

Controle de Posição do Servomecanismo por meio de Desigualdades Matriciais Lineares (LMIs) - Tempo Contínuo

Objetivo

Projetar e implementar um controlador robusto por meio de formulações expressas em termos de desigualdades matriciais lineares (LMIs) para a malha de posição do servomecanismo considerando incertezas. A realimentação de estados será feita medindo-se diretamente os estados.

1 Fundamentação Teórica

1.1 Sistemas lineares invariantes no tempo incertos

A maioria dos projetos de controle é baseada no uso de um modelo matemático (função de transferência, resposta em frequência, variáveis de estados, etc.), cuja finalidade é caracterizar um sistema real. As diferenças, ou erros, entre o modelo e a planta real são denotadas pelo termo *incerteza*, e qualquer mecanismo usado para expressar essas diferenças é dito *representação da incerteza* (politópica, limitada em norma, intervalar, etc.). De fato, o modelo é meramente uma aproximação matemática da planta, podendo conter diferentes tipos de incerteza como, por exemplo, as decorrentes de linearizações, ruídos, dinâmicas não modeladas, erros de identificação, parâmetros desconhecidos, entre outras. Na maioria dos sistemas práticos, pelo menos dois tipos estão presentes: a dinâmica não modelada e a incerteza paramétrica (devido à falta do conhecimento preciso dos parâmetros). Admitir incertezas no modelo é interessante, pois as mesmas podem causar a deterioração do desempenho e, em casos extremos, tornar o sistema controlado instável.

Na sequência, para ilustrar um sistema com incerteza, tem-se um exemplo familiar, o circuito RLC. Observe que, neste caso, o parâmetro incerto está no valor da resistência.

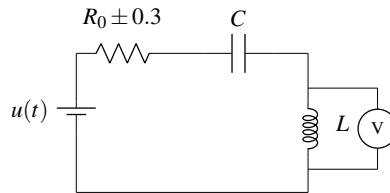


Figura 1: Circuito RLC.

Sabendo que $R = R_0 \pm 0.3$ e definindo a tensão no capacitor e a corrente no indutor como variáveis de estado, tem-se que:

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L} & -\frac{R}{L} \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} u(t) \\ y(t) &= [-1 \quad -R] x(t) + u(t) \end{aligned} \quad (1)$$

Deseja-se saber se o sistema é estável para todos os valores de R , ou seja, $R_0 - 0.3 \leq R \leq R_0 + 0.3$. Para descobrir isto, primeiro, é necessário reescrever o sistema considerando a incerteza. Uma maneira de descrever as matrizes do sistema é na forma convexa. Veja, por exemplo, como ficaria a matriz dinâmica:

$$A(\alpha) = \alpha_1 \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L} & -\frac{(R_0-0.3)}{L} \end{bmatrix} + \alpha_2 \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L} & -\frac{(R_0+0.3)}{L} \end{bmatrix} \quad (2)$$

em que $\alpha_1 \geq 0$, $\alpha_2 \geq 0$ com $\alpha_1 + \alpha_2 = 1$.

Na sequência, é enunciada a definição de uma combinação convexa.

Combinação convexa

Sejam $p_1, p_2 \in \mathbb{R}^2$ dois pontos. Um ponto p qualquer que esteja no segmento de reta entre p_1 e p_2 pode ser representado como

$$p = \alpha_1 p_1 + \alpha_2 p_2, \quad \alpha_1 \geq 0, \alpha_2 \geq 0, \quad \alpha_1 + \alpha_2 = 1. \quad (3)$$

Obs.: O mesmo é válido para matrizes!

1.1.1 Representação politópica

Nesta aula, a representação utilizada para expressar as incertezas do modelo é a politópica, conforme denotado pelo sistema

$$\begin{aligned} \dot{x} &= A(\alpha)x + B(\alpha)u \\ y &= C_y(\alpha)x \end{aligned} \quad (4)$$

sendo $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$, $y \in \mathbb{R}^{n_y}$, respectivamente, os vetores de estados, da entrada de controle e da saída medida. As matrizes do espaço de estados possuem dimensões apropriadas, são incertas e pertencem a um politopo¹, ou seja, qualquer matriz pode ser escrita como uma combinação convexa de N vértices conhecidos na forma

