# CLASS# CLASSNAME - Assignment #

By: Mason Smith Submitted: 11/6/20

# 1 Problem 1. READ DUAN

# 2 Problem 2. READ LECTURES

# 3 Problem 3.

## 3.1 Problem 3.a

**SOLUTION:**

$$|y(t)| = \Big| \int_{-\infty}^{\infty} h(t-s)u(s)ds \Big|$$

By the triangle inequality:

$$|y(t)| \leq \int_{-\infty}^{\infty} \big|h(t-s)u(s)\big|ds = \int_{-\infty}^{\infty} |h(t-s)||u(s)|ds = \int_{-\infty}^{\infty} |h(t-s)||u(s)|ds$$

Let $||u(s)||_\infty$ be the maximum value of the of $u(s)$. Since this term is fixed, we can treat it as a constant and separate it from the integral.

$$|y(t)| \leq \int_{-\infty}^{\infty} |h(t-s)|||u(s)||_\infty ds = ||u(s)||_\infty \int_{-\infty}^{\infty} |h(t-s)|ds$$

where t becomes negligible when integrating over $(-\infty, \infty)$ in the s-domain. Therefore:

$$|y(t)| \leq ||u(s)||_\infty \int_{-\infty}^{\infty} |h(s)|ds$$

which the $\mathcal{L}^1$ norm is recovered:

$$|y(t)| \leq ||u(s)||_\infty ||h(s)||_1$$

Where $||h(s)||_1$ is bounded given $h \in L^1(\mathbb{R})$ and $||u(s)||_\infty$ is bounded under the assumption that we cannot produce an input signal with infinite energy. Applying these two conclusions we can infer that $|y(t)|$ is also bounded:

$$|y(t)| \leq ||u(s)||_\infty ||h(s)||_1 < \infty$$

and conclude that for $\gamma = ||h(s)||_1$ and $\beta = 0$

$$||y(t)||_\infty = \sup_{s \in [0,\infty)} |y(t)| \leq \gamma ||u(s)||_\infty + \beta$$

## 3.2 Problem 3.b BONUS

Problem 1.b BONUS

**SOLUTION:**
INCOMPLETE

# 4 Problem 4.

## 4.1 Problem 4.a

**SOLUTION:**
Given:

$$||f||_{H^p_m(\Omega)} = \big(\int\limits_\Omega \sum_{k=0}^{m} |f^{(k)}(s)|^p ds\big)^{\frac{1}{p}}$$

We can expand the summation for $m+1$:

$$||f||_{H^p_{m+1}(\Omega)} = \big(\int\limits_\Omega \sum_{k=0}^{m+1} |f^{(k)}(s)|^p ds\big)^{\frac{1}{p}}$$

$$= \big(\int\limits_\Omega \sum_{k=0}^{m} |f^{(k)}(s)|^p ds + \int\limits_\Omega |f^{(m+1)}(s)|^p ds\big)^{\frac{1}{p}}$$

Where fore any value of p and any domain $\Omega$, $\int\limits_\Omega |f^{(m+1)}(s)|^p ds\big)^{\frac{1}{p}} > 0$. This result returns $||f||_{H^p_m(\Omega)}$ in addition to a positive term allowing the following inequalities to be satisfied:

$$\big(\int\limits_\Omega \sum_{k=0}^{m} |f^{(k)}(s)|^p ds\big)^{\frac{1}{p}} \leq \big(\int\limits_\Omega \sum_{k=0}^{m} |f^{(k)}(s)|^p ds + \int\limits_\Omega |f^{(m+1)}(s)|^p ds\big)^{\frac{1}{p}}$$

$$\int\limits_\Omega \sum_{k=0}^{m} |f^{(k)}(s)|^p ds \leq \int\limits_\Omega \sum_{k=0}^{m} |f^{(k)}(s)|^p ds + \int\limits_\Omega |f^{(m+1)}(s)|^p ds$$

$$0 \leq \int\limits_\Omega |f^{(m+1)}(s)|^p ds$$

where the term on the right has already been shown to be a non-negative term thereby satisfying the inequality and proving that

$$||f||_{H^p_m(\Omega)} \leq ||f||_{H^p_{m+1}(\Omega)}$$

### 4.2 Problem 4.b

**SOLUTION:**
Given:

$$p = 1, f(s) = e^{-s} \text{ and } \Omega = (0, inf)$$

$$f^{(k)}(s) = \frac{d^k f(s)}{ds^k} = \frac{d^k}{ds^k} e^{-s}$$

$$||f||_{H^1_m(0,\infty)} = (\int_0^\infty \sum_{k=0}^{m} |\frac{d^k}{ds^k} e^{-s}| ds)^{\frac{1}{p}}$$

for m=0:

$$||f||_{H^1_0(0,\infty)} = \int_0^\infty \sum_{k=0}^{m=0} |\frac{d^k}{ds^k} e^{-s}| ds = \int_0^\infty |\frac{d^0}{ds^0} e^{-s}| ds$$

$$||f||_{H^1_0(0,\infty)} = \int_0^\infty |e^{-s}| ds = \int_0^\infty e^{-s} = -e^{-s}\Big|_0^\infty$$

$$= -e^{-\infty} - (-e^{-\infty}) = 0 + 1$$

$$\therefore ||f||_{H^1_0(0,\infty)} = 1$$

for m=1:

$$||f||_{H^1_1(0,\infty)} = \int_0^\infty \sum_{k=0}^{m=1} |\frac{d^k}{ds^k} e^{-s}| ds$$

$$||f||_{H^1_1(0,\infty)} = \int_0^\infty |\frac{d^1}{ds^1} e^{-s}| ds + ||f||_{H^1_0(0,\infty)}$$

$$||f||_{H^1_1(0,\infty)} = \int_0^\infty |-e^{-s}| ds + 1 = \int_0^\infty e^{-s} ds + 1 = 1 + 1$$

$$\therefore ||f||_{H^1_1(0,\infty)} = 2$$

for m=2:

$$||f||_{H^1_2(0,\infty)} = \int_0^\infty \sum_{k=0}^{m=2} |\frac{d^k}{ds^k} e^{-s}| ds$$

$$||f||_{H^1_2(0,\infty)} = \int_0^\infty |\frac{d^2}{ds^2} e^{-s}| ds + ||f||_{H^1_1(0,\infty)} + ||f||_{H^1_0(0,\infty)}$$

$$||f||_{H^1_2(0,\infty)} = \int_0^\infty |e^{-s}| ds = \int_0^\infty e^{-s} ds + 1 + 1 = 1 + 1 + 1$$

$$\therefore ||f||_{H^1_2(0,\infty)} = 3$$

### 4.3 Problem 4.c

**SOLUTION:**
This pattern leads to an observation that

$$||f||_{H^1_m(\Omega)} = m + 1 \text{ or } ||f||_{H^1_m(\Omega)} = \sum_{k=0}^{m} 1$$

for the function $f(s) = e^{-s}$. This appears to be caused by the derivative of f(s) to only invert its sign with ever sequential k in $f^{(k)}(s) = \frac{d^k f}{ds^k}$. This changing of the $sign(f^{(k)})$ is then made inconsequential since the next operator is the absolute value of this result and leads to the same integration for every value of k which. This allows us to simplify the integral for all values of $\frac{d^k}{ds^k}e^{-s}$:

$$\int_0^\infty |\frac{d^k}{ds^k}e^{-s}|ds = \int_0^\infty e^{-s}ds = 1$$

# 5 Problem 5

**SOLUTION:**
**(i)Define Variables:**

**(i)Define Equations for Regulated Outputs in terms of Exogenous/Actuator Inputs:**
$z1 = r - P_0(n_{proc} + u)$
$z2 = u$

