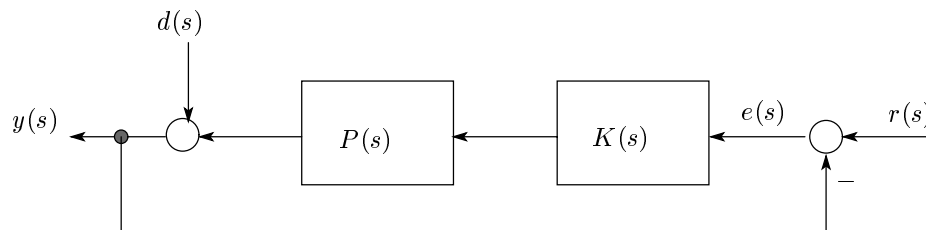


図 6: 単一フィードバック系

ーラとし、また、 $r(s)$ は目標値、 $d(s)$ は外乱、 $e(s)$ は出力と目標値との偏差である．さらに、 $P(s)$ は次式で表されるとする．

$$P(s) = \left[ \begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right] \quad (39)$$

制御系設計とは、望まれる設計仕様を満足するようなコントローラ  $K(s)$ を見つけることであるが、その設計仕様のうち非常に重要なものは、つぎのようなものである．

1. (安定条件) ノミナルプラント  $P(s)$  に対して閉ループ系は安定である (ノミナル安定性, nominal stability) .
2. (応答特性) ノミナルプラント  $P(s)$  は外乱  $d(s)$  や目標値  $r(s)$  のもとで、偏差  $e(s)$  は小さい (ノミナル性能, nominal performance) .
3. (ロバスト性) 制御対象にモデル化誤差や変動が存在しても、閉ループ系の安定性が保持され (ロバスト安定性, robust stability) , 応答特性の変化も少ない (ロバスト (制御) 性能, robust performance) .

設計仕様1)の解は,安定化補償器の Youla Parameterization として与えられ,閉ループ系を安定化するすべての線形コントローラがつぎの形で与えられることがわかっている.

$$K(s) = (Y(s) - Q(s)N(s))^{-1}(X(s) + Q(s)D(s)) \quad (40)$$

ただし,  $Q(s)$  は任意の安定プロパな伝達関数行列とし,  $(N(s), D(s))$  は次式のノミナルプラント  $P(s)$  の既約分解である.

$$P(s) = D(s)^{-1}N(s) \quad (41)$$

また,  $(X(s), Y(s))$  は Bezout 等式の解である. これらの状態空間表現の1つは次式のように表される.

$$D(s) = \left[ \begin{array}{c|c} A - FC & -F \\ \hline C & I \end{array} \right], N(s) = \left[ \begin{array}{c|c} A - FC & B \\ \hline C & 0 \end{array} \right] \quad (42)$$

$$X(s) = \left[ \begin{array}{c|c} A - FC & -F \\ \hline H & 0 \end{array} \right], Y(s) = \left[ \begin{array}{c|c} A - FC & B \\ \hline H & I \end{array} \right] \quad (43)$$

ただし,  $F, H$  は行列  $(A - FC), (A - BH)$  が安定行列となるような行列とする. 特に,  $Q(s) = 0$  とすると, コントローラは

$$K(s) = Y(s)^{-1}X(s) = \left[ \begin{array}{c|c} A - FC - BH & F \\ \hline H & 0 \end{array} \right] \quad (44)$$

となり, 状態空間表現すると, 次式のようになる.

$$u(t) = -Hz(t) \quad (45)$$

$$\dot{z}(t) = (A - FC - BH)z(t) - Fe(t) \quad (46)$$

したがって, (44) が状態フィードバック+オブザーバの形になっていることがわかる.

設計仕様2)については, 偏差  $e(s)$  が

$$e(s) = (I + P(s)K(s))^{-1}(r(s) - d(s)) \quad (47)$$

と表せる. ここで, 単一入出力系では, その周波数応答を考え,  $r(s), d(s)$  のゲインの大きくなる周波数帯域で, 伝達関数  $(1 + P(s)K(s))^{-1}$  のゲインを小さくすれば,  $e(s)$  の振幅は小さくできることがわかる. 特に, ピークゲイン  $\gamma = \sup_{\omega} |(1 + P(j\omega)K(j\omega))|$  を小さくすると, 全周波数帯域にわたって

$$|e(j\omega)| \leq \gamma|r(j\omega) - d(j\omega)| \quad (48)$$

を保障することができる. ここで,  $\gamma$  がスカラー伝達関数の  $H_{\infty}$  ノルムになっている.  $\gamma$  を小さくすることは, どのような周波数帯域をもつ入力信号が入ってきてもこれの出力への影響を減じることを意味している. また, 多変数系で伝達関数が行列になる場合の  $H_{\infty}$  ノルムは, 最大特異値を用いて, 次式のように表されることに注意しよう.

$$\|G(s)\|_{\infty} = \sup_{\omega} \sigma_{max}(G(j\omega)) \quad (49)$$

設計仕様3)はロバスト安定性とロバスト(制御)性能である. まず, ロバスト安定性について述べる. まず, プラントのモデル化誤差の定義を行う. 摂動プラントを次式のように乗法的摂動を用いて表す.

$$\tilde{P}(s) = (I + \Delta_m(s))P(s) \quad (50)$$

ただし,  $P(s)$  はノミナルプラントの伝達関数,  $\Delta_m(s)$  は乗法的不確かさの伝達関数である. 乗法的不確かさの大きさの上限値は次式のように表されると仮定する.

$$\sigma_{max}(\Delta_m(j\omega)) < |l(j\omega)|, \text{ for all } \omega > 0 \quad (51)$$

上式を満たすすべての不確かさを集合  $B_m$  で定義する.

$$B_m = \{\Delta_m(s) : \sigma_{max}(\Delta_m(j\omega)) < |l(j\omega)|, \text{ for all } \omega > 0\} \quad (52)$$

したがって, 正確なモデルが構築できない実対象をモデリングする際には, 対象の大体の動特性とそのモデル化誤差の大きさの範囲を求めるように行わなければならない. したがって, このような実対象では, ノミナルプラント  $P(s)$  の伝達関数を実プラントと見なして, 設計仕様 1), 2) を満足するようにコントローラ  $K(s)$  を決定することになるが, 本当のプラントは摂動プラント  $\tilde{P}$  であるとして,  $P(s)$  より導出した  $K(s)$  を  $\tilde{P}$  のコントローラとして用いてもなるべく不都合が起きないようにしておくように設計するのが, ロバスト制御系設計の考え方である. ロバスト制御の基本的条件はつぎの3点である.

- 低感度特性: 閉ループ系の伝達特性 (閉ループ伝達関数) がノミナルプラント  $P(s)$  と摂動プラント  $\tilde{P}(s)$  であまり変化しないことを意味し, これの指標となるのが感度関数である. ノミナルプラントに対する閉ループ伝達関数を  $T(s)$ , 摂動プラントに対する閉ループ伝達関数を  $\tilde{T}(s)$  とおくと, 次式のようなになる.

$$T(s) = (I + P(s)K(s))^{-1}P(s)K(s) = P(s)K(s)(I + P(s)K(s))^{-1} \quad (53)$$

$$\tilde{T}(s) = (I + \tilde{P}(s)K(s))^{-1}\tilde{P}(s)K(s) = \tilde{P}(s)K(s)(I + \tilde{P}(s)K(s))^{-1} \quad (54)$$

ここで,  $\delta(s)$  を次式のようにプラントが  $P(s)$  から  $\tilde{P}(s)$  に変わったときの閉ループ伝達関数の乗法的変動分として定義する.

$$\tilde{T}(s) = (I + \delta(s))T(s) \quad (55)$$

乗法的不確かさ  $\Delta(s)$  を用いると, 次式が成立する.

$$\tilde{T}(s) = (I + \Delta(s))P(s)K(s)\{I + (I + \Delta(s))P(s)K(s)\}^{-1} \approx \{I + (I + P(s)K(s))^{-1}\Delta(s)\}T(s)$$

これから

$$\delta(s) = (I + P(s)K(s))^{-1}\Delta(s) \quad (56)$$

となり,  $(I + P(s)K(s))^{-1}$  はプラントの変動分に対する閉ループ伝達関数の影響分を表しており, これを感度関数といい, 記号  $S(s)$  で書く. したがって, モデル化誤差  $\Delta(s)$  による閉ループ伝達関数があまり変化して欲しくない周波数帯域, 一般には低周波域, において, 感度関数のゲインを小さくすることが要求される. これを低感度特性という. 通常,  $d(s), r(s)$  とともに低周波信号であることから, これは設計仕様 2) と同様の条件になり, つぎのような仕様にまとめられる.

感度関数に対する設計仕様

$$\|W_s(s)S(s)\|_{\infty} < 1 \quad (57)$$

ただし,  $W_s(s)$  はプロパで零点, 極ともに安定であるような伝達関数で, かつ, それぞれの制御対象に応じて低周波域でゲインを適度に大きく取るように設定する.

- ロバスト安定性：低感度特性は閉ループ伝達関数の変動の大きさを小さくするだけであるので、ノミナルプラント  $P(s)$  を安定化するコントローラ  $K(s)$  が、同時に摂動プラント  $\tilde{P}(s)$  も安定化しているとは限らない。  $K(s)$  が  $P(s)$  とすべての変動に対する  $\tilde{P}(s)$  の集合に対しても閉ループ系が安定であるとき、ロバスト安定であるといい、  $K(s)$  をロバストコントローラという。(51)式なる誤差範囲に入っているすべての摂動プラント  $\tilde{P}(s)$  を安定化するコントローラが満たすべき必要十分条件は次式である。

相補感度関数に対する設計仕様

$$\|l_m(s)T(s)\|_\infty \leq 1 \quad (58)$$

ここで、  $T(s)$  はノミナルプラントの閉ループ伝達関数であるが、次式が成り立つことから、これを特に相補感度関数という。

$$S(s) + T(s) = I$$

- ロバスト(制御)性能：摂動プラントが安定でかつ、摂動プラントの感度関数がノミナルプラントと同様に次式の設計仕様をみたすとき、ロバスト(制御)性能を有するという。

$$\|W_s(s)\tilde{S}(s)\|_\infty < 1, \quad \text{for all } \Delta_m(s) \in \mathbf{B}_m \quad (59)$$

ただし、  $\tilde{S}(s) = (I + \tilde{P}(s)K(s))^{-1}$  とする。上式の設計仕様を周波数で記述すると、つぎのようになる。

$$\sigma_{max}[W_s(j\omega)\tilde{S}(j\omega)] < 1, \quad \text{for all } \omega \geq 0 \text{ and } \Delta_m(s) \in \mathbf{B}_m$$

ロバスト性能のための十分条件はつぎのようにして導出できる。まず、つぎのような変形が可能である。

$$\begin{aligned} W_s(s)\tilde{S}(s) &= W_s(s)(I + \tilde{P}(s)K(s))^{-1} \\ &= W_s(s)(I + (I + \Delta_m(s))P(s)K(s))^{-1} \\ &= W_s(s)(I + P(s)K(s) + \Delta_m(s)P(s)K(s))^{-1} \\ &= W_s(s)\{(I + \Delta_m(s)P(s)K(s)(I + P(s)K(s))^{-1})(I + P(s)K(s))\}^{-1} \\ &= W_s(s)(I + P(s)K(s))^{-1}(I + \Delta_m(s)(I + P(s)K(s))^{-1}P(s)K(s))^{-1} \\ &= W_s(s)S(s)(I + \Delta_m(s)T(s))^{-1} \end{aligned}$$

ここで、特異値についてつぎの不等式

$$\begin{aligned} \sigma_{max}[AC] &\leq \sigma_{max}[A]\sigma_{max}[C] \\ \sigma_{min}[A^{-1}] &= \frac{1}{\sigma_{max}[A]}, \sigma_{max}[A^{-1}] = \frac{1}{\sigma_{min}[A]} \\ \sigma_{max}[A + B] &\leq \sigma_{max}[A] + \sigma_{max}[B] \\ \sigma_{min}[A] - \sigma_{max}[B] &\leq \sigma_{min}[A + B] \leq \sigma_{min}[A] + \sigma_{max}[B] \end{aligned}$$

が成り立つことに注意すると、次式が導出できる。ただし、  $\sigma_{max}, \sigma_{min}$  は各々、最大特異値、最小特異値を表している。

$$\begin{aligned} \sigma_{max}[W_s(j\omega)\tilde{S}(j\omega)] &\leq \sigma_{max}[W_s(j\omega)S(j\omega)]\sigma_{max}[(I + \Delta_m(j\omega)T(j\omega))^{-1}] \\ &= \frac{\sigma_{max}[W_s(j\omega)S(j\omega)]}{\sigma_{min}[I + \Delta_m(j\omega)T(j\omega)]} \\ &\leq \frac{\sigma_{max}[W_s(j\omega)S(j\omega)]}{1 - \sigma_{max}[\Delta_m(j\omega)T(j\omega)]} < 1 \end{aligned}$$

このしきから，次式が導出され，この条件を満たすコントローラを求める問題を混合感度問題という．

$$\sigma_{max}[W s(j\omega)S(j\omega)] + \sigma_{max}[\Delta_m(j\omega)T(j\omega)] < 1, \quad \text{for all } \omega \geq 0 \text{ and } \Delta_m(s) \in \mathbf{B}_m \quad (60)$$

