

Lilian Fontan de Oliveira

Faculdade de Engenharia Mecânica
Universidade Estadual de Campinas

Relatório Final

ES952 - Trabalho de Graduação II

Controle com Comutação \mathcal{H}_∞ de um Pêndulo Invertido

15 de julho de 2015

Universidade Estadual de Campinas

Campus Universitário Zeferino Vaz, Barão Geraldo, Campinas - SP, 13083-970

Orientadora:

Profa. Dra. Grace S. Deaecto

grace@fem.unicamp.br

Sumário

1	Introdução	4
2	Revisão Bibliográfica	5
2.1	Estabilidade de Lyapunov	5
2.2	Norma \mathcal{H}_2	6
2.2.1	Cálculo Através de Gramianos	7
2.2.2	Cálculo através de LMIs	8
2.3	Norma \mathcal{H}_∞	9
3	Projeto de Controle \mathcal{H}_∞ via Realimentação de Estado	10
3.1	Sistemas Lineares Invariantes no Tempo	10
3.2	Sistemas com Comutação	11
4	Projeto de Controle \mathcal{H}_∞ via Realimentação de Saída	13
5	Controle do Pêndulo Invertido	16
6	Aplicação, Simulação e Análise dos Resultados	18
6.1	Realimentação de Estado	18
6.2	Realimentação de Saída	22
7	Conclusão	23
8	Anexo: Programa do Matlab	25

Resumo do Projeto

Este projeto de pesquisa tem como objetivo realizar o controle com comutação \mathcal{H}_∞ de um pêndulo invertido. Mais especificamente, deseja-se projetar simultaneamente um conjunto de controladores e uma regra de comutação, que seleciona a cada instante de tempo um deles de forma a assegurar estabilidade e desempenho \mathcal{H}_∞ para o sistema global. Se a função de comutação projetada for estritamente consistente, ela sempre fornecerá um desempenho melhor do que aquele de cada subsistema isolado, sendo uma ferramenta muito importante para a solução de problemas de controle multiobjetivos. Estes problemas se caracterizam por apresentar vários critérios diferentes e possivelmente conflitantes, o que corresponde exatamente ao projeto de controle proposto neste trabalho.

O relatório está organizado da seguinte maneira. Primeiramente apresentamos alguns conceitos fundamentais de sistemas dinâmicos, tais como o critério de Lyapunov para o estudo de estabilidade e o cálculo das normas \mathcal{H}_2 e \mathcal{H}_∞ para Sistemas Lineares Invariantes no Tempo (LIT) via Desigualdades Matriciais Lineares¹. Posteriormente, tratamos do controle via realimentação de estado e de saída de sistemas lineares com comutação utilizando referências recentes disponíveis na literatura. Por fim, aplicamos estes métodos no controle de um pêndulo invertido localizado sobre um carrinho e comparamos os resultados com aqueles obtidos utilizando técnicas de controle clássico. Como ficará claro em seguida, a função de comutação projetada é estritamente consistente para o caso de controle via realimentação de estado. Infelizmente, a mesma conclusão não foi obtida para o caso de controle via realimentação de saída. Neste caso, a obtenção de condições que asseguram a existência de uma função de comutação estritamente consistente representa um problema em aberto a ser tratado na literatura.

¹do inglês Linear Matrix Inequalities - LMIs

1 Introdução

Os sistemas dinâmicos com comutação têm se tornado objeto de interesse de diversos estudos acadêmicos. Esses sistemas são formados por diversos subsistemas e uma regra de comutação, ou também conhecida como função de comutação, que seleciona um deles a cada instante de tempo. Evidentemente, estes sistemas podem ser muito interessantes em aplicações práticas, uma vez que o projeto adequado de uma função de comutação permite obter uma trajetória estável, mesmo que todos os subsistemas sejam instáveis, ou ainda, possibilita obter um melhor desempenho quando comparado ao desempenho de cada um dos seus subsistemas estáveis.

Nesse projeto, desejamos aplicar a teoria de comutação no controle de um pêndulo invertido. O objetivo é projetar uma lei de controle com comutação responsável por movimentar o carrinho de forma a manter o pêndulo em equilíbrio na posição vertical com o carrinho posicionado na origem. Além disso, o projeto deve levar em conta um custo garantido \mathcal{H}_∞ de desempenho. Para o caso \mathcal{H}_2 este estudo já foi realizado e está disponível em [10]. Iremos abordar essa questão, primeiramente, considerando que o estado está disponível para realimentação. Nesse caso, nosso objetivo é determinar simultaneamente um conjunto de ganhos matriciais reais $\{K_1, \dots, K_N\}$ e uma função de comutação que permita sintetizar uma lei de controle de forma a assegurar um desempenho melhor do que o desempenho fornecido considerando cada critério isolado. Iremos abordar, também, o projeto via realimentação de saída. Para este caso, o objetivo é projetar um conjunto de controladores dinâmicos de ordem completa simultaneamente à regra de comutação de modo a assegurar um custo \mathcal{H}_∞ para o sistema em malha fechada.

2 Revisão Bibliográfica

2.1 Estabilidade de Lyapunov

Considere um sistema linear invariante no tempo, cuja representação no espaço de estado é dada por:

$$\dot{x}(t) = Ax(t) + Hw(t), x(0) = 0 \quad (1)$$

$$z(t) = Ex(t) + Gw(t) \quad (2)$$

onde $x(t) \in \mathbb{R}^n$ representa o vetor de estado, $w(t) \in \mathbb{R}^r$ representa a entrada externa e $z(t) \in \mathbb{R}^p$ representa a saída controlada. A função de transferência deste sistema é a seguinte

$$H(s) = E(sI - A)^{-1}H + G \quad (3)$$

Existem diversos critérios para analisar a estabilidade de um sistema dinâmico. O objeto de estudo neste momento é o critério de estabilidade de Lyapunov. Para entender este critério, podemos pensar na energia de um sistema, por exemplo, um sistema RLC ou um sistema massa-mola-amortecedor. Se pensarmos na energia armazenada em um sistema mecânico estável sem entrada externa, podemos afirmar que ela começa com um certo valor inicial e tende a zero quando $t \geq 0$ tende ao infinito. Logo, a função de energia é uma função contínua e sempre positiva. Além disso, como o sistema é estável, esta função é sempre decrescente, o que nos permite afirmar que sua derivada é negativa para todo $t \geq 0$ e somente será nula no ponto de equilíbrio $x = 0$, veja [11]. Estas são características de uma função de Lyapunov resumidas a seguir:

- $v(x) > 0, x \neq 0$
- $v(x) = 0, x = 0$
- $v(x)$ é contínua em relação a todos os componentes de x

Esta função é utilizada para o estudo da estabilidade global de um sistema dinâmico segundo o critério de Lyapunov, apresentado no seguinte teorema

Teorema 1 *Considere um sistema dinâmico com ponto de equilíbrio $x = 0$ e seja uma função de Lyapunov $v(x)$, ilimitada para todo $x(t) \in \mathbb{R}^n$ ilimitado, se*

- $\dot{v}(x) = 0$ para $x = 0$
- $\dot{v}(x) < 0$ para todo $x \neq 0$

Então o ponto de equilíbrio $x = 0$ é globalmente assintoticamente estável.

Um estudo mais aprofundado requer a leitura de [8] e [11].

Para o estudo da estabilidade vamos considerar o sistema mais simples definido pela equação (1) com $w(t) = 0$ para todo $t \geq 0$, ou seja, o sistema linear $\dot{x} = Ax$, cujo único ponto de equilíbrio é a origem $x = 0$.

Podemos concluir do que foi apresentado até agora que o desafio para determinar se o sistema é ou não estável está na tarefa de encontrar uma função de Lyapunov adequada. Para sistemas LIT uma função de Lyapunov comumente utilizada na literatura é a função quadrática apresentada a seguir

$$v(x) = x'Px \quad (4)$$

onde $P \in \mathbb{R}^{n \times n}$ é uma matriz simétrica definida positiva. Sua derivada em relação ao tempo fornece

$$\begin{aligned} \dot{v}(x) &= \dot{x}'Px + x'P\dot{x} \\ &= x'(A'P + PA)x \end{aligned} \quad (5)$$

Para atender ao Teorema 1, queremos que $\dot{v}(x) < 0$, ou seja

$$A'P + PA < 0 \quad (6)$$

que possui infinitas soluções. Para uma única solução, temos que resolver a equação matricial linear $A'P + PA = -Q$, sendo Q uma matriz simétrica definida positiva dada. Este conceito é melhor explicado no teorema abaixo:

Teorema 2 *Os autovalores de A têm parte real negativa, ou seja, o sistema em estudo é globalmente assintoticamente estável, se e somente se, dada uma matriz simétrica definida positiva Q , a equação de Lyapunov $A'P + PA = -Q$ tiver uma única solução $P = P'$ definida positiva.*

A prova deste teorema pode ser encontrada em [8].

2.2 Norma \mathcal{H}_2

Embora a norma \mathcal{H}_2 não tenha sido utilizada para o projeto de controle do pêndulo invertido, ela juntamente com a norma \mathcal{H}_∞ representam conceitos básicos e fundamentais no estudo de sistemas dinâmicos. Na verdade, o controle do pêndulo invertido utilizando a norma \mathcal{H}_2 já foi realizado e os resultados obtidos estão disponíveis em [10].

