Processamento Digital de Sinais e Aplicações em Acústica
Tutorial 6 - Transformada de Fourier de Tempo Discreto (DTFT) e Transformada Discreta de Fourier (DFT)

https://github.com/fchirono/Aulas_PDS_Acustica

Objetivos do Tutorial

Ao final desta sessão, você será capaz de:

- aplicar a Transformada de Fourier de Tempo Discreto (DTFT) a um sinal de tempo discreto finito,
- compreender as implicações do teorema da amostragem na DTFT,
- aplicar a Transformada Discreta de Fourier (DFT) e a Transformada Discreta de Fourier Inversa (IDFT) a uma sequência,
- comparar espectros calculados usando a DTFT e a DFT, e
- atribuir uma frequência correspondente a cada amostra da DFT.

Sumário

1 Transformada de Fourier de Tempo Discreto (DTFT)

1.1 Revisão

A Transformada de Fourier de Tempo Discreto (DTFT) é definida da seguinte forma:

$$X(\Omega) = \sum_{n=-\infty}^{\infty} x[n] \cdot e^{-jn\Omega}, \qquad \Omega = 2\pi \frac{f}{f_s} \in [-\pi, \pi]. \tag{1}$$

Note que $X(\Omega)$ é uma função periódica, com período 2π.

1.2 DTFT de Sinais Não-Periódicos

Leia os arquivos `A1.wav` e `A2.wav` usando o módulo `scipy.io.wavfile`. Para incluir este módulo, use o comando:

```
from scipy.io import wavfile
```

A função `wavfile.read` pode ser usada para ler um arquivo .WAV de 16 bits por amostra no ambiente de trabalho da seguinte forma:

```
# le a frequencia de amostragem e valores das amostras do arquivo .WAV
fs_wav, sinal_int16 = wavfile.read("Exemplo.wav")

# confirma o tipo de dados obtidos do arquivo wav
sinal_int16.dtype

# normalizacao dos dados de audio de int16 para ponto flutuante
N_bits = 16
sinal_float = sinal_int16/(2.**(N_bits - 1)-1)
```

Tarefas

1. Mostre analiticamente que a DTFT é periódica, com período 2π. **DICA**: Calcule $X(\Omega + 2\pi)$ usando a equação (1).

2. Ouça ambos os sinais `A1` e `A2` usando seu reprodutor de mídia local. Eles têm a mesma frequência fundamental?

3. Escreva uma função `DTFT(x, Omega)` que implemente a equação (1). O vetor de entrada `Omega` da função contém as frequências angulares normalizadas em radianos/amostra a ser usadas na análise. Note que, embora a DTFT teoricamente tenha uma resolução de frequência infinita, em uma implementação computacional só podemos especificar uma quantidade limitada de frequências discretas.

4. Crie um vetor `Omega` contendo 4096 pontos igualmente espaçados no intervalo $[-\pi, \pi]$ definindo um conjunto de frequências angulares normalizadas em radianos/amostra, e use a função recém-implementada para calcular os espectros de Fourier de ambos os arquivos `.wav`. Plote o valor absoluto de ambos os resultados em cores diferentes.

5. Qual é a resolução de frequência resultante em Hz da análise de Fourier realizada no item anterior?

6. Que aspectos visíveis nos dois espectros de amplitude podem confirmar a sua observação na Tarefa 2?

7. Defina um novo vetor `f` (baseado em `Omega`) que contenha as frequências correspondentes em Hertz. Plote o espectro de amplitude de `A1.wav` e use `f` para o eixo "x".

8. Agora redefina `Omega` para 4096 frequências angulares normalizadas uniformemente espaçadas no intervalo $[-3\pi, 3\pi]$, calcule e plote a (magnitude da) DTFT para esta nova faixa de frequências normalizadas. Quais efeitos podem ser observados nos novos espectros de amplitude, e por que isso ocorre?

2 Transformada Discreta de Fourier (DFT) e Transformada Discreta de Fourier Inversa (IDFT)

2.1 Revisão

Dada uma sequência $x[n]$ de comprimento N, a DFT de $x[n]$ pode ser definida da forma

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot W_N^{kn}, \qquad k = 0, 1, 2, \ldots, N-1 \tag{2}$$

onde, por definição,

$$W_N = e^{-\frac{j2\pi}{N}}. \tag{3}$$

A Transformada Discreta de Fourier Inversa (IDFT) é definida como

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \cdot W_N^{-kn}, \qquad n = 0, 1, 2, \ldots, N-1. \tag{4}$$

A DFT e a IDFT também podem ser definidas como transformações lineares que operam sobre sequências $\mathbf{x}_N$ e $\mathbf{X}_N$, respectivamente. Seja $\mathbf{x}_N = [x[0], \ldots, x[N-1]]^T$ e $\mathbf{X}_N = [X[0], \ldots, X[N-1]]^T$ o vetor da amostras do sinal no tempo e o vetor de amostras na frequência, respectivamente. Seja a matriz $\mathbf{W}_N$, com elementos W_N^{kn}, definida como

$$\mathbf{W}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N & W_N^2 & \cdots & W_N^{N-1} \\ \vdots & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix}. \tag{5}$$

Usando as definições acima, a DFT pode ser expressa em forma matricial como

$$\mathbf{X}_N = \mathbf{W}_N \cdot \mathbf{x}_N. \tag{6}$$

O mesmo se aplica à IDFT, ou seja,

$$\mathbf{x}_N = \frac{1}{N} \mathbf{W}_N^* \cdot \mathbf{X}_N, \tag{7}$$

onde $\mathbf{W}_N^*$ é o conjugado complexo da matriz $\mathbf{W}_N$.

