

ALOCAÇÃO DE POLOS EM REGIÕES DO PLANO COMPLEXO PARA SISTEMAS DISCRETOS VIA LMIs

Alexandre Nascimento, Jr.

29 de outubro de 2022

Sumário

Sumário	1
1 INTRODUÇÃO	3
2 REGIÃO DE DESEMPENHO GARANTIDO	5
2.1 Regiões de polos para sistemas contínuos	5
2.2 Regiões de polos para sistemas discretos	7
2.3 Aproximação das regiões do plano z via LMIs	9
2.4 Modelagem via espaço de estados	9
2.5 Condição de Liapunov	10
2.6 Regiões LMIs de interesse	11
3 ALGORITMO	13
3.1 Aproximação cônica	13
3.2 Aproximação elíptica	15
3.3 Aproximação poligonal	15
4 TESTES E SIMULAÇÕES	21
4.1 Modelagem via espaço de estados	21
4.2 Primeiro projeto	22
5 CONCLUSÃO	25

REFERÊNCIAS 27

1 Introdução

Notação:

- $\mathcal{V}(A) = A + A'$;
- $\bar{\mathcal{V}}(A) = A - A'$
- $\text{loc}(v_1, v_2)$ determina o ponto que a reta que passa por v_1 e v_2 cruza o eixo real;
- $\text{ang}(v_1, v_2)$ refere-se ao ângulo entre aquela reta e o eixo real.

2 Região de Desempenho Garantido

A alocação de polos é uma das principais ferramentas da teoria de controle, pois a partir desta, é possível projetar um sistema que seja estável e que tenha um bom desempenho Rosinová e Holič (2014). A operação de alocar polos de um sistema linear dentro de uma região específica é chamada \mathcal{D} -estabilidade Chilali e Gahinet (1996).

Entende-se por estável o sistema que, em termos de resposta a estímulos, possui uma convergência ao zero da resposta natural, restando apenas a reposta forçada (NISE, 2011). Assim, para um intervalo de tempo determinado, espera-se que o sistema apenas tenha dinâmica referente à entrada aplicada. Neste contexto, a estabilidade é o ponto de partida para projetos de compensadores.

A resposta ao impulso no domínio do tempo de um sistema linear e invariante no tempo tem as características da figura 1. O máximo da curva é o máximo sobressinal (*overshoot* em inglês), onde é o pico da saída, dado em porcentagem. O parâmetro t_s representa o tempo de acomodação, onde é o instante de tempo que o sinal entra em um intervalo definido e não sai (normalmente 2%).

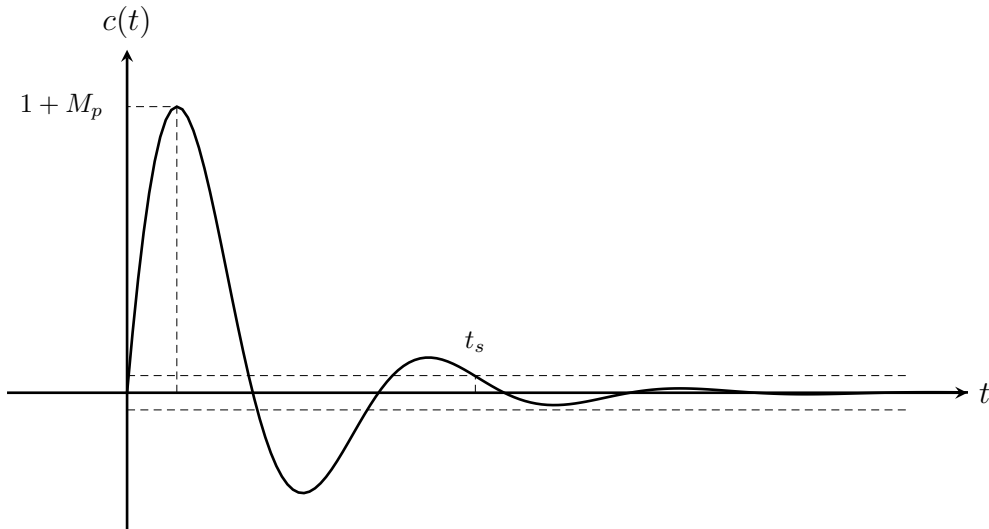


Figura 1 – Resposta ao impulso de um sistema de segunda ordem.

2.1 Regiões de polos para sistemas contínuos

Na \mathcal{D} -estabilidade, a região referente à estabilidade em sistema contínuos e invariantes no tempo é o semi-plano esquerdo do plano complexo. Dado um ponto genérico no plano s , representado por:

$$s = x + jy \quad (2.1)$$

este estará na região estável somente se a parte real de tal ponto estiver à esquerda do eixo imaginário, ou em números:

$$\operatorname{Re}(s) < 0 \implies x < 0 \quad (2.2)$$

Assim, um sistema com n polos é dito estável se todos os seus polos estão localizados à esquerda do eixo imaginário. A partir deste conceito, é possível definir estabilidade relativa: se um sistema é estável para um valor $\sigma < 0$, então aquele é dito estável relativo (ao valor de σ).

A figura 2a mostra um esboço da região comentada. À medida que o valor de σ aumenta em valor absoluto, mais a esquerda a reta limitante se encontra e menor o plano estável relativo se torna. Além disso, as equações de tais retas podem ser generalizadas via:

$$x = -|\sigma| \quad (2.3)$$

Outros parâmetros de desempenho importantes para projetos de compensadores são o fator de amortecimento ζ e a frequência natural não-amortecida ω_n . São caracterizados pela resposta de sistemas de segunda ordem à função degrau (NISE, 2011)(OGATA, 2011) e representam as oscilações não-amortecidas do modelo físico. Dado um par de polos conjugados via (2.1), é possível reescrevê-los em termos daqueles parâmetros:

$$s = -\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2} \quad (2.4)$$

com $\sigma = -\zeta\omega_n$. As regiões de \mathcal{D} -estabilidade referente a tais parâmetros são obtidos fixando um deles em (2.4) e variando o outro em um certo intervalo. Por esse motivo, (2.4) pode ser entendido como uma função de duas variáveis, dado como:

$$s(\zeta, \omega_n) = -\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2} \quad (2.5)$$

Com isso, é possível analisar as regiões geradas a partir de tais parâmetros. A região ζ -constante é obtida fixando-se um valor para ζ e variando-se o valor de ω_n . As características obtidas são representadas na figura 2b. Os ângulos formados entre as retas e o eixo imaginário têm valores absolutos $\beta = -\cotg(\arccos(\zeta))$ e diminuem à medida que o valor de ζ aumenta, tornando a região estreita.

Já as regiões ω_n -constante possuem as características esboçadas na figura 2c. Os raios das semi-circunferências formadas possuem valores iguais à ω_n e aumentam ou diminuem à medida que se varia tal parâmetro. Conforme abordado em (CHILALI; GAHINET, 1996), a intersecção das regiões comentadas formam a Região Ω de Desempenho garantido. Todos os polos dentro de tal região possuem um mínimo valor de σ , ζ e ω_n .

2.2 Regiões de polos para sistemas discretos

As regiões de \mathcal{D} -estabilidade para sistemas discretos são obtidas seguindo os mesmos métodos abordados nos contínuos, com a diferença de serem descritos no plano z . A transformada de um ponto do plano s para z é dada por:

$$z = \exp(s T_s) \quad (2.6)$$

onde T_s é o período de amostragem, parâmetro importante para sistemas discretos (KUO, 1980). Substituindo (2.4) em (2.6), chega-se a seguinte relação:

$$z = \exp \left(-\zeta \omega_n T_s \pm j \omega_n T_s \sqrt{1 - \zeta^2} \right) \quad (2.7)$$

A σ -estabilidade nos contínuos foi encontrada verificando a parte real dos polos. Utilizando-se da mesma ideia, ao analisar apenas a parte real de (2.7) (igualando a parte imaginária igual a zero), chega-se na seguinte relação:

$$z = \exp(-|\sigma| T_s) \quad (2.8)$$

onde $\sigma = -\zeta \omega_n$. Tal função descreve uma circunferência com raio $r = |\sigma|$ no plano complexo. Recordando (2.2) e (2.3), quando $\sigma = 0$ em (2.8), a circunferência gerada possui raio unitário. Dessa maneira, a região estável nos sistemas discretos é o interior de uma circunferência unitária. A figura 3a mostra esboços para vários valores de σ .

