

MASCARAMENTO DE RESPOSTA EM FREQUÊNCIA: OTIMIZAÇÃO PELO  
ALGORITMO WLS-CHEBYSHEV E IMPLEMENTAÇÃO EFICIENTE EM  
BANCOS DE FILTROS MODULADOS POR COSSENO

Luiz Cláudio Rodrigues de Barcellos

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PRO-  
GRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FE-  
DERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁ-  
RIOS PARA OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGE-  
NHARIA ELÉTRICA.

Aprovada por:

---

Prof. Paulo Sergio Ramirez Diniz, Ph.D.

---

Prof. Sergio Lima Netto, Ph.D.

---

Prof. Eduardo Antônio Barros da Silva, Ph.D.

---

Prof. Jose Antônio Apolinário Jr., D.Sc.

RIO DE JANEIRO, RJ - BRASIL

ABRIL DE 2001

BARCELLOS, LUIZ CLÁUDIO RODRIGUES DE  
Mascaramento de Resposta em  
Frequência: Otimização pelo Algoritmo  
WLS-Chebyshev e Implementação Eficiente  
em Bancos de Filtros Modulados por  
Cosseno [Rio de Janeiro] 2001

xvi,128 pp 29,7 cm (COPPE/UFRJ,  
M.Sc., Engenharia Elétrica, 2001)

Tese - Universidade Federal do Rio de  
Janeiro, COPPE

1.Filtros Digitais 2.Mascaramento  
de Resposta em Frequência 3.Bancos  
de Filtros 4.Algoritmo WLS-Chebyshev  
5.Modulação por Cosseno 6.Redução de  
Complexidade Computacional

I.COPPE/UFRJ      II.Título (série)

## Agradecimentos

Os meus agradecimentos são para todos aqueles ...

A todas estas pessoas, muito obrigado, esperando que possa retribuir à altura tudo o que fizeram por mim.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

MASCARAMENTO DE RESPOSTA EM FREQUÊNCIA: OTIMIZAÇÃO PELO ALGORITMO WLS-Chebyshev E IMPLEMENTAÇÃO EFICIENTE EM BANCOS DE FILTROS MODULADOS POR COSSENO

Luiz Cláudio Rodrigues de Barcellos

Abril/2001

Orientador: Paulo Sergio Ramirez Diniz

Orientador: Sergio Lima Netto

Programa de Engenharia Elétrica

A utilização da técnica de mascaramento de resposta em frequência vem sendo amplamente estudada, pois permite a redução drástica da complexidade computacional, ou seja, o número de operações matemáticas necessárias no filtro para se gerar uma amostra de saída. A redução da complexidade computacional é obtida realizando um filtro de faixa de transição estreita a partir de filtros de bandas de transição mais largas (filtros de mascaramento) e de um filtro de base, que também apresenta faixa de transição larga, que é interpolado para gerar transições abruptas. Análises são realizadas nas várias bandas de filtros utilizando esta técnica e é possível mostrar que a complexidade computacional pode ser ainda mais reduzida melhorando-se a otimização dos filtros componentes. Particularmente, o método de otimização WLS-Chebyshev resulta em uma boa otimização, salvando cerca de

20% de coeficientes em relação ao método de mascaramento com filtros puramente minimax.

Na segunda parte desta tese, aplicamos o conceito de mascaramento de resposta em frequência em uma aplicação de banco de filtros modulados por cosseno, particularmente para o sistema transmultiplexador. Utilizando fatores particulares de interpolação do filtro base, é possível realizar uma estrutura eficiente para filtros protótipos de faixa de transição estreita, e obter a redução da complexidade computacional destes sistemas, assim como possibilitar a redução das variáveis do sistema, facilitando o processo de otimização do banco de filtros.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

FREQUENCY-RESPONSE MASKING: OPTIMIZATION USING THE  
WLS-CHEBYSHEV ALGORITHM AND EFFICIENT IMPLEMENTATION ON  
COSINE MODULATED FILTER BANKS

Luiz Cláudio Rodrigues de Barcellos

April/2001

Advisor: Paulo Sergio Ramirez Diniz

Advisor: Sergio Lima Netto

Department: Electrical Engineering

The use of the frequency-response masking approach is being severally studied, once it ensure drastic reduction of the computational complexity (number of mathematical operations to generate one output sample) of the system. The reduction of the computational complexity is achieved by obtaining a filter with sharp transition band from the design of smoother transition band filters (masking filters) and of a base filter, that also presents a smoother transition band, that is then interpolated to generate sharp transition bands. Analysis are then realized on the various filter bands by using this approach and it is possible to show that the computational complexity can be further reduced by using enhanced optimization algorithms. Particularly, the optimization approach known as WLS-Chebyshev re-

sults in a very good optimization, saving about 20% of the coefficients in respect to the frequency-masking approach using a minimax optimization algorithm.

On the second part of the thesis, we apply the concept of the frequency-response masking on a cosine-modulated filter bank, particularly on a transmultiplex system. By using some particular interpolation factors, it is possible to realize an efficient structure for prototype filters with sharp transition bands, and to reduce the computational complexity of such systems, making it possible to reduce the number of variables on the system, and making the optimization process of the bank filter easier to handle.

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Método de Mascaramento da Resposta em Frequência</b>	<b>4</b>
2.1	Introdução . . . . .	4
2.2	Estrutura e construção da resposta do filtro FRM . . . . .	5
2.3	Exemplo numérico . . . . .	11
2.3.1	Projeto do filtro FIR minimax . . . . .	11
2.3.2	Projeto do filtro FRM utilizando o algoritmo minimax - projeto com bandas não críticas. . . . .	13
2.4	Margens de desvio de ganho . . . . .	14
2.5	Bandas especialmente críticas . . . . .	20
2.6	Conclusões . . . . .	23
<b>3</b>	<b>Algoritmo WLS-Chebyshev</b>	<b>24</b>
3.1	Introdução . . . . .	24
3.2	Algoritmo WLS . . . . .	24
3.3	O Algoritmo de restrição de picos WLS-Chebyshev . . . . .	28
3.4	Exemplo numérico . . . . .	30
3.5	Vantagens do filtro WLS-Chebyshev . . . . .	30
3.6	Conclusões . . . . .	32
<b>4</b>	<b>Projeto de filtros FRM usando o algoritmo WLS-Chebyshev</b>	<b>34</b>
4.1	Introdução . . . . .	34
4.2	Projeto de filtros de máscara com o algoritmo WLS-Chebyshev . . . . .	36
4.2.1	Projeto de filtros de máscara independentes . . . . .	38
4.2.2	Projeto com tomada da função de erro na resposta final . . . . .	39



4.3	Projeto do filtro base com o algoritmo WLS-Chebyshev . . . . .	41
4.4	Exemplos numéricos . . . . .	46
4.5	Conclusões . . . . .	49
<b>5</b>	<b>Banco de Filtros</b>	<b>55</b>
5.1	Introdução . . . . .	55
5.2	Banco de filtros criticamente decimados . . . . .	56
5.2.1	Construção da resposta em frequência do banco de filtros . . . . .	59
5.3	Banco de filtros modulados por cosseno . . . . .	61
5.4	Estrutura eficiente do CMFB . . . . .	63
5.5	Reconstrução perfeita . . . . .	66
5.6	Reconstrução próxima da perfeita . . . . .	69
5.7	Otimização do banco de filtros . . . . .	69
5.8	Conclusões . . . . .	71
<b>6</b>	<b>Banco de filtros FRM</b>	<b>72</b>
6.1	Introdução . . . . .	72
6.2	CMFRM . . . . .	72
6.3	CMFRM com interpolação múltipla ao número de canais . . . . .	73
6.3.1	Análise de resposta em frequência . . . . .	79
6.4	CMFRM com amostragem igual ao número de canais . . . . .	80
6.5	CMFRM com amostragem menor que o número de canais . . . . .	82
6.6	CMFRM com fator de interpolação genérico . . . . .	87
6.7	Exemplos numéricos . . . . .	89
6.8	Conclusões . . . . .	100
<b>7</b>	<b>Conclusões</b>	<b>103</b>
<b>A</b>	<b>Estudo das margens de desvios de ganho do FRM</b>	<b>107</b>
<b>B</b>	<b>Exemplo de construção de sinais nos filtros CMFRMs</b>	<b>115</b>
B.1	Análise da estrutura CMFRM generalizada . . . . .	115
B.1.1	Filtros componentes . . . . .	116
B.1.2	Projeto CMFB . . . . .	116
B.1.3	Projeto CMFRM equivalente . . . . .	118

B.2	Análise da estrutura simplificada para o caso $L = M$ . . . . .	120
B.2.1	Filtros componentes . . . . .	120
B.2.2	Projeto CMFB . . . . .	121
B.2.3	Projeto CMFRM equivalente . . . . .	122
<b>C</b>	<b>Reconstrução perfeita para o CMFRM utilizando uma variável a ser otimizada</b>	<b>124</b>
C.1	Reconstrução perfeita com 2 canais e $K = 2$ . . . . .	125
C.2	Reconstrução perfeita com 2 canais e $K = 3$ . . . . .	128

# Lista de Figuras

2.1	A estrutura geral de um filtro FRM. . . . .	5
2.2	Obtenção das saídas do filtro interpolado. . . . .	6
2.3	Processo de construção para o filtro FRM, com banda de mascaramento positivo no final da banda passante. . . . .	7
2.4	Processo de construção para o filtro FRM, com a banda de mascaramento negativo no final da banda passante. . . . .	8
2.5	Projeto convencional de um filtro FIR passa-baixas a partir do algoritmo minimax. . . . .	12
2.6	Filtro de base para o projeto FRM com bandas não críticas. . . . .	15
2.7	Funcionamento da máscara positiva para o projeto do filtro FRM com bandas não críticas. . . . .	16
2.8	Funcionamento da máscara negativa para o projeto do filtro FRM com bandas não críticas. . . . .	17
2.9	Resposta resultante em magnitude do filtro FRM com bandas não críticas. . . . .	18
2.10	Especificação de margem para o projeto dos filtros componentes. O subíndice <i>PROJ</i> se refere aos valores especificados para o filtro final, enquanto que $\delta_p$ e $\delta_s$ são os desvios permitidos a um dos filtros componentes. Se a margem for igual a 0%, então $\delta_p = \delta_{p,PROJ}$ e $\delta_s = \delta_{s,PROJ}$ . . . . .	20
2.11	Operações de mascaramento para a obtenção da resposta final, onde se notam as bandas não críticas (sem indicação sob o eixo das frequências), críticas (linhas simples) e as bandas especialmente críticas (linhas duplas). . . . .	22

3.1	Projetos minimax (linha tracejada) e mínimos quadrados (linha contínua), para um filtro de ordem 46 e freqüências de corte de $\omega_p = 0.4\pi$ e $\omega_s = 0.6\pi$ . . . . .	26
3.2	Resposta em magnitude de um filtro WLS-Chebyshev com os quatro primeiros picos da banda de rejeição restritos. . . . .	31
4.1	Filtros FIR projetados com o algoritmo minimax, para $\omega_p = 0.4\pi$ e $\omega_s = 0.6\pi$ , com ordens $N=19, 20$ e $21$ . Note que apenas com $N = 21$ o projeto entra na especificação de $\delta_p \leq 0.1$ dB e $\delta_s \leq -40$ dB, e ainda fornece uma margem de desvios razoável de aproximadamente 20%. . . . .	37
4.2	A banda crítica de rejeição de um filtro FRM típico. Note que apenas alguns poucos lóbulos dos filtros de máscara estão presentes nesta faixa de freqüência. . . . .	44
4.3	Filtro de base para o projeto FRM com filtro de base e máscaras WLS-Chebyshev e tomada de erro na resposta final, usando bandas não críticas. . . . .	50
4.4	Funcionamento da máscara positiva para o projeto do filtro FRM com filtro de base e máscaras WLS-Chebyshev e tomada de erro na resposta final, usando bandas não críticas. . . . .	51
4.5	Funcionamento da máscara negativa para o projeto do filtro FRM com filtro de base e máscaras WLS-Chebyshev e tomada de erro na resposta final, usando bandas não críticas. . . . .	52
4.6	Resposta resultante em magnitude do filtro FRM com filtro de base e máscaras WLS-Chebyshev e tomada de erro na resposta final, usando bandas não críticas. . . . .	53
5.1	Diagrama básico de um banco de filtros. . . . .	56
5.2	O sistema transmultiplexador. . . . .	58
5.3	Exemplo típico das respostas em magnitude dos filtros de análise e de síntese de um CMFB com $M = 8$ canais. . . . .	60
5.4	A estrutura eficiente de implementação do banco de análise de um CMFB. . . . .	65

5.5	A estrutura eficiente de implementação do banco de síntese de um CMFB. . . . .	66
6.1	Construção da resposta impulsiva de um canal $m$ no banco de análise de um CMFB. Neste caso, o número de canais é $M = 2$ e, portanto, a função $c_{m,n}$ se repete a cada 4 amostras do sinal de entrada. . . . .	75
6.2	A construção da resposta impulsiva obtida realizando a convolução das respostas impulsivas de dois filtros para o CMFB com $M = 2$ canais. . . . .	76
6.3	A estrutura em cascata (filtro interpolado seguido pelas decomposições polifásicas do filtro de máscara) do CMFRM para o caso $L = 2K_1M$ . . . . .	78
6.4	A estrutura completa (os dois ramos) do CMFRM para o caso em que $L = 2K_1M$ . . . . .	78
6.5	A interpolação por $L = 2K_1M$ faz com que os centros das bandas do filtro interpolado caiam sobre $\frac{k\pi}{2M}$ , não permitindo realizar a faixa de transição do filtro de base nesta frequência. . . . .	80
6.6	A estrutura do CMFRM com $L = M$ e $K_2 = 1$ . . . . .	81
6.7	Filtragem de vários sinais a partir de um único filtro, utilizando o conceito de sistemas multitaxas. . . . .	84
6.8	Redução dos decimadores em $Q$ na saída das componentes polifásicas do filtro de máscara. . . . .	85
6.9	A estrutura do banco de análise do CMFRM para $L < M$ . Para simplificação do desenho, apenas as saídas da componente polifásica $E_0(-z^2)$ estão representadas. Estas saídas somam apenas a cada $Lq$ componentes polifásicas. . . . .	86
6.10	A estrutura do banco de síntese para o CMFRM com $L < M$ . . . . .	87
6.11	A estrutura do banco de análise do CMFRM para $L = K_a M + \frac{M}{K_b}$ . Para simplificação do desenho, apenas as saídas da componente polifásica $E_0(-z^2)$ estão representadas. Estas saídas somam apenas a cada $L_2q$ componentes polifásicas. . . . .	90
6.12	Composição do filtro FRM para o exemplo 1. . . . .	92

6.13	Respostas em magnitude do filtro protótipo e das funções $T_i(e^{j\omega})$ e $T_{k,l}(e^{j\omega})$ responsáveis pelos valores de $\delta_1$ , $\delta_2$ e pelos valores de ISI e de ICI para o exemplo 1. . . . .	94
6.14	Composição do filtro FRM para o exemplo 2. . . . .	95
6.15	Respostas em magnitude do filtro protótipo e das funções $T_i(e^{j\omega})$ e $T_{k,l}(e^{j\omega})$ responsáveis pelos valores de $\delta_1$ e $\delta_2$ e pelos valores de ISI e ICI para o exemplo 2. . . . .	97
6.16	Composição do filtro FRM para o exemplo 4, e o filtro protótipo obtido.	99
6.17	Respostas em magnitude do banco de filtros e das funções $T_i(e^{j\omega})$ e $T_{k,l}(e^{j\omega})$ responsáveis pelos valores de $\delta_1$ , $\delta_2$ e pelos valores de ISI e ICI para o exemplo 3. . . . .	101
A.1	Detalhes dos desvios de ganho dos filtros componentes, para a faixa (1). . . . .	108
A.2	Detalhe das faixas de transição dos filtros de mascaramento. As linhas pontilhadas representam os filtros interpolado e interpolado complementar, e as linhas contínuas representam os filtros de (a) máscara positiva e (b) máscara negativa. . . . .	111
A.3	Resposta de cada ramo dos filtros (linhas tracejadas). A linha contínua representa a resposta do filtro final. Note que no começo da faixa 0.61 a 0.73, o <i>ripple</i> dos ramos nos dois gráficos é alto, necessitando de maior margem. O mesmo acontece no início da faixa de 0.5111 a 0.6, no <i>ripple</i> da faixa de passagem do filtro final. . . . .	112
B.1	Sinais nas várias componentes polifásicas para o exemplo de filtro CMFB. . . . .	117
B.2	Sinais nas matrizes moduladoras para o exemplo do CMFB. . . . .	117
B.3	Sinais na estrutura CMFRM equivalente. . . . .	119
B.4	Sinais preparados para a entrada das matrizes pré-moduladoras na estrutura CMFRM equivalente. . . . .	119
B.5	Sinais nas várias componentes polifásicas para o exemplo de filtro CMFB. . . . .	121
B.6	Sinais nas matrizes moduladoras para o exemplo do CMFB. . . . .	122

B.7	Sinais na estrutura CMFRM simplificada. . . . .	123
B.8	Sinais nas matrizes moduladoras da CMFRM simplificada. . . . .	123
C.1	Resultados obtidos com o filtro CMFRM de reconstrução perfeita utilizando $M = L = 2$ , $K = 2$ e uma única variável a ser otimizada. .	127
C.2	Resultados obtidos com o filtro CMFRM de reconstrução perfeita utilizando $M = L = 2$ , $K = 3$ e uma única variável a ser otimizada. .	131

# Lista de Tabelas

2.1	Desvios máximos obtidos para o projeto do filtro FIR minimax. . . . .	12
2.2	Linhas de redução para o projeto do filtro FRM com bandas não críticas, para $5 \leq L \leq 10$ . A melhor redução ocorre para $L = 7$ . . . . .	13
2.3	Freqüências de corte dos filtros componentes do FRM. . . . .	14
2.4	Freqüências de limite das bandas críticas dos filtros de máscara. Neste exemplo, as bandas especialmente críticas dos dois filtros de máscara ocorrem no terceiro intervalo. . . . .	14
2.5	Desvios máximos obtidos para o exemplo de projeto FRM com bandas não críticas. . . . .	14
3.1	Algoritmo WLS-Chebyshev. . . . .	29
3.2	Desvios obtidos para os filtros de ordem $N = 46$ projetados pelos métodos minimax, mínimos quadrados e WLS-Chebyshev. . . . .	31
4.1	Algoritmo para os filtros de máscara independentes. . . . .	40
4.2	Algoritmo WLS-Chebyshev com tomada de erro na resposta final . . . . .	42
4.3	Algoritmo WLS-Chebyshev para o filtro base. . . . .	45
4.4	Linha ótima de redução para o exemplo 1. . . . .	47
4.5	Desvios máximos obtidos para o exemplo 1. . . . .	47
4.6	Linha ótima de redução para o exemplo 2. . . . .	48
4.7	Desvios máximos obtidos para o exemplo 2. . . . .	48
4.8	Linha ótima de redução para o exemplo 3. . . . .	49
4.9	Desvios máximos obtidos para o exemplo 3. . . . .	49
6.1	Tabela com os melhores casos de redução para o exemplo 1. . . . .	91
6.2	Tabela com os melhores casos de redução para o exemplo 2. . . . .	95
6.3	Tabela com os melhores casos de redução para o exemplo 3. . . . .	98



# Capítulo 1

## Introdução

Atualmente, a miniaturização de circuitos pode ser vista como o resultado de uma tecnologia que combina o projeto com a implementação física de um circuito. Por um lado, temos a tecnologia de integração em pastilhas de silício, que nos permite montar micro-circuitos com trilhas de precisão abaixo de  $1.2 \mu\text{m}$  para os materiais semicondutores (polisilício, largura das junções etc), resultando em circuitos de baixa potência, com uma miniaturização eficiente. Por outro lado, esta eficiência na miniaturização depende também da otimização no desenvolvimento do circuito em si. Assim como na fase do planejamento da confecção do circuito é necessário otimizar a topologia física deste, também é comum, em uma das fases do projeto, levantar um estudo para redução do circuito visando facilitar e reduzir a implementação física do mesmo.

Particularmente, filtros reduzidos que necessitam de menor complexidade computacional (número de operações matemáticas por amostra de sinal) têm sido amplamente estudados na área de processamento de sinais, pois implicam em redução do espaço necessário para a sua implementação em uma pastilha de silício. No caso do projeto de filtros por circuitos programáveis (DSP's), a redução de complexidade computacional é interessante no sentido da diminuição de recursos gastos para realizar um filtro digital. Assim, será possível realizar este filtro em um DSP com menos recursos, ou reaproveitar os recursos não utilizados em outra aplicação dentro do mesmo DSP.

O projeto de filtros reduzidos que utiliza a técnica de mascaramento de resposta em frequência (filtros FRM – *frequency-response masking*) foi inicialmente

proposto e estudado por Y. C. Lim [1], seguido desde então por vários outros trabalhos [2]-[7]. Nestes estudos, verifica-se que o fator de redução de um filtro de banda passante larga e banda de transição abrupta chega a cerca de 35% em relação a uma implementação direta (ou seja, o número de coeficientes, e, portanto, de multiplicações por amostra de sinal, é 35% do número de coeficientes em uma realização direta). No caso de filtros de banda passante estreita, este valor pode chegar a 15%. O fator de redução depende tanto do projeto, de onde obtêm-se os parâmetros ótimos para a topologia de mascaramento (tais como fator de interpolação e ordens dos filtros componentes), como também depende da utilização eficiente das diversas faixas de frequência, na composição do filtro FRM.

De fato, na primeira parte deste trabalho, abordaremos o método de otimização WLS-Chebyshev aplicado ao FRM. O WLS-Chebyshev é um método de otimização linear que permite uma maior flexibilidade na otimização dos filtros componentes, permitindo a redução destes filtros para cerca de 30%, nos casos de banda passante larga, e cerca de 12% no caso de banda estreita. Partindo de um algoritmo simples, este método permite variar a otimização do filtro, de forma a manter a energia constante (tipo minimax) em uma parte da banda de rejeição, e em outra parte reduzir esta energia, resultando em um filtro com características minimax e *least-squares*. A facilidade de operar sobre o algoritmo de otimização permite que a função-objetivo a minimizar possa ser escolhida de modo a otimizar diretamente a resposta final do filtro FRM nas bandas críticas deste tipo de projeto, permitindo assim a redução já mencionada.

Na segunda parte deste trabalho, abordaremos a aplicação do filtro FRM em uma estrutura eficiente de banco de filtros modulados por cosseno (CMFB – *cosine-modulated filter bank*) [13]-[18]. Na verdade, na estrutura FRM, os filtros de máscara efetivamente separam certas faixas de frequência, lembrando operações realizadas em um banco de filtros, para a separação de canais. Analisando mais profundamente o comportamento de filtros FRM, é possível obter, para certos fatores de interpolação, o filtro FRM necessário para formar o filtro protótipo de um CMFB, obtendo assim uma estrutura eficiente.

Este trabalho está organizado da seguinte forma: No Capítulo 2, estudaremos alguns pontos importantes sobre o projeto de filtros FRM; No Capítulo 3,

estudaremos o algoritmo WLS-Chebyshev, e no Capítulo 4, mostraremos como é feita a otimização dos filtros FRM utilizando o algoritmo WLS-Chebyshev. Na segunda parte, teremos o Capítulo 5, onde estudaremos os conceitos dos CMFB's. No Capítulo 6, analisando o comportamento do filtro FRM, estudaremos os casos em que é possível derivar uma estrutura eficiente para a realização dos filtros combinados FRM-CMFB's. Por fim, concluiremos a tese ressaltando os seus resultados e possíveis extensões do trabalho aqui realizado.

# Capítulo 2

## Método de Mascaramento da Resposta em Frequência

### 2.1 Introdução

O projeto de filtros que utiliza o método de mascaramento da resposta em frequência (FRM – *frequency-response masking*) visa evitar a realização direta da faixa de transição abrupta que ocorre em filtros de ordens altas [1]-[2],[5]-[6]. O método FRM se baseia no projeto de um filtro de transição suave (denominado filtro base) seguido de um processo de interpolação, fazendo com que as bandas do filtro base sejam comprimidas e repetidas no espectro de frequência. Este filtro, e outro filtro que realiza a resposta em frequência complementar, podem então ser mascarados por outros filtros que terão também faixas de transição mais amplas, e selecionarão bandas do filtro interpolado para formar a resposta em frequência final desejada. Porém como não é necessário implementar diretamente a transição abrupta, os filtros componentes terão ordens bem mais baixas que a de um projeto FIR na estrutura direta. O que se perde com um projeto destes, entretanto, é um pequeno aumento do atraso na resposta do filtro digital, já que em uma etapa de interpolação, cada atraso no filtro componente interpolado resulta em  $L$  atrasos no filtro final, onde  $L$  representa o fator de interpolação.

