

Robust and Optimal Control, Spring 2015

Instructor: Prof. Masayuki Fujita (S5-303B)

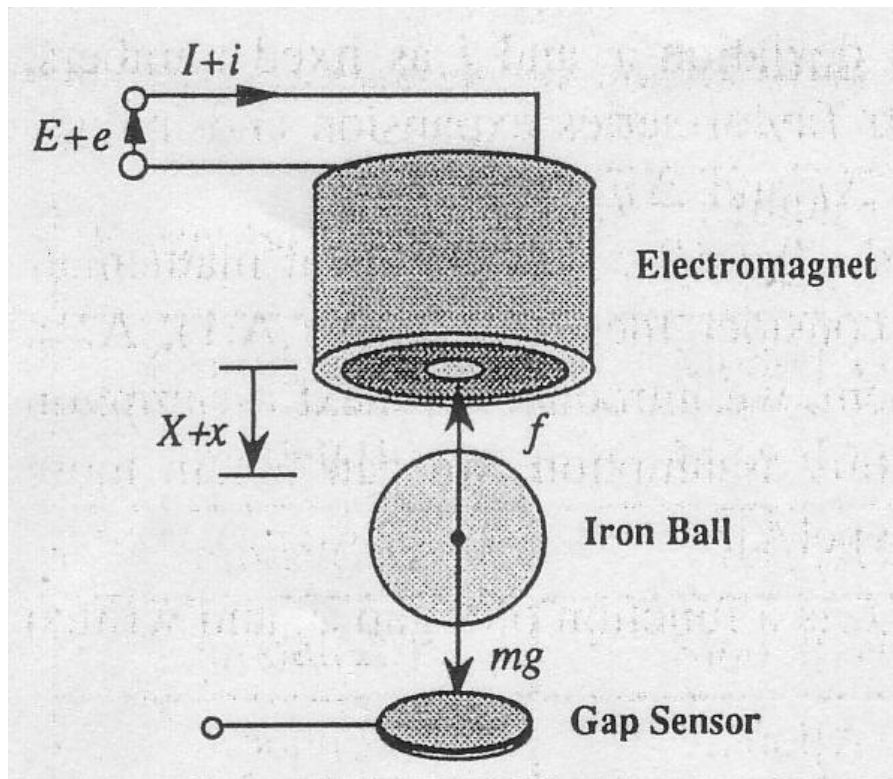
S: Magnetic Suspension System: μ -synthesis

Reference:

M. Fujita, T. Namerikawa, F. Matsumura and K. Uchida
 μ -Synthesis of an Electromagnetic Suspension System,
IEEE Transactions on Automatic Control, Vol. 40, No. 3,
530-536, 1995.

磁気浮上システム

Real Physical System



- f : electromagnetic force
- m : mass of the iron ball
- I : steady current of the electromagnet
- i : displacement from the steady current
- E : steady voltage of the electromagnet
- e : displacement from the steady voltage
- X : steady gap between the electromagnet and the iron ball
- x : displacement from the steady gap
- R : resistance of the electromagnet
- L : inductance of the electromagnet

図 磁気浮上システム

制御系設計

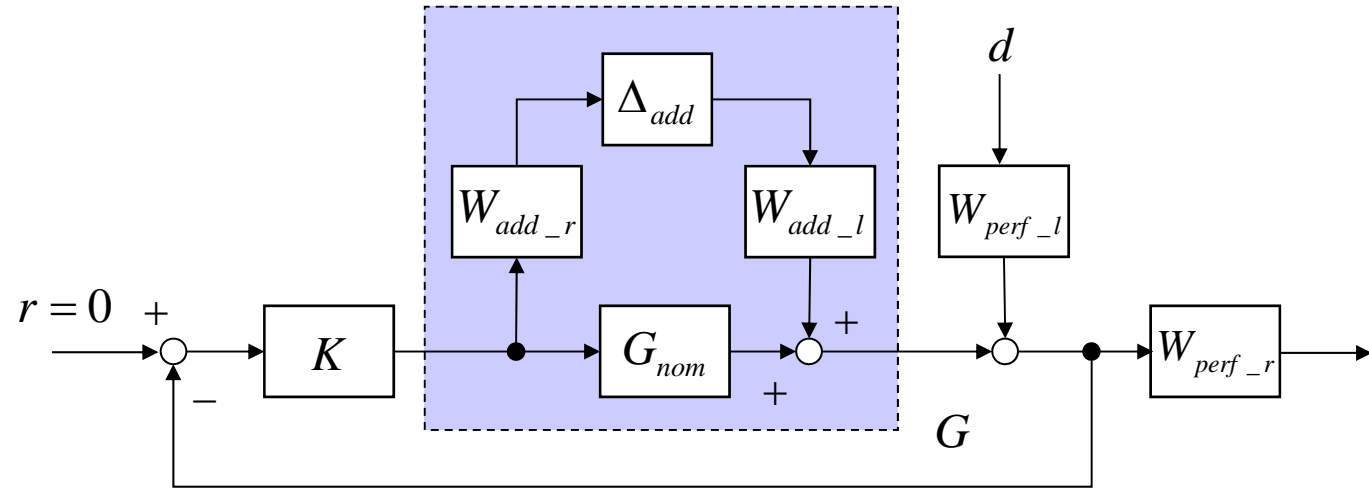


図 フィードバック構造

ロバスト安定性 $\|W_{add_r} K(I + G_{nom} K)^{-1} W_{add_l}\|_{\infty} < 1$

ロバスト性能 $\|W_{perf_r} (I + GK)^{-1} W_{perf_l}\|_{\infty} < 1,$

[制御問題]

すべての $G \in \mathbf{G}$ に対して閉ループ系を内部安定とし、かつ

$$\|W_{perf_r} (I + GK)^{-1} W_{perf_l}\|_{\infty} < 1,$$

を満たすコントローラが存在するかどうか判定し、
存在する場合はそのようなコントローラを求めよ。

制御系設計の手順

[ステップ1] 制御対象の数学的モデルを求める.

線形化

特性解析 状態空間表現 可制御性, 可観測性
 伝達関数表現 ボード線図, 極, 零点,

[ステップ2] 制御目的から, 性能仕様を決める.

[ステップ3] 性能仕様を満たすように, コントローラを設計する.

[ステップ4] シミュレーションにより, 設計された制御系を評価する.
必要ならば, 以上のステップを繰り返し設計をやり直す.

[ステップ5] コントローラを実装しハードウェアを用いてテストする.

[ステップ1] 制御対象の数学的モデルを求める.

線形化

特性解析 状態空間表現 可制御性, 可観測性
 伝達関数表現 ボード線図, 極, 零点,

Parameter

記号	最大値	公称値	最小値
m	---	1.75	---
X	5.50×10^{-3}	5.50×10^{-3}	4.50×10^{-3}
I	1.18	1.06	0.93
x	5.00×10^{-4}	0.00	-5.00×10^{-4}
i	1.18×10^{-1}	0.00	-1.26×10^{-1}
L	5.57×10^{-1}	5.08×10^{-1}	4.65×10^{-1}
R	2.37×10^1	2.32×10^1	2.27×10^1
k	3.35×10^{-4}	2.90×10^{-4}	2.53×10^{-4}
x_0	-3.32×10^{-4}	-6.41×10^{-4}	-9.42×10^{-4}
Q	6.70×10^{-4}	5.79×10^{-4}	5.06×10^{-4}
x_∞	-3.32×10^{-4}	-6.41×10^{-4}	-9.42×10^{-4}
L_0	3.96×10^{-1}	3.75×10^{-1}	3.54×10^{-1}

仮定

- A.1) 鉄心の透磁率は無限大である
- A.2) 漏れ磁束, 磁気飽和, ヒステリシスがない
- A.3) 鉄心中に生じるうず電流は無視できる
- A.A) インダクタンスは平衡点付近で定数
- A.B) コイルのインダクタンス L はギャップ x の関数であり

$$L(x) = \frac{Q}{x + X_\infty} + L_0$$

と表される.

Ideal Mathematical Model

4つの異なる線形モデル

- Type[A] {
- Model[A1]: $L=\text{定数}$; 非線形項を
1 次のテイラー展開で近似
 - Model[A2]: $L=\text{定数}$; 非線形項を
2 次のテイラー展開で近似

- Type[B] {
- Model[B1]: $L=L(x)$; 非線形項を
1 次のテイラー展開で近似
 - Model[B2]: $L=L(x)$; 非線形項を
2 次のテイラー展開で近似

Type[A]

- 数学モデル

$$m \frac{d^2 x}{dt^2} = mg - f,$$
$$f = k \left(\frac{i}{x + x_0} \right)^2$$

$$Ri + L \frac{di}{dt} = e$$

$$L = \text{Constant}$$

- 線形モデル

$$\dot{x} = Ax + Bu,$$

$$y = Cx$$

$$x = \begin{bmatrix} x & \dot{x} & i \end{bmatrix},$$
$$u = e, \quad y = x$$

線形モデル: Model[A1]

$$A = \begin{bmatrix} 0 & 1 & 0 \\ \frac{2kI^2}{m(X+x_0)^3} & 0 & -\frac{2kI}{m(X+x_0)^2} \\ 0 & 0 & -\frac{R}{L} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} \quad C = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T$$

線形モデル: Model[A2]

$$A = \begin{bmatrix} 0 & 1 & 0 \\ \frac{2kI^2}{m(X+x_0)^3} \Delta y & 0 & -\frac{2kI}{m(X+x_0)^2} \Delta y \\ 0 & 0 & -\frac{R}{L} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} \quad C = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T$$

Type[B]

• 数学モデル

$$m \frac{d^2 x}{dt^2} = mg - f,$$

$$f = k \left(\frac{i}{x + x_0} \right)^2$$

$$e = Ri + \frac{d}{dt}(Li),$$

$$L = \frac{2k}{x_0 + x} + L_0,$$

$$L = L(x)$$

• 線形モデル

$$\dot{x} = Ax + Bu,$$

$$y = Cx$$

$$x = [x \quad \dot{x} \quad i],$$

$$u = e, \quad y = x$$

線形モデル: Model[B1]

$$A = \begin{bmatrix} 0 & 1 & 0 \\ \frac{2kI^2}{m(X+x_0)^3} & 0 & -\frac{2kI}{m(X+x_0)^2} \\ 0 & \frac{2kI}{(X+x_0)\{2k+L_0(X+x_0)\}} & -\frac{R(X+x_0)}{2k+L_0(X+x_0)} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{X+x_0}{2k+L_0(X+x_0)} \end{bmatrix} \quad C = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T$$

線形モデル: Model[B2]

$$A = \begin{bmatrix} 0 & 1 & 0 \\ \frac{2kI^2}{m(X+x_0)^3} \Delta y & 0 & -\frac{2kI}{m(X+x_0)^2} \Delta y \\ 0 & \frac{2kI(1-2\Delta x + \Delta i)}{(X+x_0)\{2k(1-\Delta x) + L_0(X+x_0)\}} & -\frac{R(X+x_0)}{2k(1-\Delta x) + L_0(X+x_0)} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{X+x_0}{L_0(X+x_0) + 2k(1-\Delta x)} \end{bmatrix} \quad C = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T$$

ノミナルモデル


* MATLAB 6.5 用

状態空間表現

$$\text{Model}[A1] \quad G_{nom} \begin{cases} \dot{x} = A_{nom}x + B_{nom}u, \\ y = C_{nom}x \end{cases} \quad A = \begin{bmatrix} 0 & 1 & 0 \\ \frac{2kI^2}{m(X+x_0)^3} & 0 & -\frac{2kI}{m(X+x_0)^2} \\ 0 & 0 & -\frac{R}{L} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} \quad C = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T$$

MATLAB program


```
m = 1.75; X = 5e-3; I = 1.06; L = 5.08e-1; R = 2.32e1; k = 2.90e-4;
x0 = -6.41e-4; Q = 5.79e-4; Xinf = -6.41e-4; L0 = 3.75e-1;
A1_A = [ 0                1                0;
         (2*k*I^2)/(m*(X+x0)^3)  0  -(2*k*I)/(m*(X+x0)^2);
         0                0                -R/L];
A1_B = [ 0; 0; 1/L];
A1_C = [1 0 0];
Gnom_ss = ss ( A1_A, A1_B, A1_C, 0 );
Gnom_sys = pck ( A1_A, A1_B, A1_C, 0 );
```


$$A_{nom} = \begin{bmatrix} 0 & 1 & 0 \\ 4496 & 0 & -18.49 \\ 0 & 0 & -45.67 \end{bmatrix} \quad B_{nom} = \begin{bmatrix} 0 \\ 0 \\ 1.969 \end{bmatrix} \quad C_{nom} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T$$

可制御性, 可観測性

MATLAB program

```
CO = ctrb ( A1_A, A1_B );  
CO_rank = rank ( CO )  
OB = obsv ( A1_A, A1_C );  
OB_rank = rank ( OB )
```


$$\begin{aligned} \text{rank}(CO) &= 3 \\ \text{rank}(OB) &= 3 \end{aligned}$$

制御対象は, 可制御, 可観測である

伝達関数表現

$$G_{nom}(s) = C_{nom} (sI - A_{nom})^{-1} B_{nom} = \frac{-K_i}{(Ms^2 - K_x)(Ls + R)}$$

$$G_{nom}(s) = \frac{-32.3564}{(s + 67.05)(\underline{s - 67.05})(s + 45.67)}$$

不安定極

極 $\pm 67.05, -45.67$

3 次の線形時不変系 (LTI システム) であり, 不安定系

MATLAB program

```
zpk ( Gnom_ss )  
pole ( Gnom_ss )  
zero ( Gnom_ss )
```

MATLAB program

```
omega1=logspace(-1,3,150);  
bode ( Gnom_ss, omega1 );  
pzmap ( Gnom_ss )
```

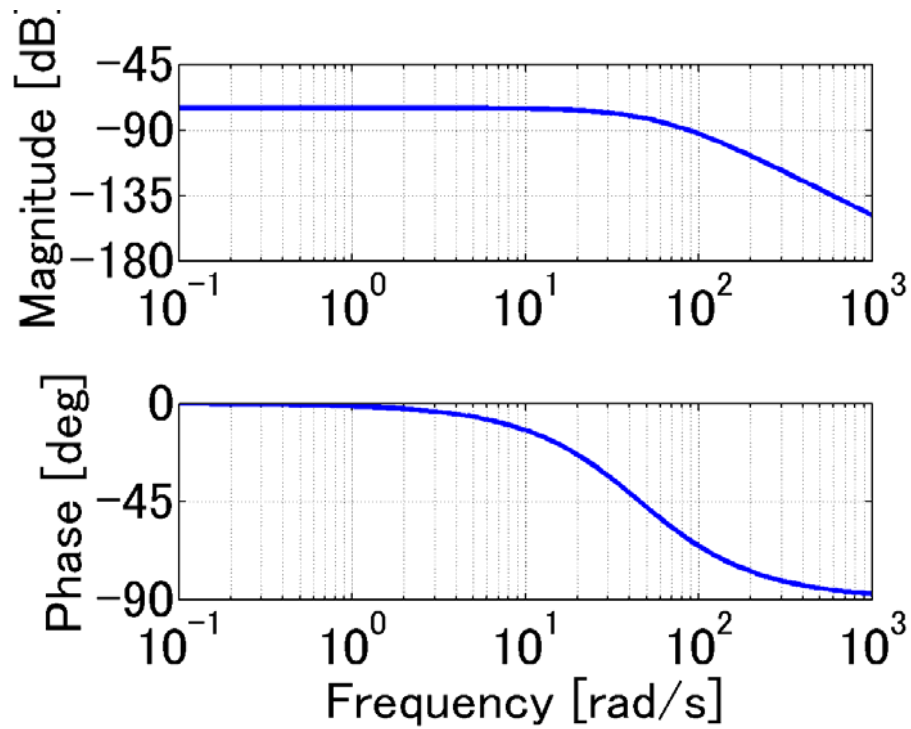


図 プラント $G_{nom}(s)$

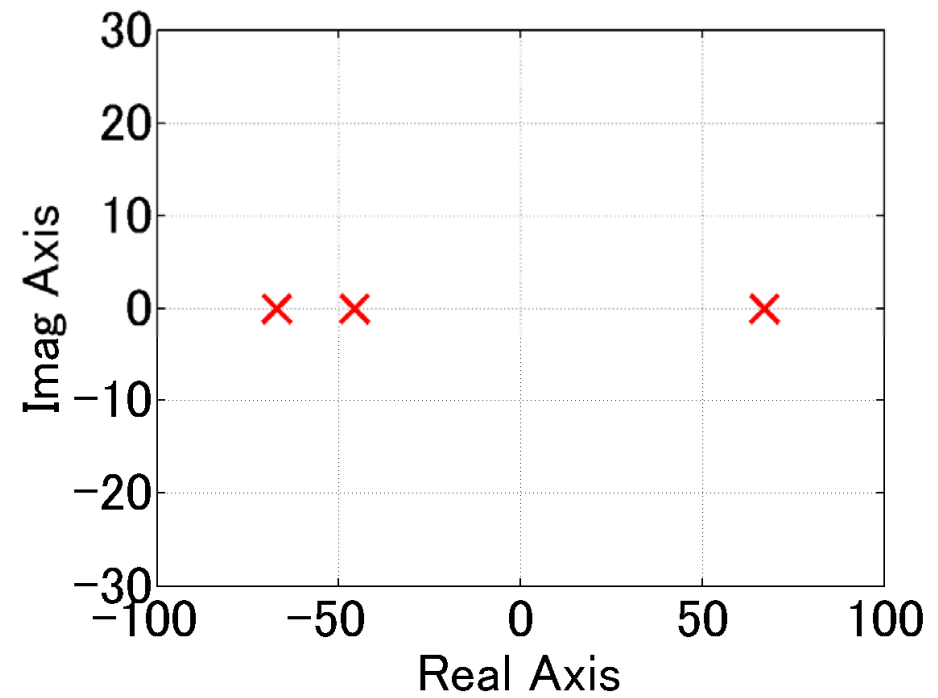


図 プラント $G_{nom}(s)$ の極・零点

[ステップ2] 制御目的から、性能仕様を決める.