2.2 Tarefas

1. Mostre analiticamente que a DFT é periódica, com período N. **DICA**: Avalie $X[k+N]$ usando a equação (2).
   Mostre analiticamente que a DFT assume que $x[n]$ é periódico, com período N. **DICA**: Avalie $x[n+N]$ usando a equação (4).

2. Escreva uma função `DFT(x, N_DFT)` que implemente a equação (6), onde `x` é a sequência de dados e `N_DFT` é o número de pontos sobre os quais a DFT é calculada.

   Note que se o comprimento de `x` for menor que `N_DFT`, é prática comum preencher o sinal com zeros no final do vetor (*zero-padding*); similarmente, se o comprimento de `x` for maior que `N_DFT`, as amostras excedentes devem ser ignoradas.

3. Gere uma sequência aleatória `x` de comprimento $L \leq 20$ com a função `numpy.random.randn`. Calcule a DFT de `x` para os casos:

   (a) `N_DFT = L`

   (b) `N_DFT > L`

   (c) `N_DFT < L`

4. Para os três casos acima, plote a magnitude e a fase da DFT calculada no ponto 2 usando a função `matplotlib.pyplot.stem`. Qual é o efeito da escolha de `N_DFT`?

5. Verifique o valor da fase em $\Omega = 0$ rad/amostra e $\Omega = \pi$ rad/amostra para a DFT calculada com `N_DFT = L`.

6. Calcule a DFT da sequência `x` usando a função `numpy.fft.fft` (leia a documentação do módulo `numpy.fft` para mais informações). Escolha `N_dft = L` e plote a magnitude e a fase. Como eles se comparam aos gráficos gerados na Tarefa 4?

7. Implemente uma função em Python para a IDFT definida na Equação (7).

8. Usando a função `IDFT` implementada na tarefa anterior, calcule a IDFT do sinal `X` para os três casos da Tarefa 3 e plote os resultados. Como os sinais resultantes se comparam ao sinal original no domínio do tempo?

3 DTFT vs. DFT

3.1 Revisão

De acordo com a definição da DFT, as amostras da DFT

$$\mathbf{X} = [X[0], \ldots, X[N-1]]$$

correspondem à frequências discretas da seguinte forma:

$$X[k] = X\left(k\frac{f_S}{N}\right),$$

onde $k = [0 \ldots N-1]$. A resolução em frequência é dada por $df = fs/N$, e a DFT estará centrada na frequência de Nyquist $f_s/2$.

No entanto, devido à sua simetria, o vetor da DFT é frequentemente rearranjado para ser consistente com o resultado da Transformada de Fourier - isto é, iniciando nas frequências *negativas*, centrado na frequência 0, e exibindo simetria conjugada complexa. Para fazer isso corretamente, é importante distinguir entre dois casos:

- N par:
  Quando N é um número par, a $(\frac{N}{2})$-ésima amostra ($k = N/2$) corresponderá **exatamente** à frequência de Nyquist.

- N ímpar:
  Quando N é um número ímpar, a frequência de Nyquist corresponde ao ponto entre as amostras $(N-1)/2$ e $(N+1)/2$, e nenhuma amostra da DFT corresponderá exatamente à frequência de Nyquist.

O vetor de frequência correto para cada caso é definido da seguinte forma:

- N par:

$$\mathbf{X} \;=\; \left[X\left[-\frac{N}{2}\right]\;,\;X\left[-\frac{N}{2}+1\right],\ldots,X\left[-1\right]\;,\;X\left[0\right],X\left[1\right],\ldots,X\left[\frac{N}{2}-1\right]\right]$$

  Note que neste caso, a amostra correspondente à frequência de Nyquist é o **primeiro** elemento do vetor, o que é consistente com o resultado da função `numpy.fft.fftshift`.

- N ímpar:

$$\mathbf{X} \;=\; \left[X\left[-\frac{(N-1)}{2}\right]\;,\;X\left[-\frac{(N-1)}{2}+1\right],\ldots,X\left[-1\right],X\left[0\right],X\left[1\right],\ldots,X\left[\frac{(N-1)}{2}\right]\right]$$

  Note que neste caso, a frequência de Nyquist não estará contida no resultado da DFT.

É **IMPORTANTE** que você compreenda bem as duas definições acima, pois **esta é uma causa frequente de erros e confusão!**

3.2 Resolução de Frequência Limitada

Tarefas

1. Pegue um trecho do arquivo de áudio "A1.wav" da amostra 5100 até 6400 e salve-o em um vetor separado `d`.
2. Usando a função que você criou, calcule a DTFT usando o vetor de frequência angular normalizada `Omega` criado na Tarefa 1.3 e plote o espectro de amplitude sobre o vetor de frequência correspondente **em Hz** na ordenada.
3. Calcule a DFT de `d` com comprimento correspondente ao comprimento de `d`. Rearranje o vetor de saída para que você obtenha um vetor com simetria conjugada complexa, e então gere um segundo vetor com a frequência correspondente em Hz para cada amostra da DFT (verifique a subseção anterior!).
4. Plote o espectro de amplitude obtido através da DFT sobre o gráfico da DTFT existente, usando o comando `plt.stem` e use uma cor diferente.
5. O que pode ser observado na figura resultante?