A norma \mathcal{H}_2 é definida para todas as funções de transferência $H(s)$ racionais, estritamente próprias

e analíticas no semiplano complexo direito fechado e é dada por:

$$\|H(s)\|_2^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{tr}(H(-j\omega)'H(j\omega))d\omega \quad (7)$$

Podemos calcular essa norma de dois modos distintos: utilizando gramianos de controlabilidade / observabilidade ou através de LMIs. Estudaremos cada um dos casos separadamente. Primeiramente consideraremos seu cálculo através de gramianos.

2.2.1 Cálculo Através de Gramianos

Utilizando o teorema de Parseval disponível em [8], temos

$$\|H(s)\|_2^2 = \int_0^{\infty} \text{tr}(h(t)'h(t))dt \quad (8)$$

onde $h(t)$ é a resposta ao impulso do sistema, ou seja, a Transformada de Laplace inversa da função de transferência $H(s)$, dada a seguir

$$h(t) = Ee^{At}H + G\delta(t), \forall t \geq 0 \quad (9)$$

A prova pode ser encontrada em [8]. Substituindo a equação (9) na equação (8) temos que:

$$\|H(s)\|_2^2 = \int_0^{\infty} \text{tr}((H'e^{A't}E' + G'\delta(t))(Ee^{At}H + G\delta(t)))dt \quad (10)$$

Logo,

$$\|H(s)\|_2^2 = \text{tr}(H'(\int_0^{\infty} e^{A't}E'Ee^{At}dt)H) + \text{tr}(H'E'G) + \text{tr}(G'EH) + \text{tr}(G'G) \int_0^{\infty} \delta(t)^2 dt \quad (11)$$

Podemos notar que a última integral diverge devido ao termo que depende de $\delta(t)^2$ e, portanto, a única maneira de obter uma norma \mathcal{H}_2 finita é fazendo $G = 0$, ou seja, o sistema deve ser estritamente próprio. Assim, podemos concluir que:

$$\|H(s)\|_2^2 = \text{tr}(H'(\int_0^{\infty} e^{A't}E'Ee^{At}dt)H) \quad (12)$$

Sendo $P_0 = \int_0^{\infty} e^{A't}E'Ee^{At}dt$ o gramiano de observabilidade solução da equação de Lyapunov

$$A'P_0 + P_0A + E'E = 0 \quad (13)$$

Note que este gramiano é uma matriz simétrica definida positiva desde que o par (A, E) seja observável.

Um sistema dinâmico $\dot{x} = Ax$, $z = Ex$ é dito observável se o conhecimento de $z(t)$ para todo $0 \leq t \leq \tau$ com $\tau > 0$ arbitrário permitir determinar numericamente $x(0) \in \mathbb{R}^n$.

Utilizando exatamente o mesmo raciocínio e a propriedade de circularidade do operador traço, podemos calcular a mesma norma trocando a ordem da resposta ao impulso $h(t)$ e seu transposto, ou seja:

$$\|H(s)\|_2^2 = \int_0^\infty \text{tr}(h(t)h(t)') \quad (14)$$

Refazendo os cálculos, temos que a norma \mathcal{H}_2 também pode ser obtida por:

$$\|H(s)\|_2^2 = \text{tr}(E(\int_0^\infty e^{At}HH'e^{A't}dt)E') \quad (15)$$

Sendo $P_c = \int_0^\infty e^{At}HH'e^{A't}dt$ o gramiano de controlabilidade solução da equação de Lyapunov

$$AP_c + P_cA' + HH' = 0 \quad (16)$$

Este gramiano é uma matriz simétrica definida positiva desde que o par (A, H) seja controlável. O sistema dinâmico $\dot{x} = Ax + Hu$ é dito controlável se existir um controle na forma $u = -Kx$ que possa alocar todos os polos em posições arbitrárias no plano complexo \mathbb{C} . Um conhecimento mais aprofundado sobre os gramianos requer a leitura das referências [1] e [2].

2.2.2 Cálculo através de LMIs

Ao invés de resolver um conjunto de equações polinomiais podemos calcular a norma \mathcal{H}_2 através da solução de um problema de programação convexa. Considere que os pares (P_1, Q_1) e (P_2, Q_2) satisfazem a equação de Lyapunov $A'P_1 + P_1A + Q_1 = 0$ e $A'P_2 + P_2A + Q_2 = 0$, respectivamente, podemos afirmar que se $Q_1 < Q_2$, então $P_1 < P_2$. Este resultado pode ser obtido diretamente do gramiano de observabilidade

$$P_1 = \int_0^\infty e^{A't}Q_1e^{At}dt \quad (17)$$

Logo, para uma matriz $S > 0$ arbitrária, temos $Q_2 = Q_1 + S > Q_1$ e, portanto

$$P_1 < \int_0^\infty e^{A't}(Q_1 + S)e^{At}dt \quad (18)$$

$$< P_2 \quad (19)$$

Assim, temos que qualquer solução $P > 0$ da desigualdade

$$A'P + PA + E'E < 0 \quad (20)$$

pode ser escrita como $A'P + PA + E'E = -S$ sendo S uma matriz simétrica definida positiva. Logo, temos $P > P_0$, sendo P_0 obtida de (13) e obtemos

$$\|H(s)\|_2^2 = \text{tr}(H'P_0H) < \text{tr}(H'PH) \quad (21)$$

Logo, a norma \mathcal{H}_2 pode ser obtida resolvendo-se o seguinte problema de otimização convexa

$$\|H(s)\|_2^2 = \inf_{P>0} \{\text{tr}(H'PH) | A'P + PA + E'E < 0\} \quad (22)$$

Alternativamente, podemos calcular a norma \mathcal{H}_2 da seguinte maneira

$$\|H(s)\|_2^2 = \inf_{P>0} \{\text{tr}(EPE') | AP + PA' + HH' < 0\} \quad (23)$$

As referências [1] e [2] apresentam a demonstração do exposto anteriormente.

2.3 Norma \mathcal{H}_∞

Consideramos o sistema linear invariante no tempo dado pelas equações (1)-(2) bem como sua função de transferência dada pela equação (3). A norma \mathcal{H}_∞ é definida para todas as funções de transferência analíticas no semiplano complexo direito fechado e é dada por

$$\|H(s)\|_\infty = \max_{\omega \in \mathbb{R}} \sigma_{max}(H(j\omega)) \quad (24)$$

com os valores singulares $\sigma_i, i = 1, \dots, \min(r, p)$ calculados como sendo

$$\sigma_i = \sqrt{\lambda_i(H(j\omega) \sim H(j\omega))} = \sqrt{\lambda_i(H(j\omega)H(j\omega) \sim)} \quad (25)$$

onde o símbolo “ \sim ” indica o conjugado transposto da função de transferência em consideração. Este conceito nos diz que, para um sistema SISO², a norma \mathcal{H}_∞ é igual ao valor máximo do diagrama de Bode de módulo.

Se considerarmos a entrada exógena como qualquer sinal pertencente ao conjunto \mathcal{L}_2 , ou seja, qualquer

²Single Input, Single Output

sinal em que a integral

$$\int_0^{\infty} w(\tau)'w(\tau)d\tau < +\infty \quad (26)$$

é finita, então a norma \mathcal{H}_{∞} ao quadrado do sistema (1)-(2) pode ser definida como o menor valor de ρ tal que

$$\sup_{w(t) \in \mathcal{L}_2} \|z(t)\|_2^2 - \rho \|w(t)\|_2^2 < 0 \quad (27)$$

De fato, adotando a função de Lyapunov quadrática $v(x) = x'Px$, e impondo

$$\dot{v} + z'z - \rho w'w < 0 \quad (28)$$

podemos verificar que após integrada de ambos os lados e, levando em conta que $v(x(\infty)) = 0$ e $v(x(0)) = 0$, $x(0) = 0$, obtemos (27). Sendo assim, se existir uma matriz $P > 0 \in \mathbb{R}^{n \times n}$ satisfazendo a seguinte LMI

$$\begin{bmatrix} H_e\{A'P\} & \bullet & \bullet \\ H'P & -\rho I & \bullet \\ E & G & -I \end{bmatrix} < 0, P > 0 \quad (29)$$

então, temos que $\|H\|_{\infty}^2 = \inf_{\{P>0, \rho>0\} \in \Phi} \rho$ em que Φ é o conjunto de soluções factíveis de (29). Importante registrar que $H_e\{\cdot\}$ representam o operador hermitiano $H_e\{U\} = U + U'$ e \bullet representa o bloco simétrico de uma matriz simétrica. Resultados complementares, bem como uma discussão mais detalhada do exposto anteriormente pode ser também encontrada em [1] e [2].