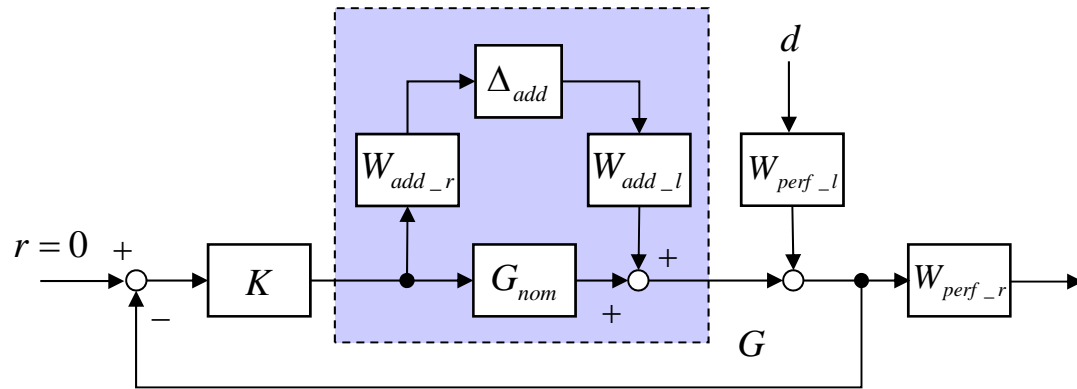
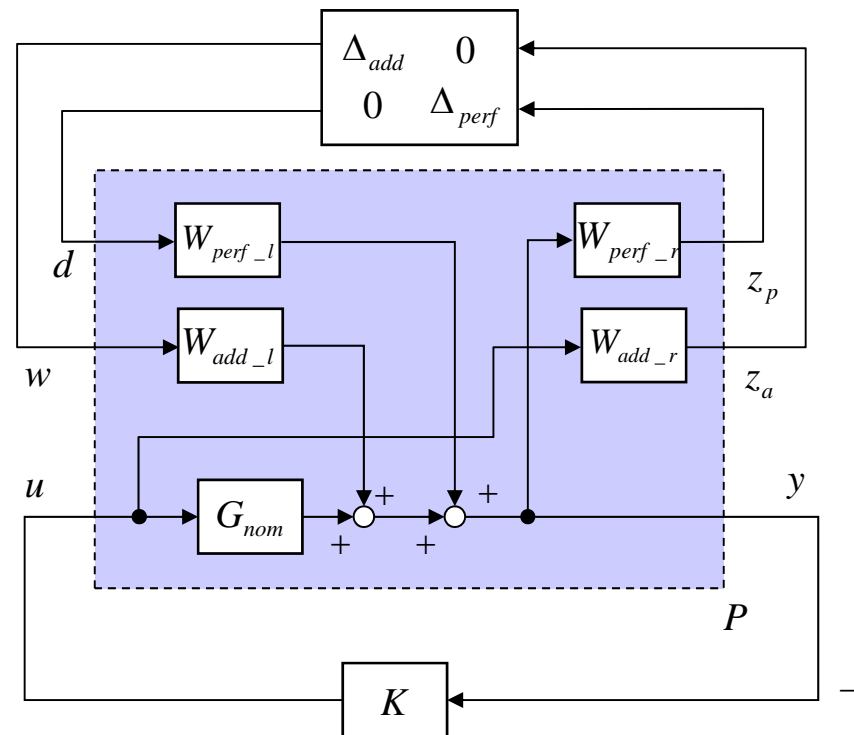


図 フィードバック構造



摂動モデル

24通りの摂動モデル

- | | |
|------------------------------|------------------------------|
| >model 1a : (A1) k_{\max} | >model 1b : (A1) k_{\min} |
| >model 2a : (A1) $x_{0\max}$ | >model 2b : (A1) $x_{0\min}$ |
| >model 3a : (A1) R_{\max} | >model 3b : (A1) R_{\min} |
| >model 4a : (A1) L_{\max} | >model 4b : (A1) L_{\min} |
| >model 5a : (A2) x_{\max} | >model 5b : (A2) x_{\min} |
| >model 6a : (A2) i_{\max} | >model 6b : (A2) i_{\min} |
| >model 7a : (B1) k_{\max} | >model 7b : (B1) k_{\min} |
| >model 8a : (B1) $L_{0\max}$ | >model 8b : (B1) $L_{0\min}$ |
| >model 9a : (B1) $x_{0\max}$ | >model 9b : (B1) $x_{0\min}$ |
| >model 10a : (B1) R_{\max} | >model 10b : (B1) R_{\min} |
| >model 11a : (B2) x_{\max} | >model 11b : (B2) x_{\min} |
| >model 12a : (B2) i_{\max} | >model 12b : (B2) i_{\min} |

cf: model ij : (モデル構造) 摂動モデルパラメータ

$$\Delta_{ij} := G_{ij_pert} - G_{nom} \quad (1 \leq i \leq 12, j = a, b)$$

MATLAB program

```
k_min = 2.53e-4;  
k = k_min;  
A1_A_per = [ 0          1          0 ;  
             (2*k*I^2)/(m*(X+x0)^3)  0  -(2*k*I)/(m*(X+x0)^2);  
             0          0          -R/L];  
Gper_ss = ss ( A1_A_per, A1_B, A1_C, 0 );  
model_1a_ss = Gper_ss - Gnom_ss;  
Bodemag ( model_1a_ss , omega1 )
```

•
•
•
•
•

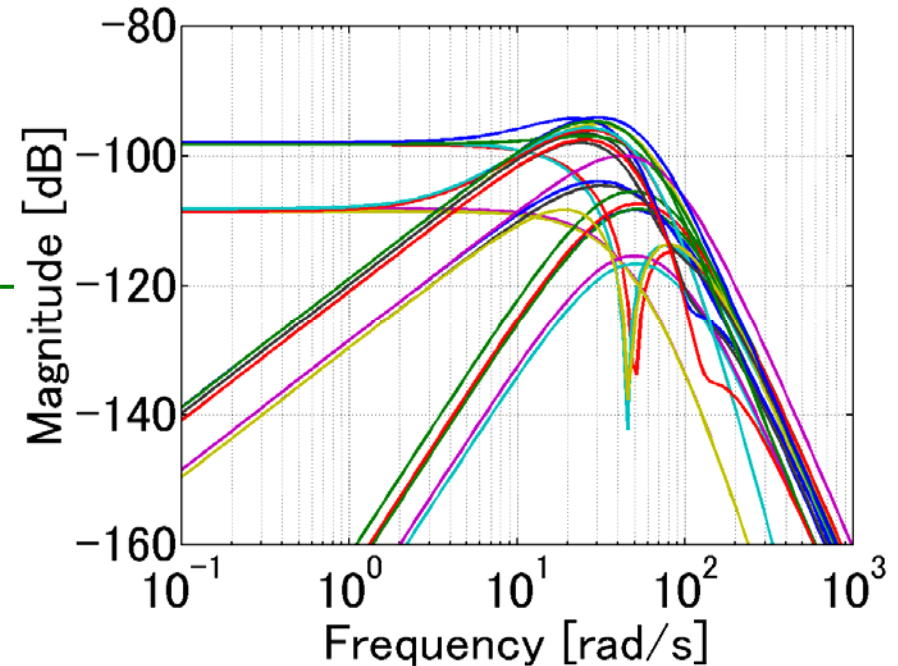


図 加法的な摂動 $|\Delta_{ij}(j\omega)|$

周波数重み $W_{add}(s)$

$$G = \{G_{nom}(s) + \Delta_{add}(s)W_{add}(s) : \|\Delta_{add}\|_{\infty} \leq 1\}$$

$$W_{add}(s) = 1.4 \times 10^{-5} \times \frac{(1+s/8)(1+s/170)(1+s/420)}{(1+s/30)(1+s/35)(1+s/38)}$$

$$W_{add}(s) = W_{add_l}(s) \times W_{add_r}(s)$$

$$W_{add_r}(s) = 1.4 \times \frac{(1+s/8)(1+s/170)(1+s/420)}{(1+s/30)(1+s/35)(1+s/38)}$$

$$W_{add_l}(s) = 1.0 \times 10^{-5}$$

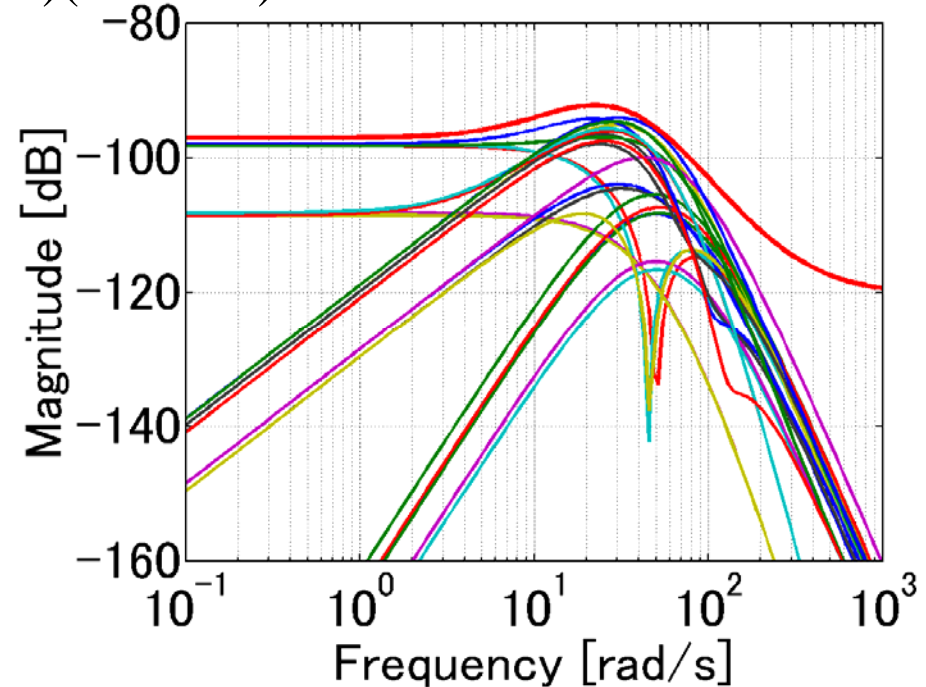


図 重み関数 $W_{add}(s)$

MATLAB program

```
w1=8; w2=170; w3=420;
```

```
w4=30; w5=35; w6=38;
```

```
Waddr_tf = tf ( [1/w1 1], [1/w4 1] ) * tf ( [1/w2 1], [1/w5 1] ) * ...  
            tf ( [1/w3 1], [1/w6 1] ) * 1.4;
```

```
Waddl_tf = tf ( 1e-5, 1 );
```

```
Wadd_tf = Waddl_tf*Waddr_tf;
```

```
Waddr_sys = mmult ( nd2sys ( [1/w1 1], [1/w4 1] ), nd2sys ( [1/w2 1], [1/w5 1] ), ...  
                  nd2sys ( [1/w3 1], [1/w6 1] ), 1.4 );
```

```
Waddl_sys = 1e-5;
```

```
Wadd_sys = mmult ( Waddr_sys, Waddl_sys );
```

```
unc = ultidyn ( 'unc' , [1 1] );
```

```
Wadd_unc = Waddl_tf * unc * Waddr_tf;
```

```
Bodemag ( Wadd_tf, omega1 )
```

*



MATLAB 7 用

MATLAB コマンド :ultidyn

不確かさを表す LTI モデルを構成

$H = \text{ultidyn}(\text{'Name'}, \text{iosize}, \text{'Property1'}, \text{Value1}, \dots);$

入力引数

- Name : 変数名
- iosize : 入出力の大きさ
- Property :
- Value :

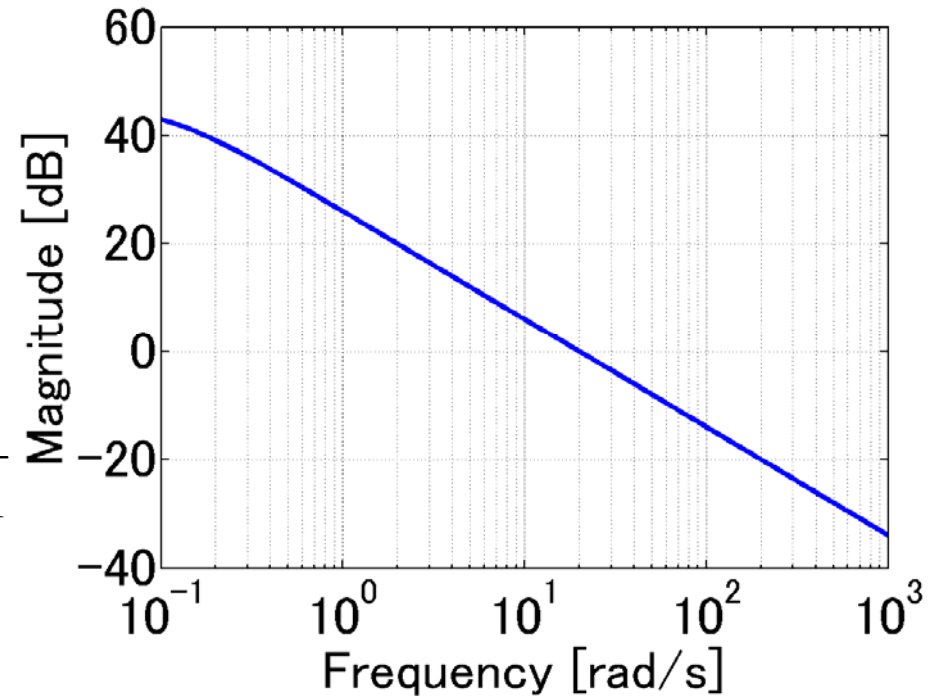
Property	
Type	Gainbounded, positivereal
Bound	有界ノルムまたは最小実現
SampleStateDim	
autosimplify	表現の低減化

周波数重み $W_{perf}(s)$

$$W_{perf}(s) = \frac{200}{1 + s/0.1}$$

$$W_{perf} = W_{perf_l} \times W_{perf_r}$$

$$W_{perf_l} = 1.0 \times 10^{-5}, \quad W_{perf_r} = \frac{2.0 \times 10^7}{1 + s/0.1}$$



MATLAB program

```
Wperfl_tf = tf ( 1e-5, 1 );  
Wperfr_tf = tf ( [200], [10 1] ) * ( 1/Wperfl_tf );  
Wperf_tf = Wperfr_tf * Wperfl_tf;  
  
Wperfl_sys = 1e-5;  
Wperfr_sys = nd2sys([200],[10 1],1/Wperfl_sys);  
Wperf_sys = mmult(Wperfr_sys,Wperfl_sys);  
  
Bodemag ( Wperf_tf , omega1 )
```