$$M(\alpha) = \sum_{i=1}^N \alpha_i M_i, \quad \alpha \in \Lambda, \quad (5)$$

em que $M(\alpha)$ caracteriza cada uma das matrizes de (4), M_i , $i = 1, \dots, N$, descreve os vértices do politopo e $\alpha = [\alpha_1 \ \alpha_2 \ \dots \ \alpha_N]^\top$ é um vetor de parâmetros invariantes no tempo que corresponde às incertezas paramétricas e pertence ao simplex unitário

$$\Lambda = \left\{ \alpha \in \mathbb{R}^N : \sum_{i=1}^N \alpha_i = 1, \alpha_i \geq 0, i = 1, \dots, N \right\}. \quad (6)$$

A fim de ilustrar o conceito apresentado, considere o politopo com cinco vértices apresentado na Figura 2. Qualquer combinação de α que satisfaça (6) corresponderá a uma combinação de vértices M_i e produzirá um ponto no interior, em uma aresta ou em um vértice do conjunto poliedral representado em azul.

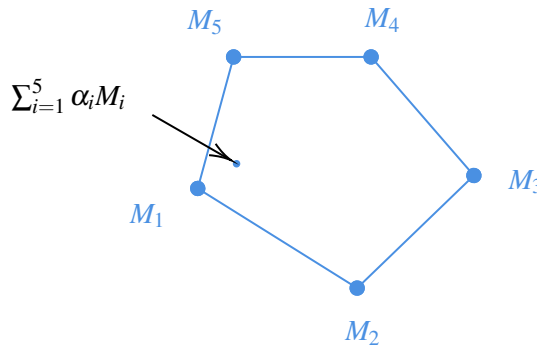


Figura 2: Politopo de matrizes composto por cinco vértices.

¹Um politopo é um conjunto poliedral limitado (convexo e fechado), ou ainda, o envelope convexo de um conjunto finito de vértices.

1.2 Teorema de Lyapunov

O Teorema de Estabilidade de Lyapunov, enunciado a seguir, é usado pela maioria dos trabalhos nas áreas de controle moderno em geral. A ideia fundamental é que, se o sistema é estável em determinada região, então sua energia sempre decresce na ausência de atuação externa.

Teoria de Lyapunov

Seja $\dot{x} = f(x)$ um sistema autônomo tal que $f(0) = 0$, e seja $D \subseteq \mathbb{R}^n$ um domínio contendo $x = 0$. A origem é estável se, e somente se, existir uma função $V(x)$, contínua e diferenciável, chamada de função de Lyapunov, tal que:

- $V(0) = 0$;
- $V(x) > 0$ para todo $x \in D - \{0\}$;
- $\dot{V}(x) \leq 0$ para todo $x \in D$.

Ainda, se

$$\dot{V}(x) < 0 \text{ para todo } x \in D - \{0\},$$

então a origem é assintoticamente estável.

Sistemas Lineares e Invariantes no Tempo

Seja o sistema linear invariante no tempo

$$\dot{x} = Ax.$$

O sistema é **assintoticamente estável** se, e somente se, existir uma matriz simétrica P tal que a função de Lyapunov

$$V(x) = x^\top P x$$

satisfaça as seguintes condições:

- $P \succ 0$ (matriz definida positiva);
- $A^\top P + PA \prec 0$ (matriz definida negativa).

1.3 Regulador Linear Quadrático (LQR)

Seja o sistema dinâmico

$$\dot{x}(t) = Ax(t) + Bu(t).$$

para o qual deseja-se projetar uma lei de controle por realimentação de estados

$$u(t) = Kx(t)$$

que estabilize assintoticamente o sistema em malha fechada.

Para atingir o resultado desejado, pode-se empregar técnicas como a alocação de polos ou o controlador LQR (Linear Quadratic Regulator). Neste último caso, ao invés de se escolher arbitrariamente a posição dos polos em malha fechada do sistema, o ganho de realimentação K é determinado de modo a minimizar a função objetivo quadrática

$$J = \int_0^\infty x(t)^\top Q x(t) + u(t)^\top R u(t) dt$$

sendo $Q = Q^\top \succeq 0$ uma matriz simétrica positiva semi-definida e $R = R^\top \succ 0$ uma matriz simétrica positiva definida. As matrizes Q e R devem ser escolhidas de forma ponderada (e proporcional) entre si, sendo que Q

penaliza o desvio dos estados do sistema em relação à origem e R penaliza o desvio do sinal de controle com relação a zero.

