

Processamento Digital de Sinais

Outline:

- Filtros Digitais
- Caracterização de Filtros Digitais
- Metodologia de Projeto de Filtros Digitais
- Filtros FIR – *Finite Impulse Response*
- Filtros de Fase Linear
- Projeto de filtros FIR pelo método de Janelamento: Janela Retangular, Janela de Hamming, Janela de Blackman, Janela de Kaiser
- CAD para projeto de filtros a partir das especificações
- Especificações para projeto de filtros FIR
- Filtros IIR – *Infinite Impulse Response*
- Filtros analógicos que servem de base para o projeto de filtros digitais IIR: Butterworth, Chebyshev e Aproximação inversa de Chebyshev.
- Filtros FIR vs Filtros IIR
- Forma Direta – Filtros IIR
- Forma Direta – Filtros FIR

Filtros Digitais

- Os **Filtros Digitais** que estudaremos têm todas as propriedades de **Sistemas Lineares de Tempo Discreto**.
- Filtros Digitais são importantes em DSP porque **todas as operações lineares em tempo discreto realizadas em uma sequência de entrada podem ser vistas como uma filtragem da sequência de entrada, para produzir uma sequência de saída.**
- Sistemas lineares são caracterizados por suas **respostas ao impulso**.
- **A resposta ao impulso de um filtro pode ter duração finita (filtro FIR) ou infinita (filtro IIR).**

Caracterização de Filtros Digitais

- Uma resposta ao impulso finita $h(n)$ tem valores não nulos sobre um intervalo de tempo finito, e zero fora deste intervalo.

A resposta ao impulso finita

$$h(n) = \{h_0, h_1, h_2, \dots, h_N, 0, 0, 0, \dots\}$$

tem valores não nulos no intervalo

$$0 \leq n \leq N$$

e é referida como um filtro de resposta ao impulso finita (*Finite Impulse Response - FIR*) de ordem N .

- Um filtro digital FIR de $N^{\text{ésima}}$ ordem tem uma resposta ao impulso com duração de $(N + 1)$ amostras.
- As amostras da função resposta ao impulso (h_0, h_1 , etc) são chamadas **coeficientes do filtro** ou **pesos do filtro**.

Caracterização de Filtros Digitais

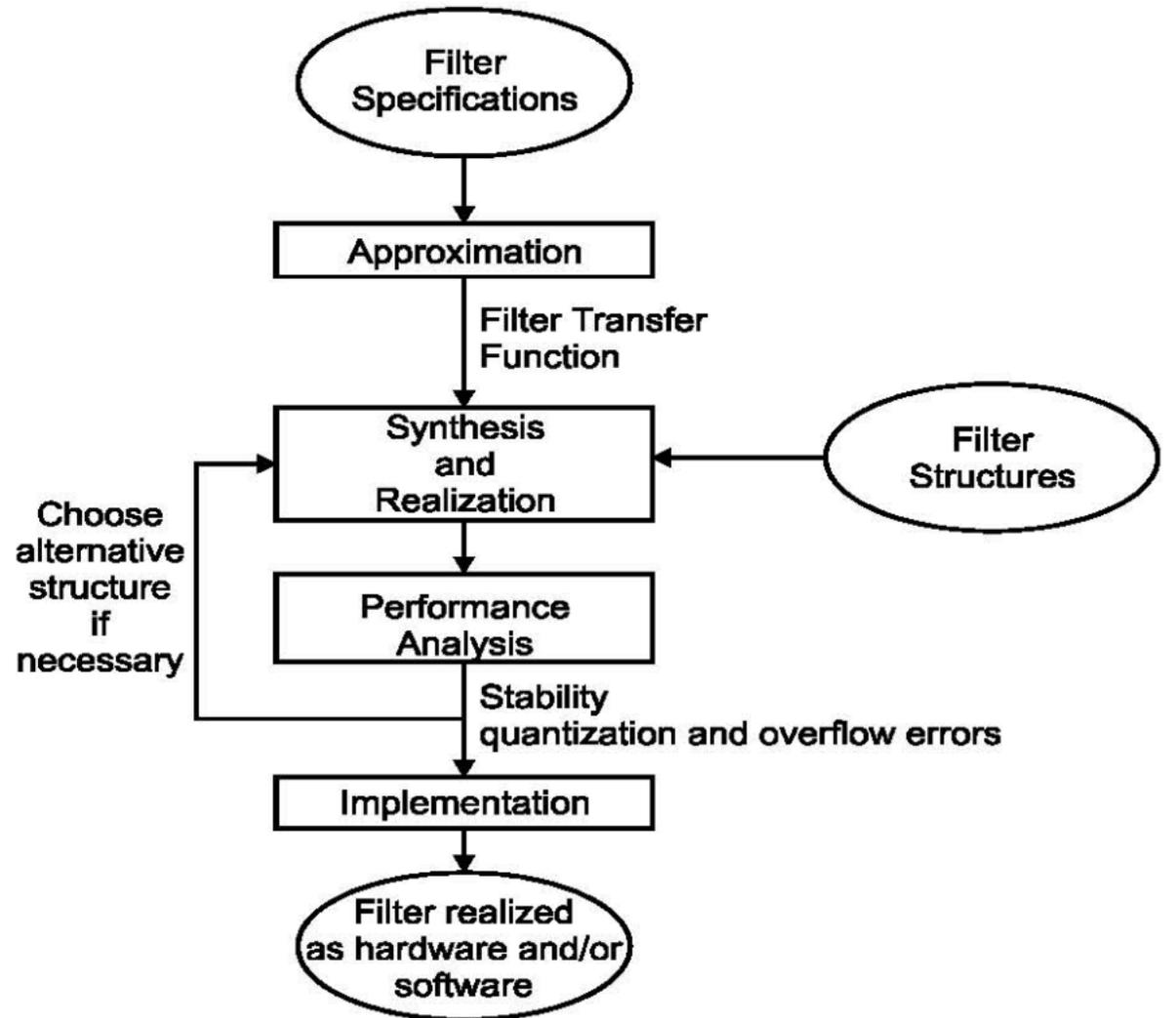
- Se a resposta ao impulso de um filtro tem duração infinita, o filtro é denominado filtro de resposta ao impulso infinita (*Infinite Impulse Response* - IIR).
- Devido ao número infinito de termos na resposta ao impulso, o uso de Filtros IIR em determinadas aplicações pode envolver alguma dificuldade computacional.
- Os filtros IIR de interesse prático relacionam entrada/saída através de uma equação de diferença linear em que a sequência de saída é computada recursivamente.
- Por esta razão os filtros IIR são também conhecidos como filtros recursivos e filtros FIR são conhecidos como filtros não recursivos.

Metodologia de Projeto de Filtros Digitais

O processo geral de projeto de um filtro digital pode ser dividido em cinco etapas principais:

- Especificação
- Aproximação
- Síntese e realização
- Análise de desempenho
- Implementação

Estas etapas são ilustradas em forma de fluxograma na figura ao lado.

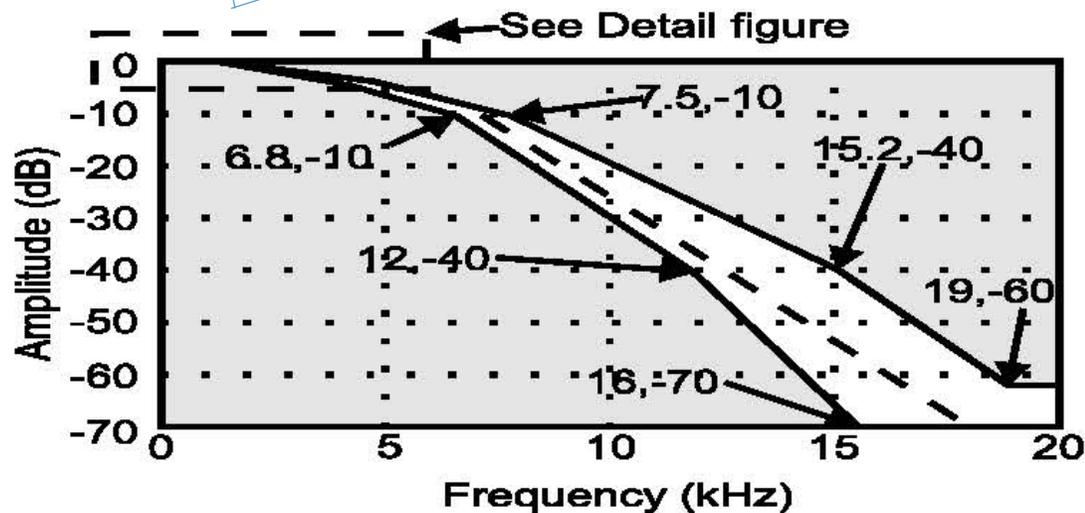
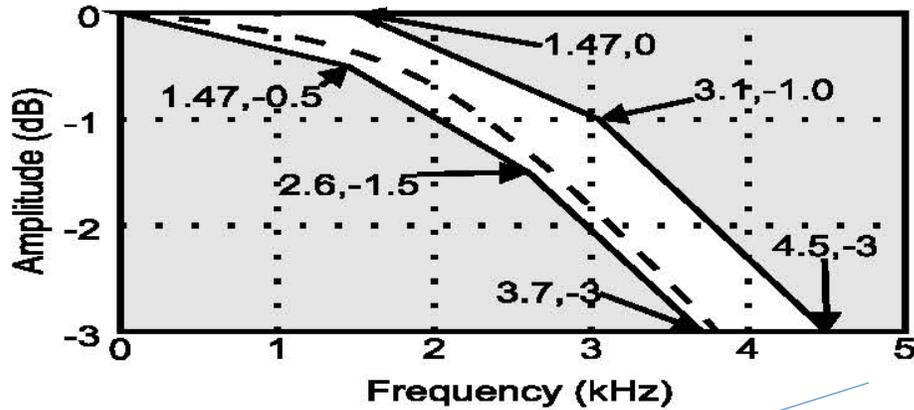


Metodologia de Projeto de Filtros Digitais - Especificação

Especificação:

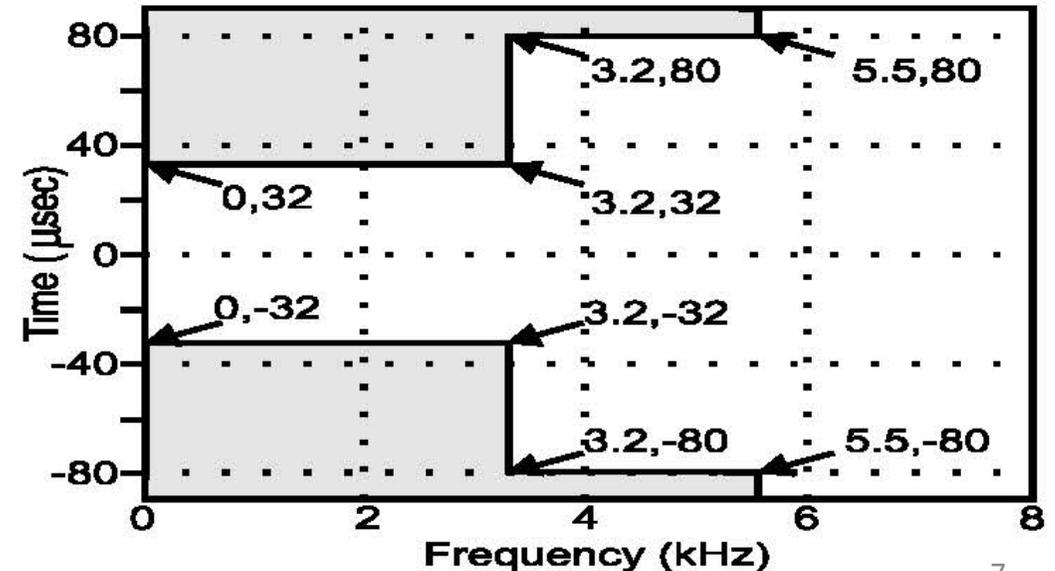
- O projeto começa com as especificações e requisitos do filtro, que estão intimamente relacionados com a aplicação em questão.
- Estas especificações incluem **características desejadas no domínio frequência**, tais como magnitude e fase da resposta em frequência do filtro.
- As especificações também podem incluir **requisitos no domínio tempo**, como atraso máximo, por exemplo.
- A maioria das especificações define os limites superior e inferior para cada uma dessas características. Alternativamente, uma resposta desejada ou ideal pode ser dada com a quantidade máxima de desvios a partir do ideal especificado, por exemplo, frequência de corte $10\text{MHz} \pm 5\%$.

Metodologia de Projeto de Filtros Digitais - Especificação



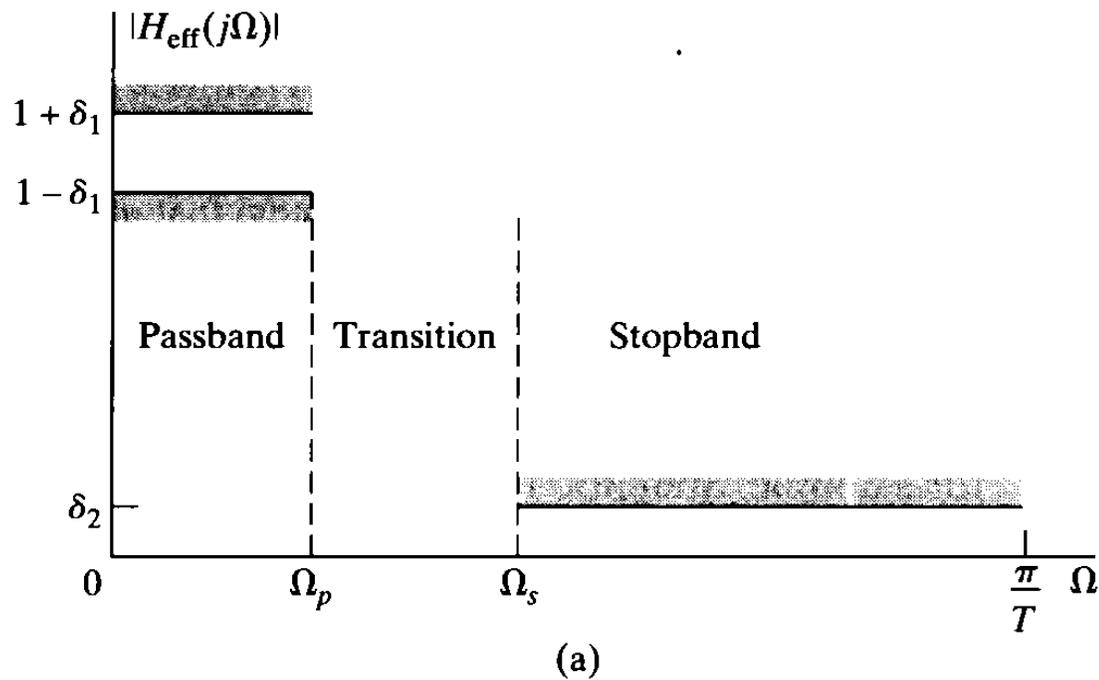
As figuras apresentam um exemplo de especificação para um filtro passa-baixa:

- máscara delimitando o **módulo da função de transferência** em dB ($20\log|H(f)|$) e
- máscara delimitando o **atraso de grupo** em μs ($\frac{d}{df} \arg\{H(f)\}$).

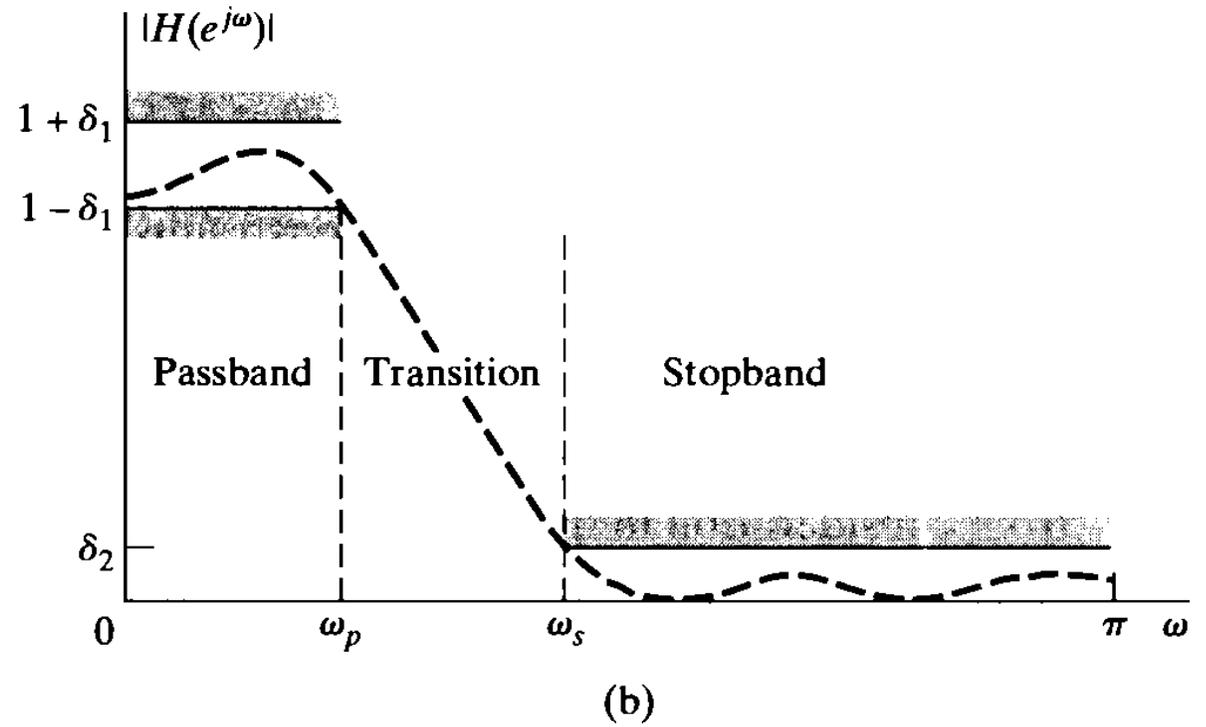


Metodologia de Projeto de Filtros Digitais - Especificação

Outra forma de especificação:



Máscara para a resposta em frequência de um sistema analógico.



Máscara correspondente para o sistema analógico da figura à esquerda, amostrado para o domínio de tempo discreto (digitalizado).

Metodologia de Projeto de Filtros Digitais - Aproximação

Aproximação

- Dadas as especificações do filtro, a primeira etapa do projeto é encontrar uma função de transferência que irá satisfazer essas especificações.
- Este processo é chamado aproximação. É assim chamado porque o que estamos fazendo é, de fato, encontrar uma função de transferência que se aproxima da resposta ideal especificada.
- Os métodos para resolver o problema de aproximação para filtros digitais podem ser classificados como método direto e método indireto.
- O método direto trata a problema da aproximação no domínio tempo discreto e frequência discreta.
- O método indireto trata a problema da aproximação no domínio contínuo. Uma função de transferência no domínio contínuo é obtida via métodos de projeto de filtros analógicos. Esta função de transferência é então transformada em uma função de transferência de tempo discreto.
- Métodos indiretos são mais comumente usados para filtros IIR, enquanto os métodos de projeto de filtro FIR compreendem principalmente os métodos diretos.

Aproximação

- Métodos para o projeto de filtros também podem ser classificados como fechados ou iterativos.
- Os métodos fechados utilizam fórmulas fechadas e geralmente contam com um número determinado (fixo) de etapas.
- Os métodos iterativos fazem uso de técnicas de otimização que partem de uma solução inicial que é refinada progressivamente até que os critérios pré-determinados de desempenho sejam satisfeitos. O número de iterações depende da solução inicial e da eficácia das técnicas de otimização empregadas.

Síntese e Realização

- Uma vez determinada a função de transferência, ela deve ser realizada em uma rede linear (conjunto de blocos lineares) em tempo discreto.
- Este procedimento é análogo ao procedimento de realização de um filtro analógico, em que a topologia do circuito e os valores dos elementos do circuito (indutâncias e capacitâncias) são escolhidos de acordo com a função de transferência de filtro que se deseja implementar.
- Vários métodos de realização podem ser considerados, e a melhor realização de uma dada função de transferência depende muito da aplicação a que se destina.

Síntese e Realização

- Em geral, a realização deve definir o número necessário de somadores e multiplicadores, e a sensibilidade de cada bloco da rede com relação à precisão numérica (desvio da especificação original por truncamento numérico, *overflow* e *underflow*).
- Por exemplo, se o sinal tem uma grande excursão de amplitude na entrada de um determinado bloco, então é conveniente usar aritmética de ponto flutuante na realização daquele bloco particular.
- Caso o sinal tenha uma pequena excursão de amplitude, não é necessário usar aritmética de ponto flutuante, sendo preferível usar aritmética inteira de ponto fixo, de menor complexidade computacional, para efeito de economizar recursos do *hardware*.

Análise de Desempenho

- Mesmo que os coeficientes do filtro sejam determinados com um alto grau de precisão no processo de aproximação, o *hardware* digital que implementa o filtro em tempo real tem precisão numérica finita, dado que os registradores de qualquer *hardware* têm um número finito de bits.
- Para um registrador com um determinado número de bits, a precisão da operação numérica executada dependerá do tipo de aritmética usada: ponto fixo ou ponto flutuante, o que se aplica particularmente para aritmética de ponto fixo em função do maior erro de quantização numérica em relação à aritmética de ponto flutuante, conforme comentado no *slide* anterior.
- O projeto deve garantir que o erro de quantização (também denominado ruído de quantização) introduzido pela precisão finita não comprometerá as especificações do filtro.
- Além disso, os efeitos numéricos de *overflow* e *underflow* devem ser examinados.

Análise de Desempenho

- Não é demais enfatizar a importância dessa etapa do projeto, especialmente para filtros IIR.
- Ao contrário dos filtros FIR, que são incondicionalmente estáveis (resposta ao impulso se extingue p/ um tempo transcorrido suficientemente grande), **os filtros IIR podem apresentar instabilidade pelo erro de quantização em cada variável numérica** (registrador do *hardware*) no processo computacional da filtragem IIR.
- O ruído de quantização auto-excita o filtro, gerando um sinal espúrio na saída, mesmo quando o sinal aplicado na entrada é de pequena amplitude. Este sinal espúrio é recursivamente realimentado no processo de filtragem IIR, aumentando sua amplitude até a saturação dos registradores do *hardware* (*overflow*), o que caracteriza a operação instável do filtro.

Implementação

- Filtros digitais podem ser implementados em *software* ou *hardware* ou em uma combinação de ambos.
- Implementações em *software* implicam em uma prévia decisão sobre o tipo de microprocessador em que o *software* será executado.
- Chips que são projetados especificamente para operações de DSP são muito eficazes, em função da aritmética de ponto flutuante neles intrinsecamente disponibilizada.
- A facilidade de desenvolvimento do *software* depende muito da qualidade das ferramentas de desenvolvimento. Embora o desempenho de alguns chips para DSP possa ser semelhante, a qualidade das ferramentas disponíveis pode ser muito diferente. Algumas ferramentas de *software* são desenvolvidas pelos próprios fabricantes de chips para DSP.
- Em aplicações muito demandantes, o filtro pode necessitar ser implementado em lógica programável (FPGA), que possibilita a paralelização de operações, ou em circuito integrado de aplicação específica (ASIC), para efeito de obter a velocidade necessária.

Filtros FIR – *Finite Impulse Response*

 Filtros FIR são sistemas lineares de tempo discreto, cuja sequência de saída está relacionada com a sequência de entrada e a resposta ao impulso do filtro pela operação de convolução, conforme

$$y(n) = \sum_{m=0}^M x(m)h(n-m) \quad (1)$$

- A equação (1) indica que qualquer amostra de saída específica depende apenas das $M + 1$ amostras de entrada para um filtro $M^{\text{ésima}}$ ordem.
- Os filtros FIR também são conhecidos como filtros não-recursivos.
- Observe também que a soma do lado direito da equação (1) é uma convolução entre a sequência de entrada $x(n)$, e a resposta ao impulso do filtro $h(n)$. Por esta razão, filtros FIR são muitas vezes chamados **filtros de convolução**.
- Do ponto de vista estatístico, a amostra de saída de um filtro FIR é a média ponderada dos $M + 1$ valores das amostras de entrada. Por esta razão, filtros FIR são muitas vezes chamados **filtros de média móvel** (*Moving Average*).

Filtros FIR – *Finite Impulse Response*



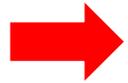
Uma das principais vantagens dos filtros FIR é a facilidade com que filtros de fase linear podem ser projetados, ao contrário dos filtros IIR .

- A característica de fase linear é importante no **processamento digital de sinais de áudio em alta fidelidade**. Um filtro cuja curva de fase $\arg\{H(f)\}$ é não-linear gera distorção audível, denominada *artifact*. Nas frequências f em que a curva $\arg\{H(f)\}$ apresenta uma variação abrupta em consequência da sua não-linearidade, o atraso de grupo $\frac{d}{df} \arg\{H(f)\}$ será grande. Isto faz com que grupos de frequências do espectro do sinal de áudio próximas às frequências f em que a curva de $\arg\{H(f)\}$ apresenta variação abrupta serão abruptamente deslocadas no tempo obedecendo ao atraso de grupo, e este deslocamento abrupto no tempo de determinadas frequências audíveis é a causa do *artifact* no sinal de áudio .
- De mesma forma, um filtro com fase linear processa de maneira uniforme as componentes espectrais do sinal digital, mantendo a proporcionalidade da fase em cada frequência do espectro do sinal de saída. Esta característica é crucial em **sistemas de comunicações digitais de banda larga**, como sistemas OFDM (*orthogonal frequency division multiplex*), por exemplo, sistemas WiFi 802.11g, 802.11n e 802.11ac.

Filtros FIR – *Finite Impulse Response*

- ➔ Outra importante propriedade dos filtros FIR é a garantia de estabilidade, porque não há realimentação da saída para a entrada (como acontece com os filtros IIR).
- ➔ Além disso, filtros FIR podem ser eficientemente implementados em processadores, General Purpose Processors – GPP (explo.: ARM), Digital Signal Processors (explo.: Texas) e em lógica programável – FPGAs (explo.: Xilinx). A maioria dos processadores de DSP têm instruções específicas no seu conjunto de instruções de forma a facilitar a implementação de filtros FIR (e IIR).

Filtros FIR – *Finite Impulse Response*



A resposta em frequência de um filtro FIR de ordem N é dada por

$$H(\omega) = \sum_{n=0}^{N-1} h(n) e^{-j\omega n}$$

onde ω é dado em radianos por segundo.

- Note que, estritamente falando, o expoente deveria ser $(-j\omega Tn)$, onde T é o período de amostragem. Podemos, então, interpretar a equação acima como válida para um período normalizado de $T = 1s$.
- Observe que, embora o filtro seja um sistema de tempo discreto, a variável ω no domínio frequência é contínua e periódica, com período 2π .
- Se determinarmos a resposta em frequência $H(\omega)$ de um filtro usando a DFT ao invés da DTFT, é importante lembrar que para uma resposta ao impulso de duração N , a DFT nos dará N pontos discretos no domínio frequência. Para N pequeno, a resposta ao impulso original $h(n)$ pode necessitar ser completada com zeros (*zero padding*), a fim de se obter uma curva de resposta em frequência $H(\omega)$ com mais pontos e, portanto, aumentar a precisão da representação.

Filtros FIR – *Finite Impulse Response*

Conforme já visto anteriormente no Capítulo III das notas de aula, a resposta em frequência de um sistema digital é geralmente representada por valores complexos, consistindo em valores de magnitude e fase que são definidos por

$$H(\omega) = A(\omega)e^{j\theta(\omega)}$$

onde $A(\omega)$ é a resposta de amplitude e $\theta(\omega)$ é a resposta de fase do sistema, sendo a magnitude da resposta em frequência do sistema dada por

$$M(\omega) = |H(\omega)| = |A(\omega)|$$

 A DFT da resposta ao impulso $h(n)$ de duração N de um filtro é definida conforme

$$C(k) = \sum_{n=0}^{N-1} h(n)e^{-j2\pi kn/N} \quad k = 0, 1, \dots, N-1$$

Filtros FIR - Exemplo

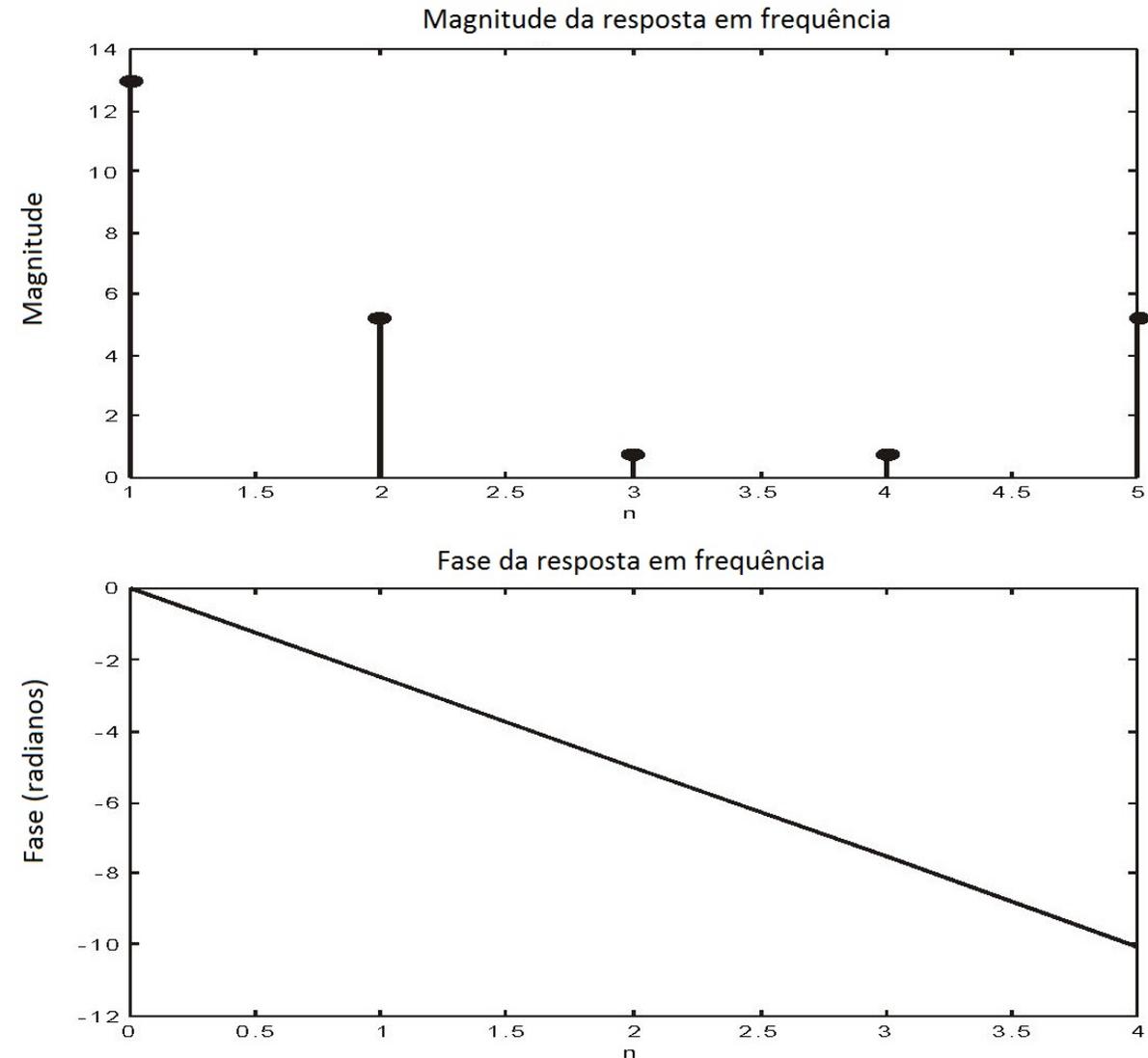
Considere um filtro FIR com resposta ao impulso dada por

$$h(n) = \{1, 3, 5, 3, 1\}$$

Aplicando a DFT em $h(n)$ obtém-se a resposta (discreta) em frequência do filtro:

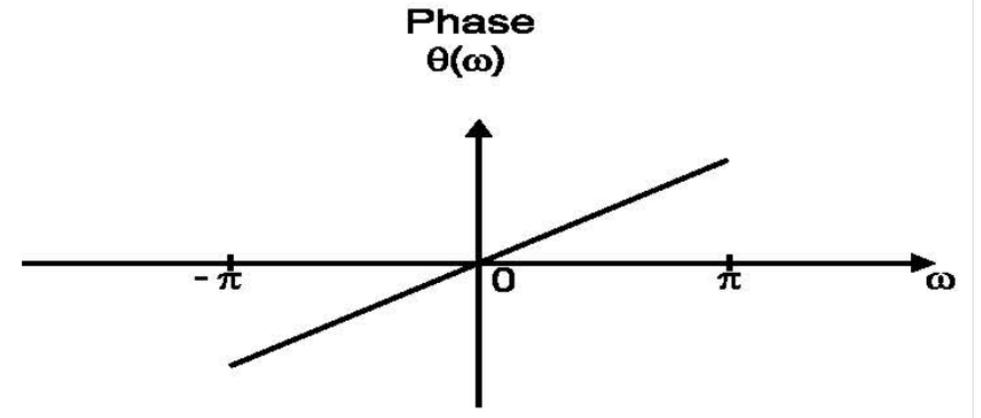
$$C(k) = \left\{ \begin{array}{l} 13.00, \\ -4.2361 - j3.0777, \\ 0.2361 + j0.7256, \\ 0.2361 - j0.7256, \\ -4.2361 + j3.0777 \end{array} \right\}$$

A magnitude e a fase da resposta em frequência são mostradas nas figuras ao lado. A fase, embora discreta, foi plotada de forma contínua para ressaltar a linearidade da curva de fase.



