

3

Conceitos básicos de filtragem digital multitaxa

3.1

Introdução

Filtragem digital multitaxa é um tipo de processamento de sinais em que são utilizados certos operadores que alteram a taxa de amostragem do sinal digital, a partir da remoção ou adição de amostras na seqüência de entrada original. Os principais operadores multitaxa são os decimadores e os expansores, que operam em conjunto com filtros digitais, formando as estruturas de filtragem digital multitaxa. Estas estruturas se combinam e formam bancos de filtros digitais [11,12].

Este capítulo introduz os conceitos básicos da filtragem digital multitaxa e apresenta identidades e equivalências utilizadas para melhorar a eficiência do processo. Também será mostrado como podem ser projetados os bancos de filtros de reconstrução perfeita do tipo árvore binária utilizados neste trabalho. São apresentados exemplos com o objetivo de ilustrar o processo de divisão em sub-bandas seguido de reconstrução perfeita do sinal.

3.2

Filtros digitais

Filtros analógicos passivos convencionais, usualmente são entendidos como combinações de resistores, capacitores e indutores em um circuito elétrico, permitindo atuação seletiva, seja no tempo ou em frequência, buscando selecionar informação no sinal tratado. Os filtros digitais são representados por equações discretas que aplicadas a um sinal digital de entrada $x(n)$, geram uma saída $y(n)$, de acordo com a Figura (3.1), com frequências dentro de certa faixa selecionada. Eles podem ser do tipo FIR (finite impulse response), ou IIR (infinite impulse response), conforme será mostrado nas Seções 3.2.1 e 3.2.2.

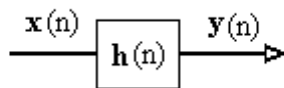


Figura (3.1) – Filtro digital

3.2.1 Estrutura básica de filtros digitais FIR

Um filtro digital linear e causal FIR de tamanho N é caracterizado pela função de transferência representada pela Eq. (3.1), que é um polinômio em z^{-1} , de ordem $N-1$.

$$H(z) = \sum_{k=0}^{N-1} h(k).z^{-k} \quad (3.1)$$

No domínio do tempo, a relação entrada-saída dos filtros FIR é descrita pelo somatório convolução

$$y(n) = \sum_{k=0}^{N-1} h(k).x(n-k) . \quad (3.2)$$

O diagrama de blocos da Figura (3.2) e a Eq. (3.3) representam um filtro FIR de terceira ordem.

$$y(n) = h(0).x(n) + h(1).x(n-1) + h(2).x(n-2) + h(3).x(n-3) \quad (3.3)$$

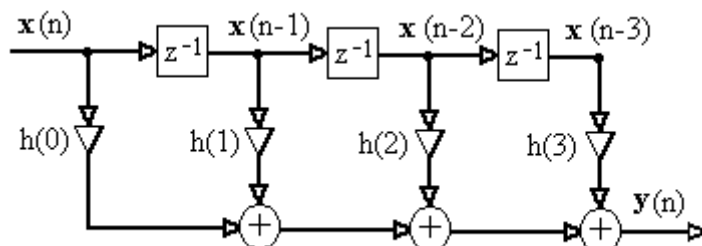


Figura (3.2)– Exemplo de estrutura de filtros digitais FIR de terceira ordem.

3.2.2 Estrutura básica de filtros digitais IIR

Um filtro digital linear IIR de ordem $N-1$ é caracterizado pela função de transferência representada na Eq. (3.4), sendo uma razão entre dois polinômios em z^{-1} , ambos de grau $N-1$.

$$H(z) = \frac{\sum_{k=0}^{N-1} p(k).z^{-k}}{1 + \sum_{k=1}^{N-1} d(k).z^{-k}} \quad (3.4)$$

No domínio do tempo, a relação entrada-saída de um filtro IIR de terceira ordem é representada pela Eq. (3.5) e pela Figura (3.3).

$$y(n) = p(0).x(n) + p(1).x(n-1) + p(2).x(n-2) + p(3).x(n-3) - d(1).y(n-1) - d(2).y(n-2) - d(3).y(n-3) \quad (3.5)$$

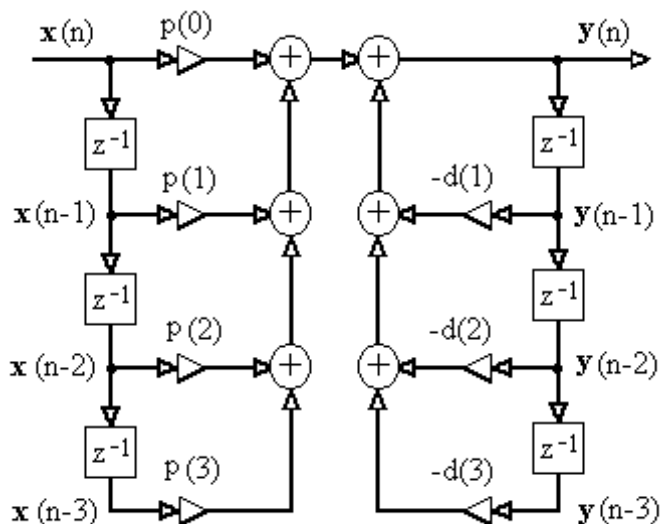


Figura (3.3) – Exemplo de estrutura não canônica de filtros digitais IIR de terceira ordem

3.3 O decimador

A Figura (3.4) apresenta o bloco operador decimador, também chamado de “Down-Sampler” ou redutor da taxa de amostragem.

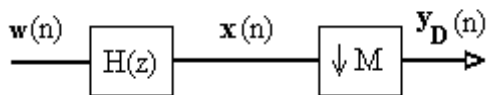


Figura (3.4) – Bloco operador decimador por M antecedido por filtro.