3 Projeto de Controle \mathcal{H}_{∞} via Realimentação de Estado

3.1 Sistemas Lineares Invariantes no Tempo

Considere o sistema mais geral descrito pelas seguintes equações no espaço de estado

$$\dot{x} = Ax + Bu + Hw, x(0) = 0 \quad (30)$$

$$z = Ex + Fu + Gw \quad (31)$$

em que, além das variáveis já definidas, $u \in \mathbb{R}^m$ é a entrada de controle do sistema. Nosso objetivo, neste momento, é generalizar os resultados anteriores para tratar do projeto de uma lei de controle

$$u = Kx \quad (32)$$

de forma a estabilizar o sistema em malha fechada (30)-(31) assegurando um desempenho \mathcal{H}_∞ . Assim, conectando a entrada de controle (32) no sistema (30)-(31), obtemos

$$\dot{x} = (A + BK)x + Hw, \quad x(0) = 0 \quad (33)$$

$$z = (E + FK)x + Gw \quad (34)$$

Aplicando as condições apresentadas em (29) neste sistema, obtemos

$$\begin{bmatrix} H_e\{P(A + BK)\} & \bullet & \bullet \\ H'P & -\rho I & \bullet \\ (E + FK) & G & -I \end{bmatrix} < 0, \quad P > 0 \quad (35)$$

que, como pode ser observado, não é uma LMI devido ao produto de matrizes P e K . Entretanto, multiplicando ambos os lados da desigualdade resultante por $\text{diag}\{P^{-1}, I, I\}$ e definindo as variáveis $S = P^{-1}$ e $Y = KP^{-1}$, obtemos que o seguinte problema de otimização

$$\|H\|_\infty^2 = \inf_{S>0, Y, \rho>0} \rho \quad (36)$$

sujeito à seguinte LMI

$$\begin{bmatrix} H_e\{AS + BY\} & \bullet & \bullet \\ H' & -\rho I & \bullet \\ ES + FY & G & -I \end{bmatrix} < 0, \quad P > 0 \quad (37)$$

fornece o ganho de realimentação de estado $K = YS^{-1}$ que garante a estabilidade da origem do sistema (30)-(31) e um desempenho \mathcal{H}_∞ dado por $\sqrt{\rho}$. Este problema de otimização foi implementado e o algoritmo utilizado está disponível no apêndice “norma_subistemas_isolados.m”.

3.2 Sistemas com Comutação

Nosso objetivo neste trabalho é projetar um controle com comutação para um sistema LIT, de forma a obter um desempenho melhor do que aquele resultante do problema de otimização (36) que utiliza a lei de controle (32). Antes, porém, vamos apresentar a teoria de sistemas com comutação que será importante para a obtenção dos nossos objetivos. Os resultados apresentados a seguir estão disponíveis

em [3], [4] e [7].

Considere um sistema linear com comutação com a seguinte realização no espaço de estado

$$\dot{x}(t) = A_\sigma x(t) + B_\sigma u(t) + H_\sigma w(t), \quad x(0) = 0 \quad (38)$$

$$z(t) = E_\sigma x(t) + F_\sigma u(t) + G_\sigma w(t) \quad (39)$$

Conectando a entrada de controle $u(x(t)) = K_\sigma x(t)$ em (38)-(39), obtemos o sistema em malha fechada

$$\dot{x}(t) = (A_\sigma + B_\sigma K_\sigma)x(t) + H_\sigma w(t), \quad x(0) = 0 \quad (40)$$

$$z(t) = (E_\sigma + F_\sigma K_\sigma)x(t) + G_\sigma w(t) \quad (41)$$

em que $\sigma(t) : t \geq 0 \rightarrow \mathbb{K} = \{1, \dots, N\}$ é a função de comutação que seleciona a cada instante de tempo um determinado subsistema dentre os N disponíveis, os quais são definidos a seguir:

$$\mathcal{G}_i = \begin{bmatrix} A_i & B_i & H_i \\ E_i & F_i & G_i \end{bmatrix}, \quad i \in \mathbb{K} \quad (42)$$

O objetivo principal é projetar uma função de comutação $\sigma(x) : \mathbb{R}^n \rightarrow \mathbb{K}$ e ganhos de realimentação de estado $\{K_1, \dots, K_N\}$ de forma a fazer com que a origem seja um ponto de equilíbrio globalmente assintoticamente estável, assegurando um custo garantido \mathcal{H}_∞ . Para isso, devemos assegurar que para um escalar $\rho > 0$ dado, a desigualdade abaixo é satisfeita:

$$\sup_{w \in \mathcal{L}_2} \|z\|_2^2 - \rho \|w\|_2^2 < 0 \quad (43)$$

As condições estão apresentadas no teorema seguinte retirado de [3]. Antes de mostrá-lo, devemos definir a subclasse de Matrizes de Metzler utilizada. Uma matriz quadrada $\Pi = \{\pi_{ji}\} \in \mathbb{R}^{N \times N}$ pertence à classe de matrizes de Metzler se possuir todos os seus elementos fora da diagonal principal não negativos. Adotaremos uma subclasse de matrizes de Metzler pertencente ao subconjunto \mathcal{M}_c que atende à seguinte condição adicional $\sum_{j=1}^N \pi_{ji} = 0$.

Teorema 3 Para $\rho > 0$ dado, se existirem matrizes simétricas S_i, T_{ij} , matrizes Y_i para $j \neq i \in \mathbb{K} \times \mathbb{K}$

e uma matriz de Metzler $\Pi \in \mathcal{M}_c$ satisfazendo as desigualdades de Ricatti-Metzler

$$\begin{bmatrix} H_e\{A_i S_i + B_i Y_i\} + \sum_{j \neq i=1}^N \pi_{ji} T_{ij} & \bullet & \bullet \\ H'_i & -\rho I & \bullet \\ (E_i S_i + F_i Y_i) & G_i & -I \end{bmatrix} < 0, i \in \mathbb{K} \quad (44)$$

$$\begin{bmatrix} T_{ij} + S_i & \bullet \\ S_i & S_j \end{bmatrix} > 0, i \neq j \in \mathbb{K} \times \mathbb{K} \quad (45)$$

então, a regra de comutação dada por:

$$\sigma(x(t)) = \arg \min_{i \in \mathbb{K}} x(t)' S_i^{-1} x(t) \quad (46)$$

e os ganhos de realimentação de estado $K_i = Y_i S_i^{-1}$ para todo $i \in \mathbb{K}$ asseguram que a origem do sistema em malha fechada seja um ponto de equilíbrio globalmente assintoticamente estável satisfazendo a condição da norma \mathcal{H}_∞ definida anteriormente pela desigualdade (43).

Prova: A prova deste teorema pode ser encontrada em [3] e [7]. □

As condições deste teorema foram implementadas e estão disponíveis no apêndice, na função “controle_comutacao_estado.m”.

4 Projeto de Controle \mathcal{H}_∞ via Realimentação de Saída

Neste momento faremos a mesma análise de controle \mathcal{H}_∞ mas agora vamos supor que o estado não está disponível para realimentação. O sistema a ser considerado apresenta a seguinte realização no espaço de estado

$$\dot{x}(t) = A_\sigma x(t) + B_\sigma u(t) + H_\sigma w(t), \quad x(0) = 0 \quad (47)$$

$$y(t) = C_\sigma x(t) + D_\sigma w(t) \quad (48)$$

$$z(t) = E_\sigma x(t) + F_\sigma u(t) + G_\sigma w(t) \quad (49)$$

em que $y(t) \in \mathbb{R}^s$ é a saída medida. Assim como no caso de realimentação de estado, o objetivo principal é projetar uma função de comutação $\sigma(y) : \mathbb{R}^s \rightarrow \mathbb{K}$ e um controlador de ordem completa de forma a fazer com que a origem do sistema (47)-(49) seja um ponto de equilíbrio globalmente assintoticamente estável, assegurando um custo garantido \mathcal{H}_∞ . Para isso, devemos assegurar que para qualquer ρ dado, a desigualdade (27) seja respeitada. O controlador a ser projetado é de ordem completa e é descrito pelas

seguintes equações no espaço de estado:

$$\hat{x}(t) = \hat{A}_{\sigma(t)}\hat{x}(t) + \hat{B}_{\sigma(t)}y(t), \hat{x}(0) = 0 \quad (50)$$

$$u(t) = \hat{C}_{\sigma(t)}\hat{x}(t) \quad (51)$$

sendo que \hat{x} é o estado do controlador. Conectando (50)-(51) no sistema (47)-(49), obtemos o seguinte sistema aumentado:

$$\tilde{x}(t) = \tilde{A}_{\sigma(t)}\tilde{x}(t) + \tilde{H}_{\sigma(t)}w(t) \quad (52)$$

$$z(t) = \tilde{E}_{\sigma(t)}\tilde{x}(t) + \tilde{G}_{\sigma(t)}w(t) \quad (53)$$

sendo $\tilde{x}(t) = [x(t)' \ \hat{x}(t)']' \in \mathbb{R}^{2n_x}$ e

$$\tilde{A}_i = \begin{bmatrix} A_i & B_i\hat{C}_i \\ \hat{B}_iC_i & \hat{A}_i \end{bmatrix}, \tilde{H}_i = \begin{bmatrix} H_i \\ \hat{B}_iD_i \end{bmatrix}, \tilde{E}_i = \begin{bmatrix} E_i' \\ \hat{C}_i'F_i' \end{bmatrix}', \tilde{G}_i = G_i \quad (54)$$

O objetivo final é determinar as variáveis citadas acima ($\hat{A}_i, \hat{B}_i, \hat{C}_i$), juntamente com a regra de comutação $\sigma(y)$ de forma a garantir estabilidade e desempenho do sistema em malha fechada. O teorema a seguir apresenta as condições para o controle \mathcal{H}_∞ do sistema (52)-(53) considerando, primeiramente, que o estado \tilde{x} está disponível para realimentação.