Filtros de Fase Linear

- Fase linear refere-se à resposta de fase sendo uma função linear da frequência ω , conforme mostra a figura ao lado.
- Outra maneira de dizer que um filtro com resposta em frequência $H(f)$ apresenta resposta de fase $\arg\{H(f)\}$ linear é dizer que o mesmo tem um atraso de grupo constante ao longo de f , lembrando que o atraso de grupo é a derivada em relação à f da resposta de fase $\arg\{H(f)\}$ de um filtro.



$$\theta(\omega) = \arg\{H(f)\}, \quad \omega = 2\pi f$$

- Pode-se demonstrar matematicamente que um filtro digital FIR possui fase $\arg\{H(f)\}$ linear se a sua resposta ao impulso for simétrica (com simetria par) ou anti-simétrica (com simetria ímpar) em relação ao ponto central do eixo tempo discreto “ n ”.
- Dado que a duração da resposta ao impulso de um filtro digital pode ter um número N de amostras que pode ser par ou ímpar, existem no total quatro tipos de filtros FIR de fase linear, conforme veremos a seguir.

Filtros de Fase Linear

Filtro de Fase Linear Tipo 1:

Para este filtro, o gráfico da resposta ao impulso tem duração de N amostras, sendo N ímpar, apresentando simetria par em relação ao ponto central do eixo tempo discreto “ n ”, isto é, $h(n) = h(N - n - 1)$.

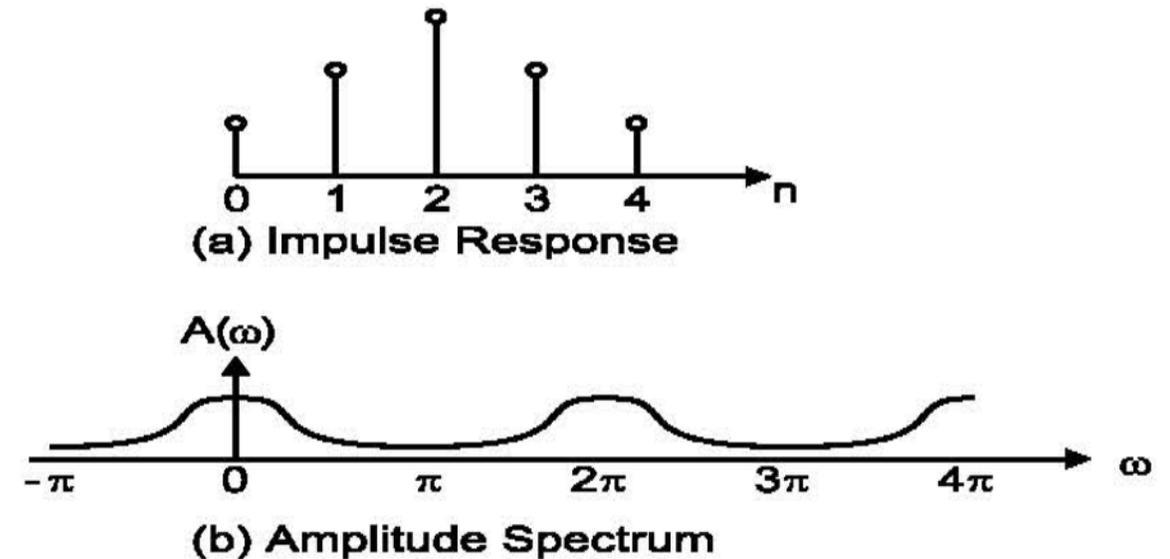
O gráfico da amplitude $A(\omega)$ da resposta em frequência neste caso resulta com simetria par em relação a $\omega = 0$ e em relação a $\omega = \pi$. A amplitude $A(\omega)$ também resulta periódica, com período 2π . Especificamente:

$$A(\omega) = A(-\omega)$$

$$A(\pi + \omega) = A(\pi - \omega)$$

$$A(\omega + 2\pi) = A(\omega)$$

A Figura ao lado mostra a resposta ao impulso e a amplitude $A(\omega)$ da resposta em frequência de um filtro FIR de fase linear tipo 1.



Filtros de Fase Linear

Filtro de Fase Linear Tipo 2:

Para este filtro, o gráfico da resposta ao impulso tem duração de N amostras, sendo N par, apresentando simetria par em relação o ponto central do eixo tempo discreto “ n ”. Note que, neste caso, o ponto central não corresponde ao instante de uma amostra “ n ”, pois N é par.

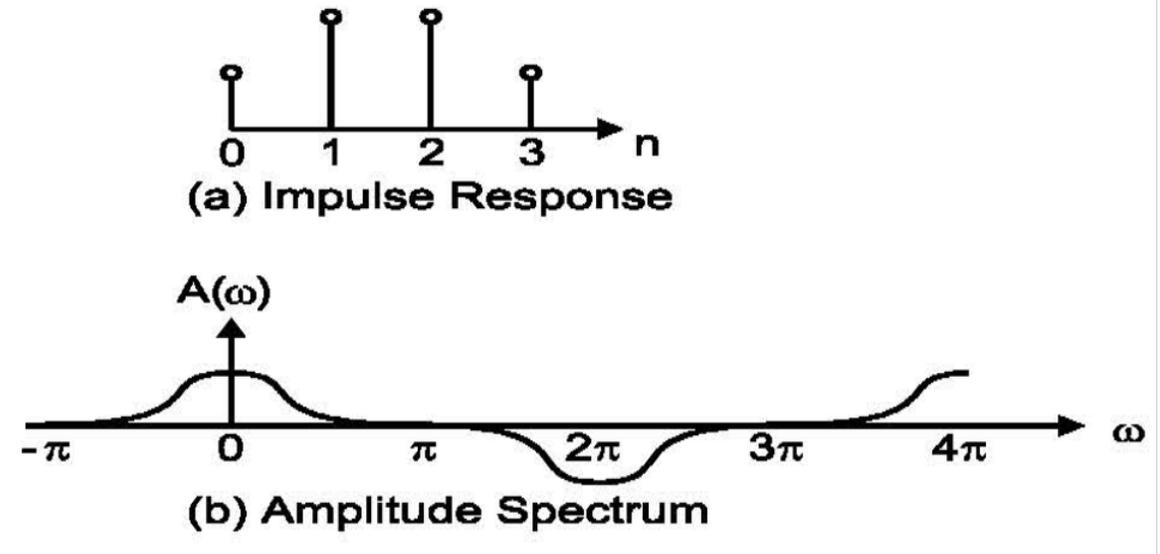
O gráfico da amplitude $A(\omega)$ da resposta em frequência neste caso resulta com simetria par em relação a $\omega = 0$ e resulta com simetria ímpar em relação a $\omega = \pi$. A amplitude $A(\omega)$ também resulta periódica, com período 4π ao invés de 2π . Especificamente:

$$A(\omega) = A(-\omega)$$

$$A(\pi + \omega) = -A(\pi - \omega)$$

$$A(\omega + 4\pi) = A(\omega)$$

A amplitude $A(\omega)$ da resposta em frequência deste tipo de filtro é zero em $\omega = \pi$, sendo, portanto, adequada para filtros passa-baixa, mas inadequada para filtros passa-alta.



Filtros de Fase Linear

Filtro de Fase Linear Tipo 3:

Para este filtro, o gráfico da resposta ao impulso tem duração de N amostras, sendo N ímpar, apresentando simetria ímpar em relação ao ponto central do eixo tempo discreto “ n ”.

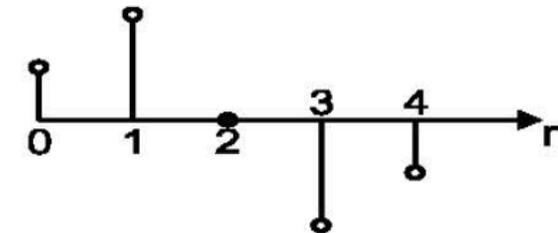
O gráfico da amplitude $A(\omega)$ da resposta em frequência neste caso resulta com simetria ímpar em relação a $\omega = 0$ e em relação a $\omega = \pi$. A amplitude $A(\omega)$ também resulta periódica, com período 2π . Especificamente:

$$A(\omega) = -A(-\omega)$$

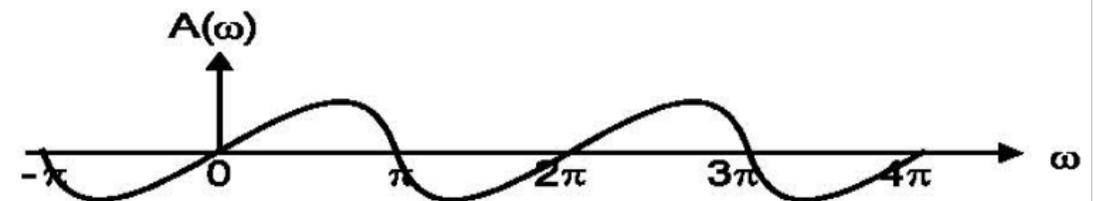
$$A(\pi + \omega) = -A(\pi - \omega)$$

$$A(\omega + 2\pi) = A(\omega)$$

A amplitude $A(\omega)$ da resposta em frequência deste tipo de filtro é zero em $\omega = 0$ e em $\omega = \pi$, sendo, portanto, inadequada para filtros passa-baixa como também inadequada para filtros passa-alta.



(a) Impulse Response



(b) Amplitude Spectrum

Filtros de Fase Linear

Filtro de Fase Linear Tipo 4:

Para este filtro, o gráfico da resposta ao impulso tem duração de N amostras, sendo N par, apresentando simetria ímpar em relação ao ponto central do eixo tempo discreto “ n ”.

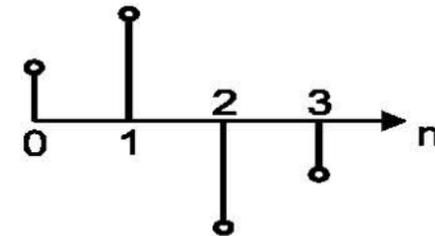
O gráfico da amplitude $A(\omega)$ da resposta em frequência neste caso resulta com simetria ímpar em relação a $\omega = 0$ e resulta com simetria par em relação a $\omega = \pi$. A amplitude $A(\omega)$ também resulta periódica, com período 4π .

$$A(\omega) = -A(-\omega)$$

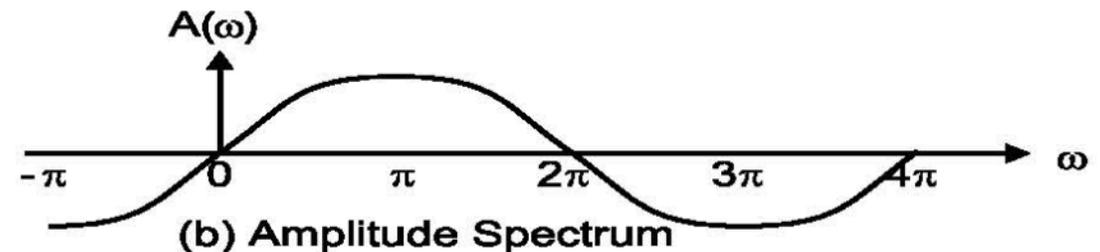
$$A(\pi + \omega) = A(\pi - \omega)$$

$$A(\omega + 4\pi) = A(\omega)$$

A amplitude $A(\omega)$ da resposta em frequência deste tipo de filtro é zero em $\omega = 0$ mas não é zero em $\omega = \pi$, sendo, portanto, inadequada para filtros passa-baixa. No entanto, é adequada para filtros passa-alta.



(a) Impulse Response



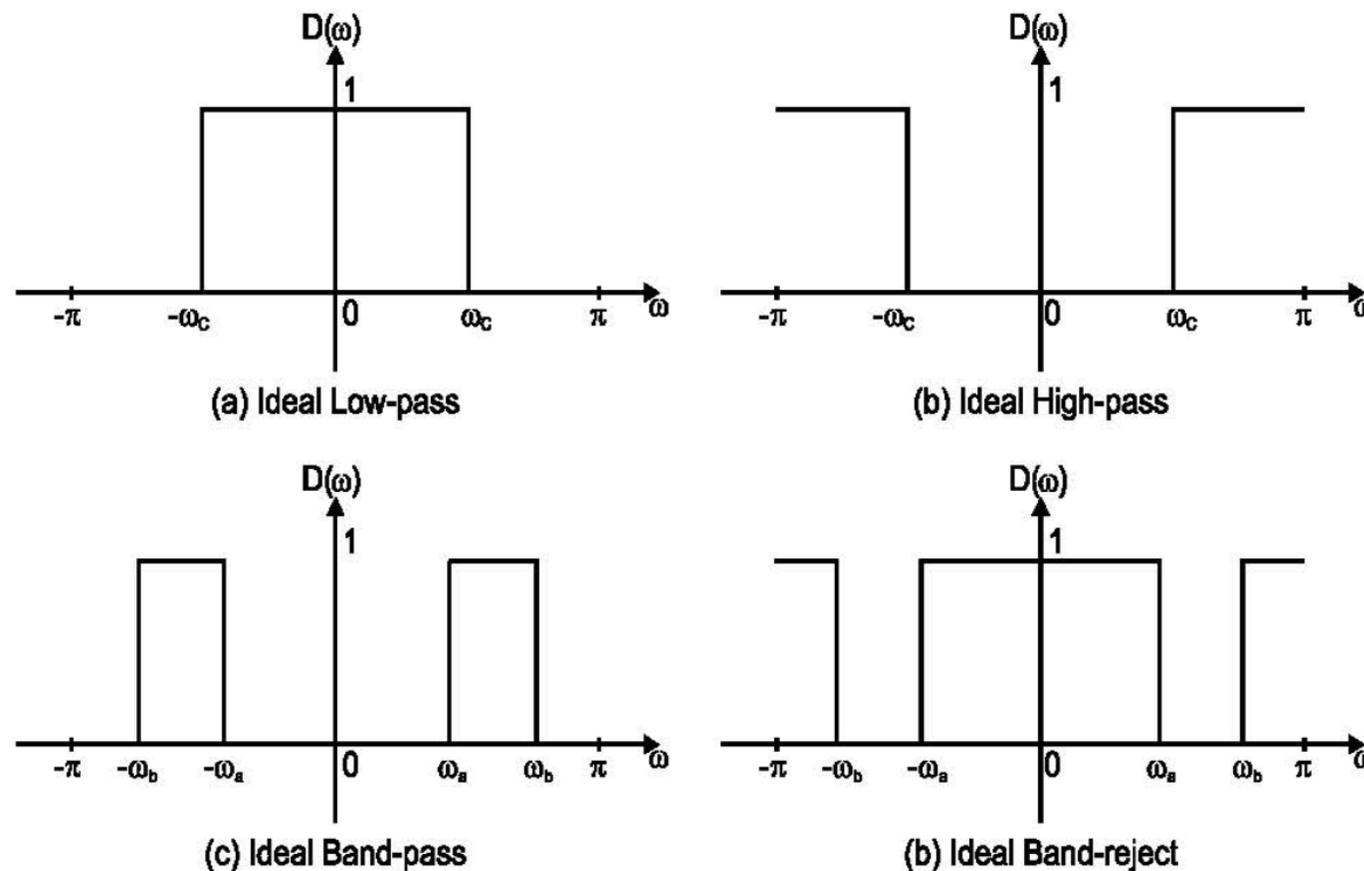
(b) Amplitude Spectrum

Projeto de filtros FIR pelo método de Janelamento

O método de janelamento é particularmente útil para projetar filtros com curvas de resposta de frequência clássicas, como filtros passa-baixa, passa-alta, passa-banda e rejeita-banda.

Exemplos dessas quatro respostas em frequência para filtros ideais são mostrados na figura ao lado.

Observe que as respostas em frequência do filtro são especificadas apenas no intervalo de frequências $-\pi \leq \omega \leq \pi$.



Isto porque, conforme já discutimos no Capítulo III das notas de aula, a resposta em frequência para filtros digitais é periódica em ω , com um período de 2π , denominado intervalo de Nyquist.

Projeto de filtros FIR pelo método de Janelamento

A resposta em frequência (contínua) e a resposta ao impulso (de tempo discreto) são relacionadas pela Transformada de Fourier em Tempo Discreto (DTFT), conforme

$$D(\omega) = \sum_{k=-\infty}^{\infty} d(k)e^{-j\omega k}$$

$$d(k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} D(\omega)e^{j\omega k} d\omega$$



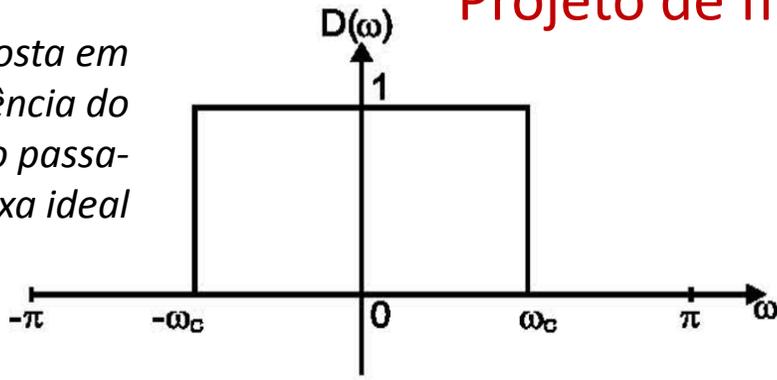
Assim, dada a resposta em frequência desejada, a resposta ao impulso do filtro pode ser obtida usando a equação para determinação da DTFT inversa.

Os coeficientes do filtro FIR serão simplesmente as amostras da resposta ao impulso.

No entanto, a resposta ao impulso obtida pela utilização da DTFT inversa, em geral, terá duas propriedades indesejáveis: a resposta será não causal no domínio tempo discreto e terá duração infinita.

Projeto de filtros FIR pelo método de Janelamento

Resposta em frequência do filtro passa-baixa ideal



Por exemplo, considere a resposta desejada do filtro passa-baixa mostrada na figura à esquerda, e expressa por

$$D(\omega) = \begin{cases} 1, & \text{if } |\omega| \leq \omega_c \\ 0, & \text{if } \omega_c \leq |\omega| \leq \pi \end{cases}$$

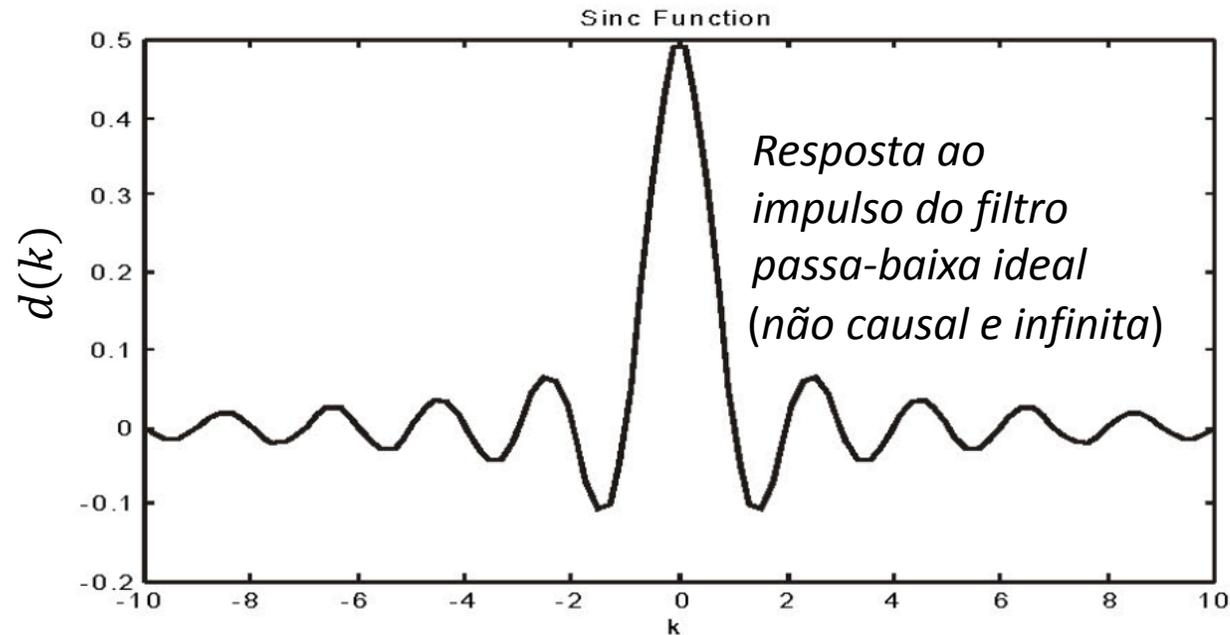
A resposta ao impulso do filtro original será

$$d(k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} D(\omega) e^{j\omega k} d\omega = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} 1 \cdot e^{j\omega k} d\omega = \frac{1}{2\pi j k} \left[e^{j\omega_c k} - e^{-j\omega_c k} \right] = \frac{\sin(\omega_c k)}{\pi k} \quad -\infty < k < \infty$$

com $d(0) = \frac{\omega_c}{\pi}$

$$\left(\lim_{k \rightarrow 0} \frac{\sin(\omega \cdot k)}{\pi \cdot k} \rightarrow \frac{\omega}{\pi} \right)$$

conforme figura ao lado (plotada como se fosse contínua para facilitar a visualização).



A resposta em frequência $D(\omega)$ é simétrica em relação a $\omega = 0$ e é real, resultando em uma resposta ao impulso $d(k)$ simétrica e também real. A resposta de fase é zero para todas as frequências.

Projeto de filtros FIR pelo método de Janelamento



Para efeito de implementação em *hardware*, é necessário truncar a resposta ao impulso $d(k)$ do filtro original, dado que a mesma apresenta uma duração infinita.

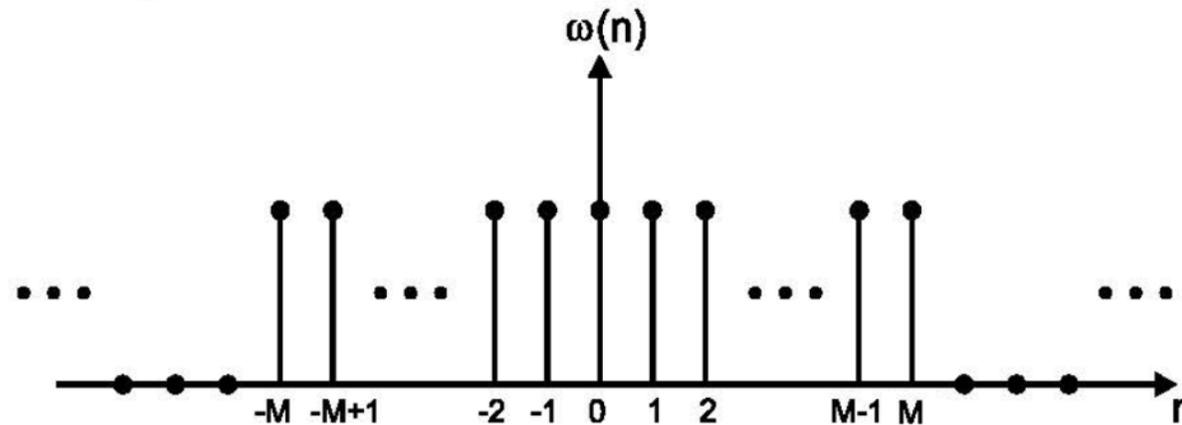
- Para truncar a resposta impulsiva $d(k)$ do filtro original, $d(k)$ é multiplicada, amostra por amostra, por uma função de janelamento, a qual tem duração finita, resultando em uma resposta impulsiva truncada e, portanto, resultando em um filtro aproximado, com características distintas do filtro original.
- Diferentes funções de janelamento podem ser utilizadas.
- Analisaremos as seguintes funções de janelamento:
 - Janela Retangular
 - Janela de Hamming
 - Janela de Blackman
 - Janela de Kaiser

Método de Janelamento – Janela Retangular

A maneira mais direta e simples de truncar a resposta impulsiva original $d(k)$ é manter os valores de $d(k)$ dentro de um determinado intervalo, por exemplo, $(-M)$ a $(+M)$, o que equivale a multiplicar $d(k)$ por uma função retangular expressa por

$$w(n) = \begin{cases} 1, & |n| \leq M \\ 0, & \text{otherwise} \end{cases}$$

conforme mostra a figura abaixo



Exemplo de janela retangular

Método de Janelamento – Janela Retangular

A resposta ao impulso resultante $h_w(n)$ tem $N = 2M$ ou $N = 2M + 1$ valores diferentes de zero.

Doravante, consideraremos N ímpar. No entanto, a análise pode ser estendida para N par.

$$h_w(n) = [d_{-M}, d_{M+1}, \dots, d_{-1}, d_0, d_1, \dots, d_{M-1}, d_M]$$



A resposta ao impulso do filtro aproximado, que resulta após a aplicação da janela retangular, é uma resposta não-causal no domínio tempo, ou seja, possui valores diferentes de zero para $n < 0$.

Note que isto viola a lei natural da causalidade, porque, se o impulso foi aplicado em $n = 0$ como sinal de excitação do filtro na entrada do mesmo, é impossível que a resposta na saída do filtro exista antes da excitação ter sido aplicada na entrada do mesmo, i.e., é impossível que exista para $n < 0$.

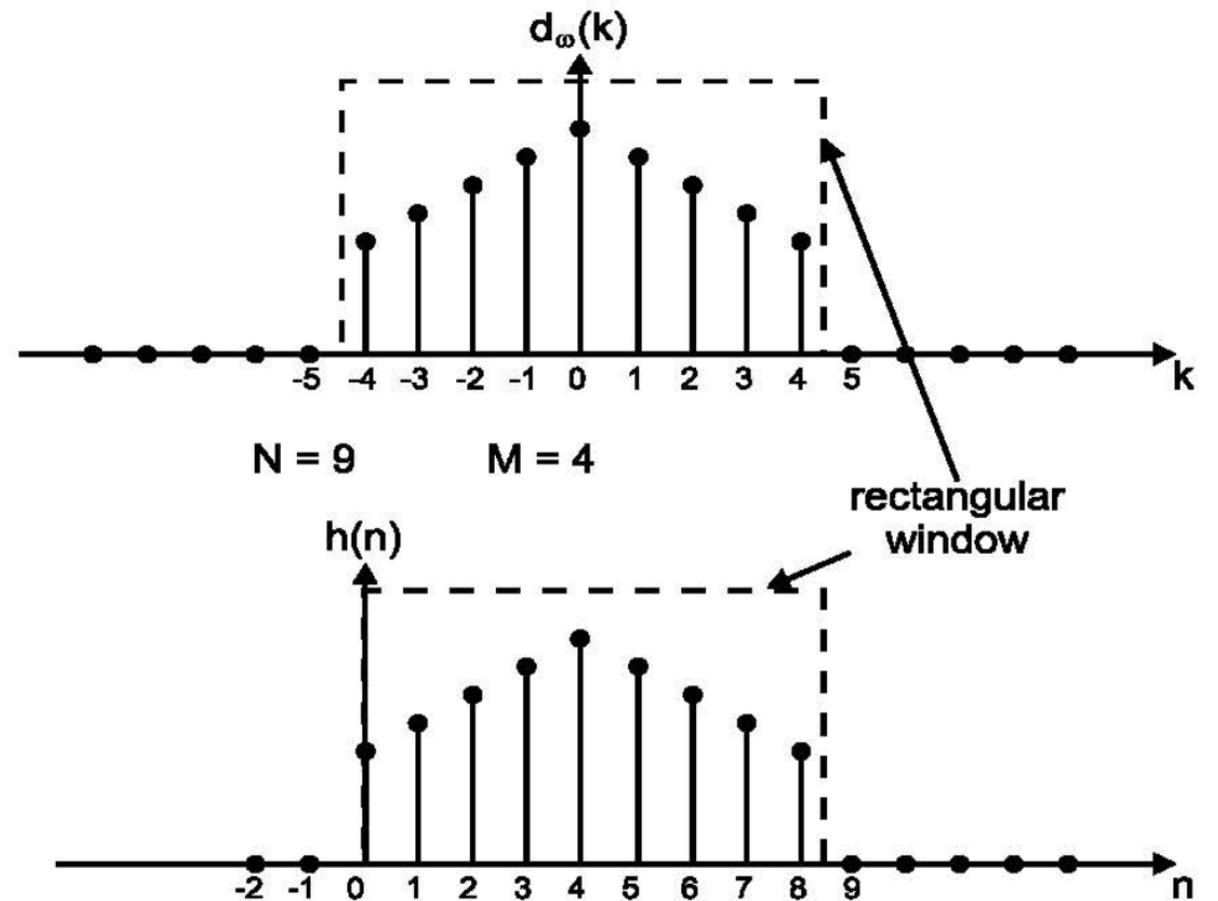
Método de Janelamento – Janela Retangular

Para tornar a resposta ao impulso causal, podemos simplesmente mudar a origem do tempo para a primeira amostra diferente de zero, e reindexar as entradas.

Neste contexto, a resposta ao impulso (e, portanto, os coeficientes) do filtro FIR será conforme

$$h(n) = d_w(n - M) \quad n = 0, 1, \dots, N - 1$$

Este processo é ilustrado na figura ao lado.