Assim como realizado nos sistemas contínuos, (2.7) pode ser enxergada em função de ζ e ω_n :

$$z(\zeta, \omega_n) = \exp \left(-\zeta \omega_n T_s \pm j \omega_n T_s \sqrt{1 - \zeta^2} \right) \quad (2.9)$$

e a partir desta, é possível descrever as regiões geradas a partir de tais parâmetros. A região ζ -constante possui o formato apresentado na figura 3a. Devido ao exponencial,

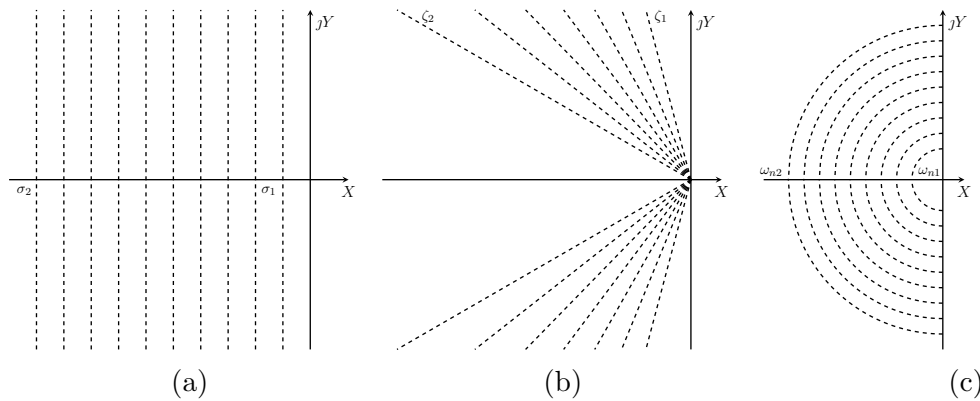


Figura 2 – Regiões de \mathcal{D} -estabilidade do plano s . Em (a) encontram-se retas verticais para vários valores de σ , sendo $|\sigma_2| > |\sigma_1|$. Em (b) encontram-se retas para vários valores de ζ , sendo $\zeta_2 > \zeta_1$. Em (c) encontram-se circunferências de raios $r = \omega_n$, sendo $\omega_{n2} > \omega_{n1}$.

as curvas geradas assemelham-se a cardioides, mas não o são, pois denominam-se espirais logarítmicas.

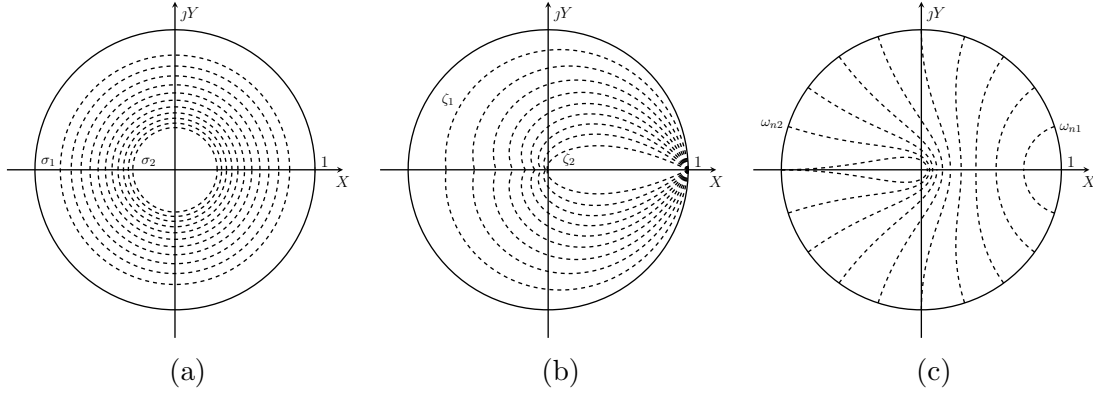


Figura 3 – Regiões de \mathcal{D} -estabilidade do plano z . Em (a) encontram-se circunferências com valores de raios crescentes, sendo $|\sigma_2| > |\sigma_1|$. Em (b) encontram regiões ζ -constantes com áreas decrescentes em relação à ζ , sendo $\zeta_2 > \zeta_1$. Em (c) encontram-se regiões semelhantes à cardioides que possuem áreas crescentes em relação à ω_n , sendo $\omega_2 > \omega_1$.

Ambos os ramos começam a ser desenhadas a partir de $(1, 0)$ (quando $\omega_n = 0 = \omega_{nmin}$), e se deslocam no sentido anti-horário, até cruzarem o eixo real primeira vez. À medida que o valor de ω_n aumenta, mais voltas o contorno dá. E a cada n meia-volta, o contorno cruza o eixo real pela n -ésima vez, conforme esboçado na figura 4. Como a espiral tende para dentro da região limitada pela primeira volta, somente a primeira volta é considerada no plano z .

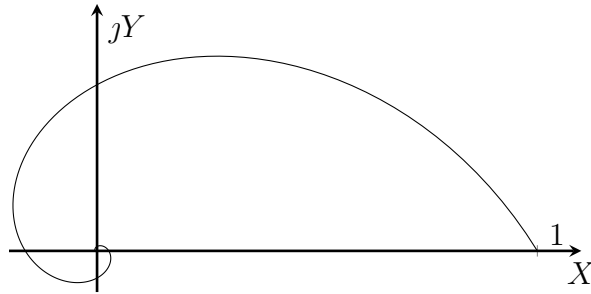


Figura 4 – Espiral logarítmica com 3 meia-voltas gerada a partir de (2.9) com $\zeta = 0.5$ constante.

O valor de ω_n no qual os ramos cruzam o eixo imaginário a cada n meia-volta é encontrado quando o argumento de (2.9) é igual à π (meia volta da espiral, em radianos), isto é:

$$\arg(z(\zeta, \omega_{nmax})) = \pi \implies \omega_{nmax} = \frac{\pi}{T_s \sqrt{1 - \zeta^2}} \quad (2.10)$$

Assim, ambos os ramos cruzam o eixo imaginário pela primeira vez quando $\omega_n = \omega_{nmax}$, para o respectivo valor fixado de ζ . Ainda, outra característica que pode ser

citada é a influência do ζ na região: quanto maior seu valor, menor a área da região ζ -constante equivalente, assim como ocorre com seu dual nos contínuos.

Em relação às regiões ω_n -constante, como o estudo de alocação de polos se restringe a sistemas de segunda ordem subamortecidos (NISE, 2011) (OGATA, 2011), os valores possíveis para a taxa de amortecimento está no intervalo $0 < \zeta < 1$. Dito isso, os ramos esboçados na figura 3c começam a ser desenhadas a partir da circunferência unitária, onde o ponto é dado por $z(\zeta_{min}, \omega_n)$ (com $\zeta = 0 = \zeta_{min}$) e vão em direção ao ponto $z(\zeta_{max}, \omega_n)$, com $\zeta_{max} = 1$.

Com tais pontos extremos, é possível aproximar as regiões do plano z utilizando Desigualdades Matriciais Lineares (LMIs, em inglês). Esse estudo será abordado na subseção a seguir.

2.3 Aproximação das regiões do plano z via LMIs

Em estudos anteriores, foram abordadas técnicas utilizando LMIs para mapear as regiões de \mathcal{D} -estabilidade no plano s . Tal feito foi realizado devido à convexidade de tais regiões, requisito para o uso de LMIs. Conforme visto na subseção 2.2, as regiões ζ -constante e ω_n -constante no plano z podem possuir características não-convexas, o que impossibilita o mapeamento exato via LMIs.