Teorema 4 Para $\rho > 0$ dado, a regra de comutação

$$\sigma(x(t)) = \arg \min_{i \in \mathbb{K}} \tilde{x}(t)' \tilde{P}_i \tilde{x}(t) \quad (55)$$

é globalmente estabilizante e o sistema com comutação (52)-(53) satisfaz a restrição (27) se existirem matrizes $\tilde{P}_i > 0$ para todo $i \in \mathbb{K}$ e uma matriz de Metzler $\Pi \in \mathcal{M}_c$ satisfazendo as desigualdades de Riccati-Metzler

$$\begin{bmatrix} H_e\{\tilde{A}_i'\tilde{P}_i\} + \sum_{j \in \mathbb{K}} \pi_{ji}\tilde{P}_j & \bullet & \bullet \\ \tilde{H}_i'\tilde{P}_i & -\rho I & \bullet \\ \tilde{E}_i & \tilde{G}_i & -I \end{bmatrix} < 0, i \in \mathbb{K} \quad (56)$$

Prova: A prova deste teorema está disponível em [3] e [5] □

Note que a implementação da função de comutação (55) exige que o conhecimento do estado \tilde{x} . No entanto, o estado do sistema $x \in \mathbb{R}^n$ não está disponível e, desta forma, a matriz definida positiva

$\tilde{P}_i \in \mathbb{R}^{2n_x \times 2n_x}$ deve ter a seguinte estrutura particular

$$\tilde{P}_i = \begin{bmatrix} Y & V \\ V' & \hat{Y}_i \end{bmatrix}, \det(V) \neq 0 \quad (57)$$

fazendo com que a regra de comutação

$$\arg \min_{i \in \mathbb{K}} \tilde{x}' \tilde{P}_i^{-1} \tilde{x} = \arg \min_{i \in \mathbb{K}} \hat{x}' \hat{Y}_i \hat{x} \quad (58)$$

$$= \sigma(\hat{x}) \quad (59)$$

dependa somente da saída medida através do estado do controlador $\hat{x} \in \mathbb{R}^n$. O teorema a seguir, disponível em [3] e [5], apresenta condições que permitem a obtenção das matrizes do controlador e a função de comutação, sem que nenhum conservadorismo seja introduzido no resultado do Teorema 4.

Teorema 5 *Seja $\rho > 0$ dado. Considerando o sistema (50)-(51), existem matrizes \hat{A}_i , \hat{B}_i e \hat{C}_i para todo $i \in \mathbb{K}$ tais que as desigualdades (56) são satisfeitas para alguma matriz \tilde{P}_i definida positiva com a estrutura dada por (57) se e somente se existirem matrizes simétricas Y , X_i e T_{ij} , matrizes M_i , L_i , W_i para todo $i \neq j \in \mathbb{K} \times \mathbb{K}$ e uma matriz Metzler $\Pi \in \mathcal{M}_c$ satisfazendo as desigualdades matriciais*

$$\begin{bmatrix} H_e\{Y A_i + L_i C_i\} & \bullet & \bullet & \bullet \\ A_i + M_i' & H_e\{A_i X_i + B_i W_i\} + \sum_{j \neq i=1}^N \pi_{ji} T_{ij} & \bullet & \bullet \\ H_i' Y + D_i' L_i' & H_i' & -\rho I & \bullet \\ E_i & E_i X_i + F_i W_i & G_i & -I \end{bmatrix} < 0, i \in \mathbb{K} \quad (60)$$

$$\begin{bmatrix} T_{ij} + X_i & \bullet & \bullet \\ X_i & X_j & \bullet \\ I & I & Y \end{bmatrix} > 0, i \neq j \in \mathbb{K} \times \mathbb{K} \quad (61)$$

Assumindo que as desigualdades acima foram satisfeitas e que V é uma matriz arbitrária não singular, temos que a regra de comutação é dada por

$$\sigma(\hat{x}) = \arg \min_{i \in \mathbb{K}} \hat{x}' (Y - X_i^{-1})^{-1} V \hat{x} \quad (62)$$

e o controlador dinâmico de ordem completa é definido pelas seguintes matrizes:

$$\hat{A}_i = V^{-1}(M_i - Y A_i X_i - Y B_i W_i - L_i C_i X_i)(I - Y X_i)^{-1} V \quad (63)$$

$$\hat{B}_i = V^{-1} L_i \quad (64)$$

$$\hat{C}_i = W_i (I - Y X_i)^{-1} V \quad (65)$$

Este controlador dinâmico faz com que a origem $x = 0$ do sistema em malha fechada seja um ponto de equilíbrio globalmente assintoticamente estável, satisfazendo a desigualdade 27.

As condições desse teorema foram implementadas e estão disponíveis no apêndice, na função “controle_comutacao_saida.m”.

5 Controle do Pêndulo Invertido

Considere o pêndulo invertido mostrado na Figura 1. Primeiramente, desejamos obter o modelo matemático deste sistema. Para isso, iremos determinar as equações de deslocamento horizontal do carro, deslocamento horizontal do pêndulo e o deslocamento vertical do pêndulo. A equação do deslocamento horizontal do carro é dada por:

$$M \frac{d^2 \chi}{dt^2} = T \cos(\phi) + F \quad (66)$$

$$M \ddot{\chi} = T \cos(\phi) + F \quad (67)$$

A equação do deslocamento horizontal do pêndulo é dada por:

$$m \frac{d^2}{dt^2} (\chi + l \cos(\phi)) + T \cos(\phi) = 0 \quad (68)$$

$$m \ddot{\chi} - ml \ddot{\phi} \sin(\phi) - ml \dot{\phi}^2 \cos(\phi) + T \cos(\phi) = 0 \quad (69)$$

A equação do deslocamento vertical do pêndulo é dada por:

$$m \frac{d^2}{dt^2} (l \sin \phi) + mg + T \sin \phi = 0 \quad (70)$$

$$ml \ddot{\phi} \cos \phi - ml \dot{\phi}^2 \sin(\phi) + T \sin \phi + mg = 0 \quad (71)$$

Das equações (67) e (69) isolando a força de Tração "T" em (67) e substituindo em (69) após linearizar o resultado em torno do ponto de interesse $(x_0, \theta_0) = (0, +90^\circ)$, obtemos (75). Ademais, multiplicando (69) por $-\sin(\phi)$ e (71) por $\cos(\phi)$ e somando as equações, obtemos

$$-m \ddot{\chi} \sin \phi + ml \ddot{\phi} + mg \cos \phi = 0 \quad (72)$$

$$l \ddot{\phi} - \sin \phi \ddot{\chi} + g \cos \phi = 0 \quad (73)$$

que após linearizar em torno de $(x_0, \theta_0) = (0, +90^\circ)$, fornece (74). As equações (74)-(75) são as seguintes

$$l \ddot{\theta} - \ddot{\chi} - g \theta = 0 \quad (74)$$

$$(M + m) \ddot{\chi} - ml \ddot{\theta} = F \quad (75)$$

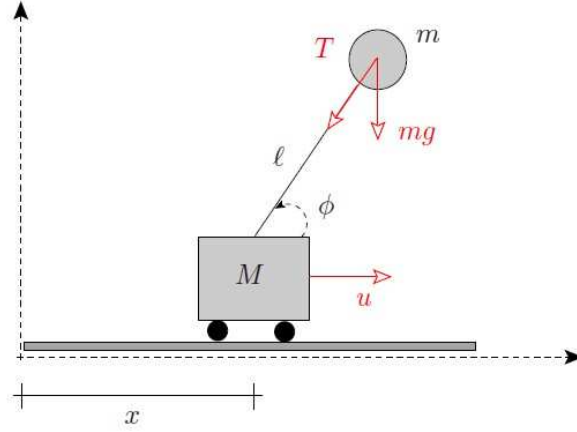


Figura 1: Pêndulo invertido

As equações acima representam o sistema linearizado em torno do ponto de equilíbrio. Faremos agora, a partir destas equações, a representação de estado do sistema considerando:

$$x_1 = \chi \quad (76)$$

$$x_2 = \dot{\chi} \quad (77)$$

$$x_3 = \theta \quad (78)$$

$$x_4 = \dot{\theta} \quad (79)$$

Representando o sistema na forma matricial, temos:

$$\dot{x} = Ax + Bu + Hw \quad (80)$$

$$z = E_\sigma x + F_\sigma u + G_\sigma w \quad (81)$$

com

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{gm}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g(M+m)}{lM} & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{lM} \end{bmatrix} \quad (82)$$

Consideramos que o ruído, representado pela matriz H ocorrerá tanto no deslocamento horizontal do carro quanto no deslocamento angular do pêndulo. Logo, a matriz H é determinada por:

$$H = \begin{bmatrix} 1 & 0 & -\pi/4 & 0 \end{bmatrix}' \quad (83)$$

Além disso, definimos dois critérios conflitantes caracterizados pelos pares (E_1, F_1) e (E_2, F_2) dados a seguir

$$E_1 = \begin{bmatrix} 50 & 0 & 0 & 0 \\ 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, F_1 = \begin{bmatrix} 0 \\ 0 \\ 0.5 \end{bmatrix}, E_2 = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, F_2 = \begin{bmatrix} 0 \\ 0 \\ 50 \end{bmatrix} \quad (84)$$

e $G_1 = G_2 = 0$. Podemos perceber que o par (E_1, F_1) permite altas amplitudes para o esforço de controle e baixas amplitudes para os estados, enquanto que para o par (E_2, F_2) a situação é exatamente a oposta, ou seja, penaliza o esforço de controle e permite altas amplitudes para os estados.