Resposta ao impulso $h(n)$ após o janelamento retangular

Método de Janelamento – Janela Retangular – Exemplo

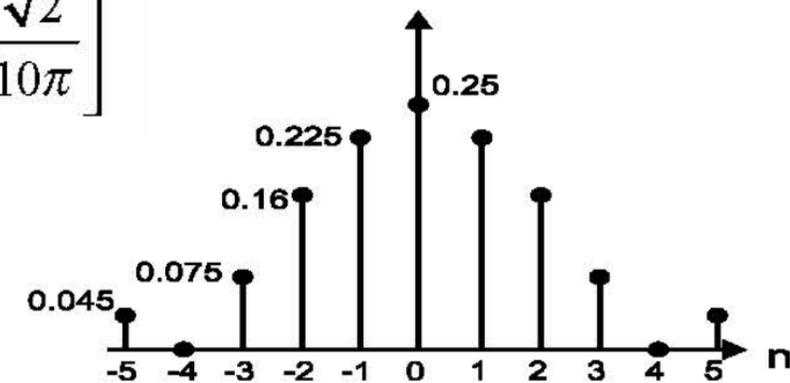
Encontre a resposta de impulso de janela retangular de um filtro passa baixa ideal com frequência de corte $\omega_c = \pi/4$. Assuma $N = 11$ amostras na resposta ao impulso após a aplicação da janela.

Dado que $N = 11$, $M = \frac{(N-1)}{2} = 5$

$$d_w(n) = \frac{\sin(\pi n/4)}{\pi n} \quad -5 \leq n \leq 5$$
$$= \left[\frac{\sqrt{2}}{10\pi}, 0, \frac{\sqrt{2}}{6\pi}, \frac{1}{2\pi}, \frac{\sqrt{2}}{2\pi}, \frac{1}{4}, \frac{\sqrt{2}}{2\pi}, \frac{1}{2\pi}, \frac{\sqrt{2}}{6\pi}, 0, \frac{\sqrt{2}}{10\pi} \right]$$

$$\left(d_w(0) = \frac{\omega_c}{\pi} = \frac{1}{4} \right)$$

A resposta ao impulso $h(n)$ do filtro aproximado é representada na figura ao lado.



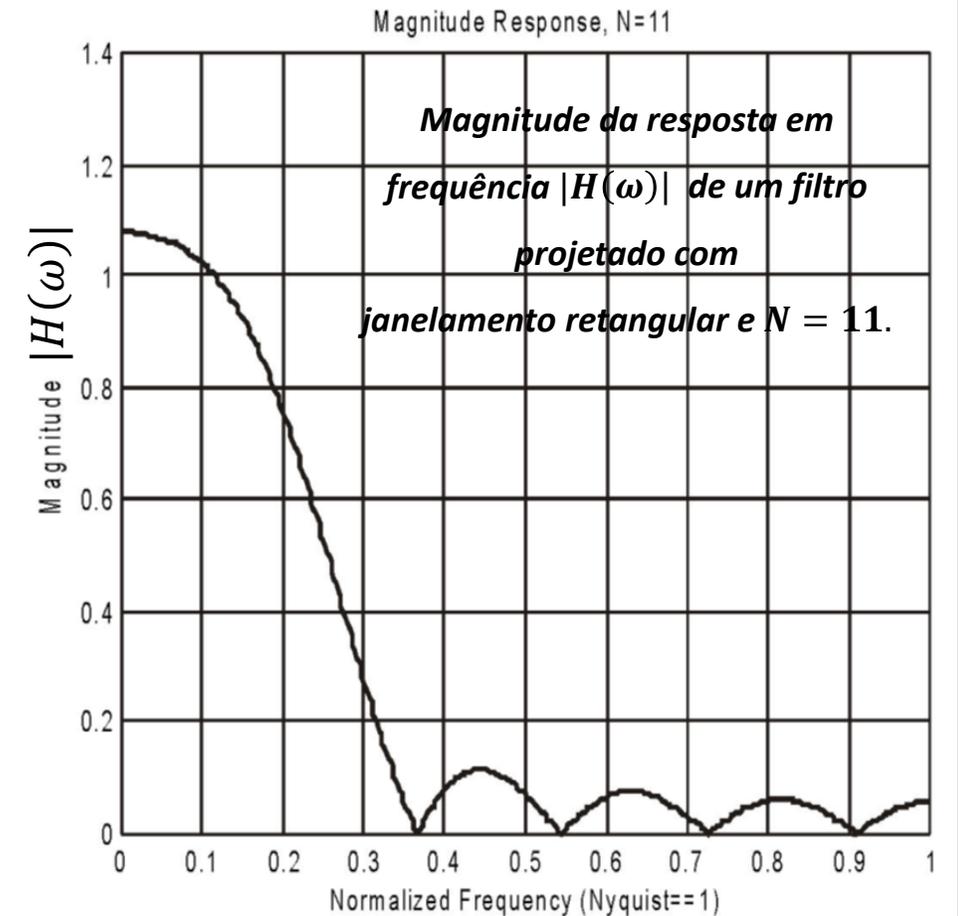
Resposta ao impulso do filtro passa-baixa ideal após a aplicação da janela retangular

Método de Janelamento – Janela Retangular – Avaliação de desempenho

A avaliação de desempenho busca verificar quão “bom” resultou o projeto usando o janelamento retangular.

Ou seja, verificar quão perto está a aproximação da resposta em frequência do filtro FIR aproximado resultante, tendo como referência a resposta em frequência original $D(\omega)$.

Para tanto, realizamos a DTFT de $h(n)$, denotada por $H(\omega)$, e plotamos a magnitude da resposta conforme mostra a figura ao lado.

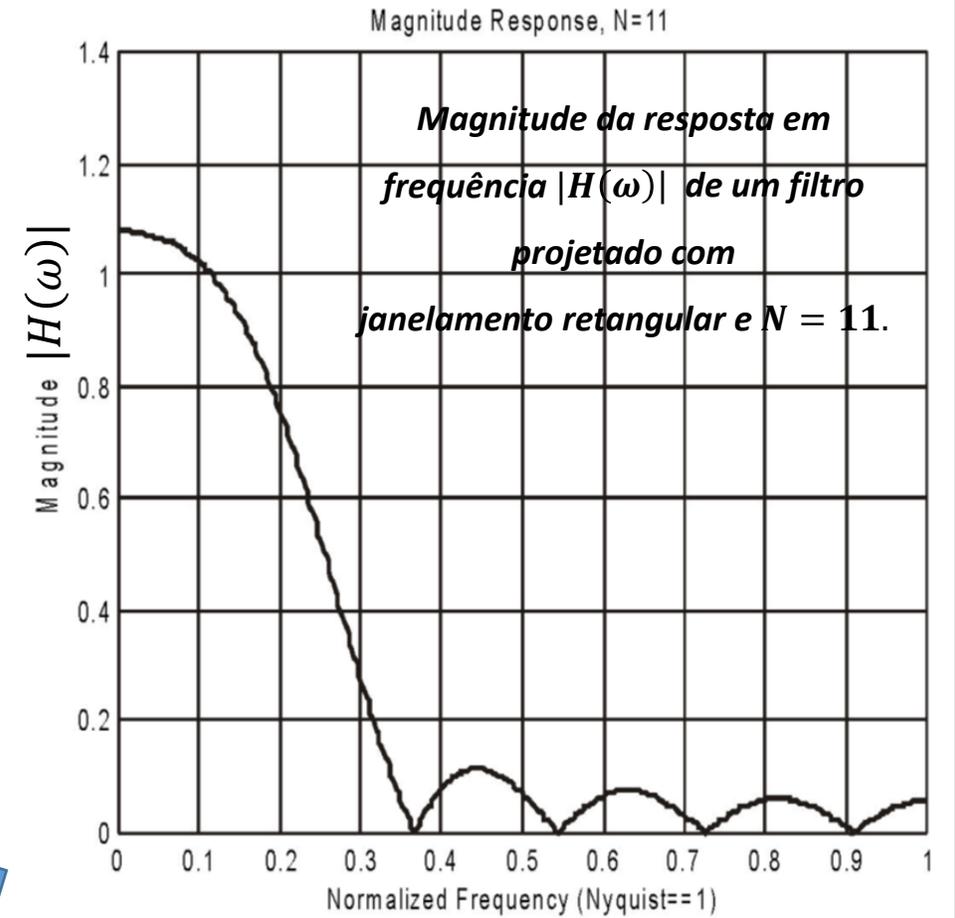
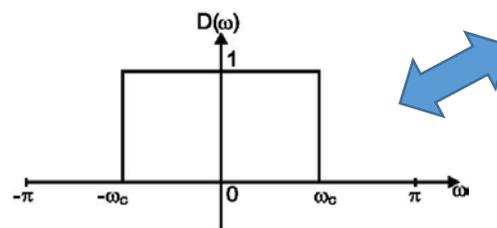


Método de Janelamento – Janela Retangular – Avaliação de desempenho

Note que, desde que $h(n)$ não é mais simétrica em relação à origem, a sua DTFT será de valor complexo. Por isso só compararemos a magnitude da resposta $|H(\omega)|$ no intervalo de frequência de 0 a π .

Observe que a curva obtida para $|H(\omega)|$ está longe de ser a “caixa quadrada” $D(\omega)$ do filtro ideal mostrada na figura (a) do *slide 27*. E que a transição da faixa de passagem para a faixa de rejeição não é abrupta como seria no caso do filtro ideal

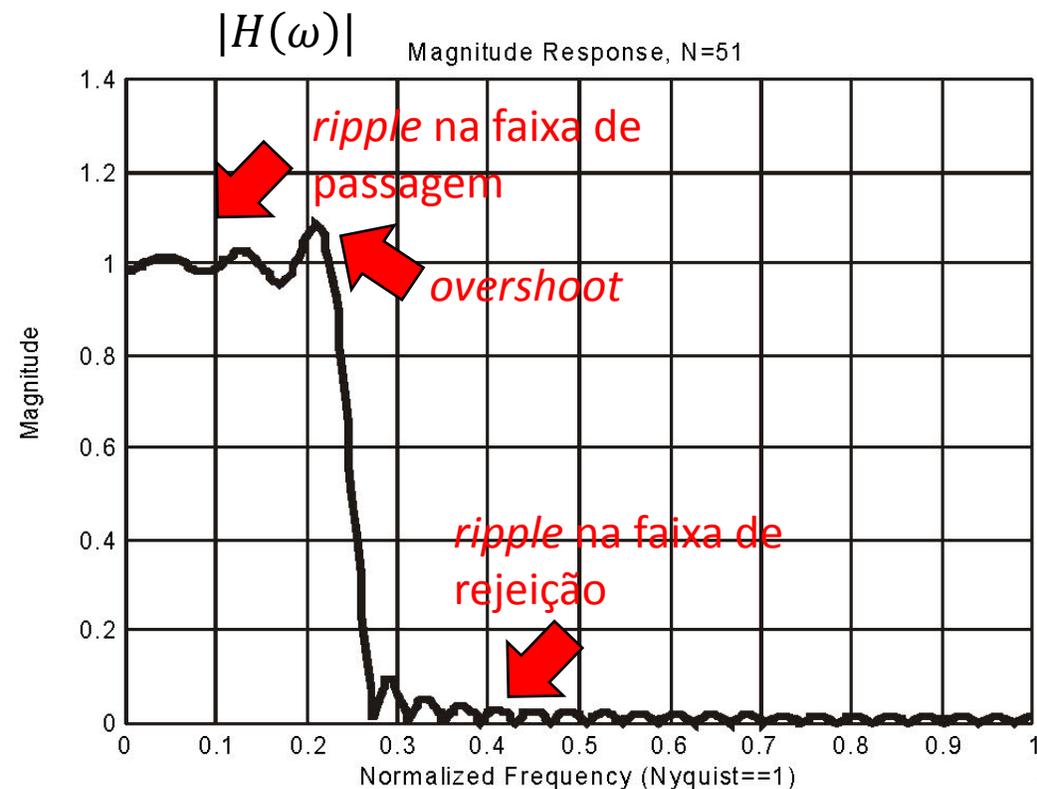
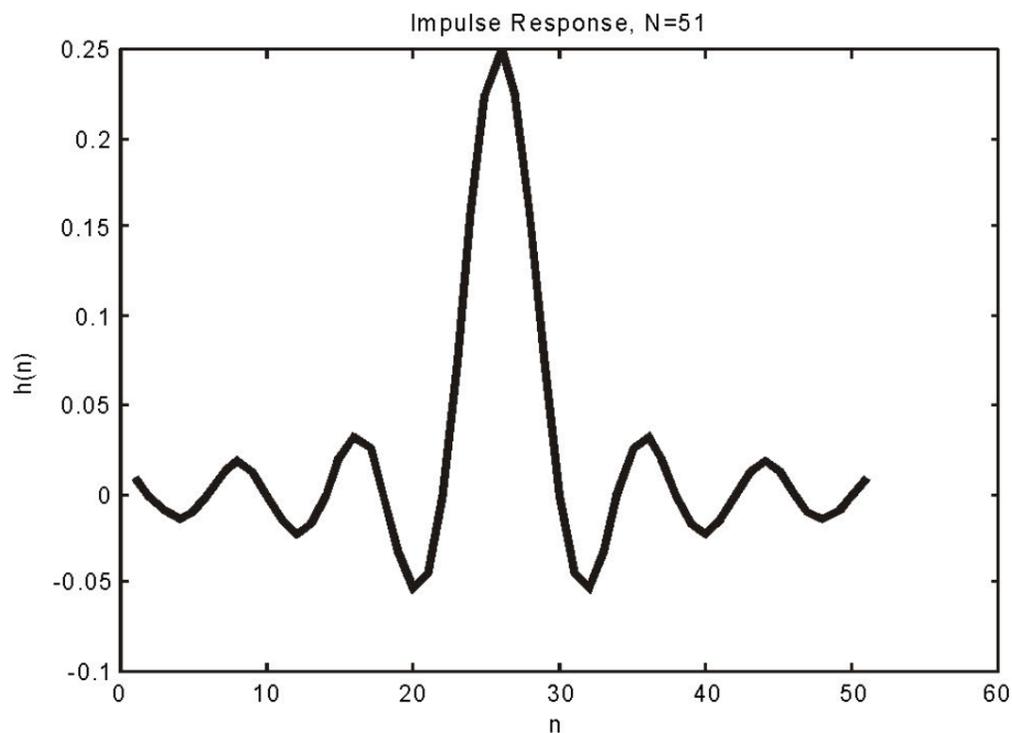
Note que a curva de $|H(\omega)|$ apresenta ondulações na faixa de rejeição, e que a curva de $|H(\omega)|$ apresenta diferenças significativas na faixa de passagem, quando comparada com a curva $D(\omega)$ do filtro passa-baixa ideal.



Método de Janelamento – Janela Retangular – Avaliação de desempenho

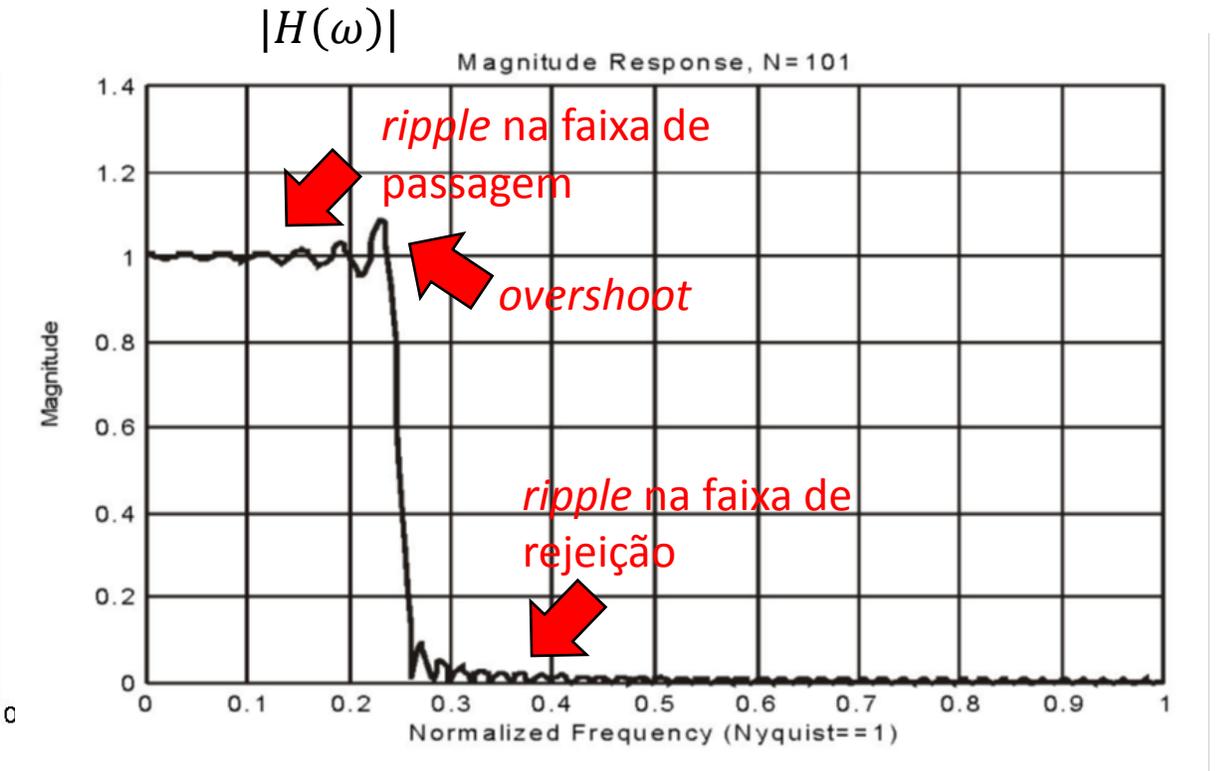
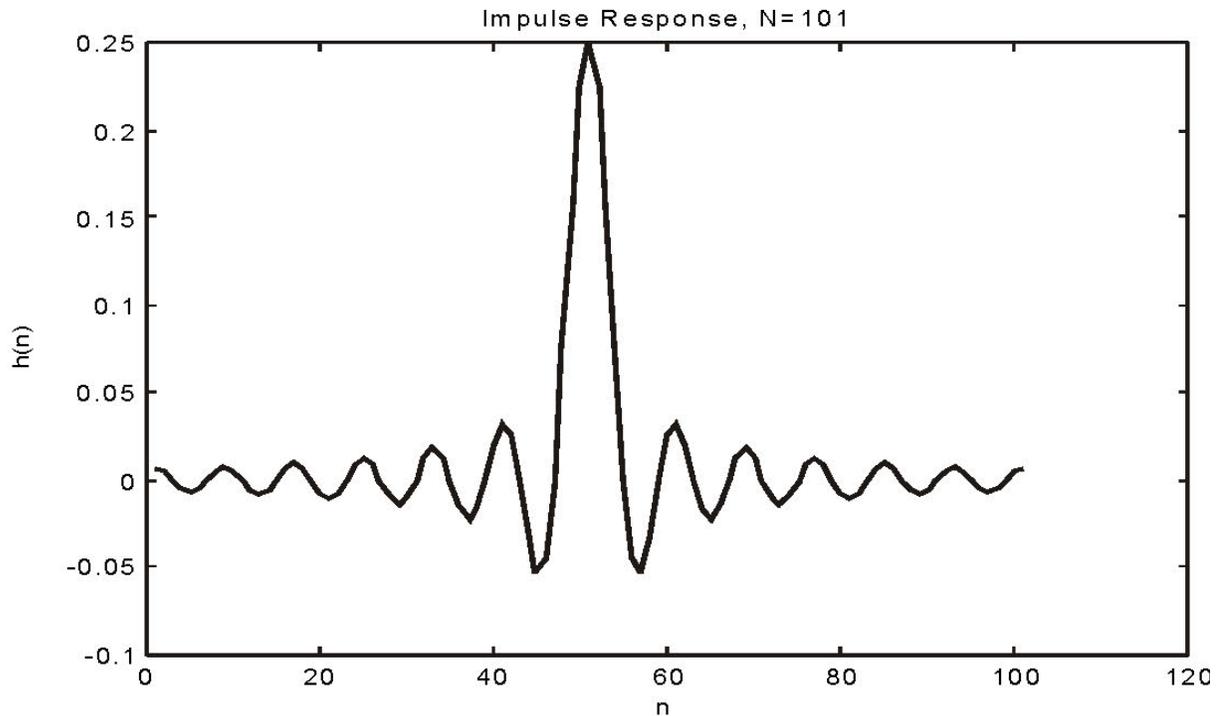
As ondulações (=ripple) observadas na faixa de passagem e na faixa de rejeição podem ser melhor observadas se reprojetermos o filtro com um maior número de amostras N na janela retangular.

A figura abaixo mostra o gráfico da resposta ao impulso (plotada como se fosse contínua para facilitar a visualização) e o gráfico de $|H(\omega)|$ para um filtro passa baixa projetado com **janela retangular de $N = 51$ amostras**.

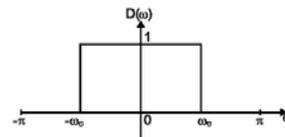


Método de Janelamento – Janela Retangular – Avaliação de desempenho

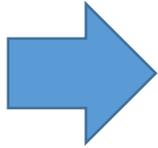
A figura abaixo mostra o gráfico da resposta ao impulso (plotada como se fosse contínua para facilitar a visualização) e o gráficos de $|H(\omega)|$ para um filtro passa baixa projetado com **janela retangular** de $N = 101$ amostras.



Comparando os exemplos para $N = 11$, $N = 51$ e $N = 101$, nota-se que aumenta a fidelidade com que as curvas de $|H(\omega)|$ aproximam a “caixa quadrada” $D(\omega)$ do filtro ideal mostrada na figura (a) do slide 27 à medida que N aumenta.



Método de Janelamento – Janela Retangular – Avaliação de desempenho

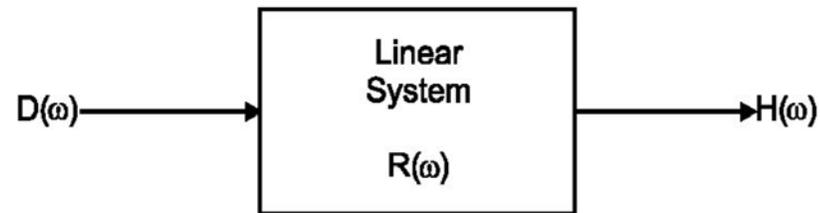


Seria de esperar que, à medida que N aumenta, a aproximação se torne melhor. Isto é verdadeiro exceto para a região perto da transição entre a faixa de passagem e a faixa de rejeição.

- Esta região de transição é uma descontinuidade abrupta na resposta de frequência ideal desejada.
- O truncamento da série de Fourier, implícito no janelamento retangular, introduz ondulações na curva de resposta em frequência devido à convergência não uniforme da série de Fourier em uma descontinuidade, efeito ondulatorio que é conhecido como Fenômeno de Gibbs.
- Por esta razão, a aproximação na borda da faixa de passagem sempre será ruim para o caso do janelamento retangular, independentemente do tamanho de N .
- As ondulações na curva de resposta em frequência são denominadas de *ripple*. De particular importância é o *ripple* na faixa de passagem de um filtro, que deseja-se ser o menor possível.

Método de Janelamento – Janela Retangular – Avaliação de desempenho

No processo de janelamento no domínio tempo da resposta ao impulso do filtro original, as distorções da aproximação $H(\omega)$ em relação à $D(\omega)$ do filtro original podem ser também interpretadas através das propriedades de sistemas lineares no domínio frequência, conforme mostra a figura abaixo.



Na figura, a resposta em frequência $D(\omega)$ desejada para o filtro original é a entrada de um sistema linear cuja resposta em frequência $R(\omega)$ é a DTFT da janela retangular no domínio tempo discreto que será aplicada à resposta ao impulso do filtro original.

A saída $H(\omega)$ do sistema é, então, a resposta em frequência do filtro FIR aproximado que resulta da aplicação da janela retangular à resposta ao impulso do filtro original, sendo dada por:

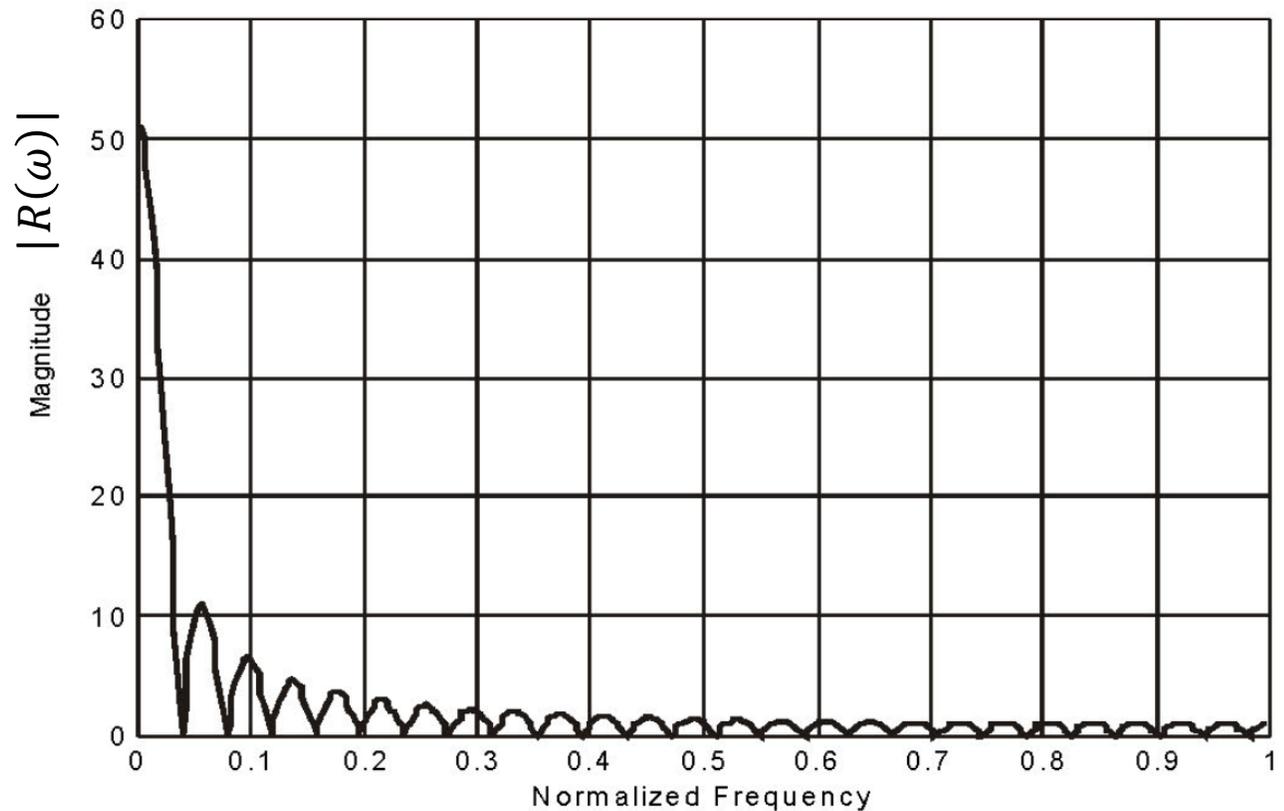
$$H(\omega) = R(\omega)D(\omega)$$

Mas a DTFT de uma janela retangular no domínio tempo resulta na resposta em frequência $R(\omega)$ mostrada na figura a seguir.

Método de Janelamento – Janela Retangular – Avaliação de desempenho

Conforme observa-se na figura ao lado, a resposta em frequência $R(\omega)$ da operação de janelamento retangular que é aplicada à resposta ao impulso do filtro original é a causa da introdução do *ripple* na curva de resposta em frequência $H(\omega)$ do filtro aproximado, seja na faixa de passagem de $H(\omega)$, seja na faixa de rejeição de $H(\omega)$.

Portanto, para minimizar o *ripple* e as distorções na curva de resposta em frequência $H(\omega)$ do filtro aproximado é necessário truncar a resposta ao impulso do filtro original com **uma janela no domínio tempo cuja resposta em frequência seja mais suave e menos ondulatória que a resposta em frequência $R(\omega)$ da janela retangular**. Neste contexto, outros tipos de janelas no tempo com estas características desejáveis em frequência podem ser utilizadas, conforme veremos adiante.



Método de Janelamento – Janela Retangular – Avaliação de desempenho

Resumo das características do filtro com janelamento retangular

- A amplitude da ondulação (*ripple*) diminui com o aumento da ordem N do filtro.
- A aproximação na faixa de passagem e na faixa de rejeição se torna melhor à medida que a ordem do filtro aumenta.
- A largura em frequência da região de transição entre a faixa de passagem e a faixa de rejeição diminui com o aumento da ordem do filtro. Para qualquer ordem N , a magnitude da resposta em frequência $H(\omega)$ do filtro é sempre igual a 0.5 na frequência de corte ω_c , isto é, $|H(\omega_c)| = 0.5$.
- A amplitude do *ripple* em frequências próximas à região de transição entre a faixa de passagem e a faixa de rejeição permanece mais ou menos a mesma à medida que N aumenta (ver slides 37 e 38). A amplitude máxima do *ripple* no início da borda de descida da região de transição, denominado *overshoot*, é aproximadamente 9% maior que o valor (idealmente constante) de $|H(\omega)|$ ao longo da faixa de passagem (ver slides 37 e 38).

Método de Janelamento – Janela de Hamming

Conforme visto anteriormente, o truncamento abrupto da resposta ao impulso do filtro original com um janela retangular gera uma descontinuidade no domínio tempo. Esta descontinuidade no domínio tempo dá origem ao *ripple* na resposta em frequência $H(\omega)$ do filtro aproximado. Para reduzir este efeito, uma alternativa é substituir a janela retangular por uma função de janela que termina suavemente nas duas extremidades da janela no tempo.