Estudos foram desenvolvidos para contornar a não-convexidade de algumas regiões do plano z , aproximando-os em regiões convexas. Em 2014, no artigo (ROSINOVÁ; HOLIČ, 2014), a autora mapeou a região ζ -constante utilizando a maior elipse ou circunferência inscrita possível. Mas foi em (ROSINOVÁ; HYPÍUSOVÁ, 2019) que foi desenvolvido um algoritmo que traz várias aproximações utilizando elipses, para aproveitar da melhor forma a área daquela região.

Já em (WISNIEWSKI et al., 2017) foi abordada uma aproximação cônica, utilizando-se apenas de quatro pontos e, conseqüentemente, dois setores cônicos. Apesar de simples, a ideia poderia facilmente ser estendida para n pontos, o que foi feito em (WISNIEWSKI; MADDALENA; GODOY, 2019). Ao aumentar a área a cada iteração, o algoritmo verifica a solução proposta.

E finalmente, utilizando-se das ideias anteriores, (CHIQUETO, 2021) trouxe aproximações cônica, elíptica e poligonal da região ω_n -constante.

2.4 Modelagem via espaço de estados

O espaço de estados é uma representação matemática desenvolvida na Teoria de Controle para estudar sistemas físicos, relacionando saídas, entradas e variáveis de estado internas, através do uso de Equações Diferenciais Ordinárias (EDOs).

Se o sistema a ser analisado for contínuo, linear e invariante no tempo, então as variáveis de estado deste podem ser representados através de vetores, enquanto que as EDOs correspondentes são descritas de forma matricial, como segue:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (2.11a)$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad (2.11b)$$

onde:

- $\dot{\mathbf{x}}, \mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$ e $\mathbf{y} \in \mathbb{R}^p$;
- $\mathbf{A} \in \mathbb{R}^{n \times n}$ é a matriz de estado;
- $\mathbf{B} \in \mathbb{R}^{n \times m}$ é a matriz de entrada;
- $\mathbf{C} \in \mathbb{R}^{p \times n}$ é a matriz de saída;
- $\mathbf{D} \in \mathbb{R}^{p \times m}$ é a matriz de transmissão direta.

Já a representação no espaço de estados de um sistema discreto, linear e invariante no tempo é:

$$\mathbf{x}^+ = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (2.12a)$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad (2.12b)$$

Em ambos os domínios, os polos do sistema são os autovalores da matriz de estado. Logo, um sistema representado via espaço de estados é dito estável se todos os polos da matriz \mathbf{A} estão no semi-plano esquerdo ou no interior de uma circunferência unitária.

2.5 Condição de Liapunov

Seja \mathcal{D} um subconjunto do plano contínuo, localizado no semi-plano esquerdo e um sistema representado como em (2.11) ou (2.12). Este conjunto é dito \mathcal{D} -estável se todos os autovalores da matriz de estado correspondente estiverem contidos em \mathcal{D} .

Caso \mathcal{D} for uma região convexa, é possível mapeá-la utilizando Desigualdes Matriciais Lineares (LMIs em inglês) (CHILALI; GAHINET, 1996), através da condição de Liapunov. A partir disso, uma matriz \mathbf{A} é estável se, e somente se, existir uma matriz simétrica \mathbf{P} , tal que:

$$\mathbf{A}\mathbf{P} + \mathbf{P}\mathbf{A}' \prec 0, \quad \mathbf{P} \succ 0 \quad (2.13)$$

onde o " $\mathbf{P} \succ 0$ " lê-se \mathbf{P} é positiva definida.

2.6 Regiões LMIs de interesse

Definidos LMIs, algumas regiões LMIs são apresentadas neste trabalho para as aproximações estudadas. Uma primeira região que pode ser citada é todo o interior de uma circunferência de raio r , dado no plano z :

$$\begin{bmatrix} -rP & * \\ PA + Z'B & -rP \end{bmatrix} \prec 0 \quad (2.14)$$

com $r = \exp(-|\sigma|T_s)$. Outra região de interesse é um setor cônico com centro em α e ângulo θ . Tal plano é simétrico em relação ao seu centro. Portando, o mapeado LMI leva em consideração tal simetria. Contudo, através de manipulações algébricas, a região LMI de um setor cônico voltado para a esquerda é definida como:

$$\begin{bmatrix} \sin(\theta)(\mathcal{V}(AP) + \mathcal{V}(BZ) - 2\alpha P) & * \\ \cos(\theta)(\bar{\mathcal{V}}(PA') + \bar{\mathcal{V}}(Z'B')) & \sin(\theta)(\mathcal{V}(AP) + \mathcal{V}(BZ) - 2\alpha P) \end{bmatrix} \prec 0 \quad (2.15)$$

com seu lado direito dado como:

$$\begin{bmatrix} \sin(\theta)(2\alpha P - \mathcal{V}(AP) - \mathcal{V}(BZ)) & * \\ \cos(\theta)(\bar{\mathcal{V}}(PA') + \bar{\mathcal{V}}(Z'B')) & \sin(\theta)(2\alpha P - \mathcal{V}(AP) - \mathcal{V}(BZ)) \end{bmatrix} \prec 0 \quad (2.16)$$

Para impedir a simetria do setor cônico direito, é definido uma reta limitante em α , dado por:

$$AP + BZ + Z'B' + PA' - 2\alpha P \succ 0 \quad (2.17)$$

onde considera a área a direita de tal reta. Isso é necessário apenas para o setor cônico direito, pois o voltado para a esquerda é limitado por (2.14), como pode ser observado na figura 3. Uma última região LMI é apresentada: uma elipse, que possui a seguinte equação (CHIQUETO, 2021):

$$\mathbb{E} : \frac{(x-1)^2}{\left(1 - \exp\left(-\frac{2\pi}{N_y}\right)\right)^2} + \frac{\left(\left(1 - \exp\left(-\frac{2\pi}{N_y}\right)\right)^2 - \left(\cos\left(\frac{2\pi}{N_y}\right) - 1\right)^2\right)y^2}{\left(1 - \exp\left(-\frac{2\pi}{N_y}\right)\right)^2 \sin\left(\frac{2\pi}{N_y}\right)} = 1 \quad (2.18)$$

onde N_y é relação entre a frequência de amostragem e a frequência natural não-amortecida, dada por:

$$N_y = \frac{\omega_s}{\omega_n} = \frac{2\pi}{\omega_n T_s} \quad (2.19)$$

A LMI correspondente é dada por:

$$\begin{bmatrix} -P & * \\ -\frac{1}{a}P + \frac{1}{2}\left(\frac{1}{a} + \frac{1}{b}\right)AP + \frac{1}{2}\left(\frac{1}{a} - \frac{1}{b}\right)PA' & -P \end{bmatrix} \prec 0 \quad (2.20)$$

onde:

$$a = \left(1 - \exp\left(\frac{-2\pi}{N_y}\right)\right) \quad (2.21a)$$

$$b = a \operatorname{sen}\left(\frac{2\pi}{N_y}\right) \quad (2.21b)$$

3 Algoritmo

O algoritmo apresentado neste trabalho é um compilado de algoritmos desenvolvidos anteriormente, utilizando-se das aproximações cônica, elíptica e poligonal das regiões de \mathcal{D} -estabilidade do plano z . O objetivo deste trabalho é desenvolver em *software* tais algoritmos e, ao informar parâmetros de projeto, determinar se é possível implementar um compensador que respeite os requisitos.

Para tal, o algoritmo pode ser dividido em três partes, uma para cada aproximação, sendo a aproximação desejada escolhida via chamada da função. O *software* utilizado foi o MATLAB, juntamente com o interpretador de LMIs YALMIP em conjunto com o solucionador numérico MOSEK.