6 Aplicação, Simulação e Análise dos Resultados

6.1 Realimentação de Estado

Nesta seção vamos utilizar a base teórica desenvolvida nas seções anteriores para realizar o controle via realimentação de estado do pêndulo invertido. Primeiramente, vamos projetar a lei de controle $u = K_i x, i \in \{1, 2\}$ considerando cada critério isolado e, posteriormente, realizar o controle simultâneo da regra de comutação $\sigma(x)$ e de um conjunto de ganhos de realimentação de estados $\{K_1, K_2\}$ de forma a obter um desempenho melhor do que o obtido utilizando cada critério isoladamente, veja [6] e [9]. Para realizar as simulações, consideramos $g = 9.8[m/s^2]$, $M = 10[kg]$, $m = 2[kg]$, $l = 1[m]$ e uma entrada externa de $w(t) = \sin(5t)$ para o intervalo $0 \leq t \leq 2\pi$ e zero nos demais instantes. Toda programação foi realizada no Matlab e está disponível no apêndice.

Caso 01: Neste primeiro caso, consideramos apenas o primeiro critério (E_1, F_1) . Resolvemos o problema (37) junto com (36) e obtivemos $\|H\|_\infty^2 = 5481.7$. Para fins de implementação numérica, consideramos $\rho = 5500$ associado ao ganho

$$K_1 = 10^5 \times \begin{bmatrix} 0.5644 & 0.4124 & -1.6898 & -0.5408 \end{bmatrix},$$

e implementamos a lei de controle $u = K_1 x$. A Figura 2 apresenta as trajetórias χ e θ em função do tempo, bem como o esforço de controle utilizado.

Caso 02: Neste caso, é considerado apenas o segundo critério (E_2, F_2) . Resolvemos o mesmo problema já citado no Caso 01 e obtivemos $\|H\|_\infty^2 = 1.86 \times 10^6$. Para fins de implementação numérica,

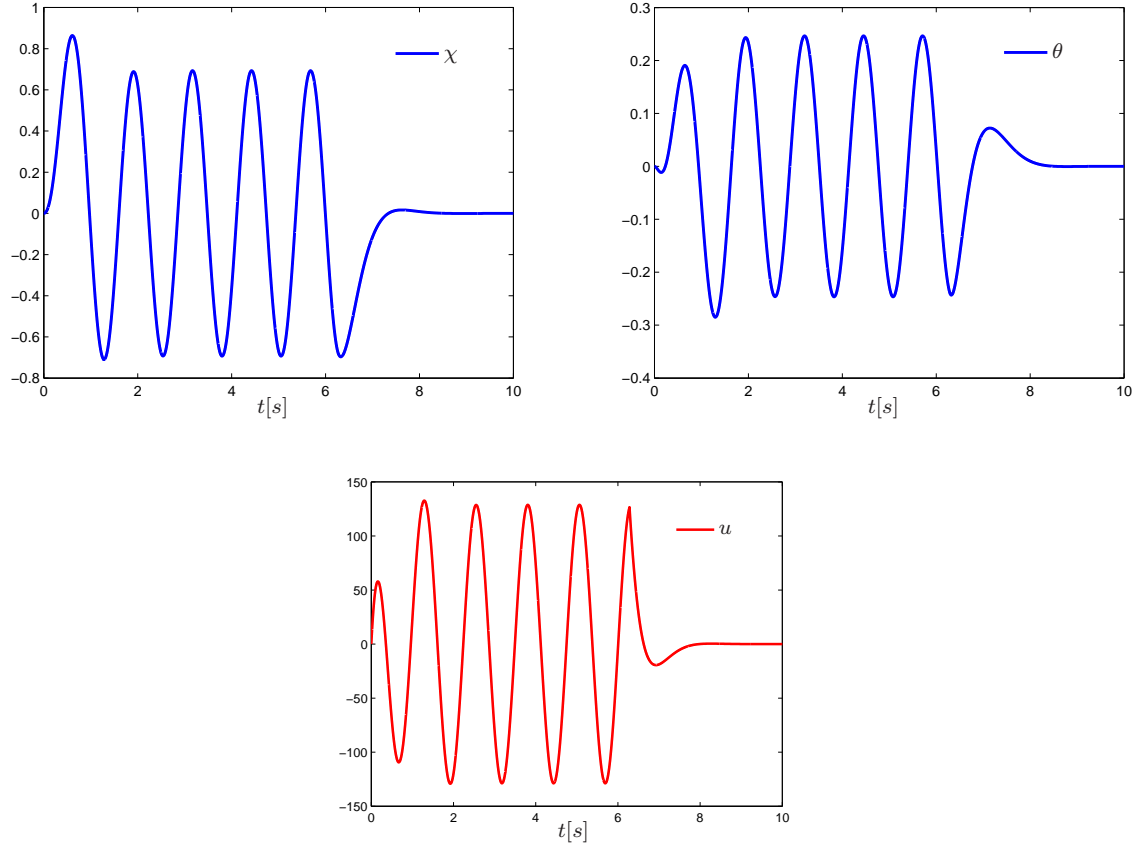


Figura 2: Trajetória dos estados, esforço de controle e função de comutação - Caso 01.

consideramos $\rho = 1.9 \times 10^6$ e o seguinte ganho de realimentação de estado correspondente

$$K_2 = 10^4 \times \begin{bmatrix} 0.0001 & 0.0045 & -1.0805 & -0.3158 \end{bmatrix},$$

A Figura 3 apresenta as trajetórias χ e θ em função do tempo, bem como o esforço de controle utilizado.

Como já esperado $u = K_2 x$ obtido no Caso 2 exige um esforço de controle muito menor do que no primeiro caso. Em contrapartida fornece uma resposta muito mais lenta com um tempo de estabilização de aproximadamente 200 segundos.

Caso 03: Consideramos agora o caso com comutação. Resolvemos as condições apresentadas no Teorema 3 para uma matriz Π dada por:

$$\Pi = \begin{bmatrix} -p & q \\ p & -q \end{bmatrix} \quad (85)$$

com (p, q) variando em uma caixa definida por $[0, 2000] \times [0, 4000]$ com passo 100.

A Figura 4 apresenta o custo garantido H_∞ em função de (p, q) , onde foram considerados apenas

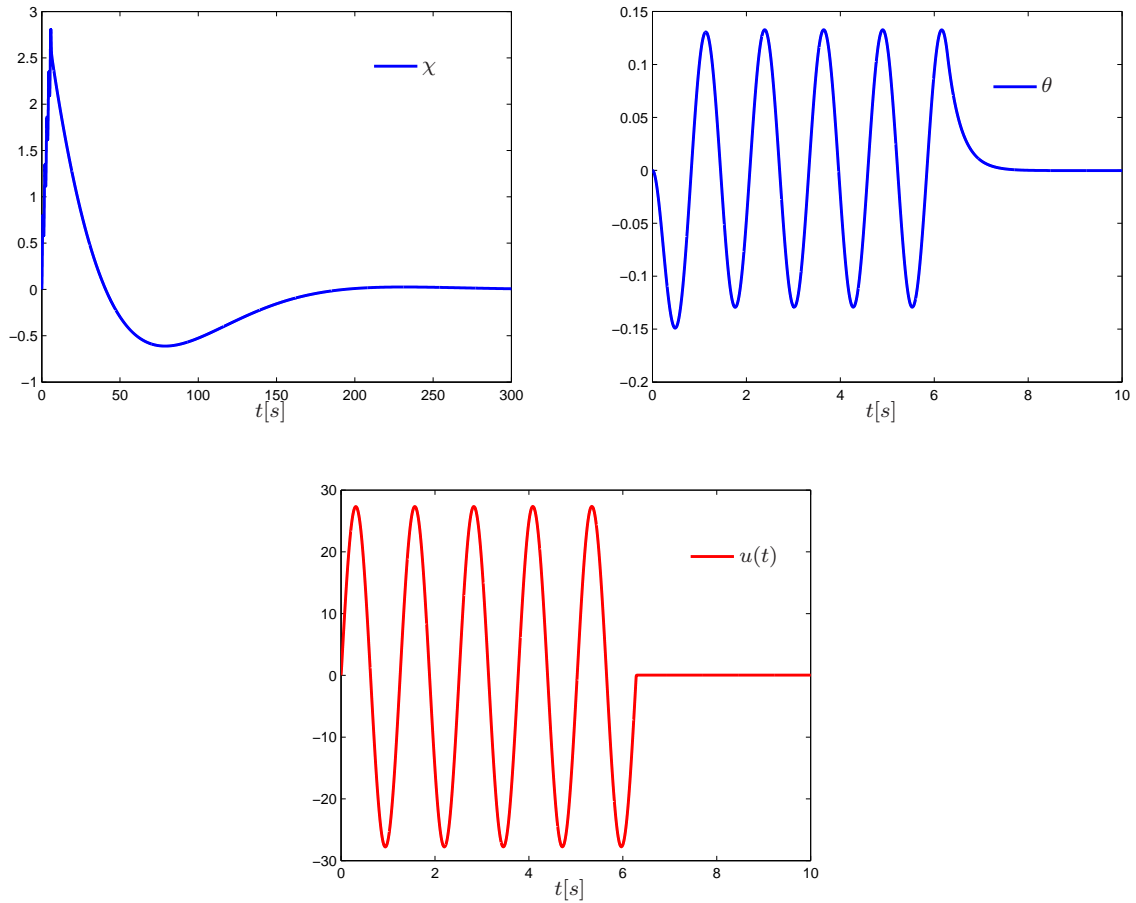


Figura 3: Trajetória dos estados, esforço de controle e função de comutação - Caso 02.