図 重み関数 $W_{perf}(s)$

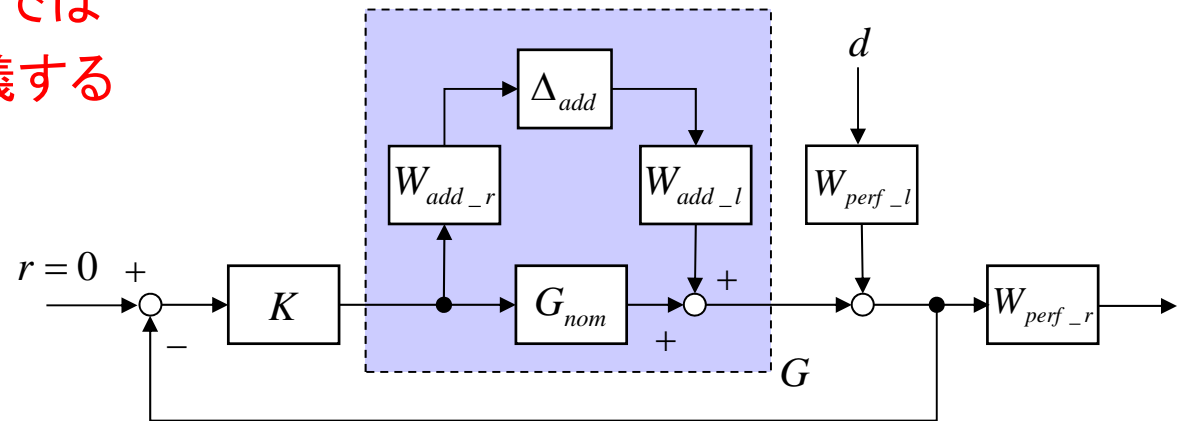
[ステップ3] 性能仕様を満たすように、コントローラを設計する。

一般化プラント (Robust Control Toolbox ver. 3)

Robust Control Toolbox ver. 3 では
不確かさのあるブロックを定義する
ことができる。

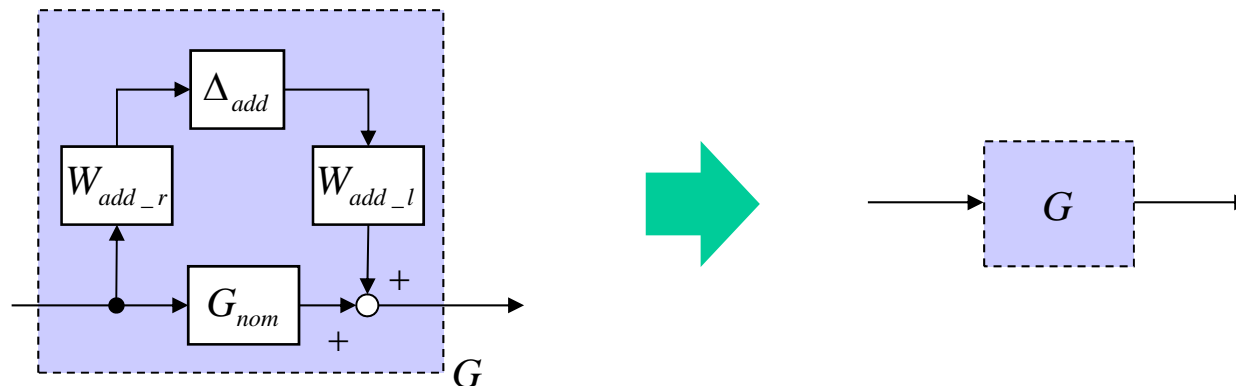
Δ_{add} を定義

```
unc = ultidyn('unc', [1 1]);
```



G を定義

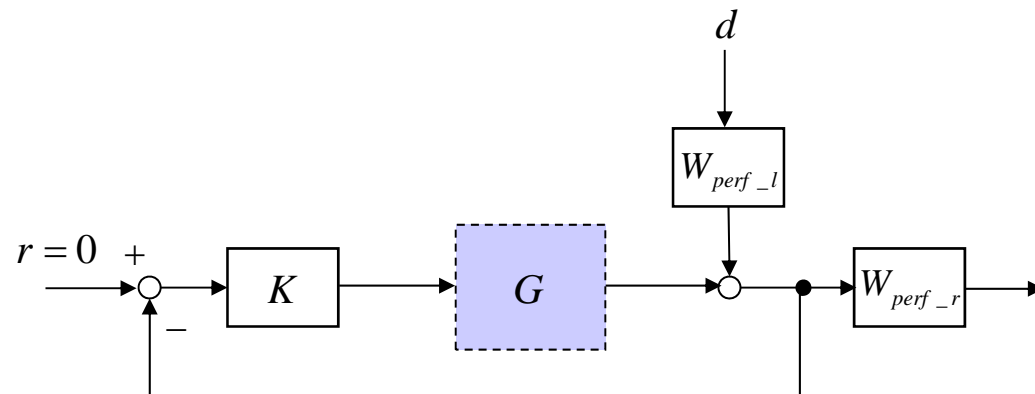
$$G = G_{nom_ss} + W_{add_l_tf} * unc * W_{add_r_tf};$$



MATLAB program

Robust Control Toolbox ver. 3 用

```
systemnames = 'Gnom_ss Wperfr_tf Wperfl_tf Wadd_unc';  
inputvar = '[ dist; control ]';  
outputvar = '[ Wperfr_tf; -Gnom_ss - Wperfl_tf - Wadd_unc ]';  
input_to_Gnom_ss = '[ control ]';  
input_to_Wadd_unc = '[ control ]';  
input_to_Wperfr_tf = '[ Gnom_ss + Wperfl_tf + Wadd_unc ]';  
input_to_Wperfl_tf = '[ dist ]';  
General_P = sysic;
```



MATLAB program

μ - Analysis and Synthesis Toolbox 用

```
systemnames = 'Gnom_sys Wperfl_sys Wperfr_sys Waddr_sys Waddl_sys';  
inputvar = '[ dist; pertin; control ]';  
outputvar = '[ Wperfr_sys; Waddr_sys; - Gnom_sys - Wperfl_sys - Waddl_sys]';  
input_to_Gnom_sys = '[ control ]';  
input_to_Waddr_sys = '[ control ]';  
input_to_Wperfr_sys =  
'[ Gnom_sys + Wperfl_sys + Waddl_sys ]';  
input_to_Wperfl_sys = '[ dist ]';  
input_to_Waddl_sys = '[ pertin ]';  
sysoutname = 'General_P';  
cleanupsysic = 'yes';  
sysic;
```

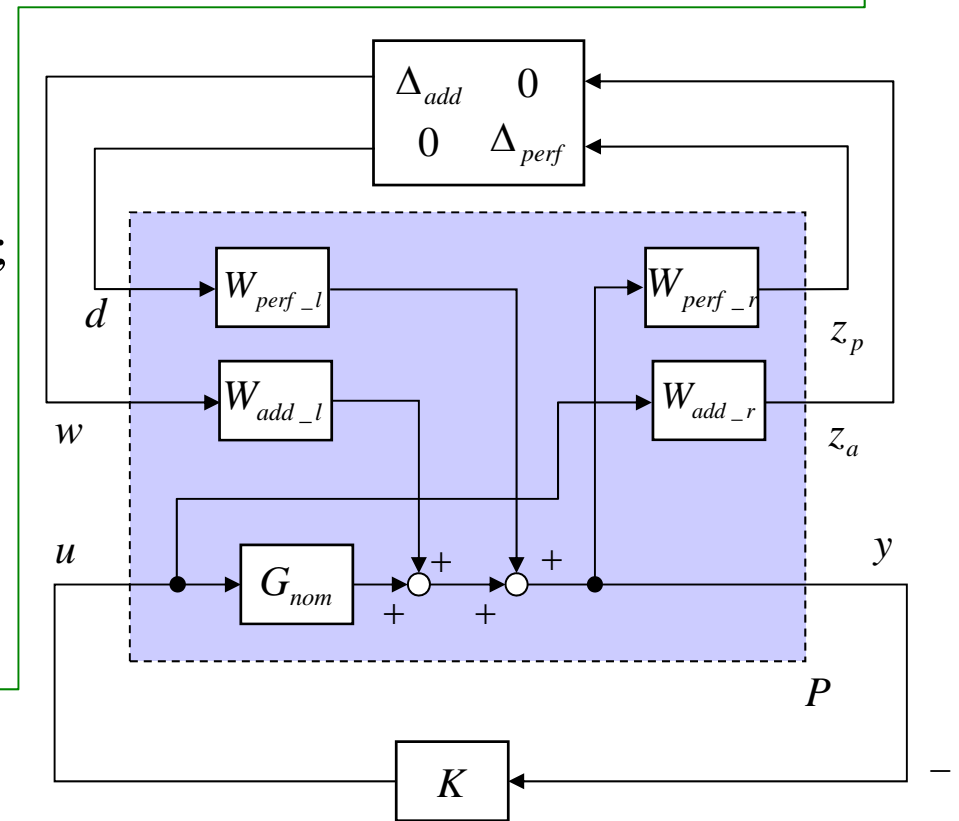


図 LFT 表現

MATLAB コマンド : `dksyn` Robust Control Toolbox ver. 3

`[k, clp, bnd] = dksyn (G, nmeas, ncont, opt);`

入力引数

- G : 一般化プラント
- nmeas : センサ観測数
- ncont : 制御入力数
- opt : オプション設定

出力引数

- k : コントローラ
- clp : 閉ループ系
- bnd : μ の上界

MATLAB コマンド : `dkitopt`

MATLAB コマンド ‘`dksyn`’ のオプション設定

引数	意味
AutoIter	自動なら ‘on’, 手動なら ‘off’
FrequencyVector	周波数の範囲
⋮	⋮
⋮	⋮

MATLAB コマンド : dkit

D-Kイテレーションを自動的に実行

入力引数

セットアップファイルにより定義

- NOMINAL_DK : 一般化プラント
- NMEAS_DK : センサ測定数
- NCONT_DK : 制御入力数
- BLK_DK : ブロック構造
- GMIN_DK : 最初の H_∞ 制御設計の下界
- GMAX_DK : 最初の H_∞ 制御設計の上界

出力引数

- bnds_dk(i) : μ の上界と下界
- dl_dk(i) : 左状態空間 D 行列
- dr_dk(i) : 右状態空間 D 行列
- gf_dk(i) : 閉ループシステムの H_∞ ノルム
- k_dk(i) : コントローラ

* i はイテレーションの数を表す

MATLAB program

Robust Control Toolbox ver. 3 用

```
options = dkitopt ( 'FrequencyVector', logspace(-1,3,60), 'AutoIter', 'off')  
[ Kdk, CLdk, gdk, dkinfo ] = dksyn(General_P,1,1,options);
```

MATLAB program

μ - Analysis and Synthesis Toolbox 用

```
DK_DEF_NAME='himat_dk1';  
dkit
```

MATLAB program (himat_dk1.m)

```
NOMINAL_DK = General_P;  
NMEAS_DK = 1;  
NCONT_DK = 1;  
BLK_DK = [1 1;1 1];  
OMEGA_DK = logspace(-1,3,60);  
GMAX_DK = 1.3;  
GMIN_DK = 1.3;  
AUTOINFO_DK = [1 2 1];  
自動にする場合
```

MATLAB コマンド `hinfsyn` の違い

γ の最小化について

μ -Analysis and
Synthesis Toolbox

2分法

$$\gamma_1 = \gamma_{\max}$$

↓ 解ける

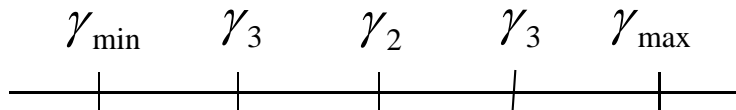
$$\gamma_2 = (\gamma_{\max} - \gamma_{\min}) / 2$$

解ける

解けない

$$\gamma_3 = (\gamma_2 - \gamma_{\min}) / 2$$

$$\gamma_3 = (\gamma_{\max} - \gamma_2) / 2$$



Robust Control Toolbox ver. 3

PK_{H_2} : H_2 コントローラと
システムの閉ループ系

$$\gamma_1 = \min \{ \gamma_{\max}, \| PK_{H_2} \|_{\infty} \}$$

2分法

$$\gamma_2 = (\gamma_1 - \gamma_{\min}) / 2$$

解ける

解けない

MATLAB コマンド `dkdyn` と `dkit` の違い

	<code>dkdyn</code>	<code>dkit</code>
Δ の構造	' <code>ultidyn</code> ' から自動的に構造を設定	手動
$\gamma_{\min}, \gamma_{\max}$	自動	1回目の D-K iteration を行うときの $\gamma_{\min}, \gamma_{\max}$ を手動で設定。2回目以降は自動だが、手動で行うこともできる
D-K iteration 1 回目の γ の許容範囲	$\gamma_{\max} = 100, \gamma_{\min} = 0$ $\text{tol} = 0.01 * (\gamma_{\max} - \gamma_{\min})$	$\text{tol} = 0.01 * (\gamma_{\max} - \gamma_{\min})$

Robust Control Toolbox ver. 3

Iteration Number: 1 D-K イテレーションの1回目

Information about the Interconnection Structure IC:
State-space model with 3 outputs, 3 inputs, and 7 states.

Singular Value plot of closed-loop system in GRAPHICS window

Make sure that chosen Frequency range is appropriate

Do you want to modify OMEGA_DK? (y/n): **n**

考慮する周波数領域を変更する(y) or しない(n)

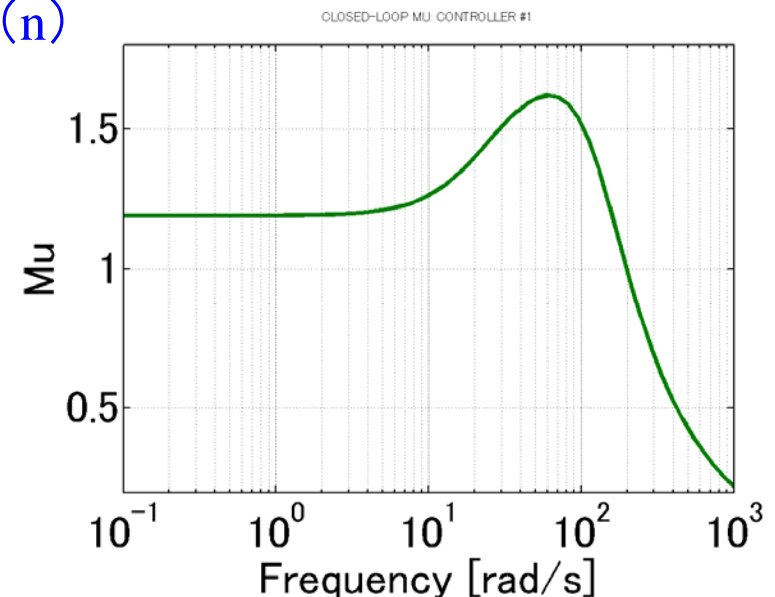
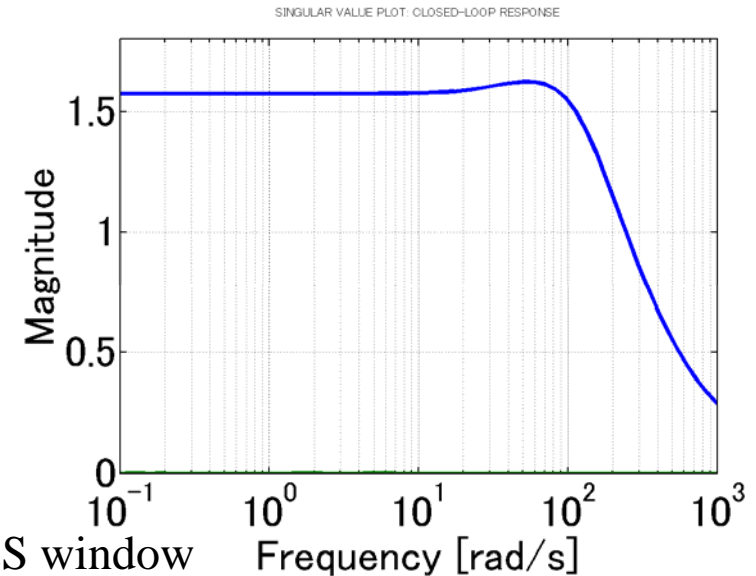
Calculating MU of closed-loop system:

points completed (of 60) ... 60

MU plots for control design:

Strike any key to continue

Enter



Iteration Summary

Iteration # 1
Controller Order 7
Total D-Scale Order 0
Gamma Acheived 1.988
Peak mu-Value 1.619

Another D-K iteration? (y/n): **y**

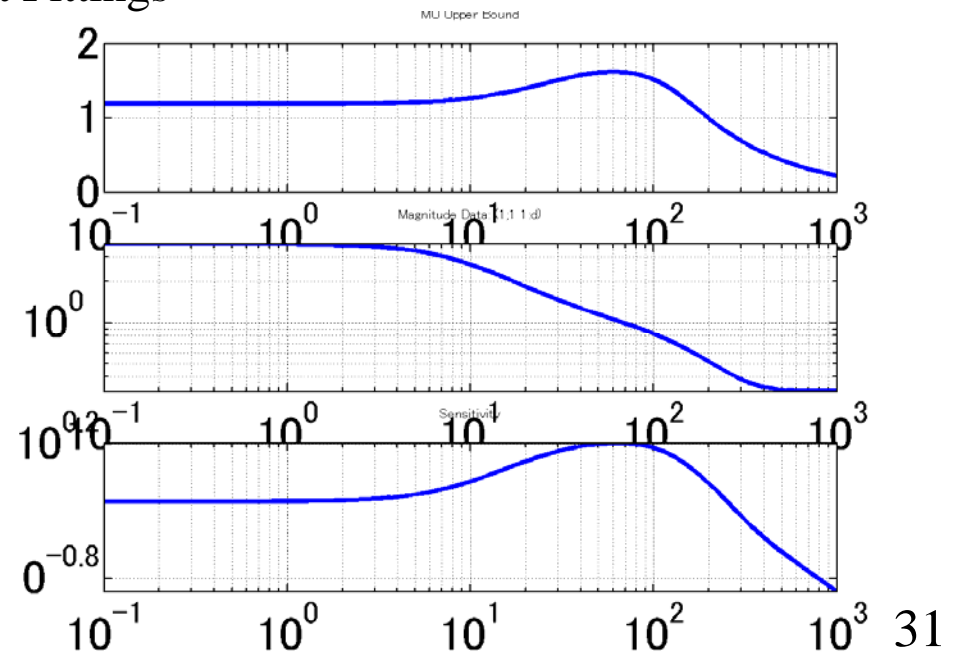
2回目のDKイテレーションを
行う(y) or 行わない(n)

Enter Choice (return for list): Enter
Choices:

- nd Move to Next D-scaling
- nb Move to Next D-Block
- i Increment Fit Order
- d Decrement Fit Order
- apf Auto-PreFit
- mx 3 Change Max-Order to 3
- at 1.01 Change Auto-PreFit Tol to 1.01
- 0 Fit with zeroth order
- 2 Fit with second order
- n Fit with n'th order
- e Exit with Current Fittings
- s See Status

Enter Choice (return for list): **apf**

自動的にフィッティングを行う



Starting Auto-PreFit...