Para uma seqüência de entrada representada pelo vetor $\mathbf{x}(n)$, a seqüência de saída é representada pelo vetor $\mathbf{y}_D(n)$, de acordo com a Eq. (3.6),

$$\mathbf{y}_D(n) = \mathbf{x}(M.n), \quad (3.6)$$

onde, M é um número inteiro. Apenas as amostras de $\mathbf{x}(n)$ em que n é múltiplo de M são utilizadas pelo decimador, conforme mostrado na Figura (3.5). Por exemplo, se um conjunto de amostras for decimado por 2, a saída $\mathbf{y}_D(n)$ será gerada apenas com os valores de $\mathbf{x}(n)$ para n par, ou n ímpar. Assim, $\mathbf{y}_D(n)$ terá a metade do número de amostras da seqüência original, ou seja, a taxa de amostragem fica reduzida a metade. A Figura (3.5) ilustra o efeito, no domínio do tempo, da aplicação do operador decimador a uma certa seqüência $\mathbf{x}(n)$.

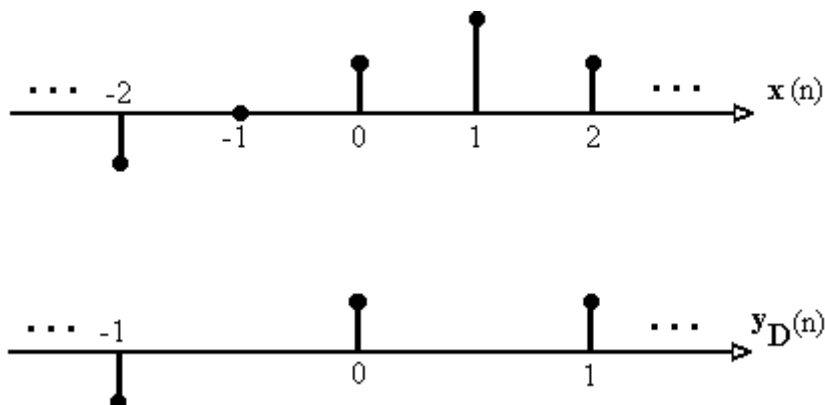


Figura (3.5) – Demonstração de decimação no domínio do tempo, $M=2$.

Conforme mostrado nas Figuras (3.6) e (3.7), o espectro do sinal no domínio da frequência se alarga após a decimação, podendo ocorrer superposição ou “aliasing”. Este fenômeno ocorre quando o espectro do sinal original é maior que π/M . Quando ocorre a superposição, informações do sinal são perdidas, o que pode impossibilitar a sua reconstrução. Assim, o bloco operador decimador é usualmente antecedido por filtro chamado de “anti-aliasing”, para garantir que não ocorra superposição. Em geral, estes filtros são passa-baixa, com ganho unitário e frequência de corte em π/M . As Figuras (3.6) e (3.7) ilustram o efeito, no domínio da frequência, da aplicação do operador decimador a uma certa seqüência.

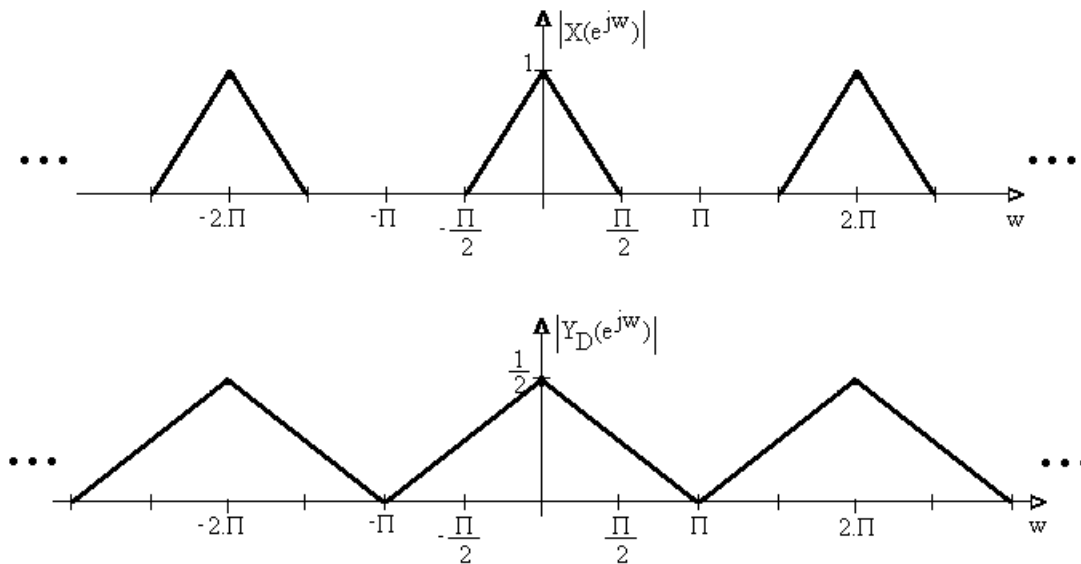


Figura (3.6) – Ilustração de decimação no domínio da frequência (sem a ocorrência de superposição ou “aliasing”), $M=2$.