Nesta classe de funções de janelamento que terminam suavemente nas duas extremidades da janela no tempo encontra-se a Janela de Hamming, definida analiticamente por

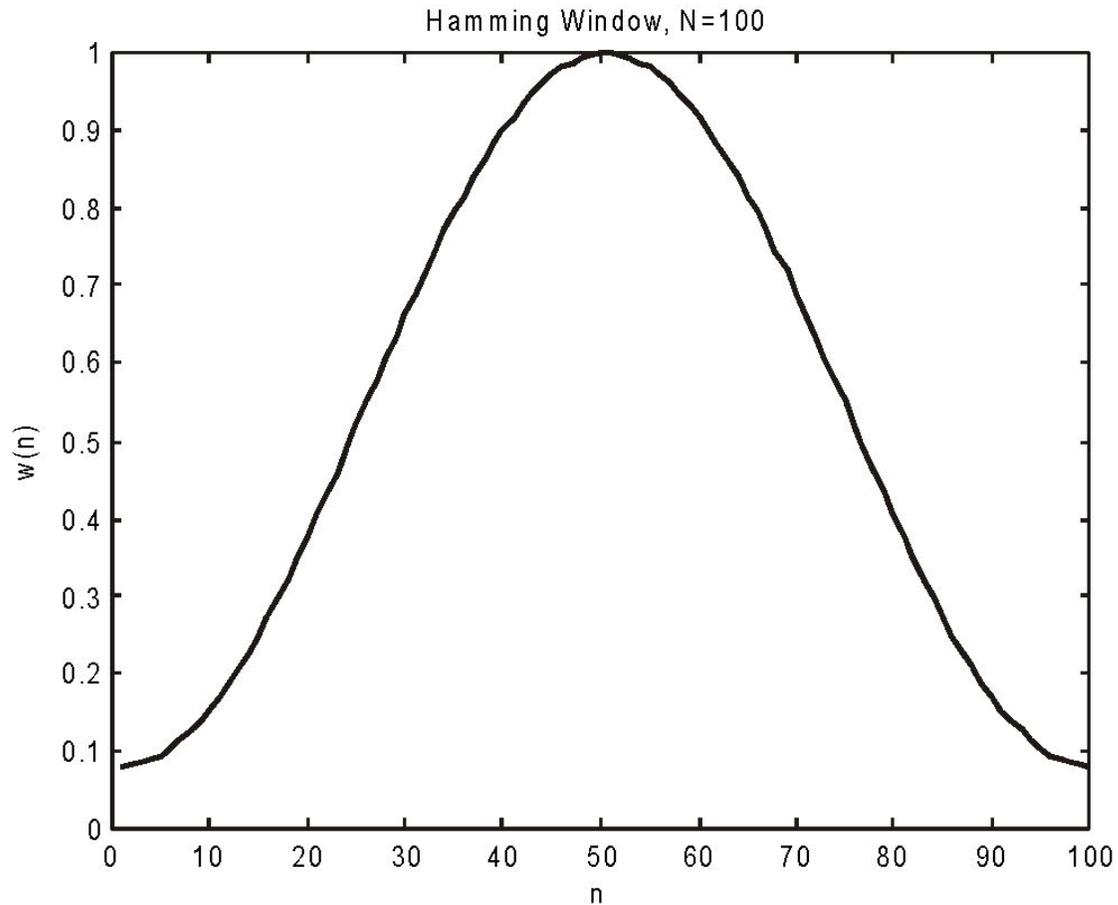
$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \quad n = 0, 1, \dots, N-1$$

Observe que esta equação define as amostras da janela como já deslocadas (índices de 0 para $N-1$). Assim, a resposta ao impulso do filtro passa baixa FIR aproximado, projetado usando janelamento de Hamming será

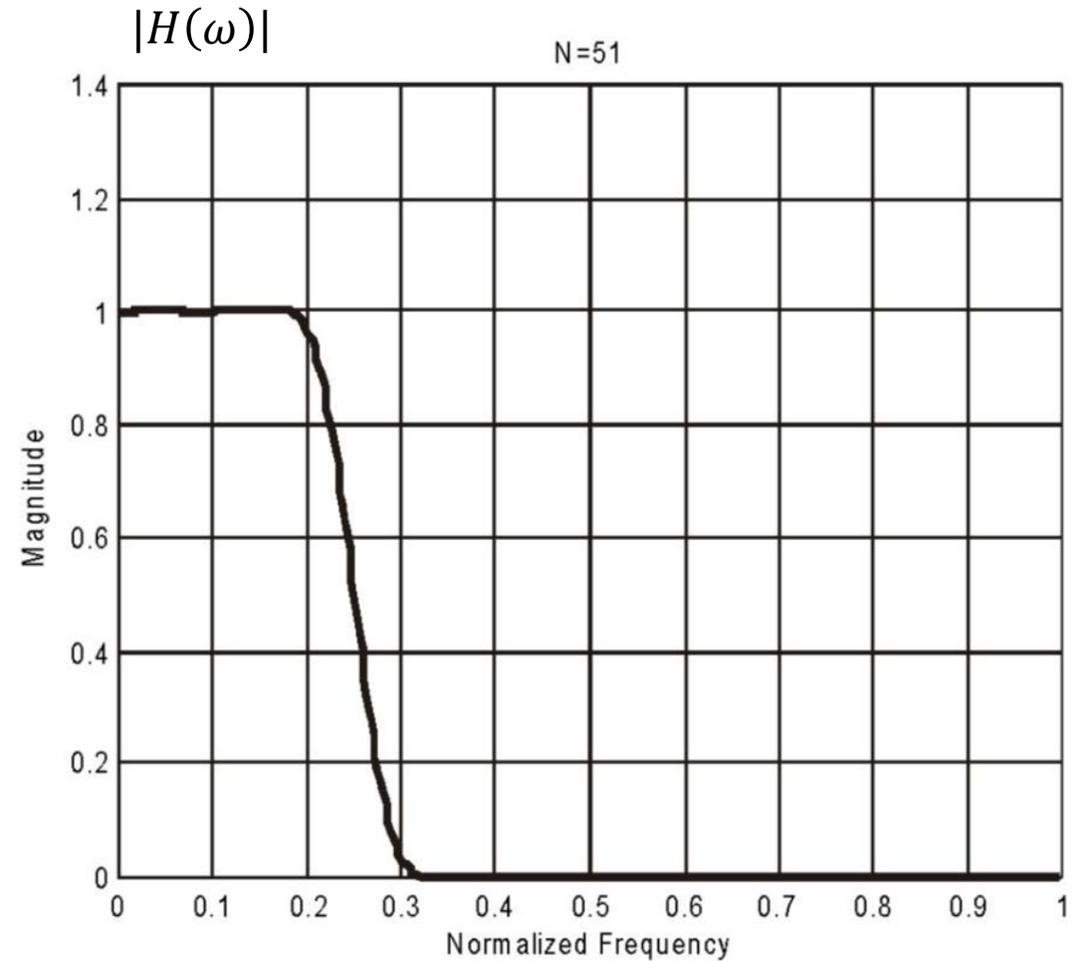
$$\begin{aligned} h(n) &= w(n)d(n-M) \\ &= \left[0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \right] \cdot \frac{\sin[(n-M)\omega_c]}{(n-M)\pi} \end{aligned}$$

As figuras a seguir mostram o gráfico da função $w(n)$ para $N = 100$ e a magnitude da resposta em frequência $|H(\omega)|$ do filtro aproximado para $N = 51$.

Método de Janelamento – Janela de Hamming



Janela de Hamming para $N = 100$.

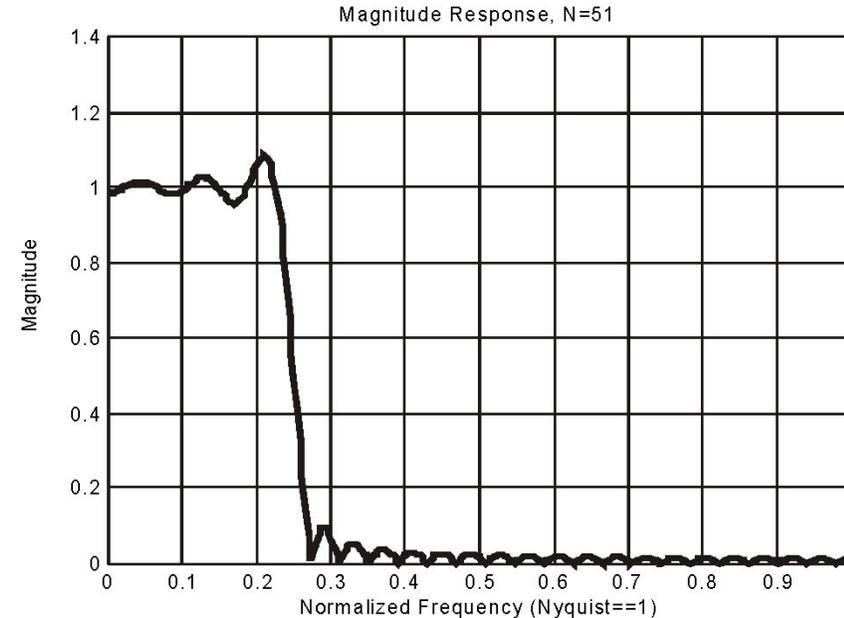
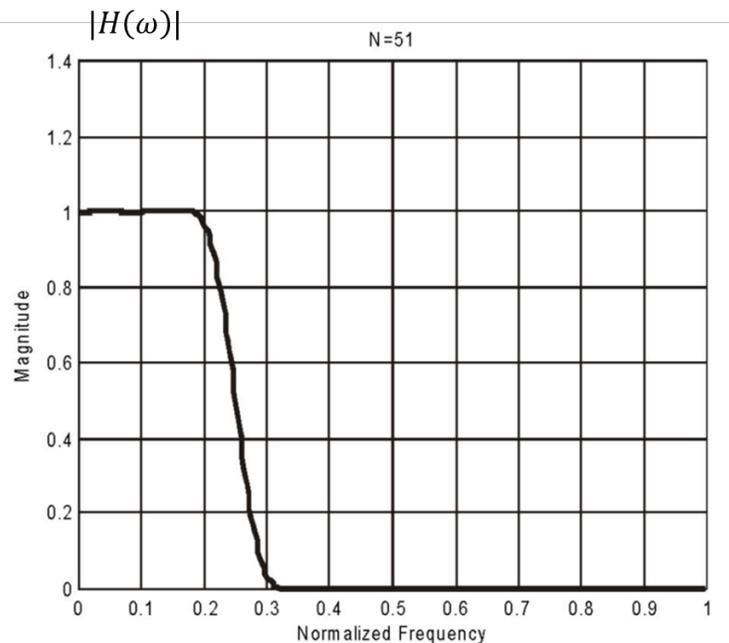


Magnitude da resposta em frequência $|H(\omega)|$ do filtro aproximado obtido por janelamento de Hamming. O filtro é passa-baixa com $N = 51$, e com frequência de corte em $\pi / 4$.

Método de Janelamento – Janela de Hamming

Comparando a resposta $|H(\omega)|$ obtida por janelamento de Hamming (à esquerda) com a resposta $|H(\omega)|$ obtida por janelamento retangular (à direita), nota-se que o projeto com janela de Hamming praticamente elimina o *ripple* na faixa de passagem e na faixa de rejeição. Quantitativamente, o *ripple* máximo na faixa de rejeição é cerca de 53dB abaixo do ganho na faixa de passagem.

No entanto, o janelamento de Hamming resulta em um aumento da largura em frequência da região de transição entre a faixa de passagem e a faixa de rejeição de $|H(\omega)|$, quando comparado ao janelamento retangular.



Método de Janelamento – Janela de Blackman

A Janela de Blackman exibe um *ripple* máximo na faixa de rejeição ainda menor do que o *ripple* de um filtro que utiliza janela de Hamming (cerca de 74 dB abaixo).

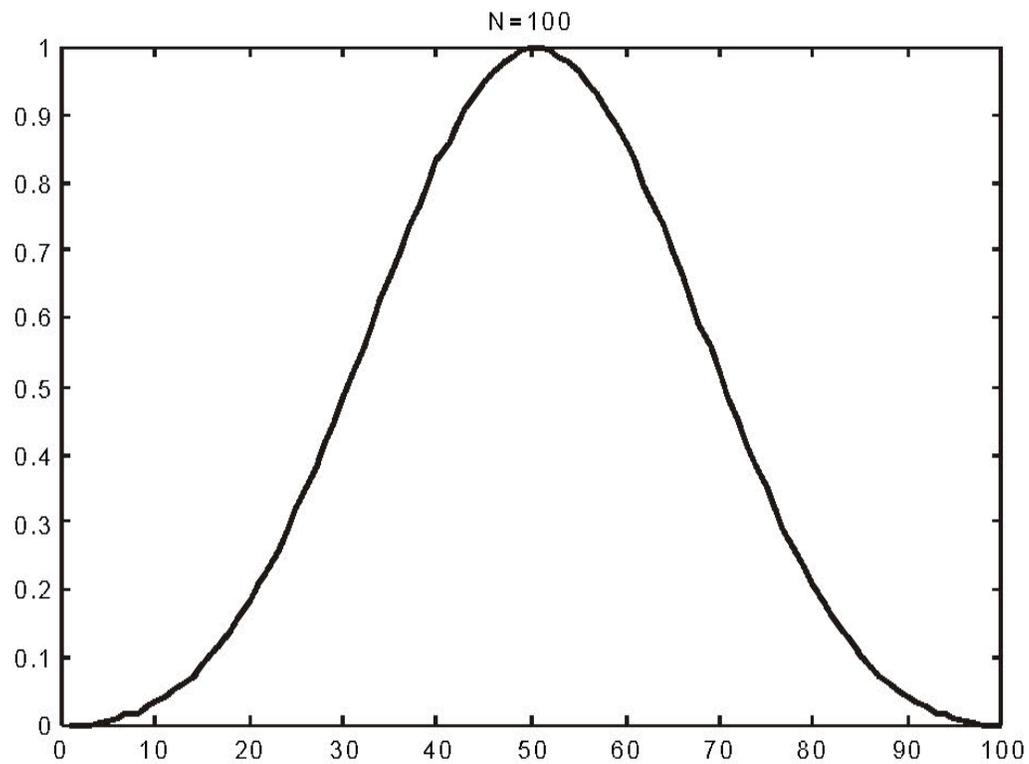
A Janela de Blackman é definida matematicamente conforme

$$w(n) = 0.42 - 0.5 \cos\left(\frac{2\pi n}{N-1}\right) + 0.08 \cos\left(\frac{4\pi n}{N-1}\right) \quad n = 0, 1, \dots, N-1$$

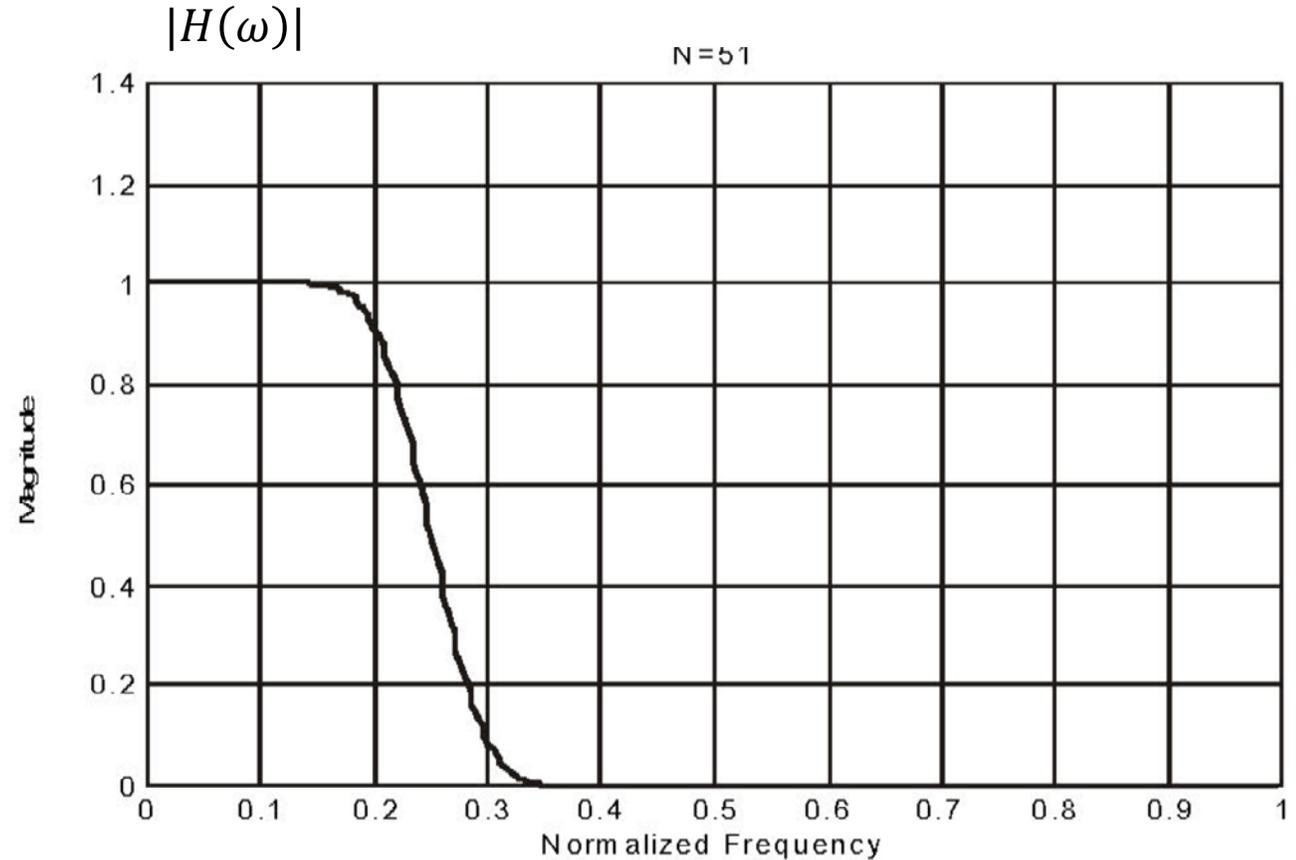
As figuras a seguir mostram o gráfico da função $w(n)$ para $N = 100$ e a magnitude da resposta em frequência $|H(\omega)|$ do filtro aproximado para $N = 51$.

Note nas figuras que seguem que o janelamento de Blackman resulta em um aumento da largura em frequência da região de transição entre a faixa de passagem e a faixa de rejeição de $|H(\omega)|$, quando comparado ao janelamento de Hamming

Método de Janelamento – Janela de Blackman



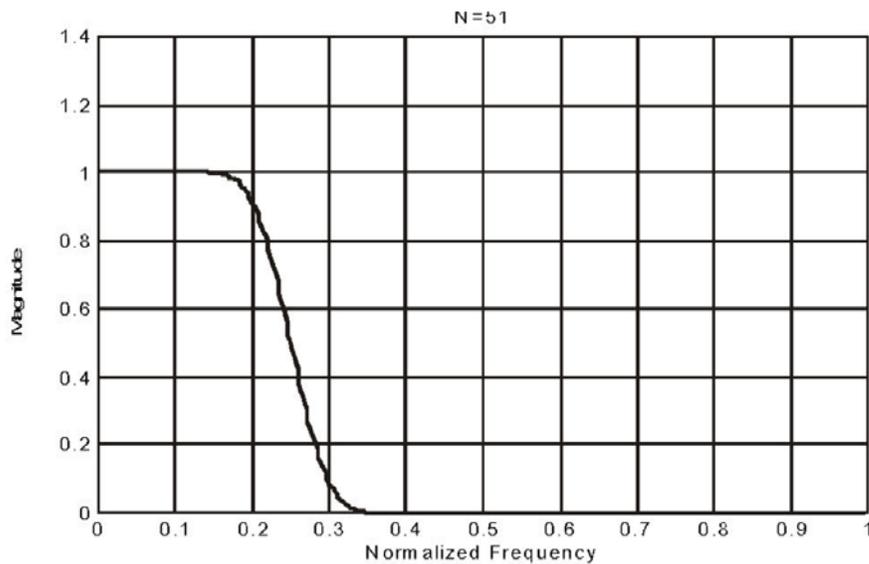
Janela de Blackman para $N = 100$.



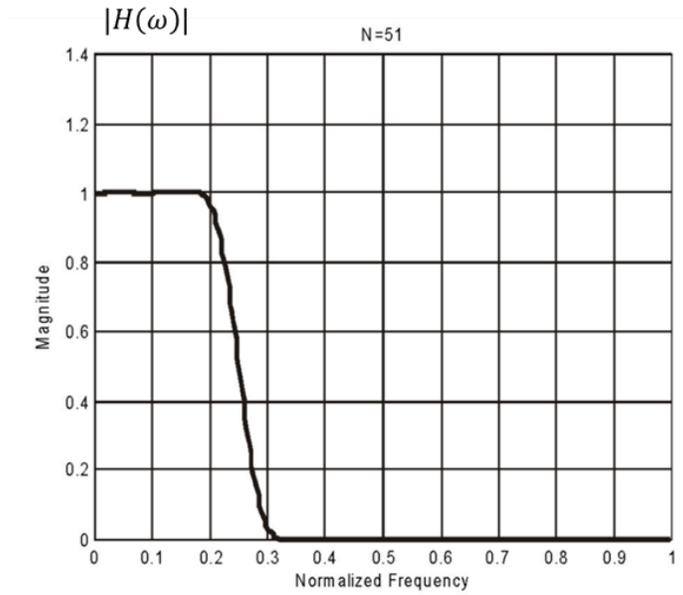
Magnitude da resposta em frequência $|H(\omega)|$ do filtro aproximado obtido por janelamento de Blackman. O filtro é passa-baixa com $N = 51$, e com frequência de corte em $\pi / 4$.

Método de Janelamento – Janela de Blackman

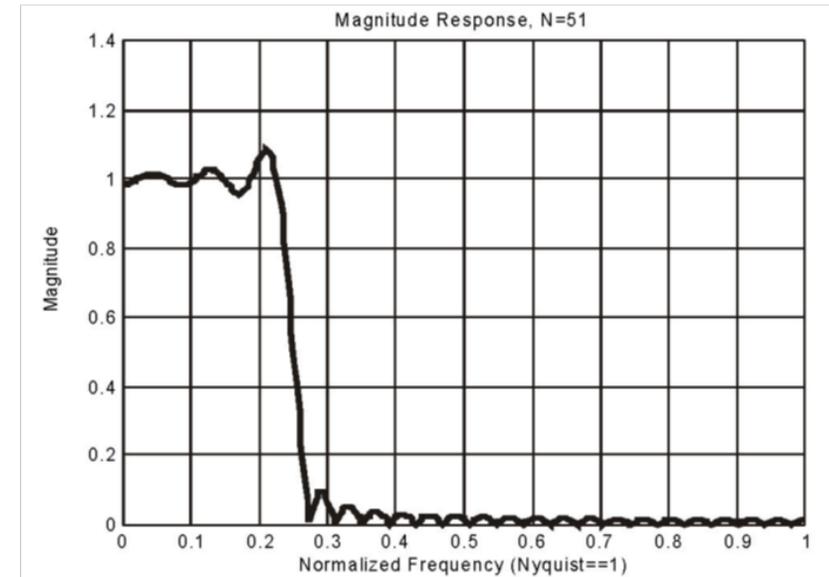
Blackman



Hamming



Retangular



Diminuição do *ripple* na faixa de passagem e na faixa de rejeição

Aumento da largura em frequência da região de transição entre a faixa de passagem e a faixa de rejeição de $|H(\omega)|$

Outros métodos de Janelamento – Janela de Kaiser

- A principal vantagem das três funções anteriores de janelamento (Retangular, Hamming e Blackman) é que elas são simples de aplicar e as características da resposta em frequência $|H(\omega)|$ do filtro resultante são uma aproximação razoável da resposta em frequência $|D(\omega)|$ do filtro original.
- Para um grande número de aplicações, projetos com a janela Hamming ou com a janela de Blackman serão suficientes para satisfazer as especificações desejadas.
- A principal desvantagem das referidas funções de janelamento é que suas características, tais como atenuação máxima na faixa de passagem (também denominado *insertion loss* do filtro) e o overshoot (amplitude máxima do *ripple* no início da borda de descida da região de transição de $|H(\omega)|$), são basicamente fixas, não podendo ser especificadas.
- Então, se as especificações desejadas exigem um determinado *overshoot* e uma determinada largura de região de transição entre a faixa de passagem e a faixa de rejeição de $H(\omega)$, pode ser interessante considerar o uso da janela de Kaiser.
- Dada a complexidade do equacionamento, para efeito de estudo da janela de Kaiser, é conveniente utilizar ferramentas de *software* específicas para tal, conforme o próximo *slide*.

CAD para projeto de filtros a partir das especificações

A sequência de procedimentos para projeto e análise de um filtro digital é extensa. Usualmente adotam-se técnicas computacionais para tanto.

Dentre elas destacam-se as ferramentas para projeto e análise de filtros digitais disponíveis no *software* Matlab, em particular, as ferramentas “filterDesigner” e “fdatool”.

O *software* Matlab está disponível nos laboratórios da Escola Politécnica. Os próximos *slides* mostram as telas de suporte ao uso da ferramenta.

No link <https://www.youtube.com/watch?v=83zjvn6pJHA> encontra-se um exemplo em vídeo de projeto de um filtro passa-faixa utilizando janela de Kaiser para efeito de supressão de ruído de áudio, utilizando a ferramenta “fdatool”. O vídeo é disponibilizado pelo canal “Matlab & Simulink Telecom”, de tutoriais sobre Matlab.

CONTENTS

- < Documentation Home
- < Signal Processing Toolbox
- < Digital and Analog Filters
- < Digital Filter Design
- < Signal Processing Toolbox
- < Digital and Analog Filters
- < Digital Filter Analysis
- < Signal Processing Toolbox
- < Apps

- Filter Designer**
- ON THIS PAGE
 - Description
 - Open the Filter Designer App
 - Examples
 - See Also

MATLAB R2017a

HOME PLOTS APPS

Search Documentation Log In

New Script New Open Compare Import Data Save Workspace New Variable Open Variable Clear Workspace Analyze Code Run and Time Clear Commands Simulink Layout Parallel Add-Ons Help Community Request Support Learn MATLAB

FILE VARIABLE CODE SIMULINK ENVIRONMENT RESOURCES

C:\Program Files\MATLAB\R2017a\bin

Current Folder

Name
m3registry
registry
util
win64
deploytool.bat
lcdata.xml
lcdata.xsd
lcdata_utf8.xml
matlab.exe
mbuild.bat
mcc.bat
mex.bat
mex.pl
mexext.bat
mexsetup.pm
mexutils.pm
mw_mpiexec.bat
worker.bat

Details

Select a file to view details

Command Window

```
>> help filterDesigner
filterDesigner Filter Designer
filterDesigner launches the Filter Designer.
Filter Designer is a Graphical User Interface (GUI) that allows you to
design or import, and analyze digital FIR and IIR filters.

If the DSP System Toolbox is installed, Filter Designer seamlessly
integrates advanced filter design methods and the ability to
quantize filters.

% Example:
% Launch Filter Designer.

filterDesigner; % Lanches Filter Designer

See also fvtool, signalAnalyzer.

Reference page for signal/filterDesigner

%>>
```

Workspace

Name	Value
------	-------

expand all in page



ic helpful? Yes No

CONTENTS Close

- < Documentation Home
- < Signal Processing Toolbox
- < Digital and Analog Filters
- < Digital Filter Design
- < Signal Processing Toolbox
- < Digital and Analog Filters
- < Digital Filter Analysis
- < Signal Processing Toolbox
- < Apps

Filter Designer

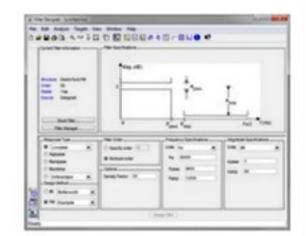
Design filters starting with algorithm selection

[expand all in page](#)

Description

The **Filter Designer** app enables you to design and analyze digital filters. You can also import and modify existing filter designs.

> **More**



Open the Filter Designer App

- MATLAB® Toolstrip: On the **Apps** tab, under **Signal Processing and Communications**, click the app icon.
- MATLAB command prompt: Enter `filterDesigner`.

Examples

Related Examples

- [Getting Started with Filter Designer](#)
- [FIR Bandpass Filter with Asymmetric Attenuation](#)
- [Arbitrary Magnitude Filter](#)

See Also

Apps

[Signal Analyzer](#) | [Window Designer](#)

Functions

`designfilt` | `fvtool` | `sptool` | `wvtool`

Introduced before R2006a

Was this topic helpful?

CONTENTS Close

- Documentation Home
- Signal Processing Toolbox
 - Getting Started with Filter Designer
 - ON THIS PAGE
 - Choosing a Response Type
 - Choosing a Filter Design Method
 - Setting the Filter Design Specifications
 - Computing the Filter Coefficients
 - Analyzing the Filter
 - Editing the Filter Using the Pole/Zero Editor
 - Converting the Filter Structure
 - Exporting a Filter Design
 - Generating a C Header File
 - Generating MATLAB Code
 - Managing Filters in the Current Session
 - Saving and Opening Filter Design Sessions

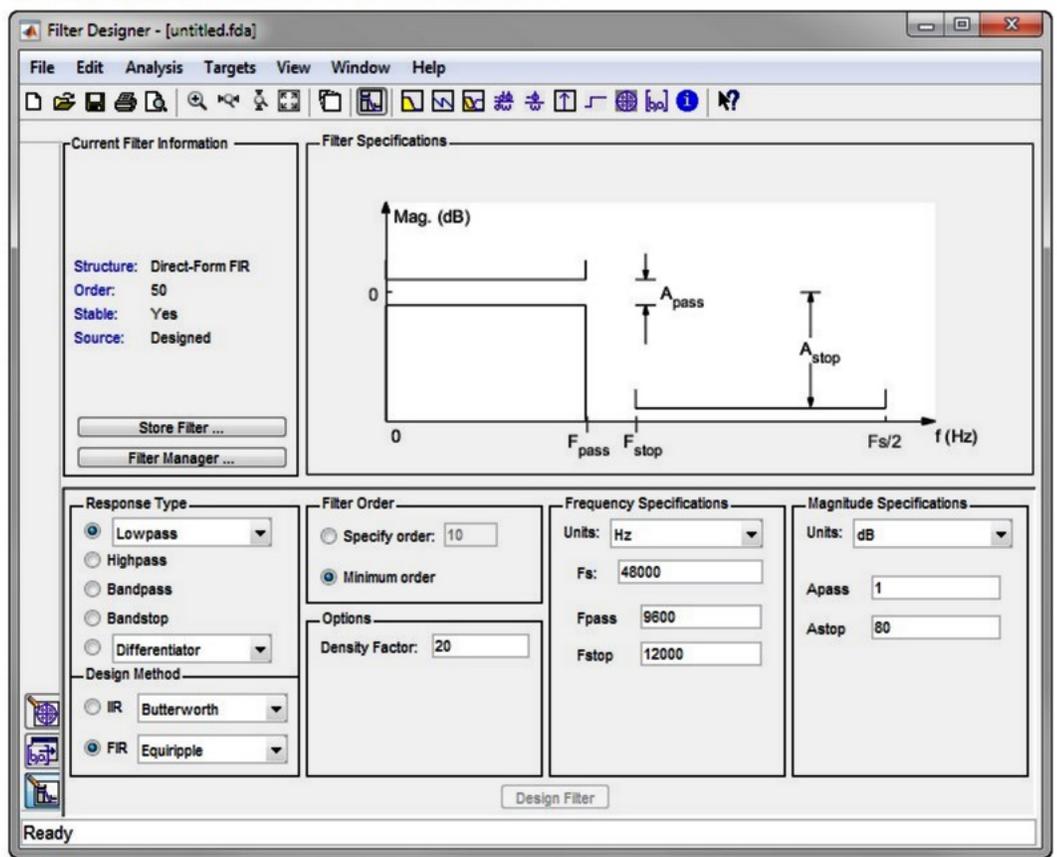
Getting Started with Filter Designer

To open the Filter Designer app, type

```
filterDesigner
```

at the MATLAB® command prompt.

The Filter Designer app opens with the Design Filter panel displayed.