Auto Fit in Progress

Block 1, MaxOrder=5, Order = 0

1

Block 2, MaxOrder=5, Order = 0

Done

Enter Choice (return for list): **e**

次に進む

Iteration Number: 2

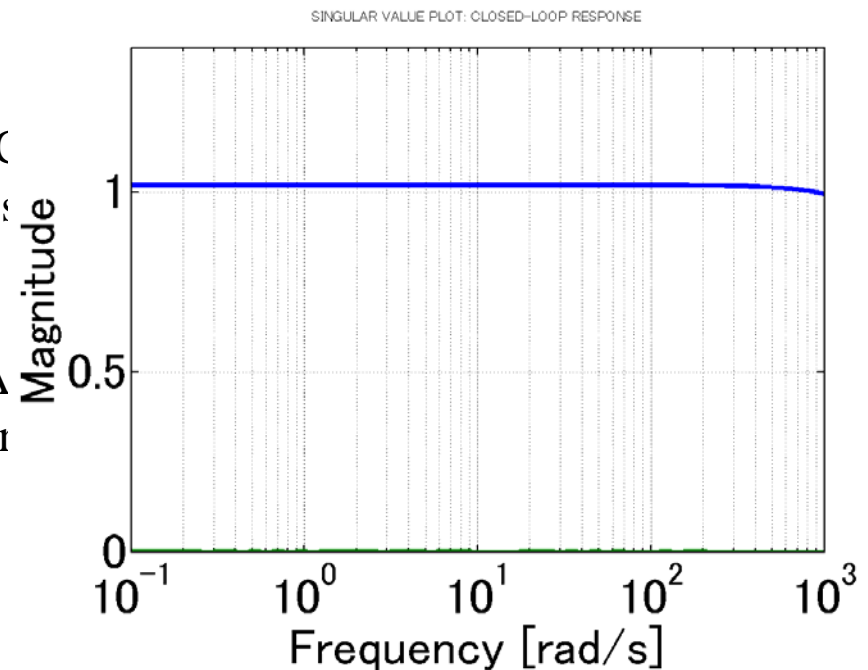
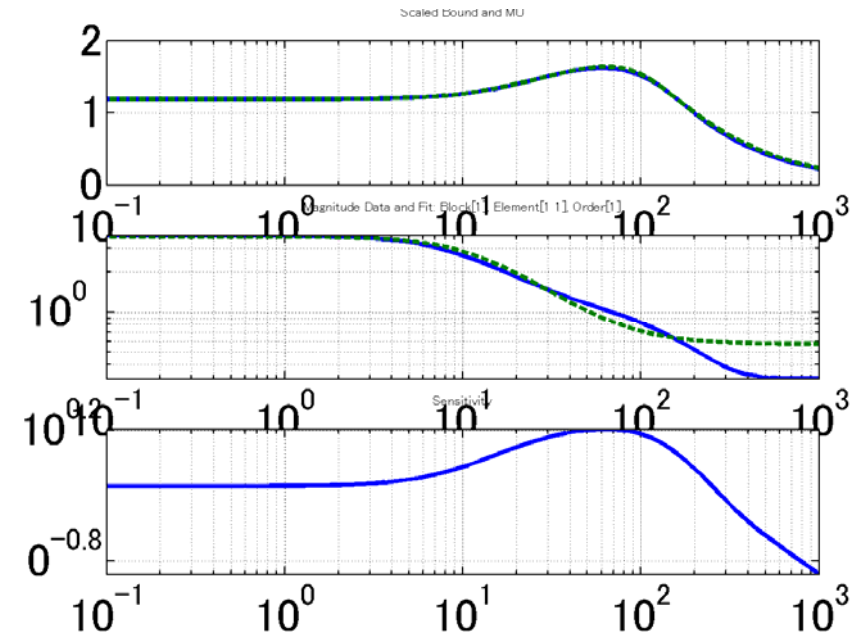
Information about the Interconnection Structure IC
State-space model with 3 outputs, 3 inputs, and 9 s

Singular Value plot of closed-loop system in GRA

Make sure that chosen Frequency range is appropriate

Do you want to modify OMEGA_DK? (y/n): **n**

考慮する周波数領域を変更する(y) or
しない(n)



Calculating MU of closed-loop system:
points completed (of 60) ... 60

MU plots for control design:

Strike any key to continue

Enter

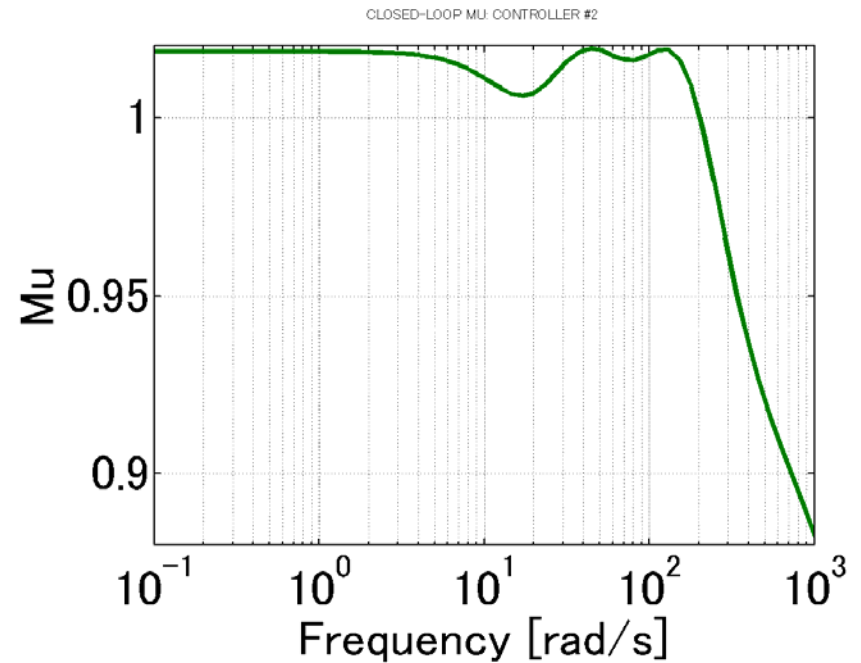
Iteration Summary

Iteration #	1	2
Controller Order	7	9
Total D-Scale Order	0	2
Gamma Acheived	1.988	1.020
Peak mu-Value	1.619	1.019

Another D-K iteration? (y/n): **n**

Next MU iteration number: 3

終了

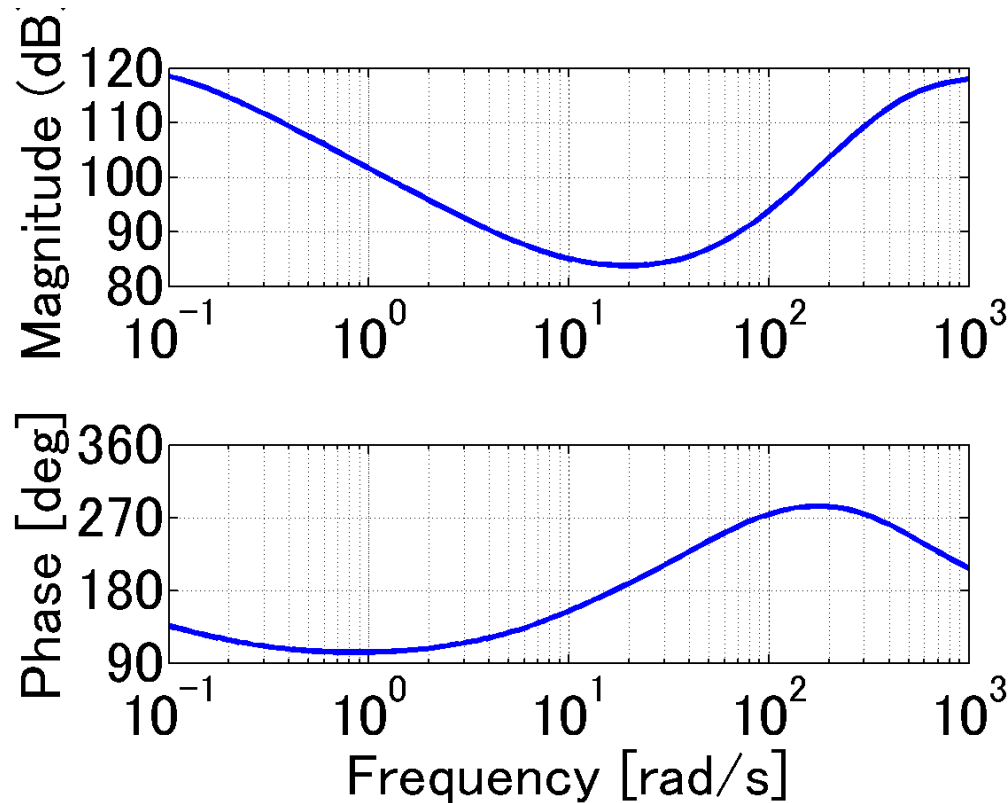


コントローラ $K_{dksyn}(s)$

$$K_{dksyn}(s) = \frac{-4.03 \times 10^9 (s + 10.24)(s + 11.96)(s + 30)(s + 35)(s + 38)}{(s + 0.1)(s + 13.36)(s + 24.74)(s + 37.40 \pm j0.74)} \times \frac{(s + 45.67)(s + 67.05)(s + 76.25)}{(s + 76.78)(s + 410.9 \pm j268.1)(s + 4454)}$$

極

- 0.1
- 13.36
- 24.74
- 37.40 ± j0.74
- 76.78
- 410.9 ± j268.1,
- 4454



安定

図 コントローラ $K_2(s)$

MATLAB program

```
[Kdk_a,Kdk_b,Kdk_c,Kdk_d] = ssdata(Kdk);  
Kdk_ss = ss ( Kdk_a, Kdk_b, Kdk_c, Kdk_d );
```

```
[k_dk2_a, k_dk2_b, k_dk2_c, k_dk2_d] = unpk ( k_dk2 );  
Kdk_ss = ss ( k_dk2_a, k_dk2_b, k_dk2_c, k_dk2_d );
```

```
zpk ( Kdk_ss )
```

```
[Kdk_pole,Kdk_zero] = pzmap ( Kdk_ss )
```

```
bode (Kdk_ss,omega1);
```

開ループ伝達関数

MATLAB program

```
L_ss = Gnom_ss*Kdk_ss;  
bode(L_ss,omega1);  
  
[Gm,Pm,Wpc,Wgc] = margin(L_ss);  
Gm=20*log10(Gm)
```

ゲイン余裕 5.95 [dB]

位相余裕 30.0 [deg]

ゲイン交差周波数 116.4 [rad/sec]

位相交差周波数 32.4 [rad/sec]