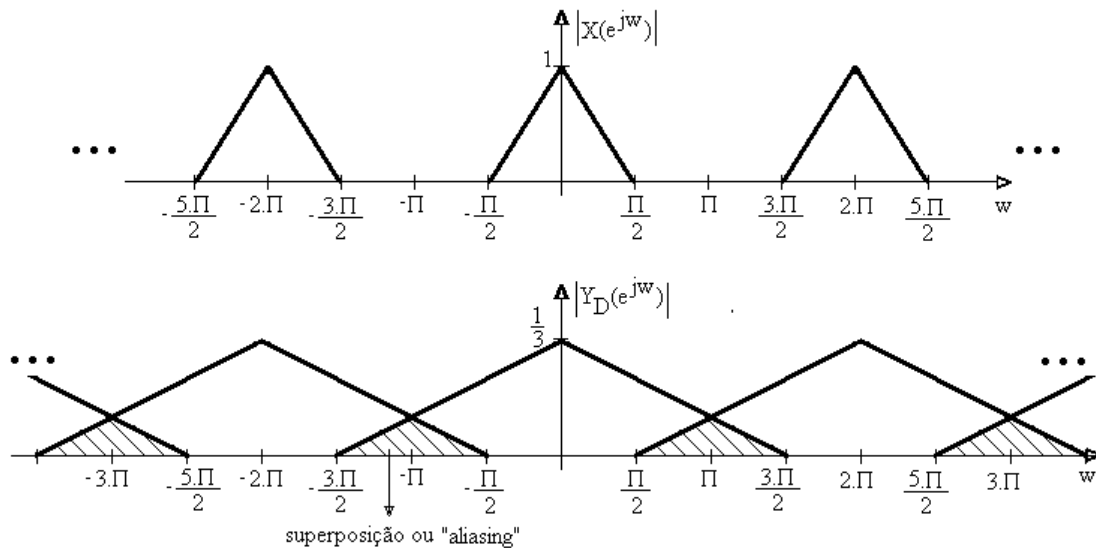


Figura (3.7) – Ilustração de decimação no domínio da frequência (com a ocorrência de superposição ou “aliasing”), $M=3$.

3.4 O expansor ou interpolador

A Figura (3.8) apresenta o bloco operador expansor, também chamado de “Up-Sampler” ou operador que aumenta a taxa de amostragem.

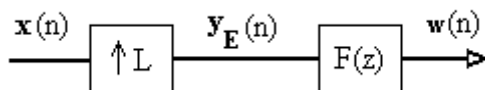


Figura (3.8) – Bloco expansor por L seguido de filtro seletivo.

Para uma seqüência de entrada $x(n)$, a saída será:

$$y_E(n) = \begin{cases} x(n/L), & \dots n = 0, \pm L, \pm 2L, \dots \\ 0, & \dots n \neq 0, \pm L, \pm 2L, \dots \end{cases}, \quad (3.7)$$

onde L é um número inteiro. O expansor produzirá na saída uma réplica de $x(n)$, se n é múltiplo de L , caso contrário a saída gerada possui valor zero, conforme mostrado na Figura (3.9). Um filtro passa baixa normalmente é utilizado depois do expansor. A Figura (3.9) ilustra o efeito, no domínio do tempo, da aplicação do operador expansor a uma certa seqüência.

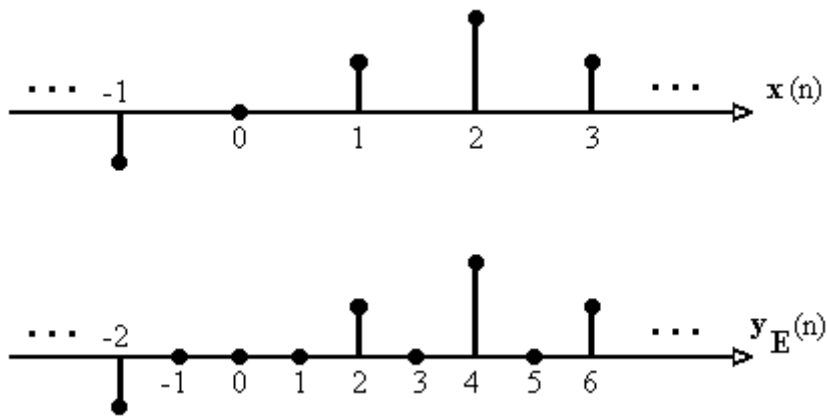


Figura (3.9) – Demonstração do expansor no domínio do tempo, $L=2$

A Figura (3.10) ilustra o efeito, no domínio da frequência, da aplicação do operador expansor a uma certa seqüência.

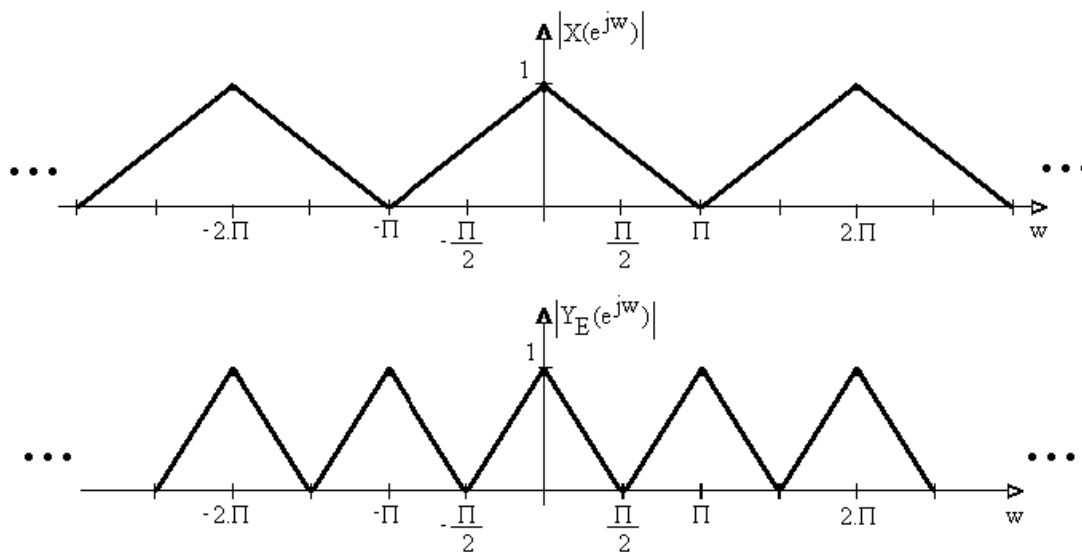


Figura (3.10) – Ilustração do expansor no domínio da frequência, $L=2$

Como pode ser observado na Figura (3.10), o expansor gera um espectro de frequência com imagens replicadas do espectro original. Um filtro passa-baixa com ganho L e frequência de corte em π/L normalmente é utilizado após o expansor para eliminar estas imagens, de maneira que o sinal volte a ter o mesmo espectro original, apenas com taxa de amostragem L vezes maior.