## 5.2 SIMULINK によるPIDコントローラによる閉ループ系応答計算

次式のサーボモータのノミナルプラントと摂動プラントを例にする．

$$P(s) = \frac{250}{s(s+130)}, \tilde{P}(s) = \frac{250}{s((1+0.01s)(s+5)+125)} = \frac{250}{s(0.01s^2+1.05s+130)} \quad (61)$$

次式のPIDコントローラによりフィードバック制御系を構成する．ただし，D動作はフィルタ付きの近似微分を採用している．

$$K(s) = K_P + K_I \frac{1}{s} + K_D \frac{s}{(1/N)s+1} \quad (62)$$

ここで， $K_P, K_I, K_D$ がPIDコントローラ的设计パラメータであり， $N$ は近似微分器の除数とよばれ，100程度の値がよく用いられる．

- 閉ループ系シミュレーション用ファイル(Simulink)

ブロック線図 DCmotor\DCmpid.mdl  $K_P = 50, K_I = 0.1, K_D = 0.01, N = 100$  のときの応答をシミュ

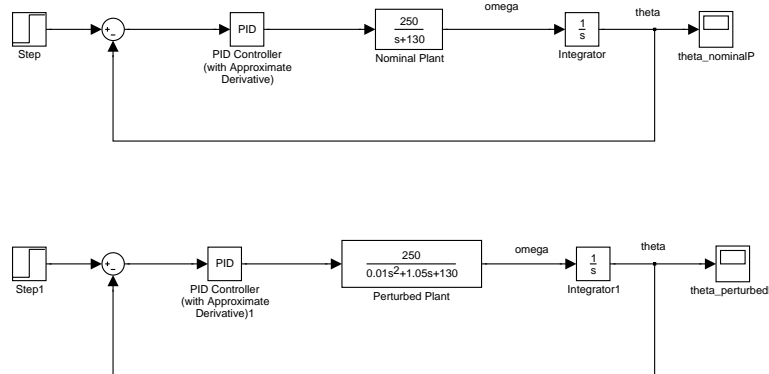


図 7: DC サーボモータのPID制御系

レーションする．出力データはScopeでグラフ化できるが，もっと精密なグラフを書くには，Scopeのプロパティのデータヒストリで”データをワークスペースに保存”をチェックし，変数名を与えて(ここでは， $\theta_p$ と $\theta_n$ としている)，フォーマットを時間付き構造体として，シミュレーションを開始する．シミュレーションが終了すると，変数がワークスペースに保存されているので，つぎのMファイルを用いてグラフ化する．

- グラフ表示Mファイル:プログラム11 DCmotor\mkgraph.m

```
% draw graph of work_spacedata
% save scope_data to workspace
% data_name theta_p, theta_n
%
plot(theta_n.time,theta_n.signals.values,'b-',theta_p.time,theta_p.signals.values,'r--')
xlabel('time[sec]'),ylabel('\theta [rad]')
```

このときのグラフはつぎのようになる．

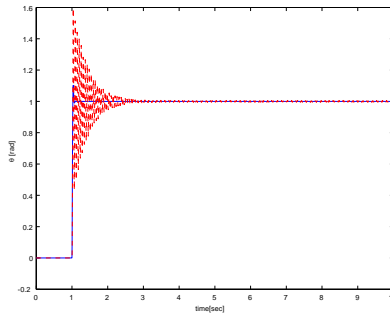


図 8: DC サーボモータのPID 制御系の角度応答 (実線: ノミナルプラント, 破線: 摂動プラント)

### 5.3 SIMULINK による感度関数と相補感度関数の計算方法

前出の DC サーボモータを例にして, PID フィードバック制御系の感度関数と相補感度関数を求め, その Bode 線図を描く.

- DC サーボモータの閉ループ系の感度関数と相補感度関数 (コントローラは PID コントローラ) を定義する Simulink ファイル  
 ブロック線図 DCmotor\DCmotorsense.mdl この図で, 入力端子 1 から出力端子 1 までの伝達関数がノ

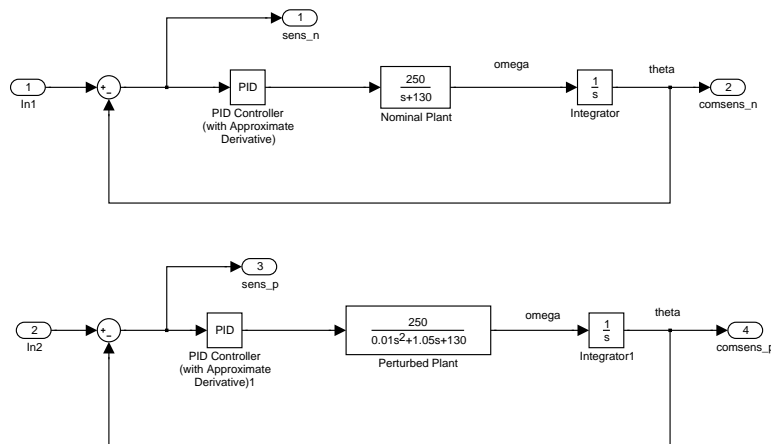


図 9: 感度関数と相補感度関数を求めるブロック線図

ミナルプラントの感度関数, 入力端子 1 から出力端子 2 までの伝達関数がノミナルプラントの相補感度関数, 入力端子 2 から出力端子 1 までの伝達関数が摂動プラントの感度関数, 入力端子 2 から出力端子 2 までの伝達関数が摂動プラントの相補感度関数である. これを linmod.m 関数を用いて, MATLAB により計算できる M ファイルをつぎに示す.

プログラム 12 DCmotor\DCm\_senscoms.m

```

D[A,B,C,D] = linmod('DCmotorsense')
Bsn = B(:,1);Bcn = B(:,1);Csn = C(1,:);Ccn = C(2,:);
Dsn = D(1,1);Dcn = D(2,1);Bsp = B(:,2);Bcp = B(:,2);
Csp = C(3,:);Ccp = C(4,:);Dsp = D(3,2);Dcp = D(4,2);
[numSn,denSn] = ss2tf(A,Bsn,Csn,Dsn)
[numTn,denTn] = ss2tf(A,Bcn,Ccn,Dcn)

```

```

[numSp,denSp] = ss2tf(A,Bsp,Csp,Dsp)
[numTp,denTp] = ss2tf(A,Bcp,Ccp,Dcp)
subplot(2,2,1),bode(numSn,denSn)
subplot(2,2,2),bode(numTn,denTn)
subplot(2,2,3),bode(numSp,denSp)
subplot(2,2,4),bode(numTp,denTp)

```

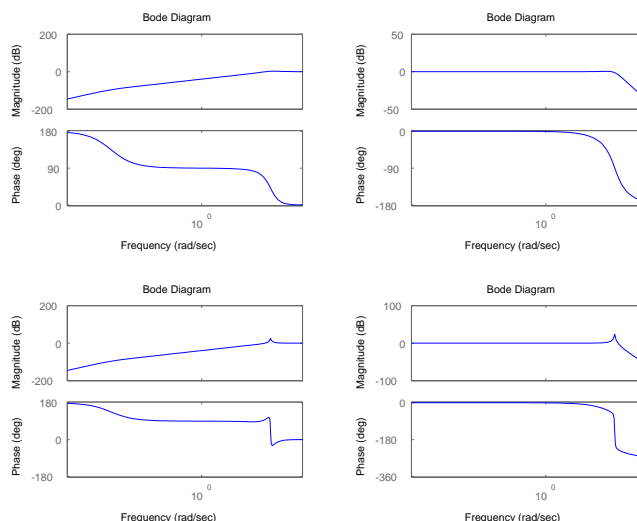


図 10: 感度関数と相補感度関数のボード線図 (左上: ノミナルプラントの感度関数, 右上: ノミナルプラントの相補感度関数, 左下: 摂動プラントの感度関数, 右下: 摂動プラントの相補感度関数)

## 6 $H_\infty$ 制御の標準問題の定式化

前述したように制御系設計の問題は, ある閉ループ伝達関数の  $H_\infty$  ノルムを指定した値以下にする問題に帰着できることがわかる. そこで, 一般的な  $H_\infty$  制御問題とロバスト制御に適用する場合の方法について述べる.

### 6.1 一般化プラントと閉ループ系

一般的ロバスト制御でのプラント表現は次図のようなものであり, これを一般化プラントあるいは標準プラントと呼ぶ. ただし, 各信号はつぎのように定義されている.

|        |                   |        |                    |
|--------|-------------------|--------|--------------------|
| $z(s)$ | 制御量 (設計者が小さくしたい量) | $w(s)$ | 外部信号 (外乱, 目標値信号など) |
| $u(s)$ | 操作入力              | $y(s)$ | 観測出力               |

ここで,  $G(s)$  は次式で与えられる伝達関数である.  $G(s)$  は必ずしもプラント自身の伝達関数のみとは限らないのが普通である (次節で例を述べる).

$$\begin{bmatrix} z(s) \\ y(s) \end{bmatrix} = G(s) \begin{bmatrix} w(s) \\ u(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} w(s) \\ u(s) \end{bmatrix} \quad (63)$$



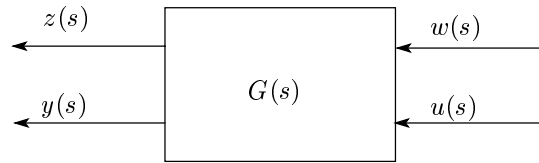


図 11: 一般化プラント

これは、次式のように状態空間表現できる．

$$\begin{bmatrix} \dot{x}(t) \\ z(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} x(t) \\ w(t) \\ u(t) \end{bmatrix} \quad (64)$$

これを Doyle 記号で書くと、つぎのようになる．

$$G(s) = \left[ \begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array} \right] \quad (65)$$

一般化プラントにコントローラ

$$u(s) = K(s)y(s) = \left[ \begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array} \right] y(s) \quad (66)$$

を入れて閉ループ系を組んだときのブロック線図は、つぎのようになる．

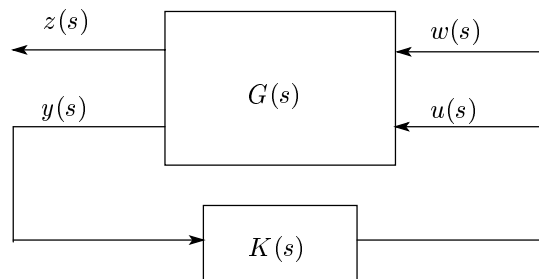


図 12: 一般化プラント

## 6.2 一般化プラントとロバスト制御の関係

前章に登場した問題を一般化プラントの問題として定式化する．

### 6.2.1 感度最小化問題

感度関数の最小化問題は、次図のような形の一般化プラントの問題に変形できる．各信号の関係は次式のようになる．

$$z(s) = W_s(s)y(s), \quad y(s) = w(s) - P(s)u(s), \quad u(s) = K(s)y(s)$$

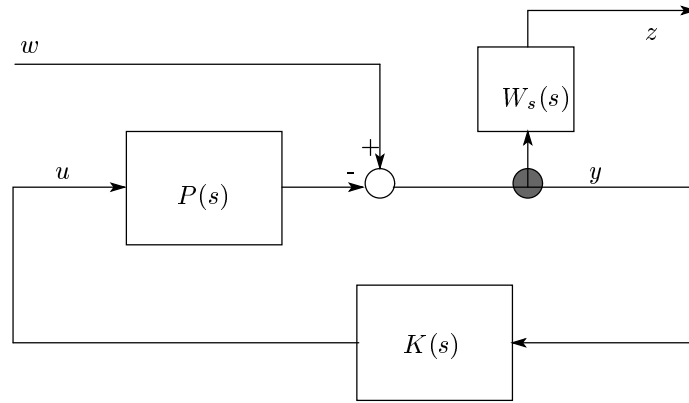


図 13: 感度最小化問題

ただし,  $W_s(s)$  は出力信号で強調される周波数領域を指定するもので, 周波数重みと呼ばれる. これから, 一般化プラントは次式のようになる.

$$\begin{bmatrix} z(s) \\ y(s) \end{bmatrix} = G(s) \begin{bmatrix} w(s) \\ u(s) \end{bmatrix} = \begin{bmatrix} W_s(s) & -W_s(s)P(s) \\ I & -P(s) \end{bmatrix} \begin{bmatrix} w(s) \\ u(s) \end{bmatrix}$$

また,  $P(s), W_s(s)$  の状態空間表現を

$$P(s) = \left[ \begin{array}{c|c} A & B \\ \hline C & D \end{array} \right], \quad W_s(s) = \left[ \begin{array}{c|c} A_W & B_W \\ \hline C_W & D_W \end{array} \right]$$

とすると,  $G(s)$  の状態空間表現はつぎのようになる.

$$G(s) = \left[ \begin{array}{cc|cc} A_W & B_W C & B_W & 0 \\ 0 & A & 0 & -B \\ \hline C_W & D_W C & D_W & 0 \\ 0 & -C & I & 0 \end{array} \right]$$

また,  $w$  から  $z$  までの伝達関数は次式のようになる.

$$z(s) = W_s(s)(I + P(s)K(s))^{-1}w(s)$$

これらの伝達関数を実際に求めるには, 前述したように Simulink でブロック線図を構成し, `linmod.m` 関数を用いるのが便利である.

