

**UFMS - UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL  
FACOM - FACULDADE DE COMPUTAÇÃO**

**EFEITO DE SENsoRES DINÂMICOS DE PRIMEIRA  
ORDEM NO CONTROLE  $H_\infty$  DE SISTEMAS  
CONTÍNUOS**

**JONATAS SANTOS GALVÃO**

**Campo Grande - MS, dezembro de 2025**

Trabalho de Conclusão de Curso apresentada ao programa de **Graduação em Engenharia de Computação** da **Faculdade de Computação - FACOM** da Universidade Federal de Mato Grosso do Sul - UFMS, como requisito à obtenção do título de **Engenheiro de Computação** a Jonatas Santos Galvão.

**Data de aprovação:**

\_\_\_\_ / \_\_\_\_ / \_\_\_\_

**Banca Examinadora:**

---

FACOM - UFMS

---

FACOM - UFMS

---

FACOM - UFMS



*Agradeço primeiramente a Deus, pela força e pela serenidade nos momentos em que as dificuldades pareciam maiores do que as possibilidades.*

*Aos meus pais, por todo o apoio, carinho e incentivo ao longo da minha vida acadêmica, vocês me ensinaram o valor do esforço, da honestidade e do estudo. Cada conquista minha carrega um pouco do exemplo e da dedicação que sempre demonstraram. Vocês me mostraram que o verdadeiro sucesso nasce da disciplina e da persistência.*

*À minha esposa, por sua paciência infinita, compreensão e amor inabalável. Foram incontáveis as noites em claro e os fins de semana de renúncia que ela enfrentou ao meu lado. Nada disso seria possível sem seu apoio constante. Você foi uma companheira incansável em todas as horas, pensei em desistir várias vezes mas você foi o alicerce que me sustentou nesta jornada, este título é tanto meu quanto seu.*

*Aos meus amigos, que tornaram essa caminhada mais leve, compartilhando aprendizados, risadas e desafios — cada um deixou uma parte importante desta história. Vocês foram apoio, leveza e motivação nos momentos em que o caminho parecia longo demais.*

## AGRADECIMENTOS

Agradeço aos meus professores pela dedicação em compartilhar conhecimento e pela inspiração que plantaram em cada aula. Aos meus orientadores de Iniciação Científica *Ricardo Ribeiro dos Santos, Renan Albuquerque Marks* expresso minha sincera gratidão pela orientação, paciência e constante incentivo à pesquisa. As experiências e aprendizados adquiridos sob sua supervisão foram fundamentais para minha formação acadêmica e para o desenvolvimento do senso crítico e científico que levo comigo. Acredito que sob vossas orientações desenvolvi habilidades essenciais que me fez estar entre os melhores profissionalmente. Ao meu orientador, *Victor Leonando Yoshimura*, expresso meu mais profundo agradecimento pela dedicação, paciência e comprometimento ao longo de todo o desenvolvimento deste trabalho. Sua orientação foi muito além do papel técnico — foi um verdadeiro guia intelectual e pessoal, sempre disposto a compartilhar conhecimento, corrigir rotas e incentivar a busca por soluções sistemáticas fundamentadas. O senhor me ensinou o valor do rigor matemático e a necessidade do mesmo para construir quaisquer tipo de soluções viáveis. Agradeço por cada conselho, pela confiança depositada em mim e pela forma como conduziu este processo com sabedoria e humanidade. Sua postura ética, rigor científico e entusiasmo pela área foram uma inspiração constante, não apenas para este projeto, mas para minha formação como engenheiro e pesquisador. Este trabalho carrega, em cada linha, a influência positiva de sua orientação e o reflexo do aprendizado construído sob sua mentoria. Por fim, agradeço a todos que, de alguma forma, contribuíram para que este momento se tornasse realidade. Cada gesto de incentivo foi combustível para chegar até aqui.

*“Mas o que para mim era ganho reputei-o perda por Cristo. E, na verdade, tenho também por perda todas as coisas, pela excelência do conhecimento de Cristo Jesus, meu Senhor; pelo qual sofri a perda de todas estas coisas, e as considero como esterco, para que possa ganhar a Cristo. E seja achado nele, não tendo a minha justiça que vem da lei, mas a que vem pela fé em Cristo, a saber, a justiça que vem de Deus pela fé;”*

**Filipenses 3:8**, Bíblia Sagrada

## RESUMO

Este trabalho aborda o projeto de controladores robustos do tipo  $\mathcal{H}_\infty$  para uma planta linear de quarta ordem sujeita a perturbações, com ênfase no impacto da dinâmica de sensores modelados por filtros de primeira ordem. A formulação teórica parte do Lema do Ganho Limitado e da representação em LMIs obtida por mudança de variáveis, o que permite sintetizar ganhos de realimentação garantindo estabilidade assintótica em espaço de estados. A contribuição central consiste em confrontar ganhos sintetizados para o modelo nominal (assumindo realimentação direta do estado) com a implementação prática em que o controlador atua sobre medidas filtradas; essa discrepância foi investigada analiticamente através da construção do sistema aumentado que inclui a dinâmica dos sensores, da análise do polinômio característico aumentado e da verificação das propriedades de estabilizabilidade e das LMIs no espaço aumentado. Simulações numéricas em Python (síntese e checagem algébrica) e experiências em Xcos/Scilab (implementação e resposta em frequência) evidenciaram que ganhos obtidos sem considerar a dinâmica sensorial podem provocar instabilidade em implementação prática. A conclusão prática é direta: quando o controle é aplicado sobre sinais filtrados, a síntese deve contemplar o modelo aumentado ou então ser acompanhada por um observador para reconstrução do estado; cuidados de escalonamento e regularização numérica são necessários para garantir reproduzibilidade.

**Palavras-chave:** controle robusto,  $\mathcal{H}_\infty$ , LMIs, sensores dinâmicos, sistema aumentado, estabilidade.

## ABSTRACT

This work investigates  $\mathcal{H}_\infty$  controller design for a fourth-order linear plant subject to disturbances, emphasizing the influence of first-order sensor dynamics. The theoretical framework relies on the Bounded Real Lemma and the LMI formulation obtained via the change of variables, enabling convex synthesis of state-feedback gains that guarantee asymptotic stability. The main contribution contrasts gains synthesized under the ideal assumption of direct state feedback with their practical implementation when control is applied to filtered measurements. This discrepancy is analyzed by forming the augmented system that explicitly includes sensor dynamics, deriving and studying the augmented characteristic polynomial, and checking stabilizability and LMI feasibility on the augmented model. Numerical synthesis and algebraic checks performed in Python, together with closed-loop simulations in Xcos/Scilab, show that controllers designed without accounting for sensor dynamics can destabilize the implemented system. The practical outcome is clear: when control acts on filtered outputs, controller synthesis must include the sensor model or be complemented by a state observer; proper scaling and numerical regularization are essential for reliable results.

**Keywords:** robust control,  $\mathcal{H}_\infty$ , LMIs, dynamic sensors, augmented system, stability.

## LISTA DE FIGURAS

Figura 2.1 – Comparação entre os dois integradores . . . . .	35
Figura 2.2 – Comparação entre os Integradores 3 e 4 . . . . .	36
Figura 2.3 – Comparação entre os Integradores 1 e 2 com Sensor Imperfeito . . . . .	36
Figura 2.4 – Comparação entre os Integradores 3 e 4 com Sensor Imperfeito . . . . .	37
Figura 3.1 – Resposta em frequência para $\omega = 1$ . . . . .	40
Figura 3.2 – Resposta em frequência para $\omega = 10$ . . . . .	41
Figura 3.3 – Resposta em frequência para $\omega = 100$ . . . . .	41
Figura 3.4 – Resposta em frequência para $\omega = 1000$ . . . . .	42
Figura 3.5 – Resposta em frequência para $\omega = 10000$ . . . . .	42
Figura 3.6 – Resposta em frequência para $\omega = 1$ (sensor imperfeito). . . . .	43
Figura 3.7 – Resposta em frequência para $\omega = 10$ (sensor imperfeito). . . . .	44
Figura 3.8 – Resposta em frequência para $\omega = 100$ (sensor imperfeito). . . . .	44
Figura 3.9 – Resposta em frequência para $\omega = 1000$ (sensor imperfeito). . . . .	45
Figura 3.10–Resposta em frequência para $\omega = 10000$ (sensor imperfeito). . . .	45
Figura 3.11–Resposta em frequência para $\omega = 100000$ (sensor imperfeito). . . .	46
Figura 3.12–Resposta em frequência para $\omega = 1$ (sensor imperfeito) - Estado Nulo . . . . .	47
Figura 3.13–Resposta em frequência para $\omega = 10$ (sensor imperfeito) - Estado Nulo . . . . .	47
Figura 3.14–Resposta em frequência para $\omega = 100$ (sensor imperfeito) - Estado Nulo . . . . .	48
Figura 3.15–Resposta em frequência para $\omega = 1000$ (sensor imperfeito) - Estado Nulo . . . . .	48
Figura 3.16–Resposta em frequência para $\omega = 10000$ (sensor imperfeito) - Estado Nulo . . . . .	49

## LISTA DE ABREVIATURAS E SIGLAS

BRL	Bounded Real Lemma (Lema do Ganho Limitado).
CVXPY	Pacote/biblioteca Python utilizado para resolver as LMIs (utilizado na implementação numérica).
EDO	Equação Diferencial Ordinária (presente no contexto de modelagem da planta).
$H\infty$	Controle $H\infty$ (ou norma $H\infty$ ), uma formulação consolidada para tratamento de robustez, que minimiza a norma de operador $H\infty$ da função de transferência do sinal de perturbação para o vetor de desempenho.
LMIs	Desigualdades Matriciais Lineares (português) / Linear Matrix Inequalities (inglês); são problemas convexos que permitem o tratamento numérico eficiente e a formulação da condição de estabilidade.
MOSEK	Solver (solucionador) utilizado pela biblioteca CVXPY para a solução dos problemas de LMIs.
PBH	Critério PBH — Popov–Belevitch–Hautus (utilizado na demonstração de controlabilidade do sistema aumentado).
SDP	Problemas de Programação Semidefinida (Semi-Definite Programming), referindo-se aos problemas de LMIs que são convexos.
SISO	Sistema de uma entrada e uma saída (Single Input, Single Output).
SLIT	Sistema Linear Invariante no Tempo (sistema linear contínuo e invariante no tempo).
TCC	Trabalho de Conclusão do Curso.
Xcos	Ambiente de simulação (do Scilab), utilizado para simulações em malha fechada e verificação prática.

## LISTA DE SÍMBOLOS

$\Gamma$	Gama.
$\lambda$	Comprimento de onda.
$\in$	Pertence.
$\mathbf{x}(t)$	Vetor de estados ( $\mathbf{n}$ ).
$\mathbf{u}(t)$	Entrada de controle ( $\mathbf{m}$ ).
$\mathbf{w}(t)$	Perturbação / sinal exógeno ( $\mathbf{r}$ ).
$\mathbf{y}(t)$	Saída / medições ( $\mathbf{q}$ ).
$\mathbf{z}(t)$	Sinal de desempenho / estados do sensor ( $\mathbf{p}$ ).
$\dot{\mathbf{x}}(t)$	Derivada temporal.
$\ddot{\mathbf{x}}(t)$	Segunda derivada temporal.
$s$	Variável de Laplace.
$\mathbf{A}, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C}, \mathbf{D}, \dots$	Matrizes do espaço de estados.
$\mathbf{K}$	Ganho de realimentação.
$\mathbf{P}$	Matriz de Lyapunov ( $\mathbf{P} \succ \mathbf{0}$ ).
$\mathbf{Q}$	Variável LMI ( $\mathbf{Q} = \mathbf{P}^{-1}$ ).
$\mathbf{W}$	Variável LMI (ex.: $\mathbf{W} = \mathbf{KQ}$ ).
$\mathbf{I}_n$	Identidade $\mathbf{n} \times \mathbf{n}$ .
$\tau, \tau_i$	Constante(s) de tempo (sensor).
$\gamma$	Cota $H_\infty$ .
$\Theta(\mathbf{P}, \mathbf{K})$	Matriz BRL (não linear).
$\mathbf{A}_{\text{cl}}$	Matriz de malha fechada.
$\mathbf{A}_{\text{aug}}$	Matriz do sistema aumentado.
$\mathbf{B}_{\text{aug}}$	Matriz de entrada aumentada.
$\boldsymbol{\eta}$	Vetor estendido $[\mathbf{x}^\top \mathbf{w}^\top]^\top$ .

<b>T</b>	Matriz de transformação (invertível).
<b>C</b>	Matriz de saída / controlabilidade (conforme contexto).
<b>R<sup>n</sup>, R<sup>n×m</sup></b>	Espaços reais (vetores / matrizes).
$\ T_{w \rightarrow z}\ _{H_\infty}$	Norma H <sub>∞</sub> (w → z).
$\sigma(\cdot)$	Maior valor singular.
<b>M ≻ 0</b>	Matriz definida positiva.
<b>M ≺ 0</b>	Matriz definida negativa.
<b>M<sup>T</sup></b>	Transposta.
<b>jω</b>	Eixo imaginário ( $s = j\omega$ ).

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>16</b>
<b>1.1</b>	<b>DECOMPOSIÇÃO DE KALMAN . . . . .</b>	<b>19</b>
1.1.1	EXEMPLO NUMÉRICO . . . . .	21
<b>1.2</b>	<b>A EQUAÇÃO DE LYAPUNOV . . . . .</b>	<b>21</b>
<b>1.3</b>	<b>DERIVAÇÃO DA LMI VIA TRANSFORMAÇÃO DE VARIÁVEIS E CONGRUÊNCIA . . . . .</b>	<b>24</b>
<b>1.4</b>	<b>ESTABILIDADE . . . . .</b>	<b>27</b>
<b>2</b>	<b>EXPERIMENTAÇÃO . . . . .</b>	<b>31</b>
<b>2.1</b>	<b>DADOS DO SISTEMA . . . . .</b>	<b>33</b>
<b>3</b>	<b>ANÁLISE DE RESULTADOS . . . . .</b>	<b>38</b>
<b>3.1</b>	<b>PROJETO DE CONTROLADOR <math>H_\infty</math> . . . . .</b>	<b>38</b>
3.1.1	IMPLEMENTAÇÃO NO XCOS: SENSOR PERFEITO . . . . .	40
3.1.2	IMPLEMENTAÇÃO NO XCOS: SENSOR DINÂMICO . . . . .	43
3.1.3	PROJETO EM ESTADO AUMENTADO . . . . .	49
<b>4</b>	<b>CONCLUSÃO . . . . .</b>	<b>62</b>
<b>REFERÊNCIAS . . . . .</b>		<b>65</b>
<b>ANEXO A – LMI: CONTROLADOR DE LYAPUNOV . . . . .</b>		<b>67</b>
<b>ANEXO B – LMI: <math>H_\infty</math> / SINTETIZAÇÃO DE GANHO . . . . .</b>		<b>69</b>

**ANEXO C – VERIFICAÇÃO DE CONTROLABILIDADE . . . . 71**

# 1 INTRODUÇÃO

Sistemas de controle aplicados em engenharia frequentemente operam em ambientes sujeitos a incertezas, perturbações externas e variações paramétricas que dificultam a manutenção simultânea de estabilidade e desempenho desejados. Exemplos práticos incluem ruído de sensores, variações de carga, dinâmica não modelada de atuadores e degradação de componentes. Nessas condições, um controlador projetado apenas em um modelo nominal tende a apresentar desempenho degradado — em alguns casos de forma inaceitável — quando implementado no sistema real. Por isso, o projeto orientado à robustez torna-se fundamental: pretende-se obter controladores capazes de garantir estabilidade e níveis aceitáveis de desempenho mesmo na presença de incertezas e sinais exógenos. (DOYLE et al., 1992). Uma das formulações mais consolidadas para tratamento de robustez é o controle  $\mathcal{H}_\infty$  (DOYLE et al., 1989). Neste paradigma representa-se o sistema pelo modelo em espaço de estados (DOYLE et al., 1989),

$$\begin{aligned}\dot{x}(t) &= Ax(t) + B_1 u(t) + B_2 w(t), \\ z(t) &= C_z x(t) + D_{zu} u(t), \\ y(t) &= C_y x(t),\end{aligned}\tag{1}$$

onde  $x(t) \in \mathbb{R}^n$  é o vetor de estados,  $u(t) \in \mathbb{R}^m$  a entrada de controle,  $w(t) \in \mathbb{R}^r$  a perturbação/exógena,  $z(t) \in \mathbb{R}^p$  os sinais de desempenho e  $y(t) \in \mathbb{R}^q$  as medições disponíveis. A ideia central da síntese  $\mathcal{H}_\infty$  é projetar um controlador que minimize, na norma de operador  $\mathcal{H}_\infty$ , a função de transferência do sinal de perturbação  $w$  para o vetor de desempenho  $z$ . Formalmente, define-se a norma

$$\|T_{w \rightarrow z}\|_{\mathcal{H}_\infty} = \sup_{\omega \in \mathbb{R}} \bar{\sigma}(T_{w \rightarrow z}(j\omega)),\tag{2}$$

onde  $\bar{\sigma}(\cdot)$  denota o maior valor singular. A síntese consiste em encontrar um controlador que garanta  $\|T_{w \rightarrow z}\|_{\mathcal{H}_\infty} < \gamma$  para o menor  $\gamma > 0$  possível, o que equivale a impor uma cota uniforme em frequência para a amplificação de perturbações. Essa abordagem oferece um critério unificado para compensações entre atenuação de perturbações, margem de estabilidade e esforço de controle, e pode ser colocada

em forma de problemas convexos (por exemplo via desigualdades matriciais lineares, LMIs), permitindo tratamento numérico eficiente e com garantias.