**(ii)Define Equations for Sensed Outputs in terms of Exogenous/Actuator Inputs:**
$y_1 = r$
$y_2 = q + n_{sensor}$

**(iii)Construct Matrices for Regulated/Sensed Outputs:**
$z = \begin{bmatrix} e \\ u \end{bmatrix} = \begin{bmatrix} r - P_0(n_{proc} + u) \\ u \end{bmatrix} = \begin{bmatrix} \omega_1 - P_0(\omega_2 + u) \\ u \end{bmatrix}$
$y = \begin{bmatrix} r \\ q + n_{sensor} \end{bmatrix} = \begin{bmatrix} r \\ P_0(n_{proc} + u) + n_{sensor} \end{bmatrix} = \begin{bmatrix} \omega_1 \\ P_0(\omega_2 + u) + \omega_3 \end{bmatrix}$

**(iiv)Construct Aggregate Plant:**
$P = \begin{bmatrix} z_1 \\ z_2 \\ y_1 \\ y_2 \end{bmatrix} = \left[\begin{array}{cccc} I & -P_0 & 0 & | & -P_0 \\ 0 & 0 & 0 & | & I \\ \hline I & 0 & 0 & | & 0 \\ 0 & P_0 & I & | & P_0 \end{array}\right] \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ u \end{bmatrix} = \left[\begin{array}{ccc} P_{11} & | & P_{12} \\ \hline P_{21} & | & P_{22} \end{array}\right] \begin{bmatrix} \omega \\ u \end{bmatrix}$

**(iv)Construct State Space Representation:**
where $\omega_2 + u$ is our input signal

$$P = \left[\begin{array}{ccc} P_{11} & | & P_{12} \\ \hline P_{21} & | & P_{22} \end{array}\right] \begin{bmatrix} \omega \\ u \end{bmatrix}$$

$$\dot{x} = Ax + B(\omega_2 + u)$$

$$y = Cx + D(\omega_2 + u)$$

**(v)Construct the 9-matrix representation**

$$P = \left[\begin{array}{cccccc} A & | & 0 & B & 0 & B \\ \hline C & | & I & -D & 0 & -D \\ 0 & | & 0 & 0 & 0 & I \\ 0 & | & I & 0 & 0 & 0 \\ C & | & 0 & D & I & D \end{array}\right]$$

# 6 Problem 6

## 6.1 Problem 6.a

**SOLUTION:**
For the regulator framework
**(i)Define Equations for Regulated Outputs in terms of Exogenous/Actuator Inputs:**
$z1 = P_0(\omega_1 + u)$
$z2 = u$

**(ii)Define Equations for Sensed Outputs in terms of Exogenous/Actuator Inputs:**
$y = P_0(\omega_1 + u) + \omega_2$

**(iii)Construct Matrices for Regulated/Sensed Outputs:**

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} P_0(\omega_1 + u) \\ u \end{bmatrix}$$

$$\begin{bmatrix} y \end{bmatrix} = \begin{bmatrix} P_0(\omega_1 + u) + \omega_2 \end{bmatrix}$$

**(iiv)Construct Aggregate Plant:**

$$\begin{bmatrix} z_1 \\ z_2 \\ y \end{bmatrix} = \left[\begin{array}{cc|c} P_0 & 0 & P_0 \\ 0 & 0 & I \\ \hline P_0 & I & P_0 \end{array}\right] \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \left[\begin{array}{c|c} P_{11} & P_{12} \\ \hline P_{21} & P_{22} \end{array}\right] \begin{bmatrix} \omega \\ u \end{bmatrix}$$

**(iv)Construct State Space Representation:**
where $\omega_1 + u$ is our input signal and the state space representation takes the form:

$$P = \left[\begin{array}{c|c} P_{11} & P_{12} \\ \hline P_{21} & P_{22} \end{array}\right]$$

$$\dot{x} = Ax + B(\omega_1 + u)$$

$$y = Cx + D(\omega_1 + u)$$

**(v)Construct the 9-matrix representation** We can the construct our reconfigured plant:

$$P = \left[\begin{array}{c|ccc} A & B & 0 & B \\ \hline C & D & 0 & D \\ 0 & 0 & 0 & I \\ C & D & I & D \end{array}\right]$$

**(vi)Matlab Result** Using the values provided for A,B,C,D the full 9-matrix representation for P can be calculated using matlab:

$$P = \begin{pmatrix} -1 & 1 & 0 & 1 & 0 & 1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 \\ -1 & -2 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -2 & -1 & 1 & 1 & -1 & 1 & 1 & 0 & 0 & 0 & -1 & 1 & 1 \\ -1 & 1 & -1 & -2 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ -1 & -1 & 1 & 1 & -2 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & -1 & -3 & -1 & 1 & 1 & 0 & 0 & 0 & -1 & 1 & 1 \\ 0 & 1 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

## 6.2 Problem 6.b

**SOLUTION:**

**(i)LMI to solve $H_\infty$ optimal state-feedback problem**

The following are equivalent.

1) There exists a $F$ such that $||\underline{S}(P, K(0,0,0,F)||_{H_\infty} \leq \gamma$

2) There exists $Y > 0$ and $Z$ such that

$$\begin{bmatrix} YA^T + AY + Z^T B_2^T + B_2 Z & B_1 & YC_1^T + Z^T D_{12}^T \\ B1_1^T & -\gamma I & D_{11}^T \\ C_1 Y + D_{12} Z & D_{11} & -\gamma I \end{bmatrix} < 0$$

Then $F = ZY^{-1}$

The lower linear fractional transformation (LFT) is used to implement a controller $K$ into the system. The lower LFT is denoted as $\underline{S}(P, K)$ and is formed by $\underline{S}(P, K) = P_{11} + P_{12}(I - KP_{22})^{-1} KP_{21}$ with $\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}$

For full-state feedback we consider a controller of the form $u(t) = Fx(t)$. This is a special case where $y(t) = x(t)$ and results in a controller of the form $K = \begin{bmatrix} 0 & 0 \\ 0 & F \end{bmatrix}$.

**(ii)Matlab Result**

$$F = \begin{bmatrix} 0.6817 & -1.2214 & 0.7896 & 0.8756 & 1.6454 & 0.6413 \\ 0.5895 & 1.3320 & -0.7950 & -0.2227 & -1.4536 & 0.2842 \\ 1.1663 & 3.1650 & -2.7014 & -1.0259 & -2.7165 & 1.4060 \end{bmatrix}$$

$||\underline{S}(P, K(0,0,0,F)||_{H_\infty} < 0.9476$

$H_\infty$ closed-loop gain $= 0.9476$

### 6.3 Problem 6.c

**SOLUTION:**
**(i)Does it exist?**
The $H_2$ closed loop gain requires $D_cl = 0$ and since this is true for the controller designed in part b, the $H_2$ closed loop gain exists.