感度最小化問題と準感度最小化問題はつぎのように定式化できる.

- 感度最小化問題:  $\min_{K(s)} \|W_s(s)(I + P(s)K(s))^{-1}\|_\infty$
- 準感度最小化問題:  $\|W_s(s)(I + P(s)K(s))^{-1}\|_\infty < \gamma$

ここで, 周波数重みについて,

$$\sigma_{max}[W_s(j\omega)] \leq |w_s(j\omega)| \quad (67)$$

であると仮定すると, 準感度最小化問題が解決するための十分条件は, 次式のようになる.

$$\sigma_{max}[S(j\omega)] < |w_s(j\omega)| \quad (68)$$

この問題は結局, 感度関数のゲインを全周波数領域において周波数整形する問題であることがわかる.

### 6.2.2 ロバスト安定性とスモールゲイン定理

次図のようなある特定の不確かさ $\Delta(s)$ がフィードバックされる閉ループ系を考える。ただし、 $\Delta(s), M(s)$ ともに単体では安定であるとする。この閉ループ系が内部安定(内部状態信号が安定)であるための十分条件は次式が成立することである。これをスモールゲイン定理と言う。

$$\sigma_{max}[M(j\omega)\Delta(j\omega)] < 1, \text{ for all } \omega \quad (69)$$

上式の十分条件として、次式がいえる。

$$\sigma_{max}[M(j\omega)] < \frac{1}{\sigma_{max}[\Delta(j\omega)]}, \text{ for all } \omega \quad (70)$$

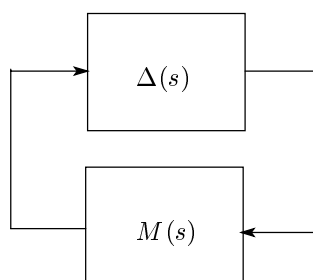


図 14: Small Gain Theorem

さらに、不確かさ $\Delta(s)$ が次の条件を満たすすべての集合 $B$ を考える。

$$B = \{\Delta(s) : \|\Delta\|_{\infty} < \frac{1}{\gamma}\} \quad (71)$$

このとき、すべての $\Delta(s) \in B$ に対して、閉ループ系が内部安定になるための必要十分条件は次式が成り立つことである。

$$\|M(s)\|_{\infty} \leq \gamma \quad (72)$$

### 6.2.3 ロバスト安定化問題(乗法的不確かさ)

乗法的不確かさ $\Delta(s)l_m(s)$ に対するロバスト安定化問題は、次図のような形の一般化プラントの問題に変形できる。ただし、 $\Delta(s)$ は $B$ の任意の要素で、また、 $l_m(s)$ はその周波数重みとする。各信号の関係は次式のようになる。

$$z(s) = -l_m(s)P(s)y(s), \quad y(s) = w(s) - P(s)u(s), \quad u(s) = K(s)y(s)$$

これから、一般化プラントは次式のようになる。

$$\begin{bmatrix} z(s) \\ y(s) \end{bmatrix} = G(s) \begin{bmatrix} w(s) \\ u(s) \end{bmatrix} = \begin{bmatrix} 0 & -l_m(s)P(s) \\ I & -P(s) \end{bmatrix} \begin{bmatrix} w(s) \\ u(s) \end{bmatrix}$$

また、 $w$ から $z$ までの伝達関数は次式のようになる。

$$z(s) = l_m(s)(I + P(s)K(s))^{-1}P(s)K(s)w(s)$$

これから、ロバスト安定化問題は、次式のように定式化される。

$$\|l_m(s)(I + P(s)K(s))^{-1}P(s)K(s)\|_{\infty} \leq \gamma$$

準感度最小化問題と同様に、これは相補感度関数 $T(s)$ を周波数整形する問題であるといえる。この場合の周波数重みは、 $H_{\infty}$ 問題の可解性のために、 $l_m(s)P(s)$ がプロパとなるように選ぶ必要がある。  $P(s)$ が厳密にプロパであるならば $l_m(s)$ はインプロパなものを選ぶ必要がある。

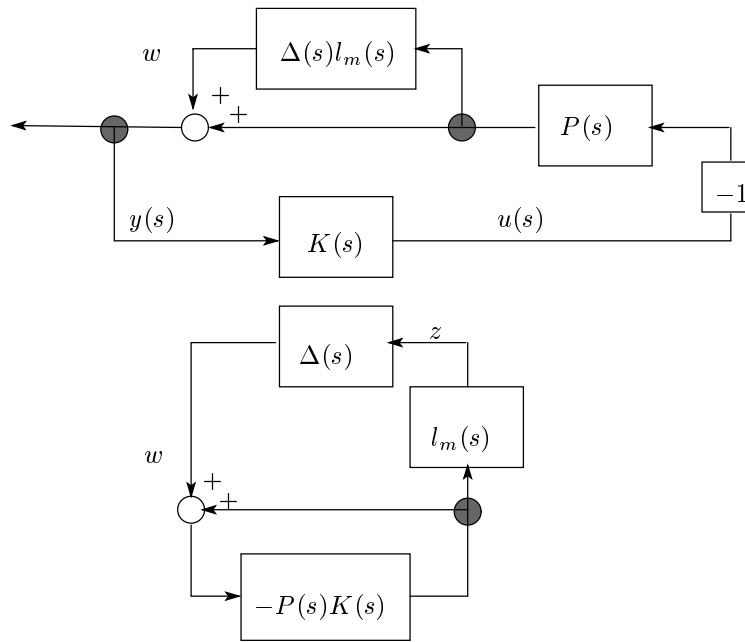


図 15: 乗法的な不確かさに対するロバスト安定化問題

#### 6.2.4 ロバスト安定化問題 (加法的な不確かさ)

加法的な不確かさに対するロバスト安定化問題は、次図のような形の一般化プラントの問題に変形できる。各信号の関係は次式のようになる。

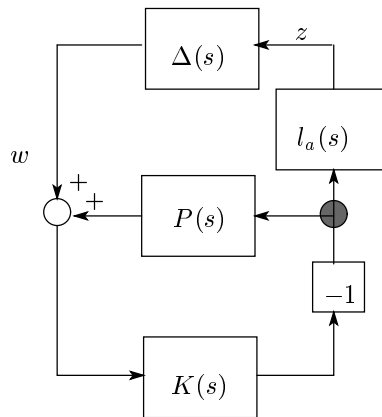


図 16: 加法的な不確かさに対するロバスト安定化問題

$$z(s) = -l_a(s)u(s), \quad y(s) = w(s) - P(s)u(s), \quad u(s) = K(s)y(s)$$

これから、一般化プラントは次式のようになる。

$$\begin{bmatrix} z(s) \\ y(s) \end{bmatrix} = G(s) \begin{bmatrix} w(s) \\ u(s) \end{bmatrix} = \begin{bmatrix} 0 & -l_a(s)I \\ I & -P(s) \end{bmatrix} \begin{bmatrix} w(s) \\ u(s) \end{bmatrix}$$

また、 $w$  から  $z$  までの伝達関数は次式のようになる。

$$z(s) = -l_m(s)K(s)(I + P(s)K(s))^{-1}w(s)$$

これから，ロバスト安定化問題は，次式のように定式化される．

$$\|l_m(s)K(s)(I + P(s)K(s))^{-1}\|_\infty \leq \gamma$$

$K(s)(I + P(s)K(s))^{-1}$ を準相補感度関数と呼ぶ．

### 6.2.5 混合感度問題

混合感度問題は，感度関数  $S(s)$  と相補感度関数  $T(s) = I - S(s)$  あるいは準相補感度関数  $K(s)S(s)$  とを同時に周波数整形する問題である．つぎのような問題が考えられる．

$$\bullet \left\| \begin{array}{c} W_s(s)S(s) \\ l_m(s)T(s) \end{array} \right\|_\infty < \gamma$$

ここで， $S(s) + T(s) = I$ であるので，周波数重みは次式を満足しなければならない．

$$\sigma_{\max}[W_s^{-1}(j\omega)] + \sigma_{\max}[l_m^{-1}(j\omega)] > \gamma^{-1}, \text{ for all } \omega$$

とくに，伝達関数がスカラの場合には， $H_\infty$  ノルム条件を特異値を用いて表すと，次式のようになる．

$$\sigma_{\max} \begin{bmatrix} W_s(j\omega)S(j\omega) \\ l_m(j\omega)T(j\omega) \end{bmatrix} = \sqrt{|W_s(j\omega)S(j\omega)|^2 + |l_m(j\omega)T(j\omega)|^2} \quad (73)$$

ここで，伝達関数がスカラの場合にはつぎの不等式が成り立つ．

$$\max(|W_s(j\omega)S(j\omega)|, |l_m(j\omega)T(j\omega)|) < \sqrt{|W_s(j\omega)S(j\omega)|^2 + |l_m(j\omega)T(j\omega)|^2}$$

これから，つぎのような十分条件が成り立つ．

$$\begin{aligned} \sqrt{|W_s(j\omega)S(j\omega)|^2 + |l_m(j\omega)T(j\omega)|^2} &< \frac{1}{2} \\ \Downarrow \\ \max(|W_s(j\omega)S(j\omega)|, |l_m(j\omega)T(j\omega)|) &< \frac{1}{2} \\ \Downarrow \\ |W_s(j\omega)S(j\omega)| + |l_m(j\omega)T(j\omega)| &< 1 \end{aligned}$$

したがって，この混合感度問題が満足されれば，ロバスト (制御) 性能も満たされることになる．

$$\bullet \left\| \begin{array}{c} W_s(s)S(s) \\ l_a(s)K(s)S(s) \end{array} \right\|_\infty < \gamma$$

$$\bullet \left\| \begin{array}{c} W_s(s)S(s) \\ l_a(s)K(s)S(s) \\ l_m(s)T(s) \end{array} \right\|_\infty < \gamma$$

### 6.3 不確かさを含んだ一般化プラントの閉ループ系

不確かさを含んだ一般化プラントは次図のように表される．ただし，各信号はつぎのように定義されている．

|               |                      |               |                      |
|---------------|----------------------|---------------|----------------------|
| $z(s)$        | 制御量 (設計者が小さくしたい量)    | $w(s)$        | 外部信号 (外乱, 目標値信号など)   |
| $u(s)$        | 操作入力                 | $y(s)$        | 観測出力                 |
| $u_\Delta(s)$ | 不確かさ $\Delta(s)$ の出力 | $y_\Delta(s)$ | 不確かさ $\Delta(s)$ の入力 |

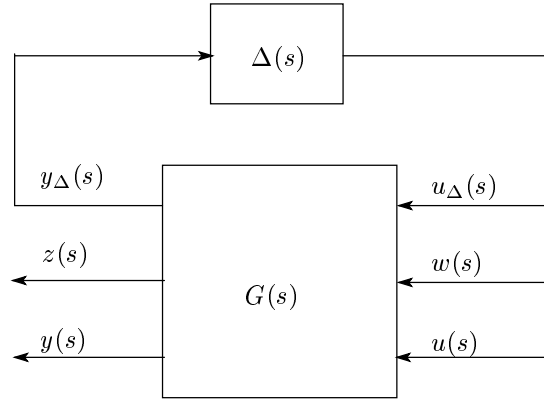


図 17: 不確かさを含む一般化プラント

ここで,  $G(s)$  は次式で与えられる伝達関数である .

$$\begin{bmatrix} y_{\Delta}(s) \\ z(s) \\ y(s) \end{bmatrix} = G(s) \begin{bmatrix} u_{\Delta}(s) \\ w(s) \\ u(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & G_{13}(s) \\ G_{21}(s) & G_{22}(s) & G_{23}(s) \\ G_{31}(s) & G_{32}(s) & G_{33}(s) \end{bmatrix} \begin{bmatrix} u_{\Delta}(s) \\ w(s) \\ u(s) \end{bmatrix} \quad (74)$$

$$u_{\Delta}(s) = \Delta(s)y_{\Delta}(s) \quad (75)$$

これは, 次式のように状態空間表現できる .

$$\begin{bmatrix} \dot{x}(t) \\ y_{\Delta}(t) \\ z(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 & B_3 \\ C_1 & D_{11} & D_{12} & D_{13} \\ C_2 & D_{21} & D_{22} & D_{23} \\ C_3 & D_{31} & D_{32} & D_{33} \end{bmatrix} \begin{bmatrix} x(t) \\ u_{\Delta}(t) \\ w(t) \\ u(t) \end{bmatrix} \quad (76)$$

$$\begin{bmatrix} \dot{x}_{\Delta}(t) \\ u_{\Delta}(t) \end{bmatrix} = \begin{bmatrix} A_{\Delta} & B_{\Delta} \\ C_{\Delta} & D_{\Delta} \end{bmatrix} \begin{bmatrix} x_{\Delta}(t) \\ y_{\Delta}(t) \end{bmatrix} \quad (77)$$

これを Doyle 記号で書くと, つぎのようになる .

$$G(s) = \left[ \begin{array}{c|ccc} A & B_1 & B_2 & B_3 \\ \hline C_1 & D_{11} & D_{12} & D_{13} \\ C_2 & D_{21} & D_{22} & D_{23} \\ C_3 & D_{31} & D_{32} & D_{33} \end{array} \right] \quad (78)$$

$$\Delta(s) = \left[ \begin{array}{c|c} A_{\Delta} & B_{\Delta} \\ \hline C_{\Delta} & D_{\Delta} \end{array} \right] \quad (79)$$

不確かさを含む一般化プラントにコントローラ

$$u(s) = K(s)y(s) = \left[ \begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array} \right] y(s) \quad (80)$$

を入れて閉ループ系を組んだときのブロック線図は, つぎのようになる .

