



PROCESSAMENTO DIGITAL DE SINAIS

AULA 5



Profª Viviana Raquel Zurro



CONVERSA INICIAL

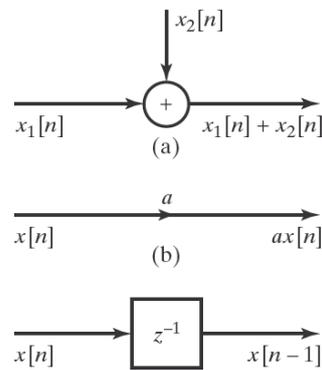
Caro aluno, nesta aula desenvolveremos estudos sobre estruturas para sistemas em tempo discreto. Essas estruturas são as bases para o desenvolvimento de filtros digitais, devido ao fato de que é necessário que os sistemas em tempo discreto sejam eficientes, com uma precisão numérica adequada. Para implementar o sistema, podemos usar a soma de convolução ou a equação de diferença de um sistema LIT. Atualmente, a implementação de um filtro digital LIT pode ser feita por software ou por hardware, dependendo da aplicação. Os coeficientes dos filtros e as variáveis do sinal não podem ser representados com precisão, devido principalmente à digitalização, e uma implementação direta de um filtro digital com base na convolução finita ou na equação de diferenças pode não ter comportamento satisfatório, devido a imprecisões aritméticas.

O primeiro passo para a implementação em hardware ou software de um filtro digital é interconectar blocos de construção básicos para formar a estrutura do sistema. Esta representação estrutural facilita a visualização da relação entre variáveis internas importantes com a entrada e a saída do sistema. Isso facilita a implementação do filtro digital (Mitra, 2010).

TEMA 1 – EQUAÇÕES DE DIFERENÇA LINEARES COM COEFICIENTES CONSTANTES

A relação entrara/saída de um sistema linear invariante no tempo discreto (LIT) é caracterizado pela função do sistema, pela equação de diferenças e pela resposta ao impulso. Para implementar um sistema LIT, são necessários os seguintes elementos: memória para armazenar dados das sequências, somadores e multiplicadores. A Figura 1 mostra os símbolos dos operadores necessários.

Figura 1 – Operadores: (a) soma de sequências, (b) multiplicação por uma constante, (c) atraso unitário



Fonte: Oppenheim; Schafer, 2012.

A função do sistema mostrada na equação (1), cuja resposta ao impulso é mostrada na equação (2), implica numa equação de diferenças de primeira ordem, mostrada na equação (3), a qual satisfaz as sequências de entrada e saída.

$$H(z) = \frac{b_0 + b_1 z^{-1}}{1 - az^{-1}} \quad (1)$$

$$h_{(n)} = b_0 a^n u(n) + b_1 a^{n-1} u(n-1) \quad (2)$$

$$y(n) - ay(n-1) = b_0 x(n) + b_1 x(n-1) \quad (3)$$

Ela pode ser escrita como:

$$y(n) = ay(n-1) + b_0 x(n) + b_1 x(n-1) \quad (4)$$

A seguir, mostraremos um exemplo de representação de uma equação de diferenças em diagrama de blocos. Eis a equação de diferenças:

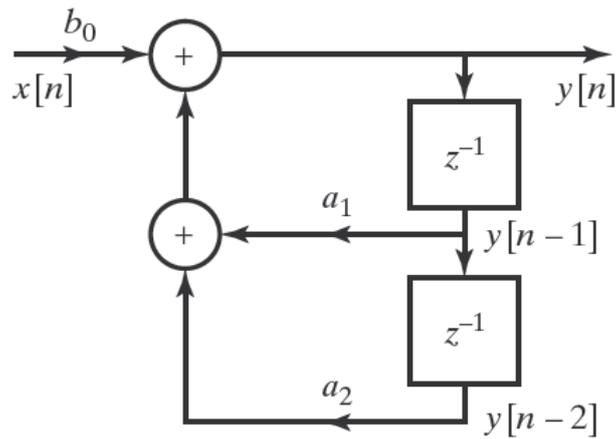
$$y(n) = a_1 y(n-1) + a_2 y(n-2) + bx(n) \quad (5)$$

A função do sistema é:

$$H(z) = \frac{b}{1 - a_1 z^{-1} - a_2 z^{-2}} \quad (6)$$

Ela será representada pelo diagrama de blocos da Figura 2.

Figura 2 – Diagrama de blocos da equação de diferenças mostrada na equação (5)



Fonte: Oppenheim; Schafer, 2012.

Equações de diferença de ordem maior podem ser representadas da seguinte maneira:

$$y(n) - \sum_{k=1}^N a_k y(n-k) = \sum_{k=0}^M b_k x(n-k) \quad (7)$$

A função do sistema é:

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}} \quad (8)$$

Reescrevendo a equação (7), temos:

$$y(n) = \sum_{k=1}^N a_k y(n-k) + \sum_{k=0}^M b_k x(n-k) \quad (9)$$

Definido $v(n)$:

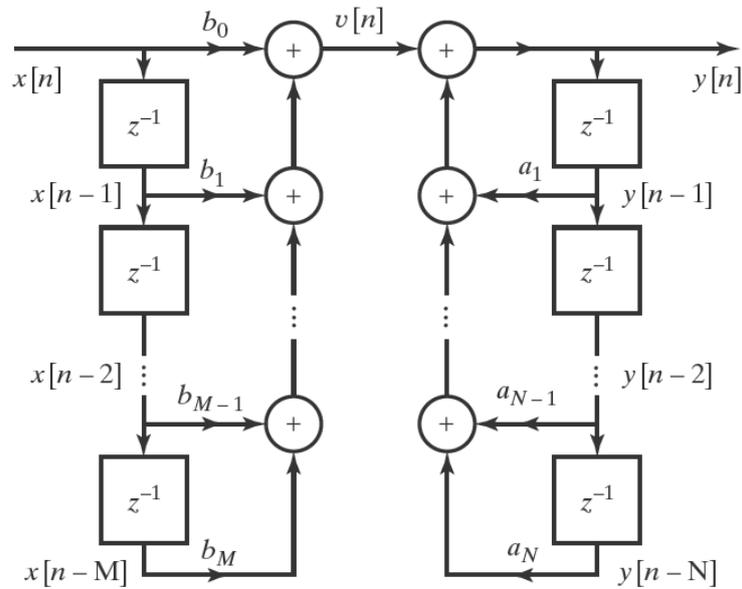
$$v(n) = \sum_{k=0}^M b_k x(n-k) \quad (10)$$

Substituindo na equação (9):

$$y(n) = \sum_{k=1}^N a_k y(n-k) + v(n) \quad (11)$$

Essa equação pode ser representada pelo diagrama de blocos mostrado na Figura 3.

Figura 3 – Diagrama de blocos de uma equação de diferenças de ordem N



Fonte: Oppenheim; Schafer, 2012.

Se definirmos $H(z)$ como:

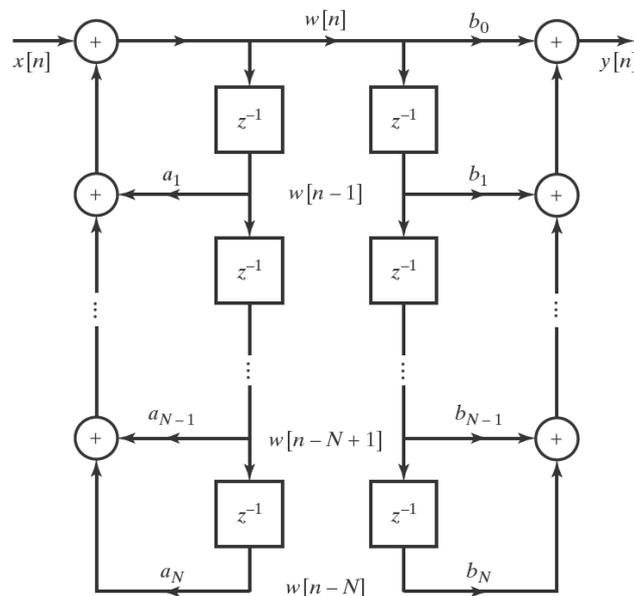
$$H(z) = H_1(z)H_2(z) = \left(\sum_{k=0}^M b_k z^{-k} \right) \left(\frac{1}{1 - \sum_{k=1}^N a_k z^{-k}} \right) \quad (12)$$

$$W(z) = H_2(z)X(z) = \left(\frac{1}{1 - \sum_{k=1}^N a_k z^{-k}} \right) X(z) \quad (13)$$

$$Y(z) = H_1(z)W(z) = \left(\sum_{k=0}^M b_k z^{-k} \right) W(z) \quad (14)$$

Assumindo $N = M$, a equação está representada na Figura 4.