**(ii)Define the close loop representation for controller**

$$\begin{aligned} A_{cl} &= A + B_2 * F \\ B_{cl} &= B_1 \\ C_{cl} &= C_1 + D_12 * F \\ D_{cl} &= D_11 \end{aligned}$$

**(iii)LMI for $H_2$ closed loop gain**
Assuming that $\hat{P}(s) = C(sI - A)^{-1}B$, this means that the following are equivalent:
1) $A$ is Hurwitz and $||\hat{P}||^2_{H_2} < \gamma$

2)
$$\begin{array}{cl} trace(C_{cl}XC_{cl}^T) & < \gamma \\ A_{cl}X + XA_{cl}^T + B_{cl}B_{cl}^T & < 0 \\ X_{cl} & > 0 \end{array}$$

**(iiv)Matlab Result**

$$X = \begin{pmatrix} 0.2447 & -0.0490 & -0.0923 & 0.0281 & -0.0720 & -0.0909 \\ -0.0490 & 0.0273 & -0.0023 & 0.0204 & 0.0169 & 0.0033 \\ -0.0923 & -0.0023 & 0.2416 & 0.0558 & 0.0082 & 0.2136 \\ 0.0281 & 0.0204 & 0.0558 & 0.1277 & -0.0069 & 0.0658 \\ -0.0720 & 0.0169 & 0.0082 & -0.0069 & 0.1077 & -0.0063 \\ -0.0909 & 0.0033 & 0.2136 & 0.0658 & -0.0063 & 0.2076 \end{pmatrix}$$

$||\hat{P}||_{H_2} < 2.1699$
$H_2$ gain=2.1699

### 6.4 Problem 6.d

**SOLUTION:**
**(ii)LMI for $H_2$-optimal state feedback control**
The following are equivalent.
1) There exists a $K$ such that $||S(K, P)||_{H_2} < \gamma$
2) There exists $X > 0$, $Z$ and $W$ such that

$$\begin{bmatrix} A & B_2 \end{bmatrix} \begin{bmatrix} X \\ Z \end{bmatrix} + \begin{bmatrix} X & Z^T \end{bmatrix} \begin{bmatrix} A^T \\ B_2^T \end{bmatrix} + B_1 B_1^T < 0$$
$$\begin{bmatrix} X & *^T \\ C_1X + D_{12}Z & W \end{bmatrix} > 0$$
$trace(W) < \gamma^2$
where $K = ZX^{-1}$

**(iii)Matlab Result**

$$K = \begin{pmatrix} 0.1348 & -0.3049 & 0.0313 & 0.5380 & 0.2327 & 0.2856 \\ 0.3234 & 0.2382 & -0.0867 & -0.1179 & -0.3702 & 0.0225 \\ 0.5747 & 0.4936 & -0.2082 & -0.2317 & -0.7455 & 0.0251 \end{pmatrix}$$

$||S(K, P)||_{H_2} < 1.6572$ $H_2$ gain $= 1.6572$

### 6.5 Problem 6.e

**SOLUTION:**
**(i) LMI for $H_\infty$ gain**
*Note: used dialated KYP lemma*

**(iii)Matlab Result**

$$X = \begin{pmatrix} 1.2639 & 0.0550 & 0.2727 & -0.0282 & 0.0170 & 0.2706 \\ 0.0550 & 1.0329 & -0.2907 & -0.0647 & -0.5094 & 0.1786 \\ 0.2727 & -0.2907 & 0.7912 & -0.0994 & 0.3676 & -0.0561 \\ -0.0282 & -0.0647 & -0.0994 & 0.8802 & 0.1414 & 0.1008 \\ 0.0170 & -0.5094 & 0.3676 & 0.1414 & 0.9138 & -0.1184 \\ 0.2706 & 0.1786 & -0.0561 & 0.1008 & -0.1184 & 0.6175 \end{pmatrix}$$

$H_\infty$ gain of closed loop system = 1.0813

## 7 Problem 7

### 7.1 Problem 7.a

**SOLUTION:**
The following are equivalent: There exists a $\hat{K} = \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix}$ such that $||S(K,P)||_{H_\infty} < \gamma$
There exists $X_1$, $Y_1$, $Z$, $A_n$, $B_n$, $C_n$, $D_n$ < such that

$$\begin{bmatrix} X_1 & I \\ I & Y_1 \end{bmatrix} > 0$$

$$\begin{bmatrix} AY_1 + Y_1A^{\mathrm{T}} + B_2C_n + C_nB_2^{\mathrm{T}} & *^{\mathrm{T}} & *^{\mathrm{T}} & *^{\mathrm{T}} \\ A^{\mathrm{T}} + A_n + (B_2D_nC_2)^{\mathrm{T}} & X_1A + A^{\mathrm{T}} + B_nC_2 + C_2^{\mathrm{T}}B_n^{\mathrm{T}} & *^{\mathrm{T}} & *^{\mathrm{T}} \\ (B_1 + B_2D_nD_{21})^{\mathrm{T}} & (X_1B_1 + B_nD_{21})^{\mathrm{T}} & -\gamma I & *^{\mathrm{T}} \\ C_1Y_1 + D_{12}C_n & C_1 + D_{12}D_nC_2 & D_{11} + D_{12}D_nD_{21} & -\gamma I \end{bmatrix} < 0$$

**Matlab Result:** $H_\infty gain = 1.1072$

## 7.2 Problem 7.b

**SOLUTION:**
(i) Construct the corresponding controller
The above LMI determines the the upper bound $\gamma$ on the $H_\infty$ norm. In addition to this the controller $\hat{K}(A_K, B_K, C_K, D_K)$ can also be recovered.
$D_K = (I + D_{K2}D_{22})^{-1}D_{K2}$
$B_K = B_{K2}(I + D_{22}D_K)$
$C_K = (I + D_K D_{22})C_{K2}$
$A_K = A_{K2} - B_K(I + D_{22}D_K)^{-1}D_{22}C_K$

where, $\begin{bmatrix} A_{K2} & B_{K2} \\ C_{K2} & D_{K2} \end{bmatrix} = \begin{bmatrix} X_2 & X_1B_2 \\ 0 & I \end{bmatrix}^{-1} \left[ \begin{bmatrix} A_n & B_n \\ C_n & D_n \end{bmatrix} - \begin{bmatrix} X_1AY_1 & 0 \\ 0 & 0 \end{bmatrix} \right] \begin{bmatrix} Y_2^T & 0 \\ C_2Y_1 & I \end{bmatrix}^{-1}$
for any full-rank $X_2$ and $Y_2$ such that

$$\begin{bmatrix} X_1 & X_2 \\ X_2^T & X_3 \end{bmatrix} = \begin{bmatrix} Y_1 & Y_2B_2 \\ Y_2^T & Y_3 \end{bmatrix}^{-1}$$