3.1 Aproximação cônica

Para o mapeamento cônico das curvas ζ -constante e ω_n -constante, são utilizados os setores cônicos determinados via (2.16) e (2.15), e retas verticais como apresentado em 2.17.

Para a primeira curva, a ideia consiste em utilizar os pontos extremos calculados na seção 2.2, onde serão os centros dos setores cônicos. Os ângulos, medidos no sentido anti-horário, são determinados a partir de um terceiro ponto, conforme a figura 5a. A escolha do ponto V é feita de maneira que a área do triângulo $\widehat{VV_oV_i}$ seja a maior possível. Um algoritmo linear foi usado para encontrar este ponto.

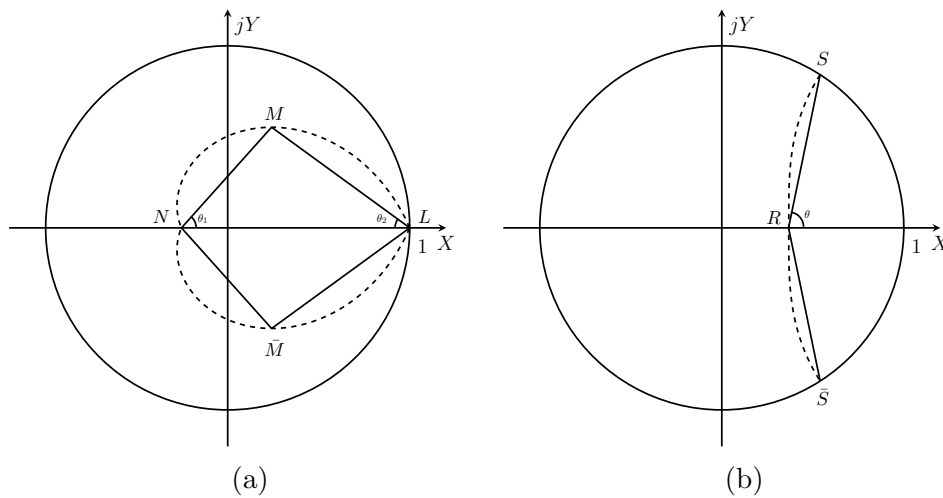


Figura 5 – Esboços da aproximação cônica das regiões ζ e ω_n -constantes.

Uma que vez conhecidas tais informações, é possível aplicar o algoritmo 1. Um setor cônico voltado para a direita, com centro em V_i e ângulo θ_1 , e outro voltado para a esquerda, com centro em V_o e ângulo θ_2 , são aplicados para a aproximação inicial. Além

Algoritmo 1 Aproximação cônica da taxa de amortecimento

Entrada: σ, ζ, T_s **Saída:** K

- 1: $V_o \leftarrow z(\zeta, \omega_{nmin})$
 - 2: $V_i \leftarrow z(\zeta, \omega_{nmax})$
 - 3: $V \leftarrow z(\zeta, \omega_n)$, onde a área do triângulo formado é a maior possível
 - 4: $F \leftarrow P \succ 0$
 - 5: $F \leftarrow F \cap (2.14)$ com $r = \exp(-|\sigma|T_s)$ ▷ Taxa de amortecimento
 - 6: $F \leftarrow F \cap (2.15)$ com $\alpha = V_o$ e $\theta = \text{ang}(V, V_i)$ ▷ Setor cônico esquerdo
 - 7: $F \leftarrow F \cap (2.16)$ com $\alpha = V_i$ e $\theta = \text{ang}(V, V_o)$ ▷ Setor cônico direito
 - 8: $F \leftarrow F \cap (2.17)$ com $\alpha = V_i$ ▷ Reta vertical
 - 9: Verificar se o problema é factível
 - 10: $K \leftarrow ZP^{-1}$
-

disso, para limitar a simetria do setor cônico com centro em V_i , uma reta que passa por este ponto é aplicada.

A região de \mathcal{D} -estabilidade resultante é a intersecção das regiões descritas. Ao ser unida com a restrição da taxa de decaimento, o setor cônico com centro em V_o é limitado por esta região. Após finalizado, o algoritmo determina a factibilidade da solução encontrada e retorna a matriz K que estabiliza o sistema com os parâmetros de projeto informados.

Em relação à região ω_n -constante, a mesma ideia é aplicada (CHIQUETO, 2021). Contudo, neste caso, somente um setor cônico com centro em N_i , que é limitado pela direita por uma reta que passa neste ponto são usados. Os pontos N_o e N_i são determinados via (2.9), com $\zeta = \zeta_{min}$ e $\zeta = \zeta_{max}$, respectivamente. Além disso, o ângulo θ é determinado através de $\text{ang}(N_o, N_i)$.

Após determinadas essas informações, é possível utilizar o algoritmo 2. Um detalhe que é facilmente observado é a rápida perda de convexidade da curva N_y . Logo, caso a constante N_y seja menor que 4.86 (CHIQUETO, 2021), o algoritmo retorna um alerta devido informando à falta de convexidade. Assim, para fins práticos, a pouca e a falta de convexidade de tais curvas não foram tratadas.

Algoritmo 2 Aproximação cônica da curva N_y

Entrada: σ, ω_n **Saída:** K

- 1: $N_o \leftarrow z(\zeta_{min}, \omega_n)$
 - 2: $N_i \leftarrow z(\zeta_{max}, \omega_n)$
 - 3: $F \leftarrow P \succ 0$
 - 4: $F \leftarrow F \cap (2.14)$ com $r = \exp(-|\sigma|T_s)$ ▷ Taxa de amortecimento
 - 5: $F \leftarrow F \cap (2.16)$ com $\alpha = N_i$ e $\theta = \text{ang}(N_i, N_o)$ ▷ Setor cônico direito
 - 6: $F \leftarrow F \cap (2.17)$ com $\alpha = N_i$ ▷ Reta vertical
 - 7: Verificar se o problema é factível
 - 8: $K \leftarrow ZP^{-1}$
-

3.2 Aproximação elíptica

Para a aproximação elíptica, apenas a região ω_n -constante foi aproximada. A ideia consiste em encontrar a maior elipse inscrita, a fim de aproveitar melhor a área. A figura 6 mostra um esboço da ideia descrita. Para tal, é preciso verificar se o valor escolhido para ω_n e T_s resultem em uma área convexa (CHIQUETO, 2021), através da constante N_y . Caso os parâmetros informados atendam às restrições, o algoritmo 3 pode ser aplicado.

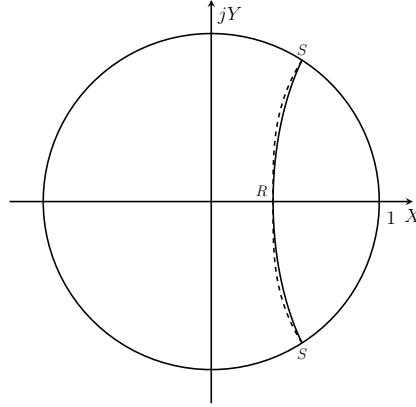


Figura 6 – Aproximação elíptica da região ω_n -constante.

Algoritmo 3 Aproximação elíptica da curva N_y

Entrada: σ, T_s, N_y

Saída: K

- 1: $F \leftarrow P \succ 0$
 - 2: $F \leftarrow F \cap (2.14)$, com $r = \exp(-|\sigma|T_s)$ ▷ Taxa de amortecimento
 - 3: $F \leftarrow (2.20)$, com $a = (2.21a)$ e $b = (2.21b)$ ▷ Elipse
 - 4: Verificar se o problema é factível
 - 5: $K \leftarrow ZP^{-1}$
-

3.3 Aproximação poligonal

A aproximação poligonal consiste na ideia de aproximar as regiões de interesse em um polígono com o maior número de lados possível. Para isto, o algoritmo irá partir de uma aproximação cônica simples. A partir daí, entre os dois pontos usados para definir o setor, um ponto intermediário é calculado e dois novos setores cônicos são definidos. Sob a ótica do número de lados, a cada iteração, um novo lado é acrescentado e, conseqüentemente, a área é incrementada. Em um número grande de iterações, a região aproximada tende a área total.