A modelagem das medições e, em particular, a dinâmica dos sensores, exerce papel decisivo no sucesso prático de um projeto robusto. Na literatura e em projetos simplificados costuma-se assumir que as medições correspondem instantaneamente a estados ou saídas, mas a realidade impõe que cada canal de medição passa por condicionamento, filtragem e latências que introduzem dinâmica adicional. Se os sensores são modelados por filtros de primeira ordem, por exemplo por

$$\hat{z}_i(s) = \frac{1}{\tau_i s + 1} \hat{x}_i(s), \quad \tau_i > 0, \quad i = 1, \dots, n, \quad (3)$$

então, ao adotar uma representação no espaço de estados, cada filtro acrescenta um estado que descreve a evolução temporal da leitura do sensor. A inclusão desses estados transforma o modelo nominal de ordem  $n$  em um sistema aumentado de ordem  $2n$ , cujas equações podem ser escritas, em forma matricial, como

$$\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} A & B_1 K \\ \tau^{-1} & -\tau^{-1} \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} B_2 \\ 0 \end{bmatrix} w, \quad (4)$$

quando a lei de controle usa  $z$  como medições filtradas e  $K$  é a matriz de ganhos do controlador. Do ponto de vista prático, essa ampliação tem várias consequências: modos adicionais aparecem, constantes de tempo pequenas nos sensores introduzem componentes de alta frequência e rigidez numérica, e atrasos de fase acumulados podem reduzir margens de estabilidade. Portanto, projetar um controlador  $\mathcal{H}_\infty$  sem considerar explicitamente a dinâmica dos sensores pode levar a soluções que parecem satisfatórias no modelo reduzido, mas que perdem garantias quando implementadas no sistema real.

O impacto da dinâmica de sensores manifesta-se em aspectos de projeto e implementação. Em projeto, a presença de polos adicionais e de defasagens reduzidas em determinadas bandas de frequência impõe restrições sobre o ganho útil do controlador: para manter robustez é possível que se tenha de sacrificar parte do desempenho nominal. Em simulação e implementação, a inclusão de dinâmicas rápidas (valores de  $1/\tau_i$  elevados) pode causar rigidez matemática e exigir integra-

dores adequados; na prática, atenuar esse problema pode implicar em políticas de modelagem que desprezem dinâmicas extremamente rápidas ou, alternativamente, em um projeto que leve explicitamente em conta a rigidez numérica. Adicionalmente, a interação entre projeto robusto e limitações físicas (saturação de atuadores, ruído de medição) torna necessário avaliar, além do critério  $\mathcal{H}_\infty$ , métricas complementares como sensibilidade local, resposta transitória e requisitos de implementação.

Este trabalho tem por objetivo principal analisar o efeito das dinâmicas de sensores de primeira ordem sobre a síntese e o desempenho de controladores projetados pela técnica  $\mathcal{H}_\infty$ . Para tanto, procede-se à modelagem explícita do sistema aumentado (planta mais filtros de sensores), à formulação da síntese de controlador em termos de LMIs apropriadas a esse modelo e à comparação sistemática, via simulação e análise numérica, entre controladores obtidos com e sem a consideração das dinâmicas dos sensores. Serão também discutidas escolhas práticas relacionadas à regularização numérica, seleção de solvers para problemas SDP e procedimentos de validação (verificação de autovalores do sistema aumentado, teste de Lyapunov e aproximação da norma  $\mathcal{H}_\infty$  por grade de frequências).

A organização deste documento segue uma sequência lógica: no capítulo 1 dedicado aos fundamentos apresentam-se os resultados teóricos necessários à síntese  $\mathcal{H}_\infty$ , incluindo o lema do ganho limitado e as transformações que levam às LMIs utilizadas; na seção de modelagem e análise de resultados 3 são desenvolvidas as equações do sistema aumentado e discutidas as implicações de modelagem; a parte de síntese e implementação 2 descreve a formulação numérica, explicitando as variáveis de decisão, as condições de factibilidade e as escolhas de ferramentas ou programas que resolvem problemas complexos de otimização, matemática, engenharia ou computação, encontrando o melhor resultado possível; por fim, os resultados de simulação e discussão crítica evidenciam os trade-offs observados em 3 e derivam recomendações práticas para projetos de controle robusto em cenários com dinâmica de sensores não negligenciável em 4.

## 1.1 DECOMPOSIÇÃO DE KALMAN

A decomposição de Kalman organiza formalmente os estados de um sistema linear de modo a separar as dinâmicas que podem ser afetadas pela entrada das que não podem. (KALMAN, 1960) Essa separação é fundamental para o projeto de controladores de realimentação de estado, porque identifica exatamente quais modos são alocáveis via ganho e quais são intrinsecamente imutáveis pela ação de controle. Considere um sistema linear contínuo e invariante no tempo (SLIT) de uma entrada e uma saída (SISO) descrito por

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A} \mathbf{x}(t) + \mathbf{B} \mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C} \mathbf{x}(t) + \mathbf{D} \mathbf{u}(t),\end{aligned}\tag{5}$$

onde  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{u}, \mathbf{y} \in \mathbb{R}$ ,  $\mathbf{A} \in \mathbb{R}^{n \times n}$ ,  $\mathbf{B} \in \mathbb{R}^{n \times 1}$ ,  $\mathbf{C} \in \mathbb{R}^{1 \times n}$  e  $\mathbf{D} \in \mathbb{R}$ . A matriz de controlabilidade clássica é

$$\mathcal{C} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \cdots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \in \mathbb{R}^{n \times n},\tag{6}$$

e o posto  $k = \text{rank}(\mathcal{C})$  quantifica a dimensão do subespaço de estados que pode ser atingido a partir da origem por uma combinação de entradas. Quando  $k = n$  dizemos que o par  $(\mathbf{A}, \mathbf{B})$  é *completamente controlável*. Se  $k < n$ , existem exatamente  $n - k$  estados não controláveis. A decomposição de Kalman consiste em encontrar uma mudança de base invertível  $T \in \mathbb{R}^{n \times n}$  que expresse o sistema numa forma em blocos onde a ação de  $\mathbf{B}$  fica confinada ao subespaço controlável. Escrevendo  $\bar{\mathbf{A}} = T^{-1}\mathbf{A}T$ ,  $\bar{\mathbf{B}} = T^{-1}\mathbf{B}$  e  $\bar{\mathbf{C}} = CT$ , existe  $T$  tal que

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_c & \mathbf{A}_{12} \\ 0 & \mathbf{A}_{\bar{c}} \end{bmatrix}, \quad \bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B}_c \\ 0 \end{bmatrix}, \quad \bar{\mathbf{C}} = \begin{bmatrix} \mathbf{C}_c & \mathbf{C}_{\bar{c}} \end{bmatrix},\tag{7}$$

onde  $\mathbf{A}_c \in \mathbb{R}^{k \times k}$ ,  $\mathbf{B}_c \in \mathbb{R}^{k \times 1}$  representam a parte controlável (dimensão  $k$ ), e  $\mathbf{A}_{\bar{c}} \in \mathbb{R}^{(n-k) \times (n-k)}$  descreve os modos não controláveis. Na forma (7) a dinâmica do sistema escreve-se como

$$\begin{aligned}\dot{\mathbf{x}}_c(t) &= \mathbf{A}_c \mathbf{x}_c(t) + \mathbf{A}_{12} \mathbf{x}_{\bar{c}}(t) + \mathbf{B}_c \mathbf{u}(t), \\ \dot{\mathbf{x}}_{\bar{c}}(t) &= \mathbf{A}_{\bar{c}} \mathbf{x}_{\bar{c}}(t), \\ \mathbf{y}(t) &= \mathbf{C}_c \mathbf{x}_c(t) + \mathbf{C}_{\bar{c}} \mathbf{x}_{\bar{c}}(t) + \mathbf{D} \mathbf{u}(t),\end{aligned}$$

onde  $\mathbf{x}_c \in \mathbb{R}^k$  e  $\mathbf{x}_{\bar{c}} \in \mathbb{R}^{n-k}$  são os estados nos subespaços controlável e não-controlável, respectivamente. A decomposição tem consequências imediatas para o projeto de realimentação de estados. Considere a lei  $\mathbf{u}(t) = -K \mathbf{x}(t)$ . No novo referencial, escrevendo  $\bar{K} = KT = [K_c \ K_{\bar{c}}]$ , a dinâmica em malha fechada é dada por

$$\bar{A} - \bar{B}\bar{K} = \begin{bmatrix} A_c - B_c K_c & A_{12} - B_c K_{\bar{c}} \\ 0 & A_{\bar{c}} \end{bmatrix}.$$

Da expressão da matriz acima segue-se a fatoração da equação característica

$$\begin{aligned} \det(sI_n - (\bar{A} - \bar{B}\bar{K})) &= \det(sI_k - (A_c - B_c K_c)) \cdot \det(sI_{n-k} - A_{\bar{c}}). \\ &= \underbrace{\det(sI - A_c + B_c K_c)}_{\text{Controlável}} \cdot \underbrace{\det(sI - A_{\bar{c}})}_{\text{Não controlável}} \end{aligned}$$

Isto deixa claro o fato central: *apenas* os autovalores associados a  $A_c$  podem ser livremente movidos pela escolha de  $K_c$ ; os autovalores de  $A_{\bar{c}}$  permanecem inalterados pela realimentação de estados (DOYLE et al., 1992).

Uma forma direta e numericamente robusta de obter  $T$  parte da matriz de controlabilidade  $\mathcal{C}$ . Seja  $k = \text{rank}(\mathcal{C})$ . Obtenha uma base ortonormal  $U_c \in \mathbb{R}^{n \times k}$  para a imagem de  $\mathcal{C}$  (por exemplo, via decomposição em valores singulares  $\mathcal{C} = U\Sigma V^\top$  e tomando as  $k$  primeiras colunas de  $U$ , ou via algoritmo QR com pivotamento). Em seguida complete  $U_c$  até uma base de  $\mathbb{R}^n$  com  $U_{\bar{c}} \in \mathbb{R}^{n \times (n-k)}$  (por exemplo, usando a complementação ortogonal). Defina

$$T = [U_c \ U_{\bar{c}}] \in \mathbb{R}^{n \times n}, \quad (8)$$

que é invertível por construção. Neste caso os blocos da forma (7) são obtidos por  $\bar{A} = T^{-1}AT$  e  $\bar{B} = T^{-1}B$ . Em termos operacionais:

$$U_c = \text{orth}(\mathcal{C}), \quad T = [U_c \ U_{\bar{c}}], \quad \bar{A} = T^{-1}AT, \quad \bar{B} = T^{-1}B.$$

Esse procedimento é estável numericamente e evita a construção explícita de vetores de estado independentes por inspeção manual. Do ponto de vista de projeto, trabalhar na base de Kalman traz vantagens claras: a síntese do ganho  $K$  pode ser feita primeiramente para a parte controlável  $(A_c, B_c)$ , com garantia de que os

modos não-controláveis  $A_{\bar{c}}$  permanecerão inalterados e deverão ser estáveis por hipótese (ou então o sistema é intrinsecamente não estabilizável). Em particular, se existir qualquer autovalor instável em  $A_{\bar{c}}$ , não há nenhum ganho de realimentação de estados que torne o sistema globalmente assintoticamente estável; esse é um diagnóstico imediato que a decomposição torna explícito.

A decomposição tem ainda uma versão dual voltada à observabilidade: trocando  $(A, B)$  por  $(A^\top, C^\top)$  obtém-se a separação entre subespaços observáveis e não-observáveis, com consequências análogas para a síntese de observadores.

### 1.1.1 EXEMPLO NUMÉRICO

Considere

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

A matriz de controlabilidade é

$$\mathcal{C} = \begin{bmatrix} B & AB \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix},$$

portanto  $\text{rank}(\mathcal{C}) = 1$  e  $k = 1$ . Uma base para  $\text{Im}(\mathcal{C})$  é  $\mathbf{v}_1 = [1 \ 0]^\top$ ; completando com  $\mathbf{v}_2 = [0 \ 1]^\top$  obtemos  $T = [\mathbf{v}_1 \ \mathbf{v}_2] = I_2$  (caso simples). A forma em blocos resulta em

$$A_c = [1], \quad A_{12} = [1], \quad A_{\bar{c}} = [2], \quad B_c = [1].$$

Assim, ao escolher uma lei  $\mathbf{u} = -K\mathbf{x}$  com  $\bar{K} = [K_c \ K_{\bar{c}}]$ , apenas o autovalor associado a  $A_c$  (igual a 1) pode ser deslocado por  $K_c$ ; o polo em  $s = 2$  associado a  $A_{\bar{c}}$  é imutável pela realimentação. Esse exemplo ilustra claramente como a decomposição evidencia quais modos são efetivamente manipuláveis pelo controle.

## 1.2 A EQUAÇÃO DE LYAPUNOV

Considere a planta linear no espaço de estados descrita por

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t), \quad \mathbf{x}(0) = \mathbf{x}_0, \tag{9}$$

onde  $\mathbf{x}(t) \in \mathbb{R}^n$ ,  $\mathbf{u}(t) \in \mathbb{R}^m$ ,  $\mathbf{A} \in \mathbb{R}^{n \times n}$  e  $\mathbf{B} \in \mathbb{R}^{n \times m}$ . Assumimos que se pretende aplicar uma lei de realimentação de estados estática da forma

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t), \quad (10)$$

onde  $\mathbf{K} \in \mathbb{R}^{m \times n}$  é a matriz de ganhos a ser projetada. Substituindo (10) em (9) obtém-se a dinâmica em malha fechada

$$\dot{\mathbf{x}}(t) = \mathbf{A}_{\text{cl}}\mathbf{x}(t), \quad \mathbf{A}_{\text{cl}} := \mathbf{A} - \mathbf{B}\mathbf{K}. \quad (11)$$

A condição clássica de Lyapunov afirma que  $\mathbf{A}_{\text{cl}}$  é assintoticamente estável (todos os autovalores com parte real negativa) se, e somente se, existe uma matriz simétrica definida positiva  $P = P^\top \succ 0$  tal que

$$\mathbf{A}_{\text{cl}}^\top P + P\mathbf{A}_{\text{cl}} \prec 0. \quad (12)$$

A necessidade e suficiência desta condição decorrem do fato de que a função candidata de Lyapunov  $V(\mathbf{x}) = \mathbf{x}^\top P\mathbf{x}$  satisfaz  $\dot{V}(\mathbf{x}) = \mathbf{x}^\top (\mathbf{A}_{\text{cl}}^\top P + P\mathbf{A}_{\text{cl}})\mathbf{x}$ , que é negativa definida exatamente quando (12) é satisfeita; por outro lado, se  $\mathbf{A}_{\text{cl}}$  é Hurwitz então a equação de Lyapunov linear associada  $\mathbf{A}_{\text{cl}}^\top P + P\mathbf{A}_{\text{cl}} = -Q$  tem solução  $P \succ 0$  para qualquer  $Q = Q^\top \succ 0$ .

A desigualdade matricial (12) envolve simultaneamente  $P$  e  $\mathbf{K}$  de modo não linear, pois  $\mathbf{A}_{\text{cl}}$  depende de  $\mathbf{K}$ . Para transformar este problema em uma desigualdade linear (LMI) em incógnitas, procede-se por uma congruência adequada e por uma mudança de variáveis canônica. Primeiramente reescreve-se (12) substituindo  $\mathbf{A}_{\text{cl}} = \mathbf{A} - \mathbf{B}\mathbf{K}$  e expandindo os termos:

$$(\mathbf{A} - \mathbf{B}\mathbf{K})^\top P + P(\mathbf{A} - \mathbf{B}\mathbf{K}) = \mathbf{A}^\top P + P\mathbf{A} - \mathbf{K}^\top \mathbf{B}^\top P - P\mathbf{B}\mathbf{K}.$$

Esta forma é bilinear em  $(P, \mathbf{K})$ . (KHALIL, 2002) A técnica que elimina a não linearidade consiste em efetuar uma congruência por  $P^{-1}$  (supondo  $P \succ 0$  invertível) e introduzir a variável  $X := P^{-1}$ . Multiplique (12) à esquerda e à direita por  $X$  (congruência preserve a desigualdade de positividade sem alterar o sinal semidefinito) e obtenha

$$X\mathbf{A}_{\text{cl}}^\top + \mathbf{A}_{\text{cl}}X \prec 0. \quad (13)$$

Substituindo novamente  $A_{\text{cl}} = A - BK$  em (13) conduz-se a

$$XA^\top - XK^\top B^\top + AX - BKX \prec 0.$$

Note que  $X$  pode ser exigida simétrica positiva definida, isto é  $X = X^\top \succ 0$ , porque  $X = P^{-1}$  e inversa de matriz simétrica positiva é simétrica positiva. Introduz-se então a nova variável de projeto  $W \in \mathbb{R}^{m \times n}$  definida por

$$W := KX. \quad (14)$$

Com  $W$  a expressão acima simplifica-se porque, usando  $XK^\top = (KX)^\top = W^\top$  (valendo-se da simetria de  $X$ ), obtemos a desigualdade linear

$$AX + XA^\top - BW - W^\top B^\top \prec 0, \quad X = X^\top \succ 0. \quad (15)$$

A equação (15) é uma LMI convexa nas incógnitas  $X$  e  $W$ . Após a resolução numérica desta LMI, recupera-se a matriz de ganho buscada por inversão da mudança de variáveis  $K = W X^{-1}$  (BOYD et al., 1994). Para verificar a equivalência formal entre (12) e (15) observe-se que, partindo de qualquer solução  $P \succ 0$ , tomando  $X = P^{-1}$  e  $W = KX$  a desigualdade (12) se transforma em (15) por congruência e substituição; reciprocamente, partindo de  $X \succ 0$  e  $W$  que satisfazem (15), definindo  $P := X^{-1}$  e  $K := WX^{-1}$  recupera-se (12) por multiplicação por  $P$  (congruência reversa). Essa equivalência estabelece que encontrar  $(X, W)$  que satisfaçam (15) é matematicamente equivalente a encontrar  $(P, K)$  que cumpram (12).

A existência de solução para a LMI (15) está intimamente relacionada à propriedade de estabilizabilidade do par  $(A, B)$ . De fato, se existe  $K$  tal que  $A - BK$  é Hurwitz, então existe  $P \succ 0$  resolvendo (12); com  $X = P^{-1}$  e  $W = KX$  obtemos uma solução de (15). Inversamente, se existem  $X \succ 0$  e  $W$  satisfazendo (15), então definindo  $K = WX^{-1}$  segue que  $(A - BK)X + X(A - BK)^\top \prec 0$ ; fazendo a congruência por  $X^{-1}$  (ou, equivalentemente, definindo  $P = X^{-1}$ ) chegamos a  $A_{\text{cl}}^\top P + PA_{\text{cl}} \prec 0$ , o que implica que  $A_{\text{cl}}$  é Hurwitz. Portanto a existência de solução da LMI (15) é equivalente à existência de algum ganho  $K$  estabilizante; a condição necessária e suficiente para que tal ganho exista é precisamente a estabilizabilidade do par  $(A, B)$  (FRANCIS, 1997).