Na seção 2.2, introduziremos o método FRM, indicando a construção da resposta em frequência dos filtros componentes. Na seção 2.3, exemplificaremos a redução obtida, comparando o projeto direto com o projeto FRM. Na seção 2.4

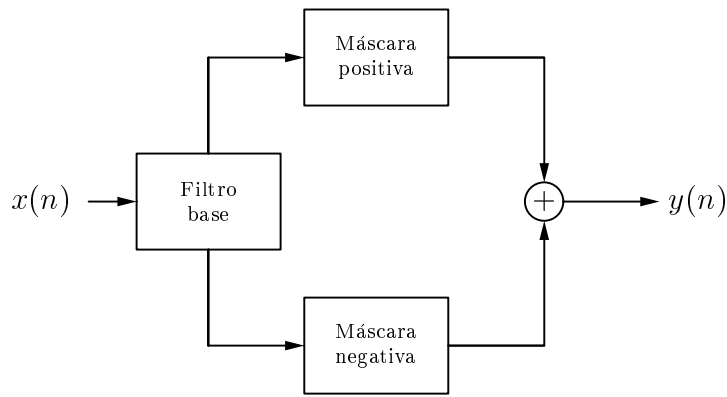


Figura 2.1: A estrutura geral de um filtro FRM.

comentaremos alguns pontos importantes a respeito dos desvios de ganho neste tipo de projeto, e na seção 2.5 verificamos que existem bandas especialmente críticas, que devem ser levadas em consideração no projeto de filtros FIR.

## 2.2 Estrutura e construção da resposta do filtro FRM

A estrutura simplificada de um filtro FRM pode ser vista na Figura 2.1, que é o caso mais simples de um filtro deste tipo. O filtro FRM é composto por um filtro na entrada que é interpolado, e possui duas saídas que são aplicadas a filtros chamados de mascaramento, sendo então somadas para produzir a saída final do filtro FRM. A função de transferência da entrada para a saída negativa (inferior) do filtro interpolado é complementar, em magnitude, da função de transferência da entrada para a saída positiva (superior) deste filtro.

O filtro interpolado, por sua vez, é obtido a partir de um filtro de base que é um filtro FIR realizado numa das formas diretas. Portanto, se o filtro de base possui a seguinte função de transferência

$$H_b(z) = \sum_{n=0}^{N_b} h_b(n)z^{-n} \quad (2.1)$$

então a função de transferência não complementar do filtro interpolado será

$$H_b^I(z) = \sum_{n=0}^{N_b} h_b(n)z^{-nL} \quad (2.2)$$

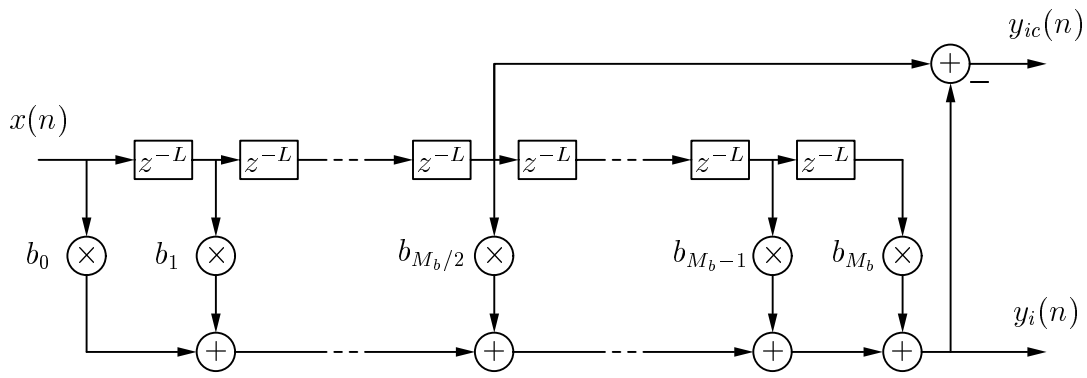


Figura 2.2: Obtenção das saídas do filtro interpolado.

onde  $N_b$  é a ordem do filtro de base,  $h_b(n)$  são os seus coeficientes e  $L$  é o fator de interpolação. A função complementar do filtro de entrada é dada pela relação

$$|H_{bc}^I(z)| = 1 - |H_b^I(z)| \quad (2.3)$$

Se o filtro de base possui resposta impulsiva simétrica (isto é, o filtro de base é projetado com fase linear) e com um coeficiente central, a função de transferência pode ser facilmente obtida a partir da resposta não complementar

$$H_{bic}(z) = 2b_{N_b/2}z^{-\frac{N_b L}{2}} - \sum_{n=0}^{N_b} b_n z^{-nL} \quad (2.4)$$

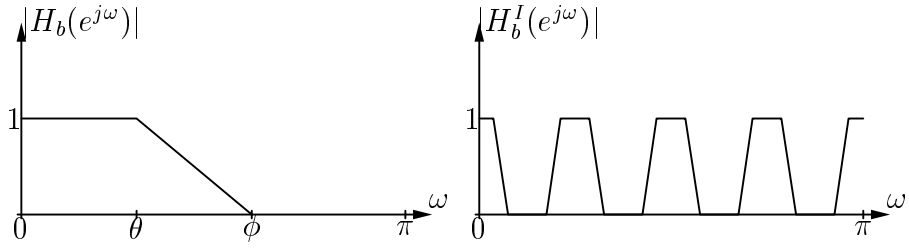
A Figura 2.2 mostra como obter as respostas não-complementar,  $y_i(n)$ , e complementar,  $y_{ic}(n)$ , a partir do filtro de base interpolado.

Em seguida, os filtros de máscara operam sobre o sinal do filtro interpolado, de modo a formar uma resposta final que seja a desejada. Estas operações podem ser vistas nas Figuras 2.3 e 2.4, sendo que se a última banda do filtro interpolado não-complementar deve ser mantida, teremos o caso da Figura 2.3, e se a última banda a ser mantida for gerada pelo filtro complementar, teremos o caso da Figura 2.4. <sup>1</sup>

Dependendo das freqüências de corte especificadas para o filtro final e das especificações para o *ripple* máximo de banda passante e de atenuação mínima na

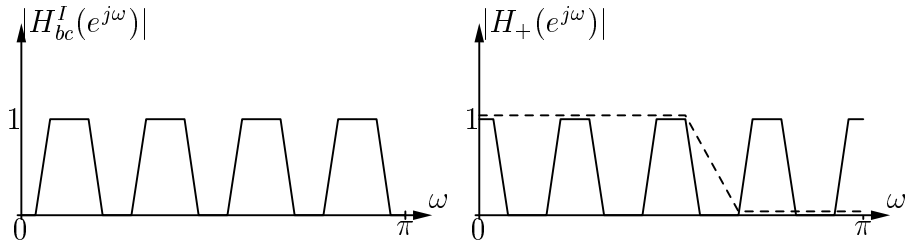
---

<sup>1</sup>Cabe lembrar que há topologias diversas para a realização do FRM, como a redução de coeficientes em banda estreita [3], o cascadeamento de filtros FRM [4], fatoração dos filtros de máscara [7] ou a utilização de filtros de máscara interpolados, que apresentam reduções de complexidade melhores para alguns casos particulares. Para o caso mais geral, entretanto, o filtro mais simplificado da Figura 2.1 continua sendo utilizado como referência, e portanto as reduções apresentadas neste capítulo são válidas.



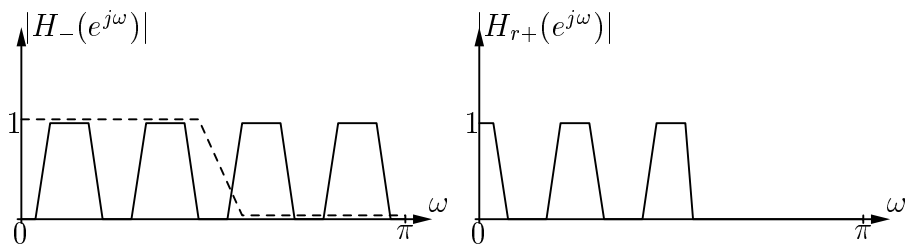
(a) Filtro base.

(b) Filtro interpolado.



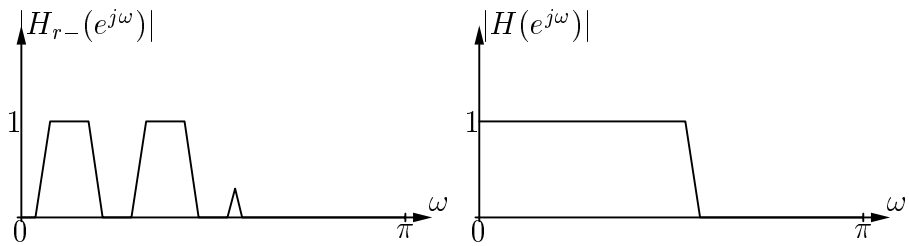
(c) Filtro interpolado complementar.

(d) Operação de máscara positiva.



(e) Operação de máscara negativa.

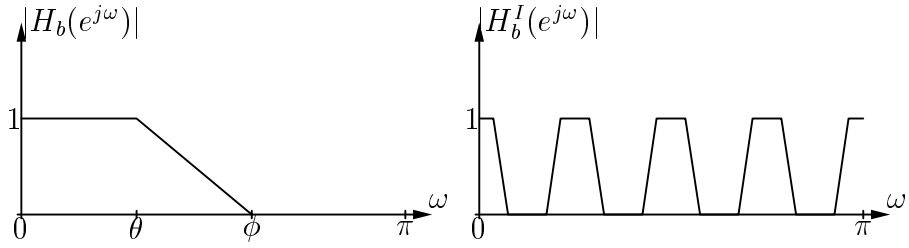
(f) Resultado da máscara positiva.



(g) Resultado da máscara negativa.

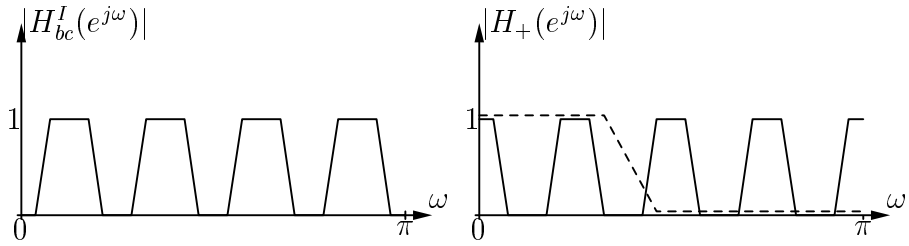
(h) Resultado final.

Figura 2.3: Processo de construção para o filtro FRM, com banda de mascaramento positivo no final da banda passante.



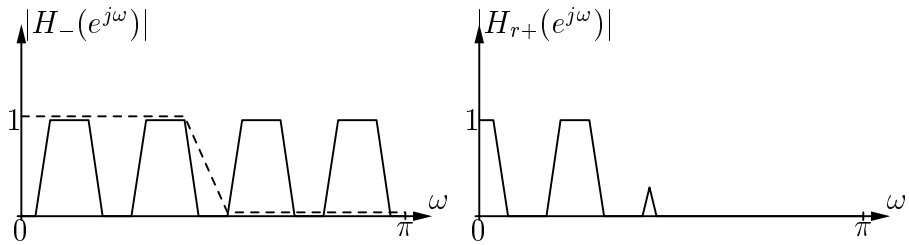
(a) Filtro base.

(b) Filtro interpolado.



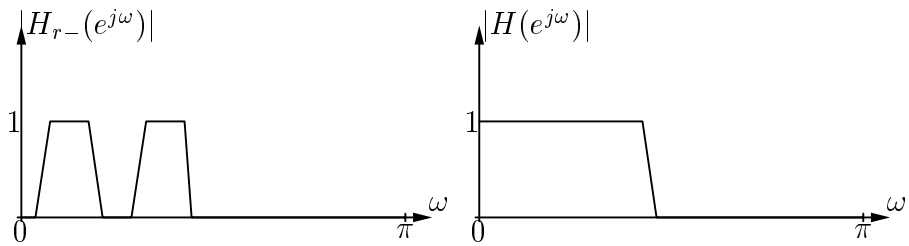
(c) Filtro interpolado complementar.

(d) Operação de máscara positiva.



(e) Operação de máscara negativa.

(f) Resultado da máscara positiva.



(g) Resultado da máscara negativa.

(h) Resultado final.

Figura 2.4: Processo de construção para o filtro FRM, com a banda de mascaramento negativo no final da banda passante.



banda de rejeição, é possível obter, para vários fatores de interpolação, estimativas das ordens dos filtros componentes (o filtro de base e os de máscara). Com isto, montamos uma tabela onde podemos escolher um fator de interpolação ótimo, isto é, o que leva às menores ordens para filtros componentes. Portanto, uma vez especificadas as frequências de corte  $\omega_p$  e  $\omega_s$  para o filtro desejado, e uma vez dado o fator de interpolação  $L$ , as frequências de corte do filtro base  $\theta$  e  $\phi$  podem ser calculadas a partir das equações abaixo [1]

$$m = \left\lfloor \frac{\omega_p L}{2\pi} \right\rfloor \quad (2.5)$$

$$\theta = \omega_p L - 2m\pi \quad (2.6)$$

$$\phi = \omega_s L - 2m\pi \quad (2.7)$$

ou

$$m = \left\lceil \frac{\omega_s L}{2\pi} \right\rceil \quad (2.8)$$

$$\theta = 2m\pi - \omega_s L \quad (2.9)$$

$$\phi = 2m\pi - \omega_p L \quad (2.10)$$

onde  $\lfloor x \rfloor$  indica o maior inteiro menor que  $x$ , e  $\lceil x \rceil$  indica o menor inteiro maior que  $x$ . Se as equações (2.5)-(2.7) satisfazem  $0 \leq \theta \leq \phi \leq \pi$ , o filtro de máscara positiva possui frequências de corte acima do filtro de máscara negativa e o caso da Figura 2.3 se aplica. Caso contrário, isto é, se as equações (2.8)-(2.10) satisfazem  $0 \leq \theta \leq \phi \leq \pi$ , o filtro de máscara negativa possui frequências de corte acima do filtro de máscara positiva, e o caso da Figura 2.4 passa a ocorrer. Note que ambos os conjuntos de equações não podem satisfazer simultaneamente à restrição de  $0 \leq \theta \leq \phi \leq \pi$ . Com os valores calculados acima, podemos calcular as frequências de corte dos filtros de máscara positiva e de máscara negativa. No primeiro caso, em que as equações (2.5)-(2.7) são válidas, teremos

$$\omega_{+,p} = \frac{2m\pi + \theta}{L} \quad (2.11)$$

$$\omega_{+,s} = \frac{2(m+1)\pi - \phi}{L} \quad (2.12)$$

$$\omega_{-,p} = \frac{2m\pi - \theta}{L} \quad (2.13)$$

$$\omega_{-,s} = \frac{2m\pi + \phi}{L} \quad (2.14)$$

e, no segundo caso, isto é, se as equações (2.8)-(2.10) são válidas, teremos

$$\omega_{+,p} = \frac{2(m-1)\pi - \phi}{L} \quad (2.15)$$

$$\omega_{+,s} = \frac{2m\pi - \theta}{L} \quad (2.16)$$

$$\omega_{-,p} = \frac{2m\pi - \phi}{L} \quad (2.17)$$

$$\omega_{-,s} = \frac{2m\pi + \theta}{L} \quad (2.18)$$

onde  $m$  é o valor calculado na equação (2.5) ou na equação (2.8).

No caso de se utilizar bandas não-críticas (“*don't care*”), é necessário obter os limites de frequência de cada banda crítica (“*care*”) devidos à repetição do espectro de frequência do filtro de base. Para o filtro de máscara positiva, estes valores serão

$$\omega_{+,a} = \frac{2m'\pi - \phi}{L} \quad (2.19)$$

$$\omega_{+,b} = \frac{2m'\pi + \phi}{L} \quad (2.20)$$

$$(2.21)$$

onde  $m' = 0, 1, \dots$ . Para o filtro de máscara negativa, teremos como limites das bandas críticas

$$\omega_{-,a} = \frac{(2m'+1)\pi - \theta}{L} \quad (2.22)$$

$$\omega_{-,b} = \frac{(2m'+1)\pi + \theta}{L} \quad (2.23)$$

$$(2.24)$$

Cabe lembrar ainda que as bandas críticas dos filtros de máscara podem ter pesos diferentes entre si. Um dos esquemas que pode ser utilizado é supor que as bandas que não são excessivamente críticas podem ter o mesmo peso, já que não diferem significativamente umas das outras. A verificação da existência de bandas especialmente críticas será vista mais tarde neste capítulo. As bandas que sejam excessivamente críticas do projeto devem possuir um peso maior que as demais. Neste esquema, podemos, portanto, modificar as seguintes relações entre os pesos das bandas:

- $w[\omega_p]/w[\omega_s]$  – Define a relação do peso entre as bandas críticas da banda passante em relação às bandas críticas da banda de rejeição;

- $w[\omega_{pp}]/w[\omega_s]$  – Define a relação do peso entre a banda especialmente crítica da banda passante em relação às bandas críticas da banda de rejeição;
- $w[\omega_{ss}]/w[\omega_s]$  – Define a relação do peso entre a banda especialmente crítica da banda de rejeição em relação às bandas críticas da banda de rejeição.

## 2.3 Exemplo numérico

Para verificarmos a eficiência do filtro FRM sobre um filtro implementado na forma direta, e também para futuras comparações nos próximos capítulos, iremos agora delinear os projetos da forma direta e FRM.

Desejamos projetar um filtro digital que atenda às seguintes especificações:

- Banda passante:  $0 \leq \omega \leq 0.65\pi$
- Banda de rejeição:  $0.66\pi \leq \omega \leq \pi$
- *Ripple* na banda passante:  $2\delta_p = 0.2$  dB
- Atenuação mínima na banda de rejeição:  $\delta_s = 40$  dB

### 2.3.1 Projeto do filtro FIR minimax

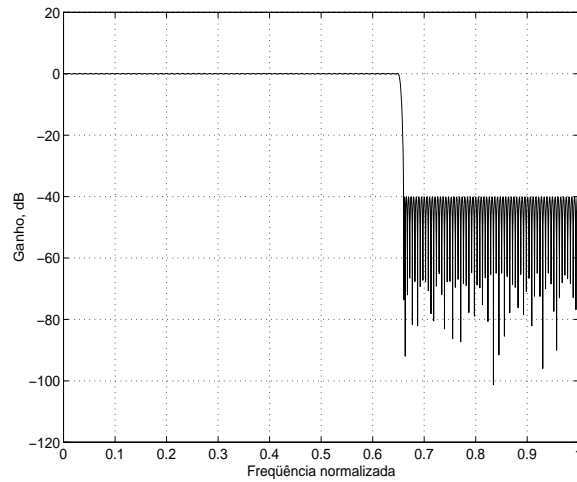
Projetando um filtro de resposta finita através do algoritmo minimax, a partir das especificações dadas para o filtro, através das funções `remez` e `remezord` do MATLAB, obteremos um filtro com ordem  $N = 381$  (ou seja, 382 coeficientes) e o resultado da Figura 2.5. Se utilizarmos o fato de que este filtro possui simetria, então o número de multiplicações necessárias se reduz para a metade, isto é, 191 multiplicações<sup>2</sup>. Uma vez que o filtro possui característica de banda de transição estreita, o número de coeficientes se torna alto, tornando interessante um projeto de filtro FRM, visando a redução de complexidade (número de coeficientes) do filtro. Para este projeto FIR minimax, obteremos os desvios máximos da sua resposta em frequência apresentados na Tabela 2.1.

---

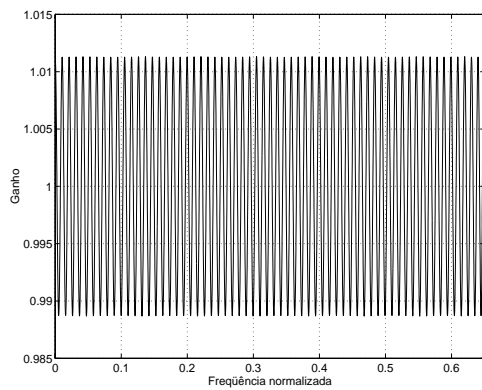
<sup>2</sup>Se o filtro é simétrico, os mesmos elementos que multiplicam o sinal pela primeira metade da resposta impulsiva do filtro servem também para multiplicar a segunda metade da resposta impulsiva.

Tabela 2.1: Desvios máximos obtidos para o projeto do filtro FIR minimax.

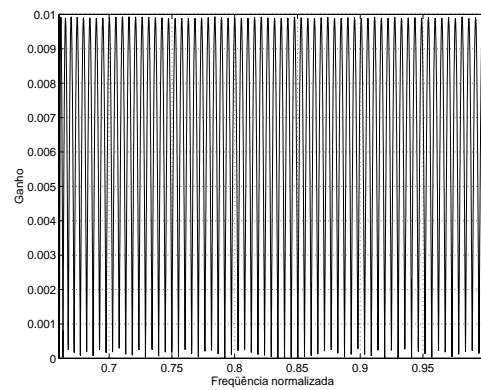
Filtro	$2\delta_{p,max}$	$\delta_{s,max}$
FIR minimax	0.198 dB	-40.06 dB



(a) Resposta em magnitude do filtro FIR minimax.



(b) Banda passante.



(c) Banda de rejeição.

Figura 2.5: Projeto convencional de um filtro FIR passa-baixas a partir do algoritmo minimax.

Tabela 2.2: Linhas de redução para o projeto do filtro FRM com bandas não críticas, para  $5 \leq L \leq 10$ . A melhor redução ocorre para  $L = 7$ .

$L$	$N_b$	$N_+$	$N_-$	$M_{Tot}$	Redução (%)
5	90	40	16	149	39.01
6	74	14	184	275	71.99
<b>7</b>	<b>64</b>	<b>38</b>	<b>28</b>	<b>133</b>	<b>34.8</b>
8	56	73	25	157	41.10
9	50	24	184	261	68.32
10	46	49	41	139	36.39

### 2.3.2 Projeto do filtro FRM utilizando o algoritmo minimax - projeto com bandas não críticas.

Utilizando o método de redução FRM visto em [1], com bandas não críticas, e utilizando uma margem de desvio de ganho nos filtros componentes de cerca de 32% com o seguinte esquema de equalização das bandas dos filtros de máscara

$$w[\omega_p] = 1.2w[\omega_s]; \quad w[\omega_{pp}] = 1.9w[\omega_s]; \quad w[\omega_{ss}] = 1.7w[\omega_s]$$

que resultam experimentalmente em uma boa equalização entre as bandas críticas do projeto, e com isto obteremos as linhas de redução apresentadas na Tabela 2.2. Esta tabela é obtida calculando-se as frequências de corte dos filtros componentes para os vários fatores de interpolação, e estimando as ordens destes filtros de acordo com a margem e a pesagem das bandas críticas. Como podemos notar, para o valor de  $L = 7$  obteremos filtros de base e de máscara cuja complexidade computacional, quando totalizada, resulta no melhor caso, isto é, em 133 multiplicações para gerar uma amostra na saída do filtro FRM. Portanto, este filtro possuirá 34.8% do número de multiplicações necessárias para realizar o mesmo projeto FIR minimax, isto é,  $\frac{133}{382} = 0.348$ .

Para este projeto FRM, obteremos, utilizando as equações (2.5)-(2.7) e (2.11)-(2.14) as frequências de corte apresentadas na Tabela 2.3. Na Tabela 2.4, temos as frequências das bandas críticas dadas pelas equações (2.19)-(2.23), enquanto que as respostas em magnitude estão apresentadas nas Figuras 2.6, 2.7, 2.8 e 2.9, e os valores

Tabela 2.3: Freqüências de corte dos filtros componentes do FRM.