Figura 4 – Representação da equação (14) em diagrama de blocos



Fonte: Oppenheim; Schafer, 2012.



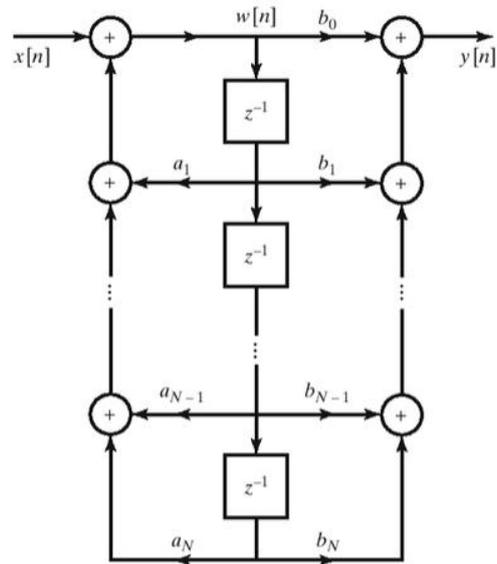
No domínio do tempo:

$$w(n) = \sum_{k=1}^N a_k w(n-k) + x(n) \quad (15)$$

$$y(n) = \sum_{k=0}^M b_k w(n-k) \quad (16)$$

O diagrama de blocos da equação (16) é mostrado na Figura 5.

Figura 5 – Representação da equação (16) em diagrama de blocos

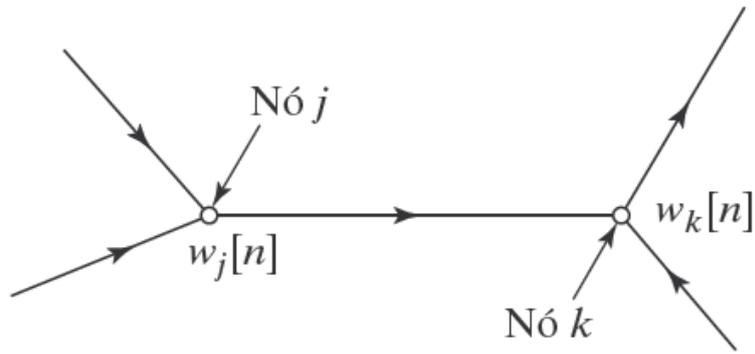


Fonte: Oppenheim; Schafer, 2012.

1.1 Diagramas de fluxo de sinais de equações de diferença lineares com coeficientes constantes

À exceção de algumas notações, o diagrama de blocos e o diagrama de fluxo de sinais de uma equação de diferenças são equivalentes. Um diagrama de fluxo está representado na Figura 6, onde cada nó tem uma variável associada. A conexão entre os nós é dada por ramos orientados.

Figura 6 – Exemplo de diagrama de fluxo de sinais



Fonte: Oppenheim; Schafer, 2012.

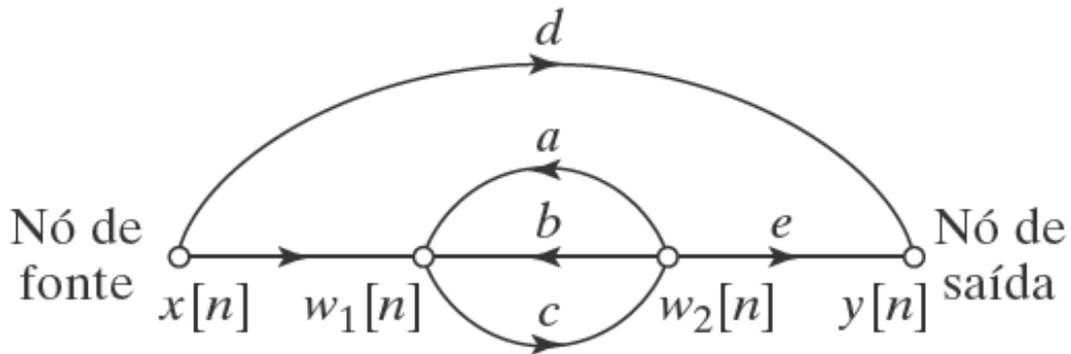
Por exemplo, o diagrama de fluxo da Figura 7 representa a equação (19), resultado das equações (17) e (18).

$$w_1(n) = x(n) + aw_2(n) + bw_2(n) \quad (17)$$

$$w_2(n) = cw_1(n) \quad (18)$$

$$y(n) = dx(n) + ew_2(n) \quad (19)$$

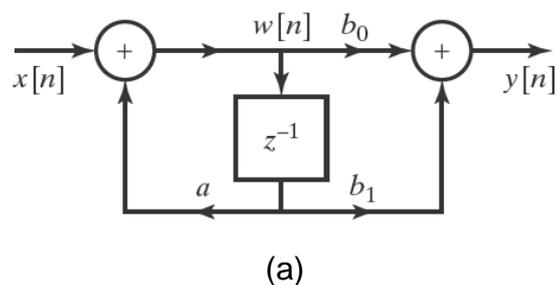
Figura 7 – Diagrama de fluxo com os nós de fonte e de saída

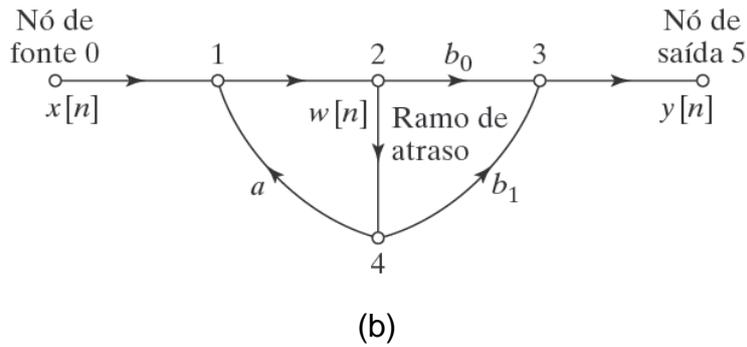


Fonte: Oppenheim; Schafer, 2012.

O seguinte exemplo mostra o processo de um filtro digital de primeira ordem: (a) diagrama de blocos de um filtro digital de ordem 1; (b) diagrama de fluxo de sinais correspondente ao diagrama de blocos do ponto (a).

Figura 8 – Diagrama filtro digital de primeira ordem





Fonte: Oppenheim; Schafer, 2012.

O diagrama de fluxo da Figura 9 corresponde às seguintes equações:

$$w_1(n) = aw_4(n) + x(n) \quad (20)$$

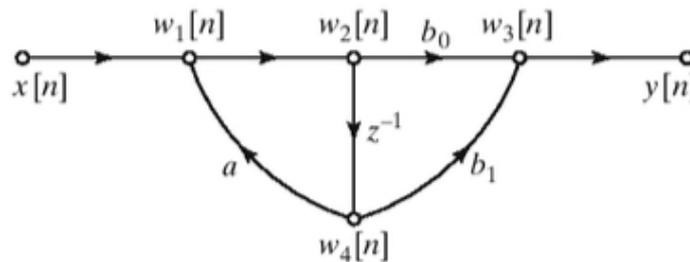
$$w_2(n) = w_1(n) \quad (21)$$

$$w_3(n) = b_0w_2(n) + b_1w_4(n) \quad (22)$$

$$w_4(n) = w_2(n - 1) \quad (23)$$

$$y(n) = w_3(n) \quad (24)$$

Figura 9 – Diagrama de fluxo de sinais correspondente às equações (20) a (24)



Fonte: Oppenheim; Schafer, 2012.

TEMA 2 – SISTEMAS IIR

A sigla IIR significa *resposta ao impulso infinita* (do inglês Infinite Impulse Response). A IIR é uma propriedade que se aplica a muitos sistemas lineares invariantes no tempo, como sistemas eletrônicos e filtros digitais. Esses filtros são conhecidos como filtros IIR e se diferenciam por ter uma resposta ao impulso que não chega a ser zero depois de certo ponto, e continua indefinidamente.

No processamento digital de sinais, é desejável que o sistema tenha a menor quantidade possível de operações, devido ao fato de que, quanto mais operações, maior será o tempo de processamento e o espaço de memória usado para armazenamento e processamento de dados. No caso de dispositivos e



hardware de processamento (dispositivos VLSI), a área do dispositivo é uma medida de eficiência importante.