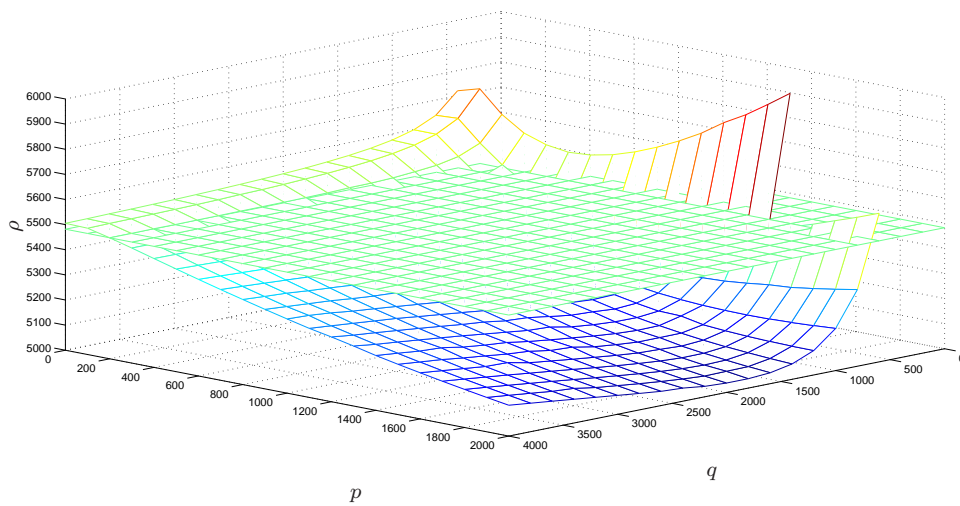


Figura 4: Custo garantido \mathcal{H}_∞ em função dos elementos (p, q) .

valores de custo menores do que 6000. Nesta figura, o plano apresenta o valor da menor norma H_∞ ao quadrado de ambos os subsistemas, ou seja, $\rho_1 = 5480$. Note que a escolha adequada de (p, q) nos permite

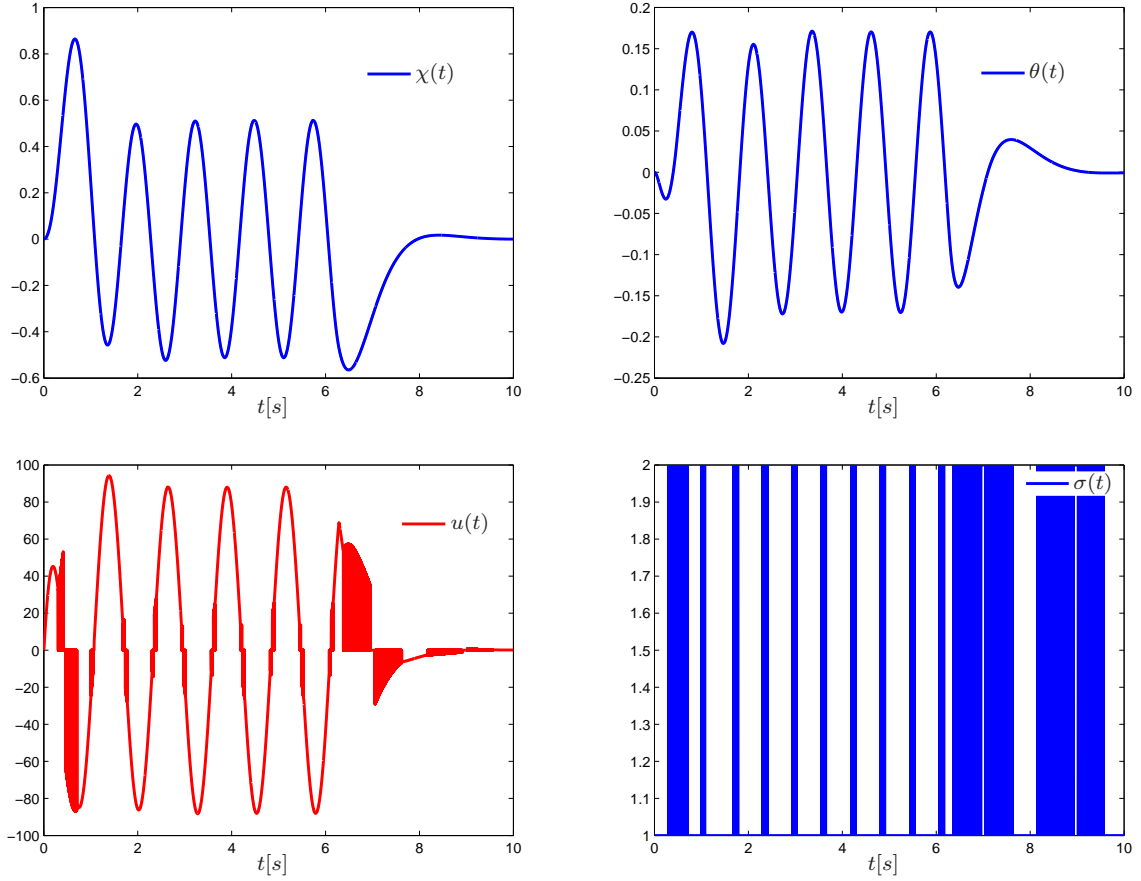


Figura 5: Trajetória dos estados, esforço de controle e função de comutação - Caso 03.

obter um desempenho melhor do que aquele obtido pelos controles clássicos $u = K_1 x$ e $u = K_2 x$. Em particular, o menor valor de $\rho^* = 5037$ foi obtido para $(p, q) = (2000, 1800)$, o que indica uma melhoria de desempenho de aproximadamente 8%. A solução das condições do Teorema 3 para esta matriz de Metzler particular fornece os seguintes ganhos matriciais

$$K_1 = 10^4 \times \begin{bmatrix} 0.5296 & 0.4907 & -2.4234 & -0.7656 \end{bmatrix},$$

$$K_2 = \begin{bmatrix} 0.5704 & 0.5285 & -2.6102 & -0.8246 \end{bmatrix}$$

e as matrizes

$$S_1 = \begin{bmatrix} 0.00235 & -0.00367 & 0.00069 & -0.00292 \\ -0.00367 & 0.01190 & -0.00362 & 0.01658 \\ 0.00069 & -0.00362 & 0.00573 & -0.01999 \\ -0.00292 & 0.01658 & -0.01999 & 0.07194 \end{bmatrix} \quad (86)$$

$$S_2 = \begin{bmatrix} 0.00236 & -0.00368 & 0.00070 & -0.00294 \\ -0.00368 & 0.01191 & -0.00363 & 0.01663 \\ 0.00070 & -0.00363 & 0.00576 & -0.02007 \\ -0.00294 & 0.01663 & -0.02007 & 0.07220 \end{bmatrix} \quad (87)$$

que são importantes para a implementação da função de comutação $\sigma(x)$. A Figura 5 apresenta as trajetórias de $\chi(t)$ e de $\theta(t)$ bem como o esforço de controle $u(t)$ e a função de comutação $\sigma(t)$ projetada. Por simulação numérica e para a entrada externa adotada, obtivemos $2799 < J_\infty(\sigma) < 5037$ onde

$$J_\infty(\sigma) = \sup_{w \in \mathcal{L}_2} \frac{\|z\|_2^2}{\|w\|_2^2}$$

Infelizmente, encontrar a entrada externa de pior caso que permite obter $J_\infty(\sigma)$ não é uma tarefa fácil, uma vez que devido à regra de comutação $\sigma(x)$, o problema associado é altamente não linear e variante no tempo. Os resultados que acabamos de apresentar comprovam a validade e a eficiência da função de comutação projetada para o controle do pêndulo invertido.

6.2 Realimentação de Saída

Nesta seção vamos utilizar a base teórica construída para realizar o controle via realimentação de saída do pêndulo invertido. Neste caso, o sistema a ser considerado é (80)-(81) com a saída medida dada por

$$y = Cx + Dw \quad (88)$$

sendo

$$C = \begin{bmatrix} 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}, \quad D = \begin{bmatrix} 0.001 \\ 0 \end{bmatrix} \quad (89)$$

Resolvemos as condições do Teorema 5 para a matriz Π com a estrutura (85), com (p, q) variando na caixa $[0, 2000] \times [0, 4000]$. Como observado na Figura 6 o melhor desempenho encontrado foi de 9.360×10^6 . Para a saída medida em consideração, calculamos a norma \mathcal{H}_∞ ao quadrado para cada um dos critérios isolados e obtivemos para o caso (E_1, F_1) o valor de 5615.6 e para o caso (E_2, F_2) o valor de 9.360×10^6 . Infelizmente, como podemos notar, para este caso em particular, a função de comutação não é consistente uma vez que o custo obtido para o caso com comutação é igual ao pior desempenho de ambos os critérios isolados representado pelo plano da Figura 6. Na verdade, as condições que permitem obter uma função de comutação estritamente consistente para o caso de realimentação de saída continua um problema em aberto que merece ser estudado.