$\theta$	$\phi$	$\omega_{p,+}$	$\omega_{s,+}$	$\omega_{p,-}$	$\omega_{s,-}$
$0.55\pi$	$0.62\pi$	$0.65\pi$	$0.7686\pi$	$0.4929\pi$	$0.66\pi$

Tabela 2.4: Freqüências de limite das bandas críticas dos filtros de máscara. Neste exemplo, as bandas especialmente críticas dos dois filtros de máscara ocorrem no terceiro intervalo.

Filtro	intervalo 1	intervalo 2	<b>intervalo 3</b>	intervalo 4
Máscara positiva	0 a $0.0886\pi$	$0.1971\pi$ a $0.3743\pi$	<b><math>0.4829\pi</math> a <math>0.65\pi</math></b>	$0.7686\pi$ a $0.9457\pi$
Máscara negativa	$0.0786\pi$ a $0.2071\pi$	$0.3643\pi$ a $0.4929\pi$	<b><math>0.66\pi</math> a <math>0.7786\pi</math></b>	$0.9357\pi$ a $\pi$

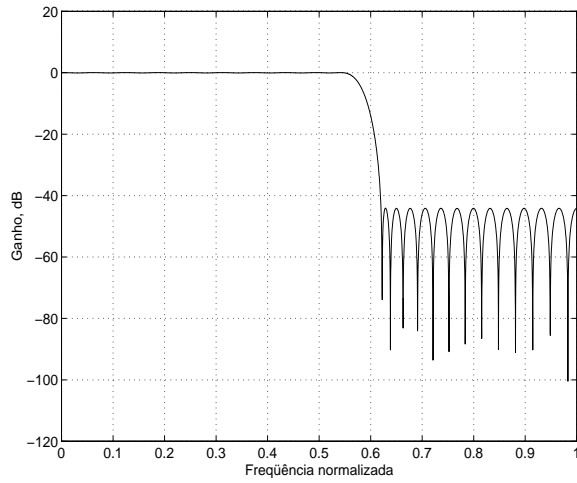
de desvios máximos dos filtros componentes estão apresentados na Tabela 2.5.

## 2.4 Margens de desvio de ganho

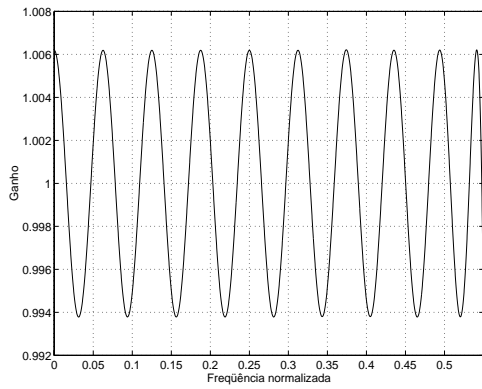
Uma questão importante no projeto de filtros FRM é em relação às margens de desvio de ganho. Nos projetos FRM, estamos trabalhando com a composição de vários filtros, que quando somados em um esquema paralelo e/ou série, resultam em um filtro de complexidade reduzida, em relação a um projeto de um único filtro FIR.

Tabela 2.5: Desvios máximos obtidos para o exemplo de projeto FRM com bandas não críticas.

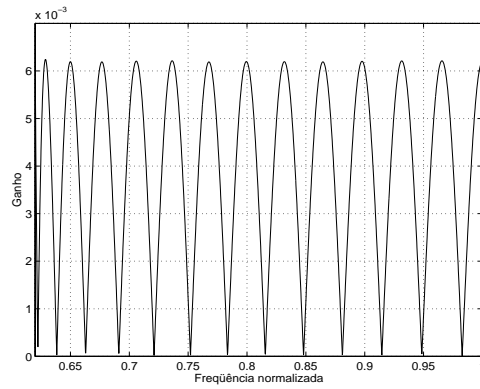
Filtro	$2\delta_{p,max}$	$\delta_{s,max}$
Base	0.108 dB	-43.29 dB
Máscara positiva	1.007 dB	-27.52 dB
Máscara negativa	0.934 dB	-21.71 dB
Filtro final	0.156 dB	-40.65 dB



(a) Resposta em magnitude do filtro.

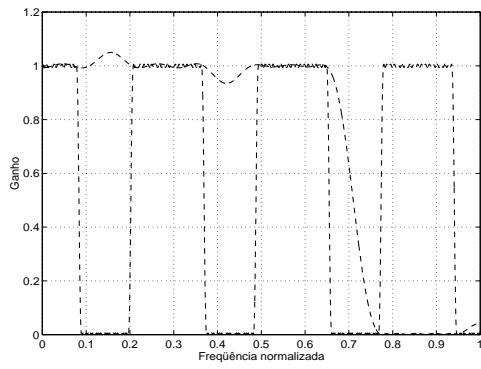


(b) Resposta na banda passante.

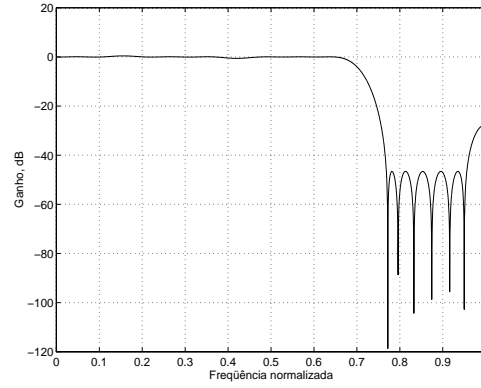


(c) Resposta na banda de rejeição.

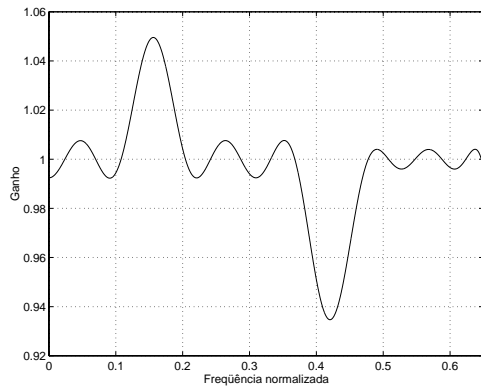
Figura 2.6: Filtro de base para o projeto FRM com bandas não críticas.



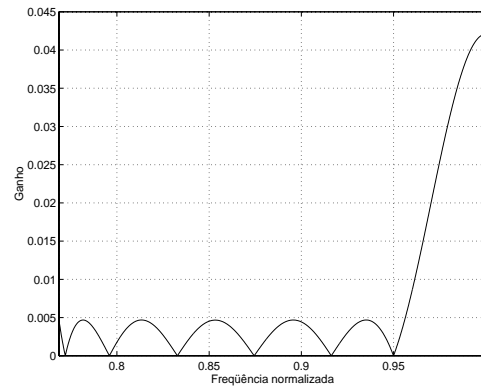
(a) Operação de máscara positiva.



(b) Resposta em magnitude do filtro de máscara positiva.



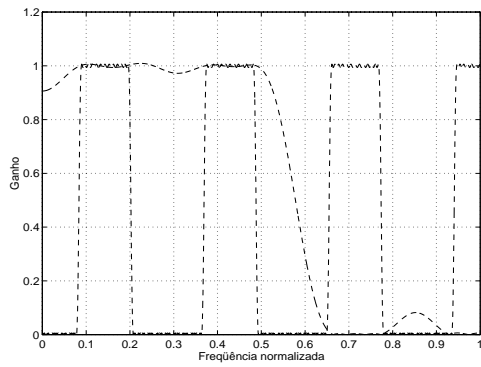
(c) Resposta na banda passante.



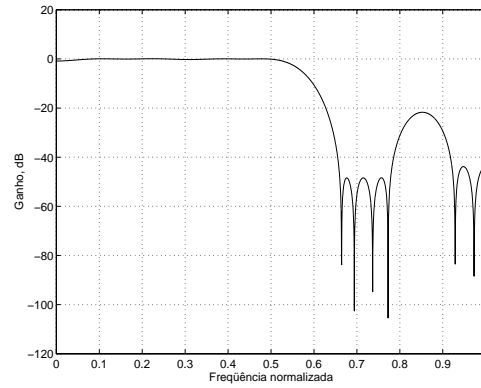
(d) Resposta na banda de rejeição.

Figura 2.7: Funcionamento da máscara positiva para o projeto do filtro FRM com bandas não críticas.

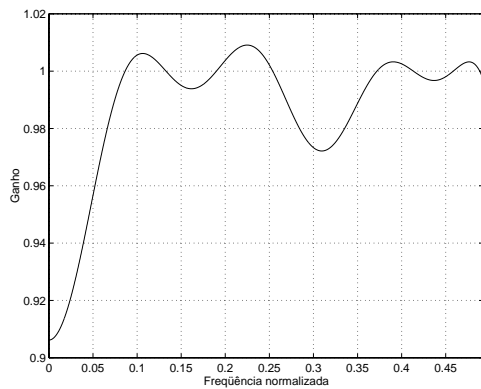




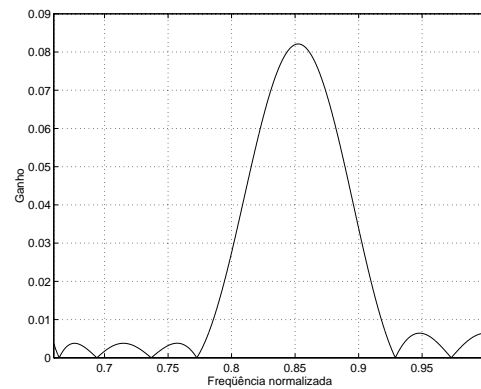
(a) Operação de máscara negativa.



(b) Resposta em magnitude do filtro de máscara negativa.

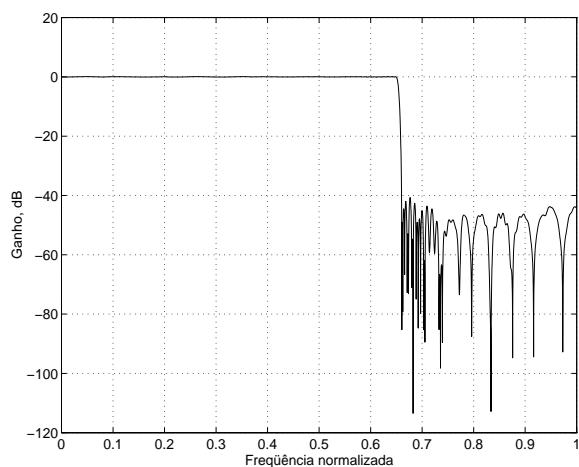


(c) Resposta na banda passante.

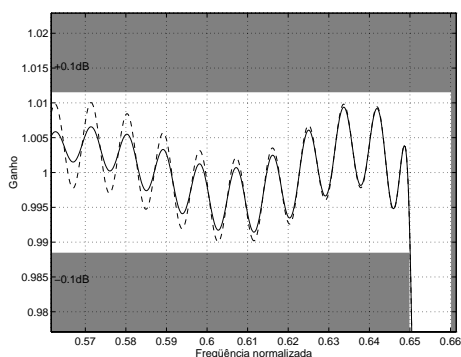


(d) Resposta na banda de rejeição.

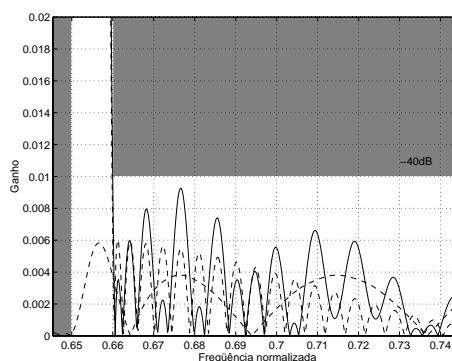
Figura 2.8: Funcionamento da máscara negativa para o projeto do filtro FRM com bandas não críticas.



(a) Resposta em magnitude do filtro resultante.



(b) Detalhe do final da banda passante: magnitudes da resposta do filtro resultante (linha sólida) e da saída do ramo positivo (linha tracejada).



(c) Detalhe do início da banda de rejeição: magnitudes da resposta do filtro resultante (linha sólida) e das saídas dos dois ramos do filtro (linhas tracejadas), sendo a mais suave devida ao ramo negativo.

Figura 2.9: Resposta resultante em magnitude do filtro FRM com bandas não críticas.

Se desejamos que o filtro resultante obedeça a uma determinada especificação de *ripple* e atenuação (isto é, desvios máximos permitidos no ganho das bandas passante e de rejeição, respectivamente), cada um dos filtros componentes deverá possuir especificações de desvios próprias, em geral mais apertadas do que as do filtro FRM final, para que as respostas, quando somadas, não excedam os desvios inicialmente especificados. Para facilitar o projeto dos filtros componentes, é utilizado o conceito de margem de desvio de ganho (ou simplesmente, margem de ganho) em vez de serem dadas especificações individuais para cada filtro [1]. As margens de desvio de ganho têm relação íntima com a ordem (e, portanto com a complexidade) dos filtros componentes, e também têm relação com o comportamento das bandas da resposta em frequência do filtro FRM.

Quando mencionamos margem de desvio de ganho, estamos nos referindo ao desvio máximo permitido nos filtros componentes, em relação à especificação do filtro FRM desejado. Assim, se deixamos uma margem de 20%, estamos querendo que os desvios de cada filtro componente cheguem a, no máximo, 80% do desvio especificado para o filtro FRM final. Se deixamos uma margem de 50% em um projeto cujo filtro final deva apresentar 40 dB de atenuação mínima, os filtros componentes devem ser projetados para  $(1-50\%) \times 40 \text{ dB} = 0.5 \times 40 \text{ dB} = 46 \text{ dB}$  de atenuação. Para facilitar o projeto de filtros FRM, é comum utilizar a mesma margem para todos os filtros componentes, particularmente se as ordens dos filtros são próximas e se o algoritmo do projeto é o mesmo (por exemplo, todos os filtros componentes sendo projetados pelo algoritmo minimax). Se o filtro FRM possui, portanto, bandas críticas, então a mesma pesagem poderia, a princípio, ser utilizada em todas as bandas críticas.

A ordem de um filtro FIR é determinada, basicamente, pela largura da faixa de transição e pelo desvio máximo desejado para o filtro. Quanto mais estreita for a faixa de transição, maior será a ordem do filtro para que esta faixa possa ser realizável, e quanto menores forem os desvios máximos desejados em cada frequência, também maiores serão a ordem e o número de coeficientes do filtro. Estas duas qualidades do filtro podem ser permutadas, isto é, uma faixa de transição menor pode ser conseguida aumentando os desvios máximos permitidos, para um mesmo número de coeficientes de um filtro FIR. Entretanto, uma vez determinadas as faixas de transição dos filtros componentes, que são determinadas exclusivamente pelas

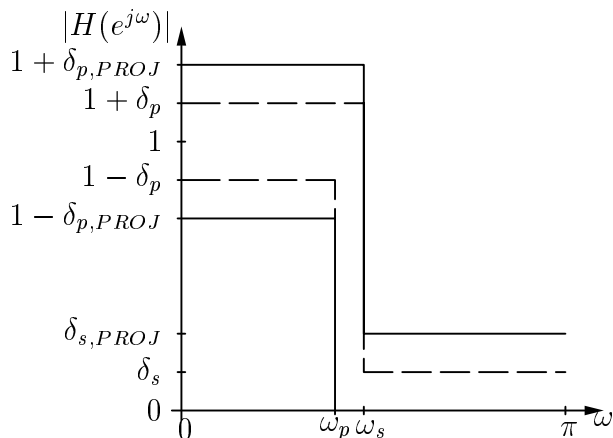


Figura 2.10: Especificação de margem para o projeto dos filtros componentes. O subíndice *PROJ* se refere aos valores especificados para o filtro final, enquanto que  $\delta_p$  e  $\delta_s$  são os desvios permitidos a um dos filtros componentes. Se a margem for igual a 0%, então  $\delta_p = \delta_{p,PROJ}$  e  $\delta_s = \delta_{s,PROJ}$ .

especificações de frequência do filtro desejado e pelo fator de interpolação  $L$ , o aumento ou diminuição da margem de ganho resulta no aumento ou diminuição da ordem, e portanto, da complexidade de um filtro particular.

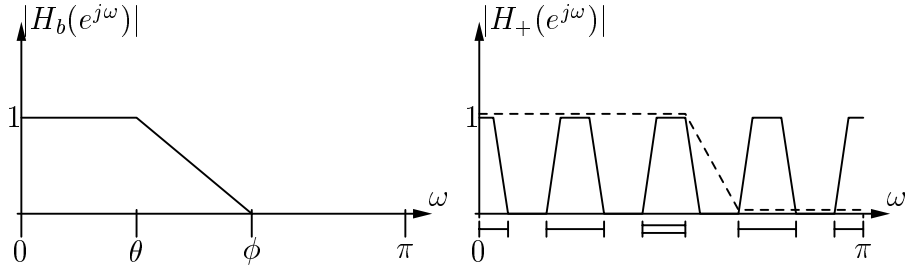
Portanto, alterar a margem de desvio de ganho em um dos filtros significa modificar a ordem deste filtro. No caso de um projeto minimax, em que os filtros componentes apresentem aproximadamente a mesma ordem, podemos inicialmente estimar a ordem dos filtros com um algoritmo de estimação, supondo determinadas margens para os filtros, e depois realizarmos os ajustes de margens, que poderiam ser vistos como o ajuste fino na ordem (e, portanto, na complexidade) de cada um dos filtros.

## 2.5 Bandas especialmente críticas

No projeto do filtro FRM, verificamos que as especificações dos desvios máximos de ganho para os filtros de máscara podem ser realizadas em cada banda de repetição do filtro interpolado, separadamente. Ou seja, no caso do filtro de máscara positiva, cada faixa de frequência correspondente à repetição da banda passante do filtro interpolado pode ser pesada de uma maneira diferente, e no filtro de máscara negativa, cada faixa de frequência correspondente à repetição da banda de rejeição do filtro interpolado pode ser pesada de uma maneira diferente das demais. Esta é

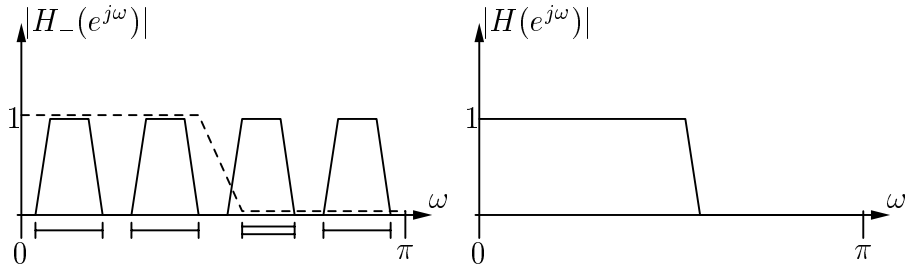
uma característica importante, pois faz com que no algoritmo de otimização seja possível reduzir-se o peso de algumas bandas, reduzindo assim a ordem dos filtros componentes. Pode-se verificar isto observando a resposta final do filtro FRM, ou realizando uma análise formal para as várias faixas de frequência. Esta análise está descrita no Apêndice A desta tese. Por esta análise, verificamos que existem duas faixas de frequências que são especialmente críticas em termos do desvio de ganho. Veja as figuras 2.9b e 2.9c, que mostram a dependência do desvio de ganho da resposta final do filtro FRM devido aos desvios do filtro de máscara e do filtro de base no final da banda passante e no início da banda de rejeição da resposta final. Note que nas outras bandas a resposta é mais suave, pois os desvios do filtro interpolado e de seu complementar praticamente se anulam. Entretanto, nestas duas bandas especialmente críticas, os desvios dos filtros interpolado e de seu complementar não se cancelam, e necessitaríamos de margens de cerca de 50% no pior caso, segundo as análises, para que o filtro final atendesse às especificações desejadas. Portanto, para cada filtro de máscara, temos três tipos de bandas: a banda não crítica, onde é possível relaxar os pesos do algoritmo de otimização; a banda crítica, cuja especificação de desvio de ganho deve estar próxima da especificação inicial para o filtro FRM; e uma banda especialmente crítica, onde o peso deve ser aumentado, fazendo com que o filtro atenda à especificação nesta faixa. Na Figura 2.11 podemos notar estas faixas de frequência. Na proximidade da faixa de transição da resposta do filtro resultante, a resposta de um dos filtros de máscara predomina sobre o outro, ou seja, no final da banda passante, a resposta do filtro de máscara positiva predomina, enquanto que no início da banda de rejeição a resposta do filtro de máscara negativa predomina, ou vice-versa, dependendo da última banda do filtro interpolado a ser mantida, se ela é invertida ou não.

No projeto do filtro FRM, quanto maiores as margens, menores os desvios máximos permitidos nos filtros componentes, e, por isso, maior a complexidade destes filtros. Para filtros componentes com o mesmo tipo de projeto, e não sendo muito diferentes entre si, isto é, número de complexidade próximos, e bandas de transição relativamente iguais, pode-se utilizar a mesma margem para todos os filtros componentes, já que a redução maior ou menor da margem em um dos filtros não apresenta vantagens sobre os outros filtros. Em contrapartida, um filtro componente



(a) Filtro base.

(b) Operação de máscara positiva.



(c) Operação de máscara negativa.

(d) Resposta em frequência (magnitude) resultante.

Figura 2.11: Operações de mascaramento para a obtenção da resposta final, onde se notam as bandas não críticas (sem indicação sob o eixo das frequências), críticas (linhas simples) e as bandas especialmente críticas (linhas duplas).

que apresente muito mais coeficientes que os outros, pode ter a sua margem reduzida, aumentando as dos outros, o que deveria fazer com que mais coeficientes sejam liberados no filtro FRM.

## 2.6 Conclusões

Como vimos, os filtros FRM implicam em uma redução drástica da complexidade computacional da filtragem, quando comparados com a implementação na forma direta. A redução se torna mais acentuada à medida em que a ordem do filtro aumenta para se obter faixas de transição estreitas, e à medida em que se utiliza métodos de mascaramento mais sofisticados (utilizando, por exemplo, bandas não-críticas). Podemos, através da análise em frequência da composição da resposta do filtro FRM final, verificar que existem três tipos de bandas envolvidas no projeto dos filtros de máscara: bandas não-críticas, onde praticamente não é necessário otimizar o filtro de máscara, pois o outro filtro de máscara está dominando a resposta final; bandas críticas, onde o filtro de máscara em questão influi na resposta final, mas é necessário deixar apenas uma pequena margem devido a erros de segunda ordem; e bandas especialmente críticas, onde um dos filtros apresenta faixa de transição, e o outro filtro necessita compensá-lo, aumentando a sua margem de desvios. Realizando o projeto do filtro FRM levando em conta estes três fatores, será possível obter-se uma melhor redução na complexidade computacional do filtro.

# Capítulo 3

## Algoritmo WLS-Chebyshev

### 3.1 Introdução

O algoritmo WLS-Chebyshev é baseado na alteração da matriz de peso em um projeto de filtro de mínimos quadrados ponderados (WLS – *weighted least squares*) [8], de modo a restringir apenas alguns picos da resposta em frequência. Com isto, obtém-se um filtro que compromete as medidas de atenuação mínima e energia total em sua banda de rejeição. O algoritmo realiza, portanto, uma resposta em frequência que será mista, isto é, possuirá uma parcela minimax e uma parcela com decaimento de mínimos quadrados. Uma vez que é possível controlar os desvios em cada pico de amplitude do filtro, este tipo de algoritmo se torna versátil, e ao mesmo tempo fácil de ser implementado. Na seção 3.2 estudaremos o algoritmo WLS, utilizado para obter-se um filtro de características minimax iterativamente, a partir do filtro de mínimos quadrados. Na seção 3.3 descrevemos as alterações do algoritmo WLS de modo que se possa restringir apenas alguns dos picos de amplitude, resultando no algoritmo WLS-Chebyshev. Na seção 3.4, ilustraremos o algoritmo por meio de um projeto FIR, e na seção 3.5, abordaremos algumas vantagens deste algoritmo.