1.4 Transformação de Congruência

Considere duas matrizes simétricas $Q \in \mathbb{R}^{n \times n}$ e $R \in \mathbb{R}^{n \times n}$ e uma matriz $T \in \mathbb{R}^{n \times n}$ não singular ($\det(T) \neq 0$, T invertível). É possível afirmar que:

- As matrizes Q e R são **congruentes** se existe T não singular tal que $Q = T^T R T$.
- Se Q e R são congruentes, então

$$Q \succ 0 \quad \text{se, e somente se,} \quad R \succ 0$$

Prova:

Seja $R \succ 0$, então para todo $x \neq 0$, $x^T R x \succ 0$. Definindo $y = T^{-1}x$, tem-se:

$$x^T R x = y^T T^T R T y = y^T Q y \succ 0, \quad \forall y \neq 0 \implies Q \succ 0$$

1.5 Complemento de Schur

O complemento de Schur é empregado na conversão de restrições convexas com desigualdades não lineares em LMIs.

Complemento de Schur

Considere as matrizes $Q = Q^T$, $R = R^T$ e S com dimensões apropriadas. Então, a LMI

$$\begin{bmatrix} Q & S^T \\ S & R \end{bmatrix} \succ 0 \quad (7)$$

é equivalente a

$$R \succ 0, \quad Q - S^T R^{-1} S \succ 0. \quad (8)$$

Em outras palavras, as desigualdades (8), que incluem uma não linearidade, podem ser representadas como a LMI (7).

1.6 Resolução Numérica de LMIs com Matlab

As LMIs podem ser programadas e resolvidas de forma eficiente utilizando, por exemplo, o Matlab². Para tanto, é necessária a instalação de um interpretador, responsável pela programação das LMIs, e de um resolvidor que, como a própria designação sugere, é responsável por determinar uma solução para o problema. Os dois pacotes mais comumente difundidos são o YALMIP (interpretador) e o SeDuMi (resolvidor), que podem ser baixados nos seguintes links:

- <https://yalmip.github.io/download/>
- https://sedumi.ie.lehigh.edu/?page_id=58

Feito o *download*, deve-se descompactar os pacotes em uma pasta do sistema e adicioná-los ao *path* do Matlab (opção “Set path” no submenu “Environment”). O comando `yalmiptest` pode ser utilizado para verificar a correta instalação dos pacotes.

A seguir é mostrado um passo a passo básico para resolução de um problema envolvendo LMIs.

²Existem também soluções para outras linguagens/ferramentas de código aberto como Python e Octave.

1) Declaração de variáveis

Para definir as variáveis utiliza-se o comando `sdpvar` do pacote YALMIP.

- `P = sdpvar(lin,col,param);`
 - `lin`: Número de linhas;
 - `col`: Número de colunas;
 - `param`: Estrutura da variável: `'symmetric'` para matrizes simétricas, `'full'` para matrizes sem estrutura específica, `'diag'` para matrizes diagonais, ...

```
% Exemplos:
P = sdpvar(2,2,'symmetric');
X = sdpvar(3,4,'full');
```

Por padrão, se o terceiro argumento for omitido, toda matriz quadrada é definida como simétrica e matrizes não quadradas como cheias (sem estrutura).

2) Definir as LMIs

- `LMI1 ≥ 0, LMI2 ≥ 0, LMI3 ≤ 0, ...`;
 - As desigualdades consideradas podem ser $\leq$ ou $\geq$;
 - O uso de desigualdades estritas ($<$ ou $>$) irá gerar erro.

```
% Exemplo:
LMIs = [P>=0, A'*P + P*A<=0];
```

3) Resolução das LMIs

- `sol = optimize(LMIs, obj, options);`
 - `sol`: Variável opcional de saída contendo o status da resolução;
 - `LMIs`: Conjunto de LMIs;
 - `obj`: Função objetivo para problemas de minimização de uma função custo. Caso seja um problema apenas de factibilidade, sem função custo associada, então `obj = 0`;
 - `options`: Parâmetro opcional, usado para configurar o comportamento do resolvidor e para escolha de diferentes resolvidores.

```
% Exemplo:
out = optimize(LMIs,0,sdpsettings('solver','sedumi','verbose',1));
```