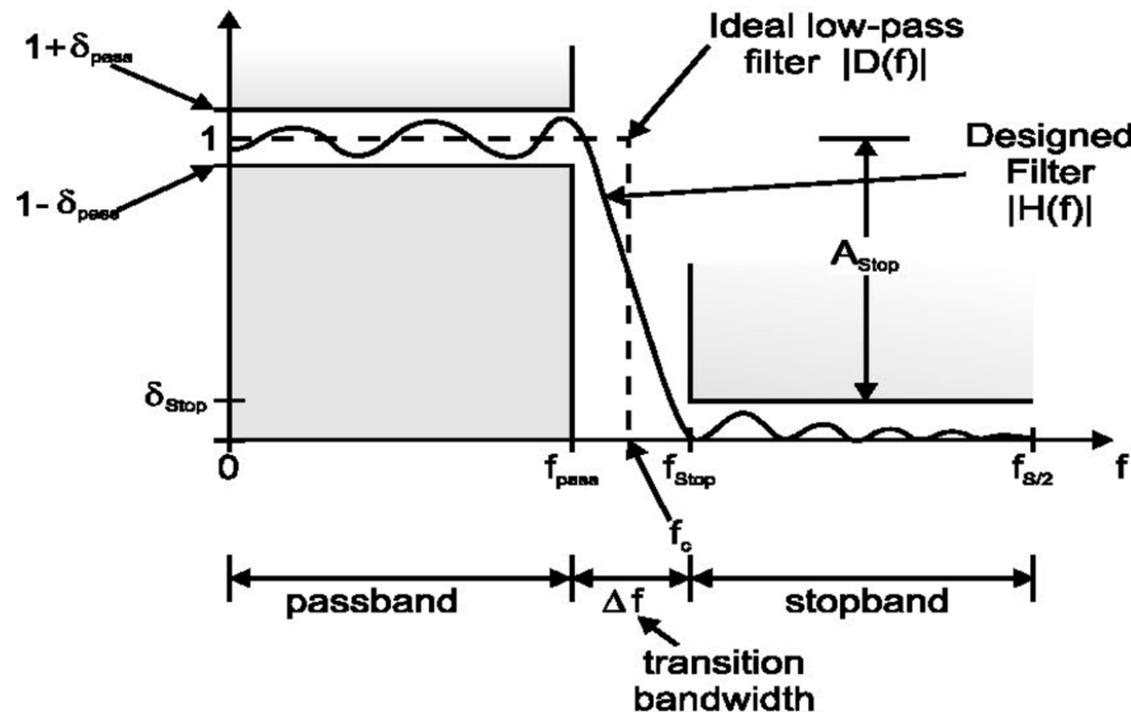
Note that when you open Filter Designer, Design Filter is not enabled. You must make a change to the default filter design in order to enable Design Filter. This is true each time you want to change the filter design. Changes to radio button items or drop down menu items such as those under Response Type or Filter Order enable Design Filter immediately. Changes to specifications in text boxes such as F_s , F_{pass} , and F_{stop} require you to click outside the text box to enable Design Filter.

Choosing a Response Type

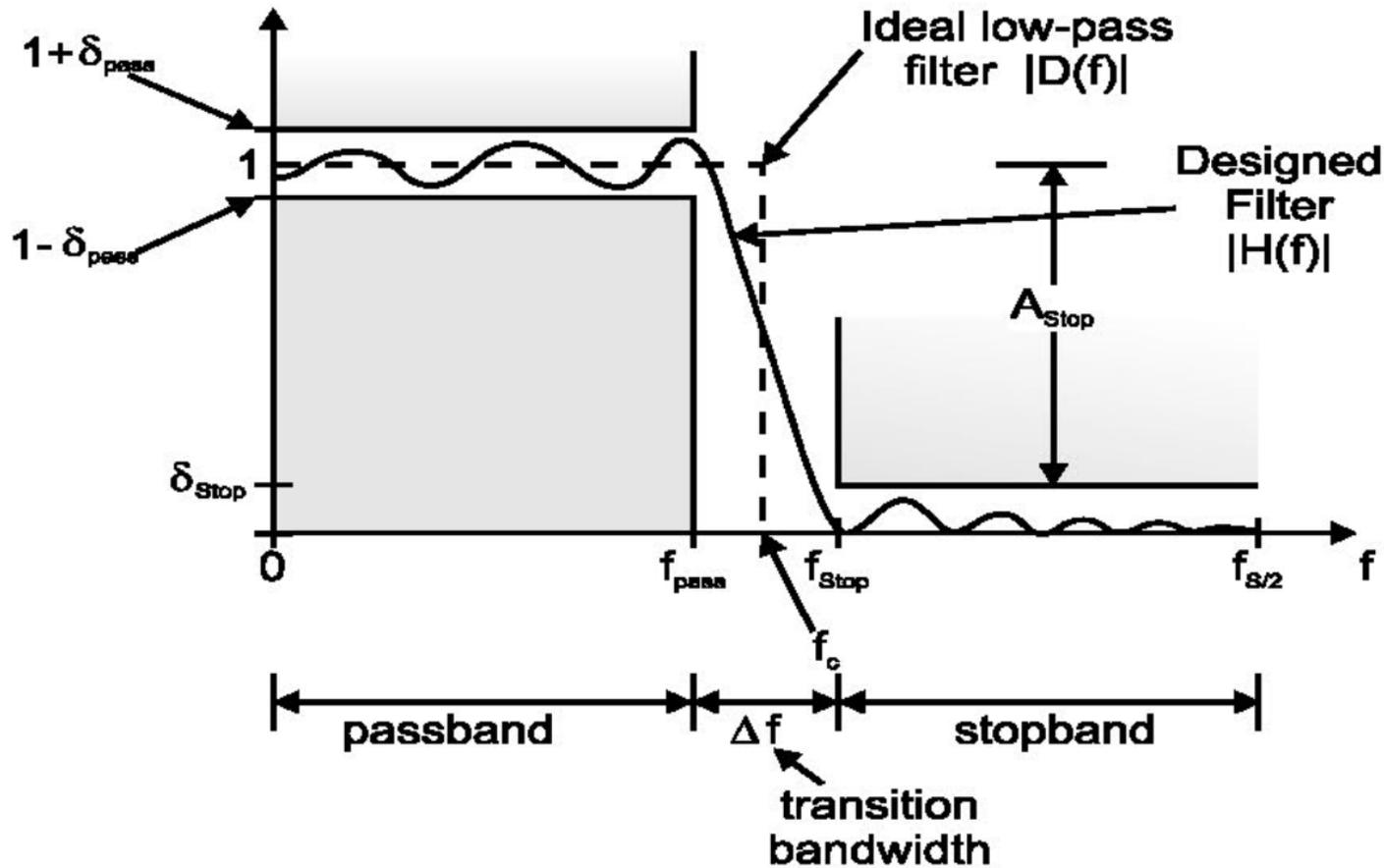
Especificações para projeto de filtros FIR

Geralmente, as principais especificações desejadas para a resposta em frequência $H(\omega)$ de um filtro passa-baixa são conforme mostra a figura a seguir.

Estas especificações são basicamente válidas também para filtros passa-alta, passa-banda e rejeita-banda (eventualmente, com pequenas adaptações ao contexto de cada filtro).



Especificações para projeto de filtros FIR



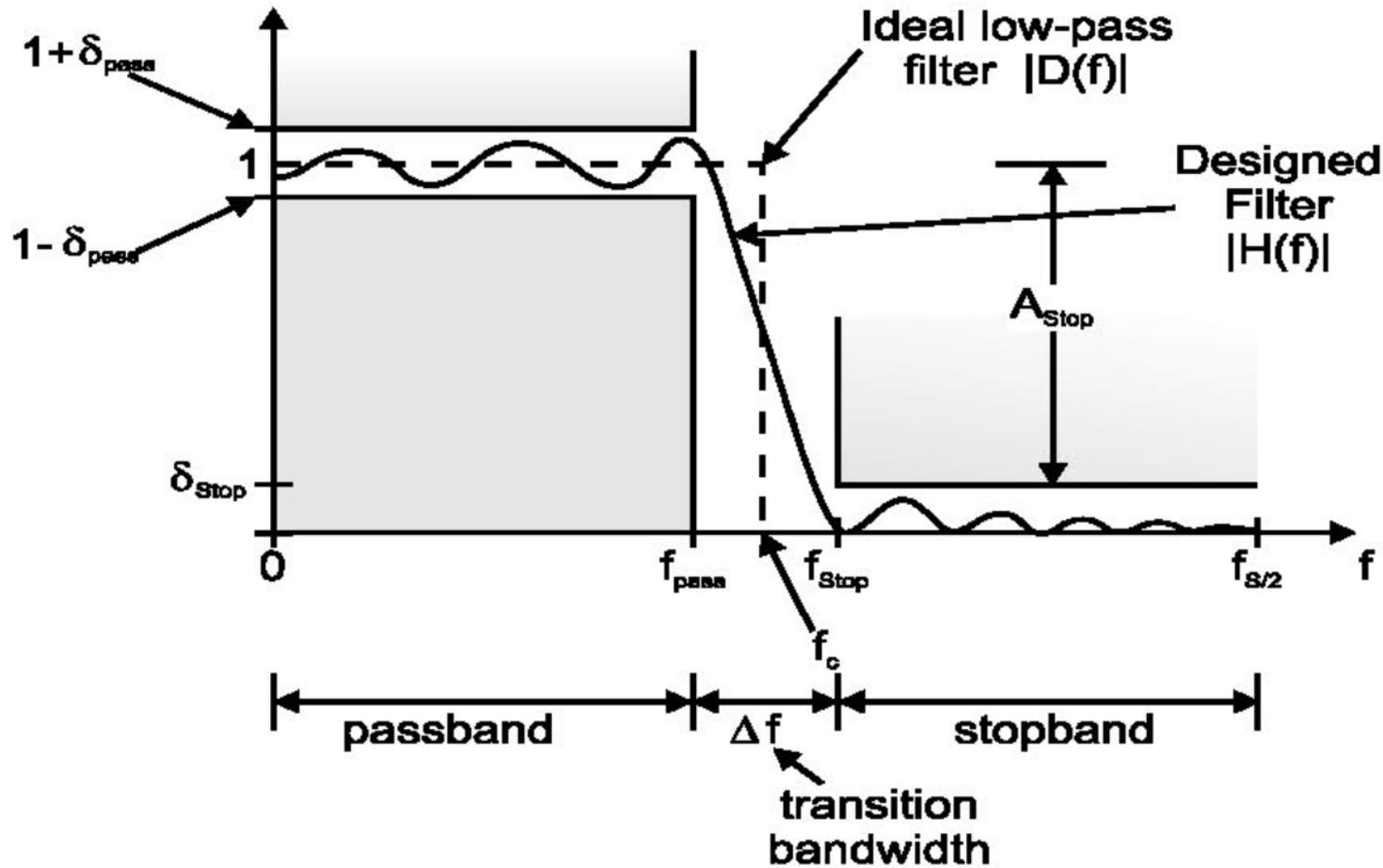
A frequência de corte f_c do filtro passa-baixas ideal original ocorre no ponto médio entre as frequências f_{pass} na borda da faixa de passagem e f_{stop} na borda da faixa de rejeição:

$$f_c = \frac{1}{2} (f_{pass} + f_{stop})$$

A largura de transição é definida por

$$\Delta f = f_{stop} - f_{pass}$$

Especificações para projeto de filtros FIR



As frequências normalizadas são as frequências digitais:

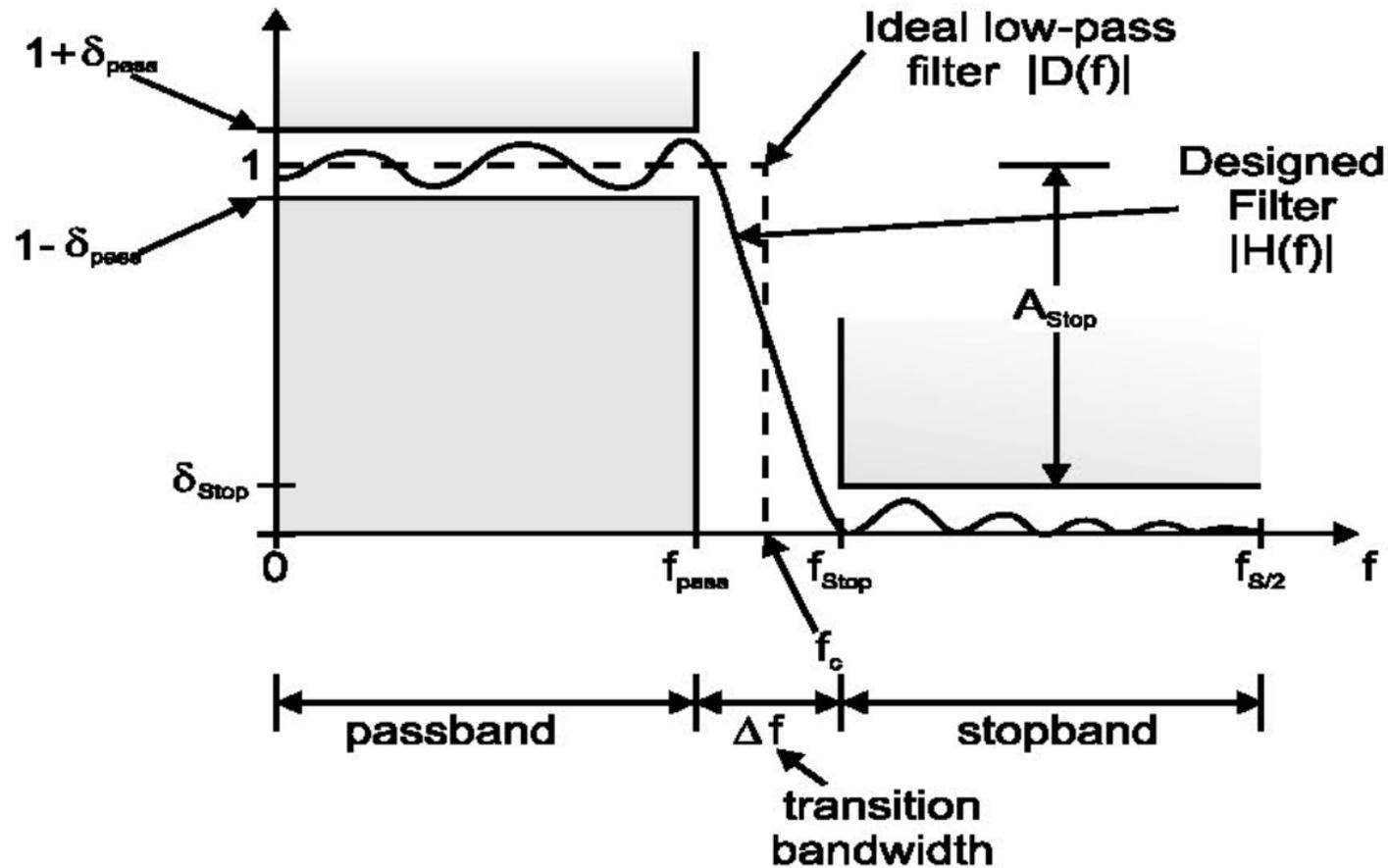
$$\omega_{\text{pass}} = \frac{2\pi f_{\text{pass}}}{f_s}$$

$$\omega_{\text{stop}} = \frac{2\pi f_{\text{stop}}}{f_s}$$

$$\omega_c = \frac{2\pi f_c}{f_s}$$

$$\Delta\omega = \frac{2\pi\Delta f}{f_s}$$

Especificações para projeto de filtros FIR



Os *ripples máximos* desejados nas faixas de passagem e de rejeição são usualmente expressos em decibéis (dB).

$$A_{\text{pass}} = 20 \log_{10} \left(\frac{1 + \delta_{\text{pass}}}{1 - \delta_{\text{pass}}} \right)$$

$$A_{\text{stop}} = -20 \log_{10} \delta_{\text{stop}}$$

Especificações para projeto de filtros FIR

Note que as equações vistas acima relacionam dois conjuntos de especificações, $\{f_{pass}, f_{stop}, A_{pass}, A_{stop}\}$ e $\{f_c, \Delta f, \delta_{pass}, \delta_{stop}\}$.

Uma propriedade desejável para a resposta em frequência $|H(\omega)|$ do filtro resultante obtido por qualquer método de janelamento é que δ_{pass} e δ_{stop} de $|H(\omega)|$ sejam iguais. Portanto, ao invés de lidar com as duas variáveis, podemos definir como *ripple* máximo, o menor dos dois

$$\delta = \text{mín}(\delta_{pass}, \delta_{stop})$$

Na prática da especificação de filtros, a atenuação A_{pass} na faixa de passagem e para a atenuação A_{stop} na faixa de rejeição são especificadas de forma a resultar um *ripple* na faixa de passagem menor do que o *ripple* na faixa de rejeição.

Filtros IIR – *Infinite Impulse Response*

➔ Filtros IIR são filtros cuja resposta ao impulso é de duração infinita.

- A relação entre a entrada e a saída de um filtro IIR é dada pela fórmula recursiva

$$y(n) = -\sum_{k=1}^N a(k)y(n-k) + \sum_{k=0}^M b(k)x(n-k) \quad (2)$$

- A amostra de saída no instante n é calculada a partir das $M + 1$ amostras de entrada (anteriores e atual), e das N amostras de saída anteriores.
- O segundo termo do lado direito da equação (2) tem a mesma forma dos filtros FIR.
- O primeiro termo do lado direito da equação (2) corresponde ao feedback, ou à parte recursiva da equação, que faz com que a resposta ao impulso de um filtro IIR possa, teoricamente, resultar com duração infinita.
- Esta é a razão pela qual os filtros IIR também são chamados de filtros recursivos.

Filtros IIR – *Infinite Impulse Response*

 Assim como os filtros FIR, os filtros IIR são sistemas lineares de tempo discreto, cuja sequência de saída está relacionada com a sequência de entrada e a resposta ao impulso do filtro pela operação de convolução, conforme

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) \quad (3)$$

- O que torna filtros IIR diferentes dos filtros FIR é que o limite superior da soma na equação acima é infinito, dado que a resposta ao impulso $h(n)$ tem duração infinita.
- Obviamente, computar a referida convolução é impraticável.
- A relação recursiva definida na equação (2) é muito mais eficiente e prática, requerendo $(N + M + 1)$ multiplicações e $(N + M)$ adições.

$$y(n) = -\sum_{k=1}^N a(k)y(n-k) + \sum_{k=0}^M b(k)x(n-k) \quad (2)$$

Filtros IIR – *Infinite Impulse Response*

Vimos que as amostras de entrada e saída de um filtro IIR estão relacionadas pela equação recursiva

$$y(n) = -\sum_{k=1}^N a(k)y(n-k) + \sum_{k=0}^M b(k)x(n-k)$$

que pode ser rearranjada conforme

$$\sum_{k=0}^N a(k)y(n-k) = \sum_{k=0}^M b(k)x(n-k)$$

com $a(0) = 1$. Alternativamente, podemos reescrevê-la no domínio frequência z como

$$\sum_{k=1}^N a(k) \left[z^{-k} y(n) \right] = \sum_{k=0}^M b(k) \left[z^{-k} x(n) \right]$$

onde z^{-k} representa o atraso de k amostras no domínio tempo discreto, conforme visto no Capítulo V das notas de aula.

Filtros IIR – *Infinite Impulse Response*

Assim, como $z^{-k}x(n) = x(n-k)$ e $\sum_{k=1}^N a(k)[z^{-k}y(n)] = \sum_{k=0}^M b(k)[z^{-k}x(n)]$,

podemos definir uma função de transferência $H(z)$ no domínio frequência z como a razão de polinômios

$$H(z) = \frac{y(n)}{x(n)} = \frac{\sum_{k=0}^M b(k)z^{-k}}{\sum_{k=0}^N a(k)z^{-k}} = \frac{b(0) + b(1)z^{-1} + \dots + b(M)z^{-M}}{1 + a(1)z^{-1} + a(2)z^{-2} + \dots + a(N)z^{-N}}$$

Note que a $H(z)$ acima definida é a razão entre a Transformada \mathcal{Z} da saída $y(n)$ do filtro e a Transformada \mathcal{Z} da entrada $x(n)$, conforme visto no Capítulo V das notas de aula.

A resposta em frequência $H(\omega)$ está relacionada com $H(z)$ por $z = e^{j\omega}$, em que o período de amostragem T é normalizado para 1 segundo.

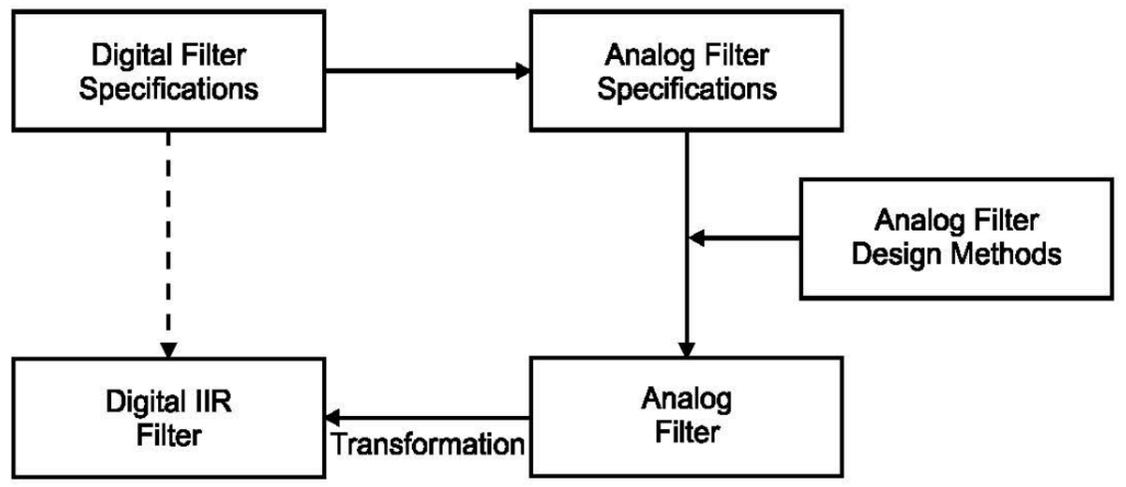
Substituindo $z = e^{j\omega}$ na equação para $H(z)$ obtém-se a resposta em frequência normalizada.

Filtros IIR – *Infinite Impulse Response*

- Ao contrário dos filtros FIR, não é possível obter fase linear exata na resposta em frequência de filtros IIR. Isto ocorre porque, conforme vimos no estudo de filtros FIR, fase linear exata implica em que a resposta ao impulso do filtro seja simétrica no tempo discreto, o que é impossível de se obter com um filtro IIR em consequência de sua operação recursiva sobre as amostras de saída. No entanto, existem filtros IIR cuja curva de fase da resposta em frequência é aproximadamente linear.
- Filtros de fase linear nos permitem remover considerações da fase da resposta em frequência durante a etapa de projeto do filtro. Como os filtros IIR não podem ter fase linear, o seu projeto é mais complexo. As equações que precisam ser resolvidas geralmente não são lineares, e técnicas de otimização são necessárias.
- A maioria das técnicas clássicas de projeto de filtros IIR assumem que a fase da resposta em frequência não é importante e apenas aproximam a magnitude da resposta em frequência.
- Sendo irrelevante a fase da resposta em frequência, então, para filtros com a mesma especificação da magnitude da resposta em frequência, a ordem necessária para um filtro IIR é normalmente inferior à exigida para um filtro FIR. Esta é uma vantagem, uma vez que uma ordem inferior implica em um menor custo computacional e um menor atraso no domínio tempo.

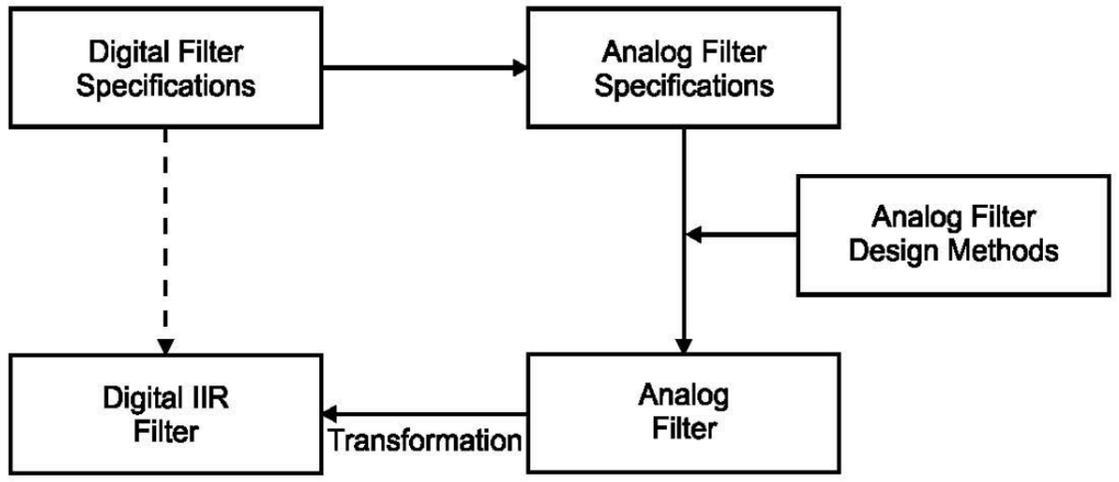
Filtros IIR – *Infinite Impulse Response*

Uma abordagem clássica para o projeto de filtros digitais IIR é baseada no correspondente filtro analógico cuja resposta ao impulso é contínua no domínio tempo. Esta abordagem é ilustrada na figura abaixo.



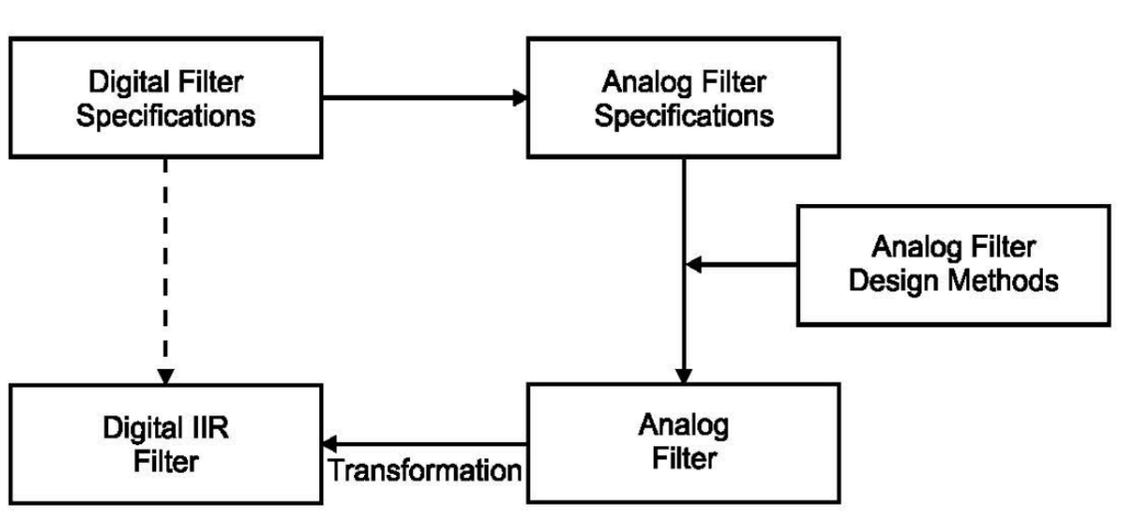
- O problema de projetar um filtro digital IIR é transformado no projeto de um filtro analógico equivalente.
- O projeto do filtro analógico é então resolvido usando técnicas clássicas de projeto de filtros analógicos.
- A resposta impulsiva $h(t)$ do filtro analógico projetado é amostrada na frequência de amostragem $f_s = 1/T_s$ no domínio tempo contínuo t , resultando na resposta ao impulso $h(nT_s) = h(n)$ no domínio tempo discreto.

Filtros IIR – *Infinite Impulse Response*



- É então aplicada a Transformada \mathcal{Z} sobre a resposta ao impulso $h(n)$, obtendo-se a função de transferência do filtro digital $H(z)$ no domínio frequência z como uma razão de polinômios no domínio frequência z , conforme mostrado no slide 62, notando que z^{-k} representa o atraso de k amostras no domínio tempo discreto.
 - Isto, portanto, já define a arquitetura do *hardware* do filtro em termos de uma combinação linear de amostras atrasadas de z^{-k} no domínio tempo discreto (vide $H(z)$ no slide 62). Cada atraso z^{-1} é um registrador no *hardware* que transfere para a sua saída o valor da amostra em sua entrada após transcorrido o intervalo T_s .
- A resposta em frequência $H(\omega)$ normalizada é obtida de $H(z)$ fazendo-se $z = e^{j\omega}$, em que o período de amostragem T_s é normalizado para 1 segundo, resultando em uma faixa de frequência normalizada no intervalo $[0.0, 1.0]$, onde 1.0 representa a frequência de Nyquist $1e^{j\pi}$ sobre o círculo de raio unitário no domínio frequência z ou $f_s/2$ sobre o eixo $j\omega$ no domínio s .

Filtros IIR – *Infinite Impulse Response*



- Neste caso, devemos notar que a sequência de procedimentos para o projeto de um filtro IIR a partir do projeto de um filtro analógico equivalente é bastante extensa e enfadonha, a começar pelo projeto do filtro analógico em si.
- Também neste contexto, é usual a utilização de técnicas computacionais para o projeto de filtros IIR (como também para o projeto de filtros FIR).

- A sugestão aqui é que sejam utilizadas ferramentas de *software*, especificamente as ferramentas “filterDesigner” (ou “fdatool”, dependendo da versão do Matlab) para o projeto de filtros.
- Desta forma, nos slides que seguem, concentraremos nosso estudo nas propriedades e características dos tipos de filtro analógicos que serão a base do filtro digital deles resultante.

Filtros IIR – *Infinite Impulse Response*

Os filtros analógicos considerados nesta abordagem são:

- (1) Butterworth
- (2) Chebyshev
- (3) Chebyshev inverso

Concentraremos o estudo em filtros passa-baixa normalizados. Sendo desejado, um filtro passa-baixa pode ser transformado para passa-alta, passa-banda, ou rejeita-banda (transformação disponibilizada na ferramenta filterDesigner do Matlab).

Filtro Butterworth

A resposta em frequência $H(\omega)$ normalizada quadrática (representando ganho de potência ao invés de ganho de tensão) da aproximação de Butterworth é

$$|H(\omega)|^2 = \frac{1}{1 + c\omega^{2n}}$$

para um filtro de $n^{\text{ésima}}$ ordem. A constante c determina em qual frequência ω ocorre a transição da faixa de passagem para a faixa de rejeição.

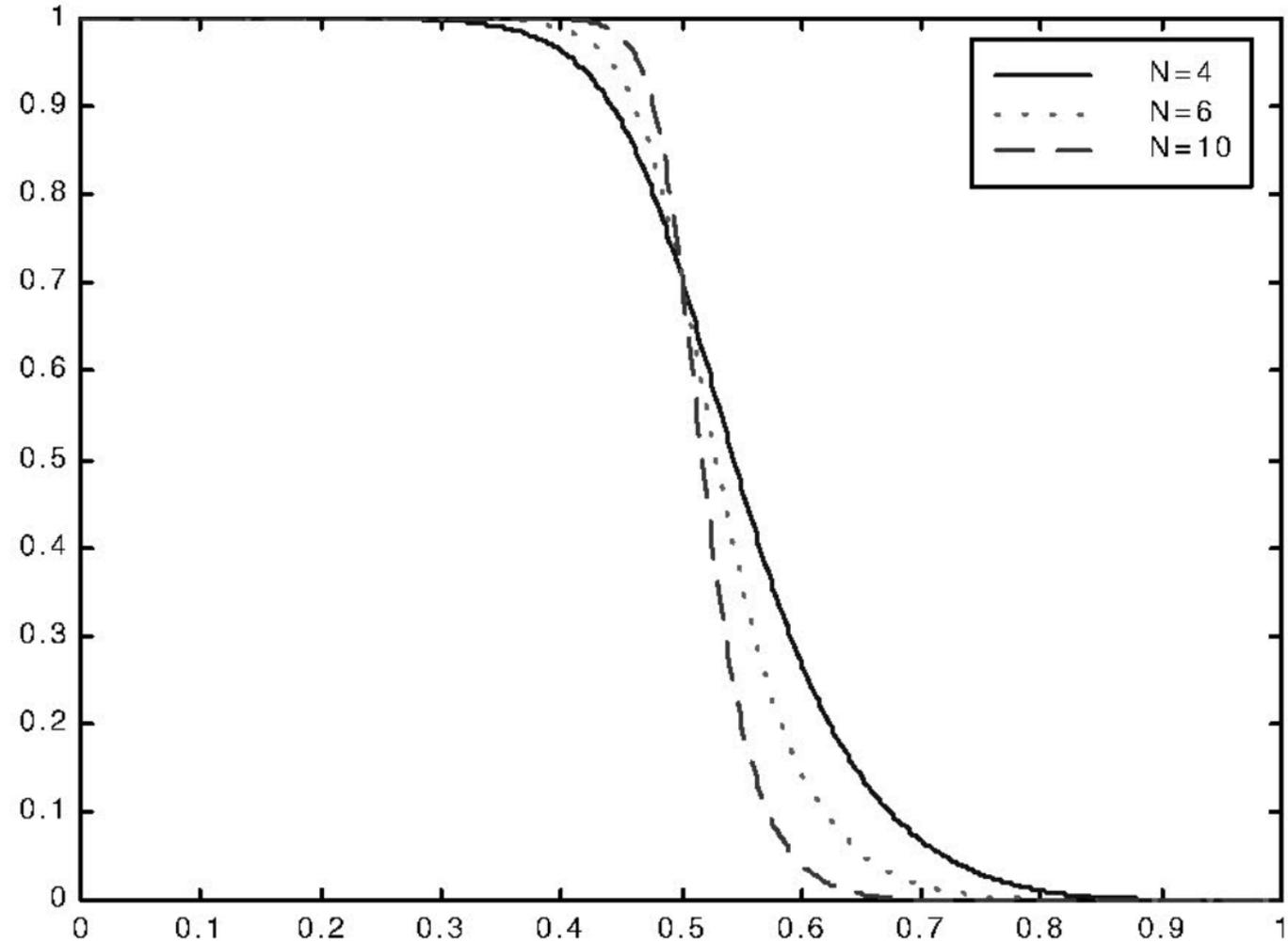
Para filtros Butterworth normalizados, este ponto está em $\omega = 1$ e, desta forma, $c = 1$.

