Sistemas_Lineares_4-SCDT_RespImpulso

November 7, 2024

1 Análise de Sistemas em Tempo Contínuo no Domínio do Tempo

1.1 Resposta ao Impulso

Quando estudarmos a resposta de estado nulo, veremos que a resposta para qualquer função de entrada pode ser determinada através de uma operação especial, denominada de integral de convolução, enntre a função de entrada e a resposta ao impulso do sistema. Neste notebook estudaremos agora como calcular a resposta à uma entrada impulsiva.

Vamos substituir $x(t) = \delta(t)$ em $Q(D)y(t) = P(D)x(t)$. Também definimos $h(t)$ como a resposta ao impulso. Ainda, vamos inicialmente considerar $N = M$. Então, temos

$$Q(D)h(t) = P(D)\delta(t) \tag{1}$$

ou

$$(D^N + a_1 D^{N-1} + ... + a_N)h(t) = (b_0 D^N + b_1 D^{N-1} + ... + b_N)\delta(t) \tag{2}$$

No momento em que ocorre a entrada impulsiva ocorrerá o armazenamento súbito de energia no sistema, e imediatamente após (denotaremos este tempo por 0^+), o impulso se extingue, e o sistema responde como na resposta de entrada nula, mas com condições iniciais alteradas pelo impulso. Portanto, a resposta $h(t)$ contém termos relativos a resposta impulsiva e termos baseados nos modos característicos, quando a entrada impulsiva não está mais presente.

$$h(t) = A\delta(t) + \sum c_i e^{\lambda_i t}. \tag{3}$$

Substituindo (3) em (2), as derivadas dos termos impulsivos nos dois lados da equação devem coincidir. Então

$$D^N A\delta(t) = D^N b_0 \delta(t) \rightarrow A = b_0.$$

Por outro lado, se $M < N$, $A = b_0 = 0$ e $h(t)$ conterá apenas os modos característicos do sistema. Como vimos, a determinação da resposta baseada nos modos característicos é obtida através das novas condições iniciais impostas pela entrada impulsiva.

Considere a equação derivada do sistema (1) abaixo:

$$Q(D)g(t) = \delta(t) \tag{4}$$

Onde substituimos $P(D) = 1$ e $g(t)$ é conhecida como a função de Green do problema. No lado direito de (4) temos apenas o impulso, de modo que $g(0)$ e suas derivadas , com excessão de $D^N g(0)$ e $D^{(N-1)}g(0)$ devem ser nulas. Caso não fosse assim, no lado direito de (4) haveria termos como a derivada do impulso, o que não ocorre. Por outro lado, $D^N g(0) = \delta(t)$, e portanto $D^{N-1}g(0) = u(t)$, um degrau unitário. Além disso, note que os sistemas descritos pelas equações (1) e (4) compartilham os mesmos modos característicos, por possuirem o mesmo polinômio característico $Q(\lambda)$.

A solução da equação (4) para $t > 0$ é facilmente encontrada se lembrarmos que o impulso deixa de existir mas novas condições iniciais são estabelecidas como discutido acima. Assim temos

$$Q(D)g(t) = 0$$

com as condições iniciais $y^{N-1}(0^+) = 1$, $y^{N-2}(0^+) = y^{N-3}(0^+) = ... = y(0^+) = 0$. Ou seja, temos um problema típico de entrada nula. Tendo encontrado a função de Green, basta agora prémultiplicar $g(t)$, e daí também o impulso, pelo polinômio $P(D)$ original, ficando com

$$Q(D)P(D)g(t) = P(D)\delta(t) \tag{5}$$

e finalmente, comparando (1) e (5) vemos que $h(t) = A\delta(t) + P(D)g(t)$.

Exemplo Seja o sistema $(D^2 + 3D + 2)y(t) = (3D^2 + 1)x(t)$, e uma entrada impulsiva $x(t) = \delta(t)$. Determine a resposta.

Inicialmente, podemos verificar que $M = N = 2$ e $A = 3$. Portanto

$$h(t) = 3\delta(t) + \sum c_i e^{\lambda_i(t)}.$$

precisamos agora determinar os modos característicos de $h(t)$. Para isso resolveremos o seguinte problema:

$$(D^2 + 3D + 2)g(t) = 0$$

com as condições iniciais $g(0^+) = 0$, $g(0^+)' = 1$. As raízes da equação característica são $\lambda_1 = -1$ e $\lambda_2 = -2$. Então

$$g(t) = c_1 e^{-t} + c_2 e^{-2t}.$$

Inserindo as condições iniciais encontramos $c_1 = 1$ e $c_2 = -1$. Assim, a resposta ao impulso é

$$h(t) = 3\delta(t) + (3D^2 + 1)(e^{-t} - e^{-2t}) = 3\delta(t) + (4e^{-t} - 13e^{-2t})u(t).$$

Com o pequeno *script* abaixo podemos fazer o gráfico da resposta baseada nos modos característicos para $t > 0^+$.

[1]:
```
#!pip install numpy
#!pip install matplotlib
```

[2]:
```
import numpy as np
import matplotlib.pyplot as plt

t=np.linspace(0,4,100)
h= (4*np.exp(-t)-13*np.exp(-2*t))

plt.plot(t,h)
plt.ylim([-10,1]);
plt.grid()
```