3.5 Equivalências e Identidades nobres

A Figura (3.11) apresenta algumas equivalências, ou identidades simples, utilizadas na representação por diagramas de blocos de sistemas multitaxa. Cada uma das equivalências abaixo é válida também para expansores substituindo os decimadores mostrados na Figura (3.11).

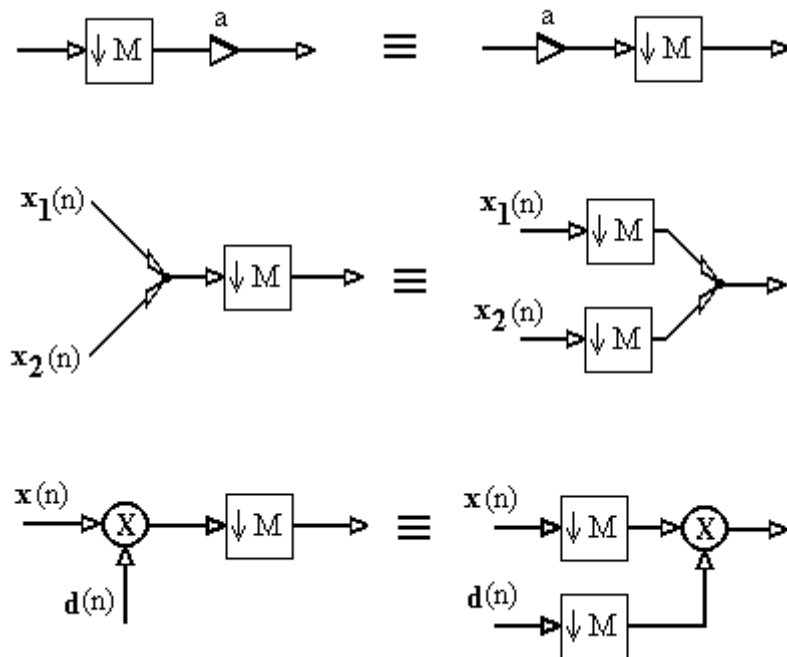


Figura (3.11) – Equivalências ou identidades simples utilizadas em sistemas multitaxa.

A Figura (3.12) apresenta identidades nobres muito utilizadas na representação por diagramas de blocos e na implementação otimizada de sistemas multitaxa. A primeira identidade representa a alteração na ordem de um filtro, que se localiza após um decimador, quando a sua posição é trocada e ele passa a estar à frente do decimador. A segunda identidade representa a alteração na ordem de um filtro, que se localiza antes de um expensor, quando a sua posição é trocada e ele passa para depois do expensor.

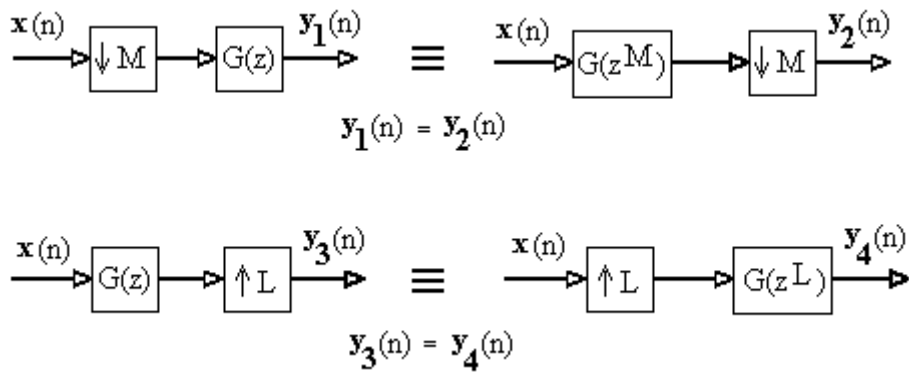


Figura (3.12) – Identities nobres utilizadas em sistemas multitaxa

3.6 Bancos de filtros digitais

Um banco de filtros digitais é uma estrutura formada por filtros digitais passa-banda que possuem uma entrada e/ou uma saída em comum, conforme mostrado na Figura (3.13). A Figura (3.13)(a) representa um banco de filtros de análise e a Figura (3.13)(b) um banco de filtros de síntese. O banco de filtros de análise é utilizado para a decomposição do sinal de entrada em várias sub-bandas de frequência, isto é, a banda de frequências original é subdividida em várias partes. O banco de filtros de síntese combina os sinais das várias sub-bandas em um único sinal, reconstruindo a banda de frequências original. Portanto, a filtragem multitaxa é, neste caso, composta por duas etapas. Primeiramente, utiliza-se um banco de análise para dividir o sinal de entrada em outros, com taxa de amostragem mais baixa, cada qual representando uma faixa de frequência do sinal original. Em seguida, a reconstrução é feita através do banco de síntese, que recompõe o sinal.