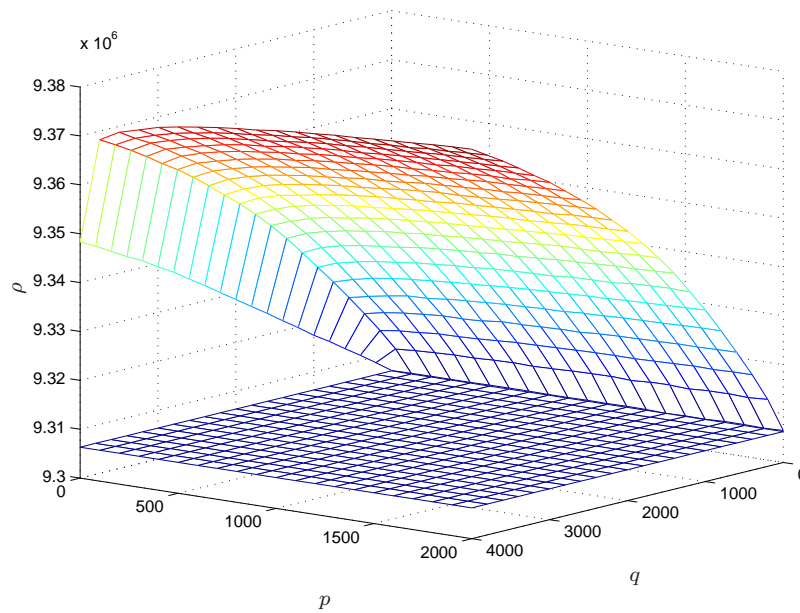


Figura 6: Custo garantido \mathcal{H}_∞ em função dos elementos (p, q) .

7 Conclusão

Neste trabalho, projetamos uma lei de controle com comutação para o projeto de controle \mathcal{H}_∞ de um pêndulo invertido sobre um carro. Definimos dois critérios diferentes e conflitantes para o pêndulo e comparamos o desempenho obtido com aquele fornecido considerando cada critério isoladamente. Como podemos concluir pela análise dos resultados obtidos, para o caso em que o estado está disponível para a realimentação, a função de comutação projetada é estritamente consistente e fornece resultados melhores do que aquele obtido considerando o projeto isolado para cada critério. Desta forma, conforme esperado, pudemos constatar que o controle \mathcal{H}_∞ com comutação é uma excelente alternativa de projeto. Infelizmente, a mesma conclusão não foi obtida para o caso de realimentação de saída. Na verdade, as condições para o projeto de uma função de comutação estritamente consistente para este caso em particular é um problema em aberto que merece ser estudado na literatura.

Referências

- [1] De Oliveira, M.C.: “Controle de sistemas lineares baseado nas desigualdades matriciais lineares”, tese de doutorado, 1999.
- [2] Deaecto, G.S.: “Síntese de controle para sistemas dinâmicos com comutação”, dissertação de mestrado, FEEC - Unicamp, 2007.
- [3] Deaecto, G.S.: “Projeto de Controladores Dinâmicos com Comutação - Aplicação em Sistemas Mecânicos e Conversores de Potência CC-CC”, tese de doutorado, FEEC - Unicamp, 2010.
- [4] Deaecto, G.S., Geromel, J.C.: “ \mathcal{H}_∞ control for continuous time switched linear systems”, Journal of Dynamic Systems, Measurement, and Control, vol 132, number 041013, 2010.
- [5] Deaecto, G.S., Geromel, J.C., Daafouz, J.: "Dynamic output feedback H_∞ control of switched linear systems", Automatica, vol. 47, pp. 1713-1720, 2011.
- [6] Geromel, J.C., Deaecto, G.S., Colaneri, P.: “Multi-objective \mathcal{H}_2 control via switched linear systems”, Proc. of the 3rd IFAC Symp. on Syst., Structure and Control, 2007.
- [7] Geromel, J.C., Deaecto, G.S.: “Switched state feedback control for continuous-time uncertain systems”, Automatica, vol. 45, pp. 593-597, 2009.
- [8] Geromel, J.C., Korogui, R. H.: “Controle Linear de Sistemas Dinâmicos : Teoria, Ensaio Práticos e Exercícios”, Editora Edgard Blucher Ltda, 2011.
- [9] Geromel, J.C., Deaecto, G.S., Daafouz, J.: “Suboptimal switching control consistency analysis for switched linear systems”, IEEE Transaction on Automatic Control, vol. 58, pp. 1857-1861, 2013.
- [10] Scarpa, V.C.: “Controle com comutação \mathcal{H}_2 de um pêndulo invertido”, Trabalho de Graduação, FEM/UNICAMP, http://www.fem.unicamp.br/~grace/ViniciusScarpa_RelatorioFinal.pdf, 2014.
- [11] Slotine, J.J.E., Li, W.: “Applied Nonlinear Control”, Prentice Hall, 1991.

8 Anexo: Programa do Matlab

Programa principal

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Determina Norma e Desenha o Gráfico 3D
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Dados do Sistema
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

global A;
global B;
global F;
global E;
global H;
global G;
global Ksub;

g = 9.8;
Mcarro = 10;
mpendulo = 2;
ell = 1; %comprimento do cabo

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Matrizes do sistema do pêndulo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [0 1 0 0; 0 0 g*mpendulo/Mcarro 0; 0 0 0 1; 0 0 g*(Mcarro+mpendulo)/(ell*Mcarro) 0];
B = [0 1/Mcarro 0 1/(Mcarro*ell)]';
E(:, :, 1) = [50 0 0 0; 0 0 50 0; 0 0 0 0];
E(:, :, 2) = [0.5 0 0 0; 0 0 0.5 0; 0 0 0 0];
F(:, :, 1) = [0 0 0.5]';
F(:, :, 2) = [0 0 50]';
H = [1; 0; -pi/4; 0];
G = [0; 0; 0];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% O Programa deve definir a matriz Pi em função de ps e qs determinados abaixo e
% determinar a norma Hoo e os ganhos utilizando a função controle_comutacao_estado
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Valores de ps e qs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
ps = [0:100:2000];
qs = [0:200:4000];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definicão da matriz Pi e o cálculo do custo para o sistema com comutação através
% da função controle_comutacao_estado
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=1:length(ps)
    p = ps(i);
    for j=1:length(qs)
```

```

        q = qs(j);
        Pi = [-p q;p -q];
        [rho(i,j),Ys,Ss,K] = controle_comutacao_estado(A,B,H,E,F,G,Pi);
    end
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Cálculo dos custos para cada subsistema isolado e o desenho do gráfico 3D para
% os subsistemas isolados e para o sistema com comutação.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

N = size(E,3);
for i=1:N
    [rhosubs(i) Ksub(:, :, i)] = norma_subsistema_isolado(A,B,H,E(:, :, i),F(:, :, i),G)
end

mesh(qs,ps,min(rhosubs)*ones(size(rho)))
hold on
aux1 = rho;
aux1 (find(aux1>=6000))=NaN;
mesh(qs, ps, aux1)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% O programa deve determinar qual é o valor mínimo do gráfico e os valores ps e qs
% correspondentes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[val,ip] = min(rho)
[cost,iq] = min(val)

po = ps(ip(iq))
qo = qs(iq)

```

Programa calcula a norma de cada critério isolado: norma_ subsistema_ isolado.m

```
function [normaHoo,K] = norma_subsistema_isolado(A,B,H,E,F,G)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% O programa deve calcular a norma de cada subsistema isoladamente
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definição do tamanho das matrizes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n = size(A,1);
m = size(B,2);
p = size(E,1);
setlmis([])

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definição das variáveis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

S = lmivar(1,[n 1]);
gama2 = lmivar(1,[1 0]);
Y = lmivar(2,[m,n]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Descrição das LMIs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ct = 1;
lmiterm([ct,1,1,S],A,1,'s');
lmiterm([ct,1,1,Y],B,1,'s');
lmiterm([ct,2,1,0],H');
lmiterm([ct,2,2,gama2],-1,1);
lmiterm([ct,3,1,S],E,1);
lmiterm([ct,3,1,Y],F,1);
lmiterm([ct,3,2,0],G);
lmiterm([ct,3,3,0],-1);
%S maior que zero
ct = ct+1;
lmiterm([-ct,1,1,S],1,1);
lmisys = getlmis;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Declaração da função objetivo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

options = [1e-7,2000,0,200,0];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calcula as matrizes 'S' e retorna o valor da norma Hoo e dos ganhos
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
status = 0;
np = decnbr(lmisys);
c = zeros(np,1);
c(decinfo(lmisys,gama2)) = 1;
[copt,xopt] = mincx(lmisys,c,options);
if (isempty(copt))
    normaHoo = NaN;
    K = NaN;
    status = -1;
end
```

