

# Filtros Digitais

Déric Augusto F. de Sales - 96718  
Departamento de Engenharia Elétrica,  
Universidade Federal de Viçosa, Viçosa - MG.  
Email: deric.sales@ufv.br

## 1. INTRODUÇÃO

A filtragem consiste no processo de geração de uma saída  $y[n]$  a partir de uma entrada  $x[n]$ , aplicando um sistema linear invariante no tempo [1]. Na filtragem é possível mudar as amplitudes relativas dos componentes em frequência de um sinal ou até eliminar por completo alguns componentes em frequência [2]. Um projeto pode inclusive partir da especificação dos níveis de oscilação permitidos na banda de passagem (Rp em decibéis) ou da largura da banda de transição (Wp - Ws, em Hertz) [1].

Os sistemas lineares invariantes no tempo que mudam a forma do espectro são conhecidos como filtros conformadores de frequência, e aqueles que são projetados para passar algumas frequências essencialmente não distorcidas e que atenuam significativamente ou eliminam outras são chamados de filtros seletivos em frequência [2].

Os filtros digitais podem ser implementados de forma eficiente em sistemas digitais específicos ou de propósito geral, sendo estes filtros de tempo discreto. Os sistemas lineares de tempo discreto, assim, podem ser recursivos, com repostas ao impulso de duração infinita (sistemas IIR) ou podem ser não recursivos, apresentando respostas ao impulso de duração finita (sistemas FIR). [2].

Os filtros FIR são obtidos truncando-se a resposta ao impulso do filtro (ideal) pretendido. Possuem, entre outras vantagens a fase linear, serem sempre estáveis e poderem ser implementados facilmente em um hardware [1]. Já os filtros IR conseguem alcançar especificações de projeto com ordem bem menor, e entre os mais utilizados estão o *Butterworth*, *Chebyshev* e *Elíptico*.

Para a análise de sinais os filtros digitais são interessantes já que, ao invés de um componente físico como uma placa de circuito impresso que filtra o sinal diretamente a partir da fonte, o filtro digital será representado por uma equação matemática em um sistema digital, como um computador ou microprocessador. Assim, é possível a coleta do sinal diretamente da fonte e a realização de sua filtragem já dentro desses dispositivos digitais.

## 2. OBJETIVOS

É objeto de estudo do trabalho a aplicação de filtros FIR e IIR em sinais gerados artificialmente e em sinais de áudio, analisando o comportamento e efetividade dessas ferramentas.

## 3. MATERIAIS E MÉTODOS

Assim como nas demais práticas da disciplina, o software MATLAB R202a [3] será utilizado para a análise matemática dos sinais, bem como para a criação de algoritmos e geração das figuras que serão aqui apresentadas. Dentro do software é possível a aplicação de diversos filtros digitais que já estão implementados por padrão através de funções, como as *fir1*, *filtfilt*, *cheby2*, *cherblord*, *filter*, *butter*, *butter*, *ellys*, *cheb2ord*, *kaiserord*, *kaiserord*, *cheby1*, *buttord* e *ellipord*. Cada função dessa possui características específicas de filtragem.

A função nativa *fir1* aplica o filtro *FIR* (resposta ao impulso finita) linear e estável. Essa função pode ser aplicada em filtros passa-baixas, altas, faixa e rejeita-faixa. Ela baseia-se no método de janelamento Hamming. A função *kaiserord*, por sua vez, estima parâmetros otimizados para a função *fir1* de forma a atender uma série de especificações na filtragem. Sua implementação no *software* MATLAB pode ser feita através da relação (1).

$$[n, w_n, beta, ftype] = kaiserord(f, a, dev) \quad (1)$$

A função descrita retorna um filtro de ordem  $n$ , com bordas da banda de frequência normalizadas  $w_n$  e um fator  $beta$  que especifica o janelamento a ser usado na função *fir1*. A variável  $f$  indica as bordas da banda,  $a$  a amplitude da banda e  $dev$  o desvio máximo permitido dado em forma de um vetor. A função *filtfilt*, por sua vez, é implementada na presença de um filtro *IIR* (resposta ao impulso finita). Como o filtro *IIR* possui características não-lineares, faz-se necessário o uso da função *filtfilt*, que permite a realização de amostragens não causais, além de apresentar fase zero, o que elimina as distorções na fase deste filtro.