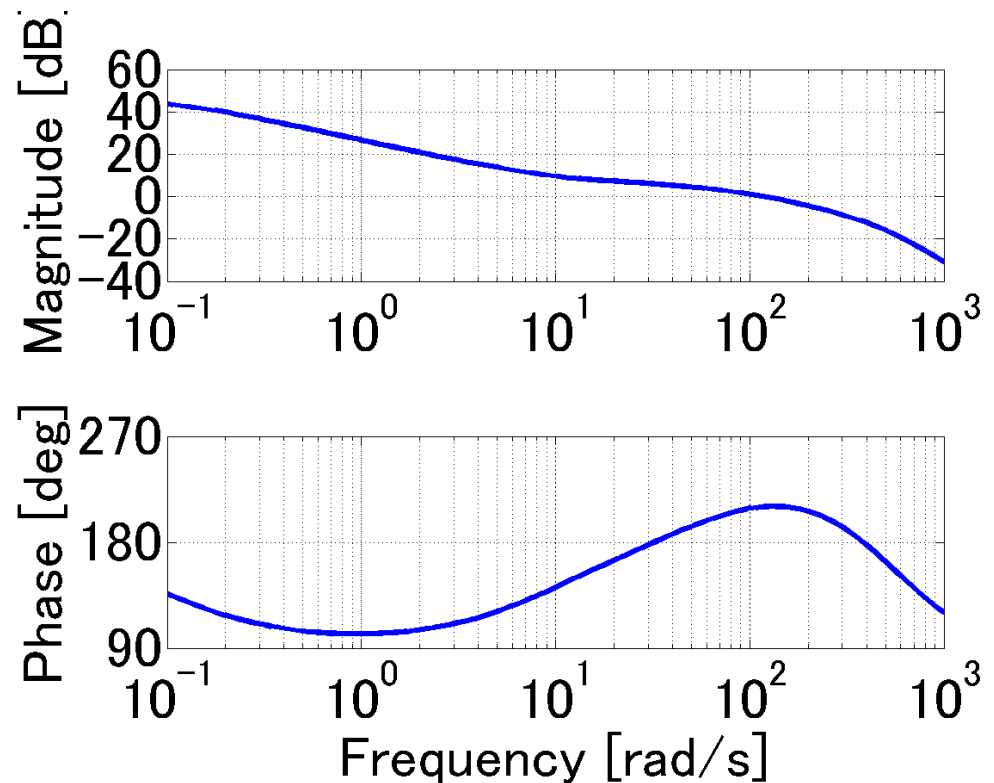


図 開ループ伝達関数

MATLAB program

```
nyquist ( L_ss )
```

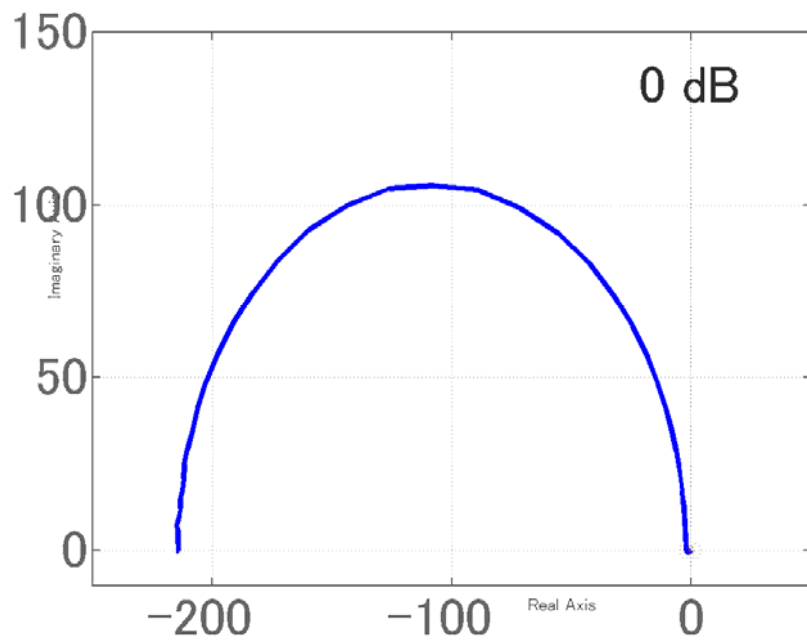


図 ベクトル軌跡

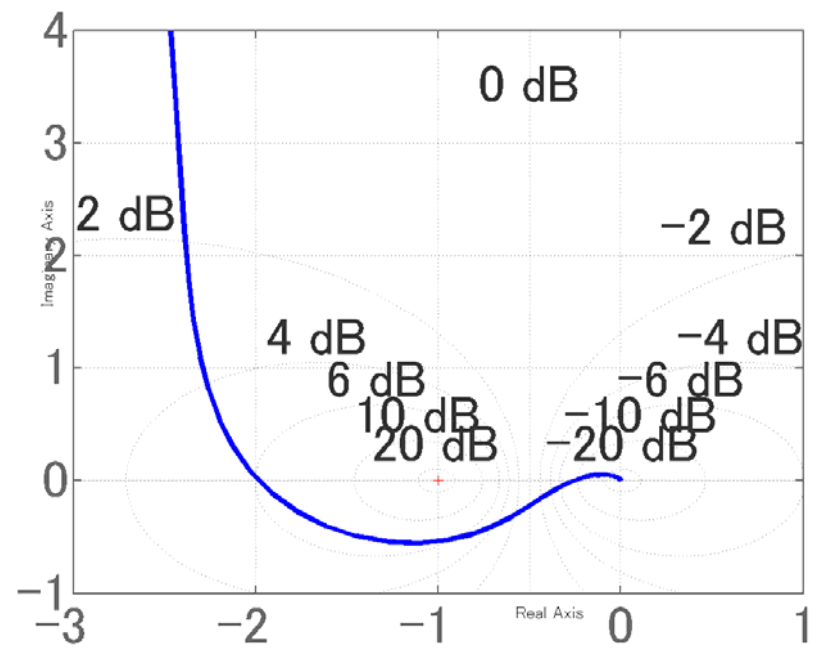
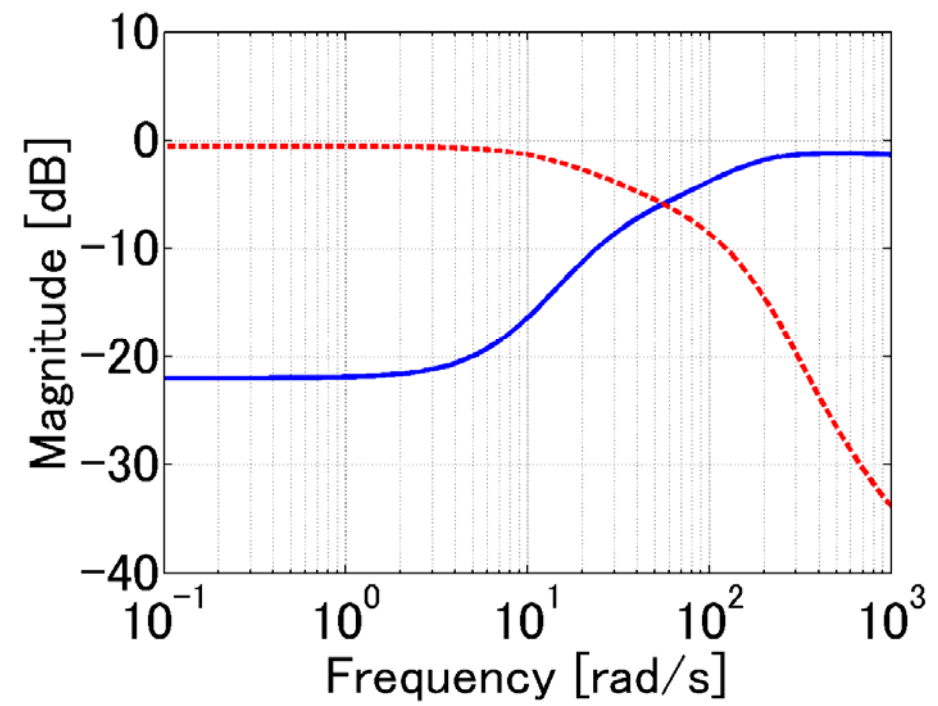
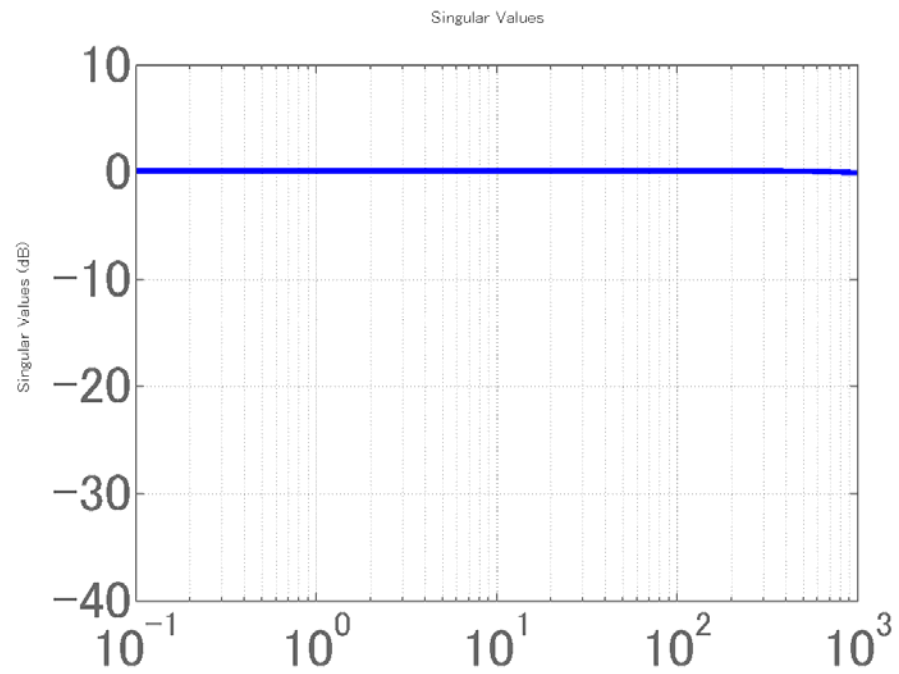


図 ベクトル軌跡

閉ループ系の周波数特性



MATLAB program

```
lft_General_P = lftdata ( General_P );  
DGD_ss = dkinfo{2}.DL * lft_General_P / dkinfo{2}.DR;
```

```
DGD_sys = mmult ( dl_dk2,General_P,minv(dr_dk2));  
[DGD_a,DGD_b,DGD_c,DGD_d] = unpck(DGD_sys);  
DGD_ss = ss(DGD_a,DGD_b,DGD_c,DGD_d);
```

```
Cloop_ss = lft(DGD_ss,Kdk_ss);
```

```
sigma(Cloop_ss,omega1)
```

```
bode(Cloop_ss(1,1),omega1);
```

```
bode(Cloop_ss(2,2),omega1);
```

閉ループ系の特性

極 $-14.19 \pm j2.38$, -39.48 , $-43.38 \pm j5.43$, -45.67 ,
 $-67.05 \pm j0.001$, $-150.5 \pm j69.66$, -424.0 , -4452

MATLAB program

```
T = feedback(L_ss,1);  
pole(T)  
zero(T)  
Pzmap (T)
```

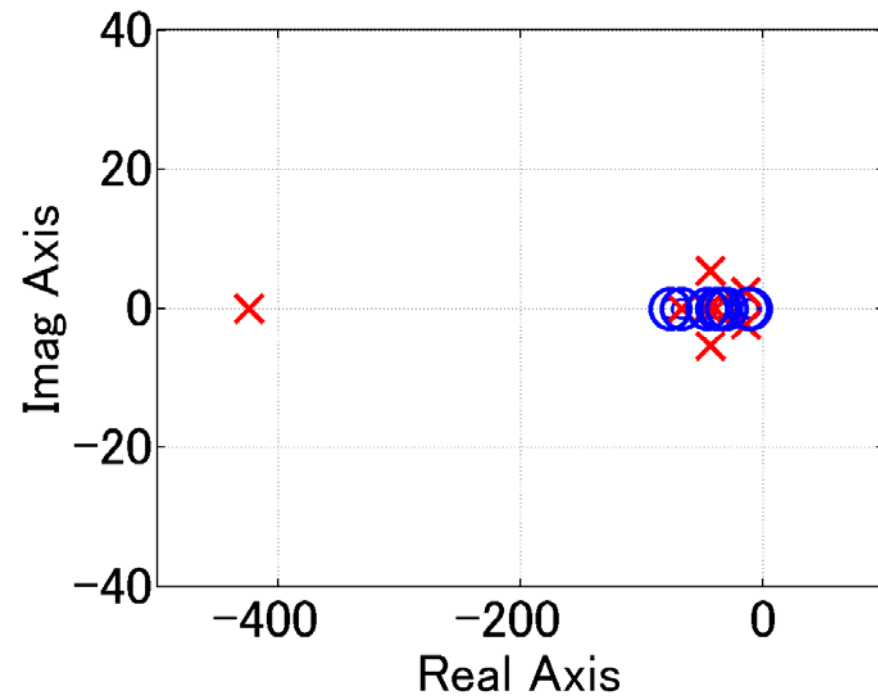
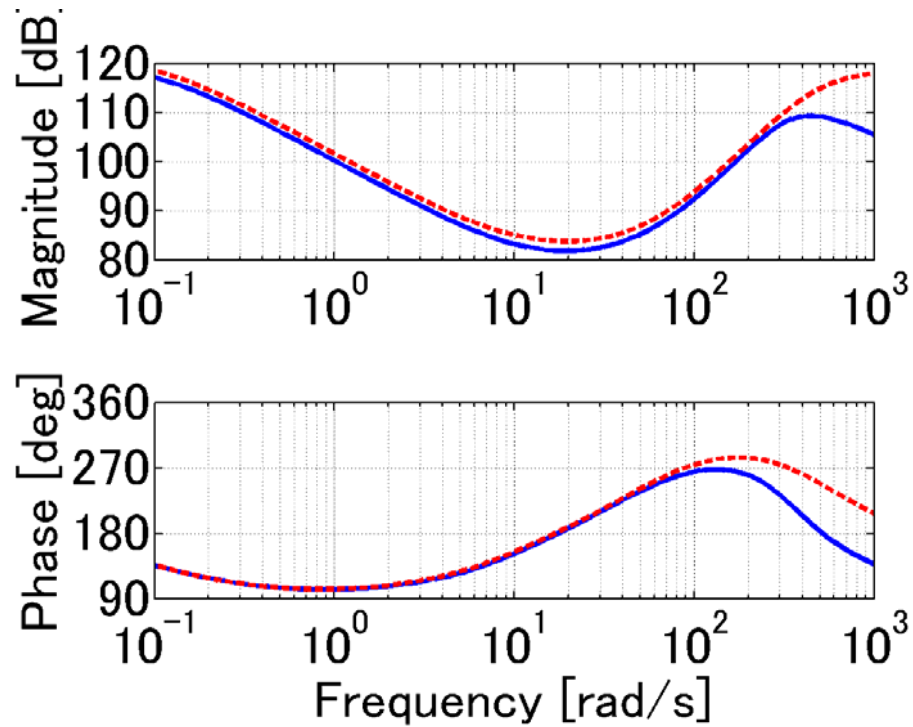


