

AULA PRÁTICA 3: CONVOLUÇÃO DISCRETA

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

CENTRO DE CIÊNCIA EXATAS E TECNOLÓGICAS, UNIVERSIDADE FEDERAL DE VIÇOSA

1 Introdução

Esta prática refere-se a operação de convolução para sistemas lineares invariantes no tempo e sua aplicação em sinais de áudio. A convolução empregada necessariamente é a discreta, pelo fato de se trabalhar com sistemas computacionais discretizados.

Para plotar ou reproduzir um sinal discretizado, é necessário especificar o incremento temporal T_s (em segundos) entre as amostras. Assim, a frequência de amostragem para tocar um sinal sonoro é definida por $F_s = 1/T_s$. Quando se carrega um arquivo de áudio, geralmente a frequência de amostragem também é carregada. Caso seja necessário ajustar as abscissas dos gráficos, é necessário definir um vetor de tempo discreto, por exemplo $\mathbf{t} = [0:T_s:T_{\text{fim}}];$, onde T_{fim} é número de pontos do sinal menos 1, multiplicado por T_s .

2 Comandos úteis

- `audiowrite` cria um arquivo de áudio no *workspace*;
- `conv` realiza a convolução entre dois sinais;
- `sound` reproduz um sinal de áudio, normalizando se necessário. Requer a frequência de amostragem do sinal;
- `help` informa sobre todos os comandos do Matlab.

3 Roteiro

3.1 Exercício 1: Convolução 1D

A função `conv` é usada para convoluir duas funções discretas $x[n]$ e $h[n]$. Ela assume que o incremento temporal é o mesmo em ambos sinais. Os sinais precisam ser finitos e o resultado da convolução tem tamanho igual à soma dos tamanhos de cada sinal restado de uma unidade ($L_1 + L_2 - 1$).

- Um sistema linear invariante no tempo é completamente descrito por sua resposta ao impulso. No Matlab, a resposta ao impulso precisa ser discreta. Por exemplo, considere o sistema com resposta ao impulso $\mathbf{h} = [1 \text{ zeros}(1,20) \ 0.5 \ \text{zeros}(1,10)];$. Crie um *script* usando o editor e plote a resposta deste impulso.
- Considere a seguinte entrada no sistema $\mathbf{x} = [0 \ 1:10 \ \text{ones}(1,5)*5 \ \text{zeros}(1,10)];$. Plote esta entrada.
- Convolua \mathbf{x} e \mathbf{h} , i.e., $\mathbf{y} = \text{conv}(\mathbf{x},\mathbf{h});$ Use o comando `subplot` para mostrar a resposta ao impulso, a entrada e a saída. Note que é preciso adicionar zeros ao final de \mathbf{x} e \mathbf{h} (para ajustar seu tamanho ao tamanho de \mathbf{y}) ou definir um vetor de tempo para cada sinal no sentido de compará-los temporalmente.

3.2 Exercício 2: Convolução de um sinal sonoro

Crie um novo *script* para este problema. Faça o *download* do arquivo `trumpet.mat` no PVAnet e carregue-o usando o comando `load` e plote-o. Use `whos` para ver que as variáveis `y` e `Fs` estão no *workspace*.

- Convolua o som do trompete com o sistema `h = [1 zeros(1,10000) 0.25 zeros(1,1000)]`, i.e., `y1 = conv(y,h)`. Reproduza o som (salve em `.wav` usando `audiowrite`) e depois plote-o. Alguma diferença foi notada em relação ao som original?
- Troque o sinal do coeficiente 0.25 de `h`. O que acontece com o som de saída?
- Altere o sistema para que o eco seja atrasado de um quarto de segundo inserindo `Fs/4` zeros antes do segundo impulso. `h = [1 zeros(1, round(Fs/4)) 0.25 zeros(1,1000)]`; Convolua o sistema com o sinal de entrada `trumpet`.

Olhe, ouça e compare a entrada, sistemas e saídas obtidas no item acima. Experimente com diferentes números de zeros. Discuta o efeito de cada alteração.

4 Exercício 3: Resposta ao impulso quadrado

Crie uma resposta ao impulso quadrada `h2 = [ones(1,50)/50 zeros(1,20)]`; Gere um novo sinal `y2` convoluindo `trumpet` com `h2`. Olhe e ouça os sinais. Qual a diferença entre entrada e saída?

O sinal convoluído parece mais "suave"? Um sistema que possui resposta ao impulso de acordo com `h2` é um filtro passa-baixa. Como a entrada é constituída de várias componentes de frequência diferentes, as componentes de alta-frequência da entrada são atenuadas ao se realizar a convolução como sistema dado. O que acontece se alterar o número de uns e zeros de `h2`?

Nota: No anexo não deve conter partes do código comentado. Exemplo:

```
% y = conv(x,h);  
% figure  
% plot(t,y)
```

Entretanto, deve haver comentários quanto ao uso das funções, criação de vetores etc. Exemplo:

```
y = conv(x,h) % realiza convolução entre x e h
```