O filtro *Butterworth* é conhecido devido a sua alta taxa de inclinação, sendo sua Função de Transferência a 2, onde  $n$  é a ordem do filtro,  $w$  sua frequência angular em radianos por segundo e  $w_c$  a frequência de corte. Nesse filtro, quanto maior a ordem utilizada, maior será a taxa de inclinação a partir de sua frequência de corte. Pode ser implementado no MATLAB através da relação (3), sendo  $w_n$  dado em forma vetorial e indicado de acordo com o tipo de filtro a ser aplicado pelo *ftype*.

$$H_n(jw) = \frac{1}{\sqrt{1 + (\frac{w}{w_c})^{2n}}} \quad (2)$$

$$[b,a] = \text{butter}(n, w_n, ftype) \quad (3)$$

Outra opção para a aplicação do filtro *Butterworth*, é através da Função 4, que retorna sua ordem  $n$  mais baixa, onde  $W_p$  e  $W_s$  são respectivamente as frequências de canto da banda de passagem e a banda de parada normalizadas entre 0 e 1, onde 1 representa  $\pi$  radianos por amostra.  $R_p$  é o *ripple* da banda passante,  $R_s$  a atenuação da banda de parada e ' $s$ ' especifica o tipo de filtro.

$$[n, w_n] = \text{buttord}(W_p, W_s, R_p, R_s, 's') \quad (4)$$

Outra opção de filtro disponível é o *Chebyshev*, definido conforme a Função de Transferência 5. No MATLAB o tipo 1, definido pela Função 6, possui um aumento na atenuação e um maior *ripple* na banda passante em comparação com os filtros *Butterworth*. Na Função de Transferência,  $n$  é a ordem do filtro,  $\epsilon$  é o parâmetro de ondulação da banda de passagem,  $w$  a frequência angular em radianos por segundo,  $w_o$  a frequência de corte e  $T_n\left(\frac{w}{w_o}\right)$  é um polinômio de *Chebyshev*. Há também as funções 7, que retorna a ordem  $n$  mais baixa do filtro *Chebyshev* tipo 1, a Função 8, onde  $R_s$  é a atenuação da banda de parada expressa em decibéis e  $W_s$  é a frequência de borda de banda de parada.

$$H_n(jw) = \frac{1}{\sqrt{1 + \epsilon^2 \cdot T_n^2 \cdot \frac{w}{w_o}}} \quad (5)$$

$$[b,a] = \text{cheby1}(n, R_p, w_p, ftype) \quad (6)$$

$$[n, W_p] = \text{cheb1ord}(W_p, W_s, R_p, R_s, 's') \quad (7)$$

$$[a,b] = \text{cheby2}(n, R_s, W_s) \quad (8)$$

$$[n, W_p] = \text{cheb2ord}(W_p, W_s, R_p, R_s, 's') \quad (9)$$

Outra opção de filtro é o elíptico, conhecido por apresentar *ripples* na banda passante e na banda rejeitada com o erro minimizado. Sua Função de Transferência pode ser conferida em (10). As Funções nativas do MATLAB para aplicação desse filtro são 11 e 12, sendo que na primeira é definida a ordem do filtro e a segunda retorna a ordem  $n$  mais baixa do filtro. Mais informações sobre estas e as demais funções podem ser conferidas através do comando *help* dentro do MATLAB [3].

$$H_n(jw) = \frac{1}{\sqrt{1 + (\epsilon \cdot R_n \cdot w)^2}} \quad (10)$$

$$[b,a] = \text{ellip}(n, R_p, R_s, w_p, ftype) \quad (11)$$

$$[n, W_p] = \text{ellipord}(W_p, W_s, R_p, R_s, 's') \quad (12)$$

#### 4. RESULTADOS

Após a realização de todos os testes demandados em [1] implementando as tarefas por meio de algoritmo no MATLAB, foram gerados os seguintes resultados.