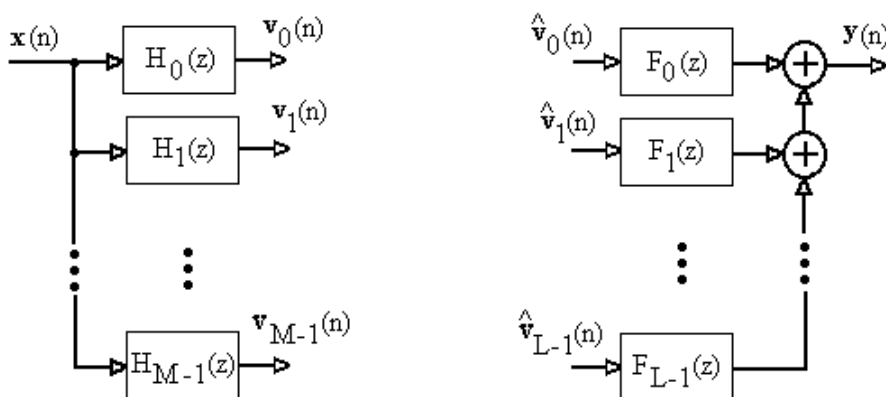


Figura (3.13) – (a) Banco de filtros de análise; (b) Banco de filtros de síntese.

A Figura (3.14) mostra o processo completo de filtragem por banco de filtros digitais multitaxa, utilizando uma estrutura do tipo árvore binária.

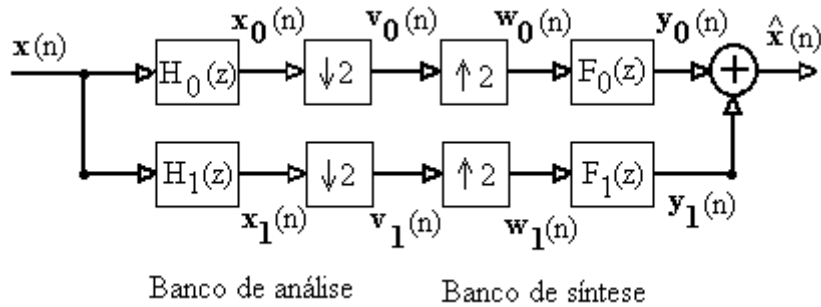


Figura (3.14) – Representação de filtragem multitaxa por árvore binária.

Cada ramo da árvore binária mostrada na Figura (3.14) pode ser novamente subdividido em sub-bandas de frequência, em quantos níveis forem necessários. O problema associado à utilização de bancos de filtros digitais é que o sinal reconstruído fica atrasado em relação ao sinal de entrada. O atraso é tanto maior quanto maior for a ordem dos filtros utilizados e as subdivisões (níveis) da árvore binária mostrada na Figura (3.14). Isto é um problema que deve ser considerado no projeto do simulador. Devem ser utilizados filtros de ordem pequena, e níveis de subdivisões tais que o simulador em sub-bandas gere atrasos compatíveis com os atrasos dos equipamentos simulados.

Uma forma interessante e muito utilizada de se reduzir o esforço computacional do banco de filtros é a Representação Polifásica [11,12,17], cuja estrutura com duas sub-bandas é mostrada na Figura (3.15). Como pode ser observado, de acordo com as identidades nobres o decimador é alocado para o início do processo e o expansor para o final, diminuindo a ordem dos filtros utilizados e em consequência o número de operações matemáticas de adições e multiplicações por unidade de tempo.

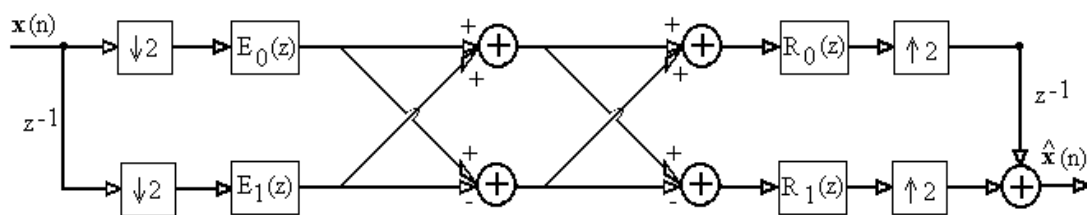


Figura (3.15) - Representação por componentes polifásicos.

3.7 Bancos de filtros de reconstrução perfeita

Nesta seção será mostrado como se executa um projeto de um banco de filtros de reconstrução perfeita [11,12,18], a partir de um filtro passa baixa $H_0(z)$. Considere o filtro passa-baixa FIR de terceira ordem e coeficientes a_0, a_1, a_2 e a_3 representado pela Eq. (3.8), onde z^0 representa o valor da amostra atual, z^{-1} o valor da amostra um instante anterior, z^{-2} o valor da amostra dois instantes atrás e z^{-3} o valor da amostra três instantes atrás. Assim, observa-se que a ordem do filtro indica quantas amostras anteriores são utilizadas no processo de filtragem e que z^{-1} corresponde a um atraso.

$$H_0(z) = a_0.z^0 + a_1.z^{-1} + a_2.z^{-2} + a_3.z^{-3} \quad (3.8)$$

O vetor de entrada $\mathbf{x}(n)$ mostrado a seguir, contém os valores das amostras que serão filtradas a cada instante de tempo, isto é, cada coluna do vetor corresponde a um determinado instante de tempo. Neste exemplo, cada valor do vetor de entrada é representado por uma letra do alfabeto.

$$\mathbf{x}(n) = [A \ B \ C \ D \ E \ F \ G \ H \dots]$$

O banco de filtros de reconstrução perfeita é projetado com estrutura do tipo árvore binária, conforme mostrado na Figura (3.14). Os demais filtros que formam o banco ($H_1(z)$, $F_0(z)$ e $F_1(z)$) devem ser escolhidos a partir de $H_0(z)$, de maneira a atender a certas condições, que possibilitam a reconstrução perfeita do sinal de entrada $\mathbf{x}(n)$. A função flip é definida abaixo e sua função é inverter a seqüência, o primeiro coeficiente passa a ser o último e o último passa a ser o primeiro, isto é, ela transforma a seqüência em um espelho da original. As condições de reconstrução perfeita, utilizadas na escolha dos filtros são descritas a seguir.

