Processamento Digital de Sinais
Tutorial 10 - Espectro de Potência

Fabio Casagrande Hirono

Objetivo de Aprendizagem

Ao final desta sessão, você será capaz de:

- Estimar o espectro de potência cruzado e auto-espectro de potência de sequências em tempo discreto
- Usar o espectro de potência cruzado e auto-espectro para estimar a resposta em frequência e as funções de resposta ao impulso de sistemas lineares, invariantes no tempo (LTI).

Sumário

1 Revisão

1.1 Estimação do espectro de potência cruzado e auto-espectro usando o método de Welch

Sejam $x[n]$ e $y[n]$, $n = 0, \ldots, K-1$ duas sequências em tempo discreto. Estas duas sequências são divididas em L quadros de comprimento N_{DFT}, com $N_{sobreposicao}$ amostras sobrepostas entre quadros adjacentes. Para o l-ésimo quadro, $l = 0, \ldots, L-1$, a estimativa do espectro de potência cruzado discreto $\widetilde{P}_{xy}^{(l)}(\Omega)$ é calculada como

$$\widetilde{P}_{xy}^{(l)}(\Omega) = \frac{1}{N_{DFT} \cdot U} X_w^{(l)}(\Omega) \left[Y_w^{(l)}(\Omega) \right]^*, \tag{1}$$

$$\widetilde{P}_{xy}^{(l)}(\Omega) = \frac{1}{N_{DFT} \cdot U} \sum_{n=0}^{N_{DFT}-1} w[n] x[n+lD] e^{-j\Omega n} \left[\sum_{m=0}^{N_{DFT}-1} w[m] y[m+lD] e^{-j\Omega m} \right]^*, \tag{2}$$

onde Ω é a frequência angular normalizada, $X^{(l)}(\Omega)$ e $Y^{(l)}(\Omega)$ são as DFT das sequências $x[n]$ e $y[n]$ no l-ésimo quadro, $w[n]$ é uma sequência de "janela" de comprimento N_{DFT}, $U = \frac{1}{N_{DFT}} \sum_{n=0}^{N_{DFT}-1} w^2[n]$ é um fator de normalização da janela, e $D = N_{DFT} - N_{sobreposicao}$.

O número de quadros pode ser calculado como

$$L = \frac{K - N_{sobreposicao}}{N_{DFT} - N_{sobreposicao}}. \tag{3}$$

O espectro de potência cruzado discreto $P_{xy}(\Omega)$ é calculado como

$$P_{xy}(\Omega) = \frac{1}{L}\sum_{l=0}^{L-1} \widetilde{P}_{xy}^{(l)}(\Omega). \tag{4}$$

Similarmente, o auto-espectro de potência $P_{xx}(\Omega)$ da sequência $x[n]$ é definido como

$$P_{xx}(\Omega) = \frac{1}{L}\sum_{l=0}^{L-1} \widetilde{P}_{xx}^{(l)}(\Omega) = \frac{1}{L}\frac{1}{N \cdot U}\sum_{l=0}^{L-1} X_w^{(l)}(\Omega)\left[X_w^{(l)}(\Omega)\right]^* \tag{5}$$

1.2 Coerência entre dois sinais (usando os espectros de potência)

A função de coerência ao quadrado entre dois sinais é definida como

$$\gamma^2(\Omega) = \frac{|P_{xy}(\Omega)|^2}{P_{xx}(\Omega)P_{yy}(\Omega)}. \tag{6}$$

1.3 Estimação da função de transferência usando o estimador H1

A estimação da função de transferência $\widetilde{H}(\Omega)$ de um sistema LTI pode ser obtida pelo estimador H_1:

$$\widetilde{H}(\Omega) = \frac{P_{yx}(\Omega)}{P_{xx}(\Omega)}. \tag{7}$$

2 Tarefas

1. Carregue a função de resposta em frequência `H_sys`, a resposta ao impulso `h_sys` e o comprimento da DFT `N_dft` fornecidos no arquivo `tutorial10.mat`. Esta resposta em frequência é uma função unilateral - ou seja, é dada apenas para $\Omega \in [0, \pi]$ rad/amostra - e foi obtida com uma frequência de amostragem de $f_s = 44100$ Hz. Calcule o vetor de frequência equivalente f correspondente aos pontos da resposta em frequência, e plote a magnitude da resposta em frequência obtida.

2. Calcule a resposta ao impulso `h_sys` a partir da função de resposta em frequência dada. Para fazer isso, assuma que a resposta ao impulso é uma função de valor real. Você portanto terá que complementar a resposta em frequência dada para obter uma função bilateral.

3. Crie um sinal de ruído branco `x` de comprimento $K = 44100$ amostras e calcule a convolução $x[n] * h[n] = y[n]$ usando as funções que você criou durante o Tutorial 4. Se você não estiver confiante sobre sua própria função, você também pode usar a função `numpy.convolve()`.

4. Crie uma função Python para estimar o espectro cruzado `Pxy = esp_cruzado(x, y, win, N_dft)` entre os sinais `x` e `y` usando o método de Welch sem sobreposição (ou seja, $D = N_{DFT}$), e valores padrão de `win=scipy.signal.hann(N_dft)` e `N_dft=4096`. Note que o auto-espectro `Pxx` de um sinal `x` pode ser estimado usando a função `esp_cruzado` com o sinal `x` fornecido duas vezes como entrada.

5. Usando a função `esp_cruzado` para calcular as quantidades apropriadas, compute e plote a coerência entre o sinal de entrada `x` (ruído branco) e a saída do sistema `y`.

6. Crie uma função `est_sistema(x, y, N_dft)` para estimar a função de resposta em frequência $H(\Omega)$ e a resposta ao impulso $h[n]$ do sistema usando o estimador H_1; esta função deve retornar uma tupla `(h, H)` contendo os vetores de resposta ao impulso e resposta em frequência. Compare os resultados com os `H_sys` e `h_sys` originais carregados do arquivo `tutorial10.mat`.

7. Agora vamos analisar o caso de um sistema com ruído aditivo na saída. Gere um sinal de ruído branco `z` do mesmo comprimento que `y`. Multiplique-o por um ganho dado α (digamos, $\alpha = 0.5$) e calcule

$$\tilde{y}[n] = x[n] * h[n] + \alpha \cdot z[n], \tag{8}$$

onde $*$ indica o operador de convolução. Repita a estimativa da FRF e IR e inspecione a função de coerência. Como isso se compara aos resultados obtidos no ponto anterior?

8. Repita os passos 3 e 6 no caso de $h[n] = \delta[n - M]$. Estude a coerência e a precisão do estimador de resposta do sistema nos seguintes casos:

   (a) $M = 200$, $N_{DFT} = 2^{12}$ ($M \ll N_{DFT}$);

   (b) $M = 1000$, $N_{DFT} = 2^{12}$ ($M < N_{DFT}$);

   (c) $M = 6000$, $N_{DFT} = 2^{12}$ ($M > N_{DFT}$).

   Qual efeito o valor de `M` tem no resultado para a coerência? Pense no porquê o resultado deve ser diferente em (c).

9. Modifique a função `esp_cruzado` para permitir que o usuário defina um parâmetro extra `N_sobreposicao` que define o número de pontos de sobreposição na estimação espectral; você pode defini-lo como um parâmetro opcional, com um valor padrão de zero amostras.