$\therefore$ we can set $Y_2 = I$ and $X_2 = I - X_1 * Y_1$ and perform the above operations in inverse order to on our feasible solutions to our optimization problem to find $A_K, B_K, C_K, D_K$ and construct the controller:

$$K = \begin{bmatrix} A_K & B_k \\ C_K & D_k \end{bmatrix}$$

**Result**

$$K = \begin{pmatrix} 8.2264e+04 & 1.0279e+04 & -2.6134e+05 & -5.1350e+04 & 2.9243e+05 & 4.1978e+04 & 2.1486e+04 & 290.9353 & 2.9722e+04 \\ 7.4234e+04 & 9.2721e+03 & -2.3584e+05 & -4.6341e+04 & 2.6390e+05 & 3.7892e+04 & 1.9390e+04 & 262.7264 & 2.6822e+04 \\ -5.0414e+04 & -6.2962e+03 & 1.6016e+05 & 3.1469e+04 & -1.7921e+05 & -2.5725e+04 & -1.3167e+04 & -178.3610 & -1.8214e+04 \\ -1.9179e+04 & -2.3969e+03 & 6.0933e+04 & 1.1971e+04 & -6.8187e+04 & -9.7979e+03 & -5.0106e+03 & -68.2194 & -6.9307e+03 \\ -1.0883e+05 & -1.3594e+04 & 3.4572e+05 & 6.7930e+04 & -3.8685e+05 & -5.5532e+04 & -2.8423e+04 & -385.0597 & -3.9318e+04 \\ 2.7914e+04 & 3.4872e+03 & -8.8679e+04 & -1.7424e+04 & 9.9226e+04 & 1.4239e+04 & 7.2903e+03 & 98.6305 & 1.0085e+04 \\ 3.8985 & -0.9459 & -5.2285 & 0.6166 & 4.1976 & -3.3055 & 0.0089 & 0.0028 & -0.0038 \\ -4.0558 & 1.2173 & 7.2609 & 0.6405 & -5.5473 & 4.8033 & 0.0028 & -0.0049 & -0.0018 \\ -2.5704 & 0.8638 & 4.8891 & 0.4711 & -5.1088 & 3.0666 & -0.0038 & -0.0018 & -0.0079 \end{pmatrix}$$

**(ii) show that it achieves the predicted closed-loop Hinf gain.**

To construct the SS representation of the closed loop sys:
$Acl = \begin{bmatrix} A & 0 \\ 0 & Ak \end{bmatrix} + \begin{bmatrix} B2 & 0 \\ 0 & Bk \end{bmatrix} \begin{bmatrix} I & -Dk \\ -D22 & I \end{bmatrix}^{-1} \begin{bmatrix} 0 & Ck \\ C2 & 0 \end{bmatrix}$
$Bcl = \begin{bmatrix} B1 + B2 * Dk * Q * D21 \\ Bk * Q * D21 \end{bmatrix}$
$Ccl = \begin{bmatrix} C1 & 0 \end{bmatrix} + \begin{bmatrix} D12 & 0 \end{bmatrix} \begin{bmatrix} I & -Dk \\ -D22, I & \end{bmatrix}^{-1} \begin{bmatrix} 0 & Ck \\ C2 & 0; \end{bmatrix}$
$Dcl = \begin{bmatrix} D11 + D12 * Dk * Q * D21 \end{bmatrix}$

where the matlab system analysis commands con be used:
$sys = ss(Acl, Bcl, Ccl, Dcl);$
$Hinf_{ctrl} = norm(sys, inf);$
**Result**
(Returned from Matlab SS System Analysis) $H_\infty$ gain = 1.1070

Therefore the controller approx achieves the predicted gain since the following terms are approximately equal:
$H_\infty(LMI) = H_\infty(controller)$; 1.1072=1.107

## 7.3 Problem 7.c

**SOLUTION:**
**(i) Use an LMI to formulate and solve the H2-optimal ouput-feedback problem.**
1) There exists a $\hat{K} = \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix}$ such that $||S(K,P)||_{H_2} < \gamma$

2) There exists $X_1$, $Y_1$, $Z$, $A_n$, $B_n$, $C_n$, $D_n$ such that

$$\begin{bmatrix} AY_1 + Y_1A^{\mathrm{T}} + B_2C_n + C_nB_2^{\mathrm{T}} & *^{\mathrm{T}} & *^{\mathrm{T}} \\ A^{\mathrm{T}} + A_n + (B_2D_nC_2)^{\mathrm{T}} & X_1A + A^{\mathrm{T}} + B_nC_2 + C_2^{\mathrm{T}}B_n^{\mathrm{T}} & *^{\mathrm{T}} \\ (B_1 + B_2D_nD_{21})^{\mathrm{T}} & (X_1B_1 + B_nD_{21})^{\mathrm{T}} & -\gamma I \end{bmatrix} < 0$$

$$\begin{bmatrix} Y_1 & *^T & *^T \\ I & X_1 & *^T \\ C_1Y_1 + D_{12}C_n & C_1 + D_{12}D_nC_2 & Z \end{bmatrix} > 0$$

$D_{11} + D_{12}D_nD_{21} = 0$

$\text{trace}(Z) < \gamma^2$

**(ii) What H2 gain diid you find?**
(Returned from LMI) $H_2$ gain = 1.9340

## 7.4 Problem 7.d

**SOLUTION:**
(i) Construct the corresponding controller

$$K = \begin{pmatrix} 7.8330 & 0.3149 & 18.4869 & 16.8449 & 2.3027 & 17.8954 & 32.3854 & 28.3374 & -20.5784 \\ -10.3488 & 2.5522 & 45.8789 & 19.9332 & 20.5579 & 34.7307 & 52.2954 & 28.4836 & -11.9234 \\ -35.6560 & 4.6924 & 16.7731 & -11.1883 & 26.7231 & 0.0492 & -16.3393 & -30.7800 & 45.8769 \\ -19.3020 & 2.0994 & -17.5058 & -19.3538 & 5.5660 & -21.0804 & -43.6287 & -40.4138 & 38.7429 \\ 38.2431 & -4.1367 & -5.2653 & 20.1470 & -24.8815 & 11.2499 & 31.8608 & 45.0072 & -57.0023 \\ 45.4199 & -5.7606 & -25.4699 & 13.0189 & -36.7500 & -1.0806 & 13.7220 & 35.5777 & -56.7713 \\ -0.0102 & 0.0020 & 0.0423 & 0.0145 & 0.0270 & 0.0219 & -1.0143e-11 & -3.2897e-12 & -2.4201e-12 \\ 0.0650 & -0.0059 & -0.0759 & -0.0063 & -0.0908 & -0.0049 & 9.1441e-12 & 1.1925e-13 & 1.1959e-11 \\ 0.1259 & -0.010 & -0.1556 & -0.0118 & -0.1793 & -0.0130 & 1.8773e-11 & 4.0258e-13 & 2.3288e-11 \end{pmatrix}$$

**(ii) show that it achieves the predicted closed-loop $H_2$ gain.**
(Returned from Matlab SS System Analysis) $H_2$ gain = 1.9340

Therefore the controller approx achieves the predicted gain since the following terms are approximately equal:
$H_2(LMI) = H_2(controller)$;1.934=1.934

# 8 8

## 8.1 Problem 8.a

**SOLUTION:**

The mixed $H_2 - H_\infty$ LMI aims to optimize both norms.