2.1 Formas

Nesta aula, iremos estudar algumas das formas mais usadas para implementar sistemas IIR lineares invariantes no tempo, a começar pela forma **direta**. A equação característica, nesse caso, é a apresentada na equação (25):

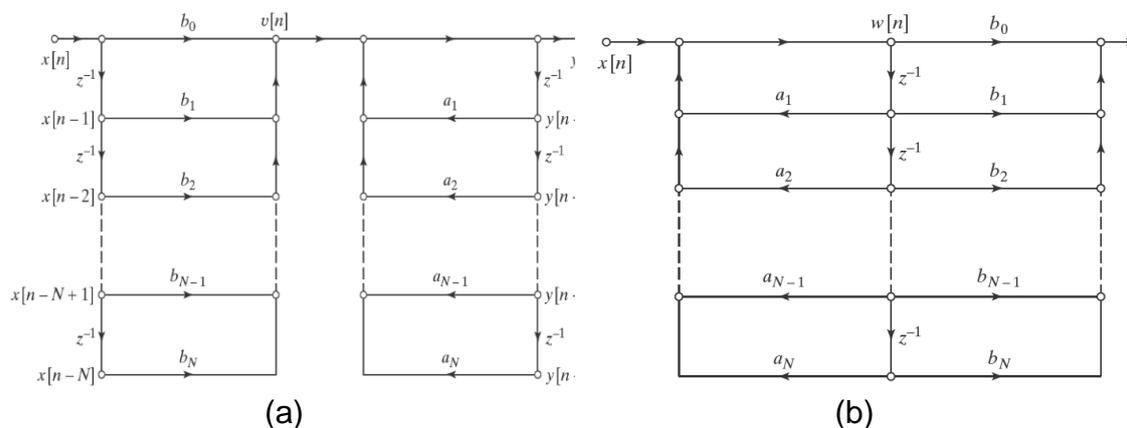
$$y(n) - \sum_{k=1}^N a_k y(n-k) = \sum_{k=0}^M b_k x(n-k) \quad (25)$$

Eis a função racional do sistema:

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}} \quad (26)$$

A Figura 10 mostra o diagrama de fluxo para a forma direta I e II.

Figura 10 – (a) Forma direta I do diagrama de fluxo de sinais; (b) forma direta II do diagrama de fluxo de sinais



Fonte: Oppenheim; Schafer, 2012.

A forma direta I usa a convenção do diagrama de fluxo de sinais. Considerando por conveniência $N = M$, teremos a forma direta II. É possível um nó do diagrama de fluxo ter várias entradas; no entanto, por convenção o sistema será montado com a noção de que cada nó tem somente duas entradas, o que facilita o desenvolvimento de programas e arquiteturas que representam a equação de diferenças.

A segunda forma é em cascata. A equação característica da estrutura em cascata é obtida pela fatoração dos polinômios do numerador e denominador da equação (26):



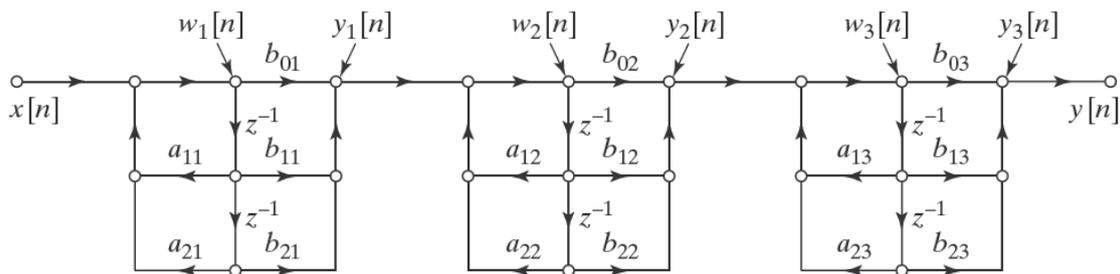
$$H(z) = \frac{\prod_{k=1}^{M_1} (1 - f_k z^{-1}) \prod_{k=1}^{M_2} (1 - g_k z^{-1}) (1 - g_k^* z^{-1})}{\prod_{k=1}^{N_1} (1 - c_k z^{-1}) \prod_{k=1}^{N_2} (1 - d_k z^{-1}) (1 - d_k^* z^{-1})} \quad (27)$$

$M = M_1 + 2M_2, N = N_1 + 2N_2$. Os fatores f_k representam os zeros reais e os c_k os polos reais. Os fatores g_k e g_k^* representam os zeros complexos conjugados e os fatores d_k e d_k^* representam os polos complexos conjugados. Para reduzir a quantidade de cálculos e o espaço de armazenamento, é conveniente usar uma estrutura combinada de pares de fatores reais e complexos:

$$H(z) = \prod_{k=1}^{N_s} \frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}} \quad (28)$$

Onde $N_s = \lfloor (N + 1) / 2 \rfloor$ é o inteiro de maior valor contido em $(N + 1) / 2$. Dessa forma, sendo $M \leq N$, os polos e zeros foram combinados em pares. A Figura 11 representa a estrutura de uma equação de diferença de sexta ordem dividida em três seções de segunda ordem na forma direta II ligadas em cascata.

Figura 11 – Subsistemas de segunda ordem na forma direta II ligados em cascata (sistema de sexta ordem)



Fonte: Oppenheim; Schafer, 2012.

A equação de diferença representada pela cascata tem a seguinte forma:

$$y_0(n) = x(n) \quad (29)$$

$$w_k(n) = a_{1k}w_k(n-1) + a_{2k}w_k(n-2) + y_{k-1}(n) \quad k = 1, \dots, N_s \quad (30)$$

$$y_k(n) = b_{0k}w_k(n) + b_{1k}w_k(n-1) + b_{2k}w_k(n-2) \quad k = 1, \dots, N_s \quad (31)$$

$$y(n) = y_{N_s}(n) \quad (32)$$



A terceira forma é a **paralela**. Aplicando expansão em frações parciais aos polinômios do numerador e denominador da função $H(z)$, a função do sistema racional é dada por:

$$H(z) = \frac{1 + 2z^{-1} + z^{-2}}{1 - 0,75z^{-1} + 1,125z^{-2}} \quad (33)$$

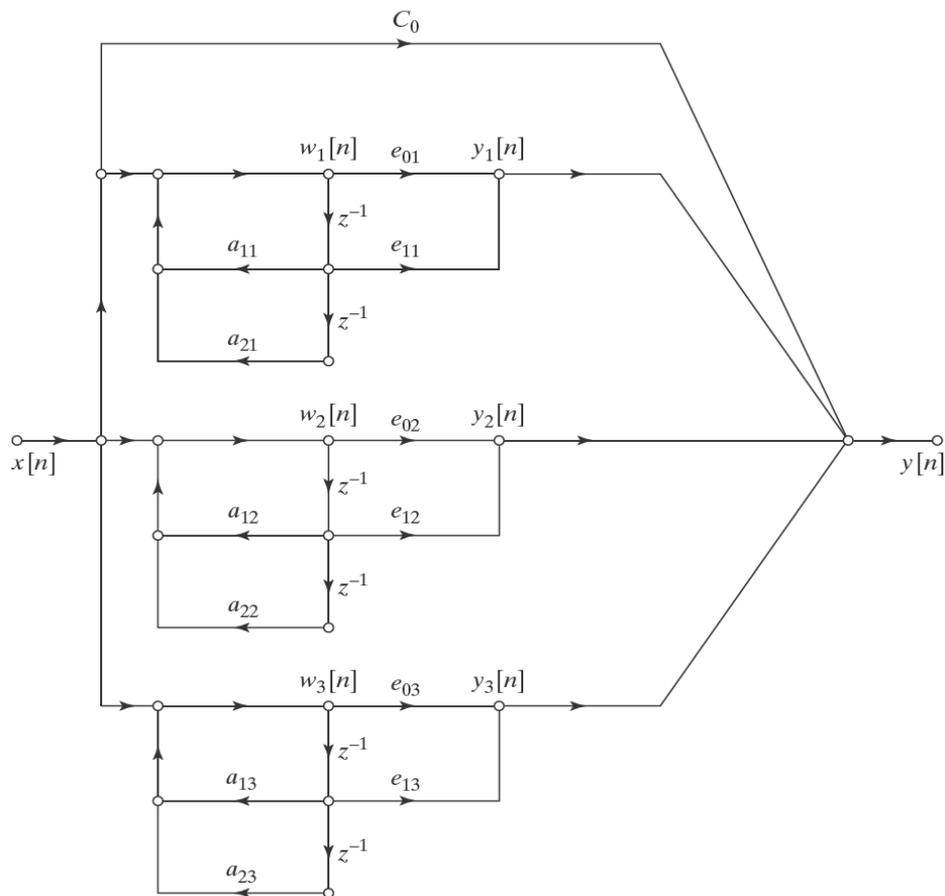
$$= \frac{(1 + z^{-1})(1 + z^{-1})}{(1 - 0,5z^{-1})(1 - 0,25z^{-1})}$$

Com $N = N_1 + 2N_2$, para $M \geq N \Rightarrow N_p = M - N$. Agrupando em pares os polos reais:

$$H(z) = \sum_{k=0}^{N_p} C_k z^{-k} + \sum_{k=1}^{N_s} \frac{e_{0k} + e_{1k}z^{-1}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}} \quad (34)$$

Onde $N_s = \lfloor (N + 1) / 2 \rfloor$ é o inteiro de maior valor contido em $(N + 1) / 2$. Se $N_p = M - N$ é negativo, o primeiro termo da equação desaparece. A Figura 12 traz um exemplo de um sistema de sexta ordem $M = N = 6$.

