1 2 9 0
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

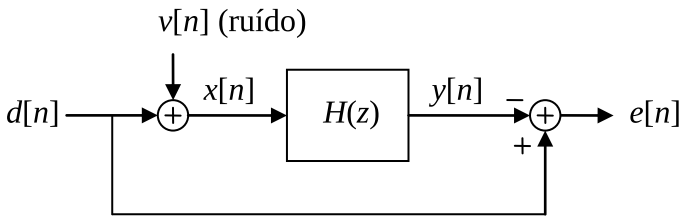
Processamento Digital de Sinal

Trabalho Prático Nº 13

Redução de Ruído por Filtragem Wiener na Frequência - Subtração Espetral

1 - Descrição

Neste trabalho vamos experimentar uma técnica de redução de ruído baseada em filtragem de Wiener no domínio da frequência. A técnica de redução de ruído com filtragem temporal implica a determinação de um filtro ótimo, causal e estável, de acordo com o esquema da figura. A abordagem na frequência permite definir este filtro e a filtragem do sinal no domínio da frequência.



O sinal de entrada do filtro é $x[n] = d[n]+v[n]$, onde $d[n]$ é o sinal desejado, mas é desconhecido porque está embebido no ruído $v[n]$. Apenas $x[n]$ é conhecido. A equação genérica que define o problema é:

$$r_{dx}[k] = \sum_{m=-\infty}^{+\infty} h_m \cdot r_x[k-m] = h[k] * r_x[k]$$

Em termos do espetro de potência, temos

$$P_{dx}(e^{j\omega}) = H(e^{j\omega}) P_x(e^{j\omega}) \Rightarrow H(e^{j\omega}) = \frac{P_{dx}(e^{j\omega})}{P_x(e^{j\omega})}.$$

Neste problema é essencial que $d[n]$ não esteja correlacionado com o ruído, o que é verdade mesmo em situações práticas. Assim, $r_{dx}[k] = E\left\{d[n+k]\left(d[n]+v[n]\right)^*\right\} = r_{dx}[k]+0$, pois $r_{dv}[k]=0$. Também por isso $r_x[k] = r_d[k]+r_v[k]$, e, portanto, $P_x(e^{j\omega}) = P_d(e^{j\omega}) + P_v(e^{j\omega})$. O filtro de Wiener é então dado por:

$$H(e^{j\omega}) = \frac{P_d(e^{j\omega})}{P_x(e^{j\omega})} = \frac{P_d(e^{j\omega})}{P_d(e^{j\omega})+P_v(e^{j\omega})} = \frac{P_d(e^{j\omega})/P_v(e^{j\omega})}{P_d(e^{j\omega})/P_v(e^{j\omega})+1} = \frac{SNR(e^{j\omega})}{SNR(e^{j\omega})+1}$$

onde $SNR(e^{j\omega}) = P_d(e^{j\omega})/P_v(e^{j\omega})$ é a relação entre o espetro de potência do sinal com o ruído. Trata-se de uma relação real e positiva pois tando $P_d(e^{j\omega})$ como $P_v(e^{j\omega})$ são densidades espetrais de potência, logo reais e positivas.

A filtragem consiste então em obter $\boxed{Y(e^{j\omega}) = H(e^{j\omega}) X(e^{j\omega})}$, aplicando um "peso" entre 0 e 1 a $X(e^{j\omega})$, para cada valor de frequência, consoante a relação sinal-ruído a essa frequência. Se $SNR(e^{j\omega})$ for elevada, esse peso será próximo de 1, se for baixa, será próximo de zero.

O problema reduz-se agora à determinação da relação sinal ruído. Para isso é necessário **estimar** a potência do **sinal** $P_d(e^{j\omega})$, que chamaremos $\hat{P}_d(e^{j\omega})$, bem como **estimar** a potência do **ruído**, $\hat{P}_v(e^{j\omega})$. A técnica usual é o método da **Subtração Espetral**:

$$\hat{P}_d(e^{j\omega}) = P_x(e^{j\omega}) - \hat{P}_v(e^{j\omega})$$

Processamento Digital de Sinal
Trabalho Prático Nº 13
Redução de Ruído por Filtragem Wiener na Frequência - Subtração Espetral

A estimação do ruído pode ser conseguida nos intervalos de tempo em que sabemos (assumimos) que $d[n]=0$, e, portanto, $x[n]=v[n]$. No caso de sinais de fala, corresponde aos intervalos onde o locutor faz pausas.

