Universidade Federal Rural do Semi-Árido – UFERSA
Centro Multidisciplinar Pau dos Ferros – CMPF
Departamento de Engenharias e Tecnologias – DETEC

Lista de Exercícios nº 02 - Processamento Digital de Sinais
Prof. Pedro Thiago Valério de Souza

■ Problema 1

Um sistema prático e simples para remoção de ruído em sistemas eletrônicos consiste no filtro de média móvel. O filtro de média móvel é descrito pela seguinte equação diferença:

$$y[n] = \frac{1}{M}\sum_{k=0}^{M-1} x[n-k]$$

Em que M é a ordem do filtro de média móvel. Diante disso:

a) Obtenha, de forma analítica, a expressão para a resposta em frequência de um filtro de média móvel de ordem $M = 3$. Plote a resposta em frequência desse filtro em Python ou em Matlab. Analisando a resposta em frequência do sistema, o filtro em questão é de que tipo (Passa-Altas, Passa-Baixas, Passa-Faixa ou Rejeita-Faixa)?

b) Implemente o filtro de média móvel de ordem 10 em Python ou em Matlab. Para testes, gere um sinal de entrada como segue:

$$x[n] = r[n] + \eta[n]$$

sendo:

$$r[n] = \sin(0.1\pi n)$$

um sinal senoidal e $\eta[n]$ um ruído Gaussiano Branco com variância σ^2 e média zero. Apresente os gráficos referentes:

i) ao sinal $r[n]$,

ii) ao sinal de ruído $\eta[n]$,

iii) ao sinal $x[n]$,

iv) o sinal após a filtragem de média móvel.

v) o erro absoluto entre o sinal $r[n]$ e o sinal filtrado, definido como:

$$e[n] = |r[n] - y[n]|$$

vi) O erro quadrático médio, definido como:

$$\text{MSE} = \frac{1}{N}\sum_{n=0}^{N-1} e^2[n]$$

Verifique o desempenho do filtro para valores de variância do ruído de 0.01, 0.05 e 0.1. Considere $0 \leq n \leq 199$. Comente os resultados encontrados.

c) Qual a influência da variância do ruído na performance do filtro?

d) Repita o item (b) para um filtro com ordem 5 e com ordem 50. Qual a influência da ordem do filtro na performance do sistema?

■ Problema 2

O eletrocardiograma (ECG) é um sinal de importância vital para o estudo dos fenômenos bioelétricos. Em particular, o ECG contém informações para o diagnóstico de doenças cardiovasculares, já que esse reflete o comportamento biológico do coração. Os sinais elétricos gerados pelo coração podem ser capturados através de eletrodos posicionados no corpo do paciente. Cada trecho de um ciclo de eletrocardiograma foi classificado de acordo com o formato de cada onda e é dividido em onda P, complexo QRS, onda T e ocasionalmente onda U. A análise de um ECG é baseado em inspeção visual e identificação do complexo QRS, que determina o inicio da contração do ventrículo esquerdo. Todavia, em alguns casos, a análise visual do ECG é complexa, o que levou ao desenvolvimento de algoritmos computacionais que são capazes de detectar o complexo QRS de um ECG. Dentre os algoritmos para detecção do complexo QRS encontra-se o método de Ahlstrom e Tompkins (1983), baseado na primeira e segunda derivada. Sendo $x[n]$ o sinal de ECG, no método de Ahlstrom e Tompkins primeiro calcula-se a primeira derivada retificada:

$$y_0[n] = |x[n+1] - x[n-1]|$$

Após isso, a primeira derivada retificada é suavizada:

$$y_1[n] = \frac{y_0[n-1] + 2y_0[n] + y_0[n+1]}{4}$$

A segunda derivada retificada é calculada:

$$y_2[n] = |x[n+2] - 2x[n] + x[n-2]|$$

A primeira derivada, retificada e suavizada é então somada à segunda derivada retificada:

$$y_3[n] = 2(y_1[n] + y_2[n])$$

O valor máximo de $y_3[n]$ é determinado e utilizado para determinar o limiar primário (λ_1) e secundário (λ_2):