$$\mathbf{h}_0(n) = [a_0 \ a_1 \ a_2 \ a_3]; \quad (3.9)$$

$$\tilde{\mathbf{h}}_0(n) = [a_0 \ -a_1 \ a_2 \ -a_3]; \quad (3.10)$$

$$\mathbf{h}_1(n) = -\text{flip}(\tilde{\mathbf{h}}_0(n)) = [a_3 \ -a_2 \ a_1 \ -a_0]; \quad (3.11)$$

$$\mathbf{f}_0(n) = \text{flip}(\mathbf{h}_0(n)) = [a_3 \ a_2 \ a_1 \ a_0]; \quad (3.12)$$

$$\mathbf{f}_1(n) = \text{flip}(\mathbf{h}_1(n)) = [-a_0 \ a_1 \ -a_2 \ a_3]. \quad (3.13)$$

Assim, no ramo de baixa frequência, o sinal de entrada $\mathbf{x}(n)$ é filtrado por $H_0(z)$, de acordo com a Eq. (3.9), obtendo-se $\mathbf{x}_0(n)$, que é a seguir decimado por $M=2$, obtendo-se $\mathbf{v}_0(n)$.

$$\mathbf{x}_0(n) = \begin{bmatrix} a_0.A \\ a_0.B + a_1.A \\ a_0.C + a_1.B + a_2.A \\ a_0.D + a_1.C + a_2.B + a_3.A \\ a_0.E + a_1.D + a_2.C + a_3.B \\ a_0.F + a_1.E + a_2.D + a_3.C \\ a_0.G + a_1.F + a_2.E + a_3.D \\ a_0.H + a_1.G + a_2.F + a_3.E \\ \dots \end{bmatrix}$$

$$\mathbf{v}_0(n) = \begin{bmatrix} a_0.A \\ a_0.C + a_1.B + a_2.A \\ a_0.E + a_1.D + a_2.C + a_3.B \\ a_0.G + a_1.F + a_2.E + a_3.D \\ \dots \end{bmatrix}$$

Neste ponto do processo, a taxa de amostragem é reduzida à metade da taxa original, isto é, trabalha-se com a metade do número de pontos. A seguir, $\mathbf{v}_0(n)$ é interpolado com $L=2$, obtendo-se $\mathbf{w}_0(n)$, que passa pelo filtro $F_0(z)$, cujos coeficientes são mostrados na Eq. (3.12), gerando $\mathbf{y}_0(n)$.

$$\mathbf{w}_0(n) = \begin{bmatrix} a_0.A \\ 0 \\ a_0.C + a_1.B + a_2.A \\ 0 \\ a_0.E + a_1.D + a_2.C + a_3.B \\ 0 \\ a_0.G + a_1.F + a_2.E + a_3.D \\ 0 \\ \dots \end{bmatrix}$$

$$\mathbf{y}_0(n) = \begin{bmatrix} a_3.a_0.A \\ a_2.a_0.A \\ a_3.a_0.C + a_3.a_1.B + a_3.a_2.A + a_1.a_0.A \\ a_2.a_0.C + a_2.a_1.B + a_2^2.A + a_0^2.A \\ a_3.a_0.E + a_3.a_1.D + a_3.a_2.C + a_3^2.B + a_1.a_0.C + a_1^2.B + a_1.a_2.A \\ a_2.a_0.E + a_2.a_1.D + a_2^2.C + a_2.a_3.B + a_0^2.C + a_0.a_1.B + a_0.a_2.A \\ a_3.a_0.G + a_3.a_1.F + a_3.a_2.E + a_3^2.D + a_1.a_0.E + a_1^2.D + a_1.a_2.C + a_1.a_3.B \\ a_2.a_0.G + a_2.a_1.F + a_2^2.E + a_2.a_3.D + a_0^2.E + a_0.a_1.D + a_0.a_2.C + a_0.a_3.B \\ \dots \end{bmatrix}$$

No ramo de alta frequência, é seguido o mesmo procedimento, isto é, o sinal de entrada $\mathbf{x}(n)$ é filtrado pelo passa-alta $H_1(z)$, cujos coeficientes são mostrados na Eq. (3.11), obtendo-se $\mathbf{x}_1(n)$, que é a seguir decimado por $M=2$, obtendo-se $\mathbf{v}_1(n)$. A seguir, $\mathbf{v}_1(n)$ é interpolado com $L=2$, obtendo-se $\mathbf{w}_1(n)$, que passa pelo filtro $F_1(z)$, cujos coeficientes são mostrados na Eq. (3.13), gerando $\mathbf{y}_1(n)$.

O sinal reconstruído, $\hat{\mathbf{x}}(n)$, é a soma dos sinais contidos nos ramos de baixa e de alta frequência, de acordo com a Eq. (3.14).

$$\hat{\mathbf{x}}(n) = \mathbf{y}_0(n) + \mathbf{y}_1(n) \quad (3.14)$$

O resultado obtido é representado pela matriz $\hat{\mathbf{x}}(n)$ representada a seguir.