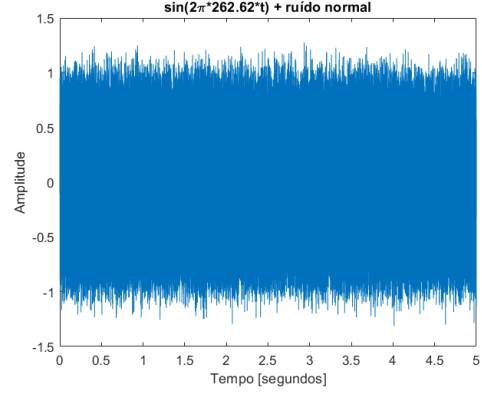


Figura 1: Senoide ruidosa gerada artificialmente.

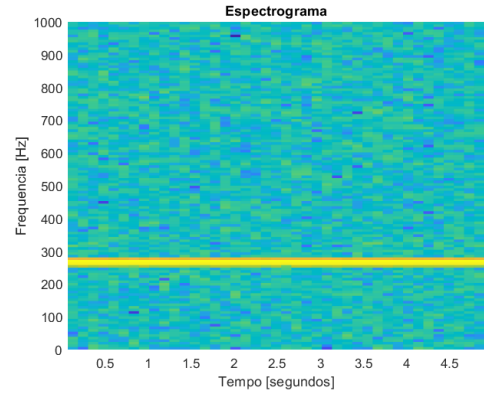


Figura 2: Espectrograma da senoide ruidosa da Figura 1.

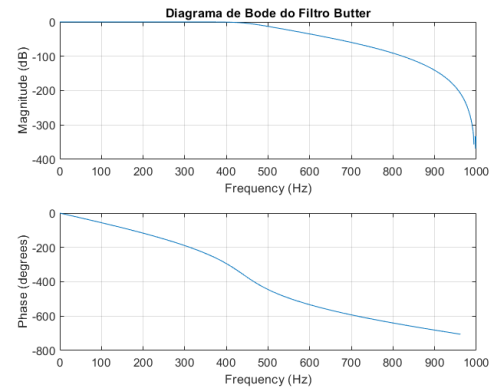


Figura 3: Diagrama de Bode do filtro *Butterworth* aplicado à senoide ruidosa da Figura 1.

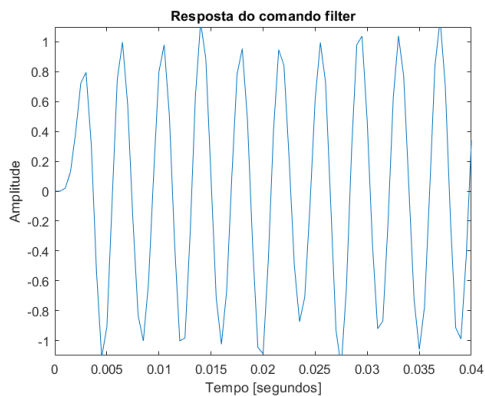


Figura 4: Sinal filtrado usando o comando *filter*.