Figura 12 – Sistema de sexta ordem com polos agrupados em pares formando uma estrutura na forma paralela



Fonte: Oppenheim; Schafer, 2012.



As equações de diferença para um sistema de sexta ordem em paralelo formadas por seções de segunda ordem na forma direta II serão:

$$w_k(n) = a_{1k}w_k(n-1) + a_{2k}w_k(n-2) + x(n) \quad k = 1, \dots, N_s \quad (35)$$

$$y_k(n) = e_{0k}w_k(n) + e_{1k}w_k(n-1) \quad k = 1, \dots, N_s \quad (36)$$

$$y(n) = \sum_{k=0}^{N_p} C_k x(n-k) + \sum_{k=1}^{N_s} y_k(n) \quad (37)$$

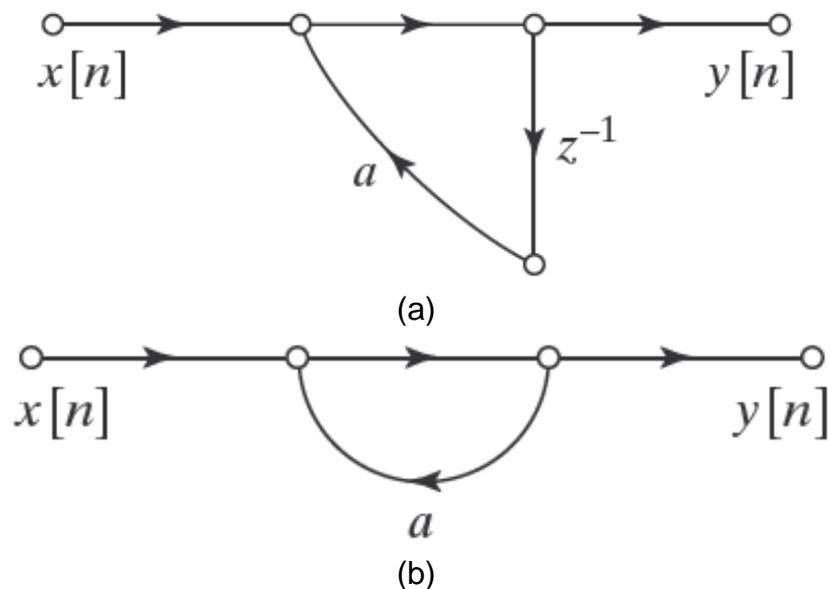
A quarta forma é a **realimentação**. Todos os diagramas de fluxo têm laços fechados de realimentação. Estes caminhos fechados, cuja direção única está indicada com uma seta, começam em um nó e direta ou indiretamente retornam a ele. Portanto, a variável do nó pode depender direta ou indiretamente dela mesma.

$$y(n) = ay(n-1) + x(n) \quad (38)$$

$$H(z) = \frac{1}{1 - az^{-1}} \quad (39)$$

Na Figura 13, é apresentado um exemplo simples de realimentação cuja equação de diferenças é a equação (38), sendo sua função de transferência apresentada na equação (39).

Figura 13 – (a) Laço de realimentação de um sistema IIR; (b) Sistema não computável



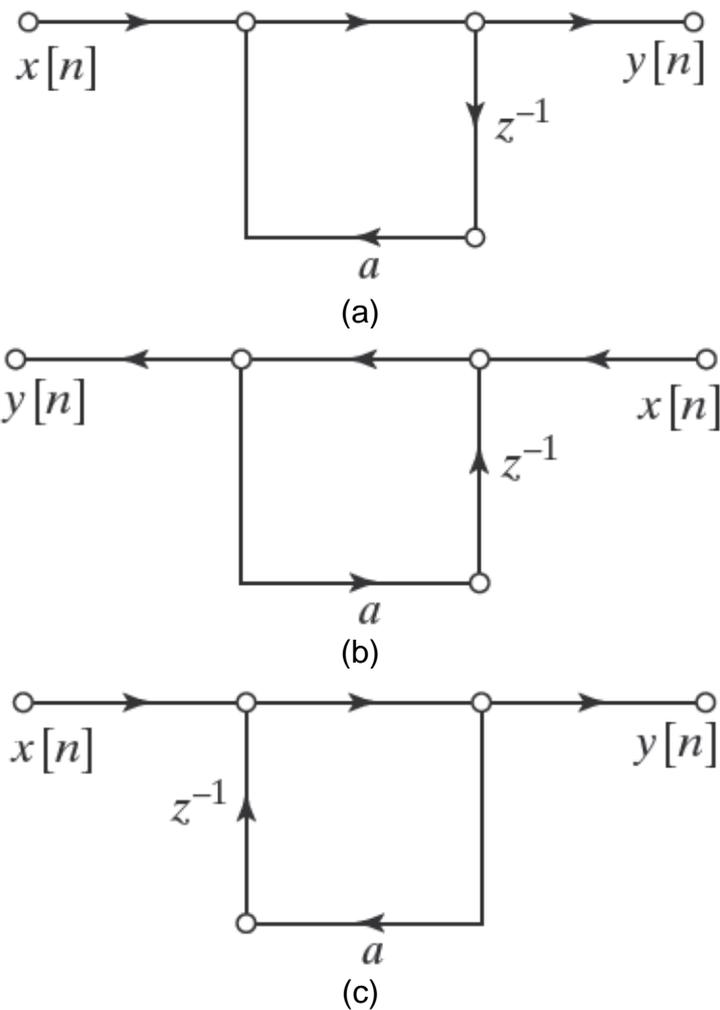
Fonte: Oppenheim; Schafer, 2012.

2.2 Formas transpostas



Transpor um diagrama de fluxo significa reverter a direção de todos os ramos na rede, mantendo os ganhos dos ramos e invertendo entrada e saída, de tal forma que os nós de saída se transformam em nós de fonte e vice-versa. Por exemplo, para achar a forma transposta da função de transferência da equação (39), procedemos como indicado na Figura 14.

Figura 14 – (a) Estrutura original; (b) estrutura transposta; (c) estrutura (b) reordenada



Fonte: Oppenheim; Schafer, 2012.

Neste exemplo, é possível verificar que tanto o sistema original quanto sua transposição têm a mesma função de sistema, o que não é tão óbvio em sistemas mais complexos (livro-texto páginas 235 a 238).

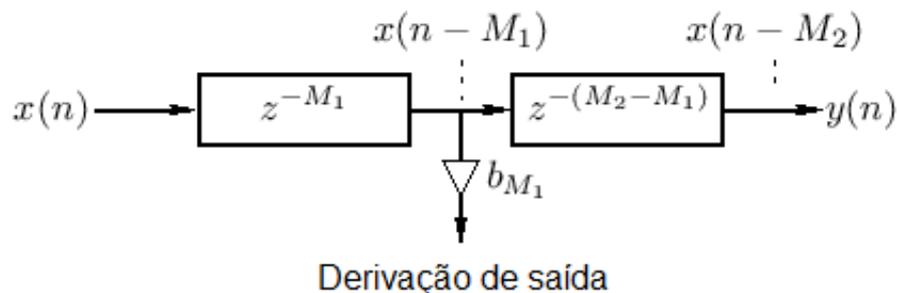
TEMA 3 – SISTEMAS FIR



A sigla FIR significa resposta ao impulso finita (do inglês Finite Impulse Response). Diferentemente do sistema IIR, a resposta ao impulso $h(t)$ do sistema FIR é exatamente igual a zero, quando $t > T$ para algum valor finito de T – por isso, dizemos que tem duração finita. Na prática, a resposta ao impulso, inclusive em sistemas IIR, aproxima-se de zero a partir de um certo ponto, e portanto pode ser desconsiderada a partir desse ponto. Mas os sistemas físicos que originam as respostas IIR ou FIR são diferentes. Os filtros analógicos compostos por componentes reais (resistores, capacitores e/ou indutores, e às vezes amplificadores lineares) geralmente são filtros IIR. Por outro lado, os filtros para tempos discretos (filtros digitais), baseados em linhas de atraso com derivação (TDL, do inglês Tapped Delay Line), que não usam realimentação, são necessariamente filtros FIR.