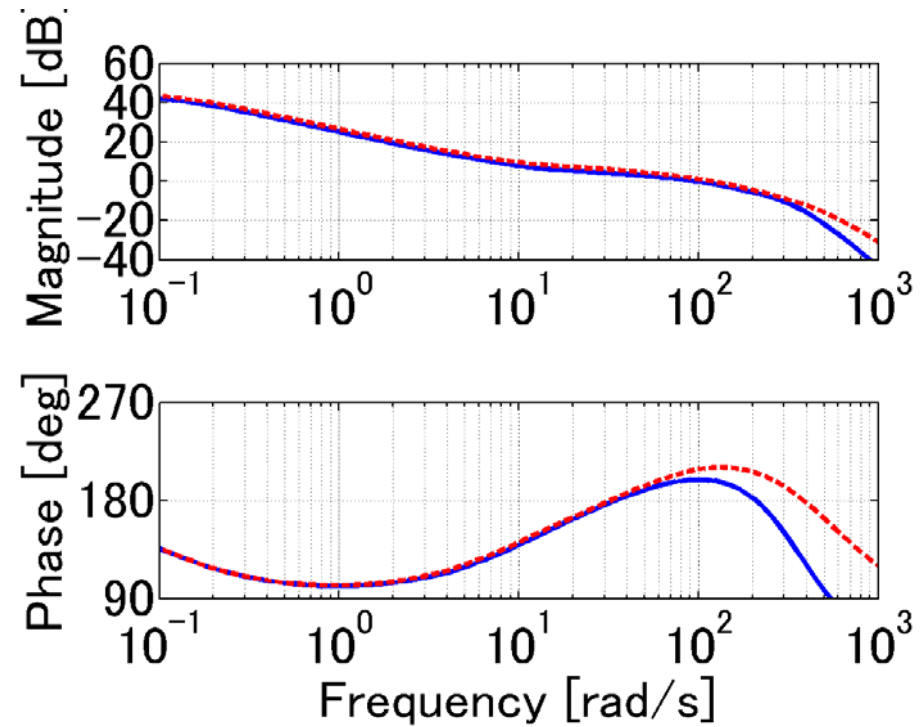
図 閉ループ系の極・零点

D-Kイテレーションによる変化

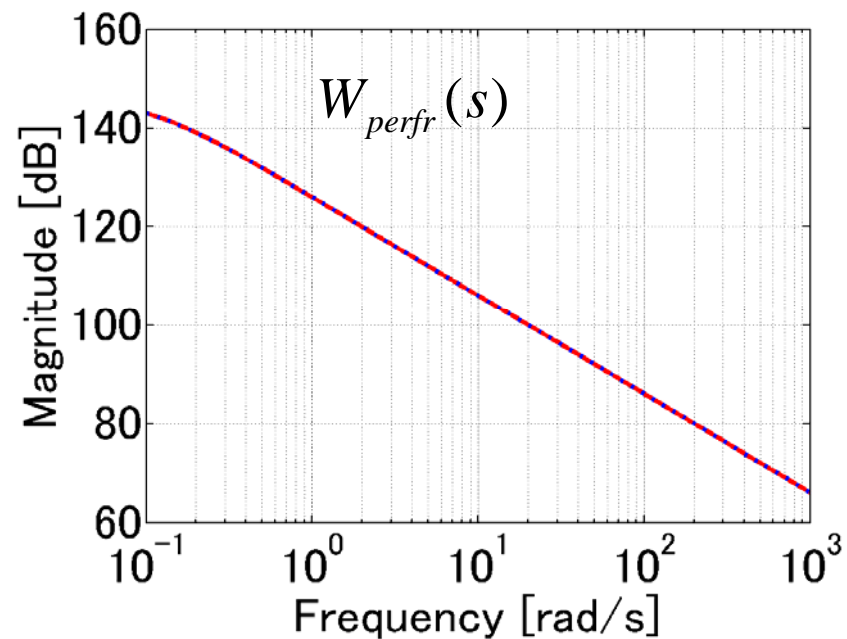
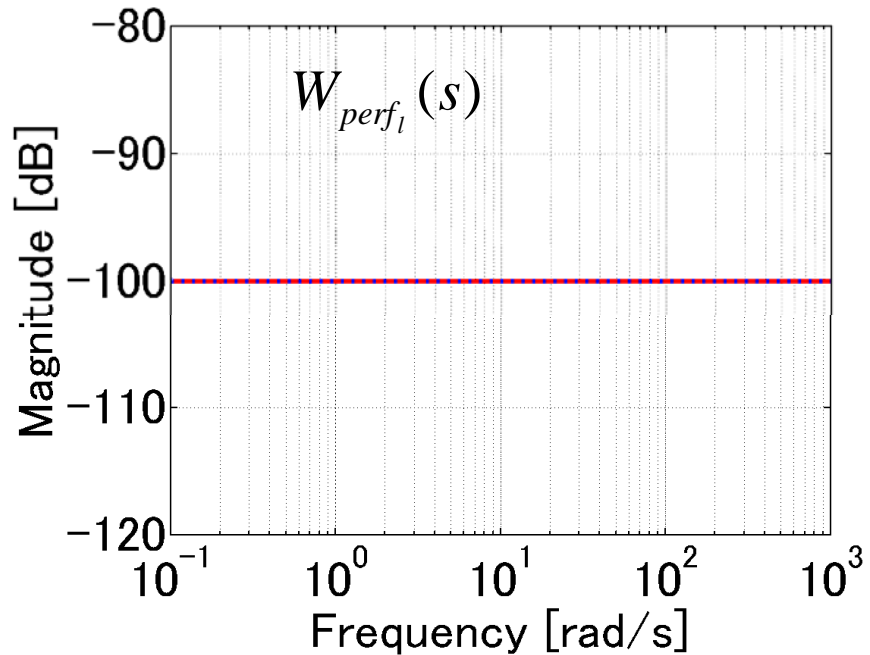
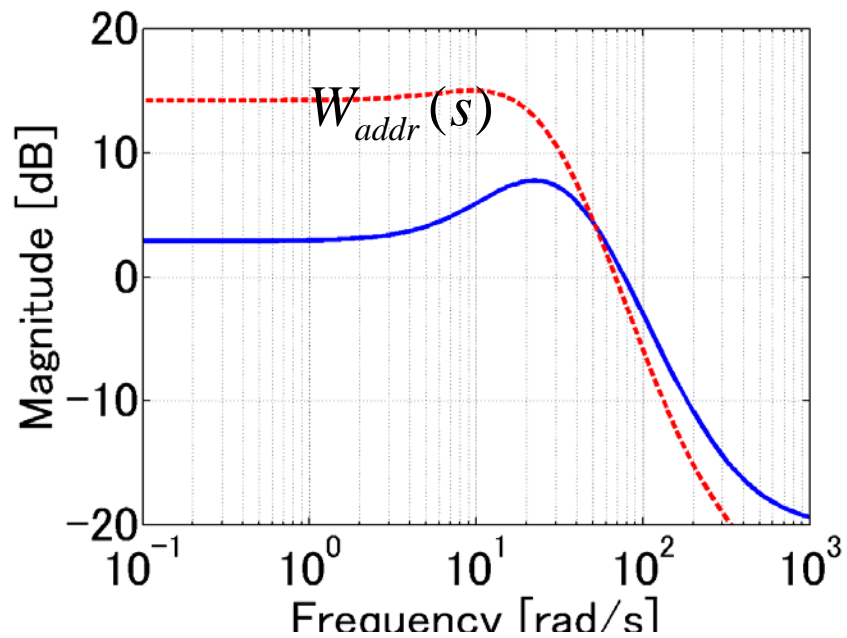
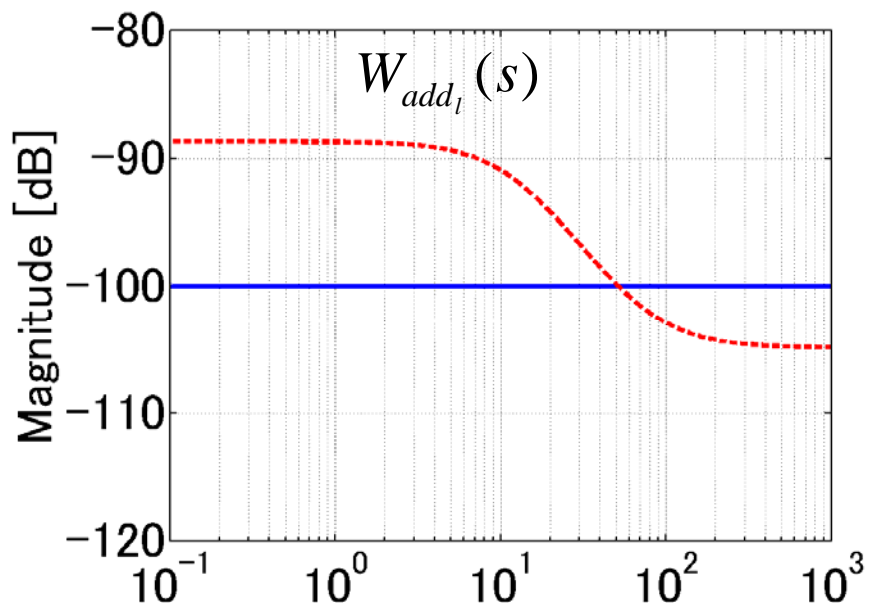
コントローラ



開ループ伝達関数



重み関数の変化



[ステップ4] シミュレーションにより, 設計された制御系を評価する.
必要ならば, 以上のステップを繰り返し設計をやり直す.

[ステップ5] コントローラを実装しハードウェアを用いてテストする.

実験結果

外乱応答 -17.15 [N] に相当する電圧

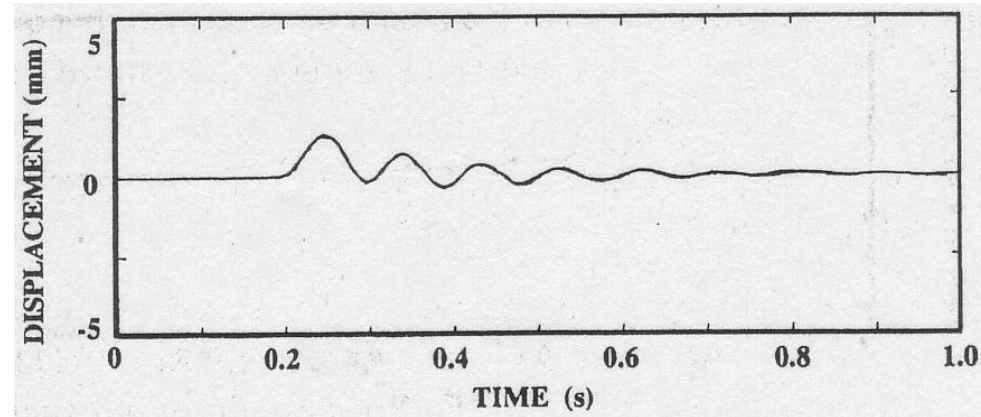


図 ステップ状外乱に対する時間応答(コントローラ K_1)

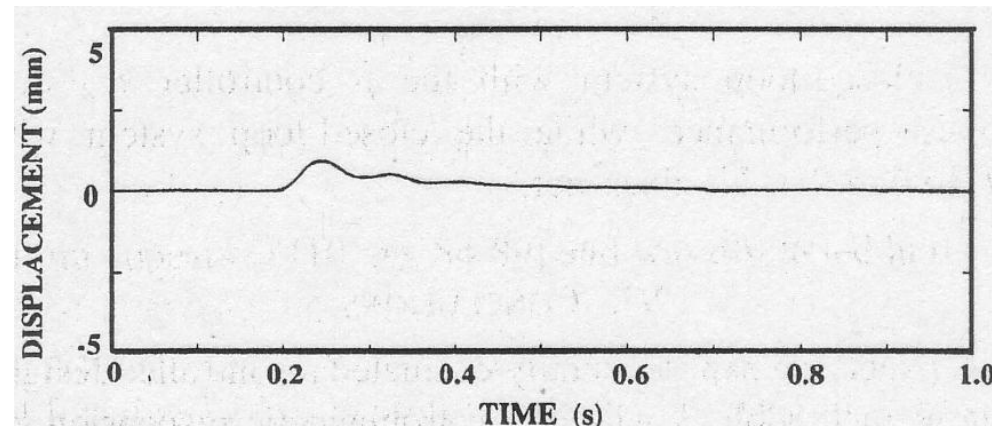


図 ステップ状外乱に対する時間応答(コントローラ K_2)

実験結果

外乱応答 -34.30 [N] に相当する電圧

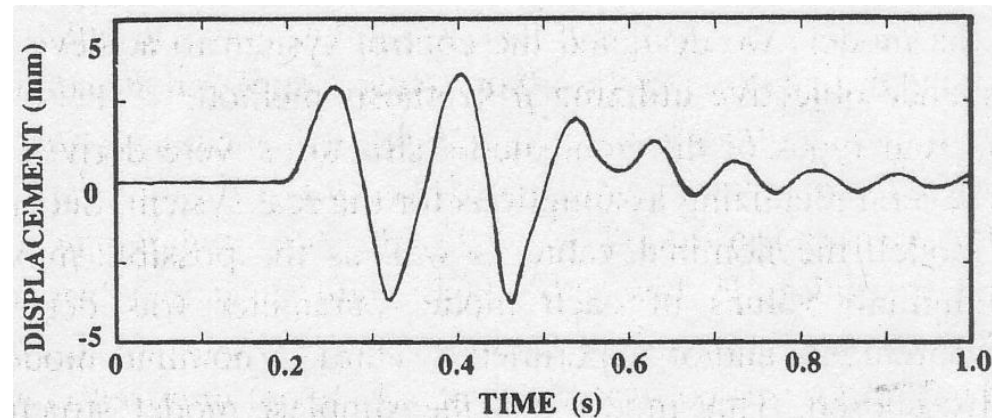


図 ステップ状外乱に対する時間応答(コントローラ K_1)

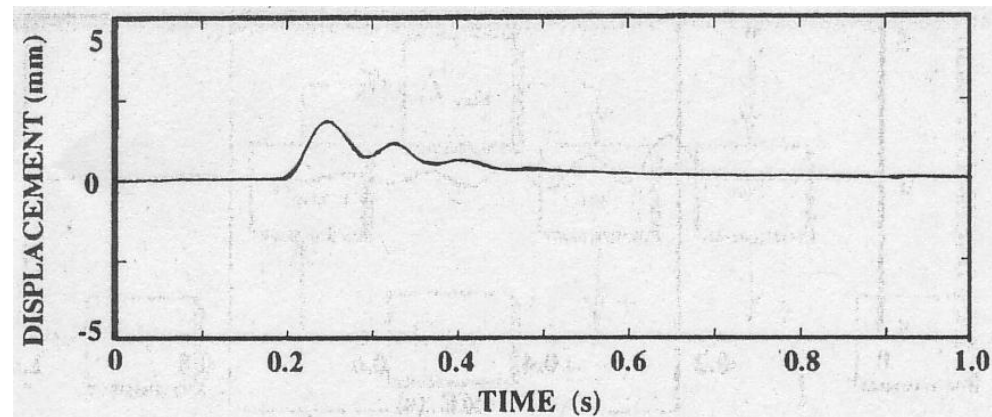


図 ステップ状外乱に対する時間応答(コントローラ K_2)

付録

μ - Analysis and Synthesis Toolbox

μ - Analysis and Synthesis Toolbox

Starting mu iteration #: 1

D-K イテレーションの1回目

Iteration Number: 1

Information about the Interconnection Structure IC_DK:

system: 7 states 3 outputs 3 inputs

Test bounds: $1.3000 < \gamma \leq 1.3000$

gamma	hamx_eig	xinf_eig	hamy_eig	yinf_eig	nrho_xy	p/f
1.300	3.0e+001	1.0e-009	1.0e-001	0.0e+000	0.6912	p

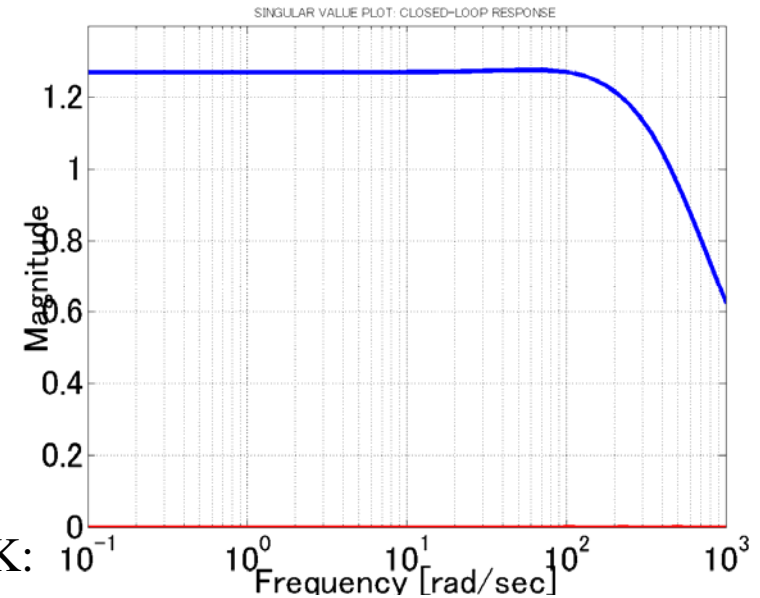
Gamma value achieved: 1.3000

Singular Value plot of closed-loop system in GRAPHICS window

Make sure that chosen Frequency range is appropriate

Do you want to modify OMEGA_DK? (y/n): n

考慮する周波数領域を変更する(y) or しない(n)



RERUN H_inf with DIFFERENT bounds/tolerances? (y/n): **n**

Calculating MU of closed-loop system:
points completed....

GMIN_DK, GMAX_DK の範囲を
変更する(y) or しない(n)

1.2.3.4.5.6.7.8.9.10.11.12.13.14.15.16.17.
18.19.20.21.22.23.24.25.26.27.28.29.30.31.32.33.34.35.
36.37.38.39.40.41.42.43.44.45.46.47.48.49.50.51.52.53.
54.55.56.57.58.59.60.

MU plots for control design: Strike any key to continue

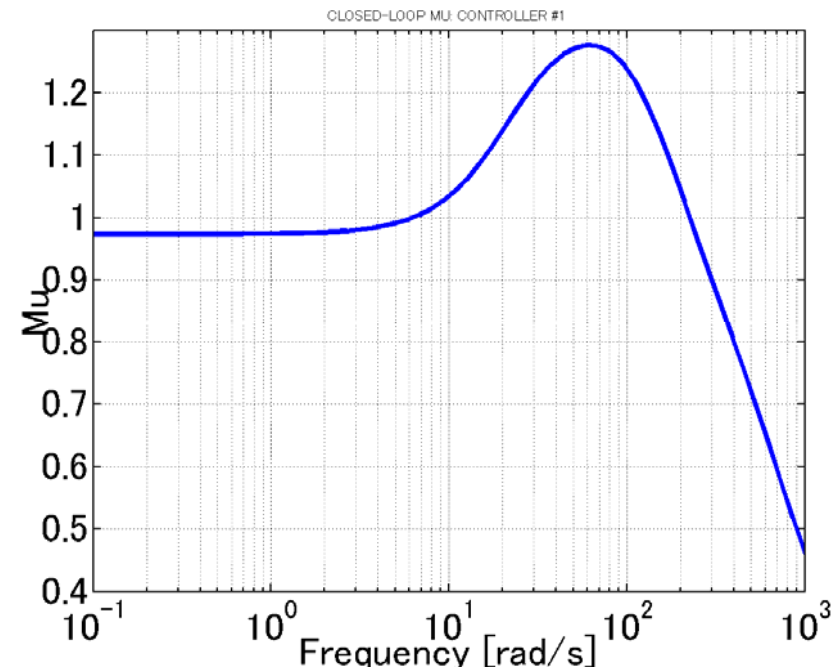
Enter

Iteration Summary

Iteration # 1
Controller Order 7
Total D-Scale Order 0
Gamma Acheived 1.300
Peak mu-Value 1.276

Another D-K iteration? (y/n): **y**

2回目のDKイテレーションを
行う(y) or 行わない(n)



Enter Choice (return for list): **Enter**

Choices:

nd	Move to Next D-scaling
nb	Move to Next D-Block
i	Increment Fit Order
d	Decrement Fit Order
apf	Auto-PreFit
mx 3	Change Max-Order to 3
at 1.01	Change Auto-PreFit Tol to 1.01
0	Fit with zeroth order
2	Fit with second order
n	Fit with n'th order
e	Exit with Current Fittings
s	See Status

Enter Choice (return for list): **S**

現在の設定を確認する

Block 1: data

Block 2: data

AutoPreFit Fit Tolerance: 1.03

AutoPreFit Maximum Order: 5

Enter Choice (return for list): **mx 1**

Max-Order を 1 次に変更する

Changing Max-Ord...

Done

Enter Choice (return for list): **S**

Block 1: data 現在の設定を

Block 2: data 確認する

AutoPrefit Fit Tolerance: 1.03

AutoPrefit Maximum Order: 1

Enter Choice (return for list): **apf**

自動的にフィッティングを行う

Starting Auto-PreFit...