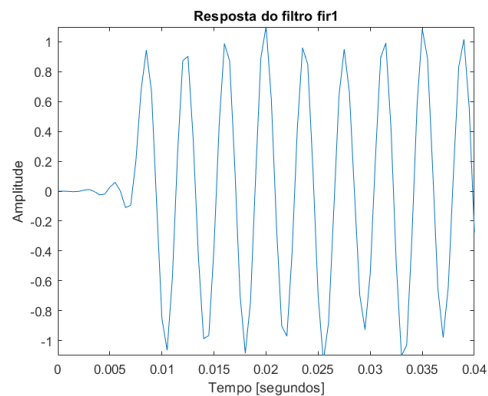


Figura 7: Sinal filtrado com o filtro *fir1*

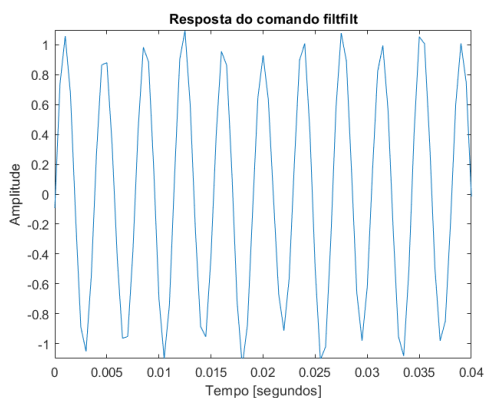


Figura 5: Sinal filtrado usando o comando *firlfilt*.

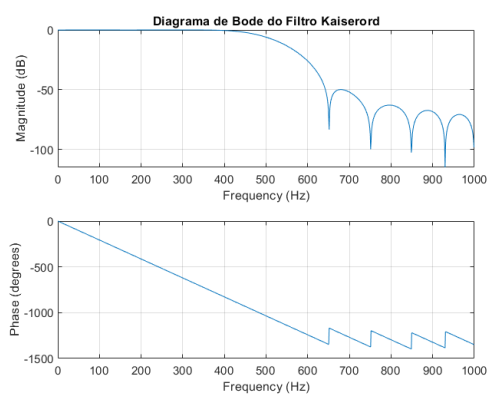


Figura 8: Filtro FIR implementado usando a função *kaiserord*.

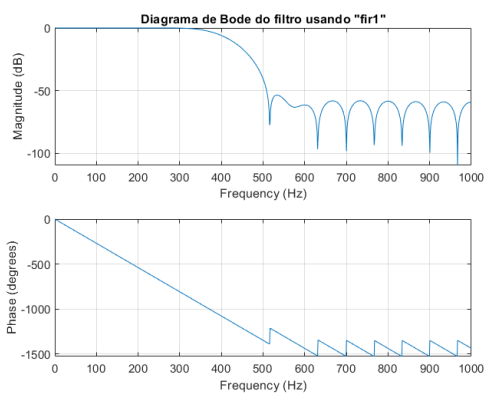


Figura 6: Diagrama de Bode do filtro implementado com a função *fir1*.

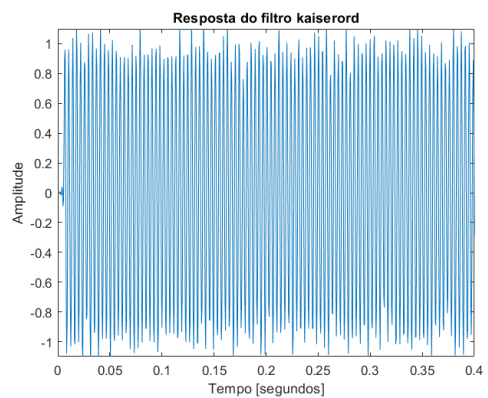


Figura 9: Sinal filtrado usando o filtro *Kaiserord*.

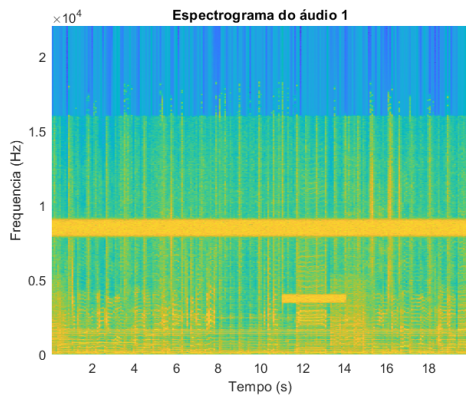


Figura 10: Espectrograma do áudio 1.