$$\hat{\mathbf{x}}(n) = \begin{bmatrix} 0 \\ (a_1.a_3 + a_2.a_0).A \\ (a_1.a_3 + a_2.a_0).B \\ (a_1.a_3 + a_2.a_0).C + (a_0^2 + a_1^2 + a_2^2 + a_3^2).A \\ (a_1.a_3 + a_2.a_0).D + (a_0^2 + a_1^2 + a_2^2 + a_3^2).B \\ (a_1.a_3 + a_2.a_0).E + (a_1.a_3 + a_2.a_0).A + (a_0^2 + a_1^2 + a_2^2 + a_3^2).C \\ (a_1.a_3 + a_2.a_0).F + (a_1.a_3 + a_2.a_0).B + (a_0^2 + a_1^2 + a_2^2 + a_3^2).D \\ a_1.a_3 + a_2.a_0).G + (a_1.a_3 + a_2.a_0).C + (a_0^2 + a_1^2 + a_2^2 + a_3^2).E \\ \dots \end{bmatrix}$$

Os valores dos coeficientes dos filtros devem ser escolhidos de modo que $(a_1.a_3+a_2.a_0)$ tenha valor nulo e $(a_0^2+ a_1^2+ a_2^2+ a_3^2)$ seja aproximadamente igual a um ou igual a um número chamado de fator de escala, que multiplica ou divide o vetor de saída no final do processo de filtragem. Assim a seqüência de entrada (A, B, C, D, E, F, G, H, ...) é reconstruída perfeitamente, apenas com alguns instantes de atraso, de acordo com a ordem dos filtros escolhidos. No exemplo apresentado, a seqüência de entrada foi reconstruída com 3 atrasos, pois foram utilizados filtros de terceira ordem e um único nível de decomposição. Qualquer banco de filtros digitais, se projetado de maneira a atender às condições apresentadas acima, é de reconstrução perfeita.

3.8 Exemplos de bancos de filtros

Nos exemplos apresentados a seguir, foram utilizados bancos com filtros digitais de terceira ordem, do tipo Doubechies 2 [19] e estruturas do tipo árvore binária, conforme descrito na Seção 3.6.

No primeiro exemplo, um sinal de entrada $\mathbf{x}(n)$, composto por um somatório de sinais senoidais contendo várias freqüências diferentes, é filtrado pelo banco de reconstrução perfeita, gerando o vetor de saída $\hat{\mathbf{x}}(n)$. No gráfico mostrado na Figura (3.16), são plotados os valores de $\mathbf{x}(n)$ e $\hat{\mathbf{x}}(n)$. Observa-se que $\hat{\mathbf{x}}(n)$ é idêntico a $\mathbf{x}(n)$, porém 3 instantes de tempo atrasado. A Figura (3.17) mostra uma

ampliação da região marcada na Figura (3.16), indicando o atraso de um sinal em relação ao outro.

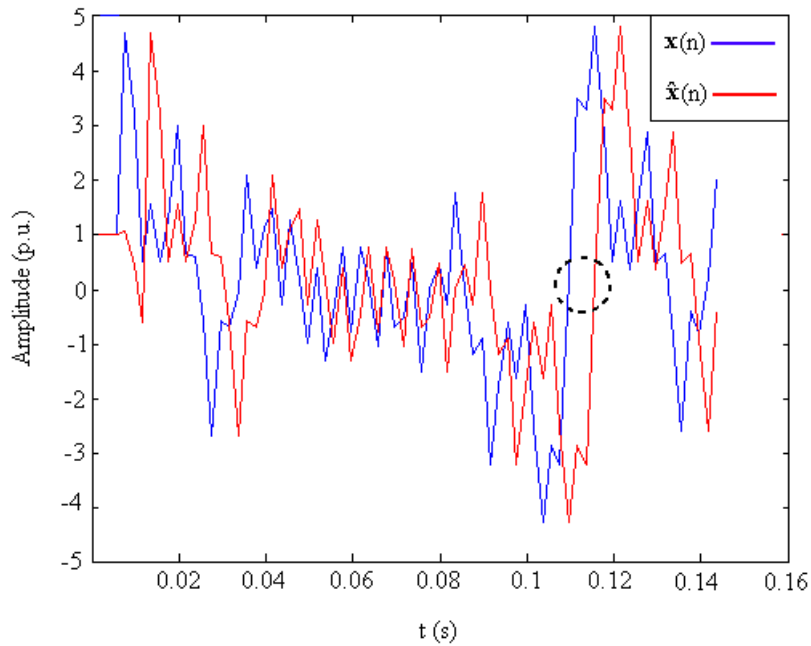


Figura (3.16) – Sinal de entrada $x(n)$ e sinal reconstruído $\hat{x}(n)$

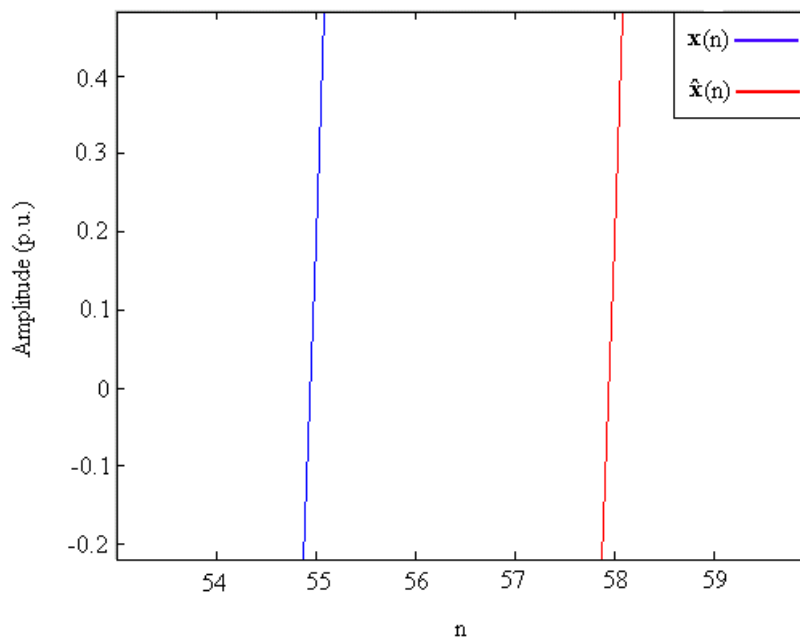


Figura (3.17) – Atraso do sinal reconstruído $\hat{x}(n)$ em relação ao sinal de entrada $x(n)$

No gráfico mostrado na Figura (3.18), $\hat{x}(n)$ é adiantado 3 instantes de tempo e plotado sobre $x(n)$. Observa-se que os dois sinais são realmente idênticos e se

superpõem completamente. Também na Figura (3.18) é mostrado o erro obtido no processo ($\text{erro} = \mathbf{x}(n) - \hat{\mathbf{x}}(n)$), que é quase nulo (da ordem de 10^{-4}).

