

PROCESSAMENTO DIGITAL DE SINAIS



PROJETO III

DETECÇÃO DE SINAIS DTMF



Departamento de Eletrônica e Sistemas - UFPE



PROJETO III

DETECÇÃO DE SINAIS DTMF

Neste projeto será mostrado como o sistema de teclas em um telefone usa sinais de diferentes frequências para indicar qual tecla foi pressionada. A Transformada de Fourier de Tempo Discreto (TFTD) de um sinal de telefone amostrado pode ser usada para identificar essas frequências. O som que você escuta quando uma tecla é pressionada é a soma de duas senóides. A senóide de frequência mais alta indica a coluna e a de frequência mais baixa indica a linha do teclado. A figura 1 a seguir mostra um teclado juntamente com as duas frequências correspondentes a cada dígito, considerando que a forma de onda contínua é amostrada a 8192 kHz. A figura mostra ainda uma tabela contendo as frequências de cada dígito. Assim, por exemplo, o dígito 5 é representado pelo sinal

$$d_5 = \text{sen}(0,5906n) + \text{sen}(1,0247n) \quad (1)$$

	Freq. (colunas)		
Freq. (linhas)	0,9273	1,0247	1,1328
0,5346	1	2	3
0,5906	4	5	6
0,6535	7	8	9
0,7217		0	

(a)

Dígito	Frequência (Linha)	Frequência (Coluna)
0	0,7217	1,0247
1	0,5346	0,9273
2	0,5346	1,0247
3	0,5346	1,1328
4	0,5906	0,9273
5	0,5906	1,0247
6	0,5906	1,1328
7	0,6535	0,9273
8	0,6535	1,0247
9	0,6535	1,1328

(b)

Figura 1 – Frequências da TFTD para os tons de um telefone a teclado.

Parte 1: Problemas Básicos

Nesses problemas você criará o tom apropriado para cada dígito e examinará a TFTD para verificar se os sinais tem as frequências corretas. Você também definirá um vetor contendo os tons de seu número de telefone.

a) Crie vetores linha **d0** a **d9** para representar todos os 10 dígitos no intervalo $0 \leq n \leq 999$. Escute cada sinal usando **sound**. Por exemplo, **sound(d2, 8192)** deve soar como o tom que você ouve quando pressiona o “2” no teclado.

b) A função **fft** pode ser usada para computar N amostras da TFTD de um sinal de comprimento finito, nas frequências $\omega_k = 2k\pi/N$. Por exemplo, **fft(x, 2048)** computa 2048 amostras igualmente espaçadas de $X(e^{j\omega})$ em $\omega_k = 2k\pi/2048$, para $0 \leq k \leq 2047$. Use **fft** para computar amostras de $D_2(e^{j\omega})$ e $D_9(e^{j\omega})$ em $\omega_k = 2k\pi/2048$. Defina **omega** como sendo um vetor contendo ω_k para $0 \leq k \leq 2047$. Esboce as amplitudes da TFTD para esses sinais em gráficos rotulados e confirme que os picos ocorrem para as frequências dadas na figura 1.

c) Defina **espaço** como sendo um vetor com 100 amostras de “silêncio” usando **zeros**. Defina seu número de telefone como **fone** através dos sinais apropriados e **espaço**. Por exemplo, se seu número é 32718210, você vai digitar

```
>> fone = [d3 espaço d2 espaço d7 espaço d1 espaço d8 espaço d2 espaço d1 espaço d0];
```

Observe que os dígitos definidos na parte (a) e **espaço** devem ser vetores linha. “Toque” seu número usando **sound** e verifique que ele soa do mesmo modo que quando você o digita no teclado.

Parte 2: Problemas Intermediários

Nesta parte você aprenderá a decodificar números de telefone. Os números a ser decodificados estão em um arquivo chamado **touch.mat**. Os dados podem ser carregados no MATLAB digitando **load touch**. Se o arquivo for carregado corretamente você deve ser capaz de listar os nomes das variáveis digitando

```
>> who
```

As variáveis são

```
hardx1  hardx2  x1  x2
```

Os vetores **x1** e **x2** contém versões amostradas das sequências representando dois diferentes números (as sequências são como na parte (c)). Os vetores **hardx1** e **hardx2** são versões discadas menos precisas dos mesmos números.

d) Usando *fft* compute 2048 amostras igualmente espaçadas da TFTD de cada dígito de *x1*. Para aplicar *fft* a cada dígito separadamente será preciso segmentar o sinal em sete dígitos usando a informação sobre os comprimentos relativos dos dígitos e espaços, ou então esboçando o sinal e identificando onde cada dígito começa e termina. Para determinar os dígitos do número de telefone representado por *x1*, esboce a amplitude da TFTD e compare as frequências de pico do sinal com aquelas da figura 1 (na sua resposta a soma dos dígitos do número encontrado deve ser 41).

e) Repita a parte (d) para o sinal *x2* e decodifique os dígitos desse número também (nesse caso a soma dos dígitos não é 41).

Parte 3: Problemas Avançados

Nesta parte você escreverá uma função para decodificar números de telefone, automaticamente, a partir dos tons gerados por um teclado. Para projetar o decodificador você observará as energias dos tons em cada uma das possíveis frequências indicadas na figura 1.

f) Usando *fft* para computar 2048 amostras de $X(e^{j\omega})$, encontre que valor de ω_k , e do índice k correspondente, é mais próximo de cada uma das frequências dos tons. Lembre que os vetores no MATLAB iniciam com o índice $k=1$, de modo que a amostra da DTFT em $\omega = 0$ é armazenada em $X(1)$.

g) O valor de $|X(e^{j\omega_k})|^2$ dá a energia em um sinal, na frequência ω_k . Calcule a DFT de **d8** (definido na parte (a)), e use a saída da *fft* para computar $|D_8(e^{j\omega_k})|^2$ para cada um dos ω_k determinado na parte (f). A energia é a maior para os valores de ω_k correspondentes a um “8”?

h) Escreva uma função *decdtmf* que aceita como entrada um sinal DTMF no formato usado na parte (c) e retorna como saída um vetor de comprimento 7 contendo o número de telefone. Teste sua função usando as entradas *x1* e *x2* e verifique que ela retorna os mesmos números que você obteve nas partes (d) e (e). Repita para o número definido na parte (c).

Entrega: data do EE3

Prof. Ricardo Campello