#### 6.4 線形分数変換 (LFT)

一般化プラントの閉ループ系を表現する方法として, 線形分数変換 (linear fractional transformation, LFT) がある . つぎの 2 つのタイプの LFT が定義されている . ここで,  $N(s), M(s)$  は次式のような伝達関数とする .

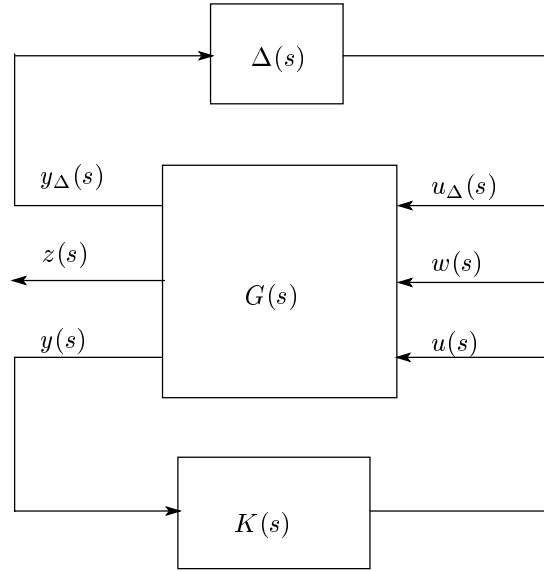


図 18: 一般化プラント

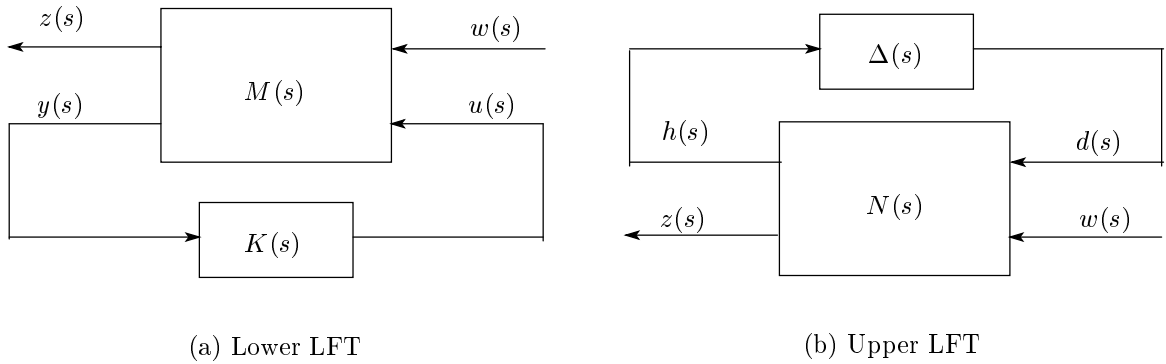


図 19: LFT

$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} \quad \begin{bmatrix} h \\ z \end{bmatrix} = \begin{bmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{bmatrix} \begin{bmatrix} d \\ w \end{bmatrix} \quad (81)$$

このとき、図(a)において、 $I - M_{22}K$ が正則のとき、下線形分数変換 (lower linear fractional transformation, LLFT)  $\mathcal{F}_L(M, K)$  は、次式のように  $w$  から  $z$  までの伝達関数として定義される。

$$\mathcal{F}_L(M, K) = M_{11} + M_{12}K(I - M_{22}K)^{-1}M_{21} \quad (82)$$

また、図(b)において、 $I - N_{11}\Delta$ が正則のとき、上線形分数変換 (upper linear fractional transformation, ULFT)  $\mathcal{F}_U(N, \Delta)$  は、次式のように  $w$  から  $z$  までの伝達関数として定義される。

$$\mathcal{F}_U(N, \Delta) = N_{22} + N_{21}\Delta(I - N_{11}\Delta)^{-1}N_{12} \quad (83)$$

## 6.5 連鎖散乱表現と LFT 表現の関係

(40) 式と LFT とは一見無関係に見えるがそうではないことを説明する。つぎのようなブロック線図の  $\Theta(s)$  を連鎖散乱 (chain scattering) 表現という。

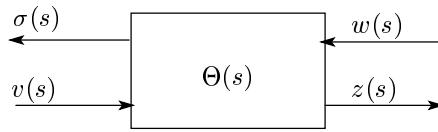


図 20: Chain Scattering 表現

$$\begin{bmatrix} \sigma \\ v \end{bmatrix} = \begin{bmatrix} \Theta_{11}(s) & \Theta_{12}(s) \\ \Theta_{21}(s) & \Theta_{22}(s) \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix} \quad (84)$$

また，一般化プラント  $M(s)$  を次図のようにおく．

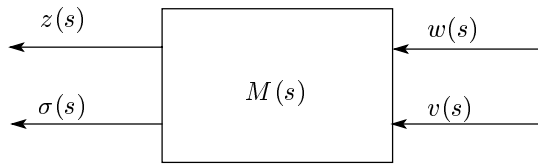


図 21: 一般化プラント表現

$$\begin{bmatrix} z \\ \sigma \end{bmatrix} = \begin{bmatrix} M_{11}(s) & M_{12}(s) \\ M_{21}(s) & M_{22}(s) \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} \quad (85)$$

$\Theta_{22}^{-1}$  が存在すると仮定し，(84) 式を変形すると次式が得られる．

$$\begin{aligned} z &= -\Theta_{22}^{-1}\Theta_{21}w + \Theta_{22}^{-1}v \\ \sigma &= (\Theta_{11} - \Theta_{12}\Theta_{22}^{-1}\Theta_{21})w + \Theta_{12}\Theta_{22}^{-1}v \end{aligned}$$

これから，連鎖散乱表現と一般化プラント表現の関係が次式のようになることがわかる．

$$\begin{aligned} M_{11} &= -\Theta_{22}^{-1}\Theta_{21} \\ M_{12} &= \Theta_{22}^{-1} \\ M_{21} &= \Theta_{11} - \Theta_{12}\Theta_{22}^{-1}\Theta_{21} \\ M_{22} &= \Theta_{12}\Theta_{22}^{-1} \end{aligned}$$

ここで，信号  $\sigma$  と  $v$  をブロック  $Q(s)$  で結合した次図の系を考える．このとき， $v = Q(s)\sigma$  であることが

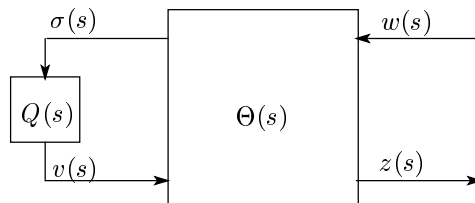


図 22: Chain Scattering 表現

ら，次式が成り立つ．

$$\begin{aligned} \sigma &= \Theta_{11}w + \Theta_{12}z \\ v &= \Theta_{21}w + \Theta_{22}z \end{aligned}$$



$v = Q(s)\sigma$ に代入すると，次式ようになる．

$$\begin{aligned}\Theta_{21}w + \Theta_{22}z &= Q(\Theta_{11}w + \Theta_{12}z) \\ (\Theta_{22} - Q\Theta_{12})z &= (\Theta_{21} - Q\Theta_{11})w\end{aligned}$$

$(\Theta_{22} - Q\Theta_{12})^{-1}$ が存在するとき，次式が得られる．

$$z = (\Theta_{22} - Q\Theta_{12})^{-1}(\Theta_{21} - Q\Theta_{11})w \quad (86)$$

これは，Youla Parameterization と同等の表現になっていることがわかる．また，これは，連鎖散乱表現と一般化プラント表現の関係から，次式のように LLFT 表現できることがわかる．

$$z = \mathcal{F}_L(M, Q)w = (M_{11} + M_{12}Q(I - M_{22}Q)^{-1}M_{21})w \quad (87)$$

## 6.6 摂動プラントと ULFT 表現

- 乗法的な不確かさをもつ摂動プラント： $\tilde{P}(s)$  は次式のように定義されている．

$$\tilde{P}_m(s) = (I + \Delta_m(s))P(s) \quad (88)$$

このとき，摂動プラントは ULFT を用いると，次のように書くことができる．

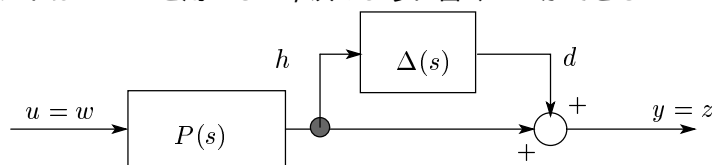


図 23: 乗法的な不確かさをもつプラント

$$\tilde{P}_m = \mathcal{F}_U(N_m, \Delta_m) \quad (89)$$

$$N_m = \begin{bmatrix} 0 & P \\ I & P \end{bmatrix} \quad (90)$$

- 加法的な不確かさをもつ摂動プラント： $\tilde{P}(s)$  は次式のように定義されている．

$$\tilde{P}_a(s) = P(s) + \Delta_a(s) \quad (91)$$

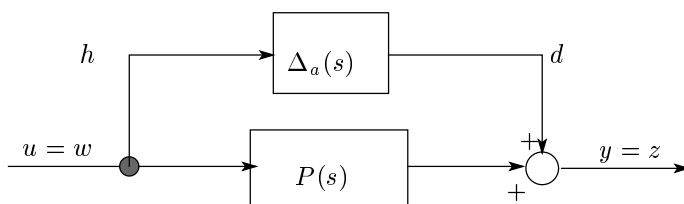


図 24: 加法的な不確かさをもつプラント

$$\tilde{P}_a = \mathcal{F}_U(N_a, \Delta_a) \quad (92)$$

$$N_a = \begin{bmatrix} 0 & I \\ I & P \end{bmatrix} \quad (93)$$

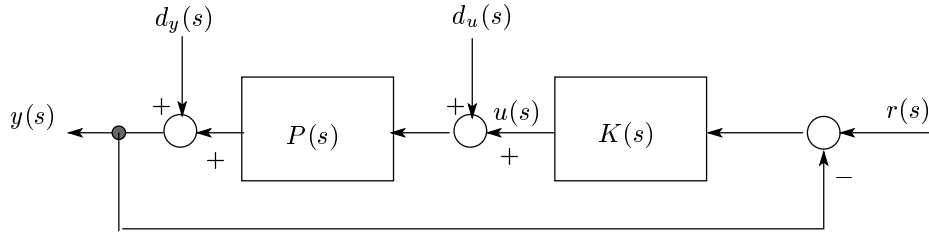


図 25: 単一フィードバック制御系

## 6.7 閉ループ伝達関数と LLFT 表現

つぎの単一フィードバック制御系を考える．この系で出力を  $y, u$  , 入力を  $r, d_y, d_u$  とする入出力関係は次式のようになる．

$$\begin{bmatrix} I & -P \\ K & I \end{bmatrix} \begin{bmatrix} y \\ u \end{bmatrix} = \begin{bmatrix} 0 & I & P \\ K & 0 & 0 \end{bmatrix} \begin{bmatrix} r \\ d_y \\ d_u \end{bmatrix} \quad (94)$$

ここで，次式のような逆行列が存在するとき，フィードバック制御系は意味をもち，このとき，制御系は well-posed であるという．

$$\begin{bmatrix} I & -P \\ K & I \end{bmatrix}^{-1} = \begin{bmatrix} S & SP \\ -KS & I - KSP \end{bmatrix}$$

ただし， $S = (I + PK)^{-1}$  とする．これから，次式が成立する．

$$\begin{bmatrix} y \\ u \end{bmatrix} = \begin{bmatrix} T & S & SP \\ KS & -KS & -KSP \end{bmatrix} \begin{bmatrix} r \\ d_y \\ d_u \end{bmatrix} \quad (95)$$

ただし， $T = (I + PK)^{-1}PK$  とする．このとき，各信号に対する閉ループ伝達関数は，LLFT を用いてつぎのように表すことができる．ここで，信号  $a$  から信号  $b$  までの閉ループ伝達関数を  $T_{ab}$  で書くことにする．

- $T_{yr} = T = \mathcal{F}_L(M, K)$

$$M = \begin{bmatrix} 0 & P \\ I & -P \end{bmatrix}, \quad z = y, \quad w = r$$

- $T_{yd_y} = S = \mathcal{F}_L(M, K)$

$$M = \begin{bmatrix} I & -P \\ I & -P \end{bmatrix}, \quad z = y, \quad w = d_y$$

- $T_{yd_u} = SP = \mathcal{F}_L(M, K)$

$$M = \begin{bmatrix} P & -P \\ P & -P \end{bmatrix}, \quad z = y, \quad w = d_u$$