4) Verificar o resultado

- `[prim,dual] = checkset(LMIs);`
 - `prim`: Resíduo da solução primal. Caso a solução computada seja factível, então `prim > 0`;
 - `dual`: Resíduo da solução dual;
 - `LMIs`: Conjunto de LMIs;
 - A função `double` é utilizada para converter objetos simbólicos ou variáveis de otimização (`sdpvar`) em valores numéricos.

```

% Exemplo
[p,d] = checkset(LMIs);
if (min(p) > 0)
    P = double(P);
else
    disp('Infactivel');
end

```

1.7 O Servomecanismo

O servomecanismo considerado é apresentado na Figura 3. Há um motor DC com um sensor de velocidade tipo tacogerador acoplado ao mesmo eixo. Há dois estágios de polias com correias, cada um com redução $n = 1/3$, que acopla o eixo do motor ao eixo do potenciômetro. Na configuração controle de velocidade, utiliza-se o eixo do motor/tacogerador, para o controle de posição, utiliza-se o eixo do potenciômetro, onde há também um disco com posição graduada para indicar a posição.

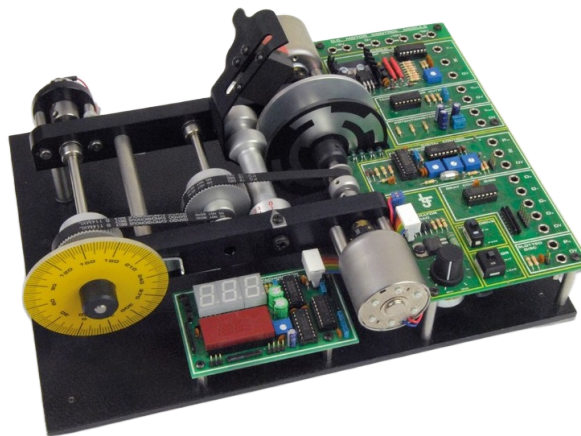


Figura 3: Foto do servomecanismo.

A tensão de armadura é representado por V_M . As tensões nas saídas do tacogerador e do potenciômetro são, respectivamente, representadas por V_T e V_P . A velocidade no eixo do motor é dada por Ω_M , enquanto que a velocidade e a posição no eixo do potenciômetro são dadas por Ω_P e Θ_P . Assume-se que a relação entre tensão de armadura e velocidade no eixo de motor seja um sistema de primeira ordem com ganho K e constante de tempo T . Os ganhos do tacogerador e do potenciômetro são dados por K_T [V/(rad/s)] e K_P [V/rad]. O diagrama linear simplificado do sistema é apresentado na Figura 4.

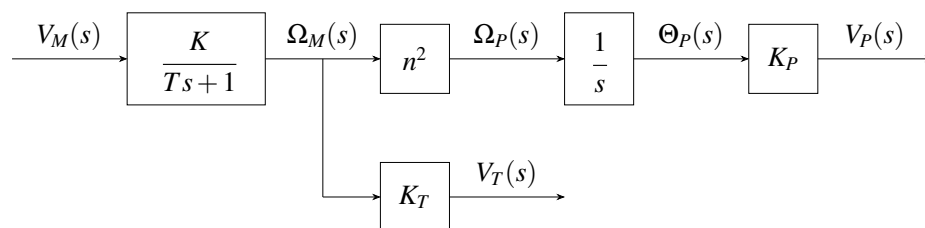


Figura 4: Modelo esquemático do sistema.

Sabe-se, do manual do equipamento, que os valores dos ganhos dos sensores são: $K_P = 1,7$ e $K_T = 0,017$. Após um simples teste de resposta ao degrau em um dos servomecanismos do laboratório, conforme apresentado na Figura 5, foi possível identificar os seguintes parâmetros do modelo: $K = 55,8$ [(rad/s)/V] e $T = 0,29$ [s].