Esta frequência é também referida como a frequência de corte.

Filtro Butterworth

A figura ao lado apresenta as respostas de magnitude de vários filtros Butterworth de diferentes ordens N .

Os filtros Butterworth são às vezes chamados de filtros planos, porque a resposta nas frequências $\omega = 0$ e $\omega = 1$ são completamente planas (ou horizontais).



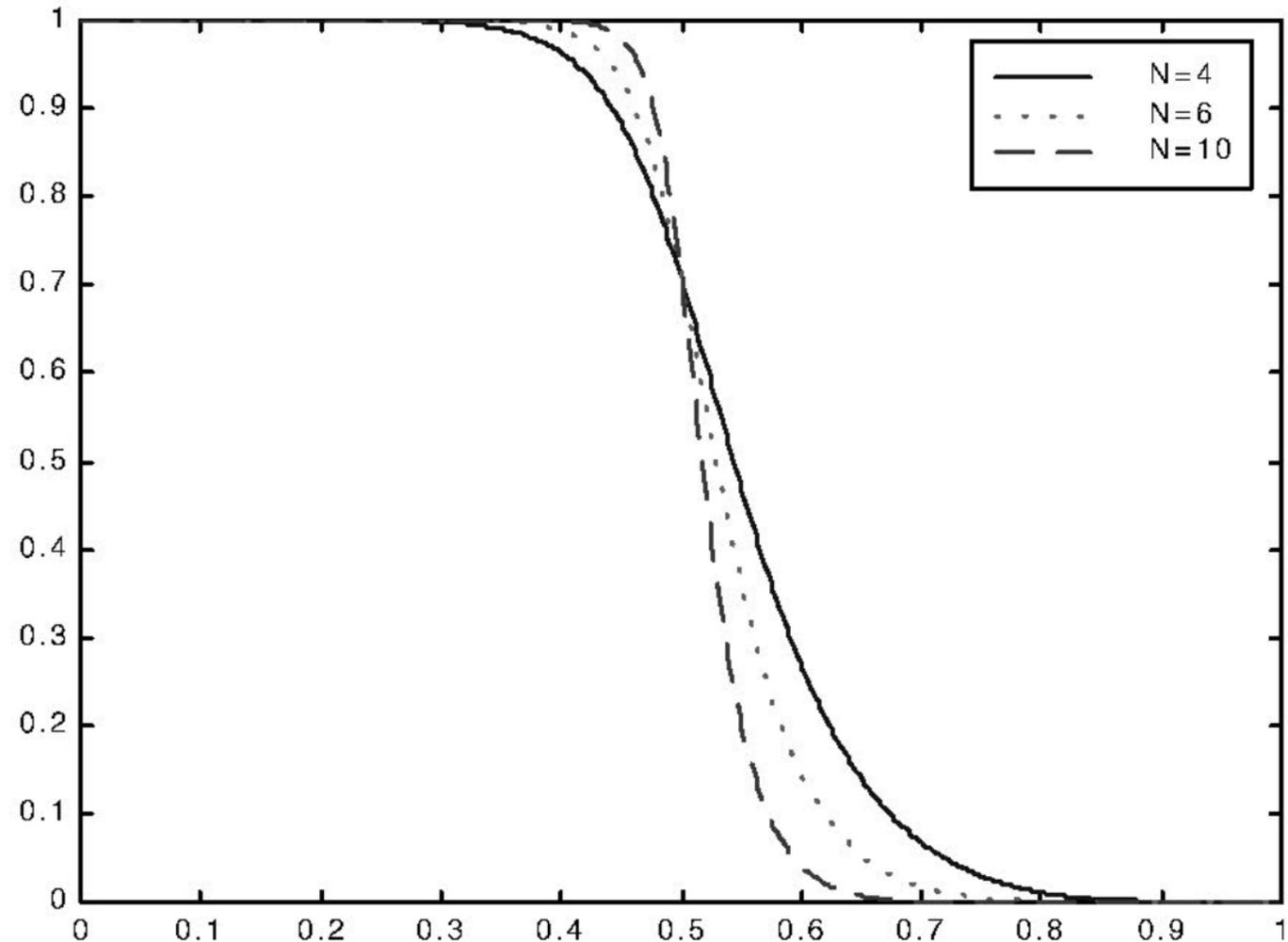
Resposta em magnitude de um filtro Butterworth

Filtro Butterworth

Nestas frequências, a resposta de Butterworth é muito próxima da resposta de um filtro ideal.

Esta resposta plana na faixa de passagem e na faixa de rejeição é obtida às custas da largura da faixa de transição, que é consideravelmente maior do que a das outras aproximações clássicas (Chebyshev, etc...).

Na aproximação Butterworth, o *roll-off* da faixa de passagem para a faixa de rejeição é relativamente lento, e a resposta de fase próxima da frequência de corte não é linear. Nas demais frequências, a resposta da fase é suave.



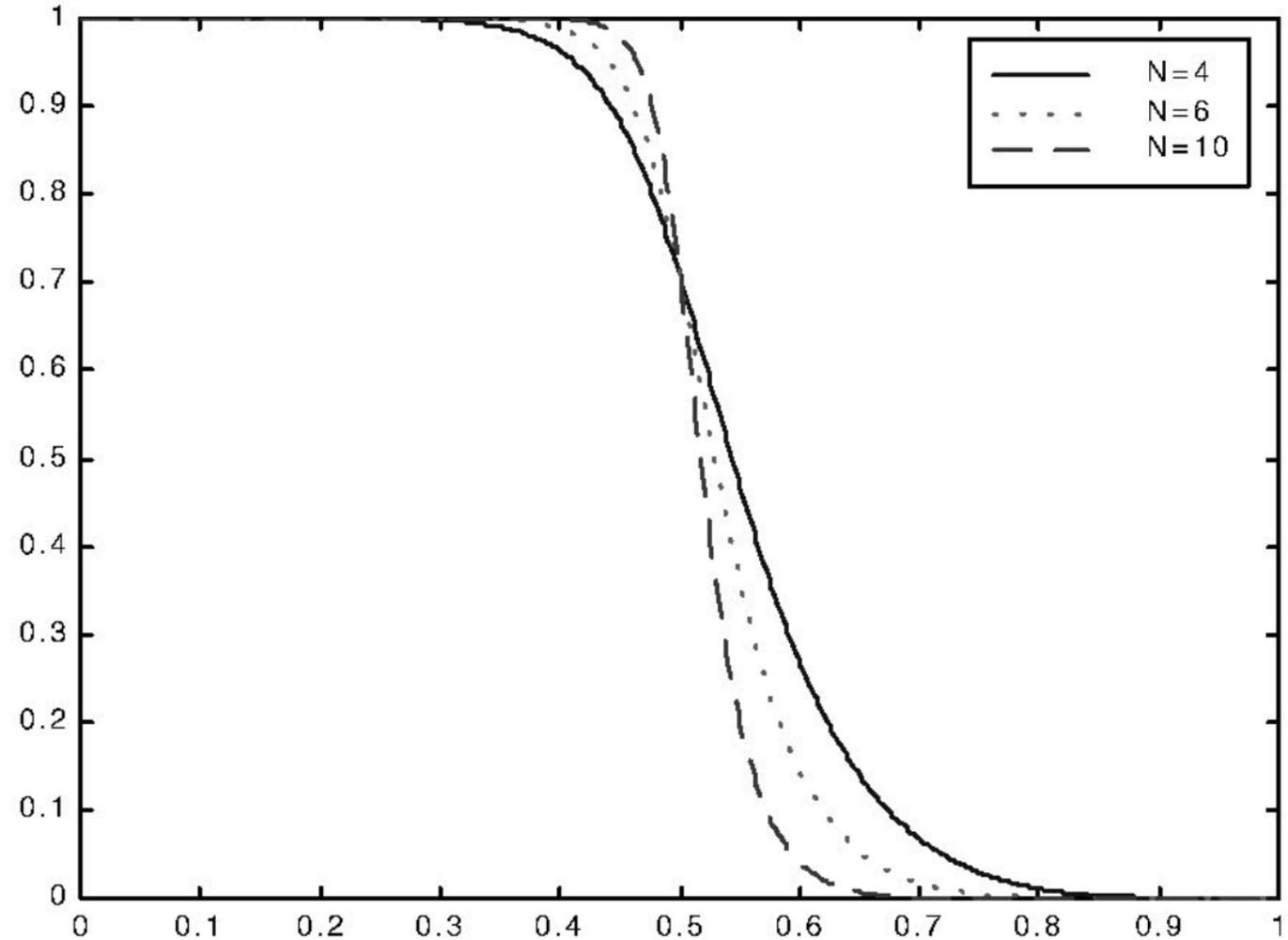
Resposta em magnitude de um filtro Butterworth

Filtro Butterworth

A parte mais importante do projeto de filtro Butterworth é determinar a ordem do filtro necessária para satisfazer as especificações.

Uma vez determinada a ordem do filtro, os coeficientes do filtro podem ser determinados ou encontrados em tabelas.

Os coeficientes do filtro são, ainda, facilmente determinados pela ferramenta “filterDesigner” do Matlab.



Resposta em magnitude de um filtro Butterworth

Filtro Butterworth

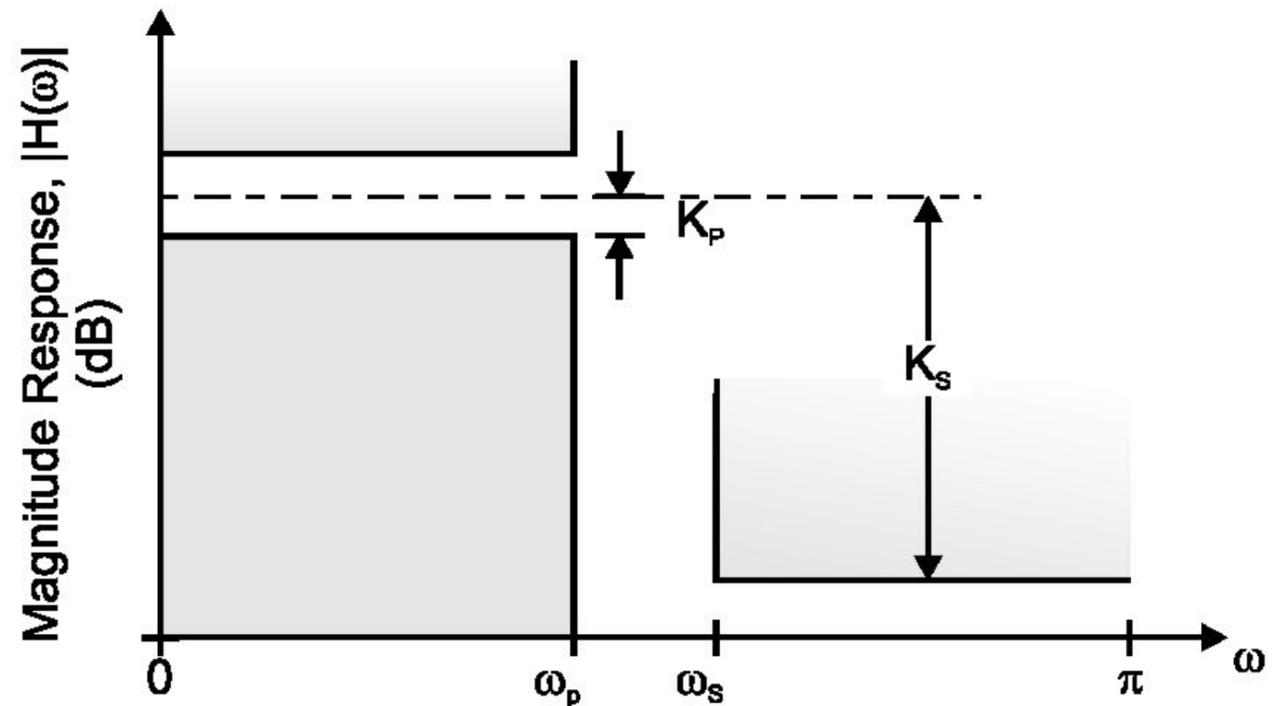
Uma especificação típica para a máscara da resposta de magnitude de um filtro a ser projetado é mostrada na figura abaixo.

A ordem do filtro Butterworth requerida é dada por

$$n \geq \frac{\log_{10} M}{\log_{10} \Omega} = \frac{\ln M}{\ln \Omega} \quad \text{onde}$$

$$\Omega = \frac{\omega_s}{\omega_p} \quad \text{e} \quad M = \sqrt{\frac{10^{0.1K_s} - 1}{10^{0.1K_p} - 1}}$$

e K_s e K_p estão em decibéis (dB).



Especificação típica da resposta em magnitude

Filtro Butterworth - Exemplo

Determine a ordem de um filtro Butterworth que atenda às seguintes especificações:

$$\omega_p = 1 \text{ rad/s}$$

$$\omega_s = 1.3 \text{ rad/s}$$

$$K_s = 22 \text{ dB}$$

$$K_p = 3.0103 \text{ dB}$$

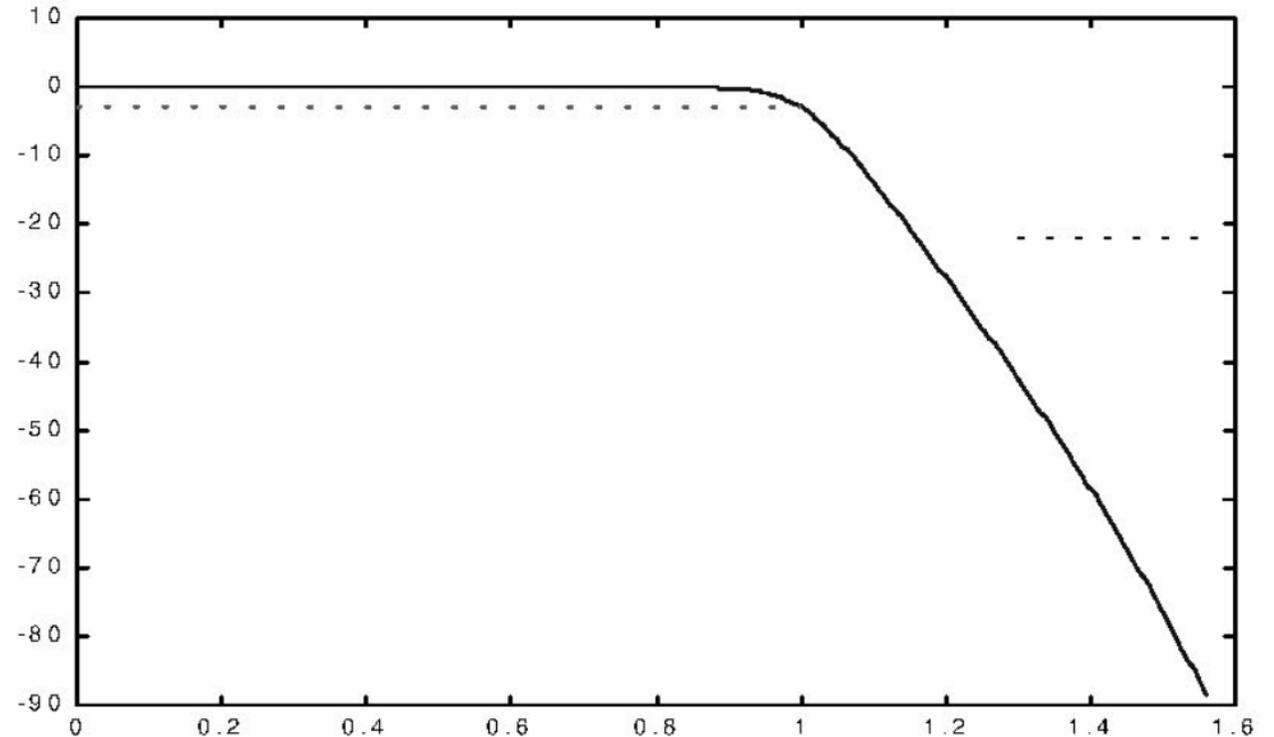
Usando as equações anteriormente vistas,

$$\Omega = \frac{1.3}{1.0} = 1.3$$

$$M = \sqrt{\frac{10^{22} - 1}{10^{0.30103} - 1}} = 12.5495$$

$$n \geq \frac{\ln(12.5495)}{\ln(1.3)} = 9.6419$$

Então, precisamos de um filtro Butterworth pelo menos de 10ª ordem.



Resposta de magnitude do filtro Butterworth de 10th ordem. Note que a normalização aqui é tal que 1.0 na abscissa do gráfico acima corresponde à frequência de corte do filtro.

Filtro Chebyshev

- O filtro Butterworth, embora possua algumas propriedades desejáveis, não fornece aproximação suficientemente boa perto da borda da faixa de passagem.
- O *roll-off* da faixa de passagem para a banda de rejeição também é relativamente gradual, o que é indesejado (o ideal é uma transição abrupta bem definida).
- Portanto, para aplicações de filtros com faixa de transição estreita, é necessário um filtro Butterworth de alta ordem.
- Se a aplicação puder tolerar algum *ripple* na faixa de passagem, um filtro Chebyshev irá superar estes problemas associados aos filtros Butterworth.
- A resposta quadrática de magnitude dos filtros Chebyshev é dada por

$$|H(\omega)|^2 = \frac{1}{1 + \varepsilon^2 C_N^2(\omega)}$$

onde $C_N(\omega)$ é um polinômio de Chebyshev de ordem N e ε é um parâmetro que é associado à amplitude do *ripple*.

Filtro Chebyshev

A figura abaixo apresenta a magnitude da resposta de 3 filtros Chebyshev de ordens diferentes, para o mesmo ϵ . O polinômio de Chebyshev de $N^{\text{ésima}}$ ordem é dado por

$$C_n(\omega) = \cos(n \cos^{-1} \omega) \quad \text{for } 0 \leq \omega \leq 1$$

$$C_n(\omega) = \frac{1}{2} \left[\left(\omega + \sqrt{\omega^2 - 1} \right)^n + \left(\omega - \sqrt{\omega^2 - 1} \right)^n \right] \quad \text{for } \omega > 1$$

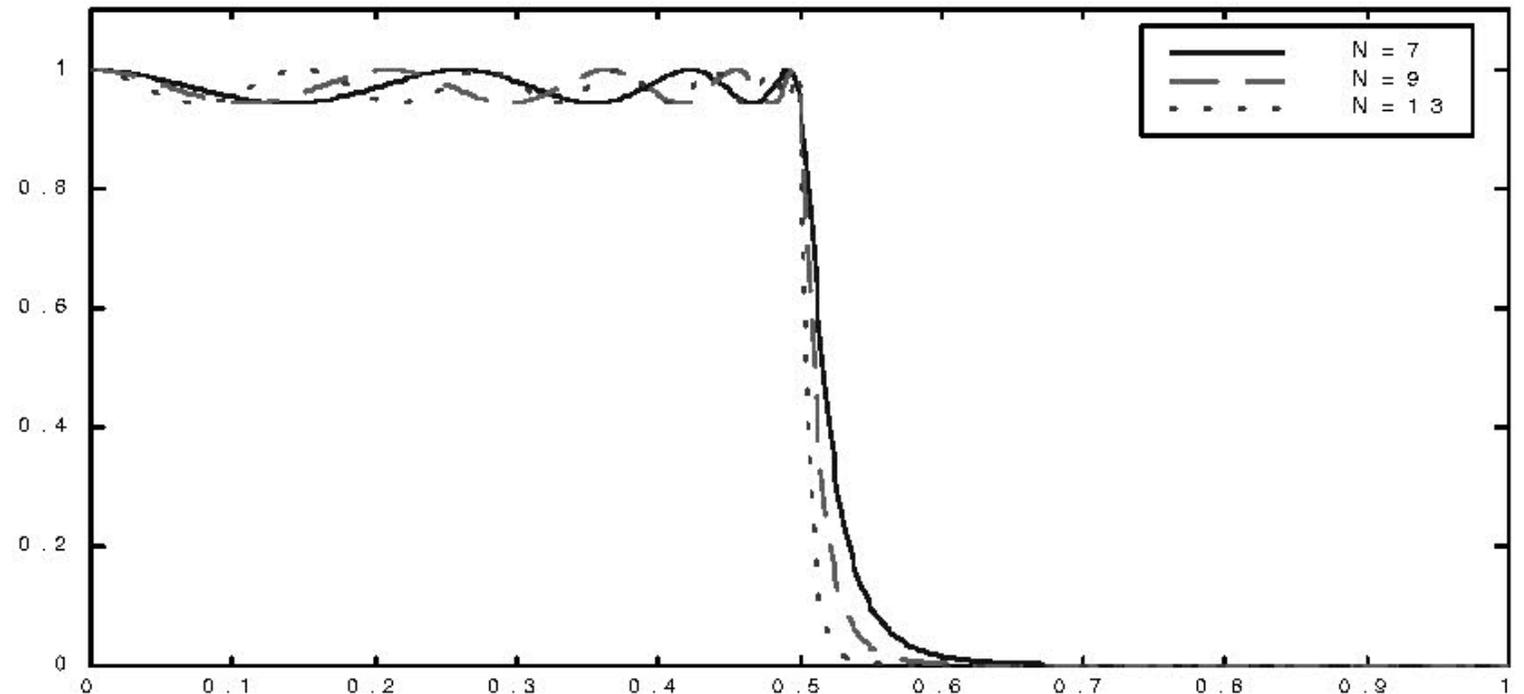
Resposta em magnitude de um filtro Chebyshev

Alternativamente, estes polinômios podem ser definidos recursivamente como

$$C_1(\omega) = \omega$$

$$C_2(\omega) = 2\omega^2 - 1$$

$$C_{n+1}(\omega) = 2\omega C_n(\omega) - C_{n-1}(\omega)$$



Filtro Chebyshev

- Assim como nos filtros FIR, a aproximação de Chebyshev minimiza o erro máximo sobre a banda de passagem, e é ótima neste sentido.
- A ordem N do filtro determina a largura da faixa de transição e o número de oscilações do *ripple* dentro da faixa de passagem.
- O projeto de filtros Chebyshev envolve a determinação tanto da ordem N quanto do parâmetro ϵ associado à amplitude do *ripple*.

Filtro Chebyshev

Consideremos as especificações genéricas para a máscara da resposta de magnitude de um filtro passa-baixa conforme figura abaixo. Os procedimentos são os seguintes:

A amplitude máxima do *ripple* permitida na faixa de passagem é d ou, em decibéis, K_p .

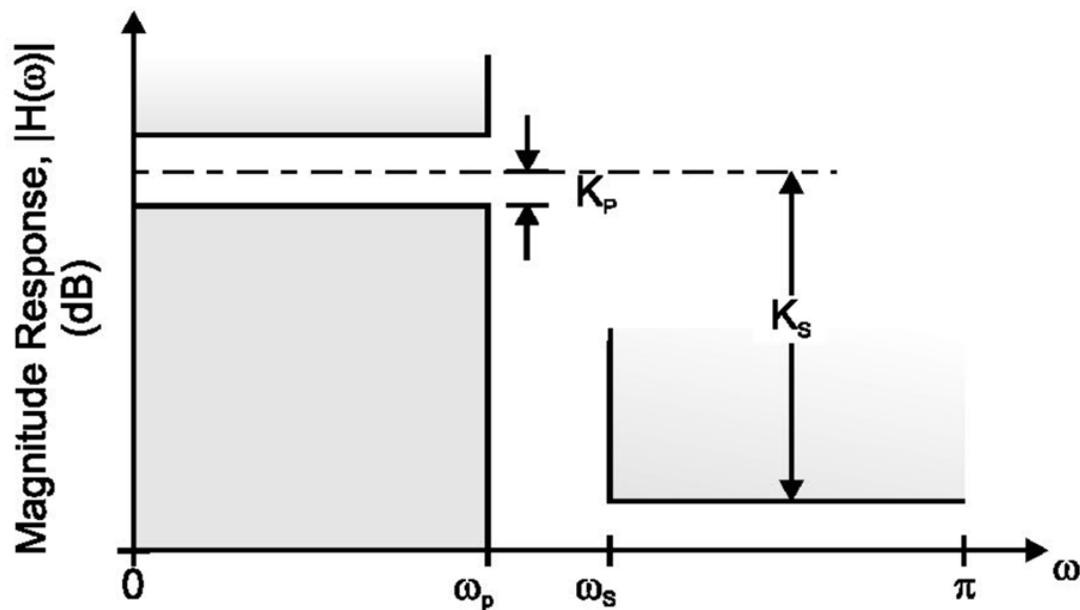
$$d = 1 - 10^{-K_p/20}$$

ε é calculado a partir de d ou K_p :

$$K_p = 10 \log(1 + \varepsilon^2)$$

$$d = 1 - \frac{1}{\sqrt{1 + \varepsilon^2}}$$

$$\varepsilon = \sqrt{\frac{2d - d^2}{1 - 2d + d^2}} = \sqrt{10^{K_p/10} - 1}$$



A ordem N do filtro é determinada usando as expressões ao lado \rightarrow

$$\Omega = \frac{\omega_s}{\omega_p}$$

$$M = \sqrt{\frac{10^{0.1 K_s} - 1}{10^{0.1 K_p} - 1}}$$

$$N = \frac{\cosh^{-1} M}{\cosh^{-1} \Omega}$$

Filtro de Chebyshev - Exemplo

Determine o filtro de Chebyshev que atenda às seguintes especificações de filtro passa-baixa (idênticas às especificações do filtro Butterworth do exemplo anterior):

$$\omega_p = 1 \text{ rad/s}$$

$$\omega_s = 1.3 \text{ rad/s}$$

$$K_s = 22 \text{ dB}$$

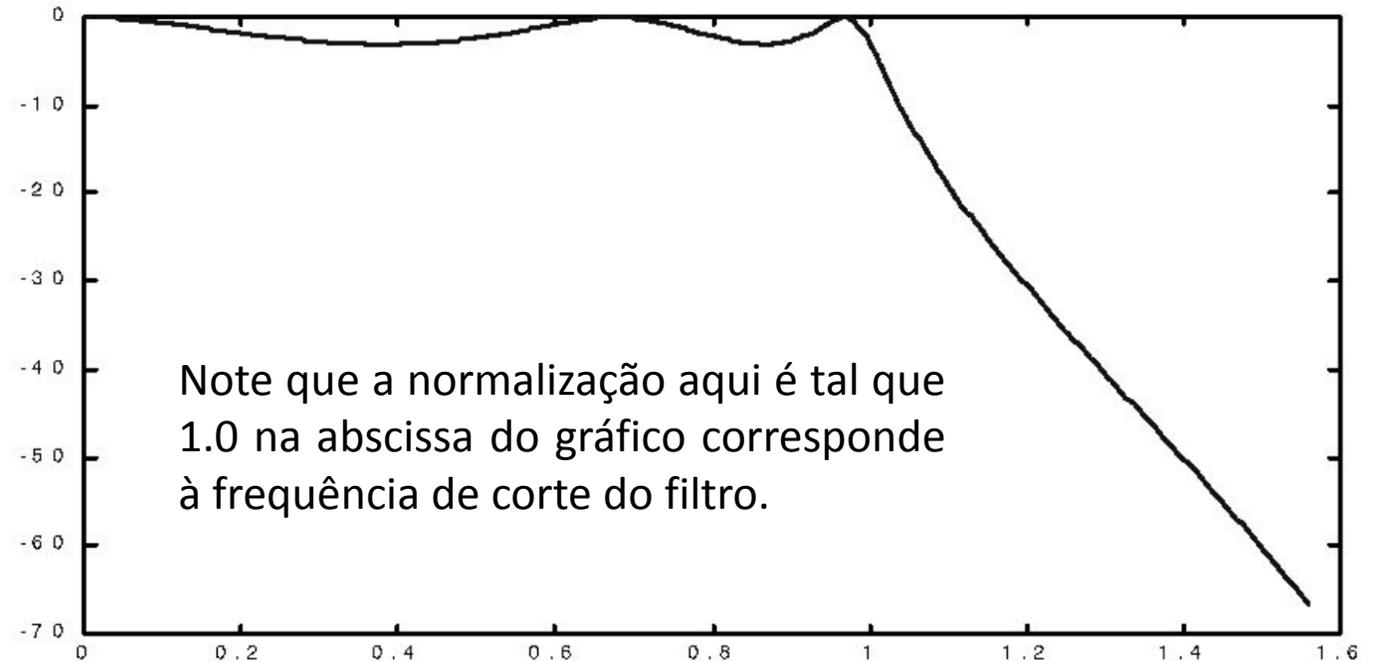
$$K_p = 3.0103 \text{ dB}$$

Usando as equações do slide anterior:

$$\Omega = 1.3$$

$$M = 12.5495$$

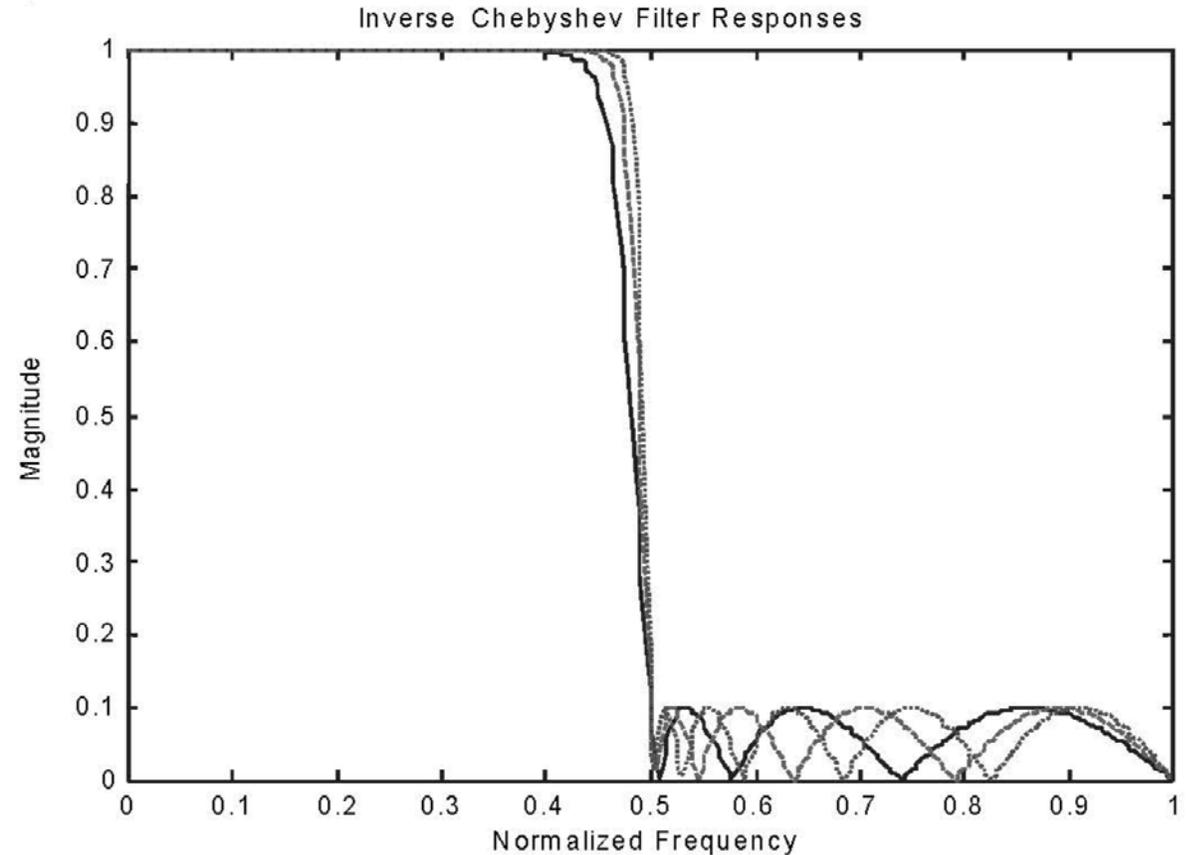
$$N \geq \frac{\cosh^{-1} 12.5495}{\cosh^{-1} 1.3} = \frac{3.2212}{0.7564} = 4.2585$$



Assim, a ordem mínima do filtro Chebyshev é $N = 5$, para uma amplitude máxima de *ripple* $K_p = 3 \text{ dB}$ permitida na faixa de passagem. A resposta do filtro é mostrada na figura acima.

