EA614 - Análise de Sinais

Exercício de Fixação de Conceitos (EFC) 1 – Sistemas LIT e Convolução

Turma A – $1^{\underline{o}}$ semestre de 2021

Prof: Levy Boccato Email: `lboccato@dca.fee.unicamp.br`
PED-C: Renan Del Buono Brotto Email: `rbrotto@decom.fee.unicamp.br`

Introdução

Neste exercício, iremos estudar alguns aspectos básicos de um problema de grande relevância na área de comunicações, conhecido como equalização de canais, tendo como base os conceitos de convolução e sistemas lineares e invariantes com o tempo (LIT).

Visão Geral do Problema

Considere que um transmissor envia um conjunto de símbolos pertencentes a algum alfabeto finito (por exemplo, ±1 e ±3), aqui modelado por um sinal a tempo discreto $s[n]$, através de um canal (atmosfera, fibra ótica, par trançado, etc), modelado por um sistema LIT, cuja resposta ao impulso é $h[n]$, conforme mostra a Figura 1:

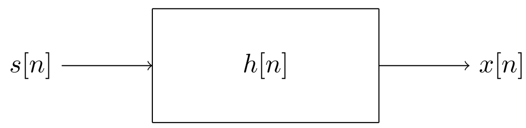

Figura 1: Transmissão através do canal $h[n]$.

Devido às características físicas do canal, o sinal que chega ao receptor $x[n]$ corresponde a uma versão distorcida do sinal original por conta de vários efeitos, entre os quais destacamos o fenômeno conhecido como interferência intersimbólica.

Assim sendo, o objetivo é projetar um filtro no receptor, denominado equalizador (também modelado como um sistema linear e invariante com o tempo, com resposta ao impulso $w[n]$), capaz de compensar as distorções observadas, como mostra a Figura 2:

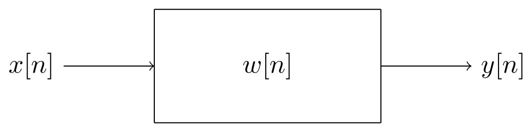

Figura 2: Uso de um equalizador $w[n]$ para processar o sinal recebido.

No caso de uma recuperação completa do sinal transmitido, temos que a saída do equalizador é $y[n] = s[n]$.

Parte Teórica

Vamos considerar um sinal de entrada $s[n]$ contendo K amostras. Vamos considerar também que o transmissor parte de uma condição de repouso, ou seja, $s[n] = 0$ para $n < 0$, como ilustrado na Figura 3:

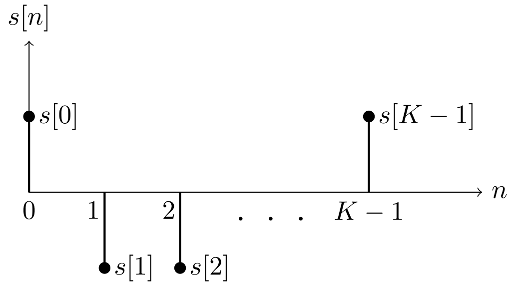

Figura 3: Sinal transmitido $s[n]$.

O sinal $s[n]$ em questão é transmitido através de um canal de resposta ao impulso finita, com comprimento D, e causal (*i.e.* $h[n] = 0$ para $n < 0$):

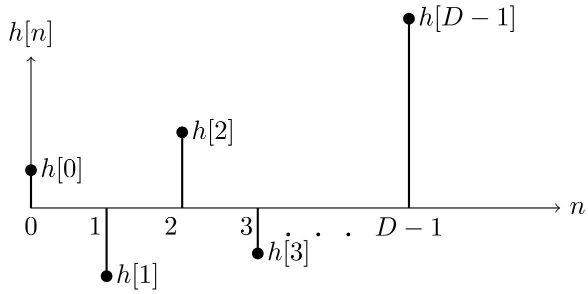

Figura 4: Resposta ao impulso do sistema $h[n]$.

Conforme visto em sala de aula, a saída $x[n]$ é obtida a partir da convolução entre $s[n]$ e $h[n]$, como mostra a expressão:

$$\boxed{x[n] = h[n] * s[n] = \sum_{k=-\infty}^{\infty} h[k]s[n-k] = \sum_{k=-\infty}^{\infty} s[k]h[n-k].} \tag{1}$$

(a) Determine o comprimento P da sequência $x[n]$ gerada na saída do canal em função de K e D, mostrando claramente a sua derivação.

(b) Como tanto a entrada $s[n]$ quanto a resposta ao impulso $h[n]$ são sequências de comprimento finito, é possível determinar a saída $x[n]$ explorando uma representação vetorial. Seja $\mathbf{x} = \begin{bmatrix} x[0] \ x[1] \ \cdots \ x[P-1] \end{bmatrix}^T$ o vetor que descreve a saída $x[n]$. Então, podemos escrever que:

$$\boxed{\mathbf{x} = \mathbf{H}\mathbf{s},} \tag{2}$$

onde $\mathbf{H} \in \mathbb{R}^{P \times K}$ é denominada a matriz de convolução do sistema e $\mathbf{s}$ é o vetor que representa o sinal transmitido. Mostre que este procedimento, definido na equação (2), para o cálculo da convolução está correto, identificando quem é a matriz $\mathbf{H}$ e o vetor $\mathbf{s}$.