- $T_{ur} = KS = \mathcal{F}_L(M, K)$

$$M = \begin{bmatrix} 0 & -I \\ I & -P \end{bmatrix}, \quad z = u, \quad w = r$$

- $-T_{ud_u} = KSP = \mathcal{F}_L(M, K)$

$$M = \begin{bmatrix} 0 & I \\ P & -P \end{bmatrix}, z = u, w = d_u$$

## 6.8 摂動プラントの閉ループ伝達関数とLFT表現

次図が不確かさをもつ一般化プラントの閉ループ系である。

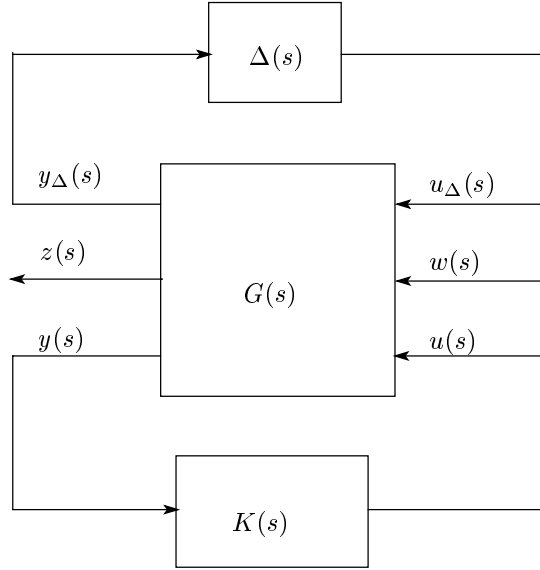


図 26: 不確かさをもつプラントの閉ループ系

$$\begin{bmatrix} y_\Delta \\ z \\ y \end{bmatrix} = G(s) \begin{bmatrix} u_\Delta \\ w \\ u \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} \begin{bmatrix} u_\Delta \\ w \\ u \end{bmatrix} \quad (96)$$

このとき、つぎのようなLFT表現を定義する。

- 摂動開ループ伝達関数:  $\tilde{G}(s) = \mathcal{F}_U(G, \Delta)$
- ノミナル閉ループ伝達関数:  $W(s) = \mathcal{F}_L(G, K) = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}$   
 $W_{11}$ : ロバスト安定性に関係,  $W_{22}$ : ノミナル性能に関係
- 摂動閉ループ伝達関数:  $\tilde{W}(s) = \mathcal{F}_L(\mathcal{F}_U(G, \Delta), K)$ , ロバスト(制御)性能に関係

乗法的な不確かさと加法的な不確かさをもつ摂動プラントを不確かさをもつ一般化プラントで表現すると、つぎのようになる。

- 乗法的な不確かさをもつ摂動プラントの場合

$$G(s) = \begin{bmatrix} 0 & 0 & -P \\ I & I & -P \\ I & I & -P \end{bmatrix}, \Delta = \Delta_m$$

- 加法的な不確かさをもつ摂動プラントの場合

$$G(s) = \begin{bmatrix} 0 & I & -I \\ I & P & -P \\ I & P & -P \end{bmatrix}, \Delta = \Delta_a$$

## 7 $H_\infty$ 制御の標準問題とMATLABによる解法

### 7.1 問題設定とGlover-Doyleの解

次式の一般化プラントとコントローラを対象とする。

$$\begin{bmatrix} z(s) \\ y(s) \end{bmatrix} = G(s) \begin{bmatrix} w(t) \\ u(t) \end{bmatrix} = \left[ \begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array} \right] \begin{bmatrix} w(t) \\ u(t) \end{bmatrix} \quad (97)$$

$$u(s) = K(s)y(s) = \left[ \begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array} \right] y(s) \quad (98)$$

ただし,  $z \in R^m, u \in R^p, y \in R^q, w \in R^r$  とし, 状態変数  $x$  は  $x \in R^n$  とする. このとき,  $w$  から  $z$  までの伝

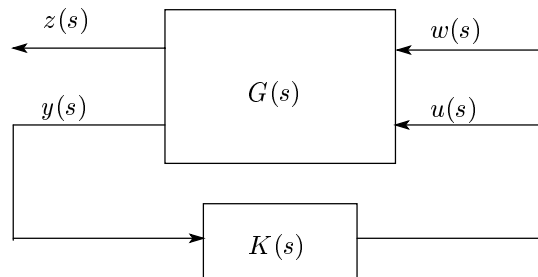


図 27: 一般化プラント

達関数である  $\mathcal{F}_L$  は次式のようになる。

$$\mathcal{F}_L = G_{11}(s) + G_{12}(s)K(s)(I - P_{22}(s))^{-1}G_{21}(s) \quad (99)$$

標準  $H_\infty$  制御問題はつぎのように定義される。

閉ループ系を安定にし, かつ, 指定された値  $\gamma$  に対して,

$$\|\mathcal{F}_L(G, K)\|_\infty < \gamma \quad (100)$$

となるようなコントローラ  $K(s)$  を求めよ。

最小の  $\gamma$  は特別な場合を除いて解析的に求まらないので, 極値探索法を用いて,  $\gamma$  の値が許容の範囲の値になるように繰返す. これを  $\gamma$ -iteration という. さらに, この問題の解の可解性は, ある条件を満たす Riccati 方程式の解の存在性に帰着されるが, そのための前提条件として, つぎのような仮定がある。

- $(A, B_2)$  は可安定
- $D_{12}$  は縦長列フルランク

- $G_{12}(s)$  は虚軸上に零点をもたない．つまり，次式が成立する．

$$\text{rank} \begin{bmatrix} A - j\omega I & B_2 \\ C_1 & D_{12} \end{bmatrix} = n + p, \text{ for all } \omega$$

- $(A, C_2)$  は可検出
- $D_{21}$  は横長行フルランク
- $G_{21}(s)$  は虚軸上に零点をもたない．つまり，次式が成立する．

$$\text{rank} \begin{bmatrix} A - j\omega I & B_1 \\ C_2 & D_{21} \end{bmatrix} = n + q, \text{ for all } \omega$$

1980年代終わり頃，Doyle-Glover は状態空間表現のみから  $H_\infty$  制御の解を導出し，これが  $H_\infty$  制御を一挙にポピュラーなものになった．ここでは， $D_{11} = 0, D_{22} = 0$  としたときの初期のこの解を紹介する．

つぎの2つの Riccati 方程式を定義する．

$$\begin{aligned} & (A - B_2(D_{12}^T D_{12})^{-1} D_{12}^T C_1)^T X_\infty + X_\infty (A - B_2(D_{12}^T D_{12})^{-1} D_{12}^T C_1) \\ & - X_\infty (B_2(D_{12}^T D_{12})^{-1} B_2 - \gamma^{-2} B_1 B_1^T) X_\infty + C_1^T C_1 - C_1^T D_{12} (D_{12}^T D_{12})^{-1} D_{12}^T C_1 = 0 \\ & (A - B_1 D_{21}^T (D_{21} D_{21}^T)^{-1} C_2) Y_\infty + Y_\infty (A - B_1 D_{21}^T (D_{21} D_{21}^T)^{-1} C_2)^T \\ & Y_\infty (C_2^T (D_{21} D_{21}^T)^{-1} C_2 - \gamma^{-2} C_1 C_1^T) Y_\infty + B_1 B_1^T - B_1 D_{21}^T (D_{21} D_{21}^T)^{-1} D_{21} B_1^T = 0 \end{aligned}$$

行列  $F, H$  を2つの Riccati 方程式の半正定解  $X_\infty, Y_\infty$  を用いて，つぎのように記述する．

$$\begin{aligned} F &= \begin{bmatrix} F_{11} \\ F_{12} \\ F_2 \end{bmatrix} = \begin{bmatrix} \gamma^{-2} B_1^T X_\infty \\ -(D_{12}^T D_{12})^{-1} (D_{12}^T C_1 + B_2^T X_\infty) \end{bmatrix} \\ H &= \begin{bmatrix} H_{11} & H_{12} & H_2 \end{bmatrix} = \begin{bmatrix} \gamma^{-2} Y_\infty C_1^T & (B_1 D_{21}^T + Y_\infty C_2^T) (D_{21} D_{21}^T)^{-1} \end{bmatrix} \end{aligned}$$

ただし，各行列の大きさはつぎのようにおいている．

$$\begin{aligned} F_{11} &\in R^{(r-q) \times n}, F_{12} \in R^{q \times n}, F_2 \in R^{p \times n} \\ H_{11} &\in R^{n \times (m-p)}, H_{12} \in R^{n \times p}, H_2 \in R^{n \times q} \end{aligned}$$

このとき， $H_\infty$  コントローラ  $K(s)$  は，つぎの2条件が成り立つとき存在する．

- 2本の Riccati 方程式の準正定解  $X_\infty \geq 0, Y_\infty \geq 0$  が存在する．
- $\lambda_{\max}[X_\infty Y_\infty] < \gamma$

さらに，すべてのコントローラは次式のような LLFT で与えられる．これを  $H_\infty$  コントローラのパラメトリゼーションという．

$$K(s) = \mathcal{F}_L(J(s), Q(s)) \quad (101)$$

$$J(s) = \left[ \begin{array}{c|cc} A_c & B_{c1} & B_{c2} \\ \hline C_{c1} & 0 & I \\ C_{c2} & I & 0 \end{array} \right] \quad (102)$$

$$A_c = A + \begin{bmatrix} H_{11} & H_{12} \end{bmatrix} C_1 + H_2 C_2$$

$$\begin{aligned}
B_{c1} &= -H_2 \\
B_{c2} &= B_2 + H_{12} \\
C_{c1} &= F_2 Z \\
C_{c2} &= -(C_2 + F_{12})Z \\
Z &= (I - \gamma^{-2} Y_\infty X_\infty)^{-1} \\
Q(s) &\in \mathbf{B}_\gamma
\end{aligned}
\tag{103}$$

ただし,  $\mathbf{B}_\gamma$  はつぎのような集合とする.

$$\mathbf{B}_\gamma = \{Q \text{ は安定でプロパな伝達関数} : \|Q\|_\infty < \gamma\}
\tag{104}$$

## 7.2 MATLAB Robust Control Toolbox を用いた $H_\infty$ コントローラの導出

### 7.2.1 関数群

Robust Control Toolbox では, つぎのような関数がある.

|                  |                                 |
|------------------|---------------------------------|
| augss.m, augtf.m | 周波数重みを指定して混合感度問題の一般化プラントの導出     |
| hinf.m           | $H_\infty$ コントローラのパラメトリゼーションの導出 |
| hinfsaf.m        | $H_\infty$ コントローラのパラメトリゼーションの導出 |
| hinfopt.m        | $\gamma$ - iteration            |
| lftf.m           | LFT 計算                          |
| lqg.m            | LQG 制御系設計                       |
| musyn.m          | $\mu$ 設計                        |
| ss.m             | Doyle 記号表現を求める (mksys.m と同様)    |
| normhinf.m       | $H_\infty$ ノルム計算                |
| normh2.m         | $H_2$ ノルム計算                     |
| sigma.m          | 特異値線図                           |
| aresolv.m        | Riccati 方程式の求解                  |
| balmr.m          | 平衡実現による低次元化                     |
| schmr.m          | Shur 法による低次元化                   |
| bilin.m          | 双1次変換によるデジタル化                   |
| hinfdm.mdl       | デモ Simulink モデル                 |

### 7.2.2 例題

hinfdm.mdl を Simulink でロードすると, つぎのようなブロック線図が表示される.

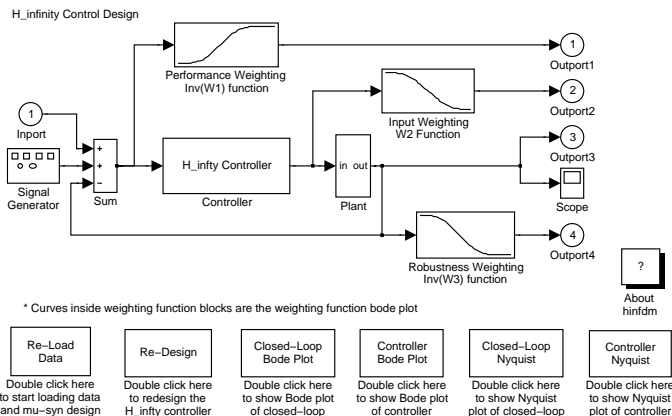
これは, 感度関数, 準相補感度関数, 相補感度関数の混合感度問題を解くためのものである. ウィンドウ内の下側に6つあるブロックをダブルクリックすると, 登録してあるMファイルが起動するようになっている. 特に, 起動Mファイルを確認するには, マウスの右ボタンをクリックしてショートカットメニューを表示させて, ブロックプロパティのオープン関数のところに, 書いてあるものを確認すればよい. 6つの各ブロックについて説明する.

- Re-Load Data のオープン関数で起動されるMファイル:hinfdata.m

```

% This demo provides a SISO closed loop control system designed
% by using the Robust Control Toolbox. Use the pulldown menus to run
% the simulation. Double-click the blocks on the bottom for
% more functions.
%
% All of the parameters are read in from MATLAB workspace variables.