### 3.2 Algoritmo WLS

O algoritmo WLS [8] é utilizado para realizar uma otimização ponderada de um filtro, isto é, o critério de otimização leva em conta que os valores a minimizar



possuem pesos diferentes entre si. Supondo que um filtro FIR possua característica de fase linear, a sua resposta será simétrica e o filtro pode ser descrito como

$$H(\omega) = e^{j\theta} \sum_{n=1}^M b_n \text{trig}(\omega, n) \quad (3.1)$$

onde  $e^{j\theta}$  é a parte devida à fase linear,  $b_n$  são os coeficientes do filtro,  $\text{trig}(\omega, n)$  é uma função de senos ou cossenos, e  $M$  é o número de coeficientes do filtro. A função de erro entre a resposta do filtro e a resposta desejada  $\hat{H}(\omega)$  será, portanto

$$E(\omega) = \sum_{n=1}^M b_n \text{trig}(\omega, n) - \hat{H}(\omega) \quad (3.2)$$

se igualarmos o atraso de fase dos dois filtros. Se escolhermos uma grade de frequências  $\omega_0, \omega_1, \dots, \omega_n$  de zero a  $\pi$ , para o cálculo da função de erro, teremos a seguinte equação matricial

$$\mathbf{E} = \mathbf{R}(\mathbf{U}\mathbf{b} - \hat{\mathbf{H}}) \quad (3.3)$$

onde

$$\mathbf{E} = [E(\omega_1) \quad E(\omega_2) \quad \dots]^T \quad (3.4)$$

$$\mathbf{U} = \begin{bmatrix} \text{trig}(\omega_1, 1) & \text{trig}(\omega_1, 2) & \dots \\ \text{trig}(\omega_2, 1) & \text{trig}(\omega_2, 2) & \dots \\ \vdots & \vdots & \ddots \end{bmatrix} \quad (3.5)$$

$$\mathbf{b} = [b_1 \quad b_2 \quad \dots]^T \quad (3.6)$$

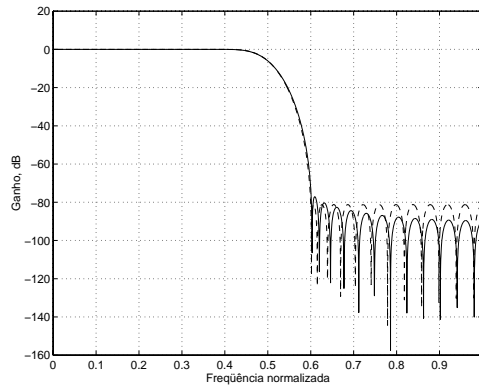
$$\hat{\mathbf{H}} = [\hat{H}(\omega_1) \quad \hat{H}(\omega_2) \quad \dots]^T \quad (3.7)$$

A solução para a equação acima será, portanto

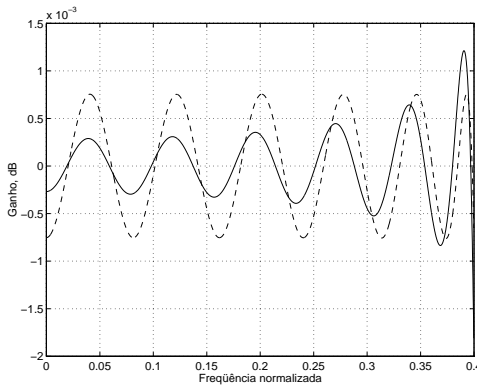
$$\mathbf{b} = (\mathbf{U}^T \mathbf{R} \mathbf{U})^{-1} \mathbf{U}^T \mathbf{R} \hat{\mathbf{H}} \quad (3.8)$$

onde  $\mathbf{R}$  será uma matriz diagonal de pesos na forma

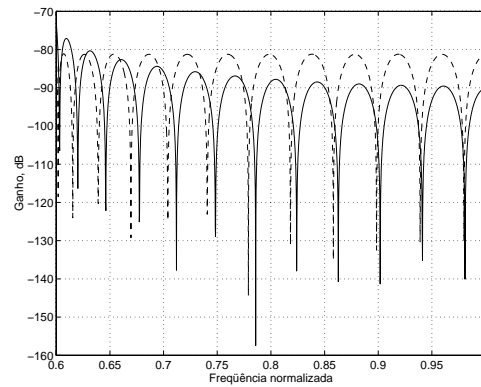
$$\mathbf{R} = \begin{bmatrix} r(\omega_1) & & 0 \\ & r(\omega_2) & \\ 0 & & \ddots \end{bmatrix} \quad (3.9)$$



(a) Resposta em magnitude dos filtros.



(b) Resposta na banda passante.



(c) Resposta na banda de rejeição.

Figura 3.1: Projetos minimax (linha tracejada) e mínimos quadrados (linha contínua), para um filtro de ordem 46 e frequências de corte de  $\omega_p = 0.4\pi$  e  $\omega_s = 0.6\pi$ .

se, para o projeto WLS, minimizarmos a função

$$\psi = \sum_n \left( r(\omega_n) E(\omega_n) \right)^2 \quad (3.10)$$

A solução WLS possui uma resposta em magnitude no formato apresentado na figura 3.1 (linha contínua). Em comparação com um projeto minimax (linha pontilhada), o filtro WLS possui a desvantagem de ter erros muito altos próximos da faixa de transição; em compensação, os picos mais afastados são bem atenuados, o que sugere que alguma compensação possa ser feita nas proximidades da faixa de transição para restringir os picos.

De fato, realizando o algoritmo WLS iterativamente, Lawson [9] pôde mostrar

que para uma dada função de pesos, podemos aproximar um filtro de característica minimax, utilizando iterativamente a seguinte equação

$$r_{k+1}(\omega) = r_k(\omega)\beta_k(\omega) \quad (3.11)$$

onde

$$\beta_k(\omega_n) = \frac{|E_k(\omega_n)|}{\sum_m r_k(\omega_m)|E_k(\omega_m)|} \quad (3.12)$$

Uma vez que o erro em uma dada frequência  $\omega_n$  pode ser próximo ou igual a zero (já que temos oscilações da resposta em frequência em torno da resposta ideal), o algoritmo apresenta problemas de convergência nestas frequências, já que  $r_{k+1}(\omega_m)$  tenderá a ser zero e o erro irá se propagar nas próximas iterações. Uma maneira de evitar que isto ocorra é utilizar a modificação do método proposta em [8], que denotamos aqui como algoritmo Lawson-LLCY (Lim, Lee, Chen e Yang), utilizando, em vez da função de erro  $|E_k(\omega_n)|$ , a sua envoltória em cada iteração, ou seja, obtemos todos os máximos locais da função  $E_k(\omega)$  nas bandas passante e de rejeição, e criamos uma função por partes interligando estes pontos de máximo. Matematicamente, podemos expressar os picos de máximos os que satisfazem a relação

$$\omega_{\mathbf{J}} = \left[ \omega_{J(1)} \quad \omega_{J(2)} \quad \dots \quad \omega_{J(n)} \right] = \{ \omega_i \mid (|E_k(\omega_i)| > |E_k(\omega_{i\pm 1})|) \} \quad (3.13)$$

e, portanto, o valor absoluto do erro nestas frequências de pico será dado por:

$$V_k(i) = |E_k(\omega_{J(i)})| \quad (3.14)$$

Criando a função de envoltória  $B_k(\omega)$ :

$$B_k(\omega) = \frac{\omega - \omega_{J(i)}}{\omega_{J(i+1)} - \omega_{J(i)}} V_k(i+1) + \frac{\omega_{J(i+1)} - \omega}{\omega_{J(i+1)} - \omega_{J(i)}} V_k(i), \quad \omega_{J(i)} < \omega < \omega_{J(i+1)} \quad (3.15)$$

Para calcularmos, agora, as funções de atualização  $\beta_k(\omega_n)$ , trocamos o termo  $|E_k(\omega_n)|$  pela sua envoltória,  $B_k(\omega_n)$ , na equação de Lawson:

$$\beta_k(\omega_n) = \frac{B_k(\omega_n)}{\sum_m r_k(\omega_m) B_k(\omega_m)} \quad (3.16)$$

### 3.3 O Algoritmo de restrição de picos WLS-Chebyshev

O algoritmo de Lawson-LLCY pode ser modificado para operar apenas sobre certos picos do erro obtido, realizando-se assim, um filtro que possua características tanto de minimax como de WLS [10]-[11]. Isto é realizado calculando a envoltória apenas na faixa de frequência em que se deseja manter a amplitude dos picos constante (minimax), e mantendo os valores  $r_k(\omega_n)$  iguais para o resto da faixa, obtendo-se assim o efeito WLS. Na prática, utilizamos o algoritmo WLS, mas fazemos, a cada iteração, com que

$$B_k(\omega_n) = B_k(\omega_J), \quad \omega_n > \omega_J > \omega_s \quad (3.17)$$

na frequência de rejeição, o que fará com que os pesos, a partir da frequência  $\omega_J$  sejam mantidos constantes. Isto resultará em um decaimento típico de mínimo quadrático após esta frequência. Da mesma maneira, podemos aplicar o algoritmo na banda passante:

$$B_k(\omega_n) = B_k(\omega_J), \quad \omega_n < \omega_J < \omega_p \quad (3.18)$$

o que resultará em uma resposta com decaimento mínimo quadrado antes da frequência  $\omega_J$ . Como a amplitude dos picos próximos da faixa de transição do filtro é o que não satisfaz a especificação no projeto, apenas esses picos necessitam ser restritos, enquanto que os outros picos, que estão relaxados, e, portanto, atendendo às especificações, podem ser mantidos com o seu decaimento. Logicamente, ao restringir os picos mais próximos da faixa de transição, o decaimento tende a ser não tão pronunciado em relação ao filtro WLS, mas a restrição de alguns poucos picos já faz com que o filtro fique próximo do resultado minimax, nesta faixa de frequência.

O algoritmo modificado, que denotamos aqui por WLS-Chebyshev é descrito na Tabela 3.1

Como podemos notar, na primeira iteração, a matriz de pesos contém os valores iniciais  $\delta_p$  e  $\delta_s$  nas frequências de banda passante e de banda de rejeição, respectivamente. Como os pesos em cada uma dessas faixas serão iguais, o resultado da primeira iteração é um filtro com característica de mínimos quadrados.

Tabela 3.1: Algoritmo WLS-Chebyshev.

<b>Algoritmo WLS-Chebyshev</b>
<p><b>Inicialização:</b></p> <ol style="list-style-type: none"> <li>1. Crie uma grade uniforme de freqüências, desconsiderando a faixa de transição;</li> <li>2. Crie a matriz <math>\mathbf{U} = \cos(\omega_i, \mathbf{n})</math> ou <math>\mathbf{U} = \sin(\omega_i, \mathbf{n})</math>;</li> <li>3. Matriz <math>\mathbf{R}_0</math> onde <math>r_0(i, i) = \delta_p</math> ou <math>r_0(i, i) = \delta_s</math> de acordo com a faixa de freqüência;</li> </ol> <p><b>Algoritmo:</b></p> <ol style="list-style-type: none"> <li>4. Calcule <math>\mathbf{b}_k = (\mathbf{U}^T \mathbf{R}_k \mathbf{U})^{-1} \mathbf{U}^T \mathbf{R}_k \hat{\mathbf{H}}</math>;</li> <li>5. Obtenha a resposta do filtro <math>H(\omega_n) = \mathbf{U} \mathbf{b}</math>;</li> <li>6. Calcule o valor absoluto do erro <math>V(\omega_n) =  E  =  \sum r(i, i)( H(\omega_i)  -  \hat{H}(\omega_i) ) </math>;</li> <li>7. Obtenha os picos de máximo nas bandas, em que <math>V(\omega_n) &gt; V(\omega_{n\pm 1})</math>;</li> <li>8. Crie a função de envelope <math>B(\omega)</math>;</li> <li>9. A partir de um certo pico <math>p_j</math>, ou de uma freqüência <math>\omega_j</math>, mantenha <math>B(\omega)</math> constante, isto é, <math>B(\omega_i) = B(\omega_j), i &gt; j</math>;</li> <li>10. Calcule a função <math>\beta = \frac{B}{\sum r(\omega_m) B(\omega_m)}</math>;</li> <li>11. Atualize <math>\mathbf{R}_{k+1} = \mathbf{R}_k \beta</math>;</li> <li>12. Retorne ao passo 4.</li> </ol>

Em seguida, a função de envelope é obtida, e os pesos são atualizados a partir da Equação 3.11 apenas para determinados picos, ou até uma certa frequência. Nesta faixa, portanto, os pesos são atualizados até que a função envelope fique constante, obtendo-se assim a característica minimax para esta faixa de frequência.

### 3.4 Exemplo numérico

Para demonstrar a utilização do algoritmo WLS-Chebyshev, iremos projetar um filtro com as especificações:

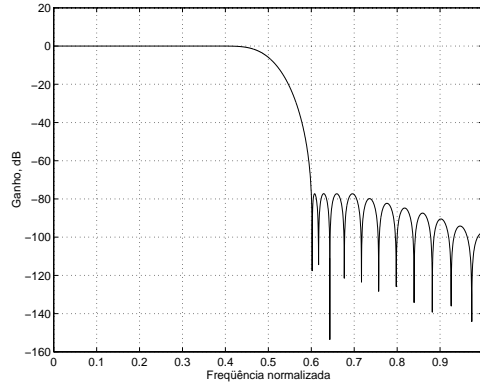
- Frequências de corte:  $\omega_p = 0.4\pi$  e  $\omega_s = 0.6\pi$
- Desvio máximo permitido na banda passante:  $\delta_p = 0.002$  dB
- Atenuação mínima na banda de rejeição:  $A_{\min} = -\delta_s = 75$  dB
- Restrição dos 4 primeiros picos na faixa de rejeição

Para este projeto, teremos, experimentalmente, que para a ordem  $N = 46$ , o filtro projetado com o algoritmo WLS-Chebyshev atende às especificações acima. A resposta em magnitude deste filtro pode ser vista na Figura 3.2. Os resultados obtidos para as otimizações minimax e mínimos quadrados está apresentada na Tabela 3.2.

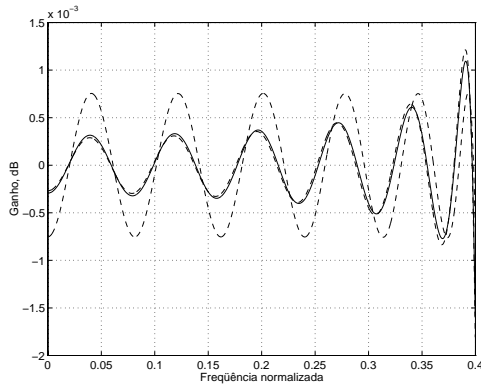
Como podemos notar pela Figura 3.2, a resposta em magnitude do filtro possui característica minimax até o quarto pico na banda de rejeição, e para frequências maiores, ocorre o decaimento da amplitude máxima dos picos. Note que o nível de amplitude dos picos restritos é ligeiramente acima do nível de amplitude do filtro minimax, e abaixo do nível do primeiro pico dos mínimos quadrados, que é a resposta em magnitude deste filtro na frequência de corte  $\omega_s$ . Portanto, ao restringirmos os primeiros picos, faremos com que o filtro passe a atender à especificação de atenuação mínima, sem, contudo, aumentar a energia total da banda de rejeição.

### 3.5 Vantagens do filtro WLS-Chebyshev

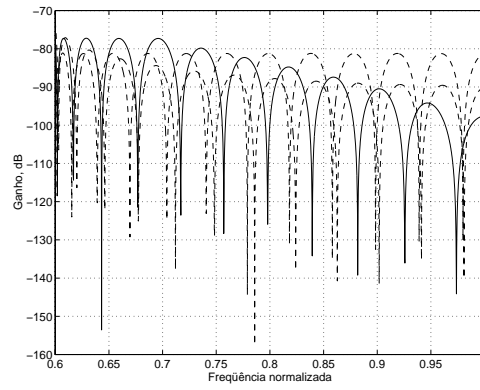
Como podemos notar, a alteração da resposta em frequência do filtro WLS-Chebyshev depende de se variar as funções de peso  $r_k(\omega_n)$ , o que torna o algoritmo bastante versátil no sentido de que qualquer forma de envoltória da resposta pode



(a) Resposta em magnitude do filtro WLS-Chebyshev.



(b) Banda passante do filtro WLS-Chebyshev (linha contínua) e dos filtros minimax e mínimos quadrados (linhas tracejadas).



(c) Banda de rejeição do filtro WLS-Chebyshev (linha contínua) e dos filtros minimax e mínimos quadrados (linhas tracejadas).

Figura 3.2: Resposta em magnitude de um filtro WLS-Chebyshev com os quatro primeiros picos da banda de rejeição restritos.

Tabela 3.2: Desvios obtidos para os filtros de ordem  $N = 46$  projetados pelos métodos minimax, mínimos quadrados e WLS-Chebyshev.

Método	$\delta_{p,max}$	$\delta_{s,max}$
minimax	0.00008 dB	-81 dB
mín. quad.	0.0016 dB	-72dB
WLS-Chebyshev	0.0014 dB	-77.3dB

ser realizada, variando-se os pesos  $r_k(\omega_n)$ , ou melhor ainda, modificando-se a própria função de envoltória. No caso da modificação realizada para se obter o algoritmo WLS-Chebyshev, o que ocorre na verdade é que a função de envoltória é mantida constante após um determinado número de picos, fazendo com que o algoritmo naturalmente modifique para a resposta de mínimos quadrados após esta frequência. Assim, se desejássemos, por exemplo, picos com atenuação muito alta no início da banda de rejeição, por exemplo, bastaria amplificar a função de envelope no início da faixa, que o problema estaria resolvido. É claro que não devemos esquecer que quanto mais um pico é restrito (ou seja, quanto mais o seu erro diminui), menor é o decaimento dos picos que não são restritos, e podem até mesmo, em vez de decaírem, começarem a ter amplitudes mais altas, sendo então necessário aumentar a ordem do filtro para sanar este problema, dependendo do que se deseje para a resposta do filtro. Pode ser visto em [8], por exemplo, que é possível realizar iterativamente um filtro WLS, de modo que a sua resposta em frequência se torne equirriple. Devido à facilidade em se operar sobre a matriz de pesos neste algoritmo, podemos empregá-lo na otimização de filtros FRM, na tentativa de compensar os desvios que ocorrem nas faixas especialmente críticas, tornando a resposta nesta faixa mais equalizada, ou ainda utilizá-lo para realizar as bandas críticas dos filtros de máscara.

### 3.6 Conclusões

Como vimos, o filtro WLS-Chebyshev utiliza a otimização de mínimos quadrados, ponderando o efeito de minimização em cada frequência. A função de peso para a obtenção da resposta WLS-Chebyshev é obtida iterativamente. Se desejamos que o filtro possua característica minimax, devemos deixar o algoritmo atualizar a matriz de peso em todas as frequências. Se desejamos que o filtro possua características de mínimos quadrados, podemos parar de atualizar os pesos, mantendo estes constantes em certas faixas de frequência. A resposta assim obtida é uma combinação dos dois tipos de otimização. As principais vantagens deste algoritmo são a facilidade na implementação e na modificação da função que atua sobre a atualização da matriz de peso. Um bom compromisso entre a energia total da banda de rejeição e a sua atenuação mínima representa um filtro que possuirá erros (desvios de ganho) não



tão altos nas proximidades da faixa de transição e redução dos picos subseqüentes, reduzindo com isso a energia da banda de rejeição.

Como veremos no próximo capítulo, a função de erro pode ser convenientemente escolhida, de acordo com o projeto, lembrando que a função da envoltória deste erro comanda a atualização da matriz de pesos do algoritmo.

# Capítulo 4

## Projeto de filtros FRM usando o algoritmo WLS-Chebyshev

### 4.1 Introdução

A utilização da otimização WLS-Chebyshev em projetos FRM é realizada na tentativa de reduzir ainda mais o número de coeficientes dos filtros componentes, aproveitando as características minimax e WLS obtidas para os filtros componentes com esta otimização, de modo a tornar o projeto mais eficiente. A partir da teoria de filtros FRM, sabemos que nos filtros obtidos, as faixas críticas do projeto, que são as duas bandas interpoladas mais próximas da faixa transição, são o resultado da adição das respostas de um dos filtros de máscaras com a resposta do filtro de base. No início da banda de rejeição verificamos, por exemplo, que existe um zero de transmissão de um dos filtros de máscara, mas logo em seguida vem o primeiro lóbulo lateral, que pode ter uma amplitude razoavelmente alta. Como este lóbulo é mapeado em algum lugar de uma das bandas do filtro interpolado, é possível reduzir o efeito de superposição realizando uma faixa de frequência tipo WLS na faixa em que o lóbulo é mapeado. Também podemos notar que no início da faixa de rejeição do filtro FRM, apenas alguns dos picos devem ser restritos para que não alcancem um nível muito alto e se tornem indesejados. Esta situação é justamente a que o filtro WLS-Chebyshev realiza. Portanto, será possível aproveitar mais eficientemente a banda de frequência nas bandas críticas do projeto do filtro FRM, e conseqüentemente reduzir a complexidade dos filtros componentes. Ainda mais, como podemos escolher

a função de erro do algoritmo WLS-Chebyshev facilmente, podemos utilizar não a resposta de um filtro componente em particular, mas sim a resposta final do filtro FRM, podendo obter assim características minimax nas faixas críticas, tornando o projeto ainda mais elaborado e eficiente.

Outra observação é quanto as margens de ganho para os filtros componentes. Se um dos filtros é projetado com o algoritmo WLS-Chebyshev, a idéia de margem de ganho perde um pouco a sua validade, já que os desvios variam para cada pico de amplitude. Na verdade, é difícil prever o número de coeficientes necessários para gerar um filtro com um número pré-determinado de picos restritos. O que se utiliza, portanto, é determinar o número de coeficientes por alto, por um estimador para filtros minimax, e depois aplicarmos o algoritmo WLS-Chebyshev, reprojetoando o filtro, se necessário. Assim, no projeto de filtros FRM com o algoritmo WLS-Chebyshev, é comum obtermos o número de coeficientes através de um estimador de ordem minimax, e depois devemos reduzir as margens dos filtros componentes até que o filtro fique com a resposta mais próxima possível da especificada. Se projetamos um filtro FRM por este método e estimamos as ordens por alto, podemos chegar a um projeto que possua margens de desvio de ganho negativas, indicando que as ordens (e, com isto, a complexidade do filtro) foram superestimadas, indicando claramente que este projeto é mais eficiente, em termos de redução, que um projeto minimax.

Finalmente, poderemos utilizar o algoritmo WLS-Chebyshev para realizar os filtros de máscara com bandas críticas, aproximando-os do caso minimax, e como pode ser visto, isto também faz com que se reduza a complexidade computacional do filtro FRM final.

Na seção 4.2 mostraremos como utilizar o algoritmo WLS-Chebyshev para otimizar os filtros de máscara. Na seção 4.3 introduziremos o algoritmo WLS-Chebyshev para o projeto do filtro de base, respeitando as condições das bandas especialmente críticas. Na seção 4.4, realizaremos alguns exemplos numéricos para ilustrar a redução obtida com estes filtros em relação ao projeto FRM convencional mostrado no Capítulo 2.

## 4.2 Projeto de filtros de máscara com o algoritmo

### WLS-Chebyshev

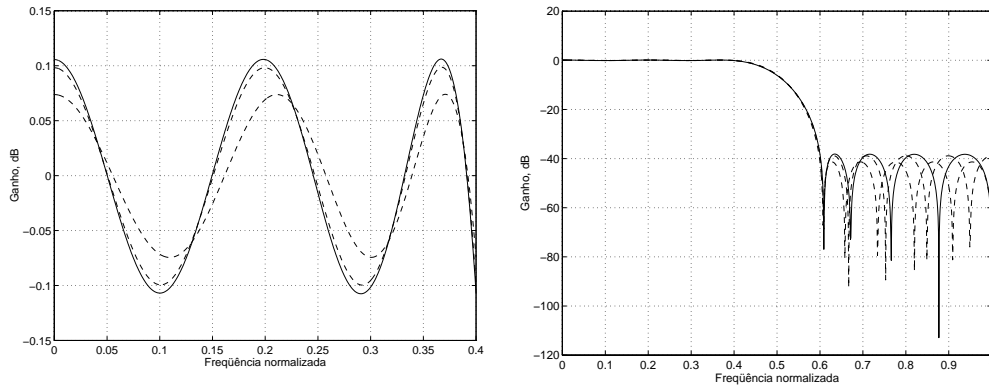
Como pode ser notado na análise do cálculo das margens de desvio de ganho necessárias em cada faixa de frequência em um projeto passa-baixas FRM vista na seção 2.5 assim como no apêndice A, as bandas especialmente críticas são a última banda do filtro interpolado que compõe o final da banda passante do filtro FRM, e a primeira banda do filtro interpolado que compõe o início da banda de rejeição do filtro FRM. Nas outras bandas, como os dois filtros de máscara estarão simultaneamente com respostas próximas de um ou próximas de zero, os desvios de ganho da resposta do filtro interpolado e da resposta complementar deste filtro praticamente se anulam<sup>1</sup>, e pode ser observado que, na maioria dos projetos, a margem necessária nas faixas não-críticas é um valor muito baixo. O projeto de filtros de máscara é feito, portanto, fazendo com que os desvios nas bandas críticas sejam ligeiramente abaixo do especificado para o filtro resultante, e tentando reduzir ao máximo os desvios nas bandas especialmente críticas. Ou seja, dado que se queira um filtro de máscara com um certo número de coeficientes, o projeto tenta maximizar o erro do filtro em até uma certa margem do filtro final desejado, nas bandas não críticas, relaxando assim o filtro de máscara ao máximo possível nestas bandas, para que se possa reduzir ao máximo o desvio nas bandas críticas. Pode-se entender esta operação como se estivéssemos alocando todos os zeros possíveis das faixas não especialmente críticas do filtro para as faixas especialmente críticas, aumentando assim a concentração de zeros nestas faixas, de modo a reduzir o desvio do filtro, cujos máximos ocorrem nos pontos entre dois zeros consecutivos.