No entanto, existe um problema: $\hat{P}_d(e^{j\omega})$ tem de ser sempre positiva, o que não é garantido na equação anterior com a diferença $P_x(e^{j\omega})-\hat{P}_v(e^{j\omega})$. Utiliza-se então a seguinte técnica:

$$\hat{P}_d(e^{j\omega})=\begin{cases}P_x(e^{j\omega})-\hat{P}_v(e^{j\omega}) & \text{, se } \; P_x(e^{j\omega})-\hat{P}_v(e^{j\omega})>\alpha P_x(e^{j\omega})\\ \alpha P_x(e^{j\omega}) & \text{, se } \; P_x(e^{j\omega})-\hat{P}_v(e^{j\omega})\le \alpha P_x(e^{j\omega})\end{cases}$$

ou

$$\hat{P}_d(e^{j\omega})=\max\left(P_x(e^{j\omega})-\hat{P}_v(e^{j\omega}),\alpha P_x(e^{j\omega})\right), \qquad (1)$$

onde α é uma constante. Por exemplo, se α=0.01 (-40dB), significa que tomamos uma estimativa do sinal que é no mínimo 40 dB inferior à do sinal com ruído.

Mostra-se que fazer $\hat{P}_d(e^{j\omega})=0$ se a diferença for negativa (retificador de meia-onda), conduz a efeitos indesejáveis (ruído musical).

O filtro de Wiener estimado corresponde então à relação:

$$\boxed{H(e^{j\omega})=\frac{\hat{P}_d(e^{j\omega})}{\hat{P}_d(e^{j\omega})+\hat{P}_v(e^{j\omega})}}. \qquad (2)$$

que é sempre real, positivo, e entre 0 e 1. Finalmente, obtém-se o sinal melhorado a partir de:

$$\boxed{Y(e^{j\omega})=H(e^{j\omega})X(e^{j\omega})} \qquad (3)$$

Uma vez que $H(e^{j\omega})$ é real e positivo, $Y(e^{j\omega})$ tem a mesma fase de $X(e^{j\omega})$.

Questões práticas:

1. Cálculo do espetro de potência do sinal $x[n]$, $P_x(e^{j\omega})$: Uma vez que as propriedades do sinal variam no tempo, temos de usar o periodogramas ao longo do tempo. Neste caso usaremos tramas (ou segmentos do sinal) de 30 ms de 10 em 10 ms e estimativas obtidas com o periodograma modificado, usando a janela de Hamming.

Em vez de espetros de potência, vamos usar apenas o módulo das DFTs das tramas para as frequências da DFT, $\omega_k=k\frac{2\pi}{N_{\text{fft}}}$, $k=0..N_{\text{fft}}-1$, onde N_{fft} é o comprimento da DFT. Neste caso basta tomar $N_{\text{fft}}\ge N$ onde N é o comprimento da janela.

2. Uma vez que $H(e^{j\omega})$ é real e positivo, a fase do sinal $Y(e^{j\omega})$ é igual à fase do sinal de $X(e^{j\omega})$. Para calcular $H(e^{j\omega})$ poderíamos fazer os cálculos apenas com o módulo e só depois aplicar a fase na equação (3), mas neste não nos vamos preocupar com questões de eficiência.

Processamento Digital de Sinal
Trabalho Prático Nº 13
Redução de Ruído por Filtragem Wiener na Frequência - Subtração Espetral

3. Síntese do sinal no tempo: usamos o método "overlap-add", isto é a soma com sobreposição das tramas de sinal obtidas a partir da transformada inversa de $Y(e^{j\omega})$. Será instrutivo ver o resultado da soma das janelas com sobreposição (quando $x[n]=1,\forall n$) para ver que esta soma com sobreposição um pequeno ripple mas que o valor DC é o que foi calculado.

2 - Trabalho Prático

2.1 Cálculo Espetral

Leia o sinal fornecido `'PT3.wav'`. Note que existem dois canais de áudio. Processe apenas um ou cada um dos canais separadamente. Considere o seguinte código:

```
[xx,fs]=audioread('PT3.wav');
Nx = size(xx,1)      %duração do sinal em amostras
ncanais = size(xx,2) %dois canais

x = xx(:,1); %canal esquerdo; um canal de cada vez; noutro passo tomar xx(:,2).
N = fix(fs*0.03) %tamanho das tramas
M = fix(fs*0.01) %avanço da janela de análise
Ntramas = fix((Nx-N)/M)+1 %nº de tramas de N amostras com avanço de M
jan = hamming(N); %janela de Hamming (uma coluna)
dcsomajan = sum(jan)/M; %fator ou ganho da soma com sobreposição das janelas.
alfa = 0.01; %-40dB

Nfft = 2^nextpow2(N); %tamanho das DFTs: 2048
X = zeros(Nfft,Ntramas); %espaço para as DFTs, uma por cada coluna
E = zeros(1,Ntramas); %espaço para a energia das tramas
i=1; %início da 1a trama
j=N; %fim da 1a trama
for m=1:Ntramas
    trama = x(i:j).*jan; %a trama m
    Xtrama = fft(trama,Nfft); %a DFT da trama m com Nfft pontos.
    X(:,m) = ???;% guardar a DFT em vez do periodograma.
    E(m) = sqrt(sum(trama.^2)); % raiz quadrada da energia da trama m
    i=i+M; %avançar M amostras para a próxima trama.
    j=j+M; %avançar M amostras para a próxima trama.
end
% mostrar o sonograma do sinal, só acima de alfa dBs:
f=(0:Nfft/2)/Nfft*fs; %vetor de frequências em Hz para o sonograma
figure(1), imagesc(1:Ntramas,f,max(db(X(1:Nfft/2+1,:)),db(alfa))), axis xy
figure(2), plot(1:Ntramas,db(E)), grid on %energia em dB (ou 10*log10(E^2))
```