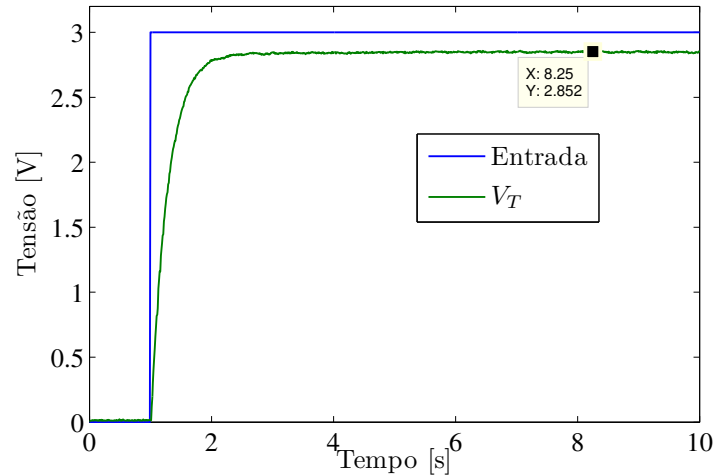


Figura 5: Resposta ao degrau.

Note que os valores de K e T são, a princípio, válidos para o servomecanismo no qual o ensaio foi feito. Para servomecanismos diferentes ou devido à depreciação dos equipamentos, é possível que tais valores não sejam adequados. Neste experimento, buscaremos determinar uma solução que valha para uma faixa maior de valores.

Realimentação de estados com integrador - modelo servo

O problema de rastreamento também pode ser resolvido com a inserção de integradores na malha. Uma forma de inserir integradores consiste em introduzir um novo vetor de estados que integre o erro entre o vetor de saída y e o vetor de comando de entrada r , ambos com m elementos, como apresentado no diagrama da Figura 6.

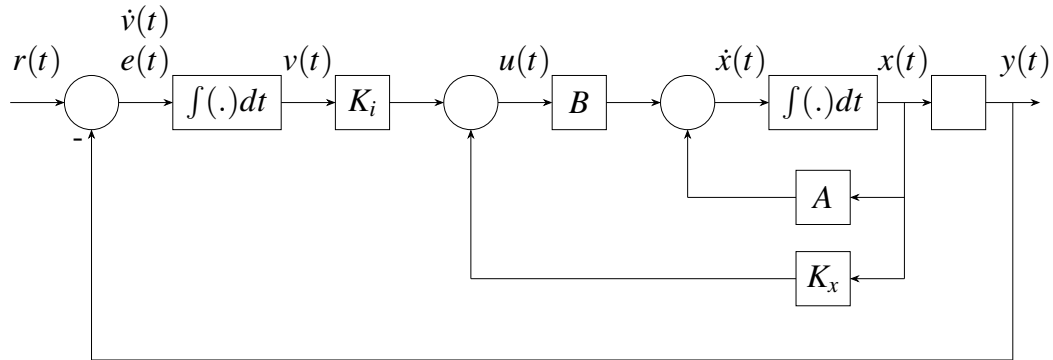


Figura 6: Diagrama de blocos de um sistema em malha fechada com inserção de integradores e realimentação de estados.

A equação de estados do integrador inserido é dada por

$$\dot{v}(t) = e(t) = r(t) - y(t) = r(t) - Cx(t), \quad (9)$$

o que resulta na seguinte equação de estados em malha fechada:

$$\dot{x}(t) = (A + BK_x)x(t) + BK_i v(t). \quad (10)$$

Tem-se, portanto, a seguinte equação para o sistema aumentado:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{v}(t) \end{bmatrix} = \begin{bmatrix} A + BK_x & BK_i \\ -C & 0_m \end{bmatrix} \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} 0_{k \times m} \\ I_m \end{bmatrix} r(t) \quad (11)$$

ou seja,

$$\begin{bmatrix} \dot{x}(t) \\ \dot{v}(t) \end{bmatrix} = \left(\underbrace{\begin{bmatrix} A & 0_{k \times m} \\ -C & 0_m \end{bmatrix}}_{\hat{A}} + \underbrace{\begin{bmatrix} B \\ 0_m \end{bmatrix}}_{\hat{B}} \cdot \underbrace{\begin{bmatrix} K_x & K_i \end{bmatrix}}_{\hat{K}} \right) \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} 0_{k \times m} \\ I_m \end{bmatrix} r(t) \quad (12)$$

Assim, basta determinar o ganho $\hat{K}$ que aloca os polos para o sistema aumentado da Equação (12). Portanto, K_x , representa as n primeiras colunas de $\hat{K}$ e K_i é dado pelas colunas restantes de $\hat{K}$.