Assim, teremos os seguintes projetos para o filtro de máscara:

1. Projetando com bandas críticas, e, usando um pouco menos do que a especificação do filtro final desejado para as bandas críticas (de acordo com o calculado para a margem de ganho nestas bandas), tentando minimizar o desvio nas faixas especialmente críticas (que é a solução de pior caso);

---

<sup>1</sup>Os desvios não serão totalmente cancelados, uma vez que os *ripples* dos filtros de máscara, que modulam os *ripples* das respostas do filtro interpolado e da resposta complementar, não serão idênticos.



(a) Resposta na banda passante.

(b) Resposta na banda de rejeição.

Figura 4.1: Filtros FIR projetados com o algoritmo minimax, para  $\omega_p = 0.4\pi$  e  $\omega_s = 0.6\pi$ , com ordens  $N=19, 20$  e  $21$ . Note que apenas com  $N = 21$  o projeto entra na especificação de  $\delta_p \leq 0.1$  dB e  $\delta_s \leq -40$  dB, e ainda fornece uma margem de desvios razoável de aproximadamente 20%.

2. Projetando com bandas críticas, e usando a mesma especificação do filtro final desejado para as bandas críticas, minimizando o desvio nas faixas especialmente críticas. Este método se aplica quando a variação de mais ou menos um na ordem o filtro é mais que suficiente para que o filtro atenda às especificações, como mostra o exemplo da Figura 4.1. Neste caso, o número da ordem necessária para realizar o projeto deixa também uma pequena margem suficiente para ser utilizada como margem de desvio nas bandas críticas do filtro;
3. Realizando um algoritmo de projeto que leve em conta a resposta do filtro final, nas faixas críticas, fazendo com que se minimize o desvio até que o filtro final obedeça à especificação.

Como dissemos, além de obedecer a uma das regras acima, é necessário também reduzir-se ao máximo o desvio do filtro de máscara nas bandas especialmente críticas do filtro FRM.

Nos dois primeiros casos enumerados acima, os filtros de máscara podem ser projetados independentes do filtro de base, enquanto que no terceiro caso é necessário que o filtro de base seja projetado antes do filtro de máscara. Também devemos

tomar cuidado com as bandas especialmente críticas no terceiro caso, já que o algoritmo utiliza a resposta final do filtro para tentar ajustar o *ripple* do filtro de máscara, mas nestas faixas o *ripple* se torna quase que exclusivamente dependente do filtro interpolado. Portanto, haverá muitas variações de *ripple* nestas faixas, o que faz com que o algoritmo não funcione, já que se estará tentando ajustar desvios positivos e negativos em frequências muito próximas em relação à baixa ordem dos filtros de máscara. De fato, como será visto mais tarde, é necessário criar uma função aproximada da resposta do filtro FRM final nestas bandas, para evitar este problema.

### 4.2.1 Projeto de filtros de máscara independentes

Nos projetos dos filtros de máscara independentes do filtro de base, estaremos projetando filtros cujas especificações são obtidas diretamente a partir da especificação do filtro reduzido desejado, pela margem de desvio a ser deixada, pelo fator  $L$  e pelas frequências de corte do filtro reduzido.

Uma vez definidos, portanto,  $L$ ,  $\omega_p$ ,  $\omega_s$ ,  $\delta_{p,PROJ}$  e  $\delta_{s,PROJ}$  para o filtro desejado, e as margens de desvio de ganho para os filtros de máscara positiva e negativa  $\delta_{m+}$  e  $\delta_{m-}$ , respectivamente, pode-se projetar os filtros de máscara do filtro FRM final, observando a construção das Figuras 2.3 e 2.4, caindo em um dos casos das equações (2.5)-(2.7) ou (2.8)-(2.10). Como foi visto no Capítulo 2, as frequências de corte dos filtros de máscara serão dados pelas equações (2.11)-(2.14) ou (2.15)-(2.18), de acordo com [1].

Uma vez que as frequências são obtidas, pode-se utilizar um algoritmo min-max, resultando em um filtro com características “equiripple” para o projeto dos filtros de máscara. A redução do número de coeficientes do filtro de máscara é obtida também utilizando bandas não críticas, e neste caso, o projeto WLS pode ser mais vantajoso que o minimax, uma vez que os maiores erros do WLS ocorrem nos limites das bandas críticas, que não são tão sensíveis ao erro, e portanto não necessitam ser tão restritas, como veremos a seguir.

Para um projeto com bandas críticas, as frequências limites de cada banda do filtro interpolado são obtidas a partir da construção da resposta em frequência.

Para a primeira banda do filtro de máscara positiva, teremos

$$\omega_{+,a} = 0 \quad (4.1)$$

$$\omega_{+,b} = \frac{\phi}{L} \quad (4.2)$$

e, para as demais, teremos os valores das equações (2.19) e (2.20) para  $m' = 1, 2, \dots$

Do mesmo modo, para o filtro de máscara negativa, teremos, para a primeira banda

$$\omega_{-,a} = \frac{\theta}{L} \quad (4.3)$$

$$\omega_{-,b} = \frac{2\pi - \theta}{L} \quad (4.4)$$

e para as demais, o resultado das equações (2.22) e (2.23). Com isto, obtemos os pares de início e de fim das bandas críticas de cada um dos filtros de máscara, e podemos agora aplicar o algoritmo WLS-Chebyshev, que é realizado em dois passos:

1. Obter resposta minimax para todas as bandas críticas do filtro de máscara em questão, e
2. Maximizar o desvio nestas bandas críticas do projeto, reduzindo ao máximo o erro na banda especialmente crítica do filtro de máscara em questão.

Assim obteremos o algoritmo da Tabela 4.1.

## 4.2.2 Projeto com tomada da função de erro na resposta final

É possível modificar o algoritmo da Tabela 4.1 para que não seja necessário especificar a margem de desvio de ganho para os filtros de máscara, bastando, para isto, calcular o erro a partir da resposta final do filtro e ajustar os coeficientes dos filtros de máscara de modo a obter os desvios máximos nas bandas não-críticas abaixo de  $\delta_{p,PROJ}$  e  $\delta_{s,PROJ}$ . Isto, entretanto, só é possível se já tivermos projetado o filtro de base, para compor a resposta do filtro FRM final. O algoritmo modificado utiliza, portanto, o valor máximo absoluto do erro do filtro final:

$$\max\{|E(\omega_n)|\} = \max\{|H_b^I(\omega_n)H_+(\omega_n) + H_{bc}^I(\omega_n)H_-(\omega_n) - \hat{H}(\omega_n)|\} \quad (4.5)$$

Tabela 4.1: Algoritmo para os filtros de máscara independentes.

<b>Algoritmo WLS para máscaras com bandas não críticas</b>
<p><b>Inicialização:</b></p> <ol style="list-style-type: none"> <li>1. Obtenha as frequências dos limites das bandas críticas;</li> <li>2. Crie uma grade linear das frequências dentro de cada banda;</li> <li>3. Utilize o algoritmo WLS-Chebyshev com <math>\delta_p</math> e <math>\delta_s</math> e com todos os picos restritos, para gerar as bandas de cada filtro;</li> </ol> <p><b>Para cada iteração <math>k</math>:</b></p> <ol style="list-style-type: none"> <li>4. Calcule <math>\mathbf{b}_k = (\mathbf{U}^T \mathbf{R}_k \mathbf{U})^{-1} \mathbf{U}^T \mathbf{R}_k \hat{\mathbf{H}}</math>;</li> <li>5. Obtenha a resposta do filtro <math>H_k(\omega_n) = \mathbf{U} \mathbf{b}_k</math>;</li> <li>6. Calcule o máximo valor absoluto do erro:  <math>\max\{ E_k(\omega_n) \} = \max\{ \sum r_k(\omega_i)[ H_k(\omega_i)  -  \hat{H}(\omega_i) ]\}</math> para cada banda crítica, que não seja a banda especialmente crítica;</li> <li>7. Calcule o erro <math>e_k(\omega_n) = \delta - \max  E(\omega_n) </math> em cada banda, onde <math>\delta = \delta_p</math> ou <math>\delta = \delta_s</math> dependendo da faixa de frequência;</li> <li>8. Atualize todos os pesos destas bandas segundo a relação:  <math>r_{k+1}(\omega_i) = \frac{r_k(\omega_i)}{1 + \min\{e_k(\omega_n)\}}</math>;</li> <li>9. Retorne ao passo 4.</li> </ol>



tentando minimizar a diferença:

$$e_i = \delta_{PROJ} - \max\{|E(\omega_i)|\} \quad (4.6)$$

para cada uma das bandas *care* dos filtros de máscara, isto é, para  $\omega_i \leq \omega_n \leq \omega_{i+1}$ , onde  $i$  é a banda considerada e  $\delta_{PROJ} = \delta_{p,PROJ}$  caso a banda pertença à banda passante, ou  $\delta_{PROJ} = \delta_{s,PROJ}$  caso a banda pertença à banda de rejeição do filtro de máscara. Como nas bandas não-críticas do projeto do filtro FRM apenas um dos filtros de máscara predomina, é possível reduzir a equação (4.5) para:

$$\max\{|E(\omega_n)|\} \approx \max\{|H_b^I(\omega_n)H_+(\omega_n) - \hat{H}(\omega_n)|\} \quad (4.7)$$

no caso da máscara positiva, ou:

$$\max\{|E(\omega_n)|\} \approx \max\{|H_{bc}^I(\omega_n)H_-(\omega_n) - \hat{H}(\omega_n)|\} \quad (4.8)$$

no caso da máscara negativa predominar, e, com isto, podemos projetar os filtros de máscara um independente do outro. O algoritmo da Tabela 4.2 apresenta as modificações necessárias.

### 4.3 Projeto do filtro base com o algoritmo WLS-Chebyshev

De acordo com a construção do filtro FRM, é o filtro de base o responsável pela transição abrupta da resposta final, e portanto, as frequências de corte do filtro de base necessitam estar bem definidas. Como, entretanto, as respostas direta e complementar deste filtro, quando interpolado, quase se anulam na maior parte da faixa de frequência do espectro excluindo-se apenas as bandas especialmente críticas, podemos projetar o filtro de base levando em conta os efeitos que ocorrem apenas nestas bandas. Será possível, portanto, mapear as respostas dos filtros de máscara destas regiões para a resposta em frequência do filtro de base, para que o projeto deste filtro possa compensar os efeitos indesejáveis dos filtros de máscara nestas regiões.

Se o filtro de máscara positiva possui corte após o filtro de máscara negativa, então a banda passante do filtro de base formará a faixa crítica da banda passante

Tabela 4.2: Algoritmo WLS-Chebyshev com tomada de erro na resposta final

<b>Algoritmo WLS Chebyshev com tomada de erro na resposta final</b>
<p><b>Inicialização:</b></p> <ol style="list-style-type: none"> <li>1. Obtenha as frequências dos limites das bandas críticas;</li> <li>2. Crie uma grade linear de frequências dentro de cada banda;</li> <li>3. Utilize o algoritmo WLS com <math>\delta_{p,PROJ}</math> e <math>\delta_{s,PROJ}</math> e com todos os picos restritos, para gerar as bandas críticas dos filtros;</li> </ol> <p><b>Para cada iteração <math>k</math>:</b></p> <ol style="list-style-type: none"> <li>4. Calcule <math>\mathbf{b}_k = (\mathbf{U}^T \mathbf{R}_k \mathbf{U})^{-1} \mathbf{U}^T \mathbf{R}_k \hat{\mathbf{H}}</math>;</li> <li>5. Obtenha a resposta do filtro <math>H_k(\omega_n) = \mathbf{U} \mathbf{b}_k</math>;</li> <li>6. Calcule o máximo do valor absoluto do erro:  <math>\max\{ \sum r_k(\omega_i)( H^I(\omega_i)H_k(\omega_i)  -  \hat{H}(\omega_i) ) \}</math> para cada banda <i>care</i> que não seja a banda crítica;</li> <li>7. Calcule o erro <math>e_k(\omega_n) = \delta - \max\{ E(\omega_n) \}</math> em cada banda, onde <math>\delta = \delta_{p,PROJ}</math> ou <math>\delta = \delta_{s,PROJ}</math> dependendo da faixa de frequência;</li> <li>8. Atualize todos os pesos desta faixa segundo a relação:  <math display="block">r_{k+1}(\omega_i) = \frac{r_k(\omega_i)}{1 + \min\{e_k(\omega_n)\}};</math></li> <li>9. Retorne ao passo 4.</li> </ol>

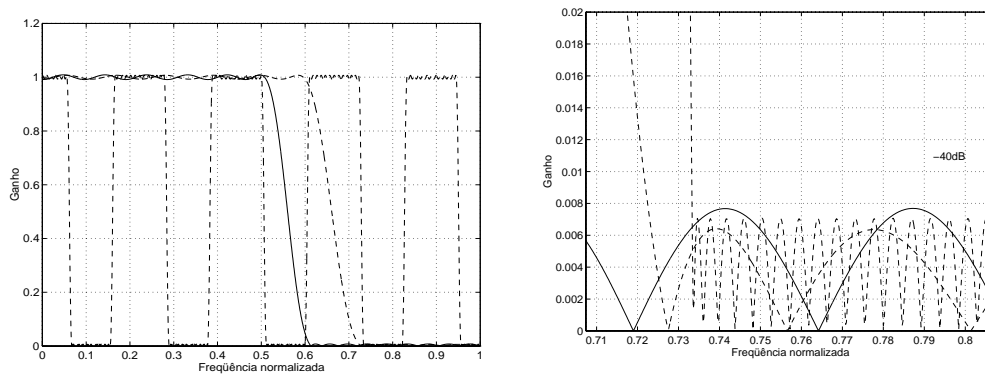
da resposta final do filtro FRM, e a banda de rejeição do filtro de base formará a faixa crítica da banda de rejeição da resposta final do filtro FRM. Caso contrário, a situação se inverte, pois a última banda do filtro interpolado que irá compor a banda passante provém da resposta complementar do filtro interpolado, e, portanto, a banda de rejeição do filtro de base formará a faixa crítica da banda passante da resposta final do filtro FRM, e a banda passante do filtro de base formará a faixa crítica da banda de rejeição da resposta final do filtro FRM.

Assim, as respostas dos filtros de máscara podem ser mapeadas diretamente ou inversamente sobre a resposta do filtro de base, para que seja possível compor a resposta na faixa de frequência que vai desde o início da primeira banda especialmente crítica até o final da segunda banda especialmente crítica. Em seguida, tentamos obter os coeficientes do filtro de base de modo a minimizar o erro do filtro resultante.

Uma vez que o fator de interpolação seja relativamente pequeno, e que as ordens dos filtros componentes sejam próximas entre si, obteremos a superposição de apenas alguns lóbulos laterais dos filtros de máscara nas bandas críticas, como podemos notar na Figura 4.2. Se  $L = 2$ , por exemplo, a ordem do filtro interpolado será cerca de duas vezes maior que a dos filtros de máscara. Para fatores ainda maiores, é fácil notar que os filtros de máscara serão bem mais suaves que o filtro interpolado (pois possuirá zeros mais esparsos no círculo unitário que o filtro interpolado). Portanto, presume-se que apenas alguns poucos lóbulos laterais serão mapeados nas bandas especialmente críticas do filtro FRM.

Ainda de acordo com a Figura 4.2, notamos claramente que um filtro com características WLS pode se tornar vantajoso como filtro de base, pois podemos utilizar o decaimento dos picos da resposta do filtro interpolado para cancelar o efeito dos lóbulos dos filtros de máscara.

O projeto, entretanto, não pode ser puramente WLS, sobretudo por causa das frequências de corte do filtro de base, que necessitam estar bem definidas. Como o filtro WLS possui o maior erro justamente nas frequências de corte [8], é necessário ao menos restringir os erros nestas duas frequências de modo que o projeto atenda às especificações iniciais, e, portanto, o algoritmo WLS-Chebyshev pode ser aqui bem aplicado. De fato, aplicando o algoritmo WLS-Chebyshev com apenas as ampli-



(a) Operação das máscaras do filtro FRM.

(b) Detalhe da banda crítica no início da banda de rejeição.

Figura 4.2: A banda crítica de rejeição de um filtro FRM típico. Note que apenas alguns poucos lóbulos dos filtros de máscara estão presentes nesta faixa de frequência.

tudes nas frequências de corte restritas, já se consegue alguma redução no número de coeficientes do filtro de base. Neste caso, o projeto do filtro de base é independente dos projetos dos filtros de máscara, e devemos especificar margens para este filtro. Como podemos notar, também, em um projeto de filtros FRM, na banda especialmente crítica de rejeição, o filtro de máscara que predomina possui um zero de transmissão que é alocado nas proximidades da faixa de transição da resposta final do filtro FRM, e, portanto, podemos relaxar as restrições de pico nas proximidades deste zero, e com isto aumentar as restrições dos picos nas outras frequências, onde ocorrem os máximos dos lóbulos laterais do filtro de máscara em questão. Com isto, chegamos a um algoritmo que tenta tornar a resposta do filtro final, nestas bandas especialmente críticas, no formato minimax até o primeiro máximo da resposta em frequência do filtro de máscara, aproveitando, após esta frequência, a característica WLS (de decaimento da resposta em magnitude) do algoritmo WLS-Chebyshev, após o primeiro máximo da resposta em frequência do filtro de máscara.

O algoritmo que realiza o método acima proposto é descrito na Tabela 4.3.

A operação de mapeamento dos filtros de máscara nas bandas especialmente críticas do filtro base é facilmente obtida, de acordo com o fator de interpolação  $L$

Tabela 4.3: Algoritmo WLS-Chebyshev para o filtro base.

<b>Algoritmo WLS-Chebyshev para o filtro base</b>
<p><b>Inicialização:</b></p> <ol style="list-style-type: none"> <li>1. Crie uma grade uniforme de frequências de <math>0 \leq \omega \leq \theta</math> e <math>\phi \leq \omega \leq \pi</math>;</li> <li>2. A partir da grade acima, mapeie as frequências nas bandas especialmente críticas: <math>\omega' = \frac{\omega}{L} + \omega_1</math> ou <math>\omega' = \omega_2 - \frac{\omega}{L}</math>;</li> <li>3. Calcule as respostas <math>H_+(\omega')</math> e <math>H_-(\omega')</math>;</li> <li>4. Crie a matriz <math>\mathbf{U} = \text{trig}(\omega, i)</math>;</li> <li>5. Crie a matriz <math>\mathbf{R}</math> de modo que <math>r(i, i) = \delta_{p, PROJ}</math> ou <math>r(i, i) = \delta_{s, PROJ}</math>;</li> </ol> <p><b>Para cada iteração <math>k</math>:</b></p> <ol style="list-style-type: none"> <li>6. Calcule <math>\mathbf{b}_k = (\mathbf{U}^T \mathbf{R}_k \mathbf{U})^{-1} \mathbf{U}^T \mathbf{R}_k \hat{\mathbf{H}}</math>;</li> <li>7. Calcule <math>H_k(\omega_n) = \mathbf{U} \mathbf{b}_k</math>;</li> <li>8. Calcule a resposta final <math>H_{r,k}(\omega_n) = H_k(\omega_n) H_+(\omega'_n) + (1 - H_k(\omega_n)) H_-(\omega'_n)</math>, onde <math>\omega'_n = \frac{\omega_n}{L} + \omega_1</math>;</li> <li>9. Calcule o valor absoluto do erro <math> E_k(\omega_n)  =  \sum r(n, n)(H(\omega_n) - H_r(\omega_n)) </math>;</li> <li>10. Calcule a envoltória do erro <math>V_k(\omega_n)</math>;</li> <li>11. A partir de um certo pico <math>p_{j,k}</math> ou de uma frequência <math>\omega_j</math>, matenha <math>B(\omega_n)</math> constante;</li> <li>12. Calcule <math>\beta_k = \frac{B(\omega_n)}{\sum r(m, m) B(\omega_m)}</math>;</li> <li>13. Atualize <math>\mathbf{R}</math>: <math>\mathbf{R}_{k+1} = \mathbf{R}_k \beta_k</math>;</li> <li>14. Retorne ao passo 6.</li> </ol>

e do cálculo das freqüências limiares das bandas críticas

$$\omega' = \frac{\omega}{L} + \omega_1 \quad (4.9)$$

onde  $\omega'$  representa a freqüência de espectro do filtro base, e  $\omega_1$  é a freqüência do centro da última banda crítica da banda passante do filtro FRM

$$\omega_1 = \frac{2m'\pi}{L} \quad (4.10)$$

Para determinar o valor de  $m'$  na equação (4.10) acima, basta verificar para os valores inteiros de  $m'$ , o valor que está imediatamente abaixo de  $\omega_{p,PROJ}$ .

Caso o mapeamento seja invertido, isto é, se o filtro de máscara negativa corta após o filtro de máscara positiva, teremos

$$\omega' = \omega_2 - \frac{\omega}{L} \quad (4.11)$$

onde  $\omega_2$  é a freqüência do centro da primeira banda de rejeição crítica

$$\omega_2 = \frac{(2m' + 1)\pi}{L} \quad (4.12)$$

e  $m'$  na equação (4.12) será o valor inteiro que satisfizer o valor de freqüência imediatamente acima da freqüência de corte  $\omega_{s,PROJ}$ , ou seja, a freqüência mais baixa a satisfazer  $\omega_2 > \omega_{s,PROJ}$ .

No cálculo do erro, poderemos então utilizar o mapeamento, obtendo o erro nas faixas críticas da resposta final do filtro FRM:

$$E_k(\omega_n) = r(n, n)[H_k(\omega_n)H_+(\omega_n) + (1 - H_k(\omega_n))H_-(\omega_n)] \quad (4.13)$$

e utilizar esta função na minimização do algoritmo WLS-Chebyshev para determinar os coeficientes do filtro de base.

## 4.4 Exemplos numéricos

Nesta seção, ilustraremos os métodos mencionados na seção anterior, comparando com os resultados obtidos nos exemplos do Capítulo 2.

1. Projeto de filtro FRM utilizando o algoritmo WLS-Chebyshev no filtro de base, WLS nos filtros de máscara, com bandas de *don't care*.

Tabela 4.4: Linha ótima de redução para o exemplo 1.

$L$	$N_b$	$N_+$	$N_-$	$M_{Tot}$	Redução (%)
7	60	35	25	126	32.2

Tabela 4.5: Desvios máximos obtidos para o exemplo 1.

Filtro	$\delta_{p,max}$	$\delta_{s,max}$
Base	0.178dB	-39.81dB
Máscara positiva	0.341dB	-41.11dB
Máscara negativa	1.269dB	-24.49dB
Filtro final	0.174dB	-40.01dB

Neste projeto, utilizamos o algoritmo WLS-Chebyshev para a obtenção dos coeficientes do filtro de base. A idéia, aqui, é restringir o último pico da banda passante e o primeiro pico da banda de rejeição do filtro de base, verificando se a característica *Least-Squares* é suficiente para a redução do filtro. Utilizamos, para este exemplo, a seguinte relação nos pesos das faixas dos filtros de máscara:

$$w[\delta_p] = 1.4w[\delta_s]; \quad w[\delta_{pp}] = 1.8w[\delta_s]; \quad w[\delta_{ss}] = 1.4w[\delta_s]$$

e com isto, poderemos reduzir as margens de desvio de ganho para cerca de 6%, obtendo para este projeto a linha ótima de redução vista na Tabela 4.4.