$$||S(K,P)_{H_2} < \gamma_1 \text{ and } ||S(K,P)_{H_\infty} < \gamma_2$$

To reformulate the problem to find

$$min_K ||S(K,P)||^2_{H_2} + ||S(K,P)||^2_{H_\infty}$$
$$||S(K,P)^2_{H_2} < \gamma_1^2 \text{ and } ||S(K,P)^2_{H_\infty} < \gamma_2^2$$

we must address the difference between the $H_2$ optimization problem ($||S(K,P)||^2_{H_2} < \gamma_1$) and the $H_\infty$ optimization problem ($||S(K,P)||_{H_\infty} < \gamma_2$) by changing the $H_\infty$ optimization problem to ($||S(K,P)||^2_{H_\infty} < \gamma_2^2$) where $\beta_2 = \gamma_2^2$ giving our minimization problem

$$min_K(weight_1 * \gamma_1 + weight_2 * \beta_2)$$

**$H_2$ Constraints**
Constraint for the $H_2$ norm that ensures norm is less than $\gamma_1$

$$\begin{bmatrix} (A*Y1 + Y1*A' + B2*Cn + Cn'*B2') & *^T & *^T \\ (A' + An + (B2*Dn*C2)') & (X1*A + A'*X1 + Bn*C2 + C2'*Bn') & *^T \\ ((B1 + B2*Dn*D21)') & ((X1*B1 + Bn*D21)') & (-I) \end{bmatrix} < 0$$

$$\begin{bmatrix} (Y1) & I & *^T \\ I & (X1) & *^T \\ (C1*Y1 + D12*Cn) & (C1 + D12*Dn*C2) & (Z) \end{bmatrix} > 0$$
$$(D11 + D12*Dn*D21) = 0$$
$$trace(Z) < \gamma_1^2$$

Since the constraint $trace(Z) < \gamma_1^2$ is not linear we must make variable substitutions for $\beta_1 = \gamma_1^2$

$$trace(Z) < \beta_1$$

and we make the substitution in our original optimization problem

$$min_K(weight_1 * \beta_1 + weight_2 * \beta_2)$$

**$H_\infty$ Constraints**
Constraint for the $H_\infty$ norm that ensures norm is less than $\gamma_2$

$$\begin{bmatrix} (A*Y1 + Y1*A' + B2*Cn + Cn'*B2') & *^T & *^T & *^T \\ (A' + An + (B2*Dn*C2)') & (X1*A + A'*X1 + Bn*C2 + C2'*Bn') & *^T & *^T \\ ((B1 + B2*Dn*D21)') & (X1*B1 + Bn*D21)' & (-\beta_2*I) & *^T \\ (C1*Y1 + D12*Cn) & (C1 + D12*Dn*C2) & (D11 + D12*Dn*D21) & (-\beta_2*I) \end{bmatrix} < 0$$

However, we can only optimize for one variable at a time and therefore must create a new variable to represent a weighted combination of the two:

$$\beta = weight_1 * \gamma_1^3 + weight_2 * \gamma_2^2 = weight_1 * \beta_1 + weight_2 * \beta_2$$

$$min_K(\beta)$$

Once the optimization for the system is ran in Yalmip, we will be left with values for $\beta_1$ and $\beta_2$ which we can then recover the values for $\gamma_1$ and $\gamma_2$ by taking the square root.

## 8.2 Problem 8.b

**SOLUTION:**
Using the constraints from part b, the equally waited optimal-feedback problem was solved by optimizing for:

$$\beta = weight_1 * \beta_1 + weight_2 * \beta_2 = \beta_1 + \beta_2$$

**(i) Matlab Result:**
$min_K ||S(K,P)||^2_{H_2} < 2.0988$ Minimized $H_2$ Gain = 2.0988 $+||S(K,P)||^2_{H_\infty} < 1.8048$ Minimized $H_\infty$ Gain = 1.8048

## 8.3 Problem 8.c

**SOLUTION:**
**(i) Construct the corresponding controller**
*Note: Controller constructed according the same process in 7.b*

$$K = 1e8 * \begin{pmatrix} 0.1718 & -0.0992 & -0.3069 & 0.0356 & -0.1671 & -0.2842 & -0.1334 & -0.0206 & -0.0713 \\ -1.1208 & 0.8249 & 2.7836 & 0.2750 & 1.0416 & 1.9349 & 1.0135 & 0.2476 & 0.5326 \\ 0.3162 & 0.2165 & 0.7141 & 0.0314 & 0.2982 & 0.5384 & 0.2728 & 0.0596 & 0.1441 \\ 0.6676 & -0.9288 & -3.5851 & -1.4153 & -0.5001 & -1.3508 & -0.9573 & -0.4263 & -0.4840 \\ 0.8452 & -0.6288 & -2.1290 & -0.2268 & -0.7836 & -1.4621 & -0.7698 & -0.1911 & -0.4042 \\ 1.4425 & -1.0623 & -3.5856 & -0.3559 & -1.3403 & -2.4905 & -1.3050 & -0.3191 & -0.6858 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

**(ii) Determine and compare the resulting H2 gain to the gains predicted by the LMI**
(Returned from Matlab SS System Analysis) $H_2$ gain = 1.9512

Therefore the controller approx achieves the predicted gain since the following terms are approximately equal:
$H2(LMI) = H2(controller)$; 2.0988>1.9512

**(iii) Determine and compare the resulting $H_inf$ gain to the gains predicted by the LMI**
(Returned from Matlab SS System Analysis)$H_inf$ gain = 1.5708

Therefore the controller approx achieves the predicted gain since the following terms are approximately equal:
$H_\infty(LMI) = Hinf_(controller)$; 1.8048>1.5708

## 8.4 Problem 8.d

**SOLUTION:**

$$\text{Individual Optimization Gains} \leq \text{Mixed Optimization Gains}$$

$$Individual_{H_2} = 1.934 \leq 2.0988 = Mixed_{H_2}$$

$$Individual_{H_\infty} = 1.1070 \leq 1.8048 = Mixed_{H_\infty}$$

This observation makes sense since we cannot truly optimize for two variables at the same time because optima for one gain may not produce a controller that is optimal for the other. In a sense, a compromise is made between the the two objective functions ($\beta = weight_1 * \beta_1 + weight_2 * \beta_2$) so that both are bounded but neither are likely to be optimal. However, there is a chance that the controller that is returned from the optimization is optimal for both (hence the $\leq$ instead of $<$) but this is a product of circumstance.