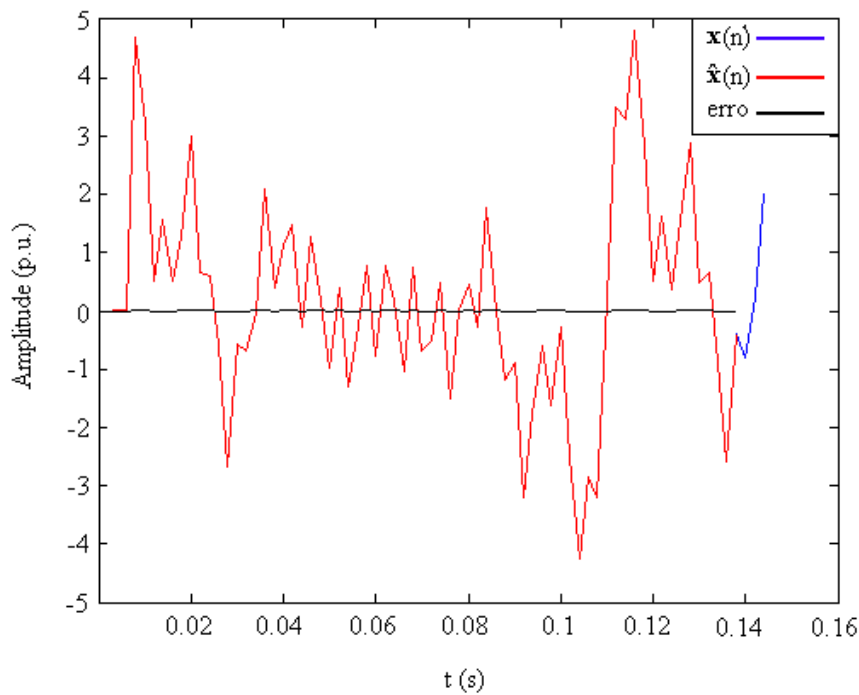


Figura (3.18) – Superposição dos sinais $\mathbf{x}(n)$, $\hat{\mathbf{x}}(n)$ e erro

A Figura (3.19), apresentada a seguir, mostra um sinal senoidal, $\mathbf{s}(n)$, com frequência 60 Hz e frequência de amostragem igual a 1076 Hz. Neste exemplo, o sinal $\mathbf{s}(n)$ é filtrado pelo mesmo banco de reconstrução perfeita do exemplo anterior, gerando um sinal de saída $\hat{\mathbf{s}}(n)$, idêntico ao sinal de entrada $\mathbf{s}(n)$, porém atrasado 3 instantes de tempo, conforme já era esperado.

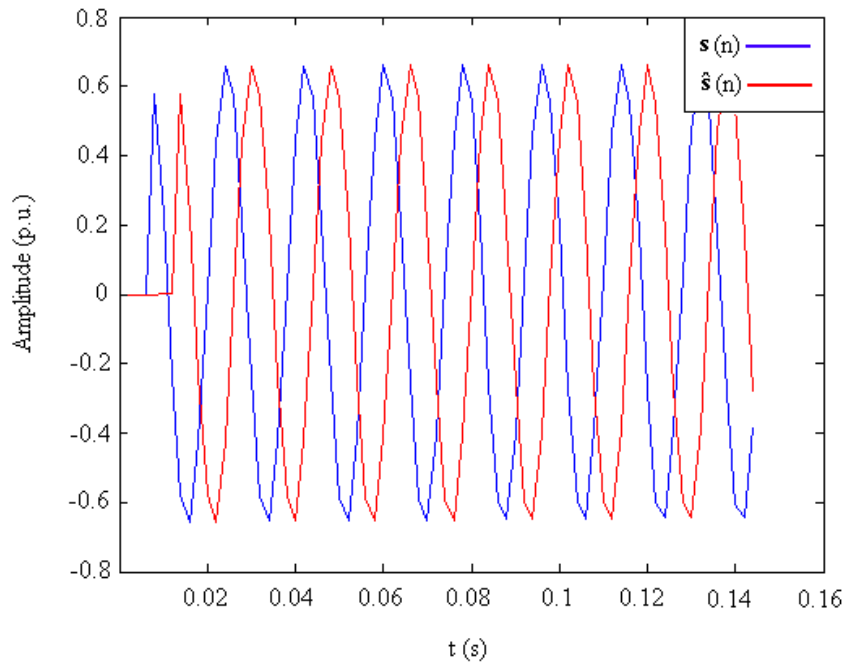


Figura (3.19) – Exemplo de filtragem por banco de reconstrução perfeita.

3.9 Conclusões

Neste capítulo foram apresentados os fundamentos da filtragem digital multitaxa, seus operadores básicos, os decimadores e expansores e algumas equivalências e identidades nobres utilizadas para melhorar a eficiência do processo de filtragem. Foi mostrado como pode ser projetado um banco de filtros de reconstrução perfeita do tipo árvore binária e apresentados exemplos de divisão em sub-bandas seguido de reconstrução perfeita do sinal.