$$\lambda_1 = 0,5 \max\{y_3[n]\}$$

$$\lambda_2 = 0,1 \max\{y_3[n]\}$$

A sequência $y_3[n]$ é percorrida até que seja encontrado uma amostra que excede o limiar primário, que será classificado como um possível complexo QRS. Para que o complexo seja confirmado, as próximas seis consecutivas devem ser maiores ou iguais ao limiar secundário. De forma a minimizar erros devido ao inicio do processamento (enquanto existem valores em transientes resultantes do processo de captura), recomenda-se descartar o primeiro 1,0 s de captura do sinal do ECG de forma a detectar os limiares primários e secundários. Diante disso:

a) Implemente esse algoritmo em MATLAB ou em Python. Considere que o sinal é obtido através de um processo de amostragem com taxa de amostragem de 200 Hz.

b) Verifique o funcionamento do algoritmo implementado. Utilize o sinal `100_norm.mat` ou `100_norm.csv` fornecido em anexo. Como resultado, deseja-se apresentar, em um mesmo gráfico, o sinal de ECG e um outro gráfico, que demarca a posição dos complexos QRS.

■ Problema 3

Em sistemas de radar, deseja-se estimar a distância que um determinado alvo está de uma base. No caso, transmite-se um sinal $x_a(t)$. Esse sinal irá refletir no alvo e retornar ao transmissor após um tempo t_d. A partir da estimação do valor t_d, e conhecendo-se que os sinais eletromagnéticos se propagam na velocidade da luz c, pode-se estimar a distância que o alvo está da base. Seja $x_a(t)$ um sinal transmitido pela base. O sinal recebido, $y_a(t)$, não será composto apenas pelo sinal transmitido atrasado, mas sim uma versão distorcida desse sinal, haja vista que o canal de comunicações sofre com efeito de desvanecimento e ruído. Desta forma, o sinal recebido $y_a(t)$ é dado por:

$$y_a(t) = \alpha x_a(t - t_d) + w_a(t)$$

em que $w_a(t)$ é um ruído branco Gaussiano com média zero e variância σ^2. Deseja-se construir um algoritmo de processamento digital de sinais para estimação da distância do alvo em relação à base. Para isso, os sinais $x_a(t)$ e $y_a(t)$ são amostrados de acordo com o teorema da amostragem:

$$x[n] = x_a(nT)$$

$$y[n] = y_a(nT) = \alpha x_a(nt - t_d) + w_a(nT)$$

Assumindo que o atraso t_d é um múltiplo inteiro do intervalo de amostragem, podemos fazer:

$$y[n] = \alpha x_a(nt - DT) + w_a(nT) = \alpha x_a((n - D)T) + w[n]$$

logo,

$$y[n] = \alpha x[n - D] + w[n]$$

a) Explique como podemos estimar o valor de D utilizando o conceito de correlação cruzada $r_{xy}[l]$.

b) Seja $x[n]$ a sequência de Barker de 13 pontos:

$$x[n] = \{+1, +1, +1, +1, +1, -1, -1, +1, +1, -1, +1, -1, +1\}$$

e $w[n]$ um ruído branco Gaussiano com média zero e variância $\sigma^2 = 0,01$. Escreva um código em Python ou Matlab para gerar a sequência $y[n]$, $0 \leq n \leq 199$ para $\alpha = 0,9$ e $D = 20$. Plote os sinais $x[n]$ e $y[n]$ com $0 \leq n \leq 199$.

c) Calcule e plote a correlação cruzada $r_{xy}[l]$. Utilize as funções próprias do Python ou Matlab para isso. Utilize o gráfico para estimar o valor de D.

d) Repita os itens (b) e (c) para $\sigma^2 = 0,1$ e $\sigma^2 = 1$. Comente seus resultados.