## 9 Matlab Main Function: Problem 6

```
%% Homework 3 Problem 6
% Regulator Problem and Optimal State-Feedback
clc
clear all
ops = sdpsettings('solver','mosek','verbose',0);
empty=[];

eta=1e-5;                          % Constant to allow for stricly > calculation during P>0

%% System Definition
A= [
-1  1   0   1   0   1;
-1  -2  -1  0   0   1;
1   0   -2  -1  1   1;
-1  1   -1  -2  0   0;
-1  -1  1   1   -2  -1;
0   -1  0   0   -1  -3
];

B =[
0   -1  -1;
0   0   0;
-1  1   1;
-1  0   0;
0   0   1;
-1  1   1
];

zB = zeros(6,3); % Zero matrix in shape of B

C = [
0   1   0   -1  -1  -1;
0   0   0   -1  0   0;
1   0   0   0   -1  0
];
zC = zeros(3,6); % Zero matrix in shape of B

D = [
0   0   0;
0   0   0;
0   0   0
];
zD = zeros(3);


%% Part (a)
disp("#####Problem 6 Part A Output:######")
disp("Use plant and the regulator problem framework to cosntruct 9-matrix representaiton")
% Regulator Framework 9 Matrix Representation
% (work in document)
P = [
A B zB B;
C D zD D;
zC zD zD eye(3);
C D eye(3) D
];
```

```
disp("P=")
disp(P)


%% Part (b)
disp("#####Problem 6 Part B Output:######")
disp("(i) LMI to formulate H_inf optimal state-feedback problem")

% LMI for state feedback (Lecture 9 Theorem 5)

B1 = [B zB];          %6x6
B2 = B;               %6x3
C1 = [C; zC];         %6x6
D11 = [D zD; zD zD];%6x6
D12 = [D; eye(3)];          %6x3

gamma = sdpvar(1,1);
Y=sdpvar(6);
Z = sdpvar(3,6);

optimize_var = gamma;
LMI1 = [Y];
LMI2 = [
    Y*A'+A*Y+Z'*B2'+B2*Z B1 Y*C1'+Z'*D12';
    B1' -gamma*eye(6) D11';
    C1*Y+D12*Z D11 -gamma*eye(6);
    ];

Fun = [
    LMI1>=eta*eye(6);
    LMI2<=-eta*eye(18);
    ];

optimize(Fun,optimize_var,ops);           % Run the optimization

gammaf = value(gamma);           % Return feasable solution gamma
Yf = value(Y);                   % Return feasable solution Y
Zf = value(Z);                   % Return feasable solution Z
F = Zf*inv(Yf);                    % Construct F


disp("F=")
disp(F)
sympref('FloatingPointOutput',true);
Latex_out = latex(sym(F))
disp("(ii) Determine the closed-loop H2 gain")
disp("H_inf gain = ")
disp(gammaf)


%% Part (c)
    ##########################################################################################

disp("#####Problem 6 Part C Output:######")
%Define the closed loop system for the Hinf-optimal controller
Acl = A+B2*F;
Bcl = B1;
Ccl = C1+D12*F;
Dcl = D11;

```

```
disp("(i) Does it exist")
disp("The H2 gain DNE if the Dcl matrix is nonzero.")
disp("Since Dcl = 0, the H2 gain exists")
disp("Dcl=")
disp(Dcl)

disp("(ii) Use an LMI to determine H2 gain of closed loop system (if it exists
    )")

X=sdpvar(size(A,1));
gamma = sdpvar(1,1);

optimize_var = gamma;
LMI1=[X];
LMI2 = [Acl*X+X*Acl'+Bcl*Bcl'];
LMI3=[trace(Ccl*X*Ccl')];

Fun = [
    LMI1>=eta*eye(size(A,1));
    LMI2<=eta*eye(6);
    LMI3<=gamma];              % Combine into single constraint function

optimize(Fun,optimize_var,ops);            % Run the optimization

%gammaf = value(gamma)           % Return feasable solution gamma
Xf = value(X);                   % Return feasable solution X
gammaf = sqrt(value(gamma));           % Return feasable solution gamma

disp("X=")
disp(Xf)
sympref('FloatingPointOutput',true);
Latex_out = latex(sym(Xf))
disp("H2 gain = ")
disp(gammaf)
%sys = ss(Acl,Bcl,Ccl,Dcl);
%g = norm(sys,2)
%% Part (d)*
disp("#####Problem 6 Part D Output:######")
disp("% (i) Use an LMI to formulate and solve the H2-optimal state-feedback")
gamma = sdpvar(1,1);
X=sdpvar(size(A,1));
W = sdpvar(6);
Z = sdpvar(3,6);

optimize_var = gamma;
LMI1 = [X];
LMI2 = [[A B2]*[X;Z] + [X Z']*[A';B2']+ B1*B1'];
LMI3 = [X (C1*X+D12*Z)';C1*X+D12*Z W];
LMI4 = [trace(W)];
Fun = [
    LMI1>=eta*eye(6);
    LMI2<=-eta*eye(6);
    LMI3>=eta*eye(12);
    LMI4<=gamma];              % Combine into single constraint function

optimize(Fun,optimize_var,ops);            % Run the optimization

gammaf = sqrt(value(gamma));            % Return feasable solution gamma
Xf = value(X);                   % Return feasable solution X
Wf = value(W);                   % Return feasable solution W
```

```
Zf = value(Z);                       % Return feasable solution Z
K= Zf*inv(Xf);                        % Construct K

disp("K=")
disp(K)
sympref('FloatingPointOutput',true);
Latex_out = latex(sym(K))
disp("(ii) Determine the closed-loop H2 gain")
disp("H2 gain = ")
disp(gammaf)

%% Part (e)*
disp("#####Problem 6 Part E Output:######")
disp("(i) Use an LMI to determine the Hinf gain of the closed loop system")

% Define closed loop system
Acl = A+B2*K;
Bcl = B1;
Ccl = C1+D12*K;
Dcl = D11;

%Dialated KYP Lemma

% Define variables
gamma = sdpvar(1,1);
X=sdpvar(6);
optimize_var = gamma;

% Construct LMI constraints
LMI1 = [X];
LMI2 = [
    (Acl'*X+X*Acl)  (X*Bcl) (Ccl');
    (Bcl'*X)  (-gamma*eye(6))  (Dcl');
    (Ccl)  (Dcl)  (-gamma*eye(6))
    ];
Fun = [
    LMI1>=eta*eye(6);
    LMI2<=eta*eye(size(LMI2))];               % Combine into single constraint
        function
optimize(Fun,optimize_var,ops);          % Run the optimization

% Return feasable solutions
gammaf = sqrt(value(gamma));           % Return feasable solution gamma
Xf = value(X);                       % Return feasable solution X

% Display results
disp("X=")
disp(Xf)
sympref('FloatingPointOutput',true);
Latex_out = latex(sym(Xf))
disp("Hinf gain of closed loop system")
disp(gammaf)
```