### 1.3 DERIVAÇÃO DA LMI VIA TRANSFORMAÇÃO DE VARIÁVEIS E CONGRUÊNCIA

Ao escrever a equação de malha fechada com realimentação de estados é comum encontrar duas convenções diferentes, que diferem apenas pelo sinal da lei de controle. Essas duas formas são equivalentes do ponto de vista dinâmico, mas é importante explicitar a convenção adotada no texto para evitar confusões na formulação das equações, na síntese de ganho e na implementação prática. Adota-se em um caso a lei de controle em (10) pela qual a matriz do sistema em malha fechada é (11). Na outra convenção a lei de controle é escrita como

$$\mathbf{u}(t) = +K \mathbf{x}(t), \quad (16)$$

e então a matriz em malha fechada passa a ser

$$A_{\text{cl}} = A + B K. \quad (17)$$

Essencialmente trata-se de uma diferença de sinal na definição do ganho  $K$ . As duas representações são formalmente equivalentes: se  $K$  estabiliza  $A - BK$  então  $\tilde{K} = -K$  estabiliza  $A + B\tilde{K}$ . Em termos de autovalores vale a igualdade

$$\sigma(A - BK) = \sigma(A + B(-K)), \quad (18)$$

ou seja, mudar a convenção apenas troca o sinal do ganho que se reporta às mesmas trajetórias do lugar das raízes. A escolha da convenção influencia diretamente a forma das desigualdades de Lyapunov/LMI empregadas na síntese. Suponha que se use a transformação canônica  $X = P^{-1}$  e a variável auxiliar  $W$  definida por  $W = KX$ . Para a convenção  $\mathbf{u} = -K\mathbf{x}$  a condição de Lyapunov  $(A - BK)^T P + P(A - BK) \prec 0$  pode ser escrita, em variáveis  $X$  e  $W$ , como a LMI (15), porque  $KX = W$  implica  $-BKK^T - XK^T B^T = -BW - W^T B^T$ . Se a convenção adotada for  $\mathbf{u} = +K\mathbf{x}$ , então a mesma manipulação conduz à LMI

$$AX + XA^T + BW + W^T B^T \prec 0, \quad X = X^T \succ 0, \quad (19)$$

ou seja, os termos envolvendo  $B$  e  $W$  aparecem com sinal oposto. Em ambos os casos a transformação é válida e a recuperação do ganho realiza-se por  $K =$

$WX^{-1}$  (lembrando, no entanto, que o sinal físico da ação de controle dependerá da convenção adotada inicialmente). Agora considere o sistema em malha fechada com realimentação de estado:

$$\dot{x}(t) = (A + B_1 K) x(t) + B_2 w(t), \quad z(t) = C x(t), \quad (20)$$

com o objetivo de garantir que o ganho do sistema  $T_{w \rightarrow z}$  satisfaça:

$$\|T_{w \rightarrow z}\|_\infty < \gamma. \quad (21)$$

Seja  $V(x) = x^T P x$ , com  $P = P^T \succ 0$ . Relembrando a condição de dissipação:

$$\dot{V}(x) + z^T z - \gamma^2 w^T w < 0.$$

Como:

$$\dot{x} = (A + B_1 K)x + B_2 w, \quad z = Cx,$$

a derivada de  $V$  é:

$$\begin{aligned} \dot{V}(x) &= \dot{x}^T P x + x^T P \dot{x} \\ &= [(A + B_1 K)x + B_2 w]^T P x + x^T P [(A + B_1 K)x + B_2 w] \\ &= x^T (A + B_1 K)^T P x + 2x^T P B_2 w + x^T P (A + B_1 K)x. \end{aligned}$$

Logo, temos:

$$x^T ((A + B_1 K)^T P + P(A + B_1 K) + C^T C) x + 2x^T P B_2 w - \gamma^2 w^T w < 0. \quad (22)$$

Definindo o vetor estendido  $\eta := \begin{bmatrix} x \\ w \end{bmatrix}$ , a inequação (22) pode ser escrita como:

$$\underbrace{\eta^T \begin{bmatrix} A_{cl}^T P + P A_{cl} + C^T C & P B_2 \\ B_2^T P & -\gamma^2 I \end{bmatrix} \eta}_{\Theta(P, K)} < 0,$$

com  $A_{cl} = A + B_1 K$ . Assim, requeremos:

$$\Theta(P, K) \prec 0, \quad \text{com } P \succ 0. \quad (23)$$

Como  $\Theta$  é *não linear* (bilinear em  $P$  e  $K$ ), realizamos as substituições:

$$Q := P^{-1}, \quad W := KQ. \quad (24)$$

Observe que:

$$K = WQ^{-1}, \quad P = Q^{-1}. \quad (25)$$

Escrevemos a matriz  $\Theta$  explicitamente como:

$$\Theta = \begin{bmatrix} A^T P + PA + K^T B_1^T P + PB_1 K + C^T C & PB_2 \\ B_2^T P & -\gamma^2 I \end{bmatrix}.$$

Substituímos (25), logo,

$$K^T B_1^T P = (WQ^{-1})^T B_1^T Q^{-1} = Q^{-1} W^T B_1^T Q^{-1},$$

$$PB_1 K = Q^{-1} B_1 W Q^{-1}.$$

Somando os dois:

$$K^T B_1^T P + PB_1 K = Q^{-1} (W^T B_1^T + B_1 W) Q^{-1}.$$

Similarmente:

$$A^T P + PA = Q^{-1} A^T + AQ^{-1}.$$

Assim, a parte superior esquerda da matriz se torna:

$$Q^{-1} (A^T + A + W^T B_1^T + B_1 W) Q^{-1} + C^T C.$$

Para eliminar os termos em  $Q^{-1}$  e obter uma inequação em  $Q$ , aplicamos uma transformação de congruência com:

$$T = \text{diag}(Q, I, I).$$

Note que, para qualquer matriz simétrica  $M$  (BOYD et al., 1994), temos:

$$M \prec 0 \iff TMT^T \prec 0.$$

A matriz original é:

$$M = \begin{bmatrix} A^T P + PA + K^T B_1^T P + PB_1 K + C^T C & PB_2 & C^T \\ B_2^T P & -\gamma I & 0 \\ C & 0 & -\gamma I \end{bmatrix}.$$

Substituindo  $P = Q^{-1}$  e aplicando a congruência:

$$TMT^T = \begin{bmatrix} Q & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} Q^{-1}(A^T + A + B_1 W + W^T B_1^T)Q^{-1} + C^T C & Q^{-1}B_2 & C^T \\ B_2^T Q^{-1} & -\gamma I & 0 \\ C & 0 & -\gamma I \end{bmatrix} \begin{bmatrix} Q & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}.$$

O resultado é:

$$\begin{bmatrix} AQ + QA^T + B_1 W + W^T B_1^T + C^T C Q & QB_2 & (CQ)^T \\ B_2^T Q & -\gamma I & 0 \\ C Q & 0 & -\gamma I \end{bmatrix} \prec 0, \quad Q \succ 0. \quad (26)$$

A desigualdade acima é linear em  $Q$ ,  $W$  e  $\gamma$  e pode ser resolvida via programação convexa. ([VANDENBERGHE; BOYD, 1996](#)) O ganho ótimo do controlador é recuperado com (25). Esse é o resultado final do **Bounded Real Lemma** reformulado como uma LMI para projeto de controladores  $\mathcal{H}_\infty$  ([GEROMEL; KOROGUI, 2019](#)).

## 1.4 ESTABILIDADE

Considerando novamente a planta (37) e suponha que o controlador adotado seja proporcional, com a lei de ação

$$u(t) = K_p [r(t) - y_m(t)],$$

onde  $y_m(t)$  é a medida fornecida pelo sensor e  $K_p$  é o ganho proporcional. Denotando por  $H(s)$  a função de transferência do sensor, a função de transferência em malha fechada que relaciona a referência  $r$  à saída  $y$  é dada por

$$T(s) = \frac{K_p G(s)}{1 + K_p G(s) H(s)}. \quad (27)$$

A estabilidade do laço fechado é determinada pelos polos de  $T(s)$ , isto é, pelas raízes da equação característica

$$1 + K_p G(s) H(s) = 0. \quad (28)$$

No caso ideal em que o sensor é instantâneo,  $H(s) = 1$ , a condição de estabilidade reduz-se a  $1 + K_p G(s) = 0$ . Quando o sensor é dinâmico e modelado por um filtro

de primeira ordem  $H(s) = (1 + \tau s)^{-1}$ , a equação característica pode ser manipulada para evidenciar como a dinâmica do sensor altera a ordem e os coeficientes do polinômio característico. Multiplicando ambos os lados de  $1 + K_p G(s)H(s) = 0$  pelo denominador comum  $(1 + \tau s)(\alpha_4 s^4 + \alpha_3 s^3 + \alpha_2 s^2 + \alpha_1 s + \alpha_0)$  obtém-se

$$(1 + \tau s)(\alpha_4 s^4 + \alpha_3 s^3 + \alpha_2 s^2 + \alpha_1 s + \alpha_0) + K_p \beta_0 = 0.$$

Expandindo, a equação característica escrita como polinômio em  $s$  é

$$\tau \alpha_4 s^5 + (\alpha_4 + \tau \alpha_3) s^4 + (\alpha_3 + \tau \alpha_2) s^3 + (\alpha_2 + \tau \alpha_1) s^2 + (\alpha_1 + \tau \alpha_0) s + (\alpha_0 + K_p \beta_0) = 0.$$

Deste desenvolvimento seguem duas observações imediatas. A presença do sensor introduz um termo em  $s^5$ , isto é, a ordem do denominador aumenta em uma unidade relativamente ao caso sem sensor. Além disso, o ganho  $K_p$  atua apenas no termo constante do polinômio expandido, deslocando o coeficiente livre de  $\alpha_0$  para  $\alpha_0 + K_p \beta_0$ . Para  $K_p = 0$  os polos fechados coincidem com as raízes de  $(1 + \tau s)A(s) = 0$ , que incluem os polos originais da planta e o polo adicional do sensor em  $s = -1/\tau$ .

A estabilização pode ser então analisada através da dependência dos coeficientes do polinômio em função de  $K_p$  e  $\tau$ . Uma condição necessária, ainda que não suficiente, para que todos os polos possuam parte real negativa é que todos os coeficientes do polinômio acima sejam estritamente positivos; em particular requer-se  $\tau \alpha_4 > 0$ ,  $\alpha_4 + \tau \alpha_3 > 0$  e  $\alpha_0 + K_p \beta_0 > 0$ . O termo constante evidencia como um aumento significativo de  $K_p$  pode, em certas configurações, inverter a sinalização dos coeficientes de baixa ordem e comprometer a estabilidade. Uma verificação completa exige a aplicação de um critério de estabilidade de polinômios, por exemplo o critério de Routh–Hurwitz aplicado ao polinômio de quinta ordem acima, cujas relações entre os coeficientes estabelecem limites explícitos sobre  $K_p$  em função de  $\tau$ .

Uma visão complementar e prática é obtida a partir do laço aberto  $L(s) = K_p G(s) H(s)$ . Pelo critério de Nyquist, a quantidade de polos fechados instáveis depende do comportamento do diagrama de Nyquist de  $L(j\omega)$  em torno do ponto  $-1$ . A inclusão de  $H(s) = (1 + \tau s)^{-1}$  reduz o ganho em altas frequências e introduz

um defasamento de fase igual a  $-\arctan(\omega\tau)$ , diminuindo assim a margem de fase nas bandas onde  $|K_p G(j\omega)|$  é da ordem de unidade. Essa redução de margem pode fazer com que cruzamentos estáveis no caso  $H(s) = 1$  se tornem instáveis quando o sensor dinâmico está presente. Por isso, o efeito do sensor manifesta-se não apenas pelo aumento da ordem do polinômio característico, mas também por uma perda de margem de fase que torna o sistema mais sensível a aumentos de ganho e a variações paramétricas (DULLERUD; PAGANINI, 2000).

No limite em que  $\tau \rightarrow 0^+$  recupera-se o caso ideal: a expressão polinomial reduz-se a  $A(s) + K_p\beta_0 = 0$ , de ordem quatro, e o polo do sensor desloca-se para  $-\infty$ , deixando de interferir na região dos polos dominantes. Se  $\tau$  aumenta (sensor mais lento), o termo em  $s^5$  e os termos cruzados envolvendo  $\tau$  tornam-se relevantes nas altas frequências, a resposta torna-se mais lenta e os modos de alta frequência passam a ter maior influência na dinâmica global, exigindo frequentemente a redução de  $K_p$  para preservar a estabilidade.

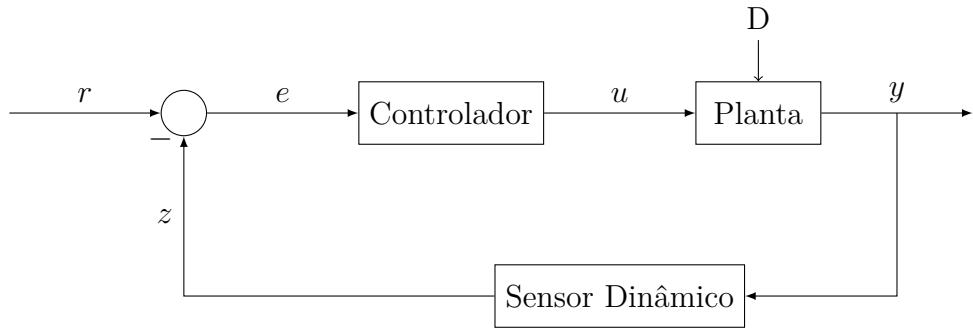
A análise do lugar das raízes é um instrumento eficaz para visualizar como os polos se deslocam quando  $K_p$  cresce a partir de zero. Para  $K_p = 0$  os polos fechados são os polos da planta e o polo do sensor em  $s = -1/\tau$ . À medida que  $K_p$  aumenta, cinco trajetórias no plano  $s$  descrevem a evolução dos polos, e a colocação desejada dos polos dominantes pode, em muitos casos, ser obtida mantendo  $K_p$  dentro de uma faixa que preserve as relações de sinais necessárias nas condições de Hurwitz e que assegure margem de fase suficiente segundo análise de Bode/Nyquist. Quando essas condições não se verificam, torna-se necessário alterar a estrutura do controlador; um ganho proporcional pode não ser suficiente para compensar o atraso introduzido por  $H(s)$ , e a inclusão de compensadores de avanço de fase ou de controladores dinâmicos mais sofisticados pode ser necessária para restaurar margem e desempenho.

Para a prática do projeto e validação recomenda-se o cálculo das margens de ganho e de fase a partir de  $L(j\omega)$ , a verificação da estabilidade por meio do critério de Nyquist e, em paralelo, a aplicação do critério de Routh–Hurwitz ao polinômio característico explicitado acima para obter limites analíticos em  $K_p$  condicionados

a  $\tau$ . Simulações temporais realizadas em ambientes como Xcos ou com rotinas de **scipy** ilustram que o sensor ideal conduz a respostas mais rápidas e menor erro em regime permanente para a mesma amplitude de ganho  $K_p$ , enquanto a presença de  $H(s) = (1+\tau s)^{-1}$  desloca os polos de modo que, a ganho fixo, tipicamente se observa degradação na velocidade de resposta e aumento do amortecimento aparente; a compensação adequada pode restaurar desempenho aceitável, entretanto implica, na prática, redução de largura de banda do controlador ou adição de elementos de avanço de fase para recuperar as margens perdidas.

## 2 EXPERIMENTAÇÃO

Seja a planta dada por,



cuja planta tem-se,

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}_s u(t) + \mathbf{B} w(t) \quad (29)$$

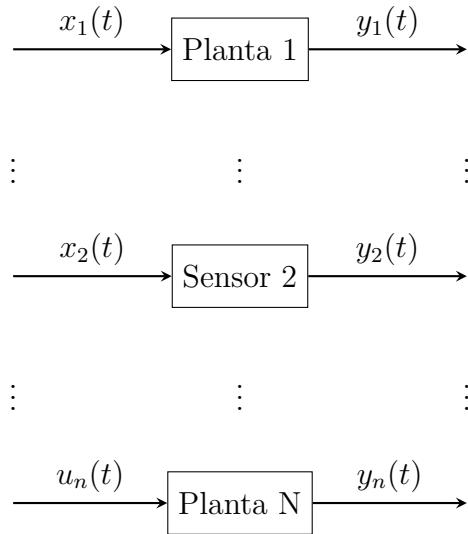
inserção da perturbação

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t). \quad (30)$$

onde,  $\mathbf{x}(t)$  é o vetor de estado,  $u(t)$  é o vetor de entrada,  $y(t)$  é o vetor de saída,  $A$  é a matriz de estado,  $B$  é a matriz de entrada,  $C$  é a matriz de saída,  $D$  é a matriz de transmissão direta, e por fim, para o sensor temos:

$$\begin{cases} \dot{\mathbf{x}}_s(t) = \mathbf{A}_s \mathbf{x}_s(t) + \mathbf{B}_s \mathbf{x}(t), \\ \mathbf{z}(t) = \mathbf{C}_s \mathbf{x}_s(t) + \mathbf{D}_s \mathbf{x}(t). \end{cases} \quad (31)$$

onde,  $x(t)$  é o vetor de estado,  $u(t)$  é o vetor de controle,  $w(t)$  é o vetor de distúrbios,  $z(t)$  é o vetor de saídas de desempenho. As matrizes  $A$ ,  $B$ ,  $B_s$ ,  $C_s$ ,  $D_s$ , e  $D_s$  têm dimensões apropriadas. Seja o controlador dado por: ( $u = kx$ ), admitiremos para o nosso estudo somente casos de sensores de primeira ordem.



Para o caso do sensor perfeito,  $z = x$ , o resultado obedece o *Bounded Real Lemma*.

### Suposições

1. Cada "Planta  $i$ " é descrita por uma EDO linear de ordem  $n_i$ , com entrada  $x_i(t)$  e saída  $y_i(t)$ .
2. As plantas podem ser vistas como sistemas do tipo  $n$ -ésima ordem, possivelmente com coeficientes e condições diferentes.