```



28: hinfdm.mdl

```

% The plant model is [a,b,c,d].
%   -1      num1(s)      num2(s)      -1      num3(s)
% w1 (s) = -----; w2(s) = -----; w3 (s) = -----
%           den1(s)      den2(s)      den3(s)
% are the performance weighting function, input weighting function, and
% robustness weighting function respectively. The curves inside the
% weighting function blocks are the magnitudes of the bode plots
% of each weighting function.
%
% The designed H_infinity controller is given by [ae,be,ce,de]. During the
% simulation system measurement noise is added.
%
% By changing the plant and the weighting function parameters, you
% can convert the example to solve a problem of your own.
%
% Re-Load Data
% Re-load data from file. This refreshes the data in the workspace.
%
% Re-Design
% After changing the workspace parameters, you should redesign the
% controller to fit your data.
%
% In this design, the following commands in the Robust Control
% Toolbox are used:
% augss    --- state space plant augmentation with weighting function
% obalreal --- balance realization
% hinfsaf  --- h_infinity controller design
%
% A MIMO control system can be designed using a similar structure.
%-----

```

```

% state-space model of nominal plant
% P(s) = [a,b,c,d]
a = [ -1.0285    0.9853   -0.9413    0.0927
      -1.2903   -1.0957    2.8689    4.7950
        0.1871   -3.8184   -2.0788   -0.9781
        0.4069   -4.1636    2.5407   -1.4236];
b = [      0
      6.6389
        0
        0];
c = [ -1.7786    1.1390         0   -1.0294];
d = 0;
%-----
% frequency weighting function
% W1(s) --> weighting for S(s)
% W2(s) --> weighting for K(s)S(s)
% W3(s) --> weighting for T(s)
%
num1=[1 2 1]/100/1.5;
den1=[1/30/30 2/30 1];
num3=3.16*[1/300 1];
den3=[1/10 1];
num2=[1/100 1/10];
den2=[1 1/10];
%-----
% Hinf controller = [ae,be,ce,de]
%
ae=[
-128.3546   64.4236 -389.6330   50.1446  -43.6734   33.6223  -27.6848  261.3307
   38.1303  -20.9851  125.3862  -15.9982   16.6089   -8.7515    5.5914  -63.4715
 -180.5533   98.3216 -622.2643   81.7920  -98.7638   37.7746  -16.4259  259.3109
   22.5212  -11.7687   74.4666   -9.9342    9.3847   -4.6403    1.3007  -24.4900
  -18.7103    9.4631  -57.4201    7.8872  -12.4971    0.8487    7.2508  -19.2360
    2.5666   -1.3160   11.4041   -0.6599    3.5805   -2.7610    2.2395  -16.7860
   -6.4443    1.7350   -5.5548   -0.0908    7.0539    5.6556  -12.4661  62.7433
   10.5030   -2.4994    5.7297    0.0394   -9.5189   -8.3279    0.8816  -62.3812
];
be=[5.5496
-1.0711
-1.6475
-1.5227
 2.7529
 2.5319
];

```



```

12.4694
-15.5621];
ce=[14.7152 -7.6451 45.7153 -6.2972 5.5336 -3.8930 3.5445 -33.0199];
de=[ 0];

```

- Re-Designのオープン関数で起動されるMファイル:hinfdes1.m

```

% A function for use with the h-infinity Control Toolbox demonstration

% Copyright 1988-2000 The MathWorks, Inc.
% $Revision: 1.6 $
format short e
clear ae be ce de;
sys = ss(a,b,c,d);
[aw1,bw1,cw1,dw1] = tf2ss(den1,num1); sysw1 = ss(aw1,bw1,cw1,dw1);
[aw2,bw2,cw2,dw2] = tf2ss(num2,den2); sysw2 = ss(aw2,bw2,cw2,dw2);
[aw3,bw3,cw3,dw3] = tf2ss(den3,num3); sysw3 = ss(aw3,bw3,cw3,dw3);
clear aw1 aw2 aw3 bw1 bw2 bw3 cw1 cw2 cw3 dw1 dw2 dw3
disp('Use AUGMENT to construct the augmented plant');
disp(' ');
sys_ =augss(sys,sysw1,sysw2,sysw3,0);
disp('Build a balanced system using OBALREAL');
disp(' ');
[A,B1,B2,C1,C2,D11,D12,D21,D22]=branch(sys_);
[aa,bb,cc,mm,tt] = obalreal(A,[B1 B2],[C1;C2]);
A = aa; B1 = bb(:,1); B2 = bb(:,2); C1 = cc(1:3,:); C2 = cc(4,:);
sys_ = rct2lti(mksys(A,B1,B2,C1,C2,D11,D12,D21,D22,'tss'));
disp(' ');
disp('Use HINF to design an H_INF controller');
disp(' ');
[ss_cp]=hinf(sys_);
[ae,be,ce,de]=ssdata(ss_cp);
clear A B1 B2 C1 C2 D11 D12 D21 D22 aa bb cc sysw1 sysw2 sysw3 ss_cp
disp('Done')

```

- Closed-Loop Bode Plot のオープン関数で起動されるコマンド

```

[x,y,z,w]=series(a,b,c,d,ae,be,ce,de);
[x,y,z,w]=cloop(x,y,z,w,-1);
[x,y,w]=bode(x,y,z,w);
clf; subplot(211); loglog(w,x(:,1));
subplot(212); semilogx(w,y(:,1));
subplot(111);

```

- Controller Bode Plot のオープン関数で起動されるコマンド

```

[x,y,w]=bode(ae,be,ce,de);

```

```
clf; subplot(211); loglog(w,x);
subplot(212); semilogx(w,y);
subplot(111);
```

- Closed-Loop Nyquist のオープン関数で起動されコマンド

```
[x,y,z,w]=feedback(a,b,c,d,ae,be,ce,de,-1);
clf, nyquist(x,y,z,w);
```

- Controller Nyquist のオープン関数で起動されるコマンド

```
clf; nyquist(ae,be,ce,de)
```

これから、ノミナルプラントの状態空間表示  $[a, b, c, d]$  と  $W_1(s), W_2(s), W_3(s)$  の周波数重みを書き換え、Re-Design をダブルクリックして全部の条件をクリアすれば、コントローラの状態空間表示  $[ae, be, ce, de]$  を導出できる。さらに、シミュレーションを実行することにより、signal generator から発生する目標値信号に対する応答をシミュレーションできる。少し変形して、つぎのような Simulink ファイルと M ファイルを作成する。また、モータの場合には、 $W_3(s)$  を 2 次の多項式型にすることと、虚軸シフトが必要になる。虚軸シフトして求めたコントローラは逆シフトしてもとに戻す必要がある。

- 多項式型周波数重みの与え方:augtf/m で、周波数重みとノミナルプラントを与えることにより一般化プラントの Doyle 表現を得ることができる。

```
[TSS_] = AUGTF(SS_G,W1,W2,W3)
[A,B1,B2,C1,C2,D11,D12,D21,D22] = AUGTF(AG,BG,CG,DG,W1,W2,W3)
```

ただし、各パラメータはつぎのように定義されている。

|                           |  |
|---------------------------|--|
| ss_g = mksys(ag,bg,cg,dg) | (オリジナルのプラントの状態空間型)   |
| w1                        | 感度関数の重み行列 $W_1$ (MIMO の場合には対角行列)<br>[N11;D11;N22;D22;...] \verb        |
| w2                        | 準相補感度関数の周波数重み行列 $W_2$ (MIMO の場合には対角行列)<br>[N11;D11;N22;D22;...] \verb  |
| w3                        | 相補感度関数の周波数重み行列、周波数無限大では多項式行列 (MIMO の場合には対角行列)<br>[N11;D11;N22;D22;...] |

ただし、重み関数を未定義にする場合には、単に空行列 [] を入力する。

- 虚軸シフト:プラントの伝達関数を次式とする。

$$G(s) = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

ここで、 $s = s' - \alpha$  として、 $G(s')$  をつぎのような伝達関数として定義する。ただし、 $\alpha > 0$  とする。

$$G(s') = G(s' - \alpha)$$

$s'$  複素平面の虚軸は  $s$  複素平面の虚軸を右方向に  $\alpha$  だけシフトしたのになっており、これを虚軸シフトという。 $s'$  複素平面で安定ということは、系の極が  $\text{Re}(s') = 0$  よりも左側にあることを意味してい

る。これを  $s$  平面で考えると、極が  $\text{Re}(s) = -\alpha$  よりも左側にあることに相当し、強い安定化が達成できる。 $G(s')$  を実際に求めるには、次式の状態空間表現を用いる。

$$G(s') = \left[ \begin{array}{c|c} A + \alpha I & B \\ \hline C & D \end{array} \right]$$

さらに、 $G(s')$  に対するコントローラを  $K(s')$ 、

$$K(s') = \left[ \begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array} \right]$$

とすると、もとの  $G(s)$  に対するコントローラ  $K(s)$  は次式のようになる。

$$K(s) = \left[ \begin{array}{c|c} A_K - \alpha I & B_K \\ \hline C_K & D_K \end{array} \right]$$

- 例:たとえば、周波数重みを次式のように選定したい場合を考える。

$$W_1(s) = \frac{\beta(\alpha s^2 + 2\zeta_1\omega_c\sqrt{\alpha}s + \omega_c^2)}{\beta s^2 + 2\zeta_2\omega_c\sqrt{\beta}s + \omega_c^2}$$

$$W_2(s) = [ ]$$

$$W_3(s) = \frac{s^2}{100}$$

$$\beta = 100, \text{ 直流ゲイン ( 応答の追従誤差の調整 )}$$

$$\alpha = 1.5, \text{ 高周波ゲイン ( 応答のピーク値のオーバシュートを調整 )}$$

$$\omega_c = 3, \text{ クロスオーバ周波数の指定}$$

$$\zeta_1 = \zeta_2 = 0.7, \text{ 減衰比}$$

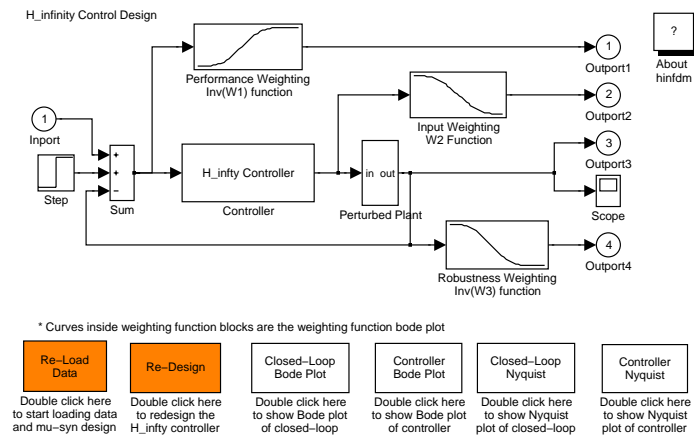
このとき、これらを指定して一般化プラントを求め、コントローラを計算する M ファイルはつぎのようになる。

```
% ag,bg,cg,dg --> nominal plant has already given
ag0 = ag + 0.01*eye(ag); % imaginary-axis shift
w2 = [ ]; w3 = [ 1 0 0; 0 0 100 ]; beta = 100; alpha = 1.5;
w1c = 3; zeta1 = 0.7; zeta2 = 0.7;
w1 = [ beta*[alpha 2*zeta1*w1c*sqrt(alpha) w1c*w1c];...
       [beta 2*zeta2*w1c*sqrt(beta) w1c*w1c] ];
ss_g = mksys(ag0,bg,cg,dg);
TSS_ = augtf(ss_g, w1, w2, w3);
[ss_cp, ss_cl, hinfo] = hinf(TSS_);
[ae,be,ce,de] = branch(ss_cp); % controller
ae = ae - 0.01*eye(ae); % inverse imaginary-axis shift of controller A-matrix
```

そこで、つぎのように書き換えてシミュレーションする。

- Re-Load Data のオープン関数で起動される M ファイル:DCm\_hinfdata.m

```
% state-space matrices of nominal plant
% DC servomotor
```



29: DCm\_hinfdm.mdl

```

% physical parameters
J = 2e-3; D = 0.01; KT = 0.5;
Ke = 0.5; R = 1; L = 0.01;
% state-space matrices of perturbed plant
ap = [ 0  1  0
       0 -D/J KT/J
       0 -Ke/L -R/L ];
bp = [ 0
       0
       1/L ];
cp = [ 1  0  0 ];
dp = 0;
%
an = [ 0  1
       0 -130 ];
bn = [ 0
       250 ];
cn = [ 1  0 ];
dn = 0;
% imaginary-axis shift
shift = 0.001;
a = an+shift*eye(size(an));
b = bn; c = cn; d = dn;
% weighting function
w = logspace(-3,3,100);
w2 = []; w3 = [ 1 110 1000; 0 0 1]*1e-6; beta = 1/100;
w1 = [ 0 200; 1 0.005]*beta;
[aw1,bw1,cw1,dw1] = tf2ss(w1(1,:),w1(2,:));
sysw1 = mksys(aw1,bw1,cw1,dw1);