2 Atividades

Projetar controladores robustos na forma de espaço de estados no modo servo assegurando um desempenho adequado na presença de incertezas nos parâmetros.

2.1 Modelo em espaço de estados

Seja $x_1 = \theta_p$ e $x_2 = \dot{\theta}_p = \omega_p$, o modelo em espaço de estados é, então, dado por

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{T} \end{bmatrix}}_A \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ \frac{KK_p n^2}{T} \end{bmatrix}}_B V_m$$

$$y = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_C \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Admitindo incertezas nos parâmetros T e K , tais que, $T \in [0.25 \ 0.33]$ e $K \in [45 \ 65]$, pode-se incluir as incertezas no sistema por meio de uma representação politópica. O sistema resultante terá 4 vértices (2^N , sendo N incertezas), podendo ser escrito conforme mostrado a seguir.

Código 1: Construção das matrizes do sistema politópico

```
A = cell(1,4);
B = cell(1,4);
C = cell(1,4);

k = 1;
for T = [0.25 0.33]
    for K = [45 65]
        A{k} = [0 1;
                0 -1/T];
        B{k} = [0;
                n*n*Kp*K/T];
        C{k} = [1 0];
        k = k+1;
    end
end
```

São adotados $n = 1/3$ e $K_p = 1,7$ para os parâmetros apresentados pelo fabricante. Os ganhos K_x e K_i de realimentação de estados serão obtidos por meio de uma condição LMI.

Como trata-se do problema de rastreamento, pode ser interessante ter uma forma de ponderar sobre a energia de controle, por isso relaciona-se a função de Lyapunov com a função quadrática do LQR, fazendo

$$\dot{V}(x) \prec -(x^\top Qx + u^\top Ru).$$

Reorganizando a desigualdade e expandindo os termos, tem-se

$$\begin{aligned} \dot{x}^\top P x + x^\top P \dot{x} + x^\top Q x + u^\top R u < 0 \\ x^\top \left[(\hat{A} + \hat{B}\hat{K})^\top P + P(\hat{A} + \hat{B}\hat{K}) + Q + \hat{K}^\top R \hat{K} \right] x < 0 \end{aligned}$$

Aplicando uma transformação de congruência ao termo entre colchetes, isto é, multiplicando à esquerda e à direita por P^{-1} , obtém-se

$$P^{-1}\hat{A}^\top + P^{-1}\hat{K}^\top \hat{B}^\top + \hat{A}P^{-1} + \hat{B}\hat{K}P^{-1} + P^{-1}QP^{-1} + P^{-1}\hat{K}^\top R \hat{K}P^{-1} < 0$$

e, fazendo as trocas de variáveis, $W = P^{-1}$ e $\hat{Z} = \hat{K}W$, chega-se em

$$W\hat{A}^\top + \hat{Z}^\top \hat{B}^\top + \hat{A}W + \hat{B}\hat{Z} + WQW + \hat{Z}^\top R \hat{Z} < 0$$

Finalmente, aplicando o complemento de Schur duas vezes à condição anterior, chega-se à LMI reportada no teorema a seguir.

Teorema 1 Dados $0 \preceq Q = Q^\top \in \mathbb{R}^{n \times n}$ e um escalar positivo R (sistema SISO), se existirem as matrizes $0 \prec W = W^\top \in \mathbb{R}^{n \times n}$ e $Z \in \mathbb{R}^{nu \times n}$, tais que³

$$\begin{bmatrix} \hat{A}W + W\hat{A}^\top + \hat{B}\hat{Z} + \hat{Z}^\top \hat{B}^\top & WQ^{1/2} & \hat{Z}^\top \\ Q^{1/2}W & -I & 0 \\ \hat{Z} & 0 & -R^{-1} \end{bmatrix} < 0$$

seja satisfeita, então $\hat{K} = \hat{Z}W^{-1}$ é um ganho de controle por realimentação de estados que assegura a estabilidade assintótica do sistema.

2.2 Parte I - Programando a condição

Para programar a condição mostrada no Teorema 1 será utilizado o YALMIP (interpretador) e o SeDuMi (resolvidor) e, para que seja possível adotar as versões mais recentes desses pacotes, será utilizado o Matlab R2025a.