Cada planta  $i$  pode ser descrita pela equação diferencial linear da forma:

$$a_{n_i} \frac{d^{n_i} y_i(t)}{dt^{n_i}} + a_{n_i-1} \frac{d^{n_i-1} y_i(t)}{dt^{n_i-1}} + \cdots + a_1 \frac{dy_i(t)}{dt} + a_0 y_i(t) = b_{m_i} \frac{d^{m_i} x_i(t)}{dt^{m_i}} + \cdots + b_1 \frac{dx_i(t)}{dt} + b_0 x_i(t), \quad (32)$$

onde:

- $n_i$  é a ordem do sistema;
- $m_i \leq n_i$  representa a maior derivada da entrada  $x_i(t)$ ;
- $a_k, b_k$  são coeficientes constantes associados à planta  $i$ .

A função de transferência  $G_i(s)$  é obtida ao aplicar a Transformada de Laplace na EDO (com condições iniciais nulas):

$$G_i(s) = \frac{Y_i(s)}{X_i(s)} = \frac{b_{m_i}s^{m_i} + b_{m_i-1}s^{m_i-1} + \cdots + b_1s + b_0}{a_{n_i}s^{n_i} + a_{n_i-1}s^{n_i-1} + \cdots + a_1s + a_0}, \quad (33)$$

onde  $s$  é a variável de Laplace. O sistema geral é composto por  $N$  plantas. A dinâmica de cada planta é independente e representada por suas próprias EDOs e funções de transferência. Para simulação, tomando a planta de quarta ordem  $x = [x_1, x_2, x_3, x_4]$  e sensores de primeira ordem para cada variável de estado, simulando no XCOS do SCILAB ([CONSORTIUM, 2021](#)) para uma entrada senoidal, tem-se as seguintes restrições de projeto. A equação de estado do sistema é:

$$\text{Planta: } \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t) \end{cases} \quad (34)$$

A entrada  $u(t)$  será uma senoide. Cada sensor será modelado por:

$$G_{\text{sensor}_i}(s) = \frac{1}{\tau_i s + 1} \quad (35)$$

A planta é o sistema físico que se deseja controlar. No nosso exemplo, ela é definida por uma equação diferencial de 4ª ordem, que pode ser escrita como:

$$\alpha_4 \frac{d^4x}{dt^4} + \alpha_3 \frac{d^3x}{dt^3} + \alpha_2 \frac{d^2x}{dt^2} + \alpha_1 \frac{dx}{dt} + \alpha_0 x = \beta_0 u. \quad (36)$$

Em termos de função de transferência, considerando  $x(s)$  a transformada de Laplace da saída  $x(t)$  e  $u(s)$  a da entrada, temos:

$$G(s) = \frac{x(s)}{u(s)} = \frac{\beta_0}{\alpha_4 s^4 + \alpha_3 s^3 + \alpha_2 s^2 + \alpha_1 s + \alpha_0}. \quad (37)$$

## 2.1 DADOS DO SISTEMA

Considera-se a seguinte realização em espaço de estados, obtida da função de transferência:

$$G(s) = \frac{1}{s^4 + 2s^3 + 5s^2 + 3s + 4},$$

utilizando a forma canônica controlável:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -4 & -3 & -5 & -2 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

O LMI de Lyapunov para o sistema com realimentação de estados é (19). A solução para este conjunto de desigualdades possibilita a obtenção do ganho de realimentação  $K$  que estabiliza o sistema com (26).

### Restrições de projeto

- Constante de tempo do sensor em torno do dobro da variável sensorada.
- Varredura frequêncial da perturbação.

A implementação conforme código em python com *CVXPY* (GRANT; BOYD, 2014) para resolver as LMIs e obter as matrizes  $P$  e  $W$  está em A. Ao executar o algoritmo, tem-se, status do solver: **optimal**, matriz  $P$  encontrada:

$$P = \begin{bmatrix} 9.45854186 \times 10^{-7} & 8.00298732 \times 10^{-8} & 2.85424264 \times 10^{-8} & 2.87039302 \times 10^{-7} \\ 8.00298732 \times 10^{-8} & 1.18353680 \times 10^{-6} & -6.77102408 \times 10^{-8} & 2.07849932 \times 10^{-9} \\ 2.85424264 \times 10^{-8} & -6.77102408 \times 10^{-8} & 1.33004457 \times 10^{-6} & -6.37666996 \times 10^{-7} \\ 2.87039302 \times 10^{-7} & 2.07849932 \times 10^{-9} & -6.37666996 \times 10^{-7} & 8.07057267 \times 10^{-7} \end{bmatrix}$$

Matriz  $W$  encontrada:

$$W = \begin{bmatrix} 0 & 0 & 0 & -3.10099846 \end{bmatrix}$$

Ganho de realimentação  $K = WP^{-1}$ :

$$K = \begin{bmatrix} 2480123.97878784 & -369385.39997333 & -3761503.39407201 & -7695501.3866239 \end{bmatrix}$$

No projeto de sistemas de controle, a realimentação é crucial para garantir a estabilidade e o desempenho desejado. No entanto, a implementação prática dos sensores pode apresentar características que diferem do modelo ideal.

1. **Sensor Perfeito:** Assume-se que a medição dos estados é instantânea e sem erros, ou seja, se a variável de interesse é  $x_1(t)$ , a saída medida é

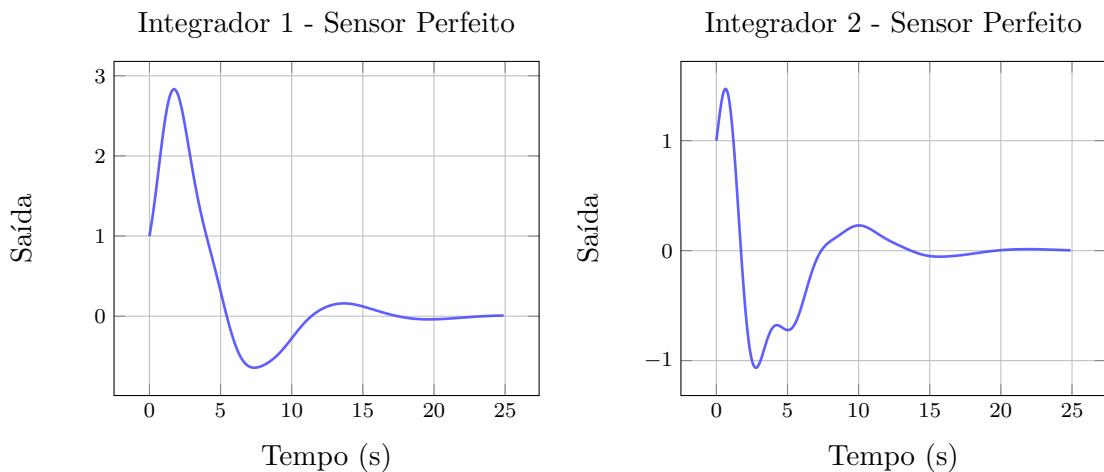
$$y_m(t) = x_1(t).$$

2. **Sensor Imperfeito (Dinâmico de 1ª Ordem):** Sensores reais apresentam dinâmica interna que pode ser modelada por uma função de transferência de primeira ordem:

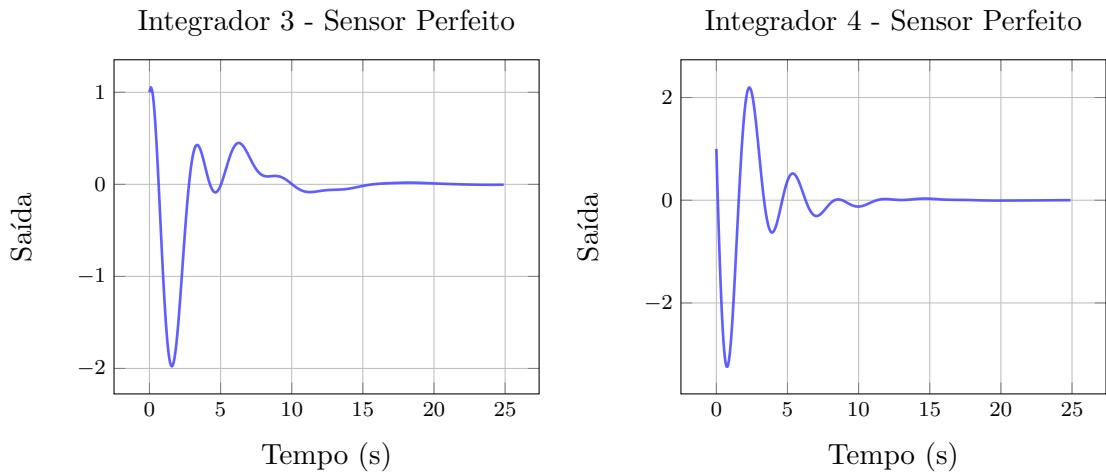
$$H(s) = \frac{1}{\tau s + 1},$$

onde  $\tau > 0$  é a constante de tempo. Assim, a medição efetiva passa a ser

$$y_m(s) = H(s)x(s) = \frac{x(s)}{\tau s + 1}.$$



**Figura 2.1 – Comparação entre os dois integradores**



**Figura 2.2 – Comparação entre os Integradores 3 e 4**

Das figuras 2.1 e 2.2, para uma dada entrada, um estado inicial diferente de zero, e um sinal de perturbação, a saída do sistema converge assintóticamente para o valor de referência o que confirma que o ganho K obtido da simulação no PYTHON utilizando o CVXPY estabiliza o sistema em malha fechada.

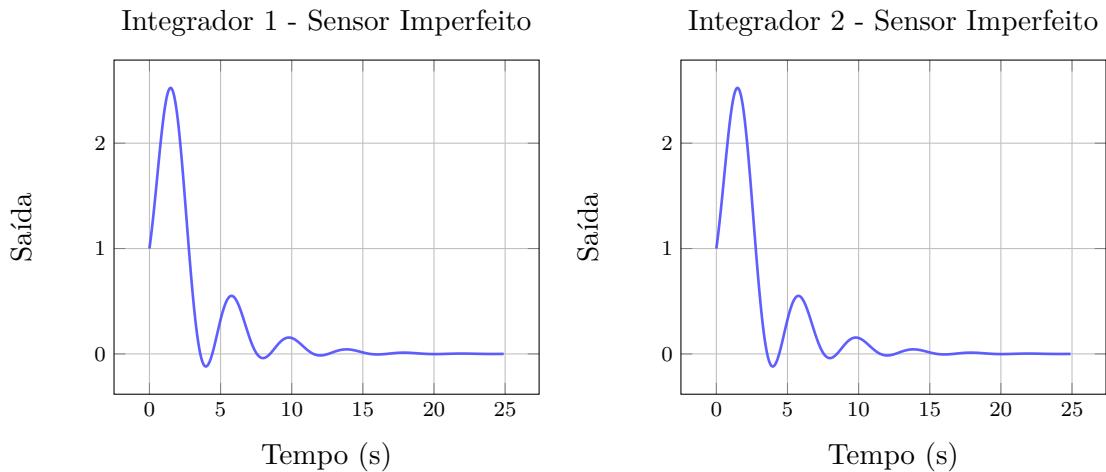
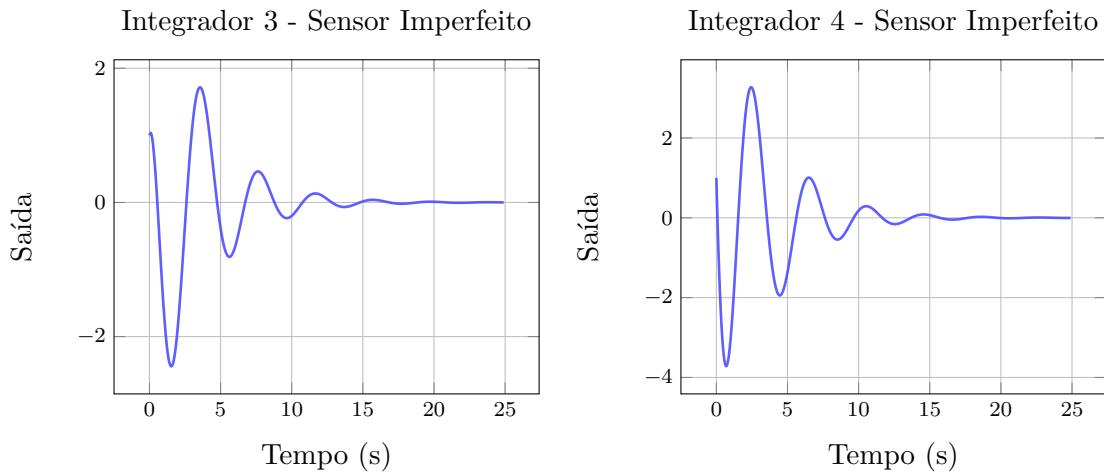


Figura 2.3 – Comparação entre os Integradores 1 e 2 com Sensor Imperfeito



**Figura 2.4 – Comparação entre os Integradores 3 e 4 com Sensor Imperfeito**

Das figuras 2.3 e 2.4, com as mesmas condições de simulação agora para sensor dinâmico a saída do sistema também converge assintóticamente para o valor de referência. Com  $y_m(t) = x_1(t)$ , a realimentação é imediata e sem alteração do sinal. Dessa modo, a resposta em malha fechada tende a acompanhar a referência de forma rápida e precisa. Ao considerar o sensor com dinâmica, o sinal medido é defasado e atenuado  $y_m(s) = \frac{x(s)}{\tau s + 1}$ . Esse filtro de primeira ordem age como um *filtro de baixa frequência*, introduzindo atraso na resposta e podendo reduzir a amplitude do sinal de realimentação, o que pode afetar negativamente o desempenho e as margens de estabilidade do sistema.

### 3 ANÁLISE DE RESULTADOS

Nesta seção apresentarei a análise detalhada dos resultados experimentais e numéricos obtidos ao confrontar ganhos K sintetizados em Python (usando o Lema do Ganho Limitado / Bounded Real Lemma e, separadamente, a síntese  $\mathcal{H}_\infty$ ) com a implementação desses mesmos ganhos no ambiente de simulação XCOS do SCILAB. Quando o sensor é dinâmico de primeira ordem e a planta está sujeita a perturbação. Na sequência de argumentos mostro por que um ganho que estabiliza o modelo nominal pode provocar instabilidade quando aplicado ao sistema real com sensor dinâmico, e indica os passos matemáticos e práticos necessários para diagnosticar e corrigir o problema. Fixando a notação em (1), modelando a entrada de perturbação. Na etapa de síntese numérica, a hipótese adotada implicitamente foi a de realimentação de estado perfeita, ou seja, a lei:  $u(t) = -Kx(t)$  com K obtida resolvendo a LMI resultante do Lema do Ganho Limitado ou a LMI  $\mathcal{H}_\infty$  para o sistema nominal sem incluir a dinâmica do sensor. Aplicando a lei da realimentação acima em (1) a dinâmica em malha fechada nominal tem matriz

$$A_{cl}^{nom} = A - BK.$$

A verificação de estabilidade na etapa numérica em Python foi feita checando-se os autovalores de  $A_{cl}^{nom}$  e, quando aplicável, solucionando a LMI de Lyapunov  $A_{cl}^{nom \top} P + PA_{cl}^{nom} \prec 0$  para  $P \succ 0$ .

#### 3.1 PROJETO DE CONTROLADOR $H_\infty$

O sistema dinâmico considerado foi (29) e (30) com,

$$A = \begin{bmatrix} -2 & 1 & 0 & 0 \\ -1 & -3 & 0 & 1 \\ 0 & 0 & -4 & 2 \\ 0 & -1 & -1 & -2 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad C = [1 \ 0 \ 0 \ 0].$$

Este sistema possui 4 estados, 1 entrada de controle, 1 perturbação, e 1 saída de desempenho. Para obter um ganho de realimentação de estados  $u(t) = Kx(t)$

que minimize a norma  $\|T_{w \rightarrow z}\|_\infty$ , foi aplicada a formulação LMI do Bounded Real Lemma. A LMI foi resolvida no Python utilizando a biblioteca `cvxpy` ([DIAMOND; BOYD, 2016](#)), com o solver `MOSEK` ([APS, 2020](#)), e a seguinte mudança de variáveis em (24). A otimização retornou:

1. Matriz  $Q$  definida positiva;
2. Matriz  $W$ ;
3. Valor mínimo  $\gamma^*$ ;
4. Ganho  $K = WQ^{-1}$ .

Após obter o ganho  $K$ , verificou-se se os autovalores da matriz em malha fechada  $A + B_2K$  possuíam parte real negativa. Isso confirma a estabilidade assintótica do sistema sob a realimentação proposta. Ao executar o algoritmo em [B](#), a saída esperada,

Status do solver: optimal

Matriz P encontrada:

```
[[ 9.45854186e-07  8.00298732e-08  2.85424264e-08  2.87039302e-07]
 [ 8.00298732e-08  1.18353680e-06 -6.77102408e-08  2.07849932e-09]
 [ 2.85424264e-08 -6.77102408e-08  1.33004457e-06 -6.37666996e-07]
 [ 2.87039302e-07  2.07849932e-09 -6.37666996e-07  8.07057267e-07]]
```

Matriz W encontrada:

```
[[ 0.          0.          0.          -3.10099846]]
```

Ganho de realimentação  $K = W \cdot P^{-1}$ :

```
[[ 2480123.97878784 -369385.39997333 -3761503.39407201 -7695501.3866239 ]]
```

### 3.1.1 IMPLEMENTAÇÃO NO XCOS: SENSOR PERFEITO

Para o caso de sensor perfeito, a medição do vetor de estados  $x(t)$  é direta. No XCOS, isso foi implementado conectando diretamente os blocos: **State-Space** com matriz  $A + B_2K$ ; **Sum** para adicionar o efeito de  $B_1w(t)$ ; **Gain** com matriz  $C$  para obter a saída  $z(t)$ . O sistema simulado recebe uma entrada  $w(t)$  do tipo **SENOIDE** e responde com  $z(t)$ , sendo a medida de desempenho do sistema frente a distúrbios.