Com isto, obteremos os desvios máximos de ganho nos filtros componentes e no filtro FRM final vistos na Tabela 4.5.

- Projeto de filtro FRM utilizando o método WLS-Chebyshev no filtro de base com tomada da função de erro na saída do filtro FRM; WLS utilizado para realizar os filtros de máscara, e uso de bandas não críticas.

Neste projeto, a margem necessária é de apenas 3%, onde se restringiu nove picos na banda passante do filtro de base e cinco na banda de rejeição. Foram realizadas 30 iterações para o WLS-Chebyshev convergir, e mais 15 iterações para cada um dos filtros de máscara. O ajuste nas regiões críticas foram:

Tabela 4.6: Linha ótima de redução para o exemplo 2.

$L$	$N_b$	$N_+$	$N_-$	$M_{Tot}$	Redução (%)
7	58	34	26	121	31.7

Tabela 4.7: Desvios máximos obtidos para o exemplo 2.

Filtro	$\delta_{p,max}$	$\delta_{s,max}$
Base	0.282dB	-36.26dB
Máscara positiva	0.882dB	-21.99dB
Máscara negativa	0.836dB	-22.69dB
Filtro final	0.197dB	-40.08dB

$$w[\delta_p] = 1.2w[\delta_s]; \quad w[\delta_{pp}] = 1.2w[\delta_s]; \quad w[\delta_{ss}] = 1.3w[\delta_s]$$

resultando na linha ótima de redução vista na Tabela 4.6.

Para este projeto, os desvios máximos de ganho para os filtros componentes e para o filtro FRM final obtidos estão apresentados na Tabela 4.7.

3. Projeto de filtro FRM utilizando o método WLS-Chebyshev no filtro de base com tomada da função de erro na saída do filtro FRM; WLS utilizado para realizar os filtros de máscara com tomada da função de erro na saída do filtro FRM, e uso de bandas não críticas.

Neste terceiro projeto, inicialmente projetamos um filtro de base minimax, e em seguida projetamos os filtros de máscara, utilizando a resposta do filtro final; em seguida, otimizamos o filtro base WLS-Chebyshev de acordo com os filtros de máscara obtidos. Portanto, esta é uma otimização em dois passos. Com isto, obtivemos a linha ótima de redução vista na Tabela 4.8. Foram, assim, realizadas 20 iterações para cada um dos filtros de máscara, e mais 20 iterações para o filtro de base, até a convergência do método. Neste caso, uma vez que o ajuste de todas as bandas críticas são realizadas tomando-se a



Tabela 4.8: Linha ótima de redução para o exemplo 3.

$L$	$N_b$	$N_+$	$N_-$	$M_{Tot}$	Redução (%)
7	56	31	25	115	30.1

Tabela 4.9: Desvios máximos obtidos para o exemplo 3.

Filtro	$\delta_{p,max}$	$\delta_{s,max}$
Base	0.318dB	-36.08dB
Máscara positiva	0.475dB	-20.74dB
Máscara negativa	0.734dB	-23.55dB
Filtro final	0.196dB	-40.11dB

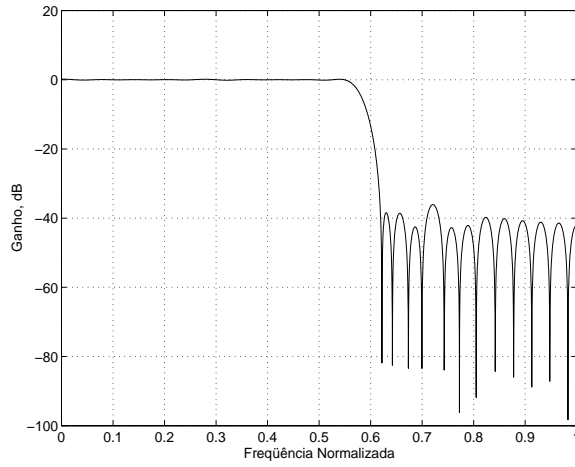
função de erro na saída do filtro final, não foi necessário especificar a pesagem das várias bandas.

Para este projeto, obteremos as respostas em magnitude ilustradas nas Figuras 4.3, 4.4, 4.5 e 4.6. Os desvios máximos de ganho para os filtros componentes e para o filtro FRM final obtidos estão apresentados na Tabela 4.9.

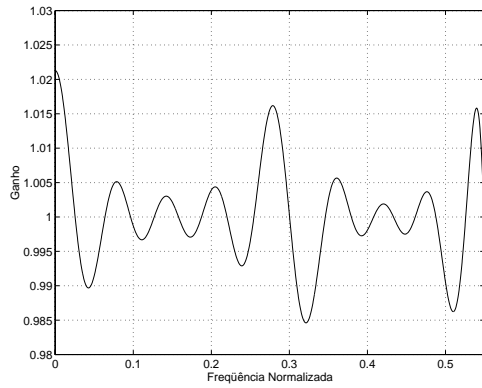
## 4.5 Conclusões

O projeto dos filtros de máscara usando o algoritmo WLS-Chebyshev apresenta uma pequena vantagem sobre os projetos anteriores, pois o decaimento WLS não é bem aproveitado, já que a resposta do filtro base, estando repetida no espectro de frequência, faz com que se deseje uma resposta cuja energia seja constante para os filtros de máscara. Entretanto para projetos que utilizam estruturas em cascata, o projeto de um filtro de máscara WLS-Chebyshev pode se tornar interessante. Se construirmos um filtro de base, por exemplo, a partir de um outro projeto de filtro FRM, o filtro WLS-Chebyshev utilizado nas máscaras resulta em um filtro de base que possui decaimento na resposta em frequência, o que pode ser interessante, e pode aumentar a redução do filtro FRM.

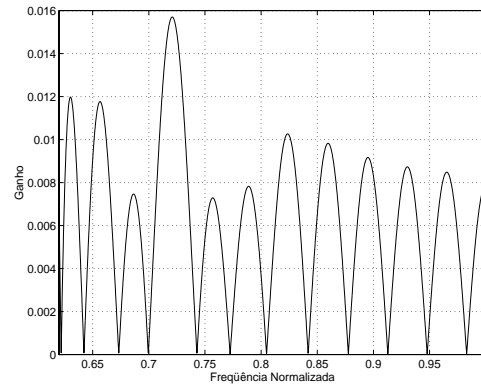
Outro caso em que o WLS-Chebyshev pode ser empregado é no caso da



(a) Resposta em magnitude do filtro.

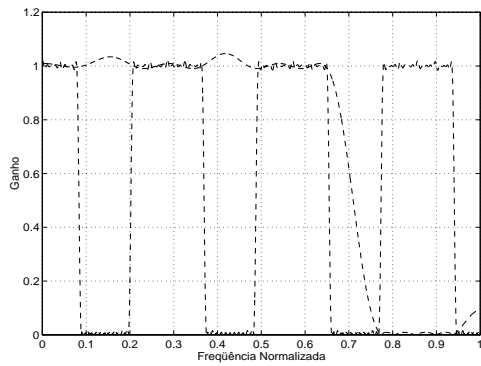


(b) Resposta na banda passante.

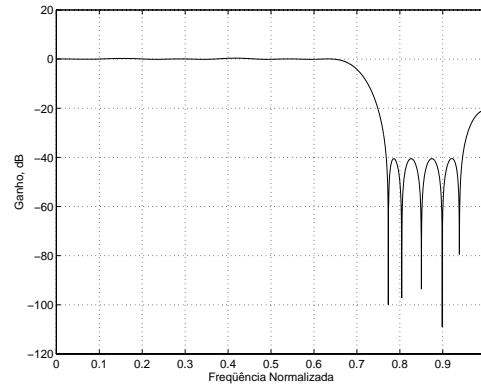


(c) Resposta na banda de rejeição.

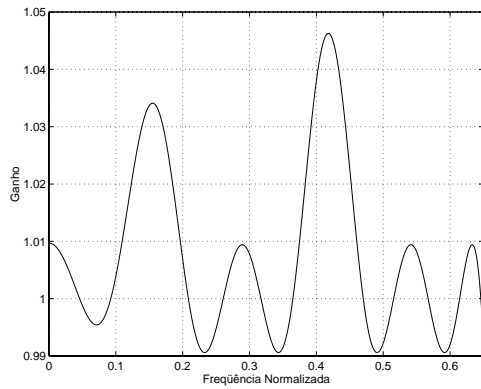
Figura 4.3: Filtro de base para o projeto FRM com filtro de base e máscaras WLS-Chebyshev e tomada de erro na resposta final, usando bandas não críticas.



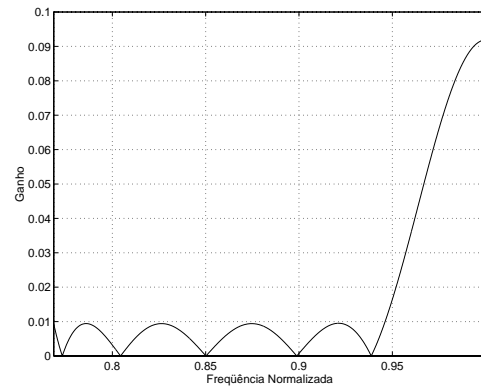
(a) Operação de máscara positiva.



(b) Resposta em magnitude do filtro de máscara positiva.

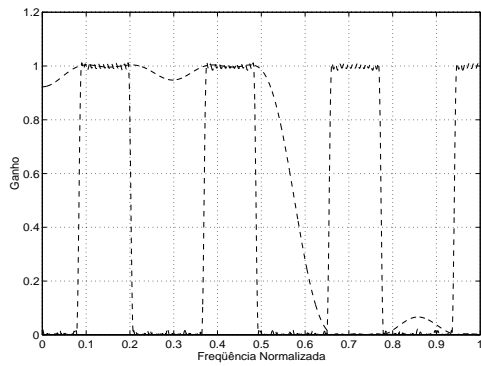


(c) Resposta na banda passante.

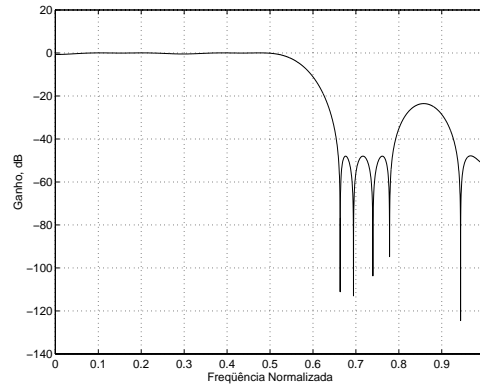


(d) Resposta na banda de rejeição.

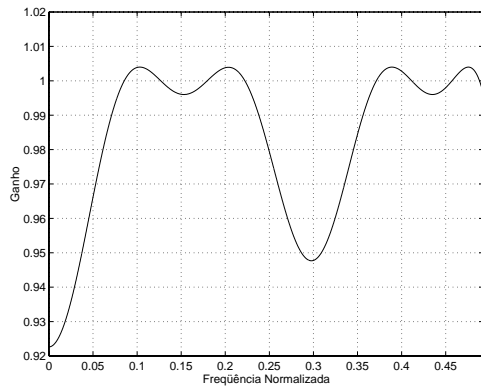
Figura 4.4: Funcionamento da máscara positiva para o projeto do filtro FRM com filtro de base e máscaras WLS-Chebyshev e tomada de erro na resposta final, usando bandas não críticas.



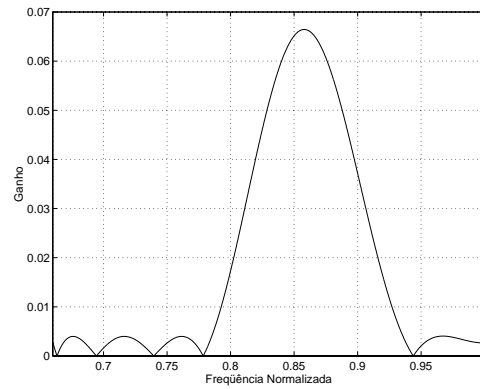
(a) Operação de máscara negativa.



(b) Resposta em magnitude do filtro de máscara negativa.

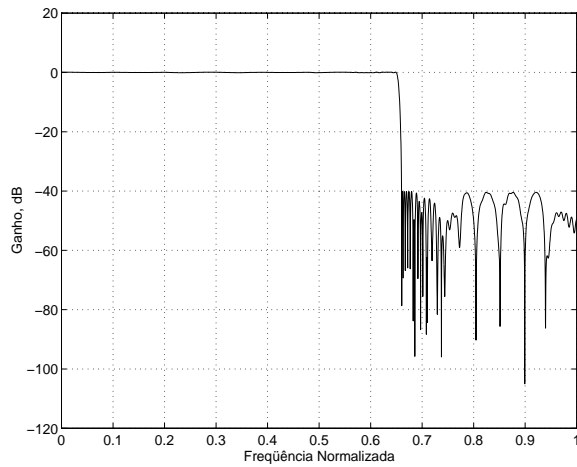


(c) Resposta na banda passante.

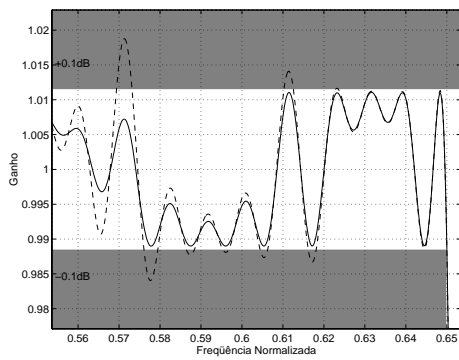


(d) Resposta na banda de rejeição.

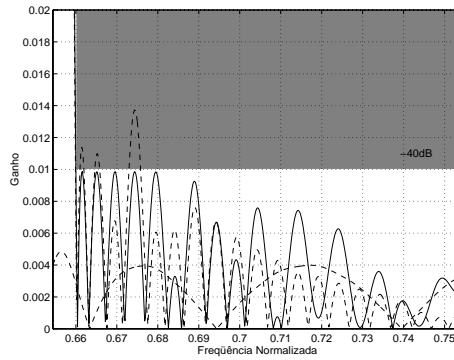
Figura 4.5: Funcionamento da máscara negativa para o projeto do filtro FRM com filtro de base e máscaras WLS-Chebyshev e tomada de erro na resposta final, usando bandas não críticas.



(a) Resposta em magnitude do filtro resultante.



(b) Detalhe do final da banda passante.



(c) Detalhe do início da banda de rejeição.

Figura 4.6: Resposta resultante em magnitude do filtro FRM com filtro de base e máscaras WLS-Chebyshev e tomada de erro na resposta final, usando bandas não críticas.

fatoração dos filtros de máscara de um projeto de filtro FRM. De fato, ao fatorarmos os filtros de máscara, retirando alguns zeros das proximidades da faixa de transição, para formar o filtro comum das duas máscaras, este filtro terá um comportamento de decaimento nos picos de amplitude. Este filtro pode, portanto, ser trocado por um filtro WLS-Chebyshev, e os zeros resultantes do dois filtros de máscara poderão ser ajustados para se obter uma resposta razoável para os filtros de mascaramento.

# Capítulo 5

## Banco de Filtros

### 5.1 Introdução

Atualmente, sistemas com múltiplos filtros são utilizados em várias aplicações em que seja necessária a combinação ou separação de informações. Nas telecomunicações, em codificação ou em computação, temos inúmeros exemplos de utilização de bancos de filtros, por exemplo ao transmitirmos vários sinais provenientes de várias origens em um único meio particular, ou no caso de se dividir sinais para serem processados paralelamente, codificados ou analisados separadamente, etc. Os métodos conhecidos para a separação ou para a combinação de informação podem ser caracterizados no domínio do tempo, em que amostras de sinais são separadas ou combinadas em seqüência, como no domínio da freqüência, separando-se os sinais em bandas de freqüência. Muitas vezes necessitamos realizar alguma operação sobre um sinal em particular, ou em diversos sinais que, mais à frente no sistema, deverão ser recombinaados ou redistribuídos. As operações envolvidas para se extrair várias informações a partir de um único sinal são denominadas operações de análise, enquanto que as operações envolvidas para se obter um único sinal a partir de várias informações são chamadas operações de síntese.

Bancos de filtros realizam operações de análise e de síntese, resultando quase sempre em uma estrutura em que os vários filtros do sistema apresentam características parecidas entre si, usualmente para dar uniformidade em relação a alguns parâmetros do sistema (por exemplo, mesmo tamanho de banda ou mesmo atraso de grupo entre os canais). Casos particulares de bancos de filtros possuem, por

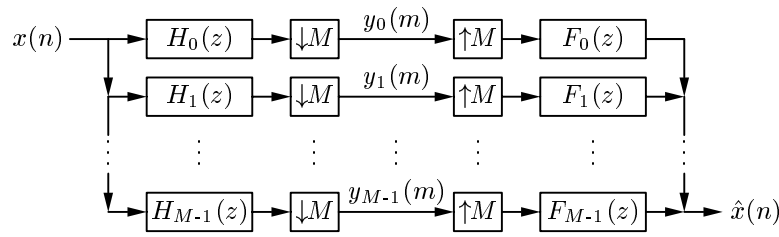


Figura 5.1: Diagrama básico de um banco de filtros.

exemplo, características de reconstrução perfeita do sinal de saída do sistema, facilidade no projeto dos filtros do banco, além de serem relacionados, isto é, o banco de análise pode ser sempre derivado a partir do banco de síntese e vice-versa.

Neste capítulo, estaremos especialmente tratando dos bancos de filtros modulados por cosseno (CMFBs – *cosine-modulated filter banks*), pois estes filtros podem ser projetados de modo a atender a todas as propriedades mencionadas, além de contarem com estruturas de implementação eficientes. Na seção 5.2, estaremos estudando os bancos de filtros criticamente (maximamente) decimados. Na seção 5.3 estaremos tratando dos bancos de filtros modulados por cossenos. Na seção 5.4, analisaremos a estrutura eficiente de um banco de filtros deste tipo, enquanto que nas seções 5.5 e 5.6 indicaremos as condições de reconstrução perfeita e próxima da perfeita, e na seção 5.7 mostraremos como se realiza uma otimização seguindo os critérios de reconstrução perfeita ou próxima da perfeita.

## 5.2 Banco de filtros criticamente decimados

Na Figura 5.1, temos, de acordo com a teoria de processamento multitaxas, o diagrama básico de um banco de filtros com canais uniformes (isto é, mesma taxa de amostragem para cada canal) e criticamente decimados.

Nesta Figura, o sinal de entrada  $x(n)$  é decomposto em  $M$  canais. Os filtros  $H_0(z)$ ,  $H_1(z)$ ,  $\dots$ ,  $H_{M-1}(z)$  são filtros de análise, e são os responsáveis, juntamente com os decimadores, por decompor o sinal de entrada em várias bandas de frequência. Os filtros  $F_0(z)$ ,  $F_1(z)$ ,  $\dots$ ,  $F_{M-1}(z)$ , ao contrário, operam sobre vários sinais e são responsáveis, junto com os interpoladores, por recombinar os vários sinais em um único sinal. Os filtros na verdade preparam os sinais, filtrando em bandas, evitando assim os efeitos de sobreposição que ocorrem quando se opera a



decimação e removendo as repetições de espectro no caso de interpolação do sinal. O processo é dito criticamente decimado se o sinal de entrada é decomposto em vários sinais contendo informações não redundantes, e isso ocorre se a taxa de decimação é igual ao número de sub-bandas, como veremos a seguir.

Uma cópia do sinal de entrada é operada pelos vários filtros de análise, que retiram informações do sinal de entrada. Basicamente, estes filtros são passa-faixa (com exceção do primeiro, que é passa-baixa, e do último, que é passa-altas), e separam o espectro do sinal de entrada em várias bandas. Supondo que estes filtros sejam ideais (janelas quadradas no domínio da frequência), não haveria interferências (efeito de sobreposição) entre as bandas do sinal. Os sinais, ao passarem pelos decimadores, com fator de  $M$ , alargam seus espectros em relação à nova taxa de amostragem. Assim, cada componente do sinal  $y_0(m), y_1(m), \dots, y_{M-1}(m)$  possui informações distintas a respeito do sinal  $x(n)$ . Note, entretanto, que como o sinal foi decimado, estes sinais  $y_i(m)$  estarão em taxa mais baixa que o sinal de entrada  $x(n)$ , e portanto, cada canal separadamente possuirá uma quantidade de informação menor. Igualmente, para a reconstrução do sinal, os vários canais possuem um interpolador de fator  $M$ , seguidos de filtros de síntese que também são passa-faixas, cuja função é anular as repetições de espectro devidas ao interpolador, e os sinais podem então ser combinados (somados) de forma a compor o sinal de saída  $\hat{x}(n)$  na mesma taxa do sinal de entrada. Se os filtros de análise e de síntese forem ideais, então o sinal na saída do sistema será idêntico ao sinal na entrada do sistema, a menos de um atraso inerente, devido aos atrasos dos filtros, e, neste caso, o sistema possui reconstrução perfeita (PR – *perfect reconstruction*).

Embora a reconstrução perfeita aparentemente necessite de filtros ideais, é possível alcançá-la utilizando filtros não-ideais, e portanto, realizáveis, ao menos com sistemas de precisão infinita. Para isto, utilizamos parte da interferência entre os canais para reconstruir perfeitamente o sinal de cada canal em particular, como veremos na próxima seção. Com sistemas de precisão finita, o sinal reconstruído sempre apresenta um nível de ruído de quantização, de modo que muitas vezes não é necessário atingir a reconstrução perfeita para que o sistema seja eficiente. No caso em que interferências entre canais e de transmissão em cada sub-banda, que são a fonte da imperfeição da reconstrução, serem comparáveis com o nível de ruído

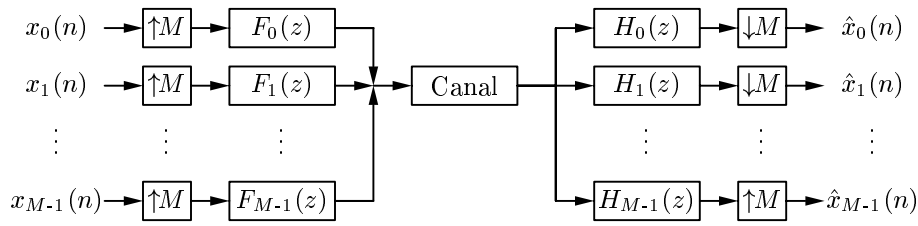


Figura 5.2: O sistema transmultiplexador.

de quantização, podemos dizer que o sistema apresenta reconstrução próxima da perfeita, ou quase-perfeita (NPR – *nearly-perfect reconstruction*).