```

```

sysw3 = mksys(-1,0,0,0);
sysw2 = mksys(-1,0,0,0);
w3poly = w3(1,:);
svw3i = bode(w3(2,:),w3(1,:),w); svw3i = 20*log10(svw3i);
svw1i = bode(w1(2,:),w1(1,:),w); svw1i = 20*log10(svw1i);
disp(' ')
disp(' ')
disp('          (strike a key to see the plot of the weightings ...)')
pause
axis([0 5 -40 40])
semilogx(w,svw1i,w,svw3i)
grid on
xlabel('Frequency - Rad/Sec')
ylabel('1/W1 & 1/W3 - db')
text(2,-20,'Sensitivity Spec.-- 1/W1(s)')
text(2,-80,'Robustness Spec.-- 1/W3(s)')
drawnow
pause
axis
disp('Done')

```

- Re-Designのオープン関数で起動されるMファイル:DCm\_hinfdes1.m

```

% A function for use with the h-infinity Control Toolbox demonstration

% Copyright 1988-2000 The MathWorks, Inc.
% $Revision: 1.6 $
format short e
clear ae be ce de;
sys = mksys(a,b,c,d);
sys_=augtf(sys,w1,w2,w3);
%sys_ = augss(sys,sysw1,sysw2,sysw3,w3poly);
disp(' ');
disp('Use HINF to design an H_INF controller');
disp(' ');
[ss_cp] = hinf(sys_);
[ae,be,ce,de] = ssdata(ss_cp);
ae = ae - shift*eye(size(ae)); % inverse imaginary-axis shift
clear A B1 B2 C1 C2 D11 D12 D21 D22 aa bb cc sysw1 sysw2 sysw3 ss_cp
disp('Done')

```

- 実行結果

<< H-inf Optimal Control Synthesis >>

Computing the 4-block H-inf optimal controller

using the S-L-C loop-shifting/descriptor formulae

Solving for the H-inf controller F(s) using U(s) = 0 (default)  
Solving Riccati equations and performing H-infinity  
existence tests:

1. Is D11 small enough? OK
2. Solving state-feedback (P) Riccati ...
  - a. No Hamiltonian jw-axis roots? FAIL
  - b. A-B2\*F stable (P >= 0)? FAIL
3. Solving output-injection (S) Riccati ...
  - a. No Hamiltonian jw-axis roots? OK
  - b. A-G\*C2 stable (S >= 0)? FAIL
4. max eig(P\*S) < 1 ? OK

-----  
NO STABILIZING CONTROLLER MEETS THE SPEC. !!

-- CLOSED-LOOP UNSTABLE --

全部 OK になるまで、重みの選定を変える。でも、なかなか出ない。

### 7.3 MATLAB $\mu$ -Analysis and Synthesis Toolbox を用いた $H_\infty$ コントローラの導出

#### 7.3.1 関数群

|                  |                             |
|------------------|-----------------------------|
| augss.m, augtf.m | 周波数重みを指定して混合感度問題の一般化プラントの導出 |
| frsp.m           | 周波数応答を varying データで出力       |
| vplot.m          | varying データのプロット            |
| pck.m            | Doyle 記号の構造体データ作成           |
| nd2sys.m         | 伝達関数を Doyle 記号に変換           |
| sysic.m          | 一般化プラントの定義                  |
| hinfsyn.m        | $H_\infty$ コントローラ設計         |
| vsvd.m           | 最大特異値の計算                    |
| pkvnorm.m        | $H_\infty$ ノルムの計算           |
| hinfnorm.m       | $H_\infty$ ノルムの計算           |

ここで、いくつかの M ファイルの使用法を説明しておく。

- nd2sys.m

使用方法: 伝達関数  $W(s) = K \frac{\text{num}(s)}{\text{den}(s)}$  を Doyle 表現  $W_s$  に変換する場合

```
num = [1 1]; den = [1 2]; K = 10;  
Ws = nd2sys(num,den,K);
```

- pck.m

使用方法: 状態空間表現  $(A, B, C, D)$  を Doyle 表現  $Pr$  に変換する場合

```
A = [0 1; -1 -2]; B = [0; 1]; C = [1 0]; D = [0];  
Pr = pck(A,B,C,D);
```

- unpcck.m

使用方法: Doyle 表現  $Pr$  から状態空間表現  $(A, B, C, D)$  に変換する場合

```
[A,B,C,D] = unpck(Pr);
```

- sysic.m

使用方法: つぎの閉ループ系での一般化プラント  $G$  を定義する場合 外部入力  $w$  から制御量

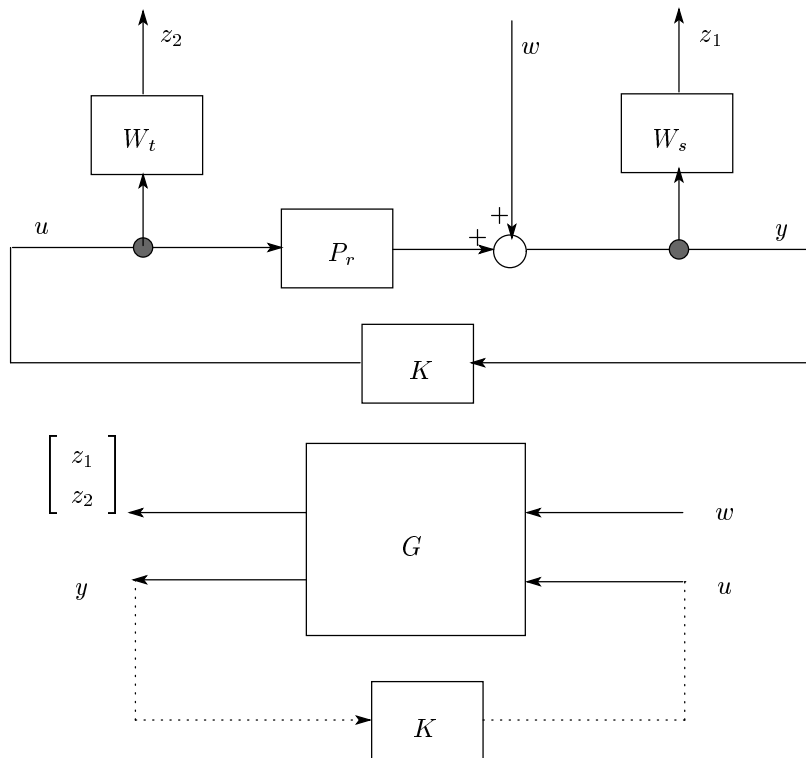


図 30: 閉ループ系と一般化プラント

$z_1, z_2$  までの閉ループ系の伝達関数は、次式のようになる。

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_S(I - P_r K)^{-1} \\ W_T K(I - P_r K)^{-1} \end{bmatrix} w$$

これは、感度関数と準相補感度関数の混合感度問題になっている。これを `sysic.m` を用いて、一般化プラントを定義するとつぎのようになる。

% まず,  $P_r, W_s, W_t$  を Doyle 表現で定義しておく必要がある。

```
systemnames = ' Pr Ws Wt ';
inputvar = '[ dist; control ]';
outputvar = '[ Ws; Wt; Pr + dist ]';
input_to_Pr = '[ control ]';
input_to_Ws = '[ Pr + dist ]';
input_to_Wt = '[ control ]';
sysoutname = 'G';
cleanupsysic = 'yes';
sysic;
```

- Simulink を用いて、一般化プラントを求める場合:sysic でコマンドベースで記述するのが面倒な場合には、Simulink によるブロック線図ベースで一般化プラントをlinmod.m関数を用いて導出できる。設計用のSimulinkファイルは,DCSMgp\_slink.mdlである。さらに、一般化プラントを定義するSimulinkファイルDCSMgp\_slink.mdlはつぎのようにおいている。ただし、プラント、周波数重みはすべて状

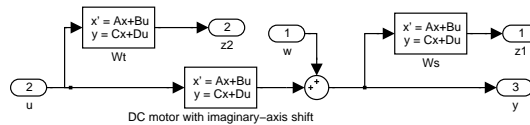


図 31: DCSMgp\_slink.mdl

態空間表現であらかじめ定義しておく必要がある。さらに、Mファイルから、つぎのように一般化プラントの  $A, B, C, D$  と Doyle 表現を求めることができる。

```
[aa,bb,cc,dd] = linmod('DCSMgp_slink');
G = pck(aa,bb,cc,dd);
nmeans = 1;% dimension of y
ncon = 1;% dimension of u
```

### 7.3.2 例題1:Mass-Spring系

ここでは、2次の振動系を例に、 $\mu$ -Analysis and Synthesis Toolboxを用いて、 $H_\infty$ コントローラを求めてみよう [35]。安定な系であるので、比較的簡単に求められる。

$$P(s) = \frac{1}{ms^2 + cs + k}, \quad m = 0.01, k = 1, c = 0.001;$$

周波数重みをつぎのように選定している。

$$W_1(s) = 1000 \frac{0.01s + 1}{100s + 1}$$

$$W_3(s) = 10 \frac{s + 10}{s + 1000}$$

このときのSimulinkファイルはつぎのようになる。また、Simulinkファイルに割り当てられているMファ

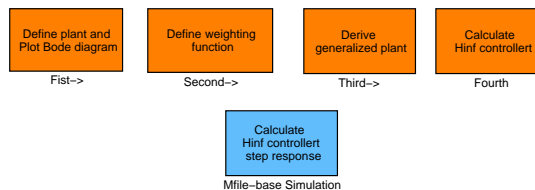
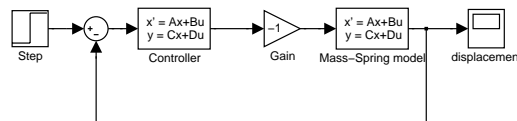


図 32: SMall.mdl

イルはつぎのとおりである。



- SM\_mkplant.m: 制御対象の定義とボード線図表示

```

% Definition of the plant (one-degree-of-freedom system)
% mkplant.m
clear
m=0.01; k=1; c=0.001;
ap = [ 0 1 ;
      -k/m -c/m ];
bp = [ 0 ; 1/m ];
cp = [ 1 0 ];
dp = [ 0 ];
Pr = pck(ap,bp,cp,dp);
w=logspace(0,2,100);
Pr_g = frsp(Pr,w);
vplot('bode',Pr_g);
end

```

- SM\_defwgt.m:

```

% Definition of the weighting functions (CASE 1)
% defwgt.m

clf
numws=[1/100 1];
denws=[1/0.01 1];
wsgain = 1000;
Ws = nd2sys(numws,denws,wsgain);
numwt=[1 10];
denwt=[1 1000];
wtgain = 10;
Wt = nd2sys(numwt,denwt,wtgain);
w=logspace(-1,4,100);
Ws_g = frsp(Ws,w);
Wt_g = frsp(Wt,w);
vplot('liv,lm',Ws_g,'-',Wt_g,'--');
xlabel('Frequency [rad/s]'), ylabel('Gain')
legend('Ws','Wt')

```

- SM\_defgp.m

```

% Definition of the generalized plant (CASE 1)
% defgp.m
systemnames = ' Pr Ws Wt ';
inputvar = '[ dist; control ]';
outputvar = '[ Ws; Wt; Pr + dist ]';
input_to_Pr = '[ control ]';

```

```

input_to_Ws = '[ Pr + dist ]';
input_to_Wt = '[ control ]';
sysoutname = 'G';
cleanupsysic = 'yes';
sysic;

```

- SM\_hinf\_cal.m

```

gmax = 10; gmin = 0;
k = hinfsyn(G,1,1,gmin,gmax,0.1,2); % k --> Doyle rep. of controller
[ak,bk,ck,dk] = unpck(k);

```

- SM\_hinf\_sim.m

```

% Performance test for H_infinity controller
% hinf_sim.m

disp('Info. of Pr')
minfo(Pr)
disp('Info. of k')
minfo(k)

ksim = mmult(k,-1);

% Feedback connection by using "sysic" and "starp"
systemnames = ' Pr';
inputvar     = '[ dist; ref; control]';
outputvar    = '[ Pr; control; ref - Pr]';
input_to_Pr  = '[ control + dist ]';
sysoutname   = 'simG';
cleanupsysic = 'yes';
sysic;
simplant     = starp(simG,ksim);
[aa,bb,cc,dd] = unpck(simplant);

% step response
t=0:0.01:1;
y=step(aa,bb,cc,dd,2,t);

plot(t,y(:,1)), title('Step response')
xlabel('Time [s]'), ylabel('Position')
disp('Please hit any key to continue'), pause

plot(t,y(:,2)), title('Step response')
xlabel('Time [s]'), ylabel('Input')
disp('Please hit any key to continue'), pause

```

```

% impulse response
yimp=impulse(aa,bb,cc,dd,1,t);

plot(t,yimp(:,1)), title('Impulse disturbance response')
xlabel('Time [s]'), ylabel('Position')
disp('Please hit any key to continue'), pause

plot(t,yimp(:,2)), title('Impulse disturbance response')
xlabel('Time [s]'), ylabel('Input')
disp('Please hit any key to continue'), pause

```

### 7.3.3 例題2:DCサーボモータ

ここでは、2次の振動系にならって、これまでに例題で用いたDCサーボモータの $H_\infty$ コントローラを求めてみよう。この場合も、感度関数と準感度関数の混合感度問題では、虚軸シフトを行うことにより、比較的簡単に $H_\infty$ コントローラを求められる。このときのSimulinkファイルDCSMa11.mdlで、構成は例題1と全く同じである。ただし、Simulinkファイルに割り当てられているMファイルはつぎのとおりである。