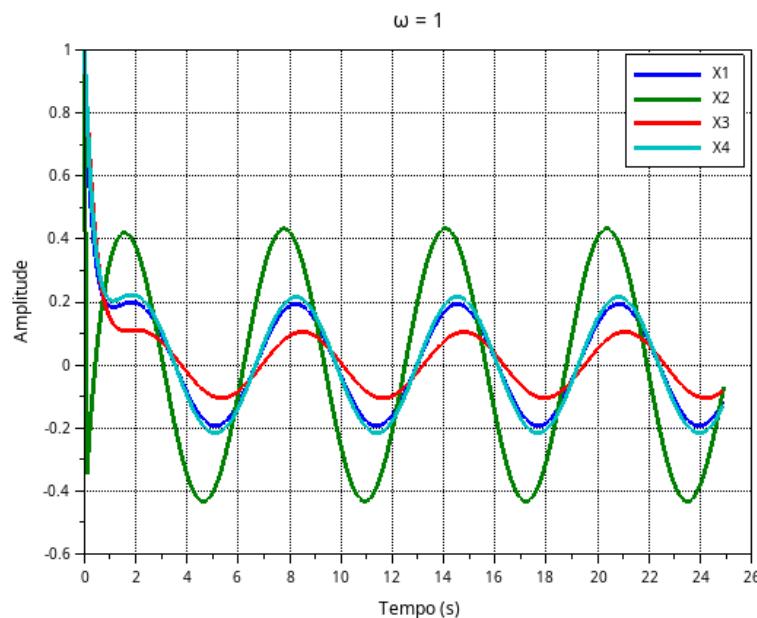
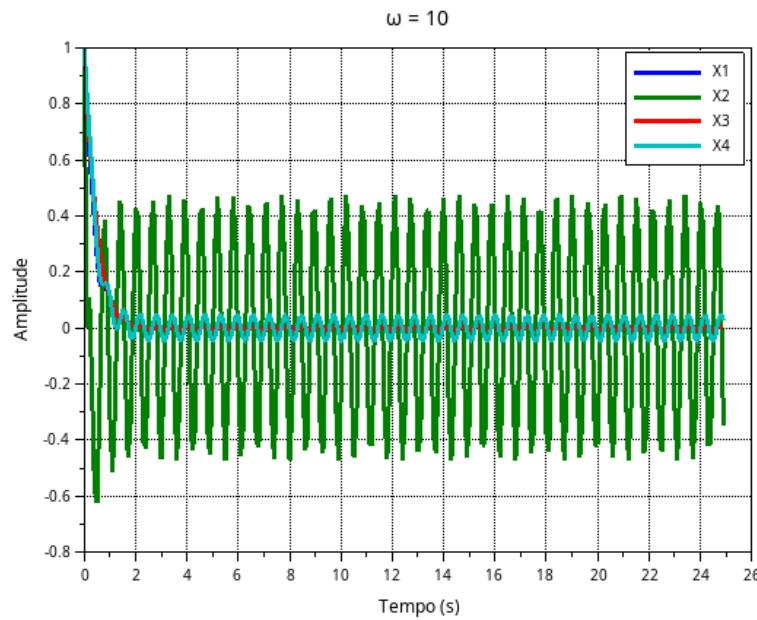
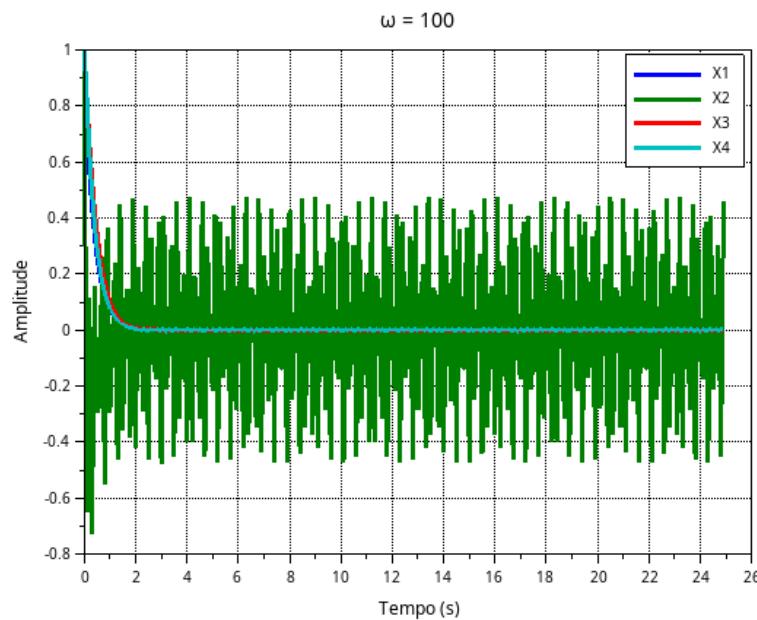


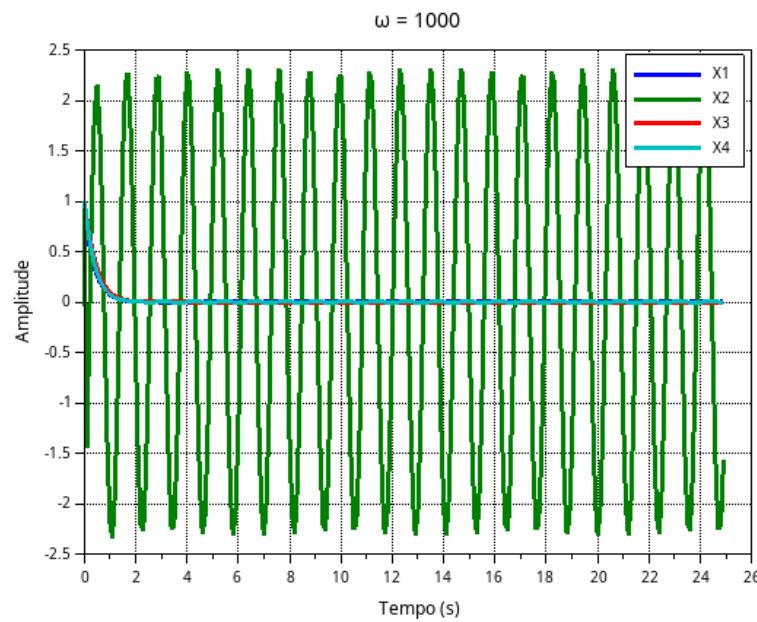
Figura 3.1 – Resposta em frequência para  $\omega = 1$ .



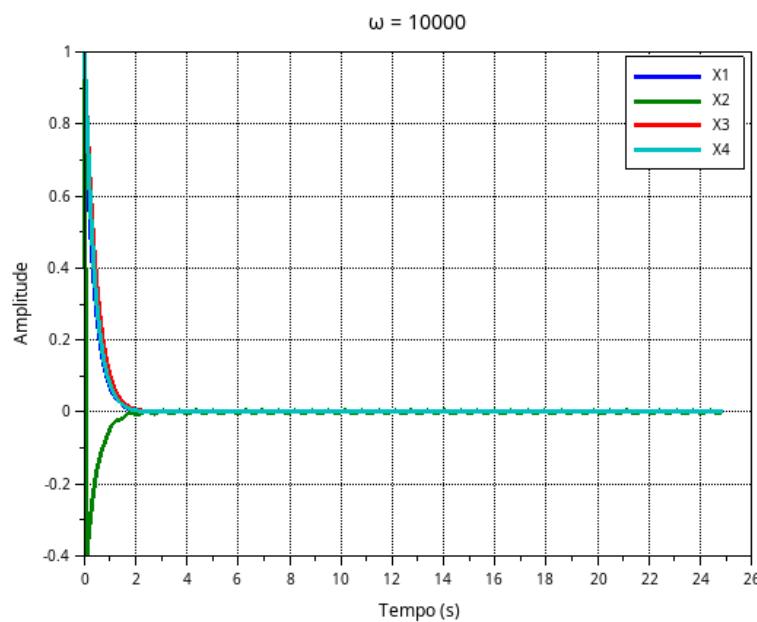
**Figura 3.2 – Resposta em frequênci para  $\omega = 10$ .**



**Figura 3.3 – Resposta em frequênci para  $\omega = 100$ .**



**Figura 3.4 – Resposta em frequência para  $\omega = 1000$ .**

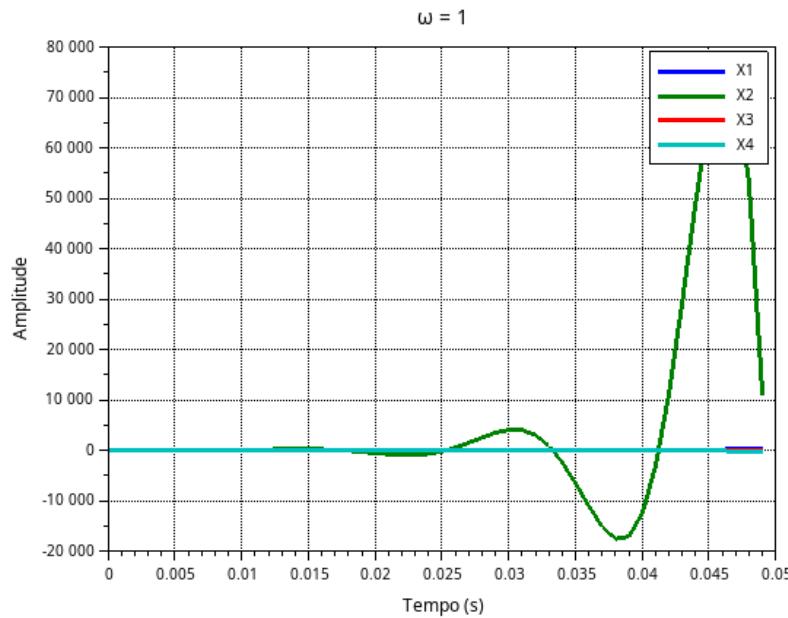


**Figura 3.5 – Resposta em frequência para  $\omega = 10000$ .**

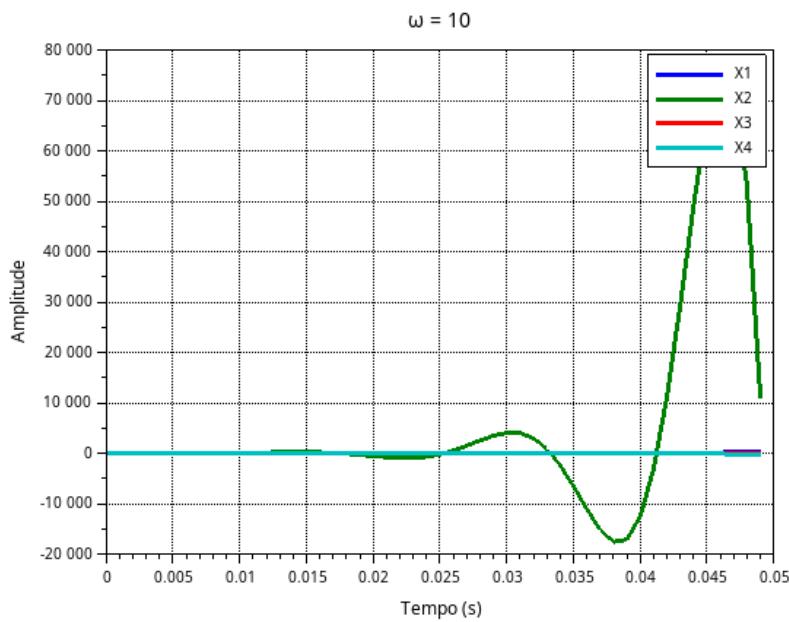
Dos gráficos 3.1, 3.2, 3.3, 3.4 e 3.5 foi aplicado um sinal de entrada  $u$  cujo valor era uma senoide, que variava em uma escala de tempo  $t$ , a idéia central é analisar o valor da saída das quatro variáveis de estado  $x_1, x_2, x_3, x_4$  a medida que a frequência varia. Nos gráficos citados para sensor perfeito, a variação da frequência faz o sinal da saída convergir assintoticamente para um valor de referência, também, é notório salientar que a resposta em frequência nas figuras citadas considera um estado inicial não nulo e o sistema responde rapidamente à perturbação com amortecimento desejável e atinge  $\|T_{w \rightarrow z}\|_\infty < \gamma^*$ , o que reitera a constatação teórica utilizada no Lema do ganho infinito.

### 3.1.2 IMPLEMENTAÇÃO NO XCOS: SENSOR DINÂMICO

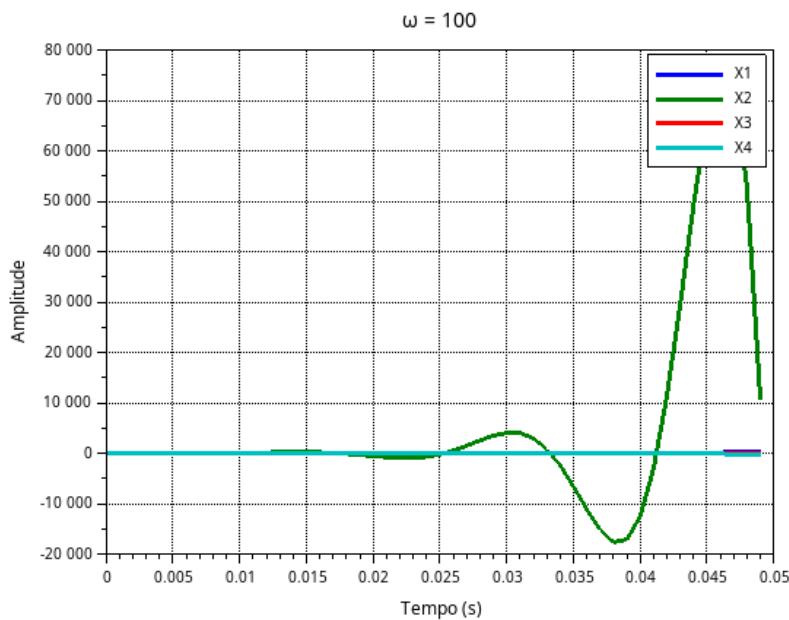
Para o cenário com sensor dinâmico, a leitura de  $x(t)$  passa por um sistema auxiliar que simula imperfeições na medição. Este subsistema foi implementado com: Um integrador ou sistema de ordem 1, a saída medida  $x_m(t)$  sendo utilizada para a realimentação:  $u(t) = Kx_m(t)$ . O atraso ou filtragem introduzido impacta a resposta do sistema ao distúrbio. A estrutura no XCOS incluiu: Blocos Gain, Sum, Integrator para modelar a dinâmica do sensor; Realimentação com ganho  $K$ ;



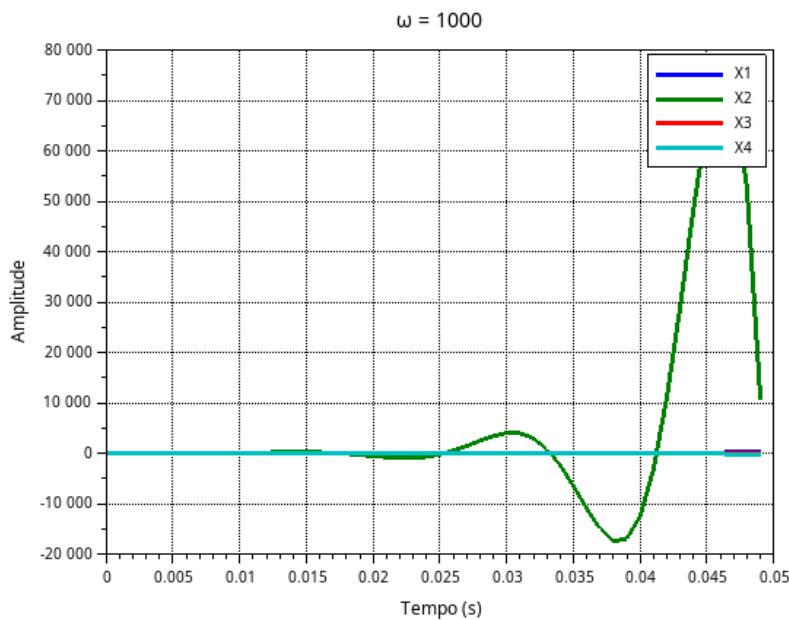
**Figura 3.6 – Resposta em frequência para  $\omega = 1$  (sensor imperfeito).**



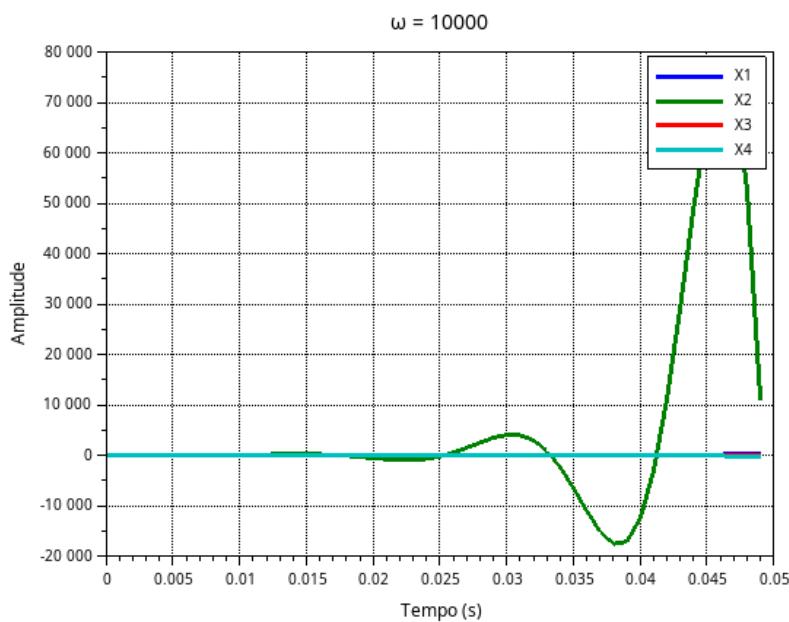
**Figura 3.7 – Resposta em frequência para  $\omega = 10$  (sensor imperfeito).**



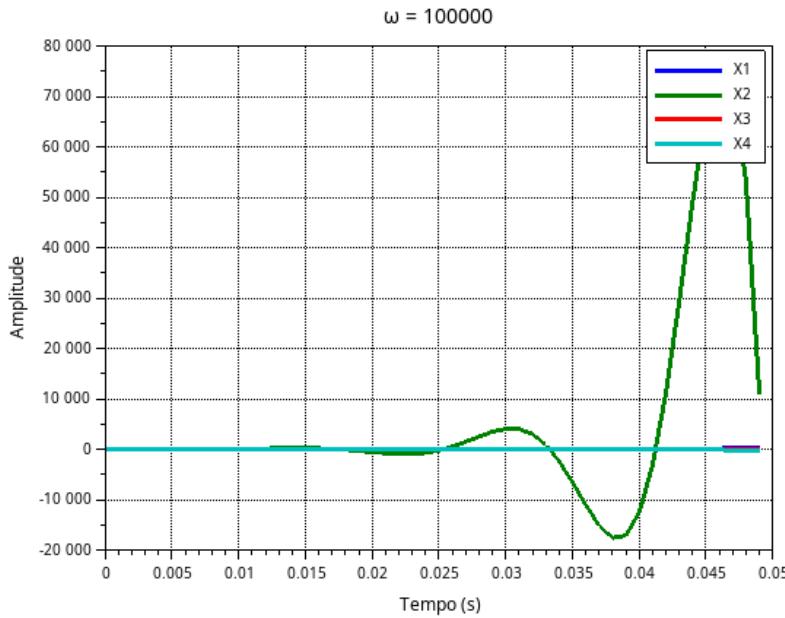
**Figura 3.8 – Resposta em frequência para  $\omega = 100$  (sensor imperfeito).**



**Figura 3.9 – Resposta em frequência para  $\omega = 1000$  (sensor imperfeito).**



**Figura 3.10 – Resposta em frequência para  $\omega = 10000$  (sensor imperfeito).**



**Figura 3.11 – Resposta em frequência para  $\omega = 100000$  (sensor imperfeito).**

Para os gráficos 3.6, 3.7, 3.8, 3.9, 3.10, 3.11 com sensor dinâmico, a resposta é mais lenta, podendo conter oscilações ou ultrapassagens devido à filtragem de  $x(t)$ . A diferença de desempenho é coerente com a degradação esperada ao introduzir imperfeições na realimentação. Nos gráficos anteriormente citados, para as variáveis de estado estudadas, a saída quando a frequência aumenta instabiliza o sistema para uma faixa marginal de tempo, neste caso, seria necessário fazer uma formulação matemática de *estado aumentado* afim de se obter uma melhor resposta da saída em malha fechada dadas as condições iniciais não nulas de simulação e as características dos sensores. Por fim, é importante frizar que os gráficos tratam de resposta completa que é a resposta de estado (zero/natural) mais a resposta de entrada (zero/forçada), nisto o leitor por constatar que não houve mudança, mas houve pois a resposta completa se sobrescreve a resposta de estado não nulo.