A Figura 15 mostra um exemplo de estrutura com TDL.

Figura 15 – Estrutura FIR com linha de atraso com derivação (TDL)



Fonte: Smith, 2010.

Os elementos reativos no filtro analógico têm “memória” e nunca atingem o zero após o impulso. No caso da estrutura FIR, quando o impulso chega ao fim da TDL, o sistema “não se lembra” do impulso e retorna a seu estado inicial, sendo sua resposta ao impulso igual a zero além desse ponto.

3.1 Formas

Primeiramente, temos a forma **direta**. No sistema causal FIR, a função só tem um polo em $z = 0$, e somente zeros. Como todos os coeficientes a_k são iguais a zero, a equação de diferença é:

$$y(n) = \sum_{k=0}^M b_k x(n - k) \quad (40)$$



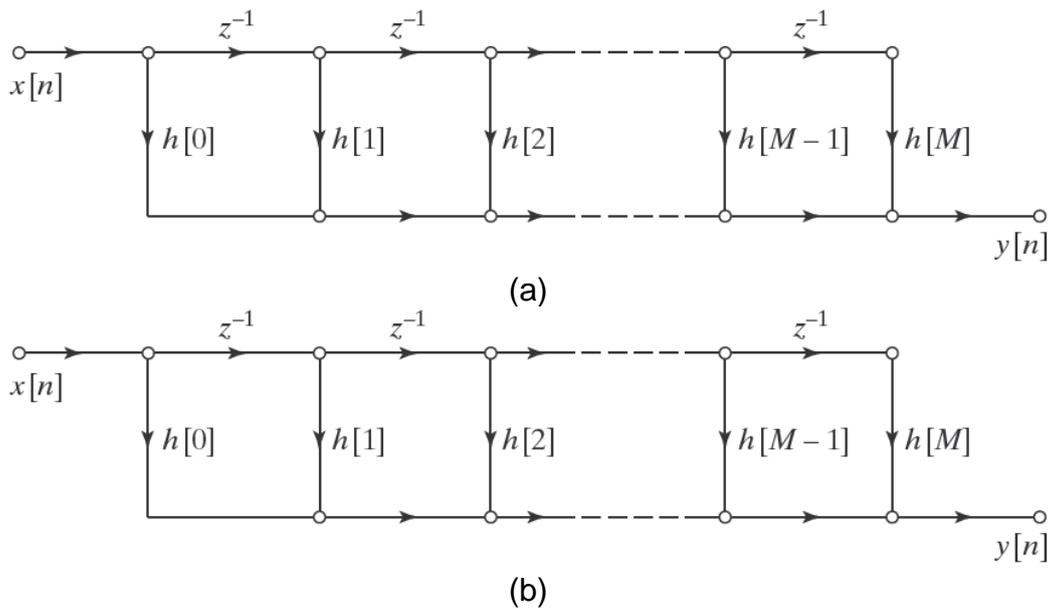
A resposta ao impulso é:

$$h(n) = \begin{cases} b_n & n = 0, 1, \dots, M \\ 0 & \text{caso contrário} \end{cases} \quad (41)$$

A Figura 16 mostra um sistema FIR na forma direta e na forma transposta.

Depois, temos a forma **em cascata**. Fatorando a função do sistema polinomial, é possível obter a forma em cascata para sistemas FIR.

Figura 16 – (a) Diagrama de fluxo de um sistema FIR na forma direta; (b) transposição do sistema FIR do ponto (a)

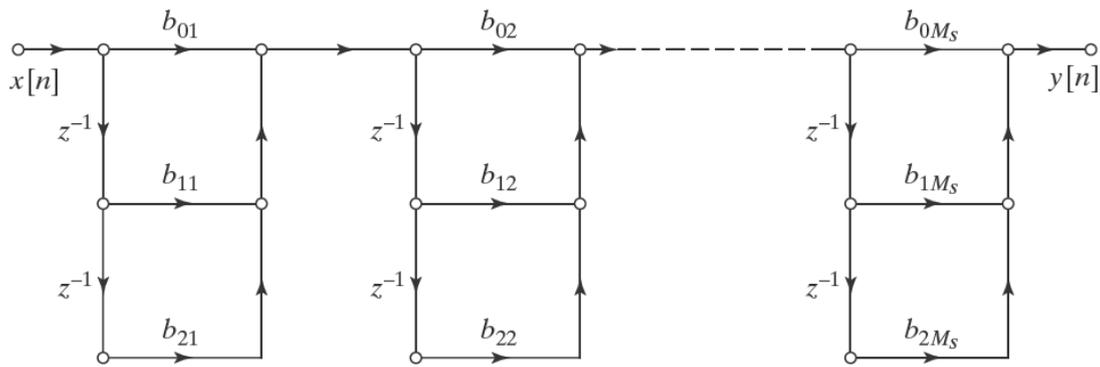


Fonte: Oppenheim; Schafer, 2012.

$$H(z) = \sum_{n=0}^M h(n)z^{-n} = \prod_{k=1}^{M_s} (b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}) \quad (42)$$

$M_s = \lfloor (M + 1) / 2 \rfloor$ é o inteiro de maior valor contido em $(M + 1) / 2$. Sendo M ímpar, um dos coeficientes b_{2k} deverá ser zero para $H(z)$ ter número ímpar de zeros reais. Na Figura 17, é apresentado o diagrama de fluxo do sistema em cascata. Esse diagrama é muito parecido com o da Figura 11, com os coeficientes a_{1k} e a_{2k} nulos.

Figura 17 – Diagrama de fluxo de um sistema FIR em cascata



Fonte: Oppenheim; Schafer, 2012.

Passamos, na sequência, aos **sistemas FIR de fase linear**. Se a resposta ao impulso satisfaz a condição de simetria, o sistema causal FIR terá fase linear generalizada:

$$h(M - n) = h(n) \quad n = 0, 1, \dots, M \quad (43)$$

$$h(M - n) = -h(n) \quad n = 0, 1, \dots, M \quad (44)$$

Sendo M um inteiro par (sistemas tipo I e III) e trabalhando com a equação de convolução discreta, temos:

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

$$\sum_{k=0}^{M/2-1} h(k)x(n - k) + h(M/2)x(n - M/2) + \sum_{k=M/2+1}^n h(k)x(n - k) =$$

$$\sum_{k=0}^{M/2-1} h(k)x(n - k) + h(M/2) - x(n - M/2) + \sum_{k=0}^{M/2-1} h(M - k)x(n - M + k)$$

Usando a equação 43, teremos um sistema tipo I:

$$y(n) = \sum_{k=0}^{M/2-1} h(k)(x(n - k) + x(n - M + k)) + h(M/2)x(n - M/2) \quad (45)$$