De modo alternativo, podemos ter um banco de filtros com a estrutura tipo transmultiplexador (TMUX) da Figura 5.2. O TMUX é uma aplicação clássica de banco de filtros, onde a idéia é combinar vários sinais provenientes de fontes independentes em um único sinal que será transmitido em um meio particular. Uma vez recebido, o sinal é decomposto, obtendo-se a reconstrução dos sinais gerados por cada fonte. Este sistema pode ser descrito pela seguinte equação no domínio  $Z$

$$\hat{\mathbf{X}}(z^M) = \mathbf{T}(z^M)\mathbf{X}(z^M) \quad (5.1)$$

onde

$$\hat{\mathbf{X}}(z^M) = \begin{bmatrix} \hat{X}_0(z^M) & \hat{X}_1(z^M) & \dots & \hat{X}_{M-1}(z^M) \end{bmatrix}^T \quad (5.2)$$

$$\mathbf{X}(z^M) = \begin{bmatrix} X_0(z^M) & X_1(z^M) & \dots & X_{M-1}(z^M) \end{bmatrix}^T \quad (5.3)$$

e onde a matriz  $\mathbf{T}(z^M)$  representa as funções de transferência de cada entrada para cada saída do sistema TMUX da Figura 5.2, sendo dada por

$$\mathbf{T}(z^M) = \begin{bmatrix} [T(z^M)]_{0,0} & [T(z^M)]_{0,1} & \dots & [T(z^M)]_{0,M-1} \\ [T(z^M)]_{1,0} & [T(z^M)]_{1,1} & \dots & [T(z^M)]_{1,M-1} \\ \vdots & \vdots & \ddots & \vdots \\ [T(z^M)]_{M-1,0} & [T(z^M)]_{M-1,1} & \dots & [T(z^M)]_{M-1,M-1} \end{bmatrix} \quad (5.4)$$

onde cada elemento é descrito como

$$[T(z^M)]_{k,l} = \frac{1}{M} \sum_{i=0}^{M-1} H_k(z e^{-j\frac{2\pi}{M}i}) F_l(z e^{-j\frac{2\pi}{M}i}) \quad (5.5)$$

onde  $k = 0, 1, \dots, M-1$  e  $l = 0, 1, \dots, M-1$ . Para que o banco de filtros possua uma boa eficiência, é necessário que as funções de transferência da diagonal principal desta matriz sejam aproximadas de uma função de atraso  $z^{-N}$ , e que as demais funções

de transferência sejam aproximadamente nulas, pois assim cada sinal de saída  $\hat{x}_i(n)$  possuirá relação apenas com o sinal de entrada  $x_i(n)$  correspondente, para  $i = 0, 1, \dots, M - 1$ , isto é, sem a interferência dos sinais nos demais canais. Portanto, filtros com reconstrução perfeita apresentam atrasos na diagonal principal desta matriz, enquanto que os outros elementos serão nulos. A partir da equação (5.5), é possível chegar aos estimadores do nível de interferência entre símbolos e do nível de interferência entre portadoras (ISI – *inter-symbol interference* e ICI – *inter-carrier interference*). As funções de estimação destes valores são:

$$\text{ISI} = \max_k \left( \sum_n (x(n) - \hat{x}(n))^2 \right) \Big|_{x(n)=\delta(n)} \quad (5.6)$$

$$\text{ICI} = \max_k \left( \sum_{l=0, l \neq k}^{M-1} |[T(e^{j\omega M})]_{k,l}|^2 \right) \quad (5.7)$$

para  $k = 0, 1, \dots, M - 1$ . Ou seja, para o nível de ISI, verificamos o erro quadrático entre o sinal de entrada e o sinal de saída do sistema, tipicamente inserindo um sinal de teste, que, para o estimador acima, é um impulso discreto. Já o ICI mede diretamente a quantidade de interferência entre os vários canais (ou seja, os termos que não pertencem à diagonal principal da matriz de transferência  $\mathbf{T}(z^M)$ , que são chamados também de termos de *cross-talk* entre canais). Note também que estas funções estimam os valores de pior caso das interferências, isto é, o valor máximo global das funções.

### 5.2.1 Construção da resposta em frequência do banco de filtros

A curva de resposta em frequência (magnitude) típica de um banco de filtros pode ser vista na Figura 5.3. Nesta Figura, ilustramos um banco de filtros com  $M = 8$  canais e, portanto, cada canal possuirá largura de banda igual a  $\frac{\pi}{8}$ .

De acordo com a equação (5.4), os elementos da diagonal principal são dados por:

$$[T(z^M)]_{k,k} = \sum_{i=0}^{M-1} H_k(z e^{-j\frac{2\pi}{M}i}) F_k(z e^{-j\frac{2\pi}{M}i}), \quad k = 0, 1, \dots, M - 1 \quad (5.8)$$

e, portanto, a sua magnitude na resposta em frequência será:

$$|[T(e^{j\omega M})]_{k,k}| = \left| \sum_{i=0}^{M-1} H_k(e^{j(\omega - \frac{2\pi}{M}i)}) F_k(e^{j(\omega - \frac{2\pi}{M}i)}) \right| \quad (5.9)$$

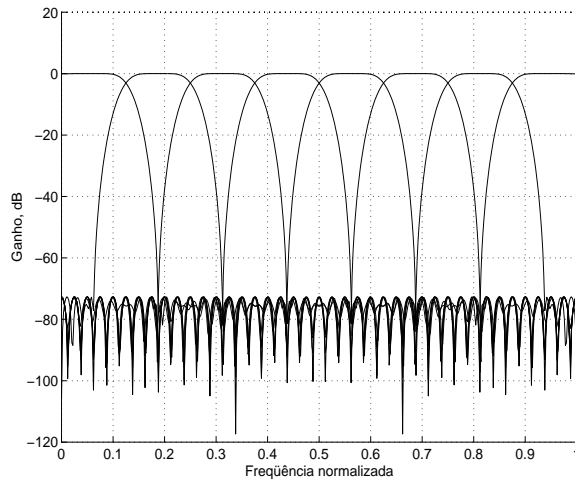


Figura 5.3: Exemplo típico das respostas em magnitude dos filtros de análise e de síntese de um CMFB com  $M = 8$  canais.

ou seja, é o somatório da magnitude das versões deslocadas por  $\frac{2\pi}{M}i$ . Uma vez que as faixas de transição de  $[T(e^{j\omega M})]_{k,k}$  não são as de um filtro ideal, é necessário que haja sobreposição com as versões vizinhas, para que a resposta, quando somada, resulte em ganho unitário. Neste caso, a diagonal principal resulta em funções de atraso. Assim, será necessário que a função de transferência relativa ao  $k$ -ésimo canal seja um filtro passa-faixa centrado em  $\frac{(2k+1)}{2M}\pi$ , que obedeça à seguinte limitação:

$$|[T(e^{j\omega})]_{k,k}| \approx \frac{1}{2}, \quad \omega = \frac{(2k+1)\pm 1}{2M}\pi \quad (5.10)$$

para garantir que não haverá erros muito altos próximo das bandas de transição dos filtros do banco. Com isto, e supondo que os filtros de análise e de síntese possuam *ripples* bem abaixo do ruído de quantização, pode-se chegar à reconstrução próxima da perfeita. Por outro lado, para garantir a reconstrução perfeita dos sinais em todos os canais, é necessário que todas as transferências entre a entrada e a saída resultem exatamente em uma resposta contínua de ganho unitário. Para evitar ter que utilizar  $M$  restrições (uma para cada canal), este problema pode ser simplificado se tomarmos como referência a Figura 5.1. Se a relação de entrada/saída deste sistema for igual a um, em módulo, então também as transferências das entradas para as saídas da Figura 5.2 serão unitárias. Definindo a relação entre a entrada  $x(n)$

e a saída  $\hat{x}(n)$  da Figura 5.1, teremos:

$$\hat{X}(z) = T_0(z)X(z) + \sum_{i=1}^{M-1} T_i(z)X(ze^{-j\frac{2\pi}{M}i}) \quad (5.11)$$

onde

$$T_0(z) = \frac{1}{M} \sum_{k=0}^{M-1} F_k(z)H_k(z) \quad (5.12)$$

$$T_i(z) = \frac{1}{M} \sum_{k=0}^{M-1} F_k(z)H_k(ze^{-j\frac{2\pi}{M}i}) \quad (5.13)$$

O termo  $T_0(z)$  é chamado de função de transferência de distorção, e mede o quanto o sinal de saída está próximo do sinal de entrada, enquanto que os demais termos  $T_i(z)$  definem a relação de superposição entre o sinal original,  $X(z)$  e as componentes de superposição,  $X(ze^{-j\frac{2\pi}{M}i})$ , originadas devido às operações de decimação e de interpolação. Como condição para a reconstrução perfeita, deveremos, portanto, ter:

$$|T_0(e^{j\omega})| = 1, \quad \forall \omega \in [0, \pi] \quad (5.14)$$

ou equivalentemente, se a reconstrução deve ser próxima da perfeita, teremos:

$$1 - \delta_1 \leq |T_0(e^{j\omega})| \leq 1 + \delta_1, \quad \forall \omega \in [0, \pi] \quad (5.15)$$

onde  $\delta_1$  indica o máximo de erro permitido para esta função  $|T_0(e^{j\omega})|$ . Também desejamos reduzir as interferências entre canais. A segunda condição para a reconstrução perfeita é

$$|T_k(e^{j\omega})| = 0, \quad k = 1, 2, \dots, M-1 \quad \forall \omega \in [0, \pi] \quad (5.16)$$

e, para a reconstrução próxima da perfeita, teremos:

$$|T_k(e^{j\omega})| \leq \delta_2, \quad k = 1, 2, \dots, M-1 \quad \forall \omega \in [0, \pi] \quad (5.17)$$

e, novamente, o parâmetro  $\delta_2$  indica o máximo de erro cometido na aproximação da reconstrução perfeita.

### 5.3 Banco de filtros modulados por cosseno

O método de projeto de CMFBs utiliza um filtro protótipo (usualmente passa-baixas) e, a partir deste filtro, gera várias modulações em cosseno, fazendo com que a resposta em magnitude do filtro seja deslocada na frequência, gerando

assim os vários filtros passa-faixa do banco a partir de um único filtro [13]. A grande vantagem deste filtro é que existe uma estrutura eficiente que implementa simultâneamente todos os filtros de análise ou de síntese com o mesmo número de coeficientes do filtro protótipo, e mais uma etapa de modulação que é realizada através de algoritmo computacional rápido [14]-[15]. Se o filtro protótipo é do tipo:

$$H_p(z) = \sum_{n=0}^N h_p(n)z^{-n} \quad (5.18)$$

então os filtros de análise e de síntese CMFB são dados, respectivamente, por:

$$h_m(n) = 2h_p(n) \cos \left[ \frac{(2k+1)\pi}{2M} \left( n - \frac{N}{2} \right) + (-1)^m \frac{\pi}{4} \right] \quad (5.19)$$

$$f_m(n) = 2h_p(n) \cos \left[ \frac{(2k+1)\pi}{2M} \left( n - \frac{N}{2} \right) - (-1)^m \frac{\pi}{4} \right] \quad (5.20)$$

onde  $m = 0, 1, \dots, M-1$ . A magnitude da resposta em frequência destes filtros será a soma das versões deslocadas do filtro protótipo, uma para o lado positivo e outra para o lado negativo, do eixo de frequências. Mais ainda, as magnitudes das respostas dos filtros de análise e de síntese de cada canal são idênticas, já que o fator  $\pm(-1)^m \frac{\pi}{4}$  só interfere na fase das respostas. Portanto, a magnitude da função  $[T(e^{j\omega})]_{k,k}$  fica:

$$|[T(e^{j\omega})]_{k,k}| = |H_k(e^{j\omega})F_k(e^{j\omega})| = |H_k(e^{j\omega})|^2 = |F_k(e^{j\omega})|^2 \quad (5.21)$$

e, portanto, para que a equação (5.10) seja satisfeita, é necessário que:

$$|H_k(e^{j\omega})| = |F_k(e^{j\omega})| = \frac{1}{\sqrt{2}} = -3\text{dB}, \quad \omega = \frac{(2k+1)\pm 1}{2M}\pi \quad (5.22)$$

e, conseqüentemente,

$$|H_p(e^{j\omega})| = -3\text{dB}, \quad \omega = \frac{1}{2M}\pi \quad (5.23)$$

Portanto, a frequência de queda de 3dB do filtro protótipo deve estar situada em aproximadamente  $\frac{1}{2M}\pi$  para garantir uma reconstrução próxima da perfeita. As modulações em cosseno levam o centro do filtro protótipo (que é em 0 rad/s) para um dos valores  $\frac{(2k+1)\pi}{2M}$  rad/s, sendo  $k$  o número do canal. Também é comum se controlar a quantidade de superposição entre as bandas dos canais, definindo

$$\omega_s = \frac{(1+\rho)}{2M}\pi \quad (5.24)$$

para o filtro protótipo, onde  $\rho$  é conhecido como o fator de sobreposição, ou fator de *roll-off* do filtro protótipo. Para valores de  $\rho$  pequenos, haverá pouca sobreposição entre os canais, o que significa uma resposta mais plana da banda passante de cada canal, desde que, em contrapartida, haja um alto número de coeficientes para implementar o filtro. Caso contrário, se o fator  $\rho$  for alto, haverá maior distorção nos extremos da banda passante de cada canal, e menor número de coeficientes no filtro protótipo. Contudo, em ambos os casos, a reconstrução perfeita pode ser alcançada.

## 5.4 Estrutura eficiente do CMFB

Uma vez projetado o filtro protótipo de um CMFB, é possível realizar a sua decomposição polifásica de modo a tornar eficiente a estrutura de modulação, isto é, sem necessitar implementar os vários filtros de análise ou de síntese separadamente. Se o filtro  $H_p(z)$  possuir ordem  $N = 2KM - 1$ , sendo  $K$  um valor inteiro positivo qualquer, então o filtro protótipo da equação (5.18) pode ser descrito em termos de  $2M$  componentes polifásicas:

$$H_p(z) = \sum_{k=0}^{K-1} \sum_{j=0}^{2M-1} h_p(2kM + j) z^{-(2kM+j)} = \sum_{j=0}^{2M-1} z^{-j} E_j(z^{2M}) \quad (5.25)$$

onde cada componente polifásica do filtro protótipo  $H_p(z)$  será dada por

$$E_j(z) = \sum_{k=0}^{K-1} h_p(2kM + j) z^{-k} \quad (5.26)$$

para  $j = 0, 1, \dots, 2M - 1$ . O fato de se decompor o filtro em  $2M$  componentes polifásicas, resultando que  $N = 2KM - 1$ , é devido ao fato de que a modulação em cosseno se repete em  $n = 2kM$ , isto é,

$$\cos \left\{ \frac{(2m+1)\pi}{2M} \left[ (n+2kM) - \frac{N}{2} \right] + \phi \right\} = (-1)^k \cos \left[ \frac{(2m+1)\pi}{2M} \left( n - \frac{N}{2} \right) + \phi \right] \quad (5.27)$$

o que torna a implementação eficiente, como será visto mais adiante. Nota-se que se o filtro protótipo possuir ordem diferente de  $2KM - 1$ , será possível ainda assim obter  $2M$  decomposições polifásicas, bastando para isto inserir zeros como coeficientes do filtro protótipo até se obter o número de coeficientes múltiplo de  $2M$ .

Uma vez que o filtro protótipo seja decomposto em componentes polifásicas, os filtros de análise da equação (5.19) serão dados por

$$\begin{aligned}
H_m(z) &= \sum_{n=0}^{N-1} h_m(n) z^{-n} \\
&= \sum_{n=0}^{2KM-1} c_{m,n} h_p(n) z^{-n} \\
&= \sum_{k=0}^{K-1} \sum_{j=0}^{2M-1} c_{m,2kM+j} h_p(2kM+j) z^{-(2kM+j)}
\end{aligned} \tag{5.28}$$

onde  $c_{m,n}$  é a função cosseno descrita na equação (5.19). Assim, podemos escrever  $H_m(z)$  em termos das componentes polifásicas de  $H_p(z)$  utilizando a relação (5.27):

$$\begin{aligned}
H_m(z) &= \sum_{j=0}^{2M-1} c_{m,j} z^{-j} \sum_{k=0}^{K-1} (-1)^k h_p(2kM+j) z^{-2kM} \\
&= \sum_{j=0}^{2M-1} c_{m,j} z^{-j} E_j(-z^{2M})
\end{aligned} \tag{5.29}$$

que pode ser escrita em notação matricial na forma:

$$\mathbf{e}(z) = \begin{bmatrix} H_0(z) \\ H_1(z) \\ \vdots \\ H_{M-1}(z) \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 \end{bmatrix} \begin{bmatrix} E_0(-z^{2M}) \\ z^{-1} E_1(-z^{2M}) \\ \vdots \\ z^{-(2M-1)} E_{2M-1}(-z^{2M}) \end{bmatrix} \tag{5.30}$$

onde  $\mathbf{C}_1$  e  $\mathbf{C}_2$  são matrizes  $M \times M$  cujos elementos  $(m, j)$  são  $c_{m,j}$  e  $c_{m,j+M}$ , respectivamente, para  $m = 0, 1, \dots, M-1$  e  $j = 0, 1, \dots, M-1$ .

A equação (5.30) nos leva diretamente à estrutura da Figura 5.4, em que temos a implementação das componentes polifásicas do filtro protótipo em cascata com uma matriz DCT-IV.

Na equação (5.30), podemos expressar  $\mathbf{C}_1$  e  $\mathbf{C}_2$  como sendo

$$\mathbf{C}_1 = \sqrt{M} (-1)^{\frac{K}{2}} \mathbf{C}^{\text{IV}} (\mathbf{I} - \mathbf{J}) \tag{5.31}$$

$$\mathbf{C}_2 = -\sqrt{M} (-1)^{\frac{K}{2}} \mathbf{C}^{\text{IV}} (\mathbf{I} + \mathbf{J}) \tag{5.32}$$

para  $K$  par, e

$$\mathbf{C}_1 = \sqrt{M} (-1)^{\frac{K-1}{2}} \mathbf{C}^{\text{IV}} (\mathbf{I} + \mathbf{J}) \tag{5.33}$$

$$\mathbf{C}_2 = \sqrt{M} (-1)^{\frac{K-1}{2}} \mathbf{C}^{\text{IV}} (\mathbf{I} - \mathbf{J}) \tag{5.34}$$



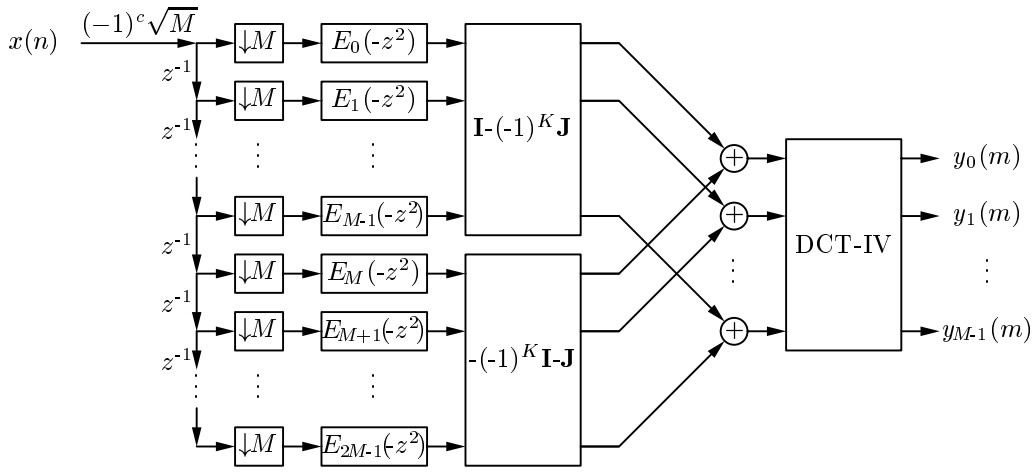


Figura 5.4: A estrutura eficiente de implementação do banco de análise de um CMFB.

para  $K$  ímpar, onde

$$\{\mathbf{C}^{\text{IV}}\}_{m,n} = \sqrt{\frac{2}{M}} \cos \left[ \frac{2m+1}{2M} \left( n + \frac{1}{2} \right) \pi \right] \quad (5.35)$$

As equações (5.31) a (5.34) podem ser convenientemente descritas na forma

$$\mathbf{C}_1 = \sqrt{M} (-1)^c \mathbf{C}^{\text{IV}} (\mathbf{I} - (-1)^K \mathbf{J}) \quad (5.36)$$

$$\mathbf{C}_2 = \sqrt{M} (-1)^c \mathbf{C}^{\text{IV}} (-(-1)^K \mathbf{I} - \mathbf{J}) \quad (5.37)$$

onde  $c = \lfloor \frac{K}{2} \rfloor$  é o maior valor inteiro menor ou igual a  $K/2$ . O fator multiplicativo  $(-1)^c \sqrt{M}$  pode ser levado para a entrada do sistema, resultando assim na estrutura da Figura 5.4, que é a estrutura para os filtros de análise. O mesmo desenvolvimento pode ser realizado com os filtros de síntese, resultando em uma estrutura análoga para o estágio transmissor do TMUX.

A complexidade computacional de um CMFB pode ser calculada como sendo o número de operações necessárias para se gerar uma amostra  $x(n)$ , que é a saída do banco de filtros de síntese. Tomamos o banco de síntese pela facilidade de visualização na geração do sinal  $x(n)$ . O banco de síntese é a operação inversa da estrutura do banco de análise, e está representado na Figura 5.5.

Como podemos observar por esta Figura, são necessárias  $K$  multiplicações por cada componente polifásica, para gerar as entradas da linha de atraso, mais as multiplicações da etapa moduladora e da constante final, que definimos como  $\alpha$ . O interpolador em  $M$  faz com que sejam inseridos  $M - 1$  valores nulos que são

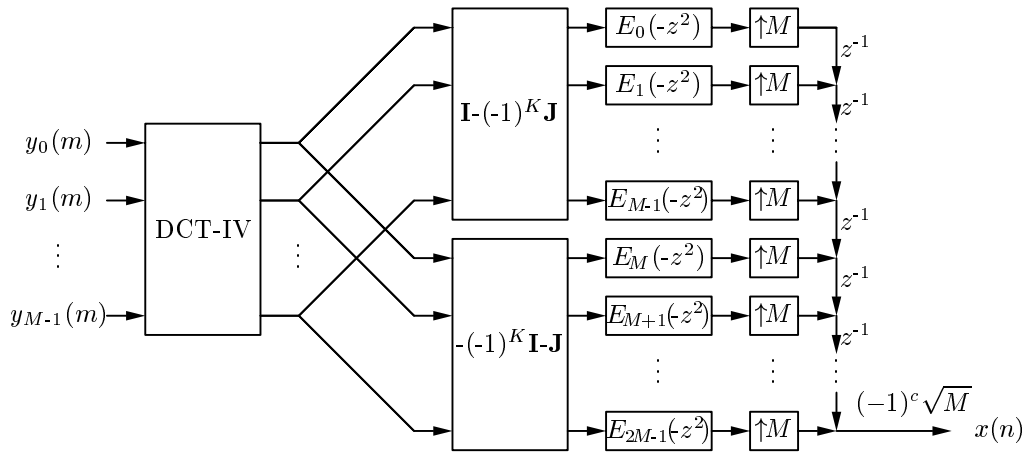


Figura 5.5: A estrutura eficiente de implementação do banco de síntese de um CMFB.

utilizados na composição do sinal na linha de atrasos. Portanto, a complexidade computacional será<sup>1</sup>

$$\mathcal{C} = \frac{2MK}{M} + \alpha = 2K + \alpha \quad (5.38)$$

## 5.5 Reconstrução perfeita

A reconstrução perfeita pode ser obtida de acordo com as restrições das equações (5.14) e (5.16) [13]. Utilizando estas relações, é possível realizar um filtro protótipo simétrico cuja resposta impulsiva seja obtida através de polinômios com termos senos e cossenos, o que garantirá que para qualquer argumento nestas funções, o banco apresentará reconstrução perfeita. O número de variáveis a serem otimizadas será  $KM$ , ou seja, metade do número de coeficientes do filtro protótipo (uma vez que este filtro é simétrico). A reconstrução perfeita não é utilizada na implementação de circuitos, uma vez que há sempre ruído de quantização. Entretanto, um dos métodos para se chegar à reconstrução próxima da perfeita é primeiro obter

---

<sup>1</sup>A complexidade computacional também pode ser calculada pelo filtro de análise, da seguinte forma: para gerar um sinal na saída de um dos canais, são necessárias  $2K$  multiplicações, sendo metade destas devidas à componente polifásica da matriz superior e a outra metade devida à componente polifásica da matriz inferior, mais o valor de  $\alpha$ , ou seja,  $\mathcal{C} = 2K + \alpha$ . Note que nos cálculos apresentados de redução de complexidade, não estamos levando em conta que existe simetria nas componentes polifásicas no banco de filtros.

uma solução semi-ótima para a reconstrução perfeita, para depois ir caminhando em sentido contrário, isto é, afastando-se do ótimo e chegando ao limite do nível de ruído de quantização especificado. A vantagem de se procurar uma solução cuja reconstrução que não seja perfeita, é aliviar as restrições, possibilitando o aumento da atenuação na banda de rejeição dos filtros do banco. Com isto, melhoramos a separação entre os canais, principalmente no que se refere a características não-ideais do sistema (sobretudo do canal).