## 10 Matlab Main Function: Problem 7

```
%% Homework 3 Problem 6
% Mixed Norm Optimization
clc
clear all
ops = sdpsettings('solver','mosek','verbose',0);
empty=[];

eta=1e-5;                       % Constant to allow for stricly > calculation
    during P>0

%% System Definition
A= [
-1  1   0   1   0   1;
-1  -2  -1  0   0   1;
1   0   -2  -1  1   1;
-1  1   -1  -2  0   0;
-1  -1  1   1   -2  -1;
0   -1  0   0   -1  -3
];

B =[
0   -1  -1;
0   0   0;
-1  1   1;
-1  0   0;
0   0   1;
-1  1   1
];

zB = zeros(6,3); % Zero matrix in shape of B

C = [
0   1   0   -1  -1  -1;
0   0   0   -1  0   0;
1   0   0   0   -1  0
];
zC = zeros(3,6); % Zero matrix in shape of B

D = [
0   0   0;
0   0   0;
0   0   0
];
zD = zeros(3);
P = [
A B zB B;
C D zD D;
zC zD zD eye(3);
C D eye(3) D
];

% Define the 9-natrux representation components
B1 = [B zB];            %6x6
B2 = B;                 %6x3
C1 = [C; zC];           %6x6
C2 = C;
D11 = [D zD; zD zD];%6x6
D12 = [D; eye(3)];          %6x3
```

```
D21 = [D eye(3)];
D22 = D;
n=6;

%% Part (a)
disp("#####Problem 8 Part A Output:######")
disp("(i) Reformulate the Hinf Output feedback problem so that it minimizes ||S(P,K)||^2_Hinf")

%% Part (b)
disp("#####Problem 8 Part B Output:######")
disp("(i) Use an LMI to formulate and solve the optimal output-feedback problem minimizing both the H2 and Hinf gains, ")
disp("     giving equal weight to each. min_K ||S(P;K)||^2_H2 +||S(P;K)||^2_Hinf")

X1=sdpvar(6);
Y1=sdpvar(6);
Z=sdpvar(6);
An=sdpvar(6,6);
Bn=sdpvar(6,3,'full');
Cn=sdpvar(3,6,'full');
Dn=sdpvar(3,3);
beta1=sdpvar(1);
beta2=sdpvar(1);

%H2 constraints
LMI1 = [(D11+D12*Dn*D21)];
LMI2 = [trace(Z)];
LMI3=[(A*Y1+Y1*A'+B2*Cn+Cn'*B2')   ((A'+An+(B2*Dn*C2)')')  ((B1+B2*Dn*D21));
      (A'+An+(B2*Dn*C2)')  (X1*A+A'*X1+Bn*C2+C2'*Bn')   (X1*B1+Bn*D21);
      ((B1+B2*Dn*D21)')    ((X1*B1+Bn*D21)')  (-eye(6))];

LMI4=[(Y1)   (eye(6))   ((C1*Y1+D12*Cn)');
      (eye(6))   (X1)   ((C1+D12*Dn*C2)');
      (C1*Y1+D12*Cn)         (C1+D12*Dn*C2)   (Z)];

% Hinf constraint
LMI5=[(A*Y1+Y1*A'+B2*Cn+Cn'*B2')   ((A'+An+(B2*Dn*C2)')')          (B1+B2*Dn*D21)  ((C1*Y1+D12*Cn)');
      (A'+An+(B2*Dn*C2)')  (X1*A+A'*X1+Bn*C2+C2'*Bn')         (X1*B1+Bn*D21)   ((C1+D12*Dn*C2)');
      ((B1+B2*Dn*D21)')    (X1*B1+Bn*D21)' (-beta2*eye(6))  ((D11+D12*Dn*D21)');
      (C1*Y1+D12*Cn)         (C1+D12*Dn*C2)   (D11+D12*Dn*D21)          (-beta2*eye(6))];

Fun = [
      LMI1==0;
      LMI2<= beta1;
      LMI3 <=0;
      LMI4 >=0;
      LMI5 <=0];                  % Combine into single constraint function

beta1_weight = 1.0;    % weight for min_K ||S(P;K)||^2_H2
beta2_weight = 1.0;    % weight for min_K||S(P;K)||^2_Hinf
optimize_var= beta1_weight*beta1 + beta2_weight*beta2;

optimize(Fun,optimize_var,ops);            % Run the optimization
```

```
% Return feasable results
X1f = value(X1);                        % Return feasable solution X1
Y1f = value(Y1);                        % Return feasable solution Y1
Anf = value(An);                        % Return feasable solution An
Bnf = value(Bn);                        % Return feasable solution Bn
Cnf = value(Cn);                        % Return feasable solution Cn
Dnf = value(Dn);                        % Return feasable solution Dn
betaf = value(optimize_var);                      % Return feasable solution beta
beta1f = value(beta1);                      % Return feasable solution beta1
beta2f = value(beta2);                      % Return feasable solution beta2

gamma1=sqrt(beta1f);
gamma2=beta2f;

H2_LMI = gamma1;
Hinf_LMI = gamma2;

disp("Minimized H2 Gain = ")
disp(H2_LMI)
disp("Minimized Hinf Gain = ")
disp(Hinf_LMI)

%% Part (c)
disp("#####Problem 8 Part C Output:######")
disp("(i) Construct the corresponding controller")
Y2 = eye(n);
X2 = eye(n)-X1f*Y1f;

K2 = (inv([X2 X1f*B2; zeros(3,n) eye(3)]))*...
     ([Anf Bnf; Cnf Dnf]- [X1f*A*Y1f zeros(6,3); zeros(3,9)])*...
     (inv([Y2' zeros(6,3); C2*Y1f eye(3)]));


Ak2 = K2(1:6,1:6);
Bk2 = K2(1:6,7:9);
Ck2 = K2(7:9,1:6);
Dk2 = K2(7:9,7:9);

Dk = inv(eye(3)+Dk2*D22)*Dk2;
Bk = Bk2*(eye(3)-D22*Dk);
Ck = (eye(3)-Dk*D22)*Ck2;
Ak = Ak2-Bk*inv(eye(3)-D22*Dk)*D22*Ck;

K = [Ak, Bk;Ck, Dk];
disp("K=")
disp(K)
sympref('FloatingPointOutput',true);
Latex_out = latex(sym(K))

% Define closed loop system
Q = inv(eye(3)- D22*Dk);
Acl = [A zeros(6);zeros(6) Ak ] + [B2, zeros(6,3);zeros(6,3),Bk]...
        *inv([eye(3), -Dk; -D22, eye(3)])*[zeros(3,6), Ck; C2,zeros(3,6)];
Bcl = [B1+ B2*Dk*Q*D21; ...
        Bk*Q*D21];
Ccl = [C1, zeros(6)]+ [D12, zeros(6,3)]*inv([eye(3), -Dk;...
        -D22, eye(3)])*[zeros(3,6), Ck; C2,zeros(3,6)];
Dcl = D11+D12*Dk*Q*D21;

sys = ss(Acl, Bcl, Ccl, 0);
```

```
disp("(ii) Determine and compare the resulting H2 gain to the gains predicted by the LMI")
H2_ctrl = norm(sys,2);
disp("(Returned from Matlab System Analysis)")
disp("H_2 gain = ")
disp(H2_ctrl)
disp("Therefore the controller approx achieves the predicted gain since the following terms are approximatly equal:")
disp(strcat("    H2_(LMI) = H2_(controller) => ",string(H2_LMI), "=", string(H2_ctrl)))

disp("(iii) Determine and compare the resulting Hinf gain to the gains predicted by the LMI")
Hinf_ctrl = norm(sys,inf);
disp("(Returned from Matlab System Analysis)")
disp("H_inf gain = ")
disp(Hinf_ctrl)
disp("Therefore the controller approx achieves the predicted gain since the following terms are approximatly equal:")
disp(strcat("    Hinf_(LMI) = Hinf_(controller) => ",string(Hinf_LMI), "=", string(Hinf_ctrl)))

%% Part (d)
%disp("#####Problem 6 Part D Output:######")
%disp("(i) Compare these gains to those produced by pure H2-optimal output feedback.")
%disp("(ii) Compare these gains to those produced by pure Hinf-optimal output feedback. ")
```