Para a região ζ -constante, um setor cônico voltado para esquerda e centro em V_o é usado como aproximação inicial (WISNIEWSKI; MADDALENA; GODOY, 2019). Contudo, devido à cúspide daquela, uma reta em V_i é usada para eliminar tal convexidade. Dito isso, surge a necessidade de calcular o ponto \bar{V} , localizado entre os pontos máximo e mínimo, onde possui a mesma parte real que V_i , conforme a figura 7a.

Algoritmo 4 Aproximação poligonal da região ζ -constante

Entrada: σ, ζ, T_s **Saída:** K

```
1:  $l \leftarrow 0$ 
2:  $V_o \leftarrow z(\zeta, \omega_{nmin}, T_s)$ 
3:  $V_i \leftarrow z(\zeta, \omega_{nmax}, T_s)$ 
4:  $V \leftarrow z(\zeta, \omega_n, T_s)$  tal que  $\text{Re}(V) = V_i$ 
5:  $\text{pontos1} \leftarrow [0 \ \omega_{ne}]$ 
6:  $\text{pontos2} \leftarrow \text{pontos1}$ 
7:  $\text{vec1} \leftarrow [V_o \ V]$ 
8:  $\text{vec2} \leftarrow \text{vec1}$ 
9:  $F \leftarrow P \succ 0$ 
10:  $F \leftarrow (2.14)$  com  $r = \exp(-|\sigma|T_s)$  ▷ Taxa de amortecimento
11:  $F \leftarrow F \cap (2.15)$ , com  $\alpha = V_o$  e  $\theta = \text{ang}(V_o, V)$  ▷ Voltado para a esquerda
12:  $F \leftarrow F \cap (2.17)$ , com  $\alpha = V_i$  ▷ Reta vertical
13: Verificar se o problema é factível
14: enquanto Problema for infactível faça
15:   se  $l < \text{número de elementos em } \text{vec1} - 1$  então
16:      $l \leftarrow l + 1$ 
17:   senão
18:      $l \leftarrow 1$ 
19:      $\text{vec1} \leftarrow \text{vec2}$ 
20:      $\text{pontos1} \leftarrow \text{pontos2}$ 
21:   fim se
22:    $F \leftarrow \emptyset$  ▷ Descarta as restrições anteriores
23:    $F \leftarrow P \succ 0$ 
24:    $F \leftarrow (2.14)$  com  $r = \exp(-|\sigma|T_s)$  ▷ Taxa de amortecimento
25:    $\text{ponto}_{new1} \leftarrow (\text{pontos1}(l) + \text{pontos1}(l+1))/2$ 
26:    $V_{new1} \leftarrow z(\zeta, \text{ponto}_{new1}, T_s)$ 
27:    $\text{pontos2} \leftarrow [\text{pontos2} \ \text{ponto}_{new1}]$ 
28:   Ordena de forma decrescente  $\text{pontos2}$ 
29:    $\text{vec2} \leftarrow [\text{vec2} \ V_{new1}]$ 
30:   Ordena de forma decrescente  $\text{vec2}$ 
31:    $F \leftarrow F \cap (2.17)$ , com  $\alpha = V_i$  ▷ Reta vertical
32:   para  $m = 1$  até número de elementos de  $\text{vec1} - 1$  faça
33:      $u_1 \leftarrow \text{loc}(\text{vec2}(m), \text{vec2}(m+1))$ 
34:     se  $u_1 < 0$  então
35:        $F \leftarrow F \cap (2.16)$ , com  $\alpha = u_1$  e  $\theta = \text{ang}(\text{vec2}(m+1), u_1)$  ▷ Voltado para a direita
36:     senão
37:        $F \leftarrow F \cap (2.15)$ , com  $\alpha = u_1$  e  $\theta = \text{ang}(\text{vec2}(m+1), u_1)$  ▷ Voltado para a esquerda
38:     fim se
39:   fim para
40:   Verificar se o problema é factível
41: fim enquanto
42:  $K \leftarrow ZP^{-1}$ 
```

Para tal, utiliza-se o cálculo numérico para encontrar uma solução aproximada. Com tal ponto calculado e a não convexidade da cúspide tratada, é possível utilizar o algoritmo 4. Os vértices iniciais V_o e V_i do ramo são guardados em um vetor de vértices. Em paralelo a isso, os valores de ω_n que geram tais pontos também são armazenados. A cada iteração, o algoritmo calcula o ponto médio entre dois vértices consecutivos em cada vetor, já que a realização do primeiro depende do segundo.

Contudo, antes do algoritmo voltar para o início dos vetores, é preciso percorrer todos

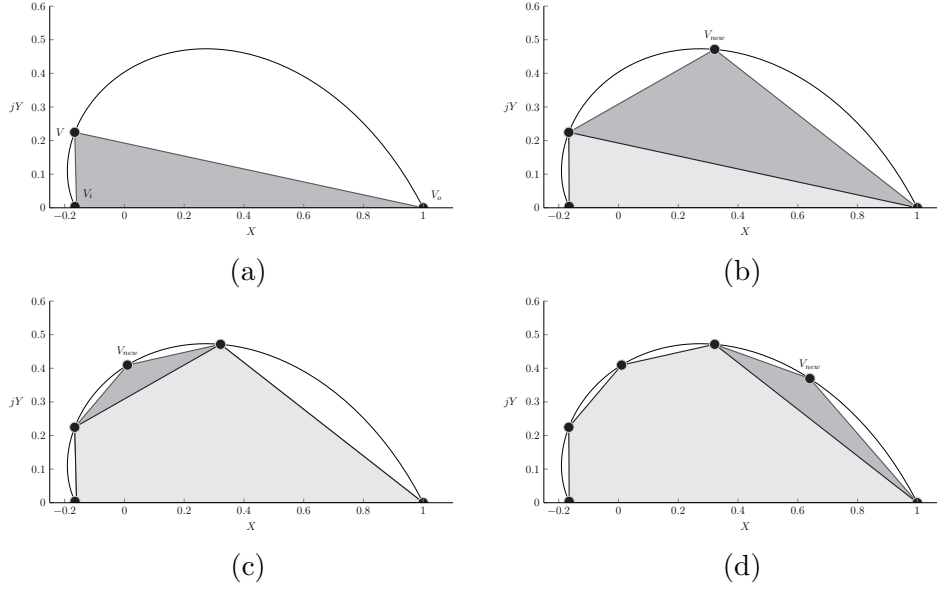


Figura 7 – As primeiras quatro aproximações poligonais da curva ζ -constante.

os pontos do vetor da iteração anterior¹. Para isso, cópias dos vetores de vértices e de pontos são inicializados, a fim de controlarem tal fluxo. Assim, quando o algoritmo terminar de percorrer o “vetor anterior”, tal conjunto é atualizado com os novos pontos e vértices calculados ao final deste processo.

Em relação ao cálculo dos pontos intermediários, a inclinação da reta que passa pelos pontos extremos locais pode ser positiva ou negativa. O uso da função loc se torna essencial, pois caso o ponto resultante for menor que zero, a reta que passa pelos pontos possui inclinação positiva, e negativa caso contrário (WISNIEWSKI; MADDALENA; GODOY, 2019). Assim, os setores cônicos gerados seguem a orientação desta, com centro naquele ponto calculado.

A figura 7 mostra as quatro primeira iterações do algoritmo 4. É possível observar o sentido horário do fluxo para o cálculo dos vértices intermediários. Também é notória a rápida abrangência da região ζ -constante.