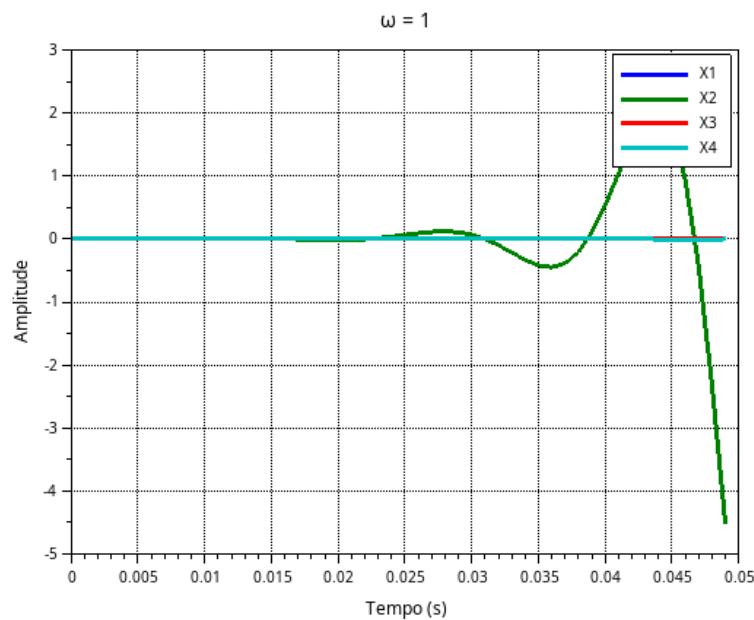


Figura 3.12 – Resposta em frequência para  $\omega = 1$  (sensor imperfeito) - Estado Nulo

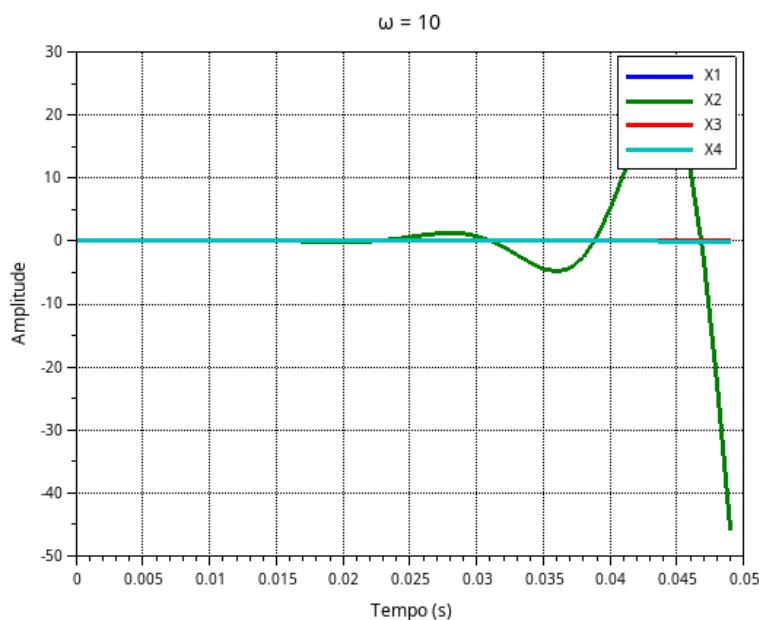


Figura 3.13 – Resposta em frequência para  $\omega = 10$  (sensor imperfeito) - Estado Nulo

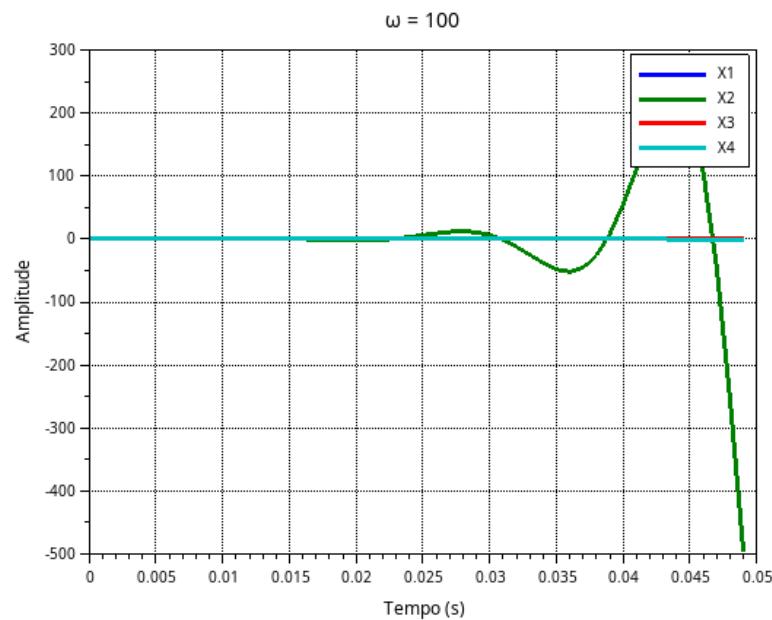


Figura 3.14 – Resposta em frequênci para  $\omega = 100$  (sensor imperfeito) - Estado Nulo

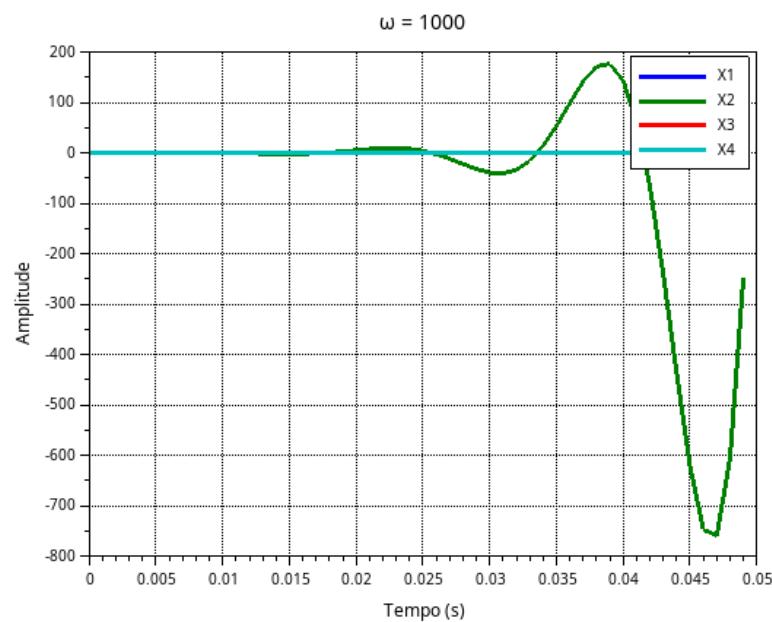
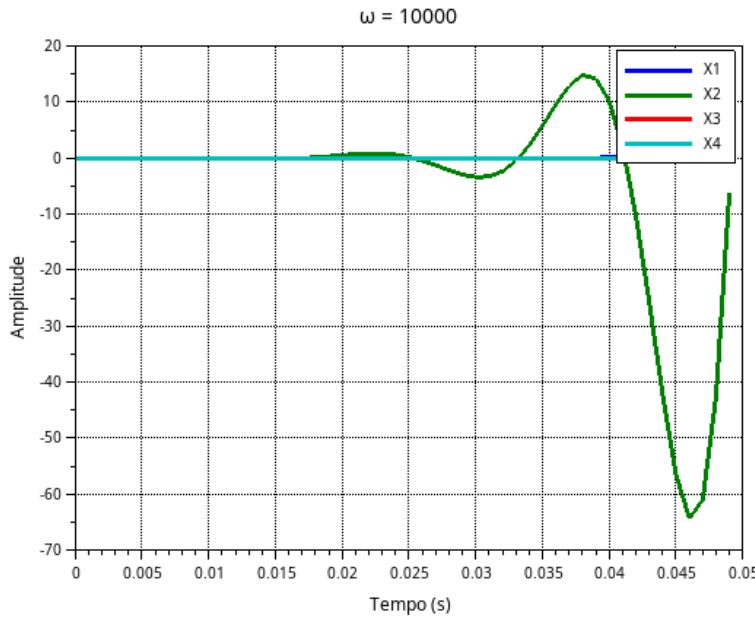


Figura 3.15 – Resposta em frequênci para  $\omega = 1000$  (sensor imperfeito) - Estado Nulo



**Figura 3.16 – Resposta em frequência para  $\omega = 10000$  (sensor imperfeito) - Estado Nulo**

Para os gráficos 3.12, 3.13, 3.14, 3.15, 3.16 o leitor pode perceber melhor a diferença entre os gráficos, pois, estamos considerando o estado inicial nulo cuja resposta completa, neste caso, é a resposta de estado nulo, mesmo considerando as mesmas condições de simulação de entrada e condicionamento dos sensores das simulações em 3.6, 3.7, 3.8, 3.9, 3.10, 3.11.

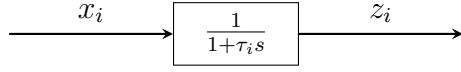
### 3.1.3 PROJETO EM ESTADO AUMENTADO

Considerando a planta linear no espaço de estados (1) com  $x(t) \in \mathbb{R}^n$ ,  $u(t) \in \mathbb{R}^m$ ,  $w(t) \in \mathbb{R}^r$ , e  $z(t) \in \mathbb{R}^n$  sendo o vetor formado pelas leituras do sensor (um sensor por componente de  $x$ ). Cada sensor é modelado por um filtro de primeira ordem (no domínio de Laplace, aplicado componente-a-componente):

$$\hat{z}_i(s) = \frac{1}{\tau_i s + 1} \hat{x}_i(s), \quad \tau_i > 0, \quad i = 1, \dots, n. \quad (38)$$

espaço de estado para sensor,

$$\tau_i s + \hat{z}_i = \hat{x}_i \xrightarrow{TIL} \tau_i \dot{z}_i + z_i = x_i \quad (39)$$



Em forma vetorial, definindo a matriz diagonal  $\tau = \text{diag}(\tau_1, \dots, \tau_n) \succ 0$ , tem-se

$$Z(s) = (I_n + \tau s)^{-1} X(s). \quad (40)$$

A lei de controle em realimentação de estado (usando as leituras do sensor) é

$$u(t) = K z(t), \quad (41)$$

onde  $K \in \mathbb{R}^{m \times n}$  é a matriz de ganhos (no caso  $m = 1$ ,  $K$  é um vetor linha  $1 \times n$ ). Aplicando transformada de Laplace à (1) (assumindo condições nulas iniciais para clareza) obtemos

$$sX(s) = AX(s) + B_1U(s) + B_2W(s).$$

Substituindo  $U(s) = KZ(s) = K(I_n + \tau s)^{-1}X(s)$  da (40):

$$sX(s) = AX(s) + B_1K(I_n + \tau s)^{-1}X(s) + B_2W(s). \quad (42)$$

Multiplicando ambos os lados por  $I_n + \tau s$  (matriz invertível para  $\Re(s) > -\min_i 1/\tau_i$ ):

$$(I_n + \tau s)sX(s) = (I_n + \tau s)AX(s) + B_1KX(s) + (I_n + \tau s)B_2W(s).$$

Reorganizando e agrupando por potências de  $s$  resulta

$$\tau s^2 X(s) + s(I_n - \tau A)X(s) - (A + B_1K)X(s) = \tau s B_2 W(s) + B_2 W(s). \quad (43)$$

Aplicando a transformada inversa  $s \mapsto \frac{d}{dt}$  obtemos a EDO (tempo contínuo) equivalente,

$$\tau \ddot{x}(t) + (I_n - \tau A)\dot{x}(t) - (A + B_1K)x(t) = \tau B_2 \dot{w}(t) + B_2 w(t). \quad (44)$$

A equação (44) é uma forma compacta de mostrar que a presença do filtro no caminho de realimentação introduz termos de segunda derivada em  $x$  e também derivadas de  $w$  na entrada, ou seja, não existe representação causal de primeira ordem somente em  $x$  sem introduzir estados auxiliares (ou usar derivadas de  $w$ ), ou seja, supondo uma planta de ordem  $n$ . Como cada sensor está ligado a uma

variável de estado da planta e gera uma nova variável de estado, então o conjunto planta + sensor, tem ordem  $2n$ . De (39), chamando  $z_i = x_{n+i}$ , temos:

$$\tau_i \dot{x}_{n+i} = -x_{n+i} + x_i \implies \dot{x}_{n+i} = -\frac{1}{\tau_i} x_{n+i} + x_i$$

então, a realimentação de estado com sensor de primeira ordem fica:

$$\mathbf{u} = [k_1 \quad k_2 \quad \cdots \quad k_n] [x_{n+1} \quad \cdots \quad x_{2n}]^T \quad (45)$$

espaço de estado para sensor:

$$\frac{d}{dt} \begin{bmatrix} \dot{x}_{n+1} \\ \dot{x}_{n+2} \\ \vdots \\ \dot{x}_{2n} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau_1} & 0 & \cdots & 0 \\ 0 & -\frac{1}{\tau_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{1}{\tau_n} \end{bmatrix} \begin{bmatrix} x_{n+1} \\ x_{n+2} \\ \vdots \\ x_{2n} \end{bmatrix} + \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (46)$$

$$\begin{aligned} \dot{\mathbf{x}}_s &= \begin{bmatrix} \dot{x}_{n+1} \\ \vdots \\ \dot{x}_{2n} \end{bmatrix} & \mathbf{x}_s &= \begin{bmatrix} x_{n+1} \\ \vdots \\ x_{2n} \end{bmatrix} \\ \text{diag}\left(-\frac{1}{\tau_i}\right) &= \begin{bmatrix} -\frac{1}{\tau_1} & 0 & \cdots & 0 \\ 0 & -\frac{1}{\tau_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{1}{\tau_n} \end{bmatrix} \end{aligned}$$

Definindo  $z(t)$  como estado do filtro:

$$\tau \dot{z}(t) + z(t) = x(t) \implies \dot{z}(t) = \frac{1}{\tau} x(t) - \frac{1}{\tau} z(t).$$

Com  $u(t) = Kz(t)$  o sistema aumentado  $x_{\text{aug}} = [x^\top \ z^\top]^\top$  tem dinâmica:

$$\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \underbrace{\begin{bmatrix} A & B_1 K \\ \frac{1}{\tau} I_n & -\frac{1}{\tau} I_n \end{bmatrix}}_{A_{\text{aug}}} \begin{bmatrix} x \\ z \end{bmatrix} + \underbrace{\begin{bmatrix} B_2 \\ 0_{n \times r} \end{bmatrix}}_{B_{\text{aug}}} w.$$

### Observações:

1.  $A_{\text{aug}} \in \mathbb{R}^{2n \times 2n}$ ,  $B_{\text{aug}} \in \mathbb{R}^{2n \times r}$ .
2. Estabilidade do laço fechado avalia-se pelos autovalores de  $A_{\text{aug}}$ .
3. Para  $\tau \rightarrow 0$ , retorna-se  $z \rightarrow x$  e  $A_{\text{aug}}$  reduz a  $\dot{x} = (A + B_1 K)x$ .

Para obter uma representação de primeira ordem via estado aumentado adequada à simulação XCOS e à análise do sistema, introduzimos o vetor de estados do sensor

$$z(t) = \begin{bmatrix} z_1(t) \\ \vdots \\ z_n(t) \end{bmatrix}, \quad \tau_i \dot{z}_i(t) + z_i(t) = x_i(t), \quad (47)$$

ou, em forma matricial,

$$\tau \dot{z}(t) + z(t) = x(t) \implies \dot{z}(t) = \tau^{-1}x(t) - \tau^{-1}z(t). \quad (48)$$

Com a lei de controle (41) temos o sistema em malha fechada descrito por:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 K z(t) + B_2 w(t), \\ \dot{z}(t) &= \tau^{-1}x(t) - \tau^{-1}z(t). \end{aligned} \quad (49)$$

Agrupando os estados em  $x_{\text{aug}}(t) = \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} \in \mathbb{R}^{2n}$ , a dinâmica acima escreve-se de forma compacta

$$\dot{x}_{\text{aug}}(t) = A_{\text{aug}}(K, \tau) x_{\text{aug}}(t) + B_{\text{aug}} w(t), \quad (50)$$

com

$$A_{\text{aug}}(K, \tau) = \begin{bmatrix} A & B_1 K \\ \tau^{-1} & -\tau^{-1} \end{bmatrix} \in \mathbb{R}^{2n \times 2n}, \quad B_{\text{aug}} = \begin{bmatrix} B_2 \\ 0_{n \times r} \end{bmatrix}. \quad (51)$$

**Neste arranjo:**

1.  $A$  e  $B_1K$  são blocos  $n \times n$  (observe que  $B_1 \in \mathbb{R}^{n \times m}$  e  $K \in \mathbb{R}^{m \times n}$ , portanto  $B_1K \in \mathbb{R}^{n \times n}$ );
2.  $\tau^{-1} = \text{diag}(1/\tau_1, \dots, 1/\tau_n)$  (bloco  $n \times n$ );
3.  $B_{\text{aug}}$  alimenta apenas as equações de  $x$  — as equações de  $z$  não recebem perturbação direta (modelo causal do sensor).