Aproximação inversa de Chebyshev

- Uma alternativa à aproximação de Chebyshev é a aproximação inversa de Chebyshev.
- Como o nome indica, o comportamento da resposta em frequência é inverso ao comportamento da aproximação de Chebyshev.
- Um filtro de Chebyshev inverso tem uma faixa de passagem plana e *ripple* na faixa de rejeição, conforme ilustra a figura ao lado.



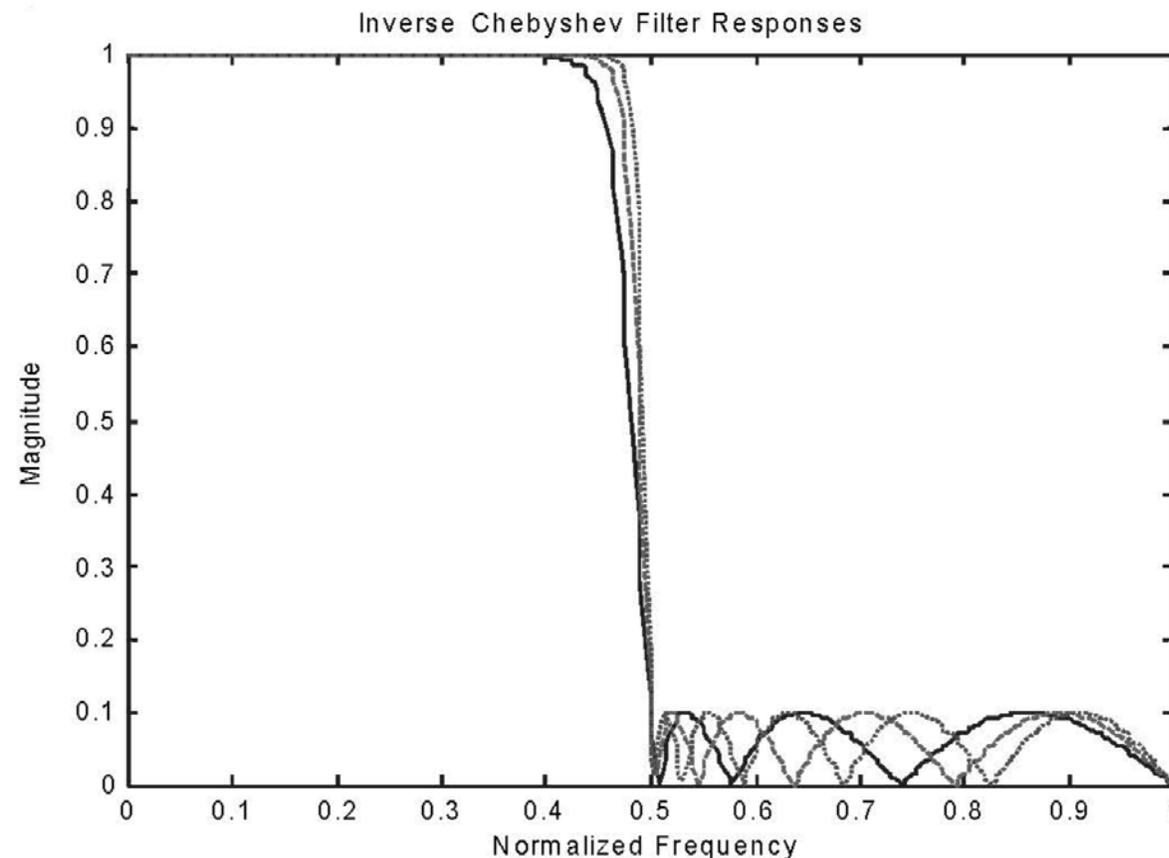
Inverse Chebyshev filter responses

Aproximação inversa de Chebyshev

- As respostas quadráticas de magnitude de vários filtros com ordens diferentes são mostradas na figura ao lado.
- A resposta quadrática de magnitude para um filtro Chbyshev inverso é expressa por

$$|H(\omega)|^2 = \frac{\epsilon^2 C_N^2(1/\omega)}{1 + \epsilon^2 C_N^2(1/\omega)}$$

para um filtro de $N^{\text{ésima}}$ ordem.



Inverse Chebyshev filter responses

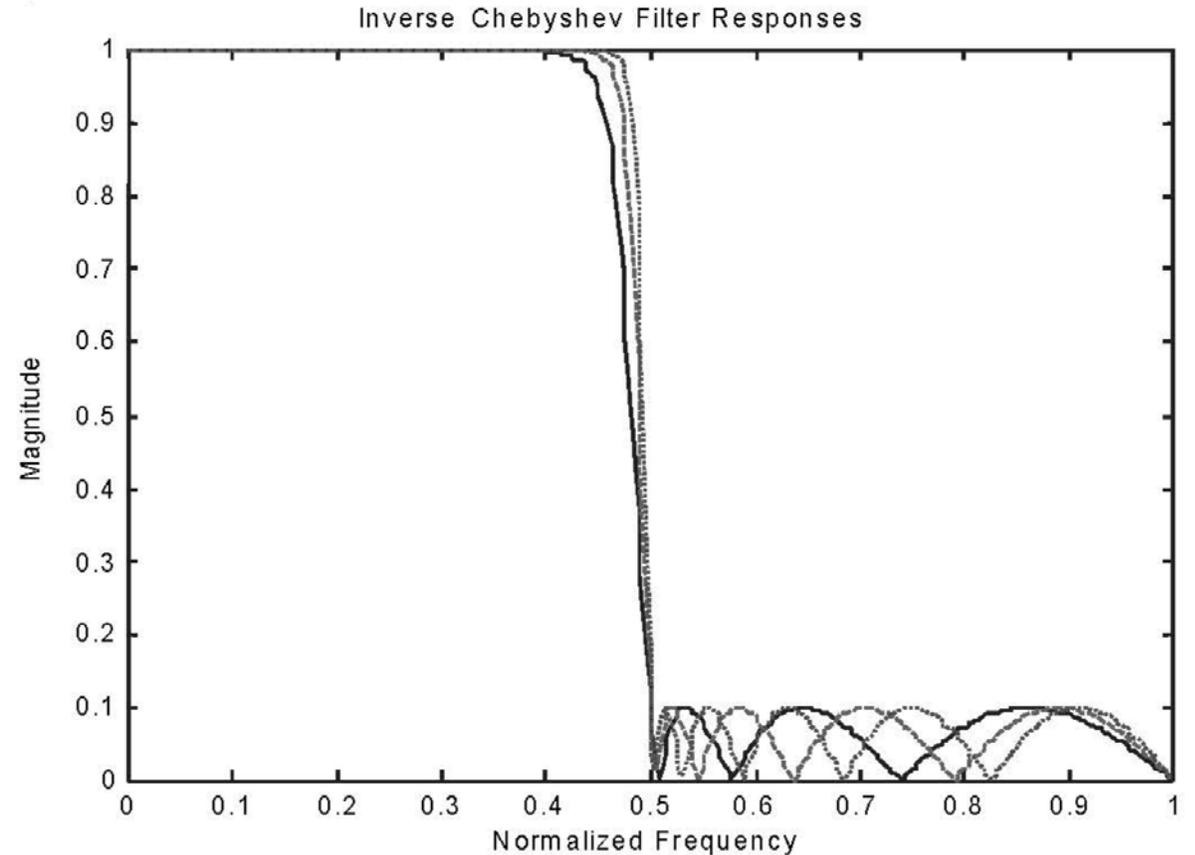
Aproximação inversa de Chebyshev

- Os procedimentos de projeto são semelhantes aos filtros Chebyshev, exceto que as fórmulas são ligeiramente diferentes.
- As fórmulas de projeto são expressas abaixo:

$$K_s = 10 \log \left(1 + \frac{1}{\epsilon^2} \right)$$

$$\epsilon = \frac{1}{10^{0.1K_s} - 1}$$

$$N = \frac{\cosh^{-1} \sqrt{(10^{0.1K_s} - 1) / (10^{0.1K_p} - 1)}}{\cosh^{-1} (1/\omega_p)}$$



Inverse Chebyshev filter responses

Aproximação inversa de Chebyshev - Exemplo

Determine a ordem de um filtro Chebyshev inverso que satisfaça às seguintes especificações:

$$\omega_p = 0.5 \text{ rad/s}$$

$$\omega_s = 1 \text{ rad/s}$$

$$K_p = 0.5 \text{ dB}$$

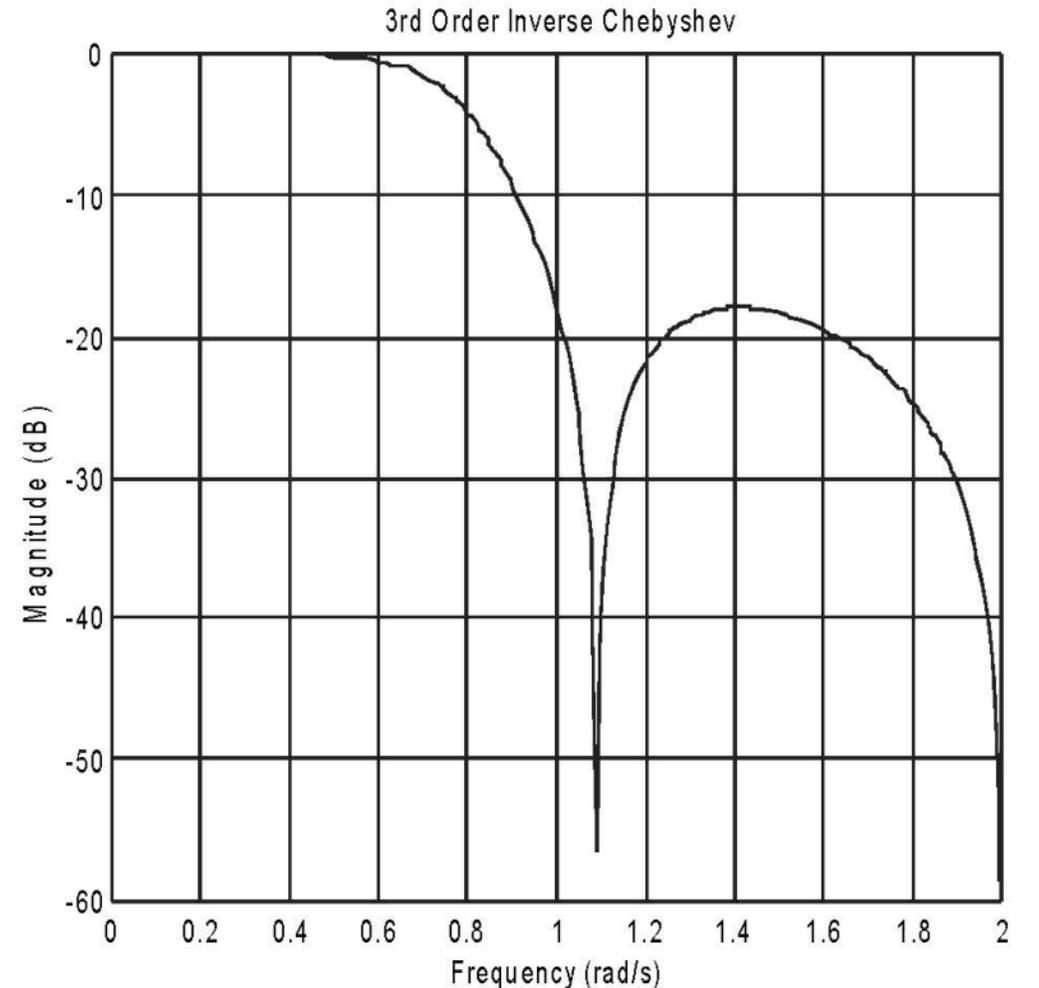
$$K_s = 18 \text{ dB}$$

Utilizando as equações do slide anterior:

$$N \geq \frac{\cosh^{-1} 22.5589}{\cosh^{-1} 2} = \frac{3.8088}{1.3170} = 2.8921$$

Um filtro de terceira ordem será suficiente.

A Figura ao lado apresenta a resposta do filtro.



Filtros FIR vs Filtros IIR

- Filtros FIR permitem obter curva de **fase linear** exata na sua resposta em frequência. A importância da fase linear já foi discutida no *slide* 17.
- Os filtros IIR geralmente têm resposta de fase não linear, especialmente próximo às bordas da faixa de passagem.
- Os filtros FIR são **estáveis**, enquanto não existe garantia de estabilidade para os filtros IIR.
- Os filtros IIR são mais adequados se for requerido **faixa de transição estreita** (corte abrupto). A ordem do filtro FIR necessária para um corte abrupto pode ser muito alta.
- Do ponto de vista da implementação, **filtros de ordem superior têm mais coeficientes** e, portanto, exigem mais armazenamento, resultando em maior custo computacional e atrasos maiores no domínio tempo.
- Deve ser enfatizado que os chips de DSP são otimizados para executar operações exigidas pela FFT e convoluções e, portanto, podem implementar a filtragem FIR de forma muito eficiente.

Filtros FIR vs Filtros IIR

- É relativamente **fácil projetar filtros FIR** com respostas de frequência arbitrárias.
- No entanto, os **filtros analógicos podem ser prontamente transformados em filtros digitais IIR** equivalentes (com especificações semelhantes). Esta é uma vantagem para os projetistas que precisam converter aplicativos analógicos existentes em digitais. Filtros FIR não têm contrapartida analógica.
- Os filtros IIR são mais suscetíveis a erros de arredondamento e a erros de quantização (**ruído de quantização**) do que os filtros FIR.
- Portanto, se precisarmos de um filtro **de corte abrupto com alta taxa de transferência** (que opera sob alta frequência de amostragem, transferindo através do filtro palavras binárias (ou amostras) sob alta taxa de transferência) **e baixo atraso**, o filtro IIR é mais adequado em razão do menor número de coeficientes. Por outro lado, se a fase linear exata é muito importante, devem ser utilizados filtros FIR.
- Os filtros FIR são a escolha mais comum se o número de coeficientes não for muito grande em função de suas melhores propriedades numéricas.

Forma Direta – Filtros IIR

- Conforme já analisamos, após a determinação dos coeficientes de um filtro digital, o problema de aproximação está solucionado.
- O próximo estágio do processo de projeto do filtro é chamado de realização ou implementação do filtro, que consiste na **definição da estrutura que relaciona a entrada e a saída do filtro, ilustrada por diagramas de blocos.**
- Os blocos que integram as estruturas dos filtros podem ser implementados em *hardware* digital, em um chip para DSP ou em VLSI.
- Algumas estruturas são mais sensíveis a erros nos coeficientes, devidos à quantização. Em alguns casos, como no caso dos filtros IIR, a estabilidade do filtro pode depender da estrutura escolhida.

Forma Direta – Filtros IIR

Consideremos um filtro IIR de segunda ordem, com função de transferência dada por

$$H(z) = \frac{B(z)}{A(z)} = \frac{b^0 + b_1z^{-1} + b_2z^{-2}}{1 + a_1z^{-1} + a_2z^{-2}}$$

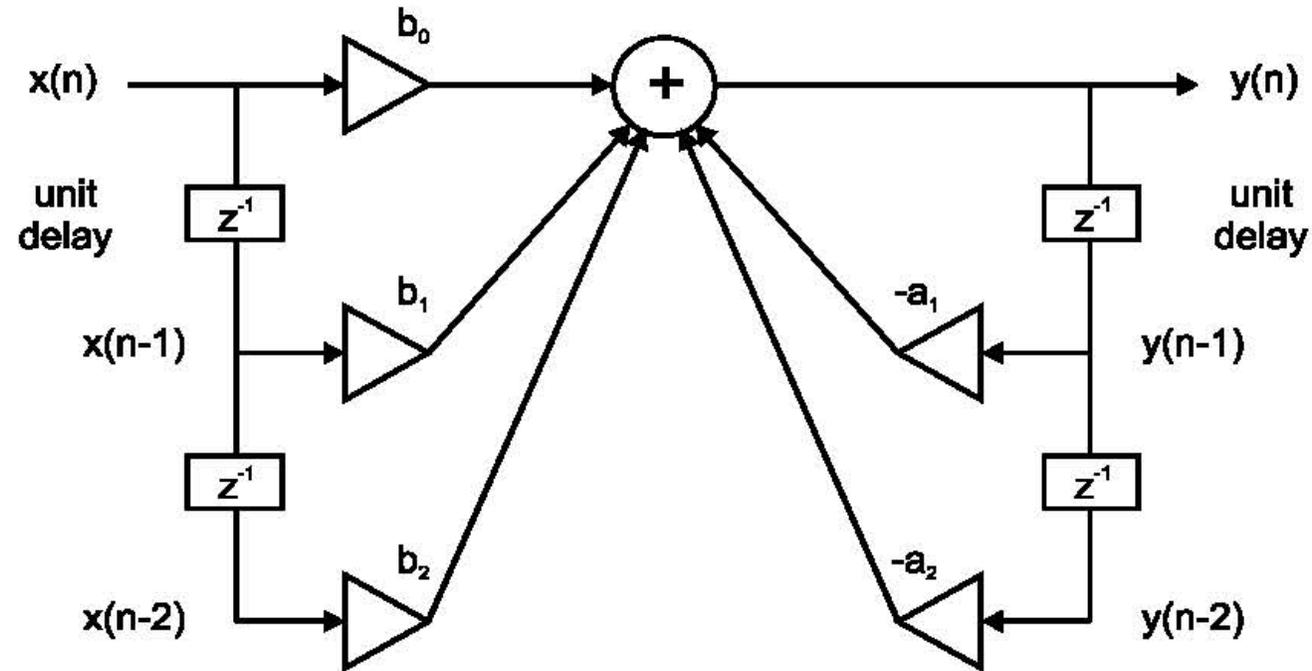
As amostras de entrada e saída são relacionadas por

$$y(n) = -a_1y(n-1) - a_2y(n-2) + b_0x(n) + b_1x(n-1) + b_2x(n-2)$$

A forma direta é baseada na implementação direta da equação de diferenças.

Forma Direta – Filtros IIR

A figura abaixo apresenta a forma direta obtida para o filtro a partir da equação de diferenças

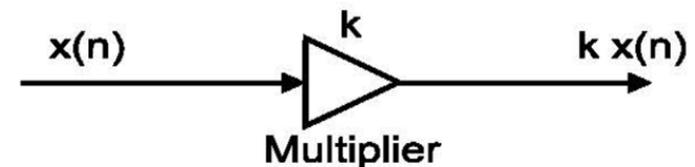
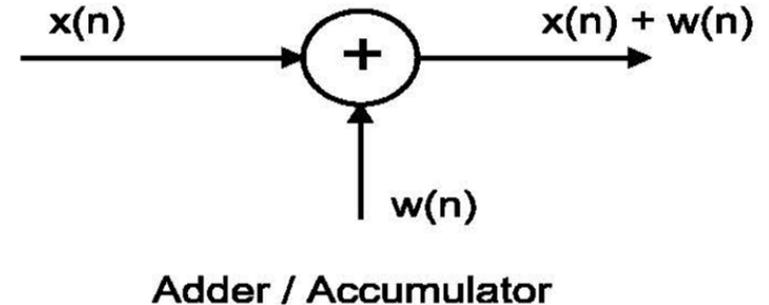
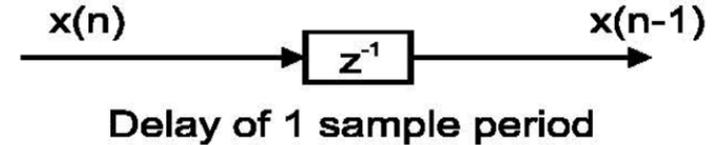


Forma direta para o filtro IIR descrito pela equação de diferenças

$$y(n) = -a_1y(n-1) - a_2y(n-2) + b_0x(n) + b_1x(n-1) + b_2x(n-2)$$

Forma Direta – Filtros IIR

- Os 3 elementos básicos da estrutura são ilustrados na figura ao lado.
- Na figura, z^{-1} representa o atraso de uma amostra.
- Na implementação prática, equivaleria a registradores de deslocamento ou a um local de memória na RAM.



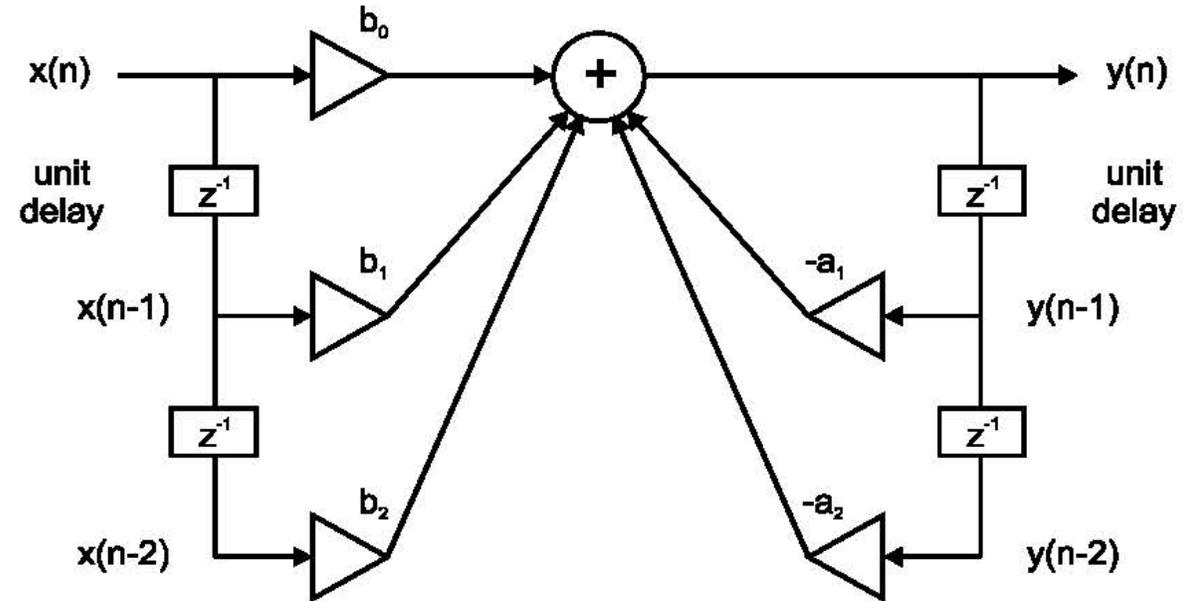
Forma Direta – Filtros IIR

A figura ao lado também ilustra as duas partes da estrutura do filtro.

$$H(z) = \frac{B(z)}{A(z)} = \frac{b^0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

Os termos do numerador, mostrados no lado esquerdo do bloco somador, são os elementos *feed-forward*.

Os termos do denominador, que dependem das amostras de saída anteriores, estão sendo realimentados (termos de *feedback*), e são mostrados à direita do bloco somador.



Forma direta para o filtro IIR descrito pela equação de diferenças

$$y(n) = -a_1 y(n-1) - a_2 y(n-2) + b_0 x(n) + b_1 x(n-1) + b_2 x(n-2)$$

Forma Direta – Filtros IIR

- A forma direta é chamada forma direta I, ou simplesmente forma direta.
- Há também uma estrutura denominada forma direta II, igualmente conhecida como forma canônica.
- Para derivar a estrutura da forma direta II, consideremos a equação de diferença do filtro de segunda ordem em análise.
- Os termos do lado direito da equação são, então, reagrupados conforme

$$y(n) = [b_0x(n) + b_1x(n-1) + b_2x(n-2)] + [-a_1y(n-1) - a_2y(n-2)]$$

- Esta estrutura pode ser vista como uma cascata de dois filtros: um filtro apenas com termos *feed-forward* e um filtro apenas com termos de *feedback*.

Forma Direta – Filtros IIR

As funções de transferência destes dois filtros são

$$H_1(z) = B(z)$$

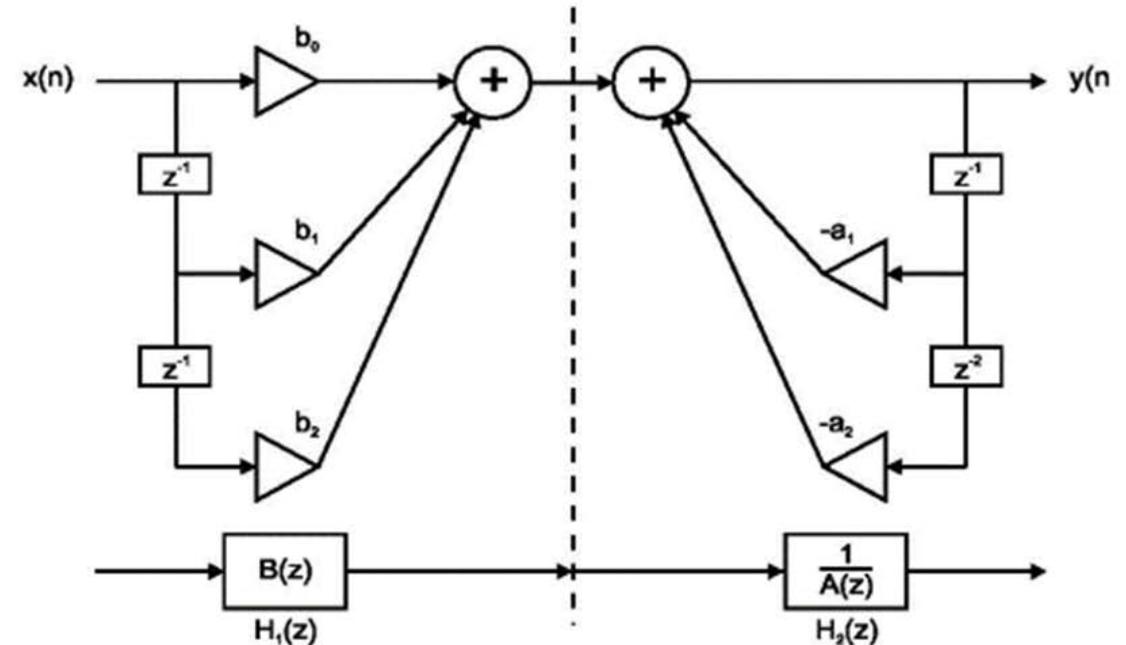
$$H_2(z) = \frac{1}{A(z)}$$

de modo que sua cascata é

$$H_1(z) \cdot H_2(z) = B(z) \cdot \frac{1}{A(z)} = H(z)$$

que é a função de transferência original.

Esse reagrupamento está representado na figura ao lado.

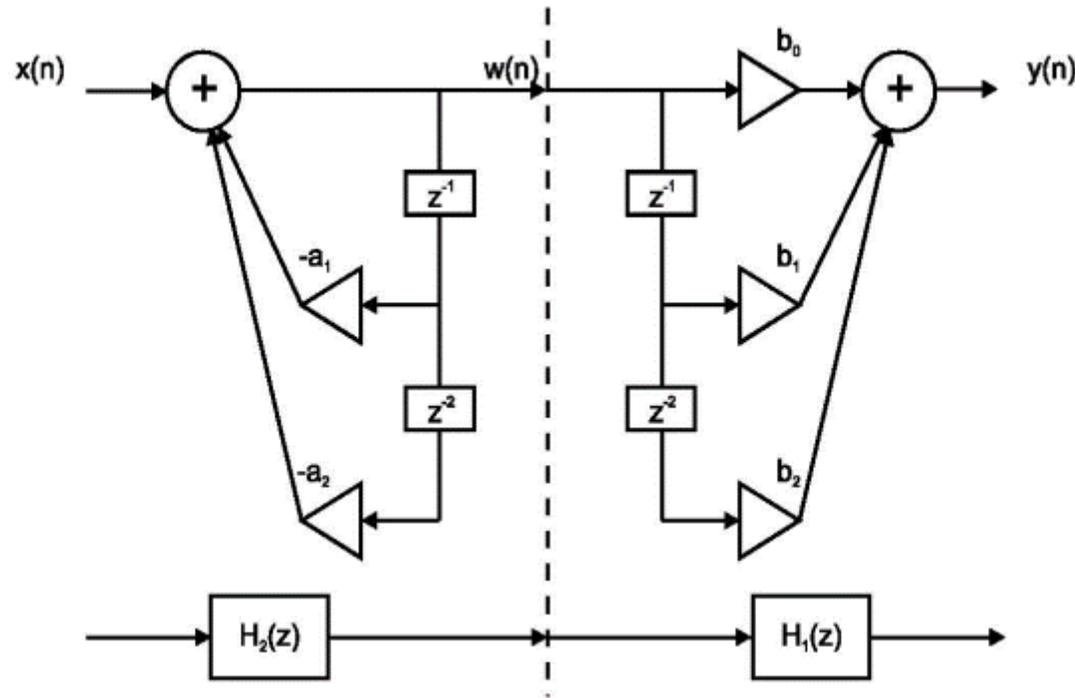


Forma direta – estrutura alternativa

Forma Direta – Filtros IIR

A ordem dos dois filtros pode ser trocada sem afetar a função de transferência do filtro digital, uma vez que ambos os filtros representam sistemas lineares.

A Figura abaixo apresenta a cascata entre os dois filtros com a ordem intercambiada.

