DCA0118 - Processamento Digital de Sinais
Roteiro do Trabalho Referente à Unidade 2
Docente: Pedro Yochinori Gushiken
Período: 2018.2
Data: 5 de Junho de 2018

1. **Seja o seguinte sinal $x_a(t)$:**

$$x_a(t) = sen(2\pi f_1 t) + sen(2\pi f_2 t) + sen(2\pi f_3 t) \tag{1}$$

Este sinal foi amostrado numa taxa fixa de 2500Hz durante 3 segundos, produzindo uma sequência x_1(n). Para obter as frequências f_1, f_2 e f_3 o aluno deve utilizar seu número de matrícula, retirando os 3 últimos dígitos não nulos e realizando as seguintes operações:

$$\begin{aligned} f_1 &= a \\ f_2 &= 10a + b \\ f_3 &= 100a + 10b + c \end{aligned} \tag{2}$$

Por exemplo: Se o número de matrícula do aluno é 2014000358, teremos $a = 3$, $b = 5$ e $c = 8$, e as frequências correspondentes ao trabalho daquele aluno serão $f_1 = 3$, $f_1 = 35$ e $f_1 = 358$.
Para produzir o sinal x_1 no scilab, assumindo que f_1, f_2 e f_3 são como no exemplo, utilize os seguintes comandos:

x_1 = sin(3*linspace(0,3,2500*3))+sin(35*linspace(0,3,2500*3))+sin(358*linspace(0,3,2500*3))

Ao utilizar este comando, o software irá gerar um vetor x_1 correspondente às 2500 amostras produzidas por segundo durante 3 segundos, nas frequências 3, 35 e 358 das senoides.

Seja também o seguinte sinal $x_b(t)$:

$$x_b(t) = \begin{cases} 3sen(2\pi f_1 t), & 0 \leq t < 1 \\ 3sen(2\pi f_2 t), & 1 \leq t < 2 \\ 3sen(2\pi f_3 t), & 2 \leq t < \infty \end{cases} \tag{3}$$

Este sinal foi amostrado numa taxa fixa de $2500Hz$ durante 3 segundos, produzindo uma sequência $x_2(n)$. Para produzir este sinal, utilize o seguinte comando no Scilab ou Matlab:

x_2 = [sin(3*linspace(0,1,2500*1)) sin(35*linspace(1,2,2500*1)) sin(358*linspace(2,3,2500*1))]

Substituindo os números em verde de forma apropriada, de acordo com seu número de matrícula.

a **Usando uma ferramenta numérica para o cálculo do espectro amostrado de $x_1(n)$ e de $x_2(n)$, tal como a DFT ou FFT, produza as sequências $X_1(k)$ e $X_2(k)$, e indique qual a resolução de frequência para o número de pontos utilizado.**

Para obter a DFT dos sinais, basta utilizar os comandos fft(x_1) e fft(x_2). Veja o exemplo a seguir:

```
 //Componentes de frequência de um sinal
//----------------------------------
```

```
// Construa um sinal amostrado a 1000Hz contendo componentes puras em
// 50 e 70 Hz
taxa_amostragem=1000;
t = 0:1/taxa_amostragem:0.6; //vetor tempo
N=size(t,'*'); //numero de amostras
s=sin(2*%pi*50*t)+sin(2*%pi*70*t+%pi/4)+grand(1,N,'nor',0,1); //sinal

y=fft(s); //fft do sinal

//s é real então a fft é simétrica e conjugada,
// portanto plotamos apenas os primeiros N/2 pontos
f=taxa_amostragem*(0:(N/2))/N; //vetor de frequências
n=size(f,'*')
clf()
plot(f,abs(y(1:n)))
```

Lembre-se que a resolução de frequência trata do mapeamento entre a frequência de amostragem e a frequência no domínio discreto. Para obter uma boa plotagem, é necessário especificar um vetor f que corresponda à localização das componentes de frequência dos sinais. Por exemplo, se a frequência de amostragem é de 100 Hz e observamos durante 10 segundos teremos 1000 amostras, o que nos dará 1000 amostras nos pontos de frequência mapeados entre -50 e +50 Hz, correspondendo a uma resolução de 0.1Hz, .

b **Reconstrua as sequências $x_1(n)$ e $x_2(n)$ a partir de seus espectros amostrados, utilizando uma ferramenta numérica tal como a IDFT ou IFFT. O comando relevante é o ifft no scilab.**

c **Produza gráficos detalhando o valor absoluto dos espectros amostrados $X_1(k)$ e $X_2(k)$, utilizando o mapeamento de frequências produzido pela frequência de amostragem de $2500Hz$. Discuta a diferença entre o valor absoluto dos espectros amostrados $X_1(k)$ e $X_2(k)$.**

d **Implemente um filtro passa faixa $H_1(k)$ capaz de admitir sinais com componentes de frequência pertencentes à frequência f_1 e rejeitar as frequências f_2 e f_3. Repita o experimento para um filtro $H_2(k)$ e $H_3(k)$.**

Para projetar os filtros, você irá precisar das funções ones(começo,fim) e zeros(começo,fim) para criar vetores que correspondem à resposta em frequência desejada do seu filtro, levando em conta o zero padding ou preenchimento com zeros apropriado. Primeiramente, preencha os vetores x_1 e x_2 com zeros:
x_1 = [x_2 zeros(1,7500)];
x_2 = [x_2 zeros(1,7500)];
A seguir, lembrando que o mapeamento de frequências é de 2500/2 até 0 e de -2500/2 até 0, com uma resolução de 15000 amostras (pois adicionamos os zeros), especifique um vetor H_1. Neste caso, há várias opções possíveis:

a **Caso deseje que H1 seja passa baixa seu formato sera:**
H1 = [ones(1,y) zeros(1,x) ones(1,y)];

b **Caso deseje que H1 seja passa alta seu formato sera:**
H1 = [zeros(1,y) ones(1,x) zeros(1,y)];

c **Caso deseje que H1 seja passa faixa seu formato sera:**
H1 = [zeros(1,n) ones(1,x) zeros(1,y) ones(1,x) zeros(1,n)];

d **Caso deseje que H1 seja rejeita faixa seu formato sera:**
H1 = [ones(1,n) zeros(1,x) ones(1,y) zeros(1,x) ones(1,n)];

e **Demonstre a capacidade de seletividade de sequência dos filtros implementados, aplicando-os aos sinais $x_1(n)$ e $x_2(n)$, utilizando a FFT e IFFT. Indique em seu relatório qual o resultado esperado. Discuta e explique por que a operação correspondente no domínio das amostras não é normalmente implementada numericamente.**

Neste caso, faça a multiplicação elemento a elemento da fft dos sinais x_1 e x_2 pelos filtros H_1, H_2 e H_3, obtendo os sinais filtrados através da IFFT deste produto.