## 11 Matlab Main Function: Problem 8

```
%% Homework 3 Problem 6
% Mixed Norm Optimization
clc
clear all
ops = sdpsettings('solver','mosek','verbose',0);
empty=[];

eta=1e-5;                         % Constant to allow for stricly > calculation
    during P>0

%% System Definition
A= [
-1  1   0   1   0   1;
-1  -2  -1  0   0   1;
1   0   -2  -1  1   1;
-1  1   -1  -2  0   0;
-1  -1  1   1   -2  -1;
0   -1  0   0   -1  -3
];

B =[
0   -1  -1;
0   0   0;
-1  1   1;
-1  0   0;
0   0   1;
-1  1   1
];

zB = zeros(6,3); % Zero matrix in shape of B

C = [
0   1   0   -1  -1  -1;
0   0   0   -1  0   0;
1   0   0   0   -1  0
];
zC = zeros(3,6); % Zero matrix in shape of B

D = [
0   0   0;
0   0   0;
0   0   0
];
zD = zeros(3);
P = [
A B zB B;
C D zD D;
zC zD zD eye(3);
C D eye(3) D
];

% Define the 9-natrux representation components
B1 = [B zB];          %6x6
B2 = B;               %6x3
C1 = [C; zC];         %6x6
C2 = C;
D11 = [D zD; zD zD];%6x6
D12 = [D; eye(3)];          %6x3
```

```
D21 = [D eye(3)];
D22 = D;
n=6;

%% Part (a)
disp("#####Problem 8 Part A Output:######")
disp("(i) Reformulate the Hinf Output feedback problem so that it minimizes ||S(P,K)||^2_Hinf")

%% Part (b)
disp("#####Problem 8 Part B Output:######")
disp("(i) Use an LMI to formulate and solve the optimal output-feedback problem minimizing both the H2 and Hinf gains, ")
disp("     giving equal weight to each. min_K ||S(P;K)||^2_H2 +||S(P;K)||^2_Hinf")

X1=sdpvar(6);
Y1=sdpvar(6);
Z=sdpvar(6);
An=sdpvar(6,6);
Bn=sdpvar(6,3,'full');
Cn=sdpvar(3,6,'full');
Dn=sdpvar(3,3);
beta1=sdpvar(1);
beta2=sdpvar(1);

%H2 constraints
LMI1 = [(D11+D12*Dn*D21)];
LMI2 = [trace(Z)];
LMI3=[(A*Y1+Y1*A'+B2*Cn+Cn'*B2')   ((A'+An+(B2*Dn*C2)')')  ((B1+B2*Dn*D21));
      (A'+An+(B2*Dn*C2)')  (X1*A+A'*X1+Bn*C2+C2'*Bn')   (X1*B1+Bn*D21);
      ((B1+B2*Dn*D21)')    ((X1*B1+Bn*D21)')  (-eye(6))];

LMI4=[(Y1)   (eye(6))   ((C1*Y1+D12*Cn)');
      (eye(6))   (X1)   ((C1+D12*Dn*C2)');
      (C1*Y1+D12*Cn)        (C1+D12*Dn*C2)   (Z)];

% Hinf constraint
LMI5=[(A*Y1+Y1*A'+B2*Cn+Cn'*B2')   ((A'+An+(B2*Dn*C2)')')           (B1+B2*Dn*D21)  ((C1*Y1+D12*Cn)');
      (A'+An+(B2*Dn*C2)')  (X1*A+A'*X1+Bn*C2+C2'*Bn')          (X1*B1+Bn*D21)   ((C1+D12*Dn*C2)');
      ((B1+B2*Dn*D21)')    (X1*B1+Bn*D21)'  (-beta2*eye(6))  ((D11+D12*Dn*D21)');
      (C1*Y1+D12*Cn)        (C1+D12*Dn*C2)   (D11+D12*Dn*D21)          (-beta2*eye(6))];

Fun = [
      LMI1==0;
      LMI2<= beta1;
      LMI3 <=0;
      LMI4 >=0;
      LMI5 <=0];               % Combine into single constraint function

beta1_weight = 1.0;    % weight for min_K ||S(P;K)||^2_H2
beta2_weight = 1.0;    % weight for min_K||S(P;K)||^2_Hinf
optimize_var= beta1_weight*beta1 + beta2_weight*beta2;

optimize(Fun,optimize_var,ops);           % Run the optimization
```

```
% Return feasable results
X1f = value(X1);                         % Return feasable solution X1
Y1f = value(Y1);                         % Return feasable solution Y1
Anf = value(An);                         % Return feasable solution An
Bnf = value(Bn);                         % Return feasable solution Bn
Cnf = value(Cn);                         % Return feasable solution Cn
Dnf = value(Dn);                         % Return feasable solution Dn
betaf = value(optimize_var);                     % Return feasable solution beta
beta1f = value(beta1);                   % Return feasable solution beta1
beta2f = value(beta2);                   % Return feasable solution beta2

gamma1=sqrt(beta1f);
gamma2=beta2f;

H2_LMI = gamma1;
Hinf_LMI = gamma2;

disp("Minimized H2 Gain = ")
disp(H2_LMI)
disp("Minimized Hinf Gain = ")
disp(Hinf_LMI)

%% Part (c)
disp("#####Problem 8 Part C Output:######")
disp("(i) Construct the corresponding controller")
Y2 = eye(n);
X2 = eye(n)-X1f*Y1f;

K2 = (inv([X2 X1f*B2; zeros(3,n) eye(3)]))*...
    ([Anf Bnf; Cnf Dnf]- [X1f*A*Y1f zeros(6,3); zeros(3,9)])*...
    (inv([Y2' zeros(6,3); C2*Y1f eye(3)]));


Ak2 = K2(1:6,1:6);
Bk2 = K2(1:6,7:9);
Ck2 = K2(7:9,1:6);
Dk2 = K2(7:9,7:9);

Dk = inv(eye(3)+Dk2*D22)*Dk2;
Bk = Bk2*(eye(3)-D22*Dk);
Ck = (eye(3)-Dk*D22)*Ck2;
Ak = Ak2-Bk*inv(eye(3)-D22*Dk)*D22*Ck;

K = [Ak, Bk;Ck, Dk];
disp("K=")
disp(K)
sympref('FloatingPointOutput',true);
Latex_out = latex(sym(K))

% Define closed loop system
Q = inv(eye(3)- D22*Dk);
Acl = [A zeros(6);zeros(6) Ak ] + [B2, zeros(6,3);zeros(6,3),Bk]...
      *inv([eye(3), -Dk; -D22, eye(3)])*[zeros(3,6), Ck; C2,zeros(3,6)];
Bcl = [B1+ B2*Dk*Q*D21; ...
      Bk*Q*D21];
Ccl = [C1, zeros(6)]+ [D12, zeros(6,3)]*inv([eye(3), -Dk;...
      -D22, eye(3)])*[zeros(3,6), Ck; C2,zeros(3,6)];
Dcl = D11+D12*Dk*Q*D21;

sys = ss(Acl, Bcl, Ccl, 0);
```

```
disp("(ii) Determine and compare the resulting H2 gain to the gains predicted by the LMI")
H2_ctrl = norm(sys,2);
disp("(Returned from Matlab System Analysis)")
disp("H_2 gain = ")
disp(H2_ctrl)
disp("Therefore the controller approx achieves the predicted gain since the following terms are approximatly equal:")
disp(strcat("    H2_(LMI) = H2_(controller) => ",string(H2_LMI), "=", string(H2_ctrl)))

disp("(iii) Determine and compare the resulting Hinf gain to the gains predicted by the LMI")
Hinf_ctrl = norm(sys,inf);
disp("(Returned from Matlab System Analysis)")
disp("H_inf gain = ")
disp(Hinf_ctrl)
disp("Therefore the controller approx achieves the predicted gain since the following terms are approximatly equal:")
disp(strcat("    Hinf_(LMI) = Hinf_(controller) => ",string(Hinf_LMI), "=", string(Hinf_ctrl)))

%% Part (d)
%disp("#####Problem 6 Part D Output:######")
%disp("(i) Compare these gains to those produced by pure H2-optimal output feedback.")
%disp("(ii) Compare these gains to those produced by pure Hinf-optimal output feedback. ")
```