■ Problema 4

Muitos geradores de sinal utilizam técnicas de processamento digital de sinais para a geração de sinais analógicos. Uma das técnicas consiste na criação de um sinal senoidal digital que é posteriormente convertido em um sinal analógico através de um conversor D/A. Uma forma simples para criar um oscilador digital consiste em criar uma sistema cuja a resposta ao impulso é o sinal senoidal desejado, ou seja:

$$h[n] = A \sin(\omega_0 n + \phi) u[n]$$

sendo A a amplitude do sinal, ω_0 a frequência angular do sinal em rad/amostra, ϕ a fase inicial do sinal em radianos e $u[n]$ a função degrau unitário. Diante disso:

a) Determine a função de transferência $H(z)$ do sistema cuja resposta ao impulso é dada por $h[n]$ e descreva a equação de diferenças que relaciona a entrada e a saída do sistema.

b) Determine a expressão que relaciona a frequência angular ω_0 do sinal senoidal com a frequência de amostragem f_s (Hz) do conversor D/A e a frequência f_0 (Hz) do sinal senoidal analógico. Calcule o valor de ω_0 para um sinal senoidal com frequência de 1 kHz e uma frequência de amostragem de 10 kHz.

c) Implemente o sistema projetado em Python ou MATLAB. Considere $A = 1$, $\phi = 0$, $F_s = 10$ kHz e $f_0 = 1$ kHz. Gere o sinal de saída do sistema para $0 \leq t \leq 0,01$ s e plote o gráfico do sinal gerado.

d) Uma forma simples para a geração de uma onda quadrada a partir de uma onda senoidal é através da utilização de um comparador. O comparador gera uma onda quadrada a partir de uma onda senoidal através da seguinte regra: se o valor da onda senoidal for maior ou igual a zero, a saída do comparador é 1; caso contrário, a saída do comparador é -1. Implemente o comparador em Python ou MATLAB e gere o sinal de saída do comparador utilizando o sinal senoidal gerado no item (c) como entrada. Plote o gráfico do sinal gerado pelo comparador.

e) Uma forma simples para a geração de uma onda dente de serra a partir de uma onda senoidal é através da utilização de um integrador, ou seja:

$$y_{\text{trig}}(t) = \int_{0^-}^{t} x_{\text{sq}}(\tau)dt$$

sendo $x_{\text{sq}}(t)$ o sinal de onda quadrada e $y_{\text{trig}}(t)$ o sinal de onda dente de serra. Utilizando uma aproximação adequada para a operação de integração em tempo discreto, implemente o integrador em Python ou MATLAB e gere o sinal de saída do integrador utilizando o sinal de onda quadrada gerado no item (d) como entrada. Plote o gráfico do sinal gerado pelo integrador. *Dica:* utilizar a aproximação do trapézio pode ser uma boa alternativa para a implementação do integrador em tempo discreto.

■ PROBLEMA 5

Demonstre que se a função de transferência de um sistema de tempo contínuo é dada por:

$$H_a(s) = \frac{b_M s^M + b_{M-1}s^{M-1} + \cdots + b_0}{a_N s^N + a_{N-1}s^{N-1} + \cdots + a_0} = K\frac{\prod_{i=1}^{M}(s - z_i)}{\prod_{i=1}^{N}(s - p_i)}$$

sendo M o número de zeros finitos, N o número de polos, $\{z_i\}$ os zeros e $\{p_i\}$ os polos do sistema de tempo contínuo, então a função de transferência do filtro digital $H(z)$ obtida através da transformação bilinear é dada por:

$$H(z) = H_a\left(\frac{2}{T}\frac{z-1}{z+1}\right) = \left(K\frac{\prod_{i=1}^{M}\frac{2}{T} - z_i}{\prod_{i=1}^{N}\frac{2}{T} - p_i}\right)\frac{\prod_{i=1}^{M} z - \frac{1+z_iT/2}{1-z_iT/2}}{\prod_{i=1}^{N} z - \frac{1+p_iT/2}{1-p_iT/2}}(z+1)^{N-M}$$

ou seja, um polo em $s = p_i$ é mapeado em $z = \frac{1+p_iT/2}{1-p_iT/2}$, um zero em $s = z_i$ é mapeado em $z = \frac{1+z_iT/2}{1-z_iT/2}$ e serão adicionados $(N - M)$ zeros em $z = -1$.