Vamos considerar um cenário específico neste exercício, no qual, ao transmitirmos o sinal $s[n]$ através do canal, recebemos sua versão distorcida $x[n]$, conforme a seguinte relação:

$$\boxed{x[n] = s[n] - 0{,}5s[n-1].} \tag{3}$$

(c) A partir da equação (3), determine a resposta ao impulso do canal $h[n]$.

Combinando os diagramas mostrados nas Figuras 1 e 2, o processo de equalização é descrito de maneira completa pelo diagrama apresentado na Figura 5:

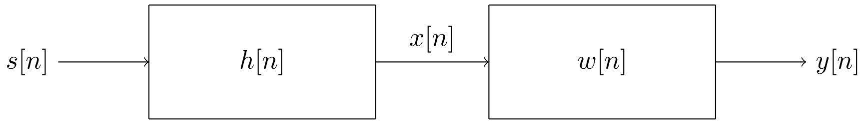

Figura 5: Processo de equalização.

Na situação ideal em que conseguimos equalizar completamente o canal, temos que a saída do equalizador corresponde ao próprio sinal de entrada, ou seja:

$$\boxed{y[n] = s[n].} \tag{4}$$

(d) Considerando a situação de equalização ideal, determine a resposta combinada canal-equalizador.
Dica: note que o canal $h[n]$ e o equalizador $w[n]$ são dois sistemas LIT em série (cascata).

Parte Computacional

(e) Vamos considerar agora dois filtros candidatos a equalizador, cujos coeficientes são mostrados a seguir:

$$\mathbf{w_1} = [1 \;\; 0.5 \;\; (0.5)^2 \;\; (0.5)^3 \;\; (0.5)^4], \tag{5}$$

$$\mathbf{w_2} = [1 \;\; 1.5 \;\; 0.7 \;\; -0.2 \;\; 0.3]. \tag{6}$$

Apresente, então, a resposta combinada para cada um dos filtros usados, ou seja, $g_1[n] = w_1[n] * h[n]$ e $g_2[n] = w_2[n] * h[n]$. A partir das respostas combinadas obtidas, discuta a qualidade de cada um dos filtros tendo em vista o objetivo desejado na tarefa de equalização.
Observação: para o cálculo da convolução, implemente uma rotina que realize a operação matricial tratada no item (b).

(f) Crie um conjunto de 100 amostras, assumindo os valores $+3$, $+1$, -1 e -3 com (aproximadamente) igual probabilidade, para o sinal $s[n]$. Em Matlab, isto pode ser feito através dos seguintes comandos:

```
s = rand(1,100);
s(s < 0.25) = 3;
s(s < 0.50) = 1;
s(abs(s) < 0.75) = -1;
s(abs(s) < 1) = -3;
```

Obs.: Também é possível usar o comando `randsrc`, configurando o alfabeto para os valores desejados; no Octave, é preciso antes carregar o pacote de comunicações, através do comando `pkg load communications`.

Em Python, isto pode ser feito através dos seguintes comandos:

```
import numpy as np
s = np.random.uniform(0, 1, 100)
s[np.where(s < 0.25)] = 3
s[np.where(s < 0.5)] = 1
s[np.where(np.abs(s) < 0.75)] = -1
s[np.where(np.abs(s) < 1)] = -3
```

Obs.: Também é possível usar o comando `random.choices`.

Simule, então, a transmissão deste sinal pelo canal $h[n]$. Ou seja, faça a convolução entre o vetor `s` gerado e o vetor `h`, composto pelo coeficientes da resposta ao impulso do canal $h[n]$ obtida no item (c). O resultado desta convolução é o vetor `x`, que contém as amostras do sinal recebido ($x[n]$). Compare os valores assumidos pelos sinais $x[n]$ e $s[n]$.

Para essa etapa, pode ser interessante fazer um histograma dos símbolos observados; em Matlab, isso pode ser feito com o comando `hist(s)`, `hist(x)`; já em Python, esse tipo de gráfico está implementado na biblioteca Matplotlib.pyplot e basta fazer `import matplotlib.pyplot as plt` e depois usar os comandos `plt.hist(s)`, `plt.hist(x)`.

(g) Filtre o sinal $x[n]$ pelos equalizadores $w_1[n]$ e $w_2[n]$ (cujos coeficientes foram apresentados no item e)), gerando as saídas $y_1[n]$ e $y_2[n]$, respectivamente.

Faça, então, dois gráficos (*i.e.* duas figuras diferentes no Matlab ou em Python), detalhados a seguir:

- Gráfico 1: em um mesmo gráfico, plote o sinal de entrada $s[n]$ em azul e a saída $y_1[n]$ em vermelho.
- Gráfico 2: em um mesmo gráfico, plote o sinal de entrada $s[n]$ em azul e a saída $y_2[n]$ em vermelho.

Os seguintes comandos no Matlab podem ser empregados para a geração dos gráficos:

`figure()` – abre uma nova figura no Matlab
`stem()` – usado para plotar gráficos de valores discretos
`hold on` – comando do Matlab usado para plotar mais de um gráfico na mesma figura
`xlabel()` – atribui um nome ao eixo x
`ylabel()` – atribui um nome ao eixo y
`title()` – título do gráfico.

Em Python, os comandos são os mesmos que os do Matlab, sendo necessário colocar `plt.` no início.

Com base nestes dois gráficos, qual das saídas obtidas está mais próxima do sinal original $s[n]$?