Para a região ω_n -constante, a ideia é similar. A aproximação inicial utiliza-se de apenas um setor cônico voltado para a direita com centro em $N_i = z(\zeta_{max}, \omega_n)$. O ângulo θ em (2.16) é definido a partir do ângulo entre a reta $\overline{N_o N_i}$ e o eixo real, onde $N_o = z(\zeta_{min}, \omega_n)$.

Caso esta região não seja factível, basta calcular o ponto intermediário entre N_o e N_i e definir dois novos setores cônicos, a fim de aumentar a área. Novamente, para o novo ponto calculado, é utilizado a função loc para determinar o centro do setor cônico correspondente e, em seguida, usa-se ang para determinar seu ângulo. Assim, a cada iteração, o algoritmo adiciona dois novos setores cônicos e os intersecta com as regiões previamente definidas. A figura 8 mostra as quatro primeiras iterações do algoritmo 5.

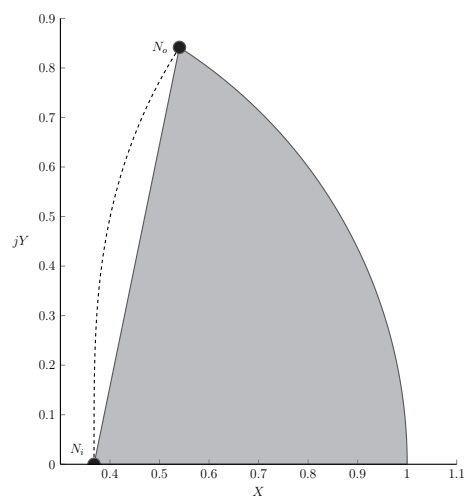
¹ Como os vetores são atualizados com os novos elementos, o tamanho daqueles aumenta $((n-1)$ pontos são adicionados a cada varredura, com n a quantidade de pontos do vetor) durante a iteração. Assim, o execução termina antes de atingir o final do vetor.

Algoritmo 5 Aproximação poligonal da região ω_n -constante

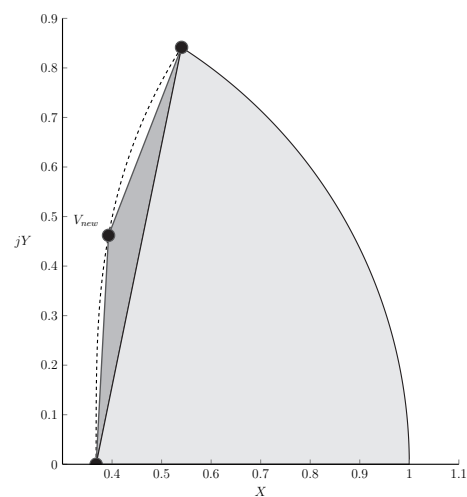
Entrada: σ, ω_n, T_s **Saída:** K

```
1:  $l \leftarrow 0$ 
2:  $N_o \leftarrow z(\zeta_{min}, \omega_n, T_s)$ 
3:  $N_i \leftarrow z(\zeta_{max}, \omega_n, T_s)$ 
4:  $pontos3 \leftarrow [0 \ 1]$ 
5:  $pontos4 \leftarrow pontos3$ 
6:  $vec3 \leftarrow [N_i \ N_o]$ 
7:  $vec4 \leftarrow vec3$ 
8:  $F \leftarrow P \succ 0$ 
9:  $F \leftarrow (2.14)$  com  $r = \exp(-|\sigma|T_s)$  ▷ Taxa de amortecimento
10:  $F \leftarrow F \cap (2.16)$ , com  $\alpha = N_i$  e  $\theta = \text{ang}(N_o, N_i)$  ▷ Voltado para a direita
11:  $F \leftarrow F \cap (2.17)$ , com  $\alpha = N_i$  ▷ Reta vertical
12: Verificar se o problema é factível
13: enquanto Problema for infactível faça
14:   se  $l < \text{número de elementos em } vec3 - 1$  então
15:      $l \leftarrow l + 1$ 
16:   senão
17:      $l \leftarrow 1$ 
18:      $vec3 \leftarrow vec4$ 
19:      $pontos1 \leftarrow pontos2$ 
20:   fim se
21:    $F \leftarrow \emptyset$  ▷ Descarta as restrições anteriores
22:    $F \leftarrow P \succ 0$ 
23:    $F \leftarrow (2.14)$  com  $r = \exp(-|\sigma|T_s)$  ▷ Taxa de amortecimento
24:    $ponto_{new} \leftarrow (pontos3(l) + pontos3(l+1))/2$ 
25:    $V_{new2} \leftarrow z(ponto_{new2}, \omega_n, T_s)$ 
26:    $pontos4 \leftarrow [pontos4 \ ponto_{new2}]$ 
27:   Ordena de forma decrescente  $pontos4$ 
28:    $vec4 \leftarrow [vec4 \ V_{new2}]$ 
29:   Ordena de forma decrescente  $vec4$ 
30:    $F \leftarrow F \cap (2.17)$ , com  $\alpha = V_i$  ▷ Reta vertical
31:   para  $m = 1$  até número de elementos de  $vec3 - 1$  faça
32:      $u_2 \leftarrow \text{loc}(vec4(m), vec4(m+1))$ 
33:      $F \leftarrow F \cap (2.16)$ , com  $\alpha = u_2$  e  $\theta = \text{ang}(vec4(m), u_2)$ 
34:   fim para
35:   Verificar se o problema é factível
36: fim enquanto
37:  $K \leftarrow ZP^{-1}$ 
```

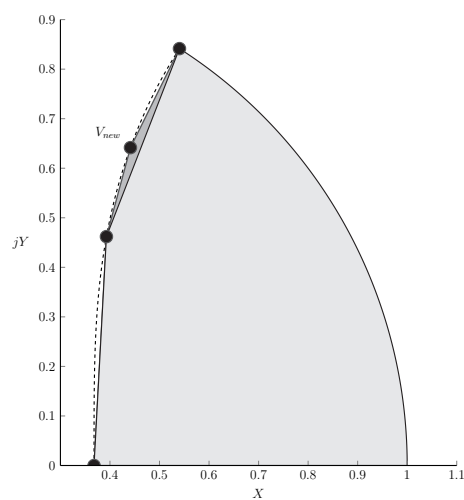
Assim como ocorre com a aproximação poligonal da região ζ -constante, há uma rápida cobertura da região. Com poucas iterações, a área é quase totalmente coberta.



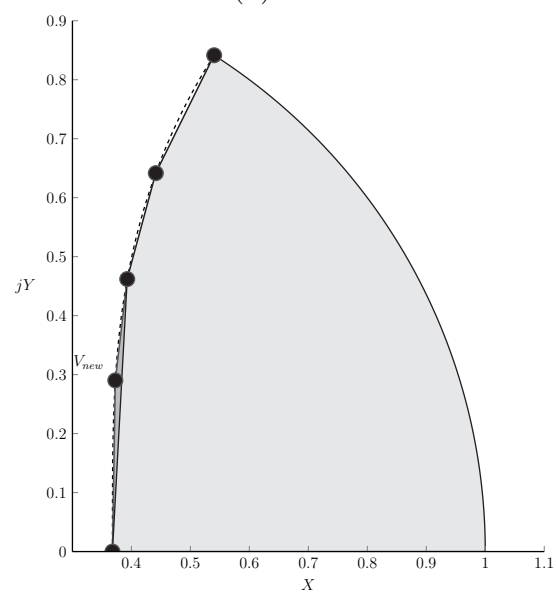
(a)



(b)



(c)



(d)

Figura 8 – As primeiras quatro aproximações poligonais da curva ω_n -constante.

4 Testes e Simulações

Para ilustrar o funcionamento dos algoritmos, foi proposta uma planta hidráulica composta por um sistema de tanques comunicantes. A figura 9 mostra um esboço de tal sistema. O tanque com capacitância C_1 é interligado com um de capacitância C_2 . Aquele é alimentado por uma vazão q e é drenado por uma vazão q_1 . Tal grandeza é controlada por um registro, que pode ser enxergado como um resistor de resistência R_1 .