Auto Fit in Progress

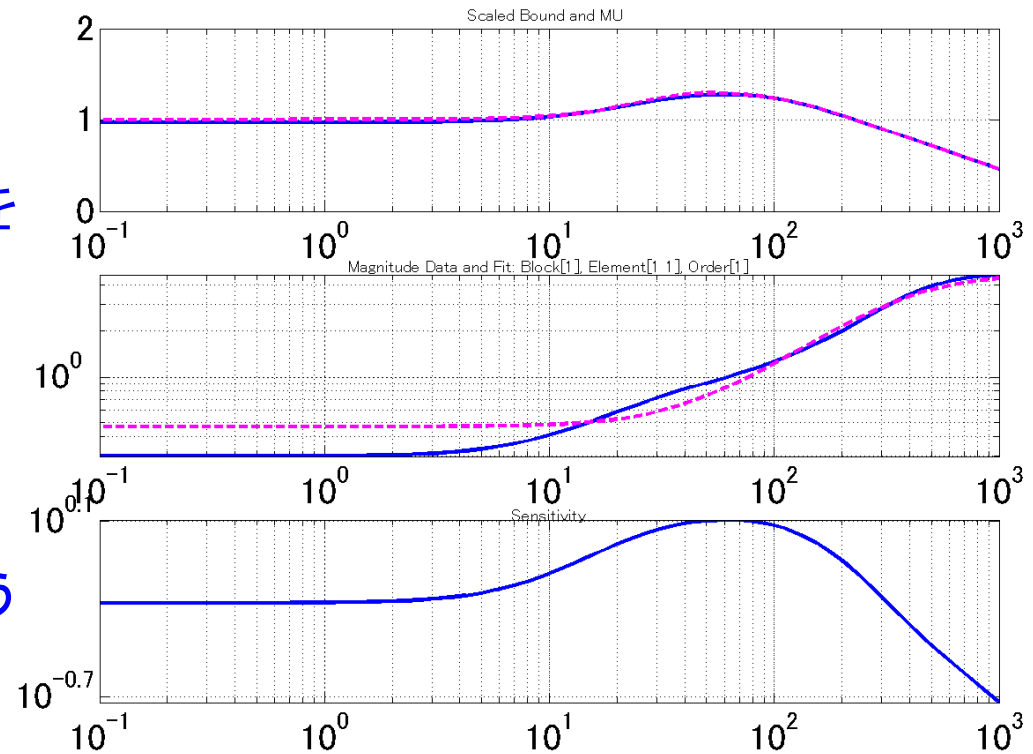
Block 1, MaxOrder=1, Order = 0 1

Block 2, MaxOrder=1, Order = 0

Done

Enter Choice (return for list): **e**

次に進む



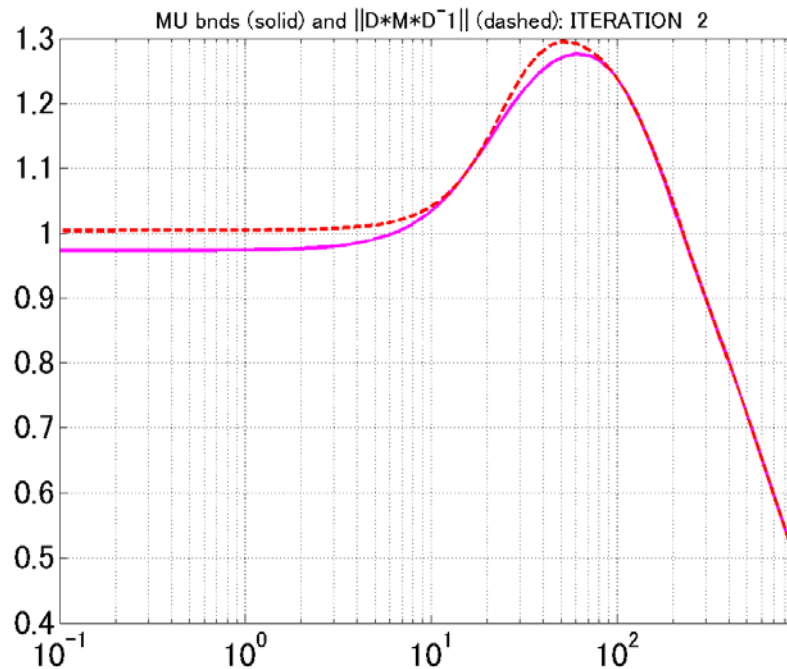
Altering the HINFSYN settings for next synthesis...

HINFSYN Settings	Previously	Next
(u) GAMMA Upper Bound	1.300	1.321
(l) GAMMA Lower Bound	1.300	1.300
(t) Bisection Tolerance	0.00e+000	4.19e-004
(p) Riccati PSD epsilon	1.00e-006	1.00e-006
(j) Riccati j-w epsilon	1.00e-008	1.00e-008

Enter (u l t p j) to change, (e) to exit: **u**

Enter new GAMMA Upper bound: **1.0**

GMAX_DK を 1.0 に変更する



HINFSYN Settings	Previously	Next
(u) GAMMA Upper Bound	1.300	1.000
(l) GAMMA Lower Bound	1.300	1.300
(t) Bisection Tolerance	0.00e+000	4.19e-004
(p) Riccati PSD epsilon	1.00e-006	1.00e-006
(j) Riccati j-w epsilon	1.00e-008	1.00e-008

Enter (u l t p j) to change, (e) to exit: **l**

Enter new GAMMA Lower bound: **1.0** GMIN_DK を 1.0 に変更する

HINFSYN Settings	Previously	Next
(u) GAMMA Upper Bound	1.300	1.000
(l) GAMMA Lower Bound	1.300	1.000
(t) Bisection Tolerance	0.00e+000	4.19e-004
(p) Riccati PSD epsilon	1.00e-006	1.00e-006
(j) Riccati j-w epsilon	1.00e-008	1.00e-008

Enter (u l t p j) to change, (e) to exit: **e**

次に進む

Iteration Number: 2

Information about the Interconnection Str
system: 9 states 3 outputs 3 inputs

Test bounds: 1.0000 < gamma <= 1.0000

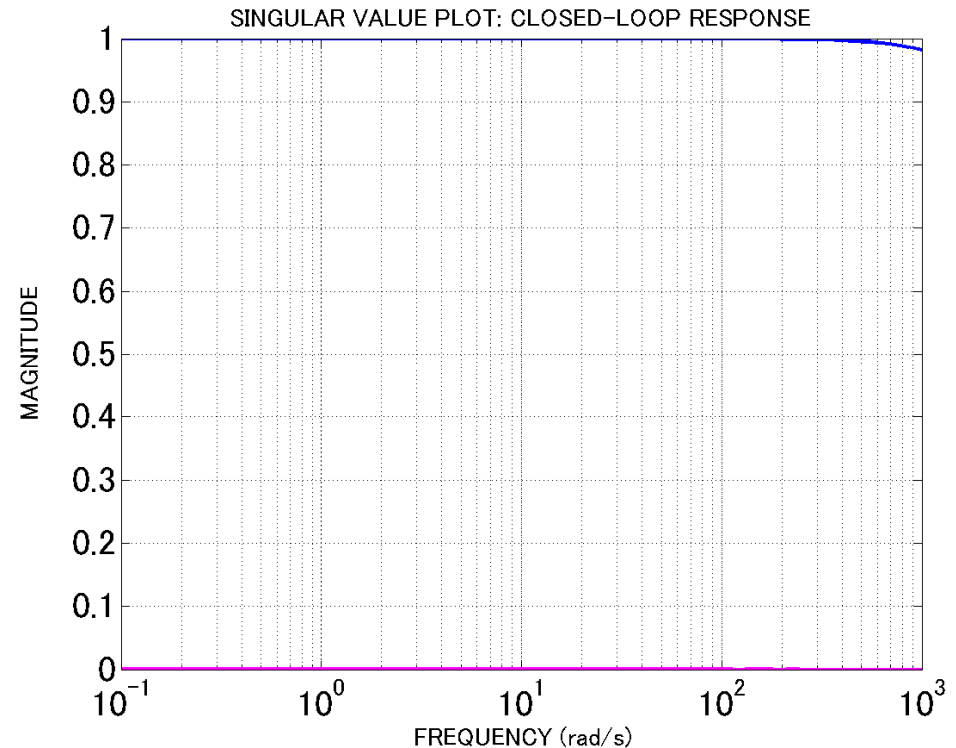
gamma	hamx_eig	xinf_eig	hamy_eig	yinf_eig	nrho_xy	p/f
1.000	2.2e+001	-3.0e-016	1.0e-001	0.0e+000	0.9615	p

Gamma value achieved: 1.0000

Singular Value plot of closed-loop system in GRAPHICS window

Make sure that chosen Frequency range is appropriate

Do you want to modify OMEGA_DK? (y/n): **n** 考慮する周波数領域を
変更する(y) or しない(n)



Do you want to modify OMEGA_DK? (y/n): **n**

RERUN H_inf with DIFFERENT bounds/tolerances? (y/n): **n**

Calculating MU of closed-loop system:
points completed....

GMIN_DK, GMAX_DK の範囲を
変更する(y) or しない(n)

1.2.3.4.5.6.7.8.9.10.11.12.13.14.15.16.17.

18.19.20.21.22.23.24.25.26.27.28.29.30.31.32.33.34.35.

36.37.38.39.40.41.42.43.44.45.46.47.48.49.50.51.52.53.

54.55.56.57.58.59.60.

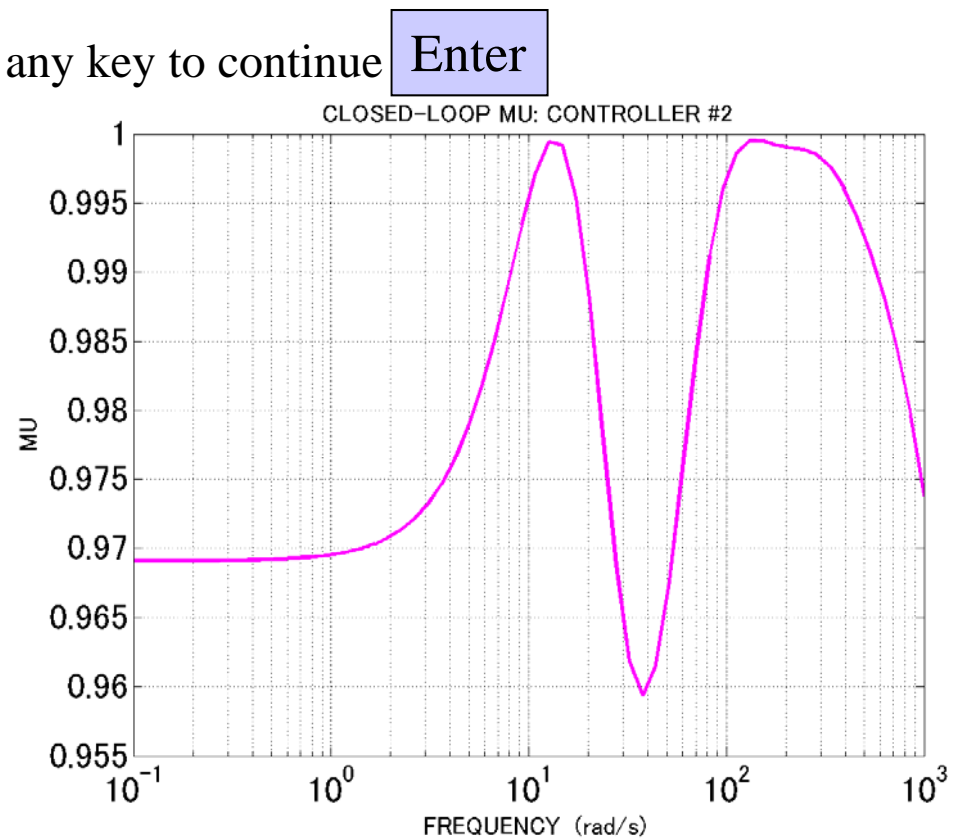
MU plots for control design: Strike any key to continue

Iteration Summary

Iteration #	1	2
Controller Order	7	9
Total D-Scale Order	0	2
Gamma Acheived	1.300	1.000
Peak mu-Value	1.276	1.000

Another D-K iteration? (y/n): **n**

終了



コントローラ $K_{dkit}(s)$

$$K_2(s) = \frac{-5.13 \times 10^9 (s + 11.36)(s + 30)(s + 35)(s + 38)(s + 39.6)}{(s + 0.1)(s + 43.09 \pm j3.31)(s + 27.02 \pm j11.96)} \\ \times \frac{(s + 45.67)(s + 67.05)(s + 397)}{(s + 407.0)(s + 423.5 \pm j281.1)(s + 5130)}$$

安定

極

- 0.1
- $-43.09 \pm j3.31$
- $-27.02 \pm j11.96$,
- 407.0
- $-423.5 \pm j281.1$,
- 5130

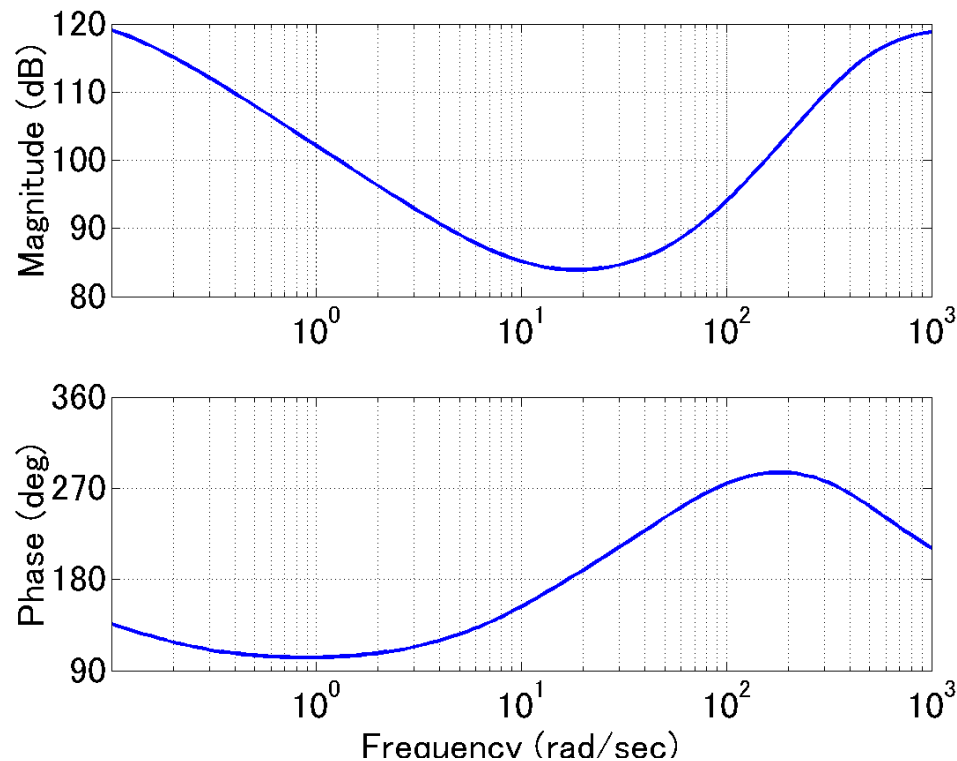


図 コントローラ $K_2(s)$

MATLAB program

```
[Kdk_a,Kdk_b,Kdk_c,Kdk_d] = ssdata(Kdk);  
Kdk_ss = ss ( Kdk_a, Kdk_b, Kdk_c, Kdk_d );
```

```
[k_dk2_a, k_dk2_b, k_dk2_c, k_dk2_d] = unpk ( k_dk2 );  
Kdk_ss = ss ( k_dk2_a, k_dk2_b, k_dk2_c, k_dk2_d );
```

```
zpk ( Kdk_ss )
```

```
[Kdk_pole,Kdk_zero] = pzmap ( Kdk_ss )
```

```
bode (Kdk_ss,omega1);
```


開ループ伝達関数

MATLAB program

```
L_ss = Gnom_ss*Kdk_ss;  
bode(L_ss,omega1);  
  
[Gm,Pm,Wpc,Wgc] = margin(L_ss);  
Gm=20*log10(Gm)
```

ゲイン余裕 6.31 [dB]

位相余裕 30.8 [deg]

ゲイン交差周波数 119.0 [rad/sec]

位相交差周波数 31.8 [rad/sec]