A seguir é mostrada uma forma de programar a condição do Teorema 1, utilizando uma variável do tipo `blkvar`. Esse tipo de variável permite escrever apenas a parte triangular inferior ou superior, além dos elementos da diagonal, de uma matriz simétrica.

No fragmento de código apresentado, N corresponde ao número de vértices do sistema incerto, W e Z são variáveis a serem determinadas e devem ser declaradas com dimensões apropriadas, e A_h e B_h correspondem às matrizes do sistema aumentado (12). Complete as posições $T(1,2)$, $T(1,3)$, $T(2,2)$ e $T(3,3)$ com os elementos enunciados no Teorema 1.

```
for k = 1:N
    T = blkvar; % cria de variaveis estruturadas em blocos
    T(1,1) = Ah{k}*W + W*Ah{k}' + Bh{k}*Z + Z'*Bh{k}';
    T(1,2) =
    T(1,3) =
    T(2,2) =
    T(3,3) =
    T = sdpvar(T);

    LMIs = [LMIs, T <= 0];
end
```

³Note que, neste caso particular, é possível escolher $Q = Q^{1/2}Q^{1/2}$, pois Q é uma matriz *diagonal*. No caso geral, o mais indicado é utilizar algum tipo de decomposição, por exemplo a Decomposição de Cholesky, de tal modo que $Q = LL^\top$. Ademais, note também que as matrizes Q e R são dadas; caso fossem variáveis de otimização, a condição não seria mais linear.

Após programar a condição LMI e obter os ganhos necessários (K_x e K_i), esses valores devem ser informados na versão anterior do Matlab para que sejam aplicados ao servomecanismo. Perceba que é necessário ajustar os valores de Q e R . Portanto, plote o sinal de referência e a saída do sistema e ajuste os valores dos parâmetros até que se obtenha algo aceitável. O diagrama de blocos a ser utilizado no Simulink é mostrado na próxima seção.

2.3 Parte II - Prática

No modo servo com integrador, o diagrama do experimento é apresentado na Figura 7.

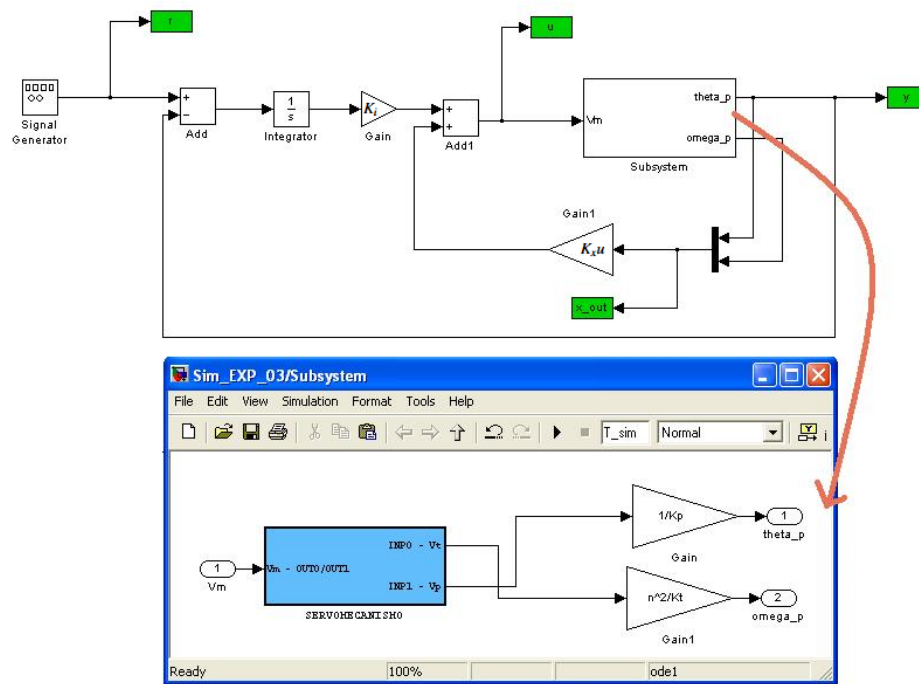


Figura 7: Diagrama de blocos do experimento no Simulink - modo servo com integrador.