Ainda, devido à ligação, a vazão de saída do primeiro tanque é a entrada do segundo. Este é drenado por uma vazão q_2 , onde é controlado por um registro R_2 . A variável controlada é a diferença $h_1 - h_2$.

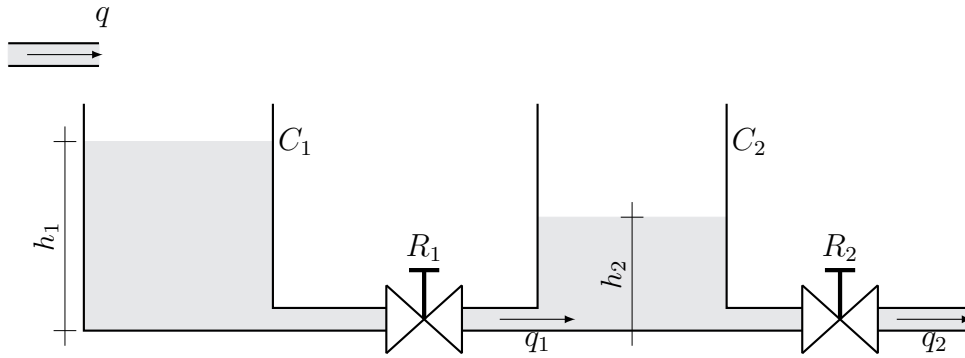


Figura 9 – Tanques comunicantes.

4.1 Modelagem via espaço de estados

Para a representação via espaço de estados, define-se as variáveis de estado $x_1 = q_1$ e $x_2 = q_2$. A partir das relações entre capacitância e vazão, chega-se a seguinte representação no espaço de estados:

$$\dot{\mathbf{x}} = \begin{bmatrix} -(R_1 C_{eq})^{-1} & (R_1 C_2)^{-1} \\ (R_2 C_2)^{-1} & -(R_2 C_2)^{-1} \end{bmatrix} \mathbf{x} + \begin{bmatrix} (R_1 C_1)^{-1} \\ 0 \end{bmatrix} \mathbf{u} \quad (4.1a)$$

$$\mathbf{y} = \begin{bmatrix} R_1 & 0 \end{bmatrix} \mathbf{x} \quad (4.1b)$$

onde $C_{eq} = C_1 C_2 / (C_1 + C_2)$. Para discretizar o sistema, é preciso de um valor para o período de amostragem T_s . A transformada usada será a bilinear de Tustin, dada por:

$$s = \frac{2}{T_s} \frac{z-1}{z+1} \quad (4.2)$$

onde o semi-plano esquerdo dos contínuos é mapeado no círculo unitário dos discretos.

4.2 Primeiro projeto

Com posse do espaço de estados, é possível sintetizar uma matriz de ganho K que possa estabilizar o sistema. Antes, é necessário atribuir valores para a planta. Em um primeiro projeto, serão escolhidas arbitrariamente tais valores, como segue:

- $C_1 = C_2 = 5$;
- $R_1 = R_2 = 1$;
- $T_s = 10$ s.

Assim, a representação via espaço de estados da planta nos contínuos é:

$$\dot{\mathbf{x}} = \begin{bmatrix} -0.4 & -0.2 \\ 0.2 & -0.2 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0.2 \\ 0 \end{bmatrix} \mathbf{u} \quad (4.3a)$$

$$\mathbf{y} = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x} \quad (4.3b)$$

Através do função `c2d` disponibilizada no MATLAB, a transformação bilinear é realizada, resultando em:

$$\dot{\mathbf{x}} = \begin{bmatrix} -0.4286 & -0.2857 \\ 0.2857 & -0.1429 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0.5714 \\ 0.2857 \end{bmatrix} \mathbf{u} \quad (4.4a)$$

$$\mathbf{y} = \begin{bmatrix} 0.2857 & -0.1426 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0.2857 \end{bmatrix} \mathbf{u} \quad (4.4b)$$

A característica notória obtida é a presença da matriz D no sistema discretizado. É um consequência da transformação bilinear: surgimento de transmissão direta. Com posse das matrizes obtidas na representação via espaço de estados, é possível escolher os parâmetros de projeto:

- $t_s = 50$ s;
- $\zeta = 0.5 \implies M_p \leq 0.16$;
- $\omega_n = 0.5$ rad/s.

Neste caso, o raio da circunferência relativo à estabilidade possui valor igual à 0.4493. Ainda, a constante N_y possui o valor de 6.2832, acima do recomendado. Além disso, a maior frequência natural não-amortecida de malha aberta do sistema discretizado possui o valor igual a 0.3780 rad/s.

Ao executar o algoritmo utilizando a aproximação cônica, a solução proposta é infactível. Ao checar a solução, apenas houve uma infactibilidade em relação à taxa de amortecimento. Tal fenômeno é comum em solucionadores numéricos, uma vez que podem admitir uma certa infactibilidade. A figura 10a mostra o diagrama de polos do

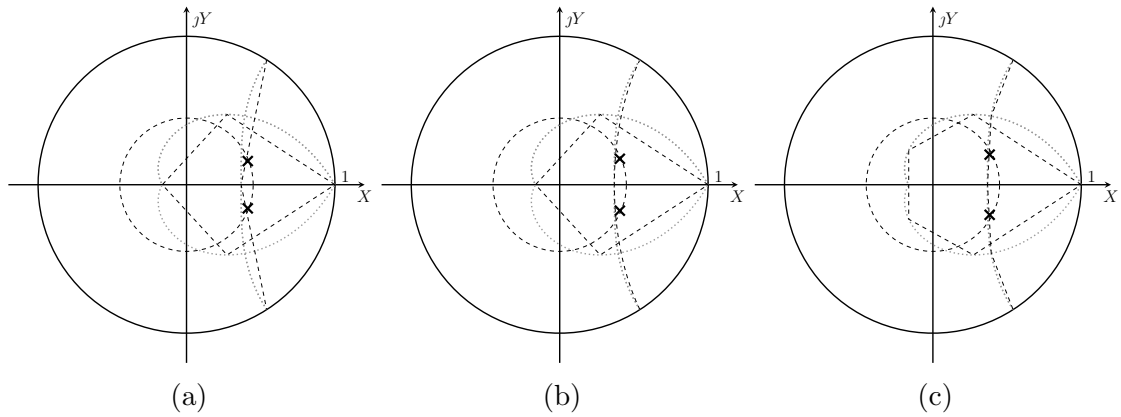


Figura 10

sistema compensado. É possível notar que o sistema é estável, mesmo com a negativa do algoritmo.

O valor da matriz K que estabiliza o sistema é:

$$K = [2.5358 \quad -0.1842] \quad (4.5)$$

A resposta ao impulso para o sistema em malha fechada está representada na figura 11. O máximo sobressinal é de aproximadamente 5,21%, bem abaixo do requisito de projeto. Como os parâmetros de projeto foram escolhidos arbitrariamente, o tempo de acomodação exigido é maior do que o sistema em malha aberta. Mesmo assim, o algoritmo conseguiu respeitar o requisito.