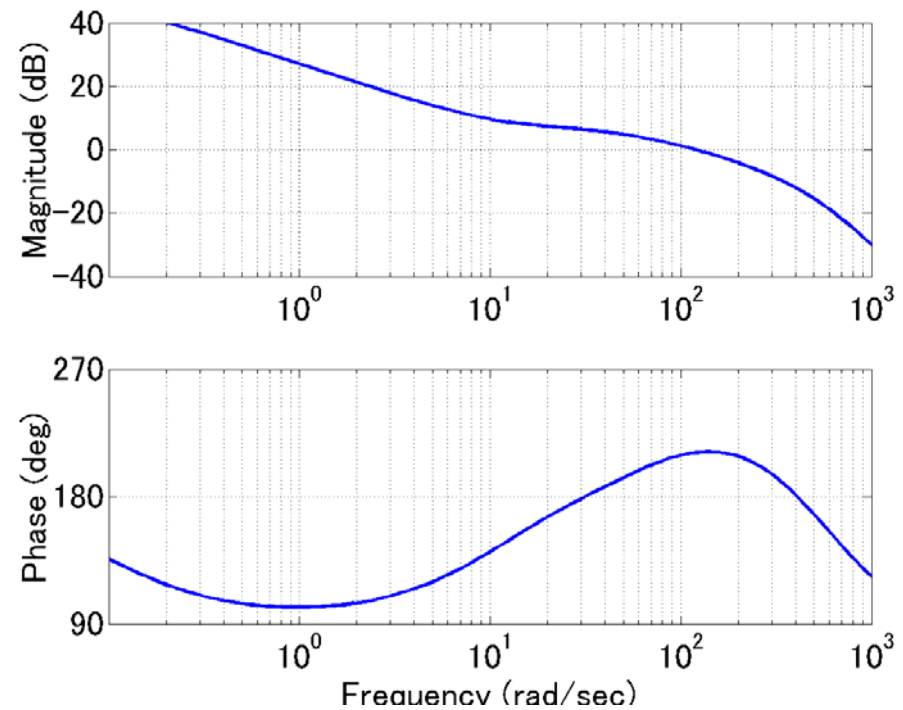


図 開ループ伝達関数

MATLAB program

```
nyquist ( L_ss )
```

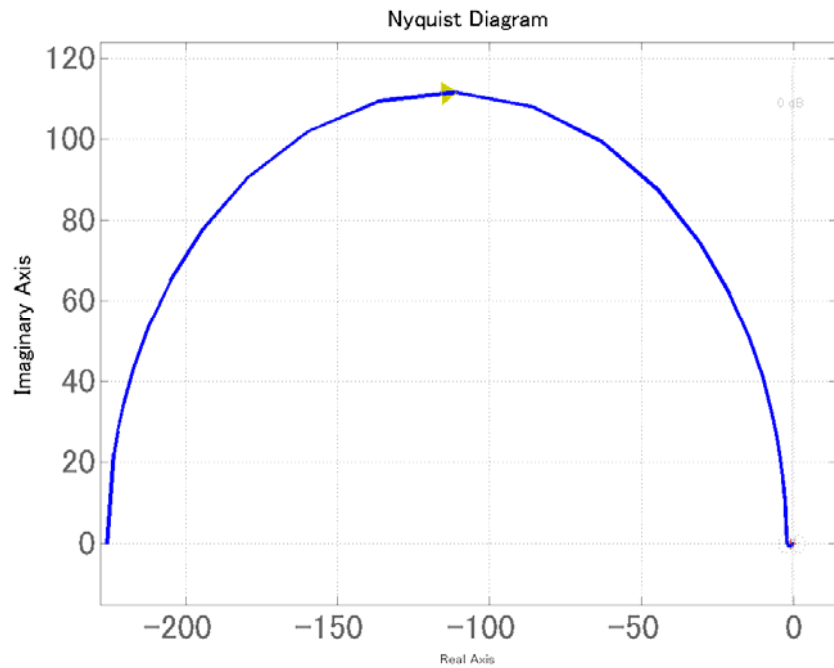


図 ベクトル軌跡

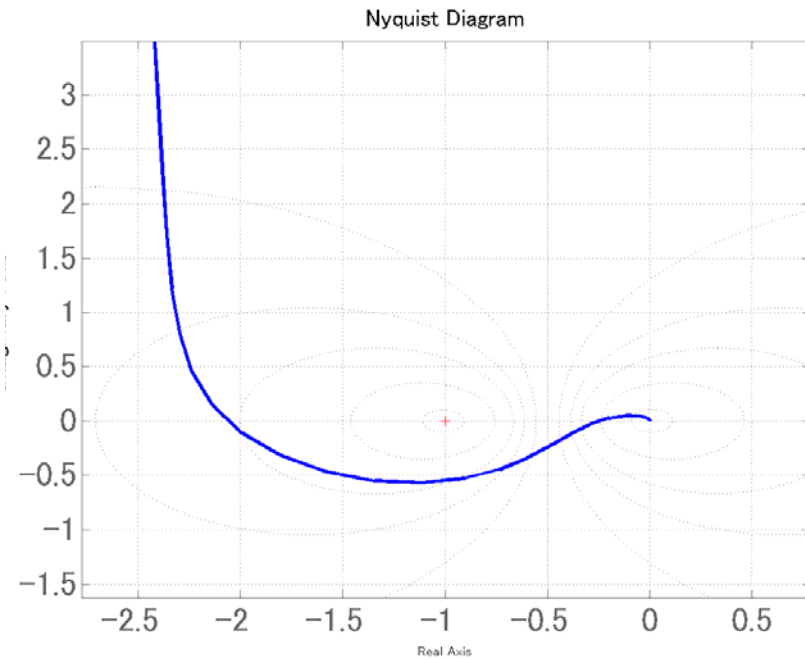
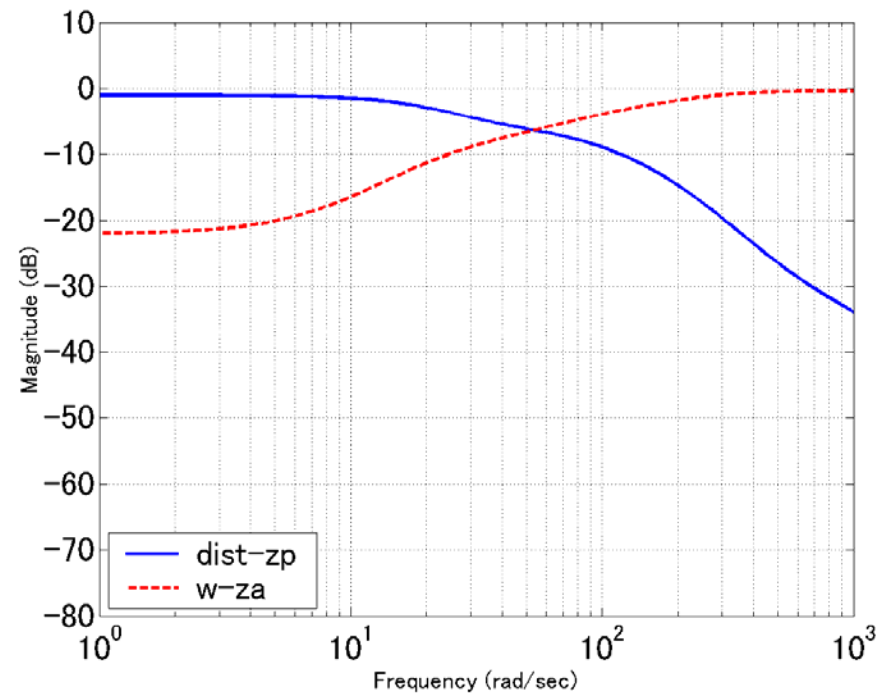
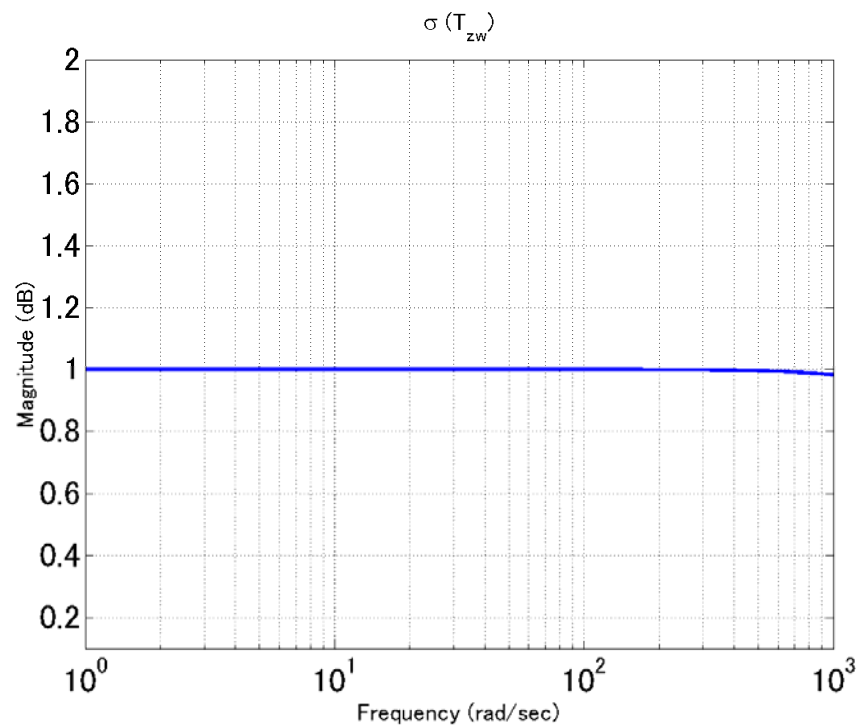


図 ベクトル軌跡

閉ループ系の周波数特性



MATLAB program

```
lft_General_P = lftdata ( General_P );  
DGD_ss = dkinfo{2}.DL * lft_General_P / dkinfo{2}.DR;
```

```
DGD_sys = mmult ( dl_dk2,General_P,minv(dr_dk2));  
[DGD_a,DGD_b,DGD_c,DGD_d] = unpck(DGD_sys);  
DGD_ss = ss(DGD_a,DGD_b,DGD_c,DGD_d);
```

```
Cloop_ss = lft(DGD_ss,Kdk_ss);
```

```
sigma(Cloop_ss,omega1)
```

```
bode(Cloop_ss(1,1),omega1);
```

```
bode(Cloop_ss(2,2),omega1);
```

閉ループ系の特性

極 $-17.84 \pm j9.87$, -45.67 , -46.11 , -57.06 ,
 -66.92 , -67.18 , -91.63 , $-277.2 \pm j61.58$, -476.9 , -5128

MATLAB program

```
T = feedback(L_ss,1);  
pole(T)  
zero(T)  
Pzmap (T)
```

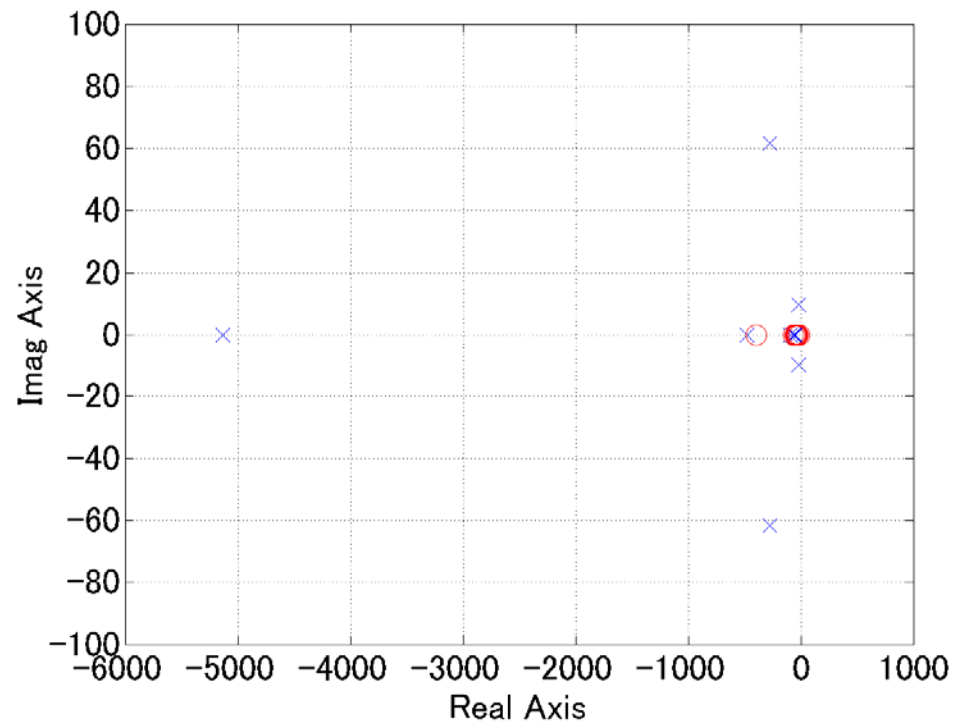
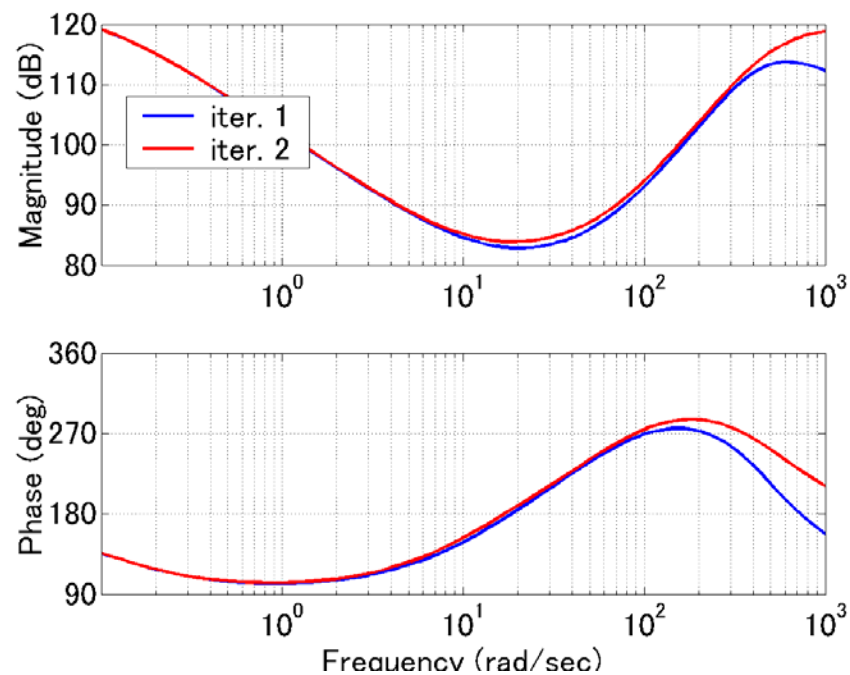


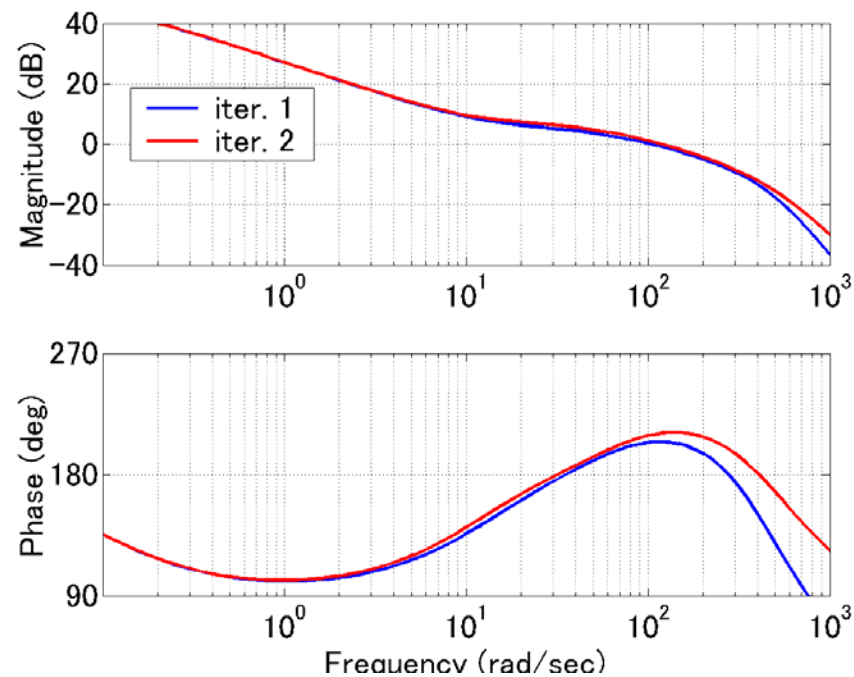
図 閉ループ系の極・零点

D-Kイテレーションによる変化

コントローラ



開ループ伝達関数



重み関数の変化