Utilizando essa demonstração, escreva uma função em Python ou MATLAB que receba como entrada os coeficientes do numerador e denominador da função de transferência de um sistema de tempo contínuo e retorne os coeficientes do numerador e denominador da função de transferência do sistema de tempo discreto obtido através da transformação bilinear sem compensão de *warping*. Teste sua função utilizando a seguinte função de transferência de tempo contínuo:

$$H_a(s) = \frac{1}{s^2 + \sqrt{2}s + 1}$$

Considere uma frequência de amostragem de 10 Hz. Apresente os coeficientes do numerador e denominador da função de transferência do sistema discreto obtido. Compare seus resultados com aqueles obtidos utilizando as funções nativas em Python ou MATLAB para discretização de sistemas de tempo contínuo.

■ PROBLEMA 6

Escreva uma função em Python ou MATLAB para projetar filtros digitais utilizando a transformação bilinear com compensação de *warping*. Como entrada, sua função deve ter:

- Tipo da resposta em frequência do filtro a ser projetado (Passa-Baixas, Passa-Altas, Passa-Faixa ou Rejeita-Faixa).
- Resposta em frequência do filtro a ser projetado (Butterworth ou Chebyshev).
- Intervalo de amostragem T (s).
- Frequências de passagem $\{\Omega_p\}$ (rad/s).
- Frequências de rejeição $\{\Omega_s\}$ (rad/s).
- Atenuação máxima na banda de passagem α_p (dB).
- Atenuação mínima na banda de rejeição α_s (dB).
- Para filtros de Butterworth, um argumento que indica se os requisitos de projeto serão atendidos na banda de passagem, na banda de rejeição ou através da média aritmética simples entre as duas bandas.

Seu algoritmo deve retornar os coeficientes do numerador e denominador da função de transferência do filtro digital projetado. *Atenção:* Não é permitido o uso de funções nativas para o projeto de filtros digitais. Deve-se calcular os polos do filtro analógico e utilizar a relação existente entre os polos do filtro analógico e os polos do filtro digital obtido através da transformação bilinear. Utilizando a função desenvolvida, projete os seguintes filtros:

a) Filtro passa-baixas Butterworth que permita a passagem de sinais com até 5 rad/s com atenuação máxima de 1 dB e rejeite sinais acima de 25 rad/s com atenuação mínima de 30 dB. Atenda aos requisitos da banda de rejeição. O sistema processa os sinais utilizando um A/D e D/A operando com $T = 0,01$ s.

b) Filtro passa-baixas Chebyshev que permita a passagem de sinais com até 100 rad/s com atenuação máxima de 1 dB e rejeite sinais acima de 200 rad/s com atenuação mínima de 22 dB. O sistema processa os sinais utilizando um A/D e D/A operando com $T = 10^{-3}$ s.

c) Filtro passa-altas de Butterworth que permita a passagem de sinais acima de 20 rad/s com atenuação máxima de 1 dB e rejeite sinais abaixo de 10 rad/s com atenuação mínima de 20 dB. O sistema processa os sinais utilizando um A/D e D/A operando com $T = 0,05$ s. Atenda aos requisitos da banda de passagem.

d) Filtro passa-faixas de Chebyshev que permita a passagem de sinais entre 50 rad/s e 150 rad/s com atenuação máxima de 1 dB e rejeite sinais abaixo de 30 rad/s e acima de 200 rad/s com atenuação mínima de 25 dB. O sistema processa os sinais utilizando um A/D e D/A operando com $T = 10^{-3}$ s.

e) Filtro rejeita-faixas de Butterworth que rejeite sinais entre 40 rad/s e 80 rad/s com atenuação mínima de 20 dB e permita a passagem de sinais abaixo de 20 rad/s e acima de 100 rad/s com atenuação máxima de 1 dB. O sistema processa os sinais utilizando um A/D e D/A operando com $T = 0,01$ s. Atenda aos requisitos da banda de rejeição.