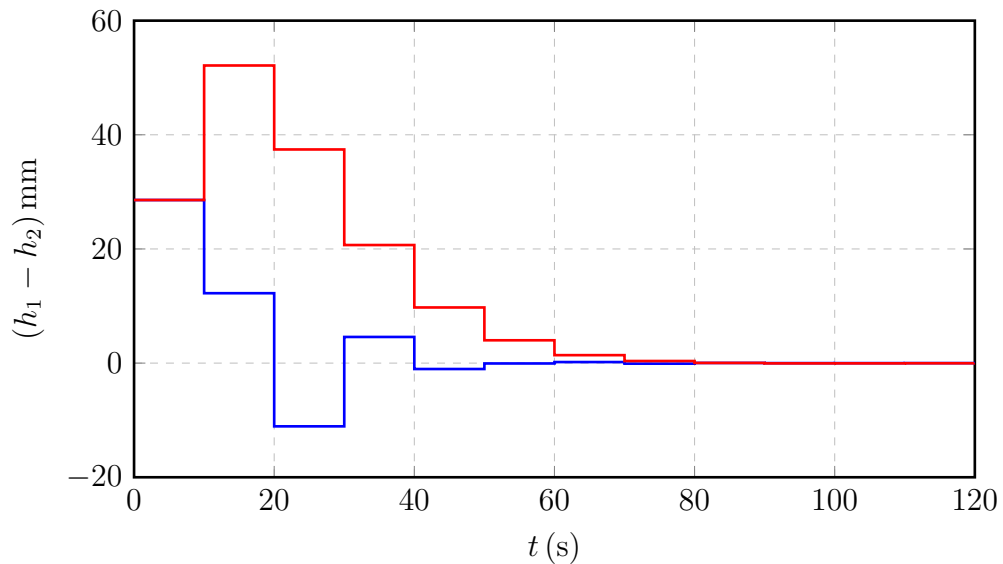


Figura 11 – Resposta ao impulso do sistema compensado a partir da aproximação cônica.

Para a aproximação elíptica, o algoritmo retornou uma solução factível. É esperado devido o melhor aproveitamento da região ω_n -constante. A figura 10b mostra o diagrama de polos do sistema compensando. A matriz de ganho K possui o seguinte valor:

$$K = [2.5076 \quad -0.1807] \quad (4.6)$$

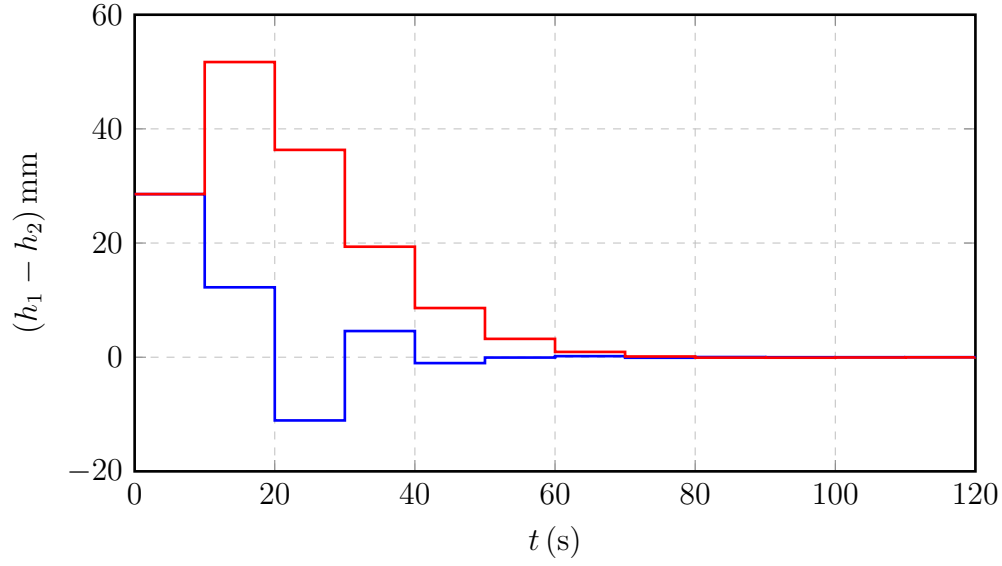


Figura 12 – Resposta ao impulso do sistema compensado a partir da aproximação elíptica.

Já para a aproximação poligonal, o algoritmo retornou uma solução factível com duas iterações. As regiões aproximadas foram representadas na figura 10b e a matriz de ganho K é dado por:

$$K = [2.4159 \quad -0.1554] \quad (4.7)$$

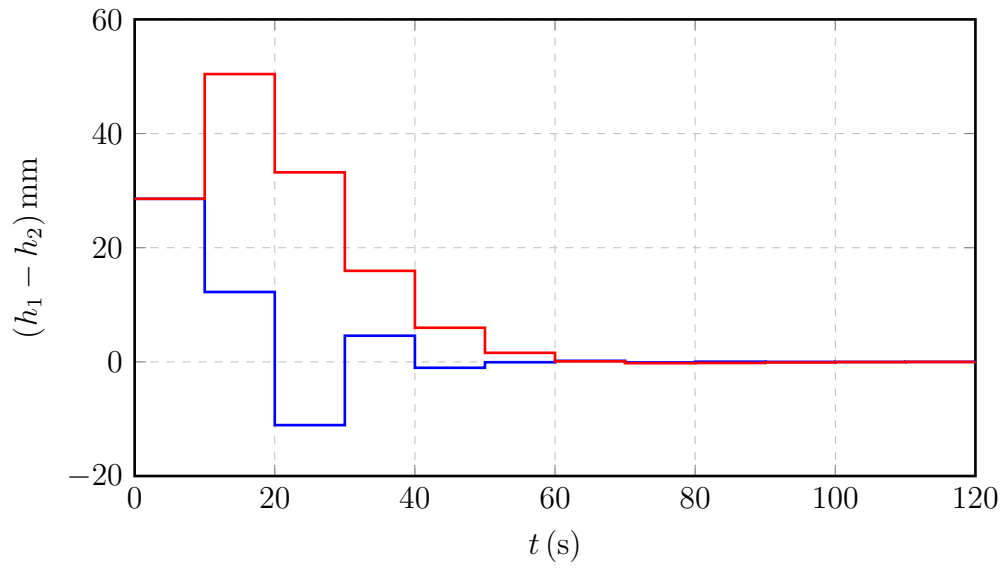


Figura 13 – Resposta ao impulso do sistema compensado a partir da aproximação poligonal.

5 Conclusão

Referências

- CHILALI, M.; GAHINET, P. H_∞ design with pole placement constraints: an lmi approach. *IEEE Transactions on Automatic Control*, v. 41, n. 3, p. 358–367, 1996.
- CHIQUETO, G. da S. Aproximações convexas via desigualdades matriciais lineares para o problema da largura de banda em sistemas em tempo discreto. 2021.
- KUO, B. *Digital Control Systems*. Holt, Rinehart and Winston, 1980. (HRW series in electrical and computer engineering). ISBN 9780030575686. Disponível em: <https://books.google.com.br/books?id=oNpSAAAAMAAJ>.
- NISE, N. *Control Systems Engineering, Sixth*. John Wiley & Sons, Incorporated, 2011. ISBN 9781118138168. Disponível em: <https://books.google.com.br/books?id=34zmCQAAQBAJ>.
- OGATA, K. *Engenharia de controle moderno*. Pearson Prentice Hall, 2011. ISBN 9788576058106. Disponível em: <https://books.google.com.br/books?id=iL3FYgEACAAJ>.
- ROSINOVÁ, D.; HOLÍČ, I. Lmi approximation of pole-region for discrete-time linear dynamic systems. In: *Proceedings of the 2014 15th International Carpathian Control Conference (ICCC)*. [S.l.: s.n.], 2014. p. 497–502.
- ROSINOVÁ, D.; HYPIUSOVÁ, M. Lmi pole regions for a robust discrete-time pole placement controller design. *Algorithms*, v. 12, n. 8, 2019. ISSN 1999-4893. Disponível em: <https://www.mdpi.com/1999-4893/12/8/167>.
- WISNIEWSKI, V.; MADDALENA, E.; GODOY, R. Discrete-time regional pole-placement using convex approximations: Theory and application to a boost converter. *Control Engineering Practice*, v. 91, p. 104102, 2019. ISSN 0967-0661. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0967066119301182>.
- WISNIEWSKI, V. L. et al. Regional pole placement for discrete-time systems using convex approximations. In: *2017 25th Mediterranean Conference on Control and Automation (MED)*. [S.l.: s.n.], 2017. p. 655–659.