Para sistemas tipo III, usaremos a equação (44):

$$y(n) = \sum_{k=0}^{M/2-1} h(k)(x(n - k) + x(n - M + k)) \quad (46)$$

Quando M é um número inteiro ímpar, as equações serão para sistemas tipo II:

$$y(n) = \sum_{k=0}^{(M-1)/2} h(k)(x(n - k) + x(n - M + k)) \quad (47)$$

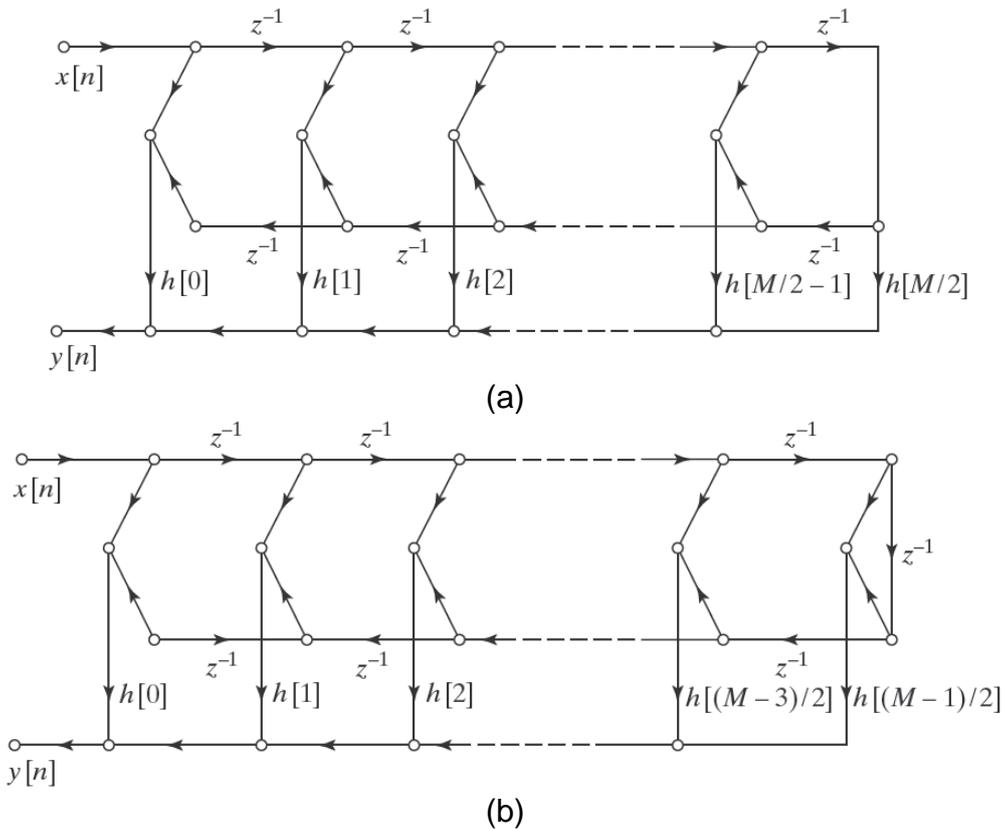


E para sistemas tipo IV:

$$y(n) = \sum_{k=0}^{(M-1)/2} h(k)(x(n-k) - x(n-M+k)) \quad (48)$$

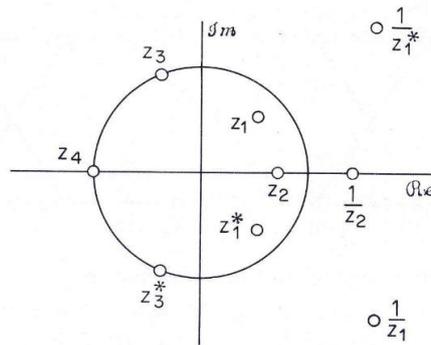
As equações (45) a (48) mostram que, no lugar de M multiplicadores de coeficientes, são usados $M/2 + 1$, $M/2$ ou $(M + 1) / 2$ multiplicadores. As condições de simetria mostradas nas equações (43) e (44) fazem com que os zeros do sistema sejam complexos conjugados. A Figura 18 mostra a estrutura para sistemas FIR de fase linear, e a Figura 19 mostra a simetria de zeros.

Figura 18 – (a) Estrutura de fase linear FIR para M par; (b) estrutura de fase linear FIR para M ímpar



Fonte: Oppenheim; Schafer, 2012.

Figura 19 – Simetria dos zeros para uma estrutura FIR de fase linear



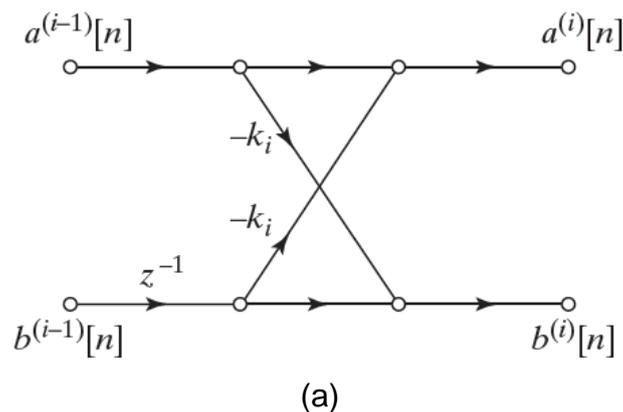
Fonte: Oppenheim; Schafer, 2012.

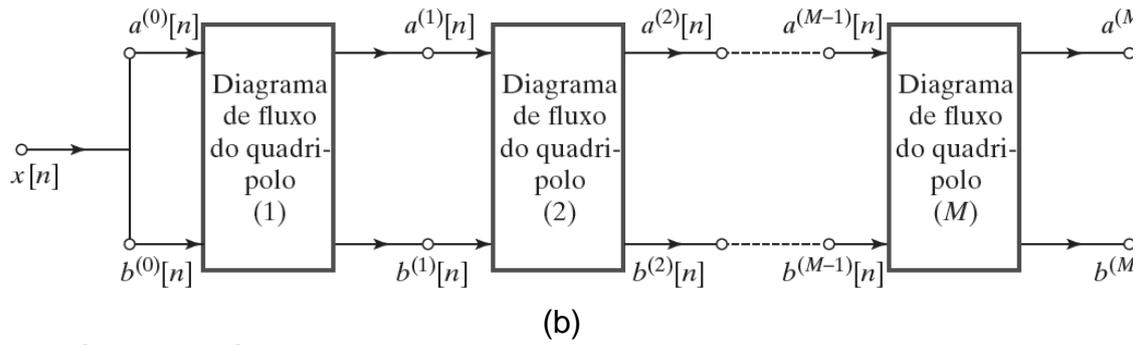
3.2 Filtros em treliça

Sistemas entrada única – saída única (SISO, do inglês Single Input – Single Output) podem ser implementados colocando em cascata subsistemas SISO na forma direta. Nesta seção, serão apresentados filtros em treliça (Lattice Filters, em inglês), nos quais os subsistemas são sistemas LIT de duas portas na estrutura borboleta. A cascata será convertida para a forma SISO requerida, com as duas extremidades terminando de acordo com as regras específicas do tipo de filtro. Existem dois tipos de estruturas FIR em treliça (MIT, 2006): estruturas só zeros e estruturas só polos.

Nesta seção, focaremos na estrutura só polos. A seção básica de duas portas (quadripolo) de um filtro FIR em treliça é uma estrutura em borboleta não recursiva, mostrada na Figura 20 (a). Na Figura 20 (b), podemos ver a estrutura em diagrama de blocos, e na Figura (21) a representação em diagrama de fluxo de um sistema FIR em treliça.

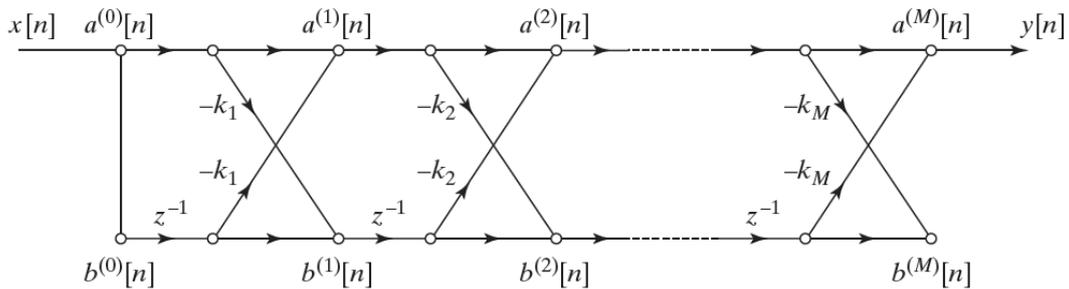
Figura 20 – (a) Diagrama de fluxo em estrutura borboleta (subsistema); (b) diagrama de blocos de uma cascata de subsistemas





Fonte: Oppenheim; Schafer, 2012.

Figura 21 – Diagrama de fluxo de um filtro em treliça



Fonte: Oppenheim; Schafer, 2012.

Na Figura 21, as variáveis $a^{(i)}(n)$ e $b^{(i)}(n)$ dependem da entrada $x(n)$ através de um conjunto de equações de diferença:

$$a^{(0)}(n) = b^{(0)}(n) = x(n) \quad (49)$$

$$a^{(i)}(n) = a^{(i-1)}(n) - k_i b^{(i-1)}(n-1) \quad i = 1, 2, \dots, M \quad (50)$$

$$b^{(i)}(n) = b^{(i-1)}(n-1) - k_i a^{(i-1)}(n) \quad i = 1, 2, \dots, M \quad (51)$$

$$y(n) = a^{(M)}(n) \quad (52)$$

A Figura 22 mostra o algoritmo para converter os parâmetros k em coeficientes do filtro FIR, e a Figura 23 mostra o algoritmo para converter os coeficientes do filtro FIR em parâmetros k .