Se a saída de desempenho desejada for  $y(t) = Cx(t)$ , a saída aumentada correspondente é

$$y(t) = \underbrace{\begin{bmatrix} C & 0_{p \times n} \end{bmatrix}}_{C_{\text{aug}}} x_{\text{aug}}(t).$$

A equação (44) pode ser escrita como um sistema de primeira ordem definindo  $v(t) = \dot{x}(t)$ . Então a variável de estado estendida  $\xi(t) = \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \in \mathbb{R}^{2n}$  satisfaz

$$\dot{x}(t) = v(t),$$

$$\tau \dot{v}(t) = -\left(I_n - \tau A\right)v(t) + \left(A + B_1K\right)x(t) + \tau B_2\dot{w}(t) + B_2w(t).$$

Essa forma contém  $\dot{w}(t)$  na entrada — portanto é menos adequada para simulação causal quando  $w$  é um sinal arbitrário (não-diferenciável). Por isso, a representação por  $x_{\text{aug}} = [x^\top \ z^\top]^\top$  dada em (50) é, em geral, a escolha recomendada para análise e implementação. A estabilidade assintótica do sistema em malha fechada (incluindo a dinâmica do sensor) é verificada pelos autovalores de  $A_{\text{aug}}(K, \tau)$ . Um certificado numérico adicional pode ser obtido resolvendo, para  $Q \succ 0$ ,

$$A_{\text{aug}}^\top Q + QA_{\text{aug}} = -I_{2n},$$

e verificando  $Q \succ 0$ .

1. **Caso**  $\tau = \tau I_n$ . Se todos os sensores têm a mesma constante de tempo  $\tau > 0$ , então  $\tau^{-1} = \frac{1}{\tau}I_n$  e

$$A_{\text{aug}} = \begin{bmatrix} A & B_1K \\ \frac{1}{\tau}I_n & -\frac{1}{\tau}I_n \end{bmatrix}.$$

2. **Limite**  $\tau \rightarrow 0^+$ . Quando  $\tau$  tende a zero (filtro rápido),  $z(t) \rightarrow x(t)$  e  $A_{\text{aug}}$  reduz, no limite, à dinâmica aproximada  $\dot{x} = (A + B_1 K)x + B_2 w$  (o subsistema do sensor se torna algebraico).
3. **Projeto levando em conta o sensor.** Se se deseja sintetizar  $K$  levando em conta explicitamente a dinâmica do sensor, a síntese (por exemplo via LMI ou otimização  $\mathcal{H}_\infty$ ) deve considerar o sistema aumentado  $(A_{\text{aug}}, B_{\text{aug}}, C_{\text{aug}})$ .

Chamando  $x = [x_1, \dots, x_n]^\top$  e  $z = [z_1, \dots, z_n]^\top$ , para  $i = 1, \dots, n$  temos

$$\dot{x}_i = \sum_{j=1}^n A_{ij}x_j + \sum_{j=1}^m (B_1)_{ij} \left( \sum_{\ell=1}^n K_{j\ell} z_\ell \right) + \sum_{j=1}^r (B_2)_{ij} w_j, \quad (52)$$

$$\tau_i \dot{z}_i = -z_i + x_i. \quad (53)$$

Dessa forma, o componente  $x_{n+i}$  do vetor aumentado representa  $z_i$  e obedece

$$\dot{x}_{n+i} = -\frac{1}{\tau_i} x_{n+i} + \frac{1}{\tau_i} x_i, \quad i = 1, \dots, n, \quad (54)$$

o que mostra explicitamente a estrutura de (50). A representação aumentada (50)–(51) é a forma prática e causal recomendada para análise, síntese e simulação (Xcos/Simulink). A EDO de segunda ordem (44) é útil para entender os efeitos dinâmicos introduzidos pelo filtro (termos  $\tau \ddot{x}$  e  $\tau B_2 \dot{w}$ ), mas para implementação numérica e projeto de controladores usa-se normalmente a forma de primeira ordem com estados auxiliares  $z$ . Usando as matrizes da simulação para dinâmica de sensor perfeito temos,

$$A = \begin{bmatrix} -2 & 1 & 0 & 0 \\ -1 & -3 & 0 & 1 \\ 0 & 0 & -4 & 2 \\ 0 & -1 & -1 & -2 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \quad D = [0].$$

Com  $K = [k_1 \ k_2 \ k_3 \ k_4]$  (linha  $1 \times 4$ ),

$$B_1 K = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ k_1 & k_2 & k_3 & k_4 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Com  $n = 4$  e  $I_4$  identidade  $4 \times 4$ ,

$$A_{\text{aug}} = \begin{bmatrix} A & B_1 K \\ \frac{1}{\tau} I_4 & -\frac{1}{\tau} I_4 \end{bmatrix}.$$

Escrevendo explicitamente (cada bloco  $4 \times 4$ ):

$$A_{\text{aug}} = \begin{bmatrix} -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -3 & 0 & 1 & k_1 & k_2 & k_3 & k_4 \\ 0 & 0 & -4 & 2 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & -2 & 0 & 0 & 0 & 0 \\ \frac{1}{\tau} & 0 & 0 & 0 & -\frac{1}{\tau} & 0 & 0 & 0 \\ 0 & \frac{1}{\tau} & 0 & 0 & 0 & -\frac{1}{\tau} & 0 & 0 \\ 0 & 0 & \frac{1}{\tau} & 0 & 0 & 0 & -\frac{1}{\tau} & 0 \\ 0 & 0 & 0 & \frac{1}{\tau} & 0 & 0 & 0 & -\frac{1}{\tau} \end{bmatrix},$$

Vetor de entrada aumentado:

$$B_{\text{aug}} = \begin{bmatrix} B_2 \\ 0_{4 \times 1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Tomando (39) e (40) ou, em forma matricial,

$$\tau \dot{z}(t) + z(t) = x(t) \implies \dot{z}(t) = \tau^{-1}x(t) - \tau^{-1}z(t). \quad (55)$$

Montamos o sistema *aumentado* (entradas ainda livres  $u$  e  $w$ ):

$$\dot{x}_{\text{aug}}(t) = A_{\text{aug}}^{(\text{open})} x_{\text{aug}}(t) + B_{\text{aug}}^{(u)} u(t) + B_{\text{aug}}^{(w)} w(t), \quad (56)$$

com

$$x_{\text{aug}} = \begin{bmatrix} x \\ z \end{bmatrix} \in \mathbb{R}^{2n}, \quad A_{\text{aug}}^{(\text{open})} = \begin{bmatrix} A & 0 \\ \tau^{-1} & -\tau^{-1} \end{bmatrix}, \quad (57)$$

$$B_{\text{aug}}^{(u)} = \begin{bmatrix} B_1 \\ 0_{n \times m} \end{bmatrix}, \quad B_{\text{aug}}^{(w)} = \begin{bmatrix} B_2 \\ 0_{n \times r} \end{bmatrix}. \quad (58)$$

**Pergunta:**

O par  $(A_{\text{aug}}^{(\text{open})}, B_{\text{aug}}^{(u)})$  é controlável?

A matriz de controlabilidade do sistema aumentado é

$$\mathcal{C}_{\text{aug}} = \begin{bmatrix} B_{\text{aug}}^{(u)} & A_{\text{aug}}^{(\text{open})} B_{\text{aug}}^{(u)} & \cdots & (A_{\text{aug}}^{(\text{open})})^{2n-1} B_{\text{aug}}^{(u)} \end{bmatrix} \in \mathbb{R}^{2n \times 2nm}. \quad (59)$$

Para estudar sua estrutura iremos calcular explicitamente as colunas  $(A_{\text{aug}}^{(\text{open})})^k B_{\text{aug}}^{(u)}$ . Defina, para  $k \geq 0$ ,

$$\Phi_k := (A_{\text{aug}}^{(\text{open})})^k B_{\text{aug}}^{(u)} = \begin{bmatrix} \top_k \\ \perp_k \end{bmatrix}, \quad (60)$$

onde  $\top_k, \perp_k \in \mathbb{R}^{n \times m}$  são os blocos superior e inferior respectivamente. Temos a recorrência

$$\Phi_{k+1} = A_{\text{aug}}^{(\text{open})} \Phi_k = \begin{bmatrix} A & 0 \\ \tau^{-1} & -\tau^{-1} \end{bmatrix} \begin{bmatrix} \top_k \\ \perp_k \end{bmatrix} = \begin{bmatrix} A\top_k \\ \tau^{-1}\top_k - \tau^{-1}\perp_k \end{bmatrix}. \quad (61)$$

Com condição inicial  $\top_0 = B_1$ ,  $\perp_0 = 0$  obtemos por indução:

$$\top_k = A^k B_1, \quad k \geq 0. \quad (62)$$

E a recorrência para  $\perp_k$  é

$$\perp_{k+1} = \tau^{-1}\top_k - \tau^{-1}\perp_k. \quad (63)$$

Resolve-se essa recorrência com somas telescópicas,

$$\perp_k = \sum_{i=0}^{k-1} (-1)^{k-1-i} \tau^{-(k-i)} A^i B_1, \quad k \geq 1, \quad \perp_0 = 0. \quad (64)$$

Assim cada bloco coluna  $\Phi_k$  é determinado pelo vetor  $A^k B_1$  (topo) e por combinações lineares dos vetores  $\{A^i B_1\}_{i=0}^{k-1}$  (fundo). Em particular, a *parte superior* da matriz de controlabilidade aumentada é exatamente a matriz de controlabilidade da planta (colocando colunas  $B_1, AB_1, \dots$ ). Se a planta  $(A, B_1)$  não é controlável, isto é,

$$\text{rank} [B_1 \ AB_1 \ \cdots \ A^{n-1} B_1] < n,$$

então a parte superior de  $\mathcal{C}_{\text{aug}}$  terá posto  $< n$ . Consequentemente  $\text{rank } \mathcal{C}_{\text{aug}} < 2n$  e o sistema aumentado não é controlável. Portanto:

### Necessidade:

$(A, B_1)$  controlável é necessário para que o sistema aumentado seja controlável.

Agora demonstramos que a controlabilidade de  $(A, B_1)$  é também suficiente para a controlabilidade do sistema aumentado. Se  $(A, B_1)$  é controlável então  $(A_{\text{aug}}^{(\text{open})}, B_{\text{aug}}^{(u)})$  é controlável. Usamos o critério PBH: o par  $(A_{\text{aug}}^{(\text{open})}, B_{\text{aug}}^{(u)})$  é não controlável se, e somente se, existe  $\lambda \in \mathbb{C}$  e vetor linha não-nulo  $(p^\top, q^\top) \in \mathbb{C}^{1 \times 2n}$  tais que

$$(p^\top, q^\top) [\lambda I_{2n} - A_{\text{aug}}^{(\text{open})} \ B_{\text{aug}}^{(u)}] = 0.$$

Separando blocos, isto é equivalente ao sistema linear (em incógnitas  $p, q$ ):

$$p^\top (\lambda I_n - A) + q^\top (-\tau^{-1}) = 0, \quad (65)$$

$$q^\top (\lambda I_n + \tau^{-1}) = 0, \quad (66)$$

$$p^\top B_1 = 0. \quad (67)$$

Suponha  $(A, B_1)$  controlável e suponha, por contradição, que existe  $(p, q) \neq (0, 0)$  e  $\lambda$  satisfazendo (65)–(67). Caso 1:  $q = 0$ . Então (65) reduz a  $p^\top (\lambda I_n - A) = 0$  junto com (67)  $p^\top B_1 = 0$ . Mas isto significa que existe vetor esquerdo  $p \neq 0$  tal que  $p^\top A = \lambda p^\top$  e  $p^\top B_1 = 0$ . Pelo critério PBH para a planta, isto contradiz a controlabilidade de  $(A, B_1)$ . Portanto  $q \neq 0$ . Caso 2:  $q \neq 0$ . Da equação (66) temos

$$q^\top (\lambda I_n + \tau^{-1}) = 0 \implies \det(\lambda I_n + \tau^{-1}) = 0 \text{ ou } q \in \ker(\lambda I_n + \tau^{-1}). \quad (68)$$

Mas  $\tau^{-1}$  é diagonal com entradas  $1/\tau_i > 0$ , logo os valores que anulam  $\lambda I_n + \tau^{-1}$  são

$$\lambda \in \left\{ -\frac{1}{\tau_i} : i = 1, \dots, n \right\}. \quad (69)$$

Assim  $\lambda$  tem de ser igual a  $-1/\tau_{i_0}$  para alguns índices  $i_0$  onde  $q$  tem componente não nula. Da (65) obtemos

$$p^\top (\lambda I_n - A) = q^\top \tau^{-1}. \quad (70)$$

Multiplique a identidade acima (à direita) por  $A^k B_1$ , para  $k = 0, \dots, n-1$ . Usando (67)  $p^\top B_1 = 0$  e a recorrência  $\top_k = A^k B_1$ , obtemos, por indução,

$$p^\top A^k B_1 = 0, \quad \forall k = 0, \dots, n-1. \quad (71)$$

Mas se  $(A, B_1)$  for controlável, os vetores  $\{A^k B_1\}_{k=0}^{n-1}$  geram  $\mathbb{R}^n$ . Assim as equações  $p^\top A^k B_1 = 0$  para  $k = 0, \dots, n-1$  implicam  $p^\top = 0$ . Voltando a (65), com  $p^\top = 0$  resulta  $q^\top (-\tau^{-1}) = 0$  e, como  $\tau^{-1}$  é invertível, segue  $q^\top = 0$ . Assim  $p = q = 0$ , contradizendo a hipótese de ser vetores não-nulos. Portanto não existe  $(p, q) \neq (0, 0)$  que satisfaça (65)–(67), ou seja, o par aumentado é controlável. Portanto,

$$(A_{\text{aug}}^{(\text{open})}, B_{\text{aug}}^{(u)}) \text{ é controlável} \iff (A, B_1) \text{ é controlável}. \quad (72)$$

A inclusão dos estados dos sensores de primeira ordem (dinâmica  $\tau^{-1}$ ) não anula nem cria controlabilidade adicional, desse modo a controlabilidade do sistema aumentado é exatamente a controlabilidade da planta original. Segue o anexo C o trecho de código em *python* para verificar numericamente o posto da matriz de controlabilidade aumentada. Na implementação em Xcos, contudo, o único sinal disponível para multiplicação por K foi a saída do sensor dinâmico. O sensor foi modelado por um filtro de primeira ordem aplicado componente a componente; escrevendo  $z(t)$  o vetor de leituras do sensor, tem-se no domínio do tempo

$$\tau \dot{z}(t) + z(t) = \mathbf{x}(t), \quad (73)$$

onde  $\tau = \text{diag}(\tau_1, \dots, \tau_n) \succ 0$  contém as constantes de tempo dos filtros. A lei implementada em Xcos foi, portanto em (41), isto é, o ganho calculado para o caso ideal foi aplicado à saída filtrada  $z$  em vez do estado  $\mathbf{x}$ . Colocando (41) e (73) juntos com (1) obtém-se o sistema aumentado que rege a simulação em Xcos em

(56). Note que, por construção, a parte superior direita do bloco  $A_{\text{aug}}^{\text{impl}}$  contém  $-BK$ . A diferença estrutural entre  $A_{\text{cl}}^{\text{nom}} = A - BK$  e a matriz de malha fechada efetiva que atua no modelo aumentado impede, em geral, que a propriedade  $\sigma(A_{\text{cl}}^{\text{nom}}) \subset \{s : \Re(s) < 0\}$  (autovalores em meia-plano esquerdo) implique automaticamente  $\sigma(A_{\text{aug}}^{\text{impl}}) \subset \{s : \Re(s) < 0\}$ . Em outras palavras, um  $K$  estabilizante para o modelo nominal nem sempre é estabilizante quando o controle é aplicado sobre um vetor medido que sofre filtragem dinâmica. Para entender matematicamente a razão da possível instabilidade observada em Xcos, considerei o polinômio característico associado ao sistema aumentado. Chamando  $n_s$  o número de estados introduzidos pelos sensores (no nosso caso  $n_s = n$ ), o polinômio característico de  $A_{\text{aug}}^{\text{impl}}$  é dado por

$$\det(sI_{n+n_s} - A_{\text{aug}}^{\text{impl}}) = \det \begin{pmatrix} sI_n - A & BK \\ -\tau^{-1} & sI_{n_s} + \tau^{-1} \end{pmatrix}. \quad (74)$$

Não existe uma fatoração simples que separe este determinante em  $\det(sI_n - (A - BK))$  vezes outro fator: a presença dos blocos fora da diagonal acopla as duas subdinâmicas (planta e sensores) e faz com que os autovalores do sistema aumentado sejam soluções do determinante acima, que dependem de  $\tau$  de modo não trivial. Quando os valores das constantes  $\tau_i$  não são desprezíveis, as raízes deste polinômio podem deslocar-se para o semiplano direito mesmo que  $A - BK$  seja Hurwitz.

Uma das formas de ver esse acoplamento é recorrer à análise de singular-perturbação para o caso de  $\tau_i$  pequenos. Se todos os  $\tau_i = \varepsilon$  são tais que  $0 < \varepsilon \ll 1$ , pode-se escrever o sistema aumentado na forma de um sistema singularmente perturbado com variáveis rápidas  $z$  e variáveis lentas  $x$ . Escrevendo  $\varepsilon \dot{z} = -z + x$ , para  $\varepsilon \rightarrow 0^+$  tem-se o limite  $z \rightarrow x$  (camada de limite rápido), e então o comportamento reduzido aproxima o caso ideal. Contudo, quando  $\varepsilon$  é pequeno mas não negligenciável numericamente, os modos rápidos podem interagir com a ação de controle aplicada sobre  $z$  e introduzir polos não desejados; em particular, a inversão numérica de escalas temporais em integradores no simulador pode amplificar erros numéricos e tornar a simulação instável mesmo que o modelo reduzido seja estabilizante teoricamente. Do mesmo modo, se  $\tau_i$  for relativamente grande, o filtro atua como atraso efetivo e desloca a resposta em fase, reduzindo margens e favorecendo a instabilidade por excesso de ganho.