2.2 Estimativa do ruído

Analise o sonograma e veja se consegue descobrir uma risca horizontal perto dos 10kHz e outra perto dos 8kHz. Verifique também (observando o sinal no tempo) que existe uma forte componente a 50

Hz e outra a cerca de 150 Hz. Pelo gráfico da figura 2 pode observar que a estimação de ruído pode ser efetuada quando a energia das tramas está abaixo de um dado limiar (locutor em pausa). Escolha esse limiar.
Estime o ruído a partir das tramas que verificam essa condição. Isto é, em primeiro lugar determine quais as tramas onde `E` é inferior ao limiar: `i=find(db(E)<limiar)`. Depois faça a média (na 2ª dimensão) de `abs(X(:,i))` (médias das `length(i)` colunas `i`) e chame-lhe `Xv_est`. Faça um plot de `Xv_est`, onde devem ser visíveis os picos nas frequências das riscas identificadas:

```
figure(3)
plot(f,db(Xv_est(1:Nfft/2+1))),grid
```

2.3 Subtração Espetral:

A estimativa do ruído é tomada assumindo que o ruído se mantém ao longo deste sinal. Daí que temos apenas um valor para cada frequência. Em vez de $P_x(e^{j\omega})$ vamos usar $\left|X(e^{j\omega})\right|$:

```
Xx = abs(X); %nos Nfft valores (duplicado de pi a 2*pi: plot(Xx))
Xd_est = max( Xx - Xv_est , alfa*Xx); %subtração espetral (não linear): Eq. 1
figure(2), imagesc(1:Ntramas,f,db(max(Xd_est(1:Nfft/2+1,:),alfa))), axis xy
colorbar
```

2.4 Filtro de Wiener

Calcule o filtro de Wiener (equação (2)) usando `Xd_est` e `Xv_est` em vez dos espetros de potência. Depois aplique a equação (3) a `X` (que tem valores complexos):

```
%filtro de Wiener:
H = Xd_est.^2./(Xd_est.^2 + Xv_est.^2); %com |X|^2; não é necessário dividir por N
size(H)    %deve ter a mesma dimensão de X
isreal(H)  %é real?
Y = H.*???; % espetro da saída (sinal melhorado)
figure(1), imagesc(1:Ntramas,f,max(db(X(1:Nfft/2+1,:)),-60)), axis xy, colorbar
figure(2), imagesc(1:Ntramas,f,max(db(Y(1:Nfft/2+1,:)),-60)), axis xy, colorbar
```

Compare a figura 1 (sonograma original) com a figura 2 (sonograma do sinal melhorado) e verifique a melhoria conseguida. Faça zoom e note que as componentes de baixa frequência foram (porventura demasiado) atenuadas.

2.5 Síntese do sinal

Faça a IDFT das tramas `Y` e divida pela média da soma das janelas:

```
y = real(ifft(Y))/dcsomajan; %tramas para OLA (overlap-add)
size(y) %Nfft valores por trama. Tomar apenas os 1ºs N valores de cada trama.
figure(3), plot(y(:,140)) %Exemplo.
```

Processamento Digital de Sinal
Trabalho Prático Nº 13
Redução de Ruído por Filtragem Wiener na Frequência - Subtração Espetral

Agora basta voltar ao sinal temporal com o método "overlap add - OLA".
Nota: os valores das tramas `y` de `N+1` até `Nfft` vão ser desprezadas.

```
%% overlap-add
yy=zeros(N+(Ntramas-1)*M,1); %espaço suficiente para o resultado
i=1; j=N;
for m=1:Ntramas
    yy(i:j)=yy(i:j)+y(1:N,m); %soma com sobreposição de N-M=882 amostras.
    i=????;
    j=????;
end
soundsc(yy,fs)
figure(3), plot(yy)
%A partir desta figura diga se o sinal foi ou não melhorado.

%Pode experimentar o resultado da soma das janelas com sobreposição
% fazendo no ciclo anterior y=jan (equivale a fazer x[n]=1):
      yy(i:j)=yy(i:j)+jan;
```

Faça um plot do resultado e veja o pormenor das 1ªas janelas a serem sobrepostas e depois o ripple periódico de período M, com valor DC `dcsomajan`: $a_0 = \frac{1}{M} H\left(e^{j0}\right) = \frac{1}{M}\sum_{n=0}^{N-1} h[n]$.

No final da aula entregue o script que produziu, na “Submissão de Trabalhos” do Nónio.