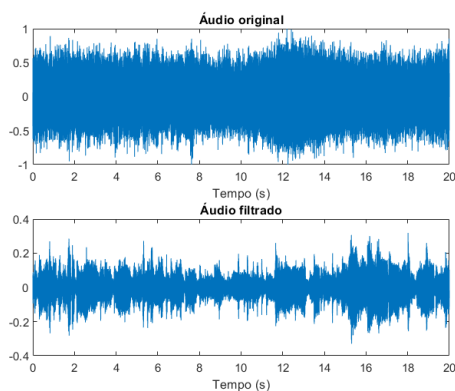


Figura 11: Áudio 1 original e após a filtragem, respectivamente

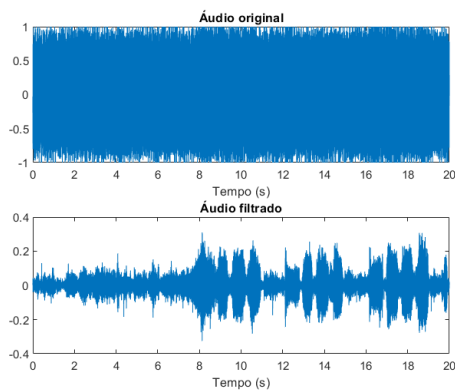


Figura 12: Áudio 2 original e após a filtragem, respectivamente

## 5. DISCUSSÕES

No primeiro exercício proposto em [1], realizou-se a filtragem de um sinal senoidal com ruído utilizando-se, a partir da análise de seu espectrograma, um filtro passa-baixas com frequência de corte de 400 Hz, disponível nas Figuras 1, 2 e 3. Utilizou-se para tal as funções *butterd* e

*butter* já descritas. A filtragem também foi implementada através dos filtros *kaiserord* e *fir1*. Com a utilização do comando *kaiserord* foi possível observar uma otimização da ordem do *FIR* equivalente ao *IIR* anteriormente usado, como se pode verificar nas figuras 7 e 9. É possível observar na figura 6 que o filtro *FIR* apresenta algumas oscilações tanto na sua magnitude quanto em sua fase e, com o uso da função *kaiserord* essas oscilações diminuem, como consta na figura 8.

No segundo exercício proposto, foram analisados dois arquivos de áudio contendo duas músicas com diferentes tipos de ruídos. Esse fato foi analisado pelo espectrograma dos áudios nos quais interpretou-se as componentes de frequência de maior magnitude como sendo as responsáveis pelos ruídos. Em relação ao primeiro áudio, assim como disponibilizado na Figura 10, foram identificados 5 ruídos diferentes que foram removidos através de 5 filtro rejeita-faixa. Após a filtragem completa, é possível conferir o resultado na Figura 11.

No segundo arquivo de áudio foi realizado um procedimento similar, sendo que dessa vez foram identificados ruídos na faixa das frequências de 4 e 11 kHz. O resultado da filtragem pode ser conferido na Figura 12. O ruído do áudio dois não foi completamente removido, apenas atenuado, já que se tratava do som de pessoas conversando numa faixa de frequência que se sobrepunha às frequências mais importantes da música, e se fosse removido, apagaria também essas características da música.

## 6. CONSIDERAÇÕES FINAIS

A prática em questão possibilitou a análise das aplicações dos filtros digitais como ferramenta de filtragem de sinais. Comparados aos filtros analógicos, os filtros digitais mostram-se mais vantajosos pois não dependem de características de componentes, consegue-se facilmente implementar filtros de ordem elevada, são programáveis via software, além de possuírem uma banda larga com transições agudas. Assim, pode-se concluir que este recurso matemático pode ser amplamente utilizado na análise do comportamento de diferentes tipos de sinais, como os de som [2].

## REFERÊNCIAS

- [1] Dr. Leonardo Bonato Felix. *ELT 355 - Sinais e Sistemas - Roteiro de Aula Prática 12 – Filtros Digitais*. CCE, DEL, Universidade Federal de Viçosa.
- [2] Alan V Oppenheim, Alan S. Willsky, Syed Hamid Nawab, Gloria Mata Hernandez, et al. *Signals Systems*. Pearson Educacion, 1997.
- [3] MathWorks. *Math. Graphics. Programming. Disponível em: [https://www.mathworks.com/products/matlab.html?s\\_tid=hp\\_products\\_matlab](https://www.mathworks.com/products/matlab.html?s_tid=hp_products_matlab)*.