Para cada um dos itens, determine a função de transferência do filtro digital projetado e plote o gráfico da resposta em magnitude do filtro em dB.

■ Problema 7

O uso de vuvuzelas em estádios de futebol tornou-se popular após a Copa do Mundo de 2010, realizada na África do Sul. Todavia, o som produzido por esses instrumentos musicais é considerado por muitos como desagradável e perturbador. O som característico da vuvuzela é um sinal quase periódico com uma frequência fundamental em torno de 235 Hz e harmônicos pares e ímpares.

Deseja-se projetar um filtro digital para atenuar o som da vuvuzela captado pelo microfone. Isso pode ser feito através da utilização de um filtro rejeita-faixas com uma largura de banda bastante estreita que atue sobre a frequência fundamental e seus harmônicos. O nome desse tipo de filtro é *notch filter*, cuja função de transferência é dada por:

$$H(z) = \frac{1 - 2\cos(\omega_0)z^{-1} + z^{-2}}{1 - 2r\cos(\omega_0)z^{-1} + r^2 z^{-2}}$$

sendo ω_0 a frequência angular (em rad/amostra) a ser atenuada e r um parâmetro que controla a largura de banda do filtro. Uma forma pratica para determinar r é descrita em [Rahmatillah, 2017]:

$$r = 1 - \frac{f_{BW}}{f_S}\pi$$

em que f_{BW} é a largura de banda do filtro *Notch* e f_S e a frequência de amostragem. Suponha que o sistema de captura do som da vuvuzela opere com uma frequência de amostragem de 44,1 kHz. Além disso, considere que serão eliminadas a frequência fundamental e os primeiros 3 harmônicos da vuvuzela, utilizando uma largura de banda de 80 Hz para cada filtro *Notch* projetado. Projete um sistema de filtragem digital para atenuar o som da vuvuzela captado pelo microfone, seguindo os passos abaixo:

i) Determine os valores de ω_0 e r para cada um dos filtros *Notch* a serem projetados.

ii) Determine a função de transferência do sistema de filtragem digital projetado.

iii) Implemente o sistema de filtragem digital projetado em Python ou Matlab. Considere que o sinal de entrada do sistema é o arquivo de áudio `vuvuzela.mp3` fornecido em anexo.

iv) Salve o sinal filtrado em um arquivo de áudio denominado `vuvuzela_filtrado.mp3`. Ouça o arquivo gerado e comente sobre a qualidade do som após a filtragem.

■ Problema 8

Em sistemas de telecomunicações, os sinais de informação são degradados por diversos efeitos degenerativos do canal. Um desses efeitos é o desvanecimento de múltiplos percursos. Nesse caso, o sinal recebido $r[n]$ é composto pelo um somatório de componentes defasadas do sinal transmitido $x[n]$. Suponha um canal de comunicações em que o sinal recebido é composto por uma componente principal e duas versões defasadas de $x[n]$ da seguinte forma:

$$r[n] = x[n] + ax[n-1] + bx[n-2]$$

Diante disso, faça o que se pede em cada um dos itens:

a) Implemente esse canal de comunicações no MATLAB ou no Python, considerando $a = 1/2$ e $b = 1/4$. Como testes, gere um sinal aleatório $x[n]$ com 50 amostras (utilizando a função `rand`). Apresente, em uma mesma janela gráfica, o sinal $x[n]$, o sinal $r[n]$ e o sinal de erro absoluto $e[n]$, calculado como:

$$e[n] = |x[n] - r[n]|$$

Calcule também o erro quadrático médio, definido como:

$$\text{MSE} = \frac{1}{N}\sum_{n=0}^{N-1} e^2[n]$$

Comente os resultados encontrados.

b) Obtenha a função de sistema de um equalizador para o canal descrito no item (a), ou seja, um sistema que a partir do sinal $r[n]$ consiga estimar o sinal $x[n]$ (obtendo $\tilde{x}[n]$). Implemente o equalizador projetado no item no MATLAB ou no Python. Como testes, aplique o sinal gerado no item (a) no equalizador projetado. Apresente, em uma mesma janela gráfica, os gráficos referentes o sinal $x[n]$ e a sequência equalizada, denominada de $\tilde{x}[n]$. Também apresente o sinal de erro absoluto e o valor do erro quadrático médio após a equalização.

c) Qual é a condição para a função de transferência do canal de comunicações de tal forma que seja possível construir um equalizador estável e causal?