- DCSM\_mkplant.m: 制御対象の定義とボード線図表示

```

% Definition of the plant (one-degree-of-freedom system)
% mkplant.m
clear
ap = [0 1
      0 -130];
bp = [0
      250];
cp = [1 0];
dp = 0;
shift=0.1; % imaginary axis-shift
aps = ap + shift*eye(size(ap));
Pr = pck(aps,bp,cp,dp);
w=logspace(0,2,100);
Pr_g = frsp(Pr,w);
vplot('bode',Pr_g);
end

```

- DCSM\_defwgt.m:

```

% Definition of the weighting functions (CASE 1)
% defwgt.m
clf
numws=[1/100 1];
denws=[1/0.01 1];
wsgain = 1000;

```

```

Ws = nd2sys(numws,denws,wsgain);
numwt=[1 10];
denwt=[1 1000];
wtgain = 10;
Wt = nd2sys(numwt,denwt,wtgain);
w=logspace(-1,4,100);
Ws_g = frsp(Ws,w);
Wt_g = frsp(Wt,w);
vplot('liv,lm',Ws_g,'-',Wt_g,'--');
xlabel('Frequency [rad/s]'), ylabel('Gain')
legend('Ws','Wt')

```

- DCSM\_hinf\_cal.m

```

% calculation of Hinf controller using hinfsyn.m
gmax = 10; gmin = 0;
k = hinfsyn(G,1,1,gmin,gmax,0.1,2); % k --> Doyle rep. of controller
[ak,bk,ck,dk] = unpck(k);
ak = ak-shift*eye(size(ak)); % inverse imaginary axis shift

```

- DCSM\_hinf\_sim.m: SM\_hinf\_simと同じ

sysic.mを用いずに、前述のSimulinkファイルDCSMgp\_slink.mdlを用いて、コントローラ設計する場合のファイル構成はつぎのようになっている。

- 設計用SimulinkファイルDCSMall\_slink.mdl
- 一般化プラント定義SimulinkファイルDCSMgp\_slink.mdl
- ノミナルプラント定義MファイルDCSM\_mkplant.m:前と同じ
- ノミナルプラント定義MファイルDCSM\_defwgt2.m

```

% Definition of the weighting functions (CASE 1)
% defwgt.m
clf
numws=[1/100 1];
denws=[1/0.01 1];
wsgain = 1000;
Ws = nd2sys(numws,denws,wsgain);
[aws,bws,cws,dws] = tf2ss(wsgain*numws,denws);
numwt=[1 10];
denwt=[1 1000];
wtgain = 10;
[awt,bwt,cwt,dwt] = tf2ss(wtgain*numwt,denwt);
Wt = nd2sys(numwt,denwt,wtgain);
w=logspace(-1,4,100);
Ws_g = frsp(Ws,w);

```

```

Wt_g = frsp(Wt,w);
vplot('liv,lm',Ws_g,'-',Wt_g,'--');
xlabel('Frequency [rad/s]'), ylabel('Gain')
legend('Ws','Wt')

```

- ノミナルプラント定義MファイルDCSM\_defgpslink.m

```

% Definition of the generalized plant (CASE 1)
% defgp.m
[aa,bb,cc,dd] = linmod('DCSMgp_slink');
G = pck(aa,bb,cc,dd);
nmeans = 1;% dimension of y
ncon = 1;% dimension of u

```

- $H_\infty$  コントローラ設計DCSM\_hinf\_cal.m:同じ
- 応答シミュレーションDCSM\_hinf\_sim.m:同じ

このときのコントローラの $\gamma$  - iteration の出力結果はつぎのようになる .

```

Test bounds:      0.1000 < gamma <=      10.0000

gamma   hamx_eig  xinf_eig  hamy_eig  yinf_eig  nrho_xy  p/f
10.000  6.7e+000  1.7e-011  1.0e-002  0.0e+000  0.0004   p
 5.050  6.7e+000  1.7e-011  1.0e-002  -2.8e-045  0.0017   p
 2.575  6.5e+000  1.7e-011  1.0e-002  -7.1e-013  0.0078   p
 1.337  6.0e+000  1.8e-011  1.0e-002  -2.8e-045  0.0943   p
 0.719  3.3e+000  -1.5e-001# 1.0e-002  -2.8e-045  0.0283   f
 1.028  5.4e+000  -1.2e+000# 1.0e-002  -2.8e-045  0.1095   f
 1.183  5.8e+000  1.8e-011  1.0e-002  -2.8e-045  2.1130#  f
 1.264  5.9e+000  1.8e-011  1.0e-002  -2.8e-045  0.1758   p

```

Gamma value achieved: 1.2638

閉ループ系のステップ応答波形はつぎのようになる .

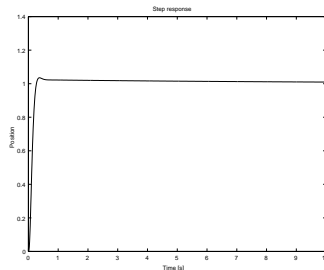


図 33:  $H_\infty$  制御系のステップ応答.mdl

## 7.4 MATLAB LMI Toolbox を用いた $H_\infty$ コントローラの導出

### 7.4.1 関数群

|                          |  |
|--------------------------|--|
| augss.m, augtf.m         | 周波数重みを指定して混合感度問題の一般化プラントの導出  |
| [gopt,K]=hinflmi(P,r)    | LMI ベースで $H_\infty$ コントローラを計算<br>P: 一般化プラント, $r:r=[p2,m2]$ の形で与える. ただし, $p2$ は $y$ の次元, $m2$ は $u$ の次元 |
| [gopt,K]=hinfric(P,r)    | K: コントローラの Doyle 表現<br>Riccati 方程式ベースで $H_\infty$ コントローラを計算  |
| lmiedit.m                | LMI 問題設定のための GUI   |
| [a,b,c,d]=ltiss(sys)     | Doyle 表現から $A, B, C, D$ を抽出  |
| sys=ltisys(a,b,c,d)      | $A, B, C, D$ から Doyle 表現に変換  |
| sys=ltisys(a,b,c,d,e)    | ディスクリプタ表現 $A, B, C, D, E$ から Doyle 表現に変換   |
| sys=ltisys('tf',num,den) | 伝達関数から Doyle 表現に変換   |
| [num,den]=ltitf(sys)     | Doyle 表現から伝達関数に変換  |
| magshape.m               | 周波数整形用フィルタ設計のための GUI   |
| msfsyn.m                 | 多目的制御系設計   |
| smult.m                  | 直列結合   |
| sinv.m                   | 逆システム  |
| sadd.m                   | 並列結合   |
| sloop.m                  | フィードバック結合  |
| splot.m                  | 様々な周波数応答と時間応答を計算   |
| sresp.m                  | 周波数応答  |

### 7.4.2 例題

LMI Control Toolbox を用いて,  $H_\infty$  コントローラを計算する M ファイルはつぎのようになる.

- 
- SM\_hinf\_cal.m

```
gmax = 10;
[gopt,k] = hinflmi(G,1,1,gmax); % k --> Doyle rep. of controller
[ak,bk,ck,dk] = unpck(k);
```

サーボモータの例で hinfsyn.m のかわりに, hinflmi.m を用いると, つぎのような計算結果が表示される.

Minimization of gamma:

Solver for linear objective minimization under LMI constraints

Iterations : Best objective value so far

```
1
2
3
4 1.698937e+004
5 1.207188e+004
6 5571.115079
7 5571.115079
8 5571.115079
```

|    |             |
|----|-------------|
| 9  | 5571.115079 |
| 10 | 817.886944  |
| 11 | 521.039546  |
| 12 | 521.039546  |
| 13 | 173.286161  |
| 14 | 173.286161  |
| 15 | 70.974525   |
| 16 | 45.010104   |
| 17 | 24.919856   |
| 18 | 11.108469   |
| 19 | 11.108469   |
| 20 | 8.382987    |

Result: reached the target for the objective value  
 best objective value: 8.382987  
 f-radius saturation: 21.103% of R = 1.00e+008

Optimal Hinf performance: 5.315e+000

閉ループ系のステップ応答波形はつぎのようになる .

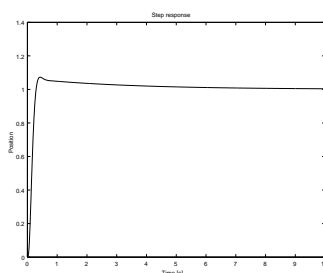


図 34:  $H_{\infty}$  制御系のステップ応答.mdl

## 8 $\mu$ 解析/設計

設計例については, [23, 26, 34]などを参考にしてください .

## 9 LMIによる制御系設計

設計例に付いては, [29, 36]などを参考にしてください .

## 参考文献

- [1] 有本 卓: 線形システム理論 産業図書 (1974)
- [2] 木村英紀: 動的システムの理論 産業図書 (1974)

- [3] 伊藤, 木村, 細江: 線形制御系の設計理論 計測自動制御学会 (1978)
- [4] 木村: 多変数制御系の理論と応用I, システムと制御, vol.22, no.5, pp.293-301 (1978)
- [5] H.Kwakernaak and R.Sivan: Linear Optimal Control Systems, Wiley(1972)
- [6] M.Vidyasagar: Control System Synthesis, MIT Press(1985)
- [7] 前田, 杉江: アドバンスト制御のためのシステム制御理論 朝倉書店 (1990)
- [8] K. Glover, "All optimal Hankel-norm approximations of linear multivariable systems and their  $\mathcal{L}_\infty$ -error bounds, Int. J. Control, 43-3, pp.1115-1193 (1984).
- [9] B.A.Francis: A Course in  $H_\infty$  Control Theory, Springer(1987)
- [10] J.Doyle, A.Packard and K.Zhou: Review of LFTs, LMIs, and  $\mu$  (1991)
- [11] S.P. Boyd and C.H. Barratt: Linear Controller Design, Limits of Performance, Prentice-Hall, Inc. (1991).
- [12] R.L.Dailey: Lecture Notes for the Workshop on  $H_\infty$  and  $\mu$  Methods for Robust Control, 1990ACC, San Diego, CA (1990)
- [13] 木村英紀: LQG から  $H_\infty$  へ, 計測と制御, vol.29-2, pp.111-119 (1990)
- [14] H.Kimura: Conjugation, interpolation and model-matching in  $H_\infty$ , Int.J of Control, vol.49-1, pp.269-307 (1989)
- [15] 木村英紀: J-Lossless Factorization にもとづく  $H_\infty$ , 第41回システム制御情報講習会テキスト (1992)
- [16] J.C.Doyle, K.Glover, P.P.Khargonekar and B.A.Francis: State-Space Solutions to Standard  $H_2$  and  $H_\infty$  Control Problems, IEEE Trans. on Automatic Control, vol.34-8, pp.831-846 (1989)
- [17] J.C.Doyle, K.Glover, P.P.Khargonekar and B.A.Francis: State-Space Solutions to Standard  $H_2$  and  $H_\infty$  Control Problems, 1988ACC, Atlanta, GA, June (1988)
- [18] A.Feintuch, R.Saeks: System Theory – A Hilbert Space Approach, AP (1982)
- [19] B.D.O. Anderson and J.B. Moore: Optimal Control – Linear Quadratic Control – (1990).
- [20] J.C.Doyle, B.A.Francis and A.R.Tannenbaum: Feedback Control Theory, Macmillan (1992) (藤井 監訳: フィードバック制御の理論, コロナ社)
- [21] B.R. Barmish: New Tools for Robustness of Linear Systems, Macmillan, Inc. (1994).
- [22] M.A.Rotea and P.P.Khargonekar:  $H^2$ -optimal Control with an  $H^\infty$ -constraint: The State Feedback Case, Automatica, vol.27-2, pp.307-316 (1991)
- [23] 美多 勉:  $H_\infty$ 制御, 昭晃堂 (1994).
- [24] 木村, 藤井, 森: ロバスト制御, コロナ社 (1994).
- [25] 吉川, 井村: 現代制御論, 昭晃堂 (1994).
- [26] 細江, 荒木: 制御系設計, 朝倉書店 (1994).



- [27] 早勢 実： $H_\infty$ 制御入門，オーム社 (1996).
- [28] S. Skogestad and I. Postlethwaite, Multivariable Feedback Control, John Wiley & Sons, (1996).
- [29] 岩崎徹也：LMIと制御，昭晃堂 (1997).
- [30] 細江：システムと制御，オーム社 (1997).
- [31] F.W. Fairman, Linear Control Theory, The State Space Approach, John Wiley & Sons, Inc. (1998).
- [32] K. Zhou, J.C. Doyle, and K. Glover, Robust and Optimal Control, Prentice Hall, Inc. (1996). 日本語訳，劉，羅訳，ロバスト最適制御，コロナ社 (1997).
- [33] 木村英紀： $H_\infty$ 制御，コロナ社 (2000).
- [34] 藤森 篤：ロバスト制御，コロナ社 (2001).
- [35] 野波健蔵編著：MATLABによる制御系設計，東京電機大学出版局 (1998).
- [36] 梶原宏之：LMI Control Toolboxによる制御系設計入門，MATLAB Product News, 1996,no.3,no.4,no.5.