Figura 22 – Algoritmo para converter os parâmetros k em coeficientes do filtro FIR



Algoritmo para conversão de parâmetros k em coeficientes

Dados k_1, k_2, \dots, k_M	
para $i = 1, 2, \dots, M$	
$\alpha_i^{(i)} = k_i$	Equação 6.66(b)
se $i > 1$, então para $j = 1, 2, \dots, i-1$	
$\alpha_j^{(i)} = \alpha_j^{(i-1)} - k_i \alpha_{i-j}^{(i-1)}$	Equação 6.66(a)
fim	
fim	
$\alpha_j = \alpha_j^{(M)} \quad j = 1, 2, \dots, M$	Equação 6.68(b)

Fonte: Oppenheim; Schafer, 2012.

Figura 23 – Algoritmo para converter os coeficientes do filtro FIR em parâmetros k

Algoritmo para conversão de coeficientes em parâmetros k

Dado $\alpha_j^{(M)} = \alpha_j \quad j = 1, 2, \dots, M$	
$k_M = \alpha_M^{(M)}$	Equação 6.69
para $i = M, M-1, \dots, 2$	
para $j = 1, 2, \dots, i-1$	
$\alpha_j^{(i-1)} = \frac{\alpha_j^{(i)} + k_i \alpha_{i-j}^{(i)}}{1 - k_i^2}$	Equação 6.71(a)
fim	
$k_{i-1} = \alpha_{i-1}^{(i-1)}$	Equação 6.71(b)
fim	

Fonte: Oppenheim; Schafer, 2012.

O processo completo de obtenção dos parâmetros k pode ser visto no livro-texto páginas 241 a 247 (Oppenheim; Schafer, 2012).

TEMA 4 – REPRESENTAÇÕES NUMÉRICAS

Ao implementar sistemas de processamento digital de sinais, os sinais e coeficientes devem ser representados num sistema numérico cuja precisão é finita. Cabe lembrar que valores digitais provêm de um processo de quantização no qual perde-se a informação do sinal analógico original. No caso dos sistemas digitais, se representarmos um número real na forma de complemento de dois, ele pode ter precisão infinita.

$$x = X_m \left(-b_0 + \sum_{i=1}^{\infty} b_i 2^{-i} \right) \quad (53)$$



X_m é um fator de escala arbitrário, b_0 é o bit de sinal e os fatores b_i podem ter valor “0” ou “1”. Para representar um número real qualquer, seria necessário um número infinito de bits, o que é praticamente irrealizável. Ao usarmos um conversor A/D, trabalharemos com um número finito de bits $B + 1$, portanto a equação 5 deve ser adequada a essa situação:

$$\hat{x} = Q_B(x) = X_m \left(-b_0 + \sum_{i=1}^B b_i 2^{-i} \right) = X_m \hat{x}_B \quad (54)$$

A menor diferença entre os números é:

$$\Delta = X_m 2^{-B} \quad (55)$$

A digitalização de um número implica em arredondamento ou truncamento, devido ao número limitado de bits. Assim, o erro de digitalização pode ser definido como:

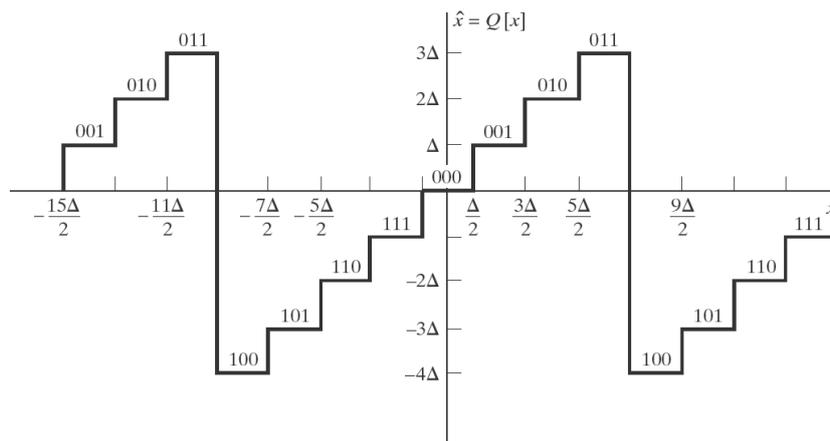
$$e = Q_B(x) - x \quad (56)$$

Para arredondamento em complemento de dois: $-\Delta / 2 < e \leq \Delta / 2$.

Para truncamento em complemento de dois: $-\Delta / 2 < e \leq 0$.

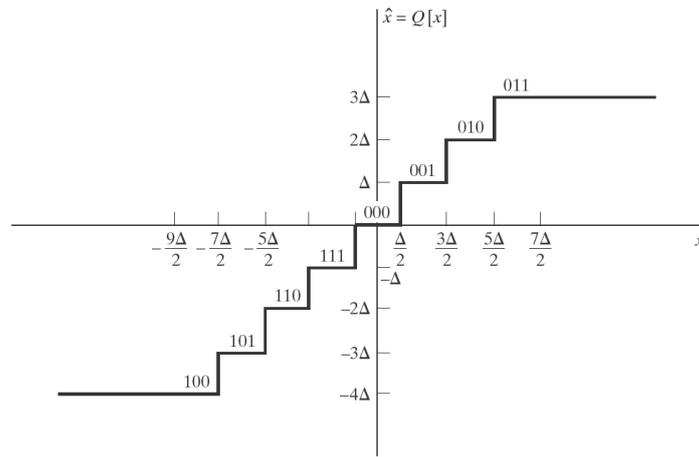
As Figuras 24 e 25 mostram as relações entrada – saída (função de transferência) para arredondamento e transbordamento de saturação (clipping).

Figura 24 – Digitalizador em complemento de dois com arredondamento, com efeito de transbordamento



Fonte: Oppenheim; Schafer, 2012.

Figura 25 – Efeito do transbordamento de saturação ou clipping

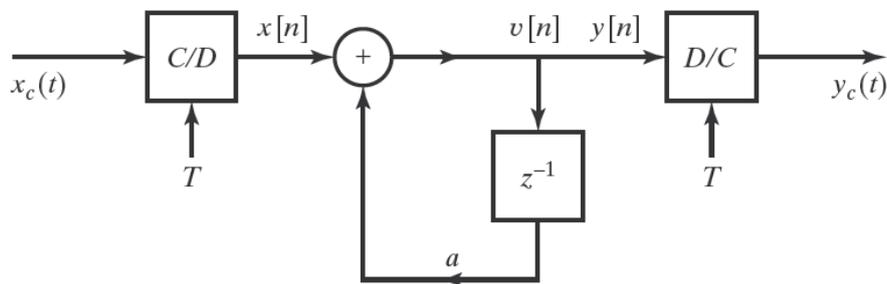


Fonte: Oppenheim; Schafer, 2012.

4.1 Digitalização

A Figura 26 mostra o diagrama em blocos de um sistema ideal com entrada em tempo contínuo (analógico), um bloco de conversão de tempo contínuo para discreto, um bloco de filtragem, um bloco de conversão de tempo discreto para contínuo, e o sinal de saída analógico.

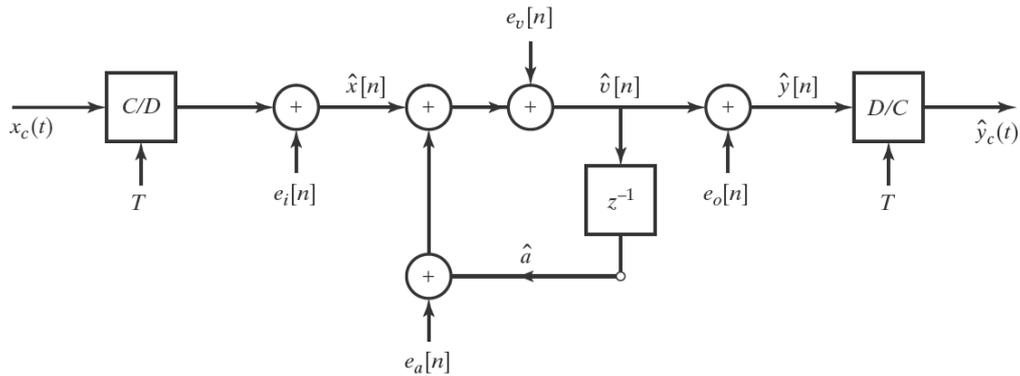
Figura 26 – Sistema ideal de processamento digital e conversão A/D e D/A



Fonte: Oppenheim; Schafer, 2012.