■ PROBLEMA 9

Sendo $x[n]$ o sinal de áudio original com N amostras não nulas e $y[n]$ o sinal de áudio processado, implemente os seguintes efeitos de áudio em Python ou em MATLAB:

a) Reversão temporal:

$$y[n] = x[N - n + 1] \qquad 0 \leq n \leq N - 1$$

b) Subamostragem por um fator de 2 (aumento da velocidade de reprodução por 2):

$$y[n] = x[2n] \qquad 0 \leq n \leq \frac{N-1}{2}$$

c) Sobreamostragem por um fator de 2 (diminuição da velocidade de reprodução por 2):

$$y[n] = \begin{cases} x[n/2] & n = 0, 2, 4, \ldots \\ 0 & \text{c.c.} \end{cases}$$

d) *Delay*:

$$y[n] = x[n] + \eta x[n - N]$$

em que η é a profundidade do efeito ($|\eta| < 1$) e N é a quantidade de amostras de atraso, calculada como:

$$N = F_s(\Delta t)$$

tal que F_s é a frequência de amostragem do sinal de áudio e Δt é o atraso desejado (em segundos).

e) *Overdrive*:

$$y[n] = \begin{cases} 2x[n] & -1/3 \leq x[n] < 1/3 \\ -\dfrac{3-(2-3x[n])^2}{3} & -2/3 \leq x[n] < -1/3 \\ \dfrac{3-(2-3x[n])^2}{3} & 1/3 \leq x[n] < 2/3 \\ 1 & x[n] \geq 2/3 \\ -1 & x[n] < -2/3 \end{cases}$$

f) *Tremolo*:

$$y[n] = x[n]\left[1 + \eta \cos\left(2\pi \frac{F_x}{F_s} n\right)\right]$$

em que F_x é a frequência do efeito, em Hz (teste alguns valores próximos à 5 Hz) e η é a profundidade do efeito ($0 < \eta < 1$).

g) *Fuzz*:

$$y_e[n] = \frac{x[n]}{|x[n]|}\left[1 - \exp\left(\frac{ax^2[n]}{|x[n]|}\right)\right]$$

$$y[n] = \eta y_e[n] + (1-\eta)x[n]$$

em que a é o ganho (teste alguns valores próximos à 10) e η é a profundidade do efeito ($0 < \eta < 1$).

Verifique os resultados dos efeitos de forma visual (plote o gráfico no tempo) e de forma auditiva. Os efeitos deverão ser testados em cima do áudio `guitar.wav`.

■ Problema 10

Muitos computadores e calculadoras calculam a raiz quadrada de um número positivo A através do seguinte algoritmo recursivo:

$$y[n] = \frac{1}{2}\left(y[n-1] + \frac{x[n]}{y[n-1]}\right)$$

em que $x[n] = Au[n]$, sendo A o valor no qual deseja-se estimar a raiz quadrada e $y[-1]$ é uma estimativa inicial para a raiz quadrada de A. O algoritmo é resolvido de forma recursiva para $n \geq 0$. Após uma quantidade satisfatórias de iterações, $y[n]$ tenderá ao valor real de $\sqrt{A}$. Implemente esse algoritmo em Python ou Matlab. Verifique seus resultados para os seguintes casos, em que N é a quantidade de iterações realizadas pelo algoritmo. Como estimativa inicial, utilize $y[-1] = A/2$.

a) $A = 5,\ N = 10$

b) $A = 21,\ N = 25$

c) $A = 21,\ N = 4$

d) $A = 121,\ N = 30$

Comente seus resultados, mostrando o gráfico referente a $y[n]$ e calculando o erro absolutoentre o valor real de $\sqrt{A}$ e o seu valor estimado pelo algoritmo recursivo.