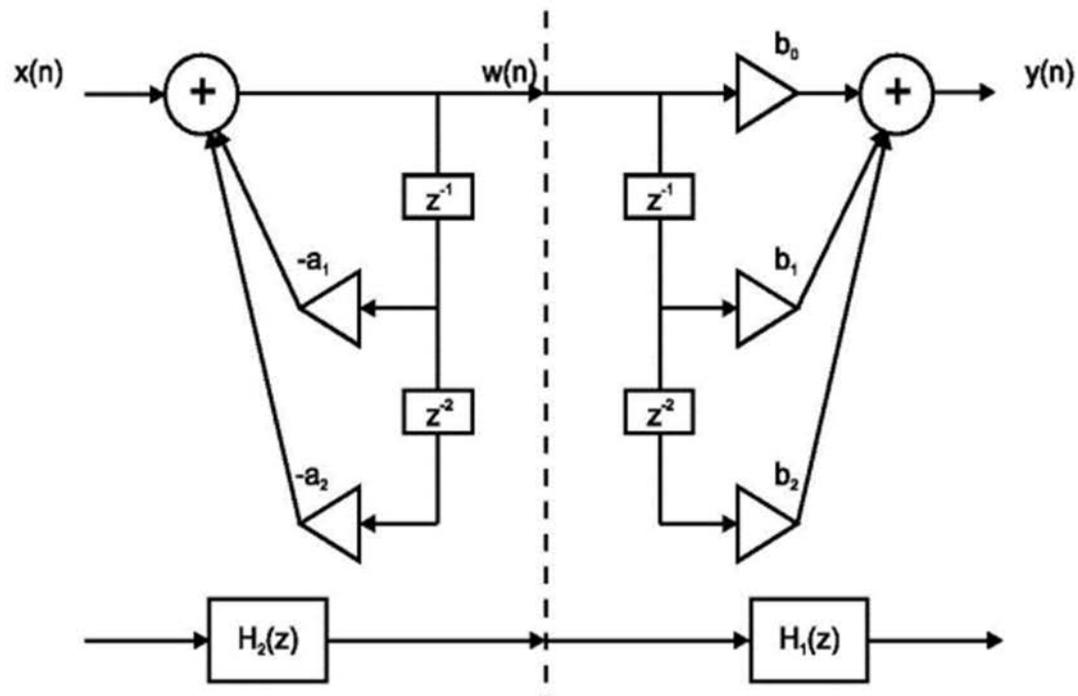


Cascata - ordem intercambiada

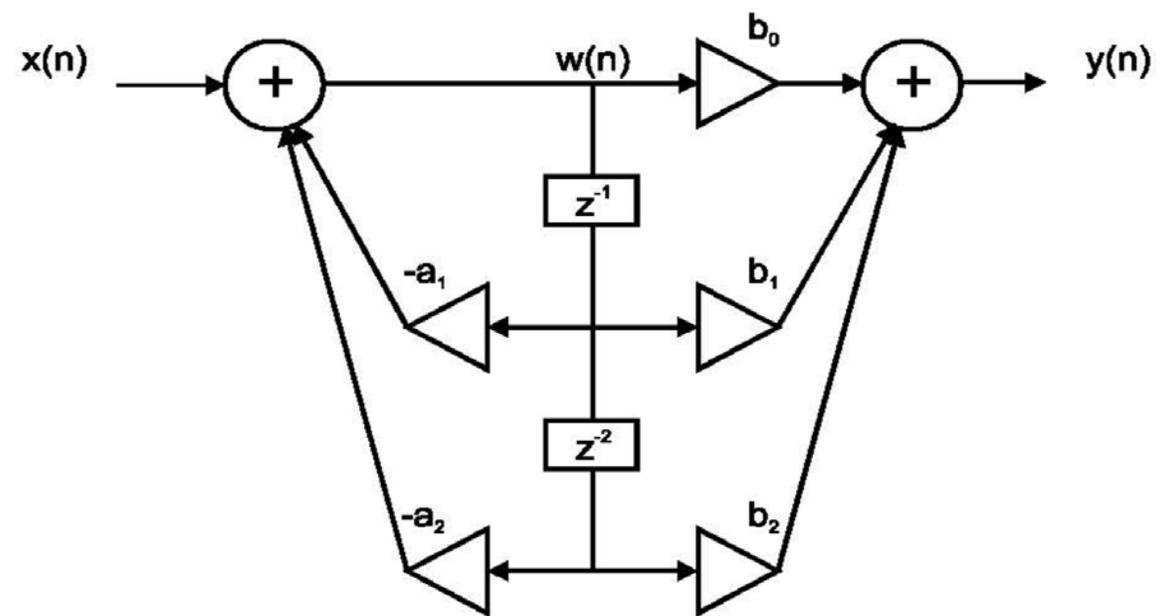
Forma Direta – Filtros IIR

Na figura abaixo, à esquerda, a saída do filtro $H_2(z)$ é agora a entrada para o filtro $H_1(z)$. Note que a saída de $H_2(z)$, denotada por $w(n)$, está sendo atrasada da mesma maneira pelos dois filtros.

Portanto, não são necessários dois conjuntos separados de atrasos, e a nova forma pode ser conforme a figura à direita. Essa forma é denominada **forma canônica**.



Cascata - ordem intercambiada



Forma Canônica

Forma Direta – Filtros IIR

A forma canônica implementa a equação de diferenças do filtro IIR.

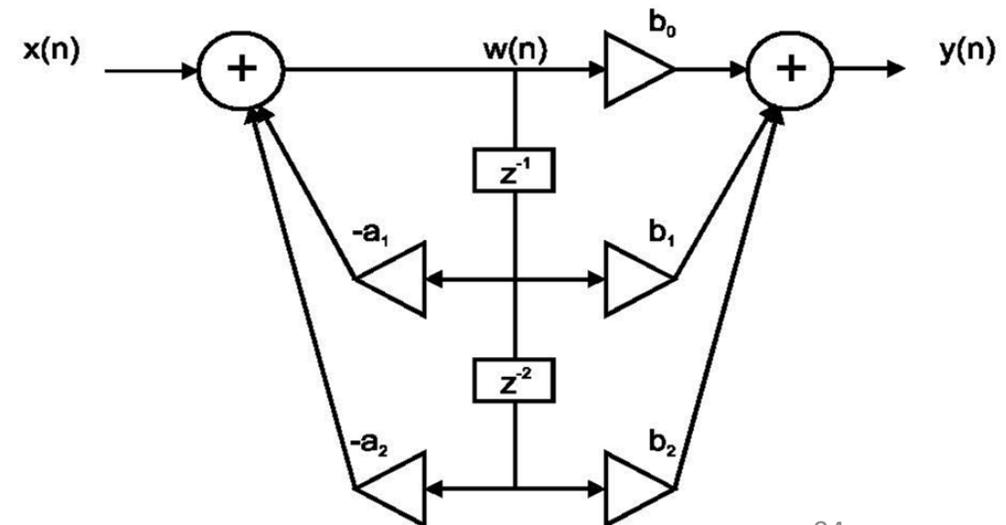
No primeiro somador, à esquerda, temos

$$w(n) = -\sum_{k=1}^2 a_k w(n-k) + x(n) = -a_1 w(n-1) - a_2 w(n-2) + x(n)$$

A saída do segundo somador é

$$y(n) = \sum_{k=0}^2 b_k w(n-k) = b_0 w(n) + b_1 w(n-1) + b_2 w(n-2)$$

Substituindo as expressões para $w(n)$, $w(n-1)$, e $w(n-2)$ na equação acima, obtemos

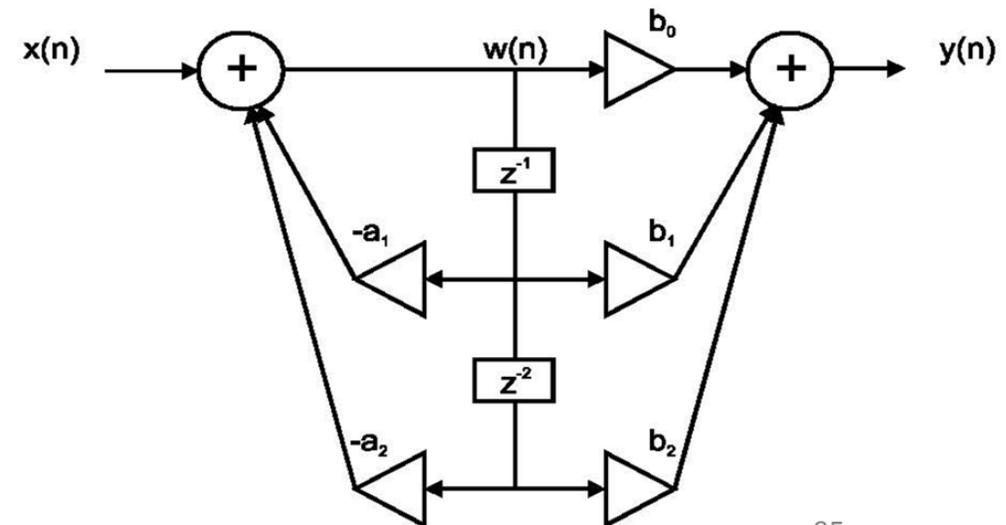


Forma Direta – Filtros IIR

$$\begin{aligned}y(n) &= b_0 [-a_1 w(n-1) - a_2 w(n-2)] + b_0 x(n) + b_1 [-a_1 w(n-2) - a_2 w(n-3)] + b_1 x(n-1) + \\ &\quad + b_2 [-a_1 w(n-3) - a_2 w(n-4)] + b_2 x(n-2) \\ &= \sum_{k=0}^2 b_k x(n-k) - a_1 [b_0 w(n-1) + b_1 w(n-2) + b_2 w(n-3)] - a_2 [b_0 w(n-2) + b_1 w(n-3) + b_2 w(n-4)] \\ &= \sum_{k=0}^2 b_k x(n-k) - a_1 y(n-1) - a_2 y(n-2)\end{aligned}$$

que é a equação de diferença original.

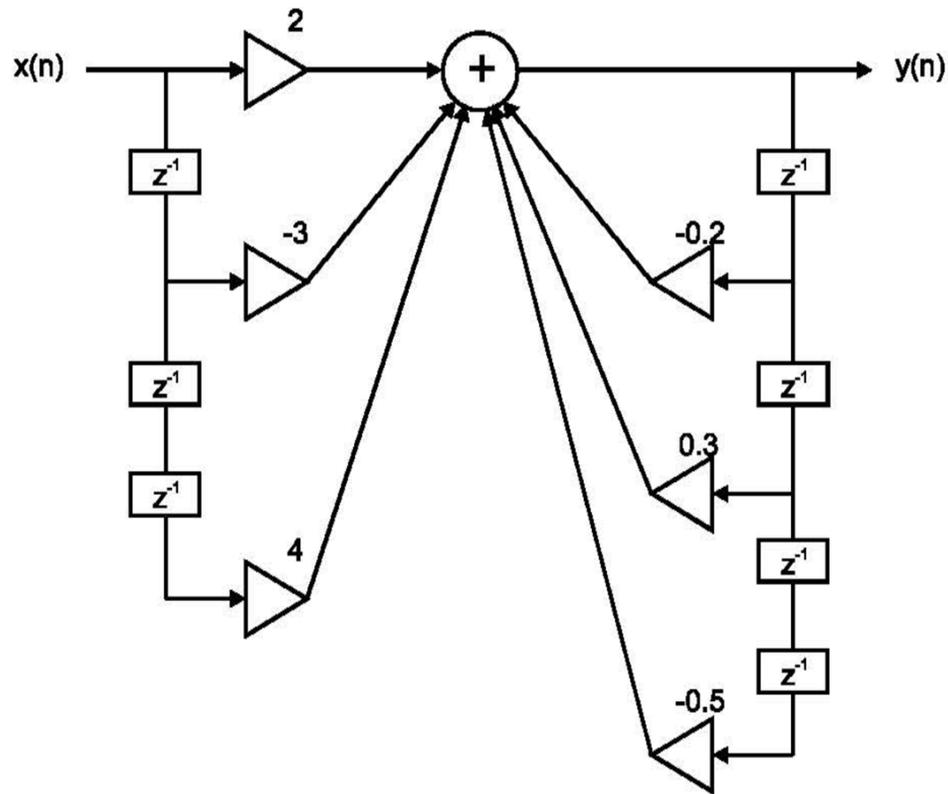
Embora tenhamos usado o filtro IIR de segunda ordem como exemplo, as estruturas podem facilmente ser generalizadas para funções de transferência de ordens superiores.



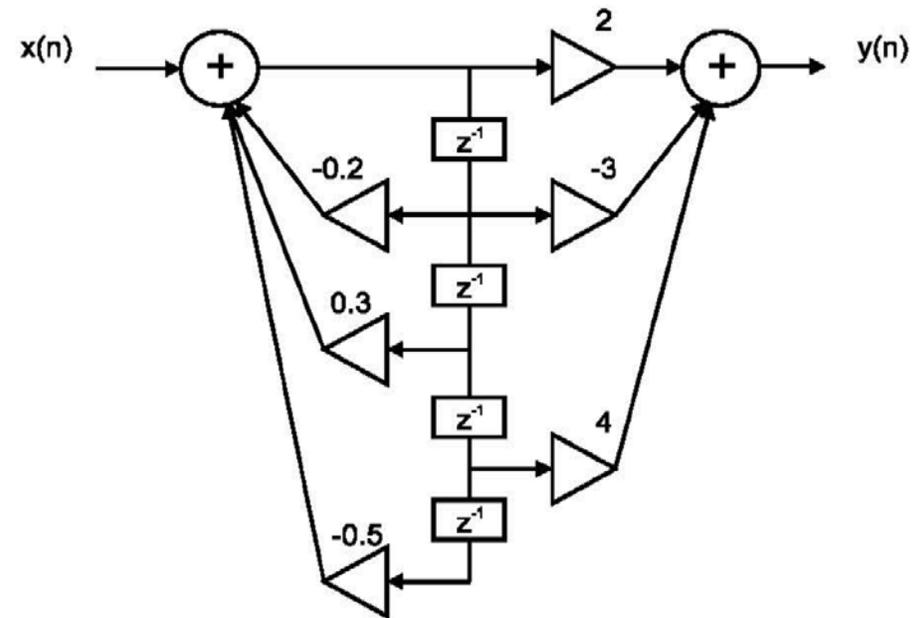
Forma Direta – Filtros IIR – Exemplo

Para a função de transferência abaixo,
$$H(z) = \frac{2 - 3z^{-1} + 4z^{-3}}{1 + 0.2z^{-1} - 0.3z^{-2} + 0.5z^{-4}}$$

desenhe as implementações das formas diretas I e II para o filtro.



Forma direta I



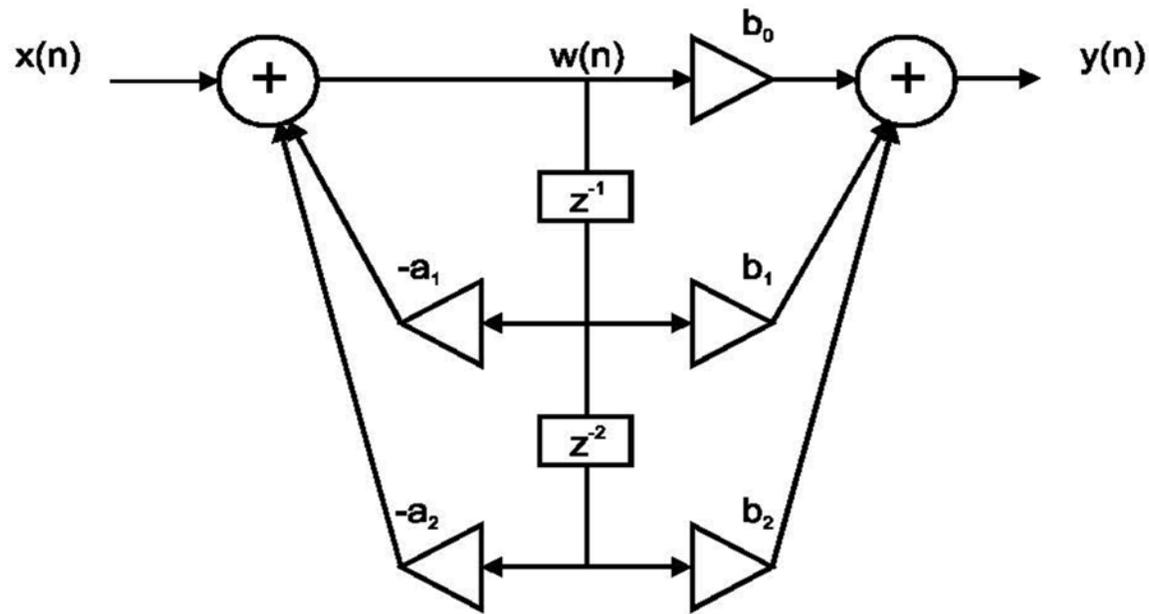
Forma direta II

Estruturas Transpostas – Filtros IIR

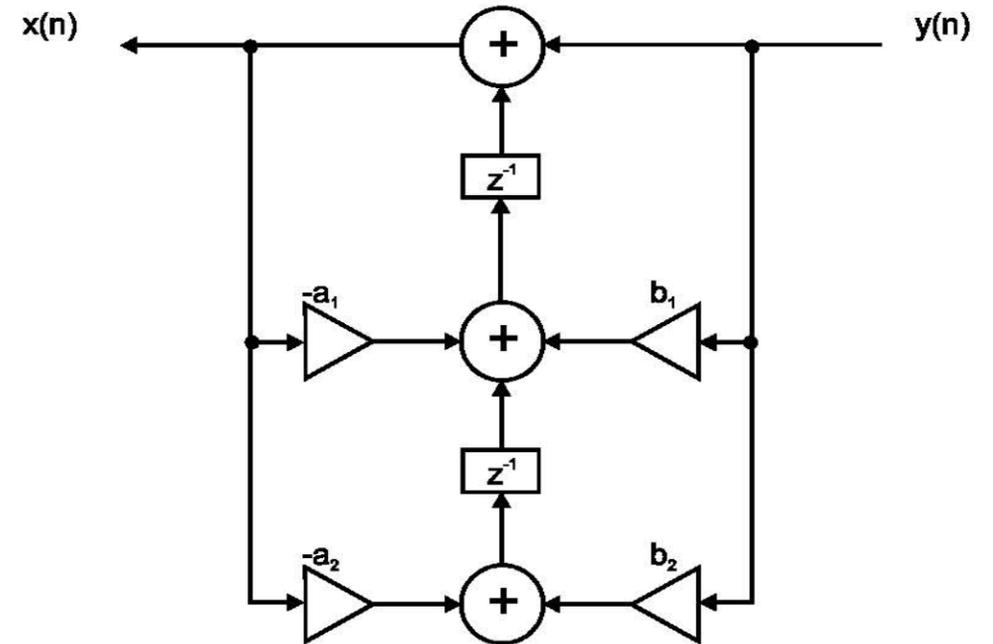
Uma estrutura alternativa baseada na forma canônica pode ser obtida por transposição.

A estrutura é obtida invertendo todas as direções do fluxo de sinal, e substituindo nós (pontos de conexão) por somadores e vice-versa.

O resultado é apresentado abaixo.



Forma Canônica



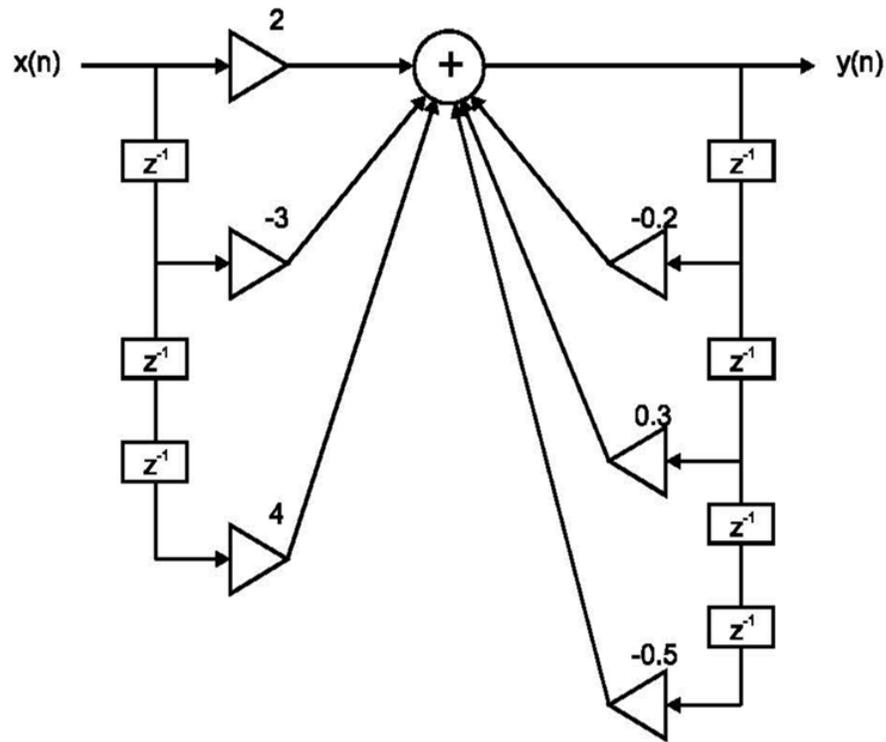
Forma Canônica Transposta

Estruturas Transpostas – Filtros IIR

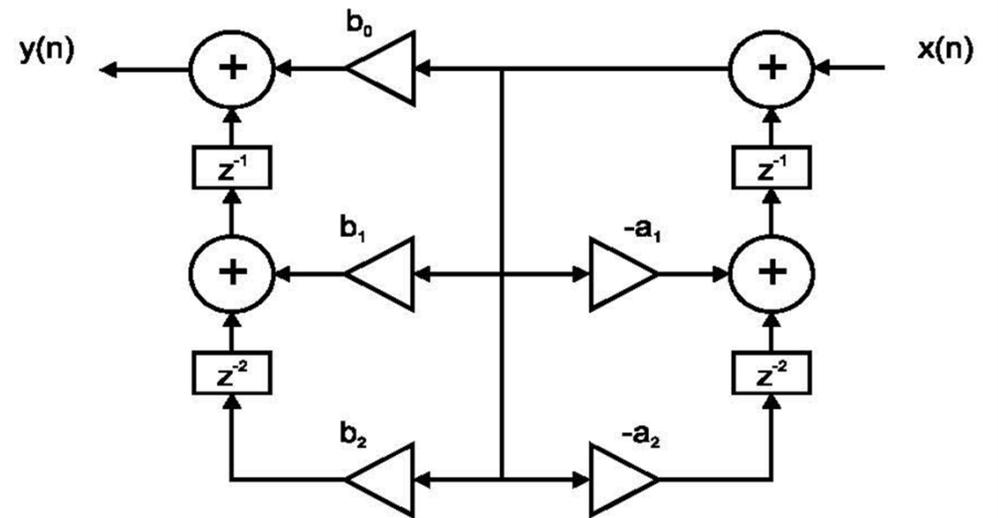
Uma estrutura de forma direta transposta I pode ser obtida da mesma maneira.

O resultado é mostrado na figura abaixo.

Em ambos os casos, a função de transferência permanece a mesma após a transposição.



Forma direta I



Forma direta I transposta

Forma Direta – Filtros FIR

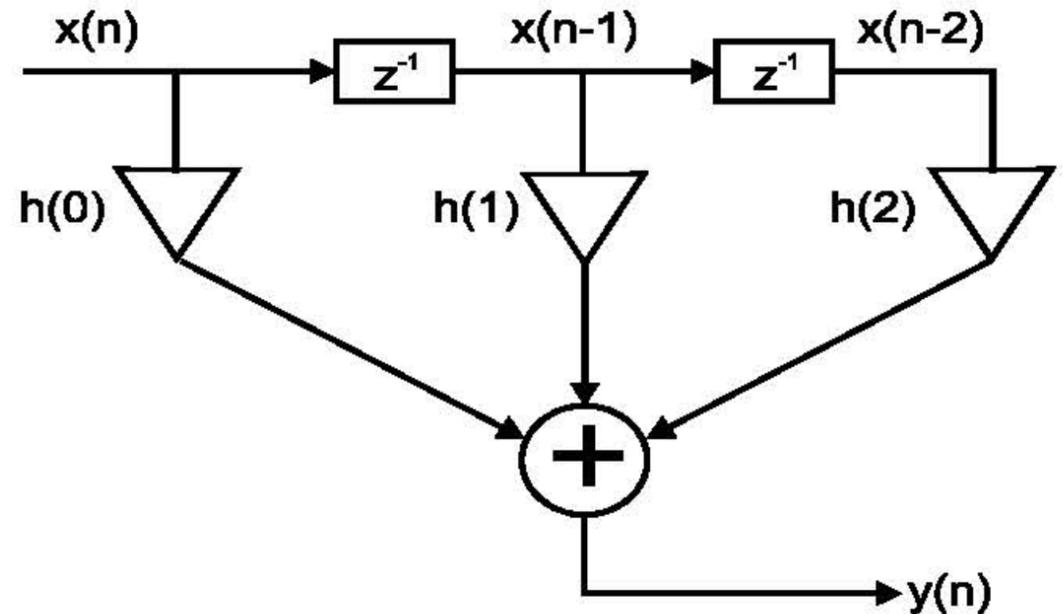
Para um filtro FIR, o polinômio denominador é simplesmente igual a 1, ou seja, $A(z) = 1$.

Portanto, existem unicamente os elementos *feedforward*, conforme mostra a figura abaixo.

Esta implementação corresponde à equação para um filtro FIR de segunda ordem, ou seja,

$$y(n) = \sum_{m=0}^2 h(m)x(n-m)$$

onde $h(m)$ é a resposta ao impulso, ou os coeficientes do filtro.



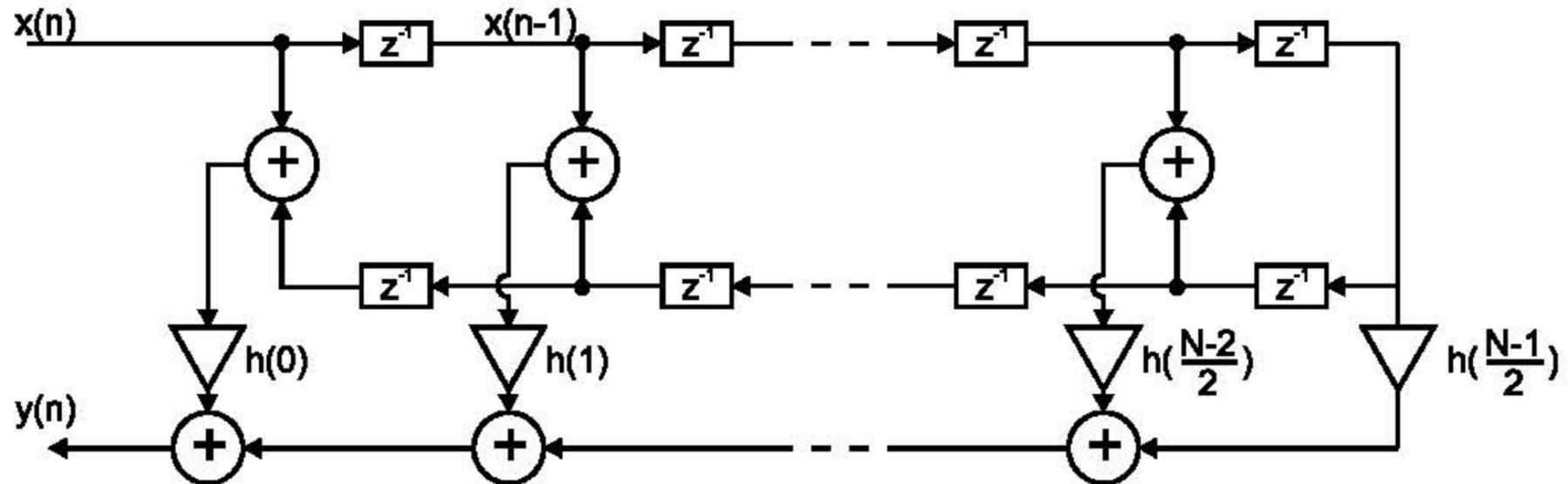
Filtro FIR - estrutura

Forma Direta – Filtros FIR

Outra forma para a estrutura direta de um filtro FIR é a [estrutura transversal](#) ou a [estrutura de linha de atraso de derivação](#).

Para filtros FIR de fase linear, os coeficientes do filtro são simétricos ou anti-simétricos, assim, para um filtro de ordem N , o número de multiplicações pode ser reduzido de N para $N/2$ para N par, e reduzido para $(N + 1)/2$ para N ímpar.

A figura a seguir a forma direta de realização de um filtro FIR de fase linear, cuja implementação considera esta redução.

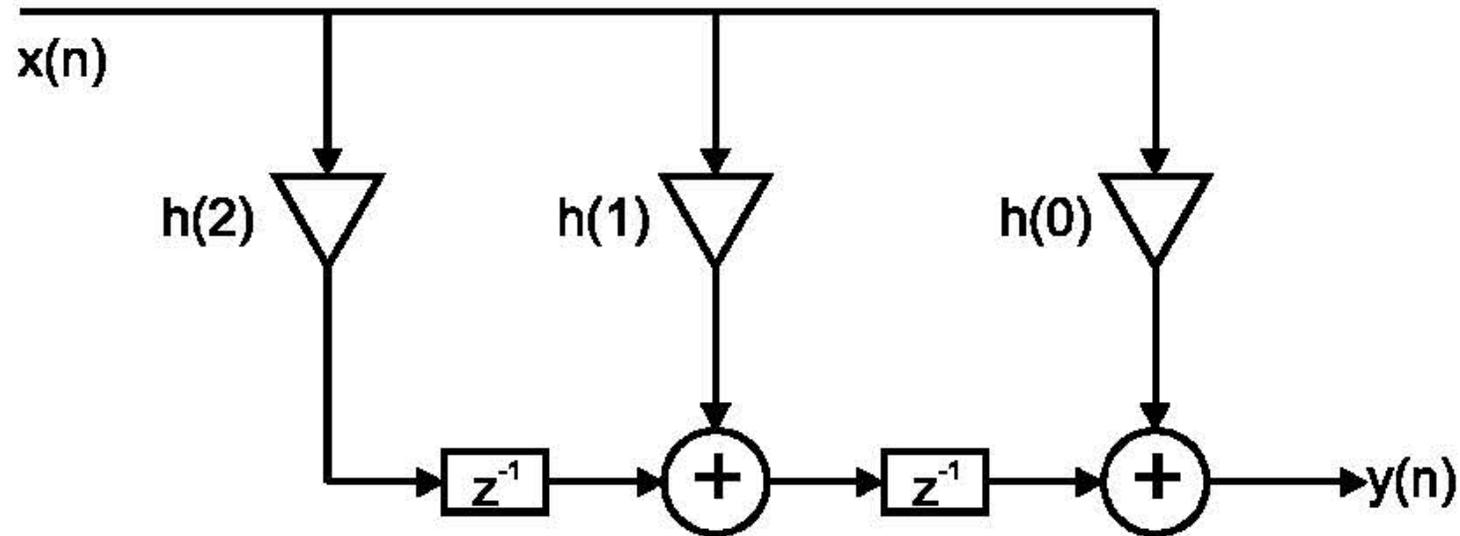


Implementação em forma direta de um filtro FIR de ordem ímpar e fase linear

Estrutura Direta Transposta – Filtros FIR

Uma **estrutura direta transposta** também pode ser obtida para o filtro FIR através do método discutido para a transposição de estruturas IIR.

A estrutura resultante para o filtro é mostrada na figura abaixo.



Implementação em forma direta transposta de um filtro FIR