```
    return
end
Sa = dec2mat(lmisys,xopt,S);
Ya = dec2mat(lmisys,xopt,Y);
normaHoo = dec2mat(lmisys,xopt,gama2);
K = Ya*inv(Sa);
```

Programa para simulação numérica em um determinado ponto

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Dados do problema para simulação e variáveis globais
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global A;
global B;
global F;
global E;
global H;
global G;
global S;
global K;

g = 9.8;
Mcarro = 10;
mpendulo = 2;
ell = 1; %comprimento do cabo

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Matrizes do sistema do pêndulo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
A = [0 1 0 0; 0 0 g*mpendulo/Mcarro 0; 0 0 0 1; 0 0 g*(Mcarro+mpendulo)/(ell*Mcarro) 0];
B = [0 1/Mcarro 0 1/(Mcarro*ell)]';
E(:, :, 1) = [50 0 0 0; 0 0 50 0; 0 0 0 0];
E(:, :, 2) = [0.5 0 0 0; 0 0 0.5 0; 0 0 0 0];
F(:, :, 1) = [0 0 0.5]';
F(:, :, 2) = [0 0 50]';
H = [1; 0; -pi/4; 0];
G = [0; 0; 0];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Valores mínimos determinado pelo programa principal: p0 = 2000 e q0 = 1800
% A simulação a seguir leva em conta os valores encontrados para o mínimo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definição da matriz Pi
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

p = 2000;
q = 1800;
Pi = [-p q; p -q];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Cálculo do custo para este ponto e simulação numérica
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[rho, Y, S, K] = controle_comutacao_estado(A, B, H, E, F, G, Pi);
tin = 0;
tfin = 15;
sim('simulac.mdl', [tin tfin])
```

Programa que contém as regras para cada opção de simulação: simulacao.m

```
function xdata = simulacao(u)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definição das variáveis globais
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

global A;
global B;
global F;
global E;
global H;
global G;
global S;
global K;

w = u(1);
x = u(2:end);

opt = 3;
%1 = sistema 1
%2 = sistema 2
%3 = comutacao

switch opt

    case 1

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Simulação para o subsistema 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        uc = K(:, :, 1)*x;
        xdot = A*x + B*uc + H*w;
        z = E(:, :, 1)*x + F(:, :, 1)*uc + G*w;
        sigma = 1;
        xdata = [sigma; z'*z; z; w'*w; uc; xdot];

    case 2

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Simulação para o subsistema 2
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        uc = K(:, :, 2)*x;
        xdot = A*x + B*uc + H*w;
        z = E(:, :, 2)*x + F(:, :, 2)*uc + G*w;
        sigma = 2;
        xdata = [sigma; z'*z; z; w'*w; uc; xdot];

    case 3

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Simulação para a comutação
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```

if((x'*inv(S(:,:,1))*x) <= (x'*inv(S(:,:,2))*x))

    uc = K(:,:,1)*x;
    xdot = A*x + B*uc + H*w;
    z = E(:,:,1)*x + F(:,:,1)*uc + G*w;
    sigma = 1;

else

    uc = K(:,:,2)*x;
    xdot = A*x + B*uc + H*w;
    z = E(:,:,2)*x + F(:,:,2)*uc + G*w;
    sigma = 2;

end

xdata = [sigma;z'*z;z*w'*w;uc;xdot];

```

Programa que calcula o controle com comutação do sistema: controle_comutacao_estado.m

```
function [NHinfinito, Ys, Ss, K] = controle_comutacao_estado(A,B,H,E,F,G,Pi)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% O Programa deve calcular o custo para um sistema com comutação
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definição do tamanho das matrizes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n = size(A,1);
m = size(B,2);
p = size(E,1);
N = size(E,3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definição das variáveis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

setlmis([])
for i=1:N
    S(i)=lmivar(1,[n 1]);
    for j=1:N
        if i ~= j
            T(i,j) = lmivar(1,[n,1]);
        end
    end
    Y(i)=lmivar(2,[m,n]);
end
rho = lmivar(1,[1 0]);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Descrição das LMIs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ct = 0;

for (i=1:N)
    ct = ct+1;
    lmiterm([ct,1,1,S(i)],A,1,'s');
    lmiterm([ct,1,1,Y(i)],B,1,'s');
    for (j=1:N)
        if (i ~= j)
            lmiterm([ct,1,1,T(i,j)],Pi(j,i),1);
        end
    end
    lmiterm([ct,2,1,0],H');
    lmiterm([ct,2,2,rho],-1,1);
    lmiterm([ct,3,1,S(i)],E(:, :, i),1);
    lmiterm([ct,3,1,Y(i)],F(:, :, i),1);
    lmiterm([ct,3,2,0],G);
    lmiterm([ct,3,3,0],-1);
end
for i=1:N
    ct = ct+1;
    for j=1:N
        if (i ~= j)
```



```

        lmiterm([-ct,1,1,T(i,j)],1,1);
        lmiterm([-ct,1,1,S(i)],1,1);
        lmiterm([-ct,2,1,S(i)],1,1);
        lmiterm([-ct,2,2,S(j)],1,1);
    end
end
end
lmisys = getlmis;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Declaração da função objetivo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

options = [1e-4,2000,0,200,0];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calcula as matrizes 'S' e retorna o valor da norma Hoo e dos ganhos
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

status = 0;
np = decnbr(lmisys);
c = zeros(np,1);
c(decinfo(lmisys,rho)) = 1;

[copt,xopt] = mincx(lmisys,c,options);
if (isempty(copt))
    NHinfnito = NaN;
    K = NaN;
    status = -1;
    return
end
NHinfnito = dec2mat(lmisys,xopt,rho);
for i = 1:N
    Ys(:, :, i) = dec2mat(lmisys,xopt,Y(i));
    Ss(:, :, i) = dec2mat(lmisys,xopt,S(i));
    K(:, :, i) = Ys(:, :, i)*inv(Ss(:, :, i));
end
end

```

Programa que calcula o controle realimentação de saída: controle_comutacao_saida.m.

```
function [NHinfinito] = controle_comutacao_saida(A,B,H,E,F,G,C,D,Pi)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% O programa deve calcular o custo para um sistema com comutação para o caso da
% Realimentação de Saída
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definição do tamanho das matrizes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

n = size(A,1);
m = size(B,2);
p = size(E,1);
N = size(E,3);
r = size(C,1);
N = size(E,3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definição das variáveis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

setlmis([])
for i=1:N
    X(i)=lmivar(1,[n 1]);

    for j=1:N
        if i ~= j
            T(i,j) = lmivar(1,[n,1]);
        end
    end
    W(i)= lmivar(2,[m,n]);
    M(i)= lmivar(2,[n,n]);
    L(i)= lmivar(2,[n,r]);
end

Y = lmivar(1,[n 1]);
rho = lmivar(1,[1 0]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Descrição das LMIs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ct = 0;
%para a primeira desigualdade
for (i=1:N)
    ct = ct+1;
    lmiterm([ct,1,1,Y],A,1,'s');
    lmiterm([ct,1,1,L(i)],1,C(:, :, i), 's');

    lmiterm([ct,2,1,0],A);
    lmiterm([ct,2,1,-M(i)],1,1);

    lmiterm([ct,2,2,X(i)],A,1,'s');
    lmiterm([ct,2,2,W(i)],B,1,'s');
```

```

for (j=1:N)
    if (i ~= j)
        lmiterm([ct,1,1,T(i,j)],Pi(j,i),1);
    end
end

lmiterm([ct,3,1,Y],H',1);
lmiterm([ct,3,1,-L(i)],D',1);
lmiterm([ct,3,2,0],H');
lmiterm([ct,3,3,rho],-1,1);

lmiterm([ct,4,1,0],E(:, :, i));
lmiterm([ct,4,2,X(i)],E(:, :, i),1);
lmiterm([ct,4,2,W(i)],F(:, :, i),1);
lmiterm([ct,4,3,0],G);
lmiterm([ct,4,4,0],-1);

%para a segunda desigualdade
for (j=1:N)
    if (i ~= j)
        ct = ct+1;
        lmiterm([-ct,1,1,T(i,j)],1,1);
        lmiterm([-ct,1,1,X(i)],1,1);
        lmiterm([-ct,2,1,X(i)],1,1);
        lmiterm([-ct,2,2,X(j)],1,1);

        lmiterm([-ct,3,1,0],1);
        lmiterm([-ct,3,2,0],1);
        lmiterm([-ct,3,3,Y],1,1);
    end
end
end

lmisys = getlmis;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Declaração da função objetivo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

options = [1e-4,2000,0,200,0];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calcula o custo do sistema com comutação
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

status = 0;
np = decnbr(lmisys);
c = zeros(np,1);
c(decinfo(lmisys,rho)) = 1;

[copt,xopt] = mincx(lmisys,c,options);

if (isempty(copt))
    NHinfinito = NaN;
    status = -1;
    return
end
NHinfinito = dec2mat(lmisys,xopt,rho);

```

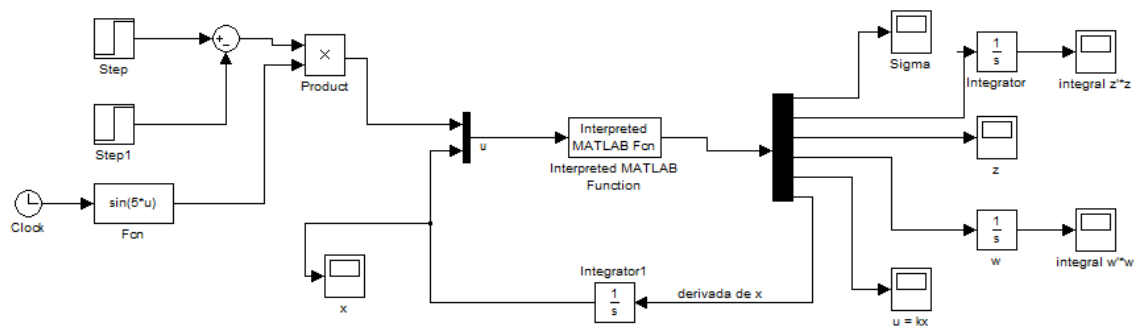


Figura 7: Simulink da simulação