O modelo mais aproximado com a realidade é o mostrado na Figura 27, onde são consideradas cada uma das fontes de ruído.

Figura 27 – Modelo linearizado da implementação da filtragem em tempo discreto de um sinal analógico



Fonte: Oppenheim; Schafer, 2012.

TEMA 5 – EFEITOS DA DIGITALIZAÇÃO

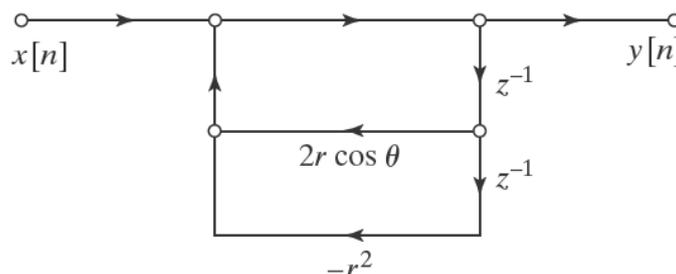
Os filtros digitais projetados, considerando aritmética de precisão infinita, podem apresentar sérios problemas de estabilidade quando implementados em um dispositivo DSP ou em um computador. Isso se deve aos efeitos dos erros de quantização. Como consequência, os polos e zeros originais do sistema racional se posicionam em outras posições no plano z , provocando modificações na resposta em frequência.

A quantização, pelo fato de que não é linear, introduz ruído no sinal. Por exemplo, um sinal analógico contínuo no tempo com variações infinitesimais de amplitude, quando digitalizado, tem que corresponder a um conjunto finito de valores discreto pré-determinados, arredondando os valores para o nível mais próximo, o que gera erros por arredondamento.

5.1 Sistemas IIR

Vamos considerar um sistema de polos complexos conjugados implementado na forma direta como mostra a Figura 28.

Figura 28 – Representação de um par de polos complexos conjugados implementado na forma direta

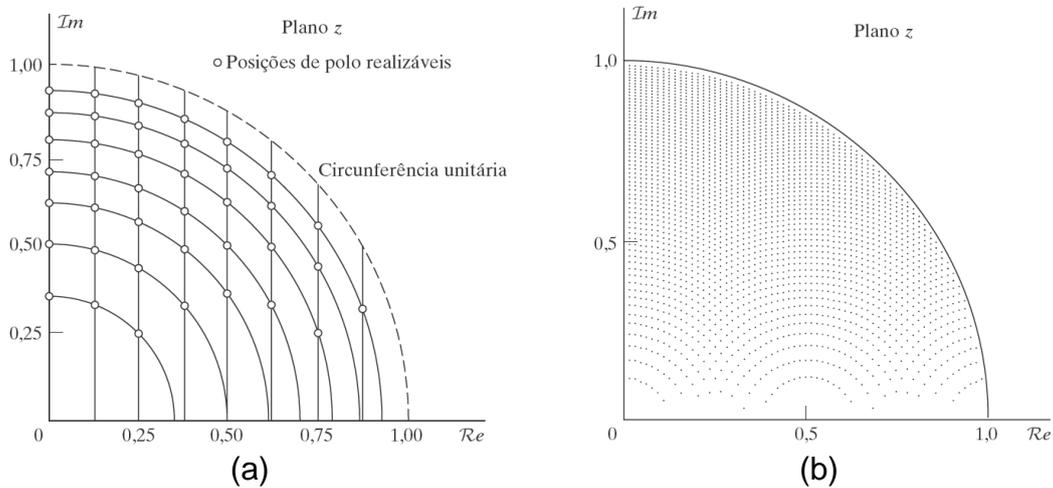


Fonte: Oppenheim; Schafer, 2012.



Para que o sistema seja mais estável, esses polos devem estar numa grade do plano z , como a mostrada na Figura 29.

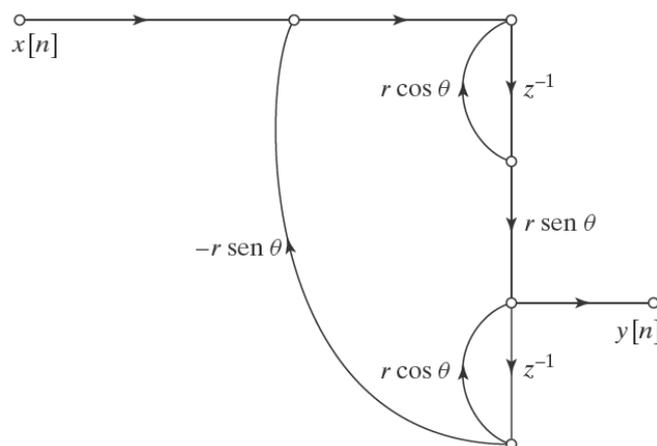
Figura 29 – (a) Digitalização de coeficientes com 4 bits; (b) digitalização de coeficientes com 7 bits



Fonte: Oppenheim; Schafer, 2012.

Uma estrutura alternativa de segunda ordem pode ser vista na Figura 30. O nome desta estrutura é *estrutura de forma acoplada para sistemas de segunda ordem*.

Figura 30 – Sistema de par de polos complexos conjugados implementados na forma acoplada



Fonte: Oppenheim; Schafer, 2012.

5.2 Sistemas FIR



Em sistemas causais FIR, como todos os polos estão em $z = 0$, só deveremos levar em conta a posição dos zeros do sistema. Em sistemas FIR, a estrutura na forma direta é comumente usada (em sistemas IIR de ordem alta, deve ser evitada). Consideramos a função de sistema de um sistema FIR na forma direta:

$$H(z) = \sum_{n=0}^M h(n)z^{-n} \quad (57)$$

Digitalizando os coeficientes de $h(n)$:

$$\hat{h}(n) = h(n) + \Delta h(n) \quad (58)$$

Portanto, a função do sistema digitalizado é:

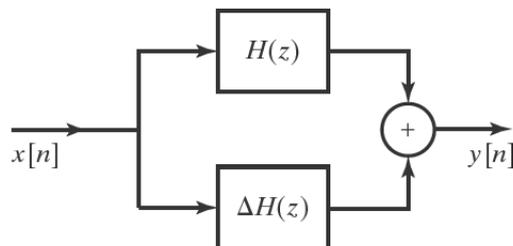
$$\hat{H}(z) = \sum_{n=0}^M \hat{h}(n)z^{-n} = H(z) + \Delta H(z) \quad (59)$$

Com:

$$\Delta H(z) = \sum_{n=0}^M \Delta h(n)z^{-n} \quad (60)$$

A função de sistema para o sistema digitalizado é a mostrada na Figura 31.

Figura 31 – Função do sistema para o sistema FIR digitalizado



Fonte: Oppenheim; Schafer, 2012.

Para aprofundar os conhecimentos dos efeitos da digitalização e do ruído de arredondamento, consultar as páginas 251 a 277 do livro-texto.

FINALIZANDO

Nesta aula, estudamos as estruturas básicas para implementação de filtros digitais. Estudamos diagramas de blocos para equações de diferença lineares com coeficientes constantes. Verificamos os diagramas de fluxo dessas equações e estudamos as estruturas básicas para o desenvolvimento de filtros



digitais. Verificamos também que há problemas inerentes à digitalização do sinal, devido aos ruídos de quantização no processo de conversão A/D. Essas estruturas são a base para o desenvolvimento de filtros digitais, que serão estudados na aula 6.

Para finalizar, cabe lembrar que esta aula é somente um guia, e que o aluno, além de ler este guia de estudos, deve estudar pelo livro-texto e pelo material de leitura obrigatória disponibilizado na aula.

- Capítulo 6: Estruturas para sistemas de tempo discreto – página 222.



REFERÊNCIAS

MIT – Massachusetts Institute of Technology. **IIR, FIR Filter Structures**. 2006. Disponível em: <<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-341-discrete-time-signal-processing-fall-2005/lecture-notes/lec07.pdf>>. Acesso em: 5 jun. 2018.

MITRA, S. K. **Digital Signal Processing: a computer based approach**. 4. ed. New York: McGraw-Hill Education, 2010.

OPPENHEIM, V.; SCHAFER, R. W. **Processamento em Tempo Discreto de Sinais**. 3. ed. São Paulo: Pearson Education do Brasil, 2012.

SMITH, J. O. I. **Physical Audio Signal Processing**. 2010. Disponível em: <https://ccrma.stanford.edu/~jos/pasp/Tapped_Delay_Line_TDL.html>. Acesso em: 5 jun. 2018.