Os resultados experimentais obtidos — ganhos  $K$  viáveis segundo o Lema do Ganho Limitado e a sintetização  $\mathcal{H}_\infty$  em Python ([PYTHON-CONTROL...](#), 2021), mas comportamento instável em Xcos — coadunam com as duas causas principais descritas acima: primeiro, inconsistência entre hipótese de projeto (realimentação por  $\mathbf{x}$ ) e implementação (realimentação por  $z$ ); segundo, efeitos numéricos e de escala temporal introduzidos pelos filtros do sensor. Para desambiguar e diagnosticar, seguimos três verificações matemáticas simples e determinísticas. Em primeiro lugar, calculamos os autovalores de  $A_{\text{aug}}^{\text{impl}}$  e validamos que, para os valores  $\tau_i$  usados na simulação, existe pelo menos um autovalor com  $\Re(\lambda) > 0$ ; a presença desses autovalores explica diretamente a divergência observada na resposta em frequência. Em segundo lugar, verificamos a controlabilidade do par  $(A_{\text{aug}}, B_{\text{aug}})$ : quando o par aumentado deixa de ser estabilizável, ou seja, mesmo a inclusão dos estados dos sensores de primeira ordem não anula e nem cria crontrolabilidade adicional logo, não existe  $K$  capaz de tornar a malha fechada estável, independentemente do método de síntese. Em terceiro lugar, quando aplicamos a síntese  $H_\infty$  aplicamos o teste de Lyapunov no espaço aumentado, isto é, recuperamos  $K_{\text{aug}} = Y_{\text{aug}}X_{\text{aug}}^{-1}$ . Além do diagnóstico matemático direto, é importante registrar algumas causas práticas que frequentemente aparecem nessas comparações entre síntese numérica e implementação em simuladores e que justificam sempre uma checagem dupla: diferenças de sinal (por exemplo, sinais escalares vs vetoriais em Xcos), erros de montagem do bloco de ganho (transposição acidental de vetores), representação discreta do integrador no simulador, saturação ou limites aplicados involuntariamente a blocos, elementos de rotina de integração (método explícito com passo fixo inadequado para sistemas stiff), e problemas de precisão numérica no solver quando a LMI é mal condicionada. Todas essas fontes práticas devem ser descartadas antes de concluir que o problema é estrutural; no caso em análise, as verificações algébricas acima mostraram que a causa principal foi estrutural, ou seja, o ganho não considerou a dinâmica dos sensores.

A implicação direta para projeto é imediata e pode ser enunciada de forma operacional: se o controlador será implementado sobre medidas filtradas  $z$  (como no cenário real em Xcos), então a síntese do ganho deve ser feita sobre o modelo aumentado que inclui explicitamente as equações do sensor. Proceder à

síntese  $\mathcal{H}_\infty$  ou à síntese via Lema do Ganho Limitado sobre  $A_{\text{aug}}$  e  $B_{\text{aug}}$  garante que os efeitos dinâmicos do sensor serão considerados no critério de estabilidade e desempenho e, em geral, leva a ganhos  $K_{\text{aug}}$  que estabilizam o sistema implementado. Alternativamente, se for impraticável aumentar a ordem do controlador por razões de restrição computacional, recomenda-se projetar um observador (por exemplo um observador de Luenberger ou filtro de Kalman) conforme apresentamos no início deste trabalho que reconstrua  $\mathbf{x}$  a partir de  $z$  e aplicar o ganho de estado sobre a estimativa  $\hat{\mathbf{x}}$ . A síntese combinada (projeto do observador + projeto do ganho de estado) pode também ser formulada por LMIs acopladas, ou por abordagem separada quando se satisfaz o princípio da separação (casos com ruído de processo/medição adequadamente modelados).

Por fim, a sequência de acontecimentos observada — ganhos  $K$  válidos numericamente para o modelo nominal somado com a instabilidade prática na simulação com sensor dinâmico — é explicada de maneira unívoca pelo fato de que a hipótese de projeto (realimentação exata do estado) não coincide com a hipótese de implementação (realimentação a partir de estados filtrados). A solução consistente, confirmada por verificações algébricas e por solução da LMI correspondente ao sistema aumentado, é refazer a síntese incluindo a dinâmica do sensor ou, quando isso não for possível, projetar um observador que permita aplicar  $K$  sobre uma estimativa do estado. Por fim, recomenda-se sempre, após qualquer síntese numérica, verificar explicitamente os autovalores do sistema implementado, resolver a desigualdade de Lyapunov no sistema implementado e simular em malha fechada no mesmo ambiente de execução final (no caso, Xcos) para garantir que as hipóteses matemáticas foram preservadas na implementação.

## 4 CONCLUSÃO

Este trabalho apresentou uma análise aprofundada do projeto de controladores robustos do tipo  $\mathcal{H}_\infty$  para uma planta linear de quarta ordem na presença de perturbações e de sensores dinâmicos de primeira ordem. A investigação combinou síntese numérica (Lema do Ganho Limitado e síntese  $\mathcal{H}_\infty$  implementadas em Python) com verificação prática em ambiente de simulação (Xcos/Scilab). O contraste entre ganhos obtidos a partir do modelo nominal e o comportamento observado na simulação mostrou-se o fio condutor das conclusões técnicas: ganhos que estabilizam o modelo suposto não necessariamente garantem estabilidade quando aplicados a sinais medidos após filtragem sensorial.

Do ponto de vista teórico, demonstrou-se que a condição de estabilidade formulada pela inequação de Lyapunov pode ser transformada em uma LMI convexa por meio da mudança de variáveis  $X = P^{-1}$  e  $W = KP^{-1}$ . Esta formulação permite recuperar a lei de realimentação  $K = WX^{-1}$  e proporciona uma ferramenta numérica robusta para síntese de ganhos que assegurem estabilidade assintótica (ÅSTRÖM; MURRAY, 2008), desde que a hipótese de projeto (acesso direto aos estados) seja satisfeita. A inclusão explícita da dinâmica do sensor no modelo, conduzindo à construção do sistema aumentado, revelou ser fundamental: a análise algébrica do polinômio característico do sistema aumentado e a verificação direta dos autovalores de  $A_{\text{aug}}$  explicaram por que ganhos projetados para o caso ideal falharam quando implementados sobre leituras filtradas.

A investigação numérica e experimental permitiu identificar dois efeitos distintos, ambos com impacto prático. O primeiro é estrutural e decorre do fato de a implementação aplicar o ganho sobre a saída filtrada  $z$  em vez do estado  $x$ ; essa diferença altera a estrutura da matriz de malha fechada e pode introduzir modos instáveis que não aparecem no modelo nominal. O segundo é numérico e relaciona-se ao condicionamento e às escalas temporais: constantes de tempo de sensor muito pequenas ou muito grandes afetam o condicionamento da LMI e a integração num simulador, podendo induzir comportamentos rígidos ou sensíveis a

erros de discretização. Ambas as causas foram confirmadas por cálculo explícito de autovalores, por testes de estabilizabilidade do par aumentado e por solução da LMI aplicada ao sistema ampliado.

A partir desses resultados seguem recomendações práticas claras para projetos semelhantes. Quando o controlador for implementado sobre medidas filtradas, a síntese deve considerar o sistema aumentado que inclui as equações dos sensores; quando isso for impraticável por limitação computacional, deve-se projetar um observador que reconstrua o estado completo a partir das medidas disponíveis e aplicar o ganho sobre a estimativa. Antes da implementação, é imprescindível verificar numericamente os autovalores do sistema realmente implementado, resolver a inequação de Lyapunov para o modelo final e simular em tempo contínuo no mesmo ambiente de execução previsto para a implementação real. Além disso, cuidados de engenharia tais como escalonamento adequado, regularização de  $X$  (por exemplo  $X \succeq \varepsilon I$ ), escolha de solvers robustos e validação da modelagem dos blocos de integração no simulador reduzem a probabilidade de discrepâncias entre projeto e comportamento efetivo.

As limitações deste trabalho residem, principalmente, na hipótese de linearidade e no foco em controladores de realimentação de estado estática ([GREEN, 1995](#)). Extensões naturais incluem a síntese de controladores dinâmicos de ordem reduzida que incorporem explicitamente a dinâmica dos sensores, a análise de robustez frente a incertezas paramétricas mais amplas (por exemplo via  $\mu$ -synthesis), e a validação experimental em bancada para avaliar efeitos não lineares e ruídos reais. Uma investigação complementar recomendada é a comparação entre estratégias separadas (observador + controlador) e estratégias integradas (controlador dinâmico projetado diretamente para o sistema aumentado) em termos de desempenho e custo computacional.

Em síntese, a principal mensagem deste trabalho é que a robustez matemática obtida por meio de LMIs só se traduz em robustez operacional quando o modelo utilizado na síntese coincide de forma fiel com a estrutura implementada. Incluir a dinâmica dos sensores no modelo de projeto ou fornecer estimativas de

estado de qualidade é, portanto, requisito necessário para garantir que ganhos sintetizados teoricamente resultem em comportamento estável e satisfatório na prática.

## REFERÊNCIAS

- APS, M. **MOSEK Fusion and CLI User Manual.** [S.l.], 2020. Manual do solver MOSEK.
- BOYD, S.; GHAOUI, L. E.; FERON, E.; BALAKRISHNAN, V. **Linear Matrix Inequalities in System and Control Theory.** Philadelphia: SIAM, 1994.
- CONSORTIUM, S. **Scilab and Xcos Documentation.** [S.l.], 2021. Scilab/Xcos manuals.
- DIAMOND, S.; BOYD, S. Cvxpy: A python-embedded modeling language for convex optimization. In: **Proceedings of the 15th Python in Science Conference.** [S.l.: s.n.], 2016.
- DOYLE, J. C.; FRANCIS, B. A.; TANNENBAUM, A. R. **Feedback Control Theory.** New York: Macmillan, 1992.
- DOYLE, J. C.; GLOVER, K.; KARGONEKAR, P. P.; FRANCIS, B. A. State-space solutions to standard  $\mathcal{H}_2$  and  $\mathcal{H}_\infty$  control problems. **IEEE Transactions on Automatic Control**, Piscataway, NJ, v. 34, n. 8, p. 831–847, 1989.
- DULLERUD, G. E.; PAGANINI, F. **A Course in Robust Control Theory: A Convex Approach.** Cambridge: Cambridge University Press, 2000.
- FRANCIS, B. A. **A Course in  $\mathcal{H}_\infty$  Control Theory.** London: Springer, 1997.
- GEROMEL, J. C.; KOROGUI, R. H. **Controle linear de sistemas dinâmicos: teoria, ensaios práticos e exercícios.** São Paulo: Edgard Blücher, 2019. ISBN 978-85-212-1469-4.
- GRANT, M.; BOYD, S. **CVX: Matlab Software for Disciplined Convex Programming.** [S.l.], 2014. Acesso em: data de consulta.
- GREEN, M. **Linear Robust Control.** London: Prentice Hall, 1995.
- KALMAN, R. E. A new approach to linear filtering and prediction problems. **Journal of Basic Engineering**, v. 82, p. 35–45, 1960.
- KHALIL, H. K. **Nonlinear Systems.** 3. ed. Upper Saddle River, NJ: Prentice Hall, 2002.
- PYTHON-CONTROL: Control Systems Library for Python. 2021. <<https://python-control.readthedocs.io/>>. Acesso em: data de consulta.

VANDENBERGHE, L.; BOYD, S. Semidefinite programming. **SIAM Review**, v. 38, n. 1, p. 49–95, 1996.

ÅSTRÖM, K. J.; MURRAY, R. M. **Feedback Systems: An Introduction for Scientists and Engineers**. Princeton: Princeton University Press, 2008.

## ANEXO A – LMI: CONTROLADOR DE LYAPUNOV

```
1 import cvxpy as cp
2 import numpy as np
3
4 # Definição das matrizes do sistema
5 A = np.array([[ 0,  1,  0,  0],
6                 [ 0,  0,  1,  0],
7                 [ 0,  0,  0,  1],
8                 [-4, -3, -5, -2]])
9 B = np.array([[0],
10                  [0],
11                  [0],
12                  [1]])
13
14 n = A.shape[0] # número de estados (n = 4)
15 m = B.shape[1] # número de entradas (m = 1)
16
17 # Variáveis de decisão
18 P = cp.Variable((n, n), symmetric=True)
19 W = cp.Variable((m, n))
20
21 # Expressão do LMI de Lyapunov
22 LMI_expr = A.T @ P + P @ A + B.T @ W.T + W @ B
23
24 # Conjunto de restrições
25 constraints = [
26     P >> 1e-6 * np.eye(n),
27     # P > 0 (com margem para evitar problemas numéricos)
28     LMI_expr << -1e-6 * np.eye(n) # LMI estritamente negativa
29 ]
30
31 # Definição e resolução do problema de otimização (problema de viabilidade)
32 problem = cp.Problem(cp.Minimize(0), constraints)
33 problem.solve(solver=cp.SCS)
34
35 # Exibição dos resultados
36 print("Status do solver:", problem.status)
```

```
37 if problem.status in [cp.OPTIMAL, cp.OPTIMAL_INACCURATE]:  
38     print("\nMatriz P encontrada:")  
39     print(P.value)  
40     print("\nMatriz W encontrada:")  
41     print(W.value)  
42  
43     # Cálculo do ganho de realimentação  
44     K = W.value @ np.linalg.inv(P.value)  
45     print("\nGanho de realimentação K = W*P^{-1}:")  
46     print(K)  
47 else:  
48     print("Não foi possível encontrar uma solução viável.")
```

---

## ANEXO B – LMI: $H_\infty$ / SINTEZAÇÃO DE GANHO

```
1 import numpy as np
2 import cvxpy as cp
3
4 # Dados do sistema
5 A = np.array([[ -2.,   1.,   0.,   0.],
6               [ -1.,  -3.,   0.,   1.],
7               [  0.,   0.,  -4.,   2.],
8               [  0.,  -1.,  -1.,  -2.]])
9 B1 = np.array([[ 0.], [ 1.], [ 0.], [ 0.]])
10 B2 = np.array([[ 0.], [ 0.], [ 0.], [ 1.]])
11 C = np.array([[ 1.,   0.,   0.,   0.]])
12
13 n, m, nw, nz = A.shape[0], B1.shape[1], B2.shape[1], C.shape[0]
14 eps = 1e-6
15
16 # Variáveis de otimização
17 Q = cp.Variable((n, n), symmetric=True)
18 W = cp.Variable((m, n))
19 gamma = cp.Variable(nonneg=True)
20
21 # Construção do LMI
22 M11 = Q @ A.T + A @ Q + W.T @ B1.T + B1 @ W
23 M12 = B2; M13 = Q @ C.T
24 M21 = B2.T; M22 = -np.eye(nw); M23 = np.zeros((nw, nz))
25 M31 = C @ Q; M32 = np.zeros((nz, nw)); M33 = -gamma * np.eye(nz)
26
27 LMI = cp.bmat([[M11, M12, M13],
28                 [M21, M22, M23],
29                 [M31, M32, M33]])
30
31 # Restrições
32 constraints = [
33     LMI << -eps * np.eye(n + nw + nz),
34     Q >> eps * np.eye(n)
35 ]
36
```

```

37 # Problema e solução
38 prob = cp.Problem(cp.Minimize(gamma), constraints)
39 prob.solve(solver=cp.MOSEK, verbose=True)
40
41 # Verifica status
42 if prob.status not in ["optimal", "optimal_inaccurate"]:
43     raise RuntimeError(f"Solver falhou (status: {prob.status})")
44
45 # Recupera ganhos
46 Q_val = Q.value
47 K      = W.value @ np.linalg.inv(Q_val)
48
49 print(f"* = {gamma.value:.6f}\nK =\n{K}")
50
51 # Função para checar estabilidade (autovalores 2x2)
52 def checar_estabilidade(A, B2, K):
53     Af = A + B2 @ K
54     # seleciona subbloco 2x2 para característico
55     M = Af[1:3, 0:2]
56     a, b = 1, -np.trace(M)
57     c    = np.linalg.det(M)
58     delta = b**2 - 4*a*c
59     if delta >= 0:
60         roots = [(-b + np.sqrt(delta)) / 2, (-b - np.sqrt(delta)) / 2]
61     else:
62         real = -b / 2
63         imag = np.sqrt(-delta) / 2
64         roots = [complex(real, imag), complex(real, -imag)]
65     print("Autovalores:", roots)
66     print("Sistema estável?", all(rt.real < 0 for rt in roots))
67
68 checar_estabilidade(A, B2, K)

```

## ANEXO C – VERIFICAÇÃO DE CONTROLABILIDADE

```
1 import numpy as np
2 from numpy.linalg import matrix_rank
3
4 # Exemplo: defina A (n x n), B1 (n x m), Tau (n x n diagonal)
5 # Por exemplo, para seu caso n=4:
6 # A = np.array([[[-2., 1., 0., 0.],
7 # [-1., -3., 0., 1.], [0., 0., -4., 2.], [0., -1., -1., -2.]])
8 # B1 = np.array([[0.], [1.], [0.], [0.]])
9 # Tau = np.diag([tau1, tau2, tau3, tau4])
10
11 n = A.shape[0]
12 Aaug = np.block([[A, np.zeros((n,n))],
13                  [np.linalg.inv(Tau), -np.linalg.inv(Tau)]]])
14 Baug = np.vstack([B1, np.zeros((n, B1.shape[1]))])
15
16 # construir matriz de controlabilidade
17 Caug = Baug
18 for k in range(1, 2*n):
19     Caug = np.hstack([Caug, np.linalg.matrix_power(Aaug, k) @ Baug])
20
21 rankC = matrix_rank(Caug)
22 print("rank(C_aug) =", rankC, "expected", 2*n)
23 if rankC == 2*n:
24     print("Sistema aumentado controlável pelo canal u")
25 else:
26     print("Sistema aumentado NÃO controlável pelo canal u")
```