A reconstrução perfeita pode ser obtida, de acordo com [13], definindo-se o filtro simétrico

$$G_p(z) = (M\sqrt{2})H_p(z) \quad (5.39)$$

e, portanto, se

$$g_p(n) = (M\sqrt{2})h_p(N/2 + n) = (M\sqrt{2})h_p(N/2 - n), \quad n = 0, 1, \dots, \frac{N}{2} \quad (5.40)$$

teremos, lembrando que  $N = 2KM - 1$ , que:

$$G_p(z) = z^{-(KM-1/2)} \sum_{n=0}^{KM-1} g_p(n) \left[ z^{(n-1/2)} + z^{-(n-1/2)} \right] \quad (5.41)$$

Com isto, as restrições de reconstrução perfeita resultam diretamente em subconjuntos de restrição para  $g_p(n)$  que são, ainda de acordo com [13], dados por

$$\begin{aligned} \Theta_r = \{ & g_p(r), g_p(M + 1 - r), g_p(M + r), \\ & g_p(2M + 1 - r), g_p(2M + r), g_p(3M + 1 - r), \dots, \\ & g_p((K - 1)M + r), g_p(KM + 1 - r) \}, \quad \text{para } r = 1, 2, \dots, \lfloor M/2 \rfloor \end{aligned} \quad (5.42)$$

Se o número de canais  $M$  for ímpar, as restrições primeiramente resultam que

$$g_p((M + 1)/2) = \sqrt{1/2} \quad (5.43)$$

e

$$g_p((M + 1)/2 + kM) = 0, \quad k = 1, 2, \dots, K - 1 \quad (5.44)$$

Secundariamente, para o subconjunto  $\Theta_r$ , as restrições são escritas em termos de

$$c_{rk} = \cos(\phi_{rk}) \quad (5.45)$$

e

$$s_{rk} = \sin(\phi_{rk}) \quad (5.46)$$

onde  $\phi_{rk}$  para  $k = 1, 2, \dots, K$  são os parâmetros ajustáveis. Assim, é possível escrever as funções polinomiais

$$P_r^{(K)}(z) = \sum_{n=0}^{K-1} p_r(n) z^{-n} \quad (5.47)$$

e

$$Q_r^{(K)}(z) = \sum_{n=0}^{K-1} q_r(n) z^{-n} \quad (5.48)$$

recursivamente como:

$$P_r^{(k)} = s_{rk} P_r^{(k-1)} + c_{rk} z^{-1} Q_r^{(k-1)}(z) \quad (5.49)$$

e

$$Q_r^{(k)} = c_{rk} P_r^{(k-1)} - s_{rk} z^{-1} Q_r^{(k-1)}(z) \quad (5.50)$$

inicializando com:

$$P_r^1(z) = s_{r1} \quad (5.51)$$

e

$$Q_r^1(z) = c_{r1} \quad (5.52)$$

de modo que os coeficientes de  $g(n)$  (e, portanto, os de  $h_p(n)$ ) possam ser obtidos através destas funções polinomiais. Para os valores de  $K$  pares, teremos:

$$g_p(2kM + r) = p_r\left(\frac{K}{2} + k\right), \quad k = 0, 1, \dots, \frac{K}{2} - 1 \quad (5.53)$$

$$g_p((2k + 1)M + r) = q_r\left(\frac{K}{2} + k\right), \quad k = 0, 1, \dots, \frac{K}{2} - 1 \quad (5.54)$$

$$g_p(2kM + 1 - r) = p_r\left(\frac{K}{2} - k\right), \quad k = 1, 2, \dots, \frac{K}{2} \quad (5.55)$$

$$g_p((2k - 1)M + 1 - r) = q_r\left(\frac{K}{2} - k\right), \quad k = 1, 2, \dots, \frac{K}{2} \quad (5.56)$$

e para valores de  $K$  ímpares, teremos:

$$g_p(2kM + r) = q_r\left(\frac{K-1}{2} + k\right), \quad k = 0, 1, \dots, \frac{K-1}{2} \quad (5.57)$$

$$g_p((2k - 1)M + r) = p_r\left(\frac{K-1}{2} + k\right), \quad k = 1, 2, \dots, \frac{K-1}{2} \quad (5.58)$$

$$g_p(2kM + 1 - r) = q_r\left(\frac{K-1}{2} - k\right), \quad k = 1, 2, \dots, \frac{K-1}{2} \quad (5.59)$$

$$g_p((2k + 1)M + 1 - r) = p_r\left(\frac{K-1}{2} - k\right), \quad k = 0, 1, \dots, \frac{K-1}{2} \quad (5.60)$$

Portanto, o conjunto de variáveis a serem ajustadas é formado por ângulos nas funções polinomiais das equações (5.47) e (5.48), ou seja, o vetor

$$\begin{aligned} \Phi = & [\phi_{11}, \phi_{12}, \dots, \phi_{1K}, \phi_{21}, \phi_{22}, \dots, \phi_{2K}, \\ & \dots, \phi_{\lfloor M/2 \rfloor 1}, \phi_{\lfloor M/2 \rfloor 2}, \dots, \phi_{\lfloor M/2 \rfloor K}]^T \end{aligned} \quad (5.61)$$

e, neste caso, a resposta do filtro pode ser escrita como:

$$G_p(\Phi, \omega) = 2 \sum_{n=1}^{KM} g_p(n, \Phi) \cos \left[ \left( n - \frac{1}{2} \right) \omega \right] \quad (5.62)$$

## 5.6 Reconstrução próxima da perfeita

Pode se chegar à reconstrução próxima da perfeita tanto a partir da reconstrução perfeita, utilizando, depois de obtê-la, um algoritmo de otimização não linear para reduzir a energia na banda de rejeição do filtro protótipo, como também é possível chegar a uma solução que não necessite da reconstrução perfeita [17]. Uma das idéias é supor que os *ripples* nas faixas de atenuação e nas bandas passantes se sobreponham pouco, fazendo, por exemplo, com que o *ripple* nas faixas de rejeição seja mais baixo que o *ripple* na banda passante dos filtros do banco. Pode-se tentar também um filtro com características *least-squares*, pois, neste caso, a energia da banda de rejeição diminui ao longo do eixo de freqüência, fazendo com que as sobreposições sejam menores para canais mais distantes, e utilizar um algoritmo WLS-Chebyshev para restringir os picos para os canais próximos (adjacentes), mapeando as respostas destes filtros e verificando as faixas que resultam em maiores valores de ISI e de ICI, tentando reduzir os *ripples* nestas faixas. A vantagem de se fazer isto é evitar algoritmos de programação não-linear.

## 5.7 Otimização do banco de filtros

O problema de otimização de um CMFB traduz-se em se reduzir a energia da banda de rejeição dos vários filtros de análise e de síntese, respeitando, por outro lado, as condições de reconstrução perfeita ou próxima da perfeita vistas nas seções anteriores. Uma vez que os filtros de análise e de síntese são versões moduladas do filtro protótipo, a minimização de energia nas bandas de rejeição dos filtros do banco

implica em minimizar a função de energia na banda de rejeição do filtro protótipo.

A energia pode ser dada no sentido médio quadrático

$$\epsilon = \int_{\omega_s}^{\pi} |H_p(e^{j\omega})|^2 d\omega \quad (5.63)$$

como no sentido de desvio máximo

$$\epsilon = \max_{\omega} |H_p(e^{j\omega})|^2, \quad \omega_s \leq \omega \leq \pi \quad (5.64)$$

e, portanto a minimização da primeira equação leva a uma solução *least-squares*, enquanto que a minimização da segunda equação leva a uma solução *minimax*. O problema de otimização se torna não-linear ao inserirmos as restrições para reconstrução PR ou NPR, e portanto algoritmos mais sofisticados necessitam ser empregados, como é o caso de [13], que usa o algoritmo de Dutta-Vidyasagar para minimizar estas funções. Primeiramente um banco de filtros com reconstrução perfeita é obtido, utilizando um algoritmo de otimização que pode ser tanto o de Dutta-Vidyasagar, como o de Fletcher-Powell, partindo da otimização de um banco com apenas dois canais, e depois aumentando-se o número de canais, introduzindo com isto, novos coeficientes entre os coeficientes já otimizados no filtro protótipo e refazendo o processo, até que se obtenha um banco com o número de canais desejado. Utilizando os coeficientes deste filtro como inicialização no algoritmo de Dutta-Vidyasagar, otimiza-se então uma das funções objetivos acima, juntamente com as restrições de reconstrução próxima da perfeita que devem ser mantidas. A otimização de Dutta-Vidyasagar tenta minimizar a equação:

$$\epsilon = \max_{1 \leq i \leq I} \{f_i(\Phi)\} \quad (5.65)$$

condicionado às restrições

$$\mathcal{G}_j(\Phi) \leq 0, \quad j = 1, 2, \dots, J \quad (5.66)$$

e

$$\mathcal{H}_l(\Phi) = 0, \quad l = 1, 2, \dots, L \quad (5.67)$$

onde, para o caso do CMFB,  $\mathcal{G}_j(\Phi)$  e  $\mathcal{H}_l(\Phi)$  são as funções de restrição  $|T_0(ze^{j\omega})|$  e  $|T_k(ze^{j\omega})|$  de reconstrução perfeita [13]. Introduzindo uma variável de controle  $\psi$ , a

idéia é minimizar

$$\mathcal{P}(\Phi, \psi) = \sum_{i|f_i(\Phi) > \psi} [f_i(\Phi) - \psi]^2 + \sum_{j|\mathcal{G}_j(\Phi) > 0} w_j [\mathcal{G}_j(\Phi)]^2 + \sum_l v_l [\mathcal{H}_l(\Phi)]^2 \quad (5.68)$$

e juntamente, minimizar  $\psi$ . Neste caso, será possível utilizar recursivamente o algoritmo *least-squares* para obter a otimização de  $\mathcal{P}(\Phi, \psi)$ . O problema principal neste tipo de otimização são os vetores de peso  $\mathbf{w} = [w_1, w_2, \dots, w_J]^T$  e  $\mathbf{v} = [v_1, v_2, \dots, v_L]^T$ , além de necessitar de um vetor de partida  $\Phi$  bem próximo do ótimo.

## 5.8 Conclusões

De acordo com o que foi estudado, o método de modulação por cossenos permite que se obtenha uma estrutura eficiente para os bancos análise e de síntese do sistema. Ao utilizarmos a matriz de modulação DCT-IV, foi possível derivar os vários filtros a partir de um único filtro protótipo, bastando para isto decompor os coeficientes filtro em componentes polifásicas. As características de reconstrução perfeita podem ser obtidas utilizando restrições que fazem com que os coeficientes do filtro protótipo sejam multiplicações de termos cossenos, onde os ângulos destes termos são as variáveis a serem otimizadas. Os coeficientes da reconstrução próxima da perfeita podem, portanto, ser obtidos obtendo-se a reconstrução perfeita, e em seguida afastando-se do ponto ótimo mantendo a atenuação dos filtros compatíveis com os níveis de ruído de quantização do sistema.

# Capítulo 6

## Banco de filtros FRM

### 6.1 Introdução

Como já foi visto em capítulos anteriores, o método de projeto de filtros de mascaramento de resposta em frequência (FRM) resulta em redução de coeficientes, e portanto da complexidade (número de operações matemáticas envolvidas para gerar uma amostra do sinal de saída) do filtro. O uso do método FRM pode ser vantajoso em várias aplicações práticas, quando dentro do projeto de um sistema digital, encontramos um filtro de ordem alta devido à faixa de transição estreita. Nos capítulos precedentes ilustramos como realizar este projeto de modo a reduzir ainda mais a complexidade do filtro utilizando o algoritmo WLS-Chebyshev, permitindo ao projetista aplicar restrições em cada faixa de frequência e tornando assim o projeto mais versátil. Neste capítulo, estudaremos a aplicação do método FRM em bancos de filtros modulados por cosseno (CMFBs), derivando uma estrutura eficiente para tal projeto. O projeto combinado FRM com CMFB resulta em um filtro modulado por cosseno de mascaramento de resposta em frequência (CMFRM – *cosine-modulated FRM*) [16].

### 6.2 CMFRM

Para utilizarmos o método FRM combinado com o CMFB, será necessário que o sistema realize a convolução entre os sinais do filtro interpolado e do filtro de máscara, de modo a compor corretamente os sinais na entrada das matrizes



pré-moduladoras. Levando em conta apenas o ramo superior do filtro FRM, a sua resposta pode ser escrita como:

$$R(z) = H_{b_1}^I(z)G_1(z) = H_{b_1}(z^L)G_1(z) \quad (6.1)$$

onde

$$H_{b_1}(z) = \sum_{n=0}^{N_b} h_{b_1}(n)z^{-n} \quad (6.2)$$

e

$$G_1(z) = \sum_{n=0}^{N_m} g_1(n)z^{-n} \quad (6.3)$$

onde  $N_b$  é a ordem do filtro de base,  $h_{b_1}(n)$  são os coeficientes deste filtro,  $N_m$  é a ordem do filtro de máscara positiva e  $g_1(n)$  são os coeficientes do filtro de máscara positiva. Portanto, a resposta no tempo do filtro FRM,  $r(n)$ , será dada por uma convolução discreta das respostas  $h_{b_1}^I(n)$  e  $g_1(n)$ , onde  $h_{b_1}^I(n)$  é a versão interpolada da resposta impulsiva temporal do filtro de base,  $h_{b_1}(n)$ . Com isto, a equação dos filtros de análise (5.28) será

$$\begin{aligned} H_m(z) &= \sum_{n=0}^N h_m(n)z^{-n} = \sum_{n=0}^N c_{m,n}r(n)z^{-n} \\ &= \sum_{n=0}^N c_{m,n}(h_{b_1}^I * g_1)(n)z^{-n} \end{aligned} \quad (6.4)$$

onde  $N$  será a ordem do filtro resultante da convolução. A decomposição polifásica do filtro agora já não é tão óbvia, uma vez que seus elementos dependem dos termos que pertencem aos filtros  $H_{b_1}(z)$  e  $G_1(z)$  pois o filtro protótipo é na verdade a convolução dos dois filtros no domínio do tempo. Utilizando a mesma abordagem, é possível realizar a mesma análise para a parte complementar do FRM, isto é, para o filtro interpolado complementar e para o filtro de máscara negativa.

### 6.3 CMFRM com interpolação múltipla ao número de canais

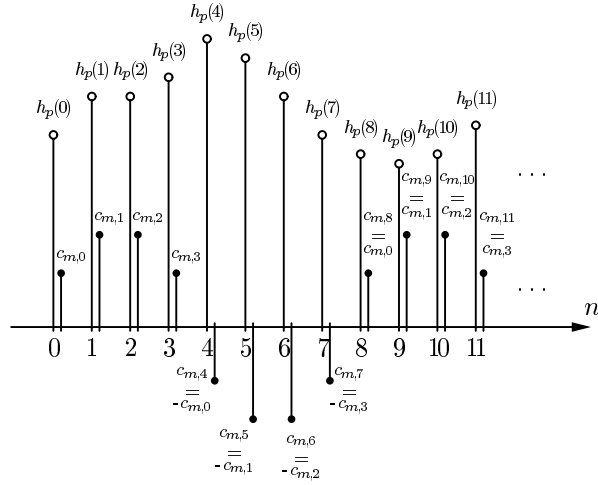
No caso especial em que  $L = 2K_1M$ , sendo  $K_1$  um inteiro positivo, será possível realizar a decomposição polifásica da convolução (6.4). Uma vez que o filtro

interpolado apresentará coeficientes apenas a cada  $2K_1M$  amostras, resulta que os coeficientes de  $h_{b1}^I(n)$  convoluídos por cada coeficiente  $g_1(n)$  são sempre multiplicados por  $c_{m,n+2K_1M}$ , que, utilizando a relação da equação (5.27), resultam na mesma multiplicação por  $c_{m,n}$ .

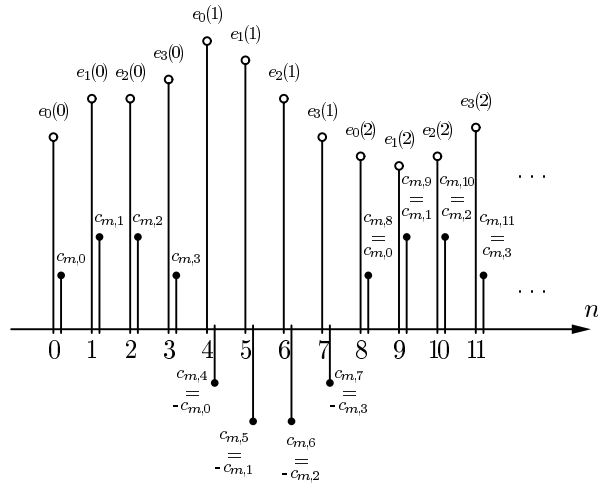
Podemos verificar isto pelas Figuras 6.1 e 6.2. As componentes polifásicas são obtidas justamente para se obter sempre a mesma componente multiplicando um único valor de  $c_{m,n}$ . Como já vimos, uma vez que os valores de  $\pm c_{m,n}$  possuem periodicidade de  $2M$ , isto torna possível a estrutura eficiente da Figura 5.4. No caso em que utilizamos o FRM, a convolução resulta nos valores mostrados na Figura 6.2a. Se observarmos o resultado desta convolução quando separamos o filtro de máscara nas mesmas  $2M$  componentes polifásicas, obteremos o resultado da Figura 6.2b. Note como nesta Figura cada valor obtido é função dos valores do filtro de base multiplicados por valores de uma única componente polifásica do filtro de máscara. Se observarmos atentamente a Figura 6.2b, veremos que os valores obtidos são convoluções entre o filtro base  $h_{b1}(n)$  e cada componente polifásica  $e_k(n)$ , para  $k = 0, 1, \dots, M - 1$ . Isto ocorrerá sempre que  $L$  for múltiplo de  $2M$ .

Uma vez que  $L > M$ , o filtro FRM necessitará obrigatoriamente dos dois ramos (o positivo e o negativo), portanto a estrutura da Figura 5.4 deverá aparecer repetida, com exceção da etapa moduladora (isto é, as matrizes multiplicativas **I** e **J**, e a DCT-IV). Além disto, também devido ao fato que  $L > M$ , o filtro protótipo apresentará fator de *roll-off* pequeno, próximo de zero para  $K_1$  alto. As ordens dos filtros de máscara também deverão ser suficientemente altas para recortar corretamente a resposta em frequência (em magnitude) do filtro interpolado. Portanto, este caso possui maior eficiência possivelmente quando o fator  $K$  do CMFB for alto, ou seja, para filtros protótipos mais seletivos.

Se o filtro  $h_{b1}^I(n)$  apresentar coeficientes apenas em  $n = 2K_1Mi$ , para  $i =$

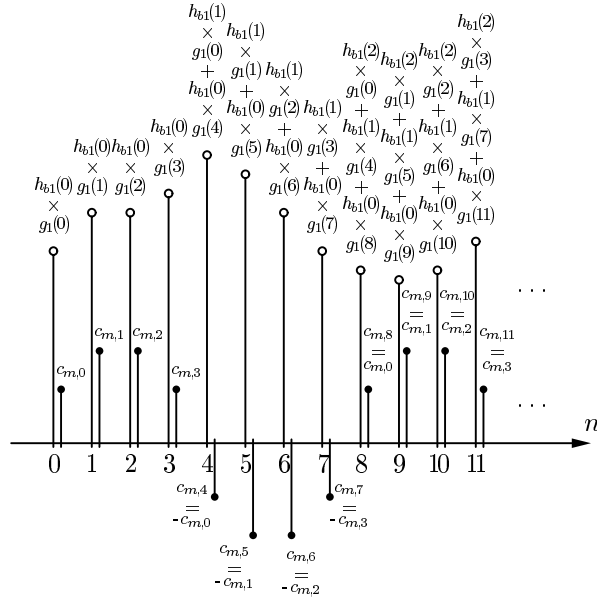


(a) Funções  $h_p(n)$  e  $c_{m,n}$  para um canal  $m$ , mostrando a repetição da função  $c_{m,n}$  com período de  $2M$ .

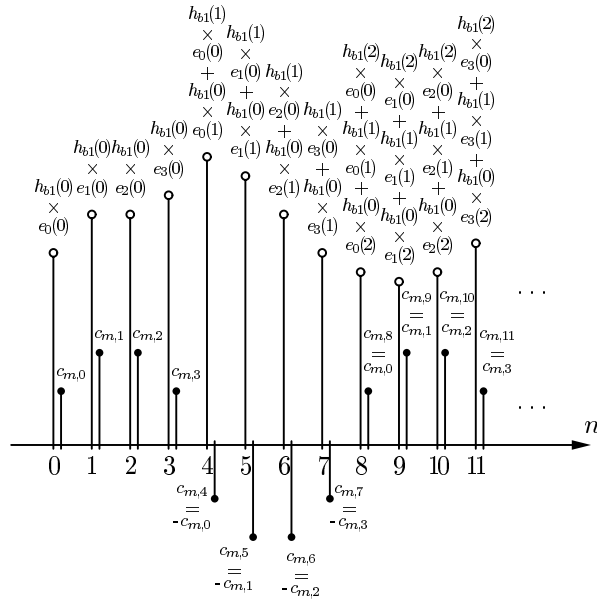


(b) Utilizando as componentes polifásicas do filtro protótipo.

Figura 6.1: Construção da resposta impulsiva de um canal  $m$  no banco de análise de um CMFB. Neste caso, o número de canais é  $M = 2$  e, portanto, a função  $c_{m,n}$  se repete a cada 4 amostras do sinal de entrada.



(a) A convolução do filtro de máscara com um filtro interpolado de fator  $L = 2K_1M$ . Para este exemplo,  $K_1 = 1$ .



(b) A convolução em função da decomposição polifásica do filtro de máscara  $g_1(n)$ .

Figura 6.2: A construção da resposta impulsiva obtida realizando a convolução das respostas impulsivas de dois filtros para o CMFB com  $M = 2$  canais.

$0, 1, \dots$ , então a equação (5.28) pode ser convenientemente decomposta como:

$$\begin{aligned}
H_m(z) &= \sum_{n=0}^N h_m(n) z^{-n} \\
&= \sum_{n=0}^{2K_2M-1} c_{m,n} (h_{b1}^I * g_1)(n) z^{-n} \\
&= \sum_{i=0}^{N_b} \left( h_{b1}(i) z^{-2K_1Mi} \sum_{n=0}^{2K_2M-1} c_{m,n+2K_1Mi} g_1(n) z^{-n} \right)
\end{aligned} \tag{6.5}$$

utilizando a relação entre  $c_{m,n}$  e  $c_{m,n+2kM}$  da equação (5.27), teremos:

$$\begin{aligned}
H_m(z) &= \sum_{i=0}^{N_b} \left( h_{b1}(i) z^{-2K_1Mi} \sum_{n=0}^{2K_2M-1} (-1)^i c_{m,n} g_1(n) z^{-n} \right) \\
&= \left( \sum_{i=0}^{N_b} (-1)^i h_{b1}(i) z^{-Li} \right) \left( \sum_{n=0}^{2K_2M-1} c_{m,n} g_1(n) z^{-n} \right) \\
&= \left( \sum_{i=0}^{N_b} (-1)^i h_{b1}(i) z^{-Li} \right) \left( \sum_{k=0}^{K_2-1} \sum_{j=0}^{2M-1} c_{m,2kM+j} g_1(2kM+j) z^{-(2kM+j)} \right)
\end{aligned} \tag{6.6}$$

e, portanto, em função das  $E_j(z)$  componentes polifásicas de  $G_1(z)$ , teremos:

$$H_m(z) = H_{b1}(-z^L) \sum_{j=0}^{2M-1} c_{m,j} z^{-j} E_j(-z^{2M}) \tag{6.7}$$

para  $L = 2K_1M$ , e, para que o filtro de máscara seja decomposto em  $2M$  componentes polifásicas, a sua ordem será  $N_m = 2K_2M - 1$ , embora seja possível obter-se uma decomposição polifásica para  $N_m$  diferente deste valor, bastando para isso inserir zeros como coeficientes do filtro de máscara, aumentando o seu atraso (e, portanto, sua ordem) até se obter os  $2K_2M$  coeficientes.

O filtro  $H_{b1}(-z^L)$  pré-filtra os sinais de cada componente polifásica, e pode ser colocado após a linha de atrasos e antes dos decimadores, ou na entrada do sistema, evitando assim a sua repetição na estrutura. A estrutura combinada pode ser vista na Figura 6.3.

Esta estrutura serve para os dois ramos do FRM, e como a modulação será a mesma e o filtro  $H_{b2}(z)$  é uma versão complementar de  $H_{b1}(z)$ , a etapa moduladora não necessita ser repetida, e a estrutura final será a da Figura 6.4, onde podemos notas as  $2M$  componentes polifásicas  $E_j(z)$  provenientes da decomposição do filtro de máscara positiva  $G_1(z)$ , e as  $2M$  componentes polifásicas  $P_j(z)$  provenientes da decomposição do filtro de máscara negativa  $G_2(z)$ , onde  $j = 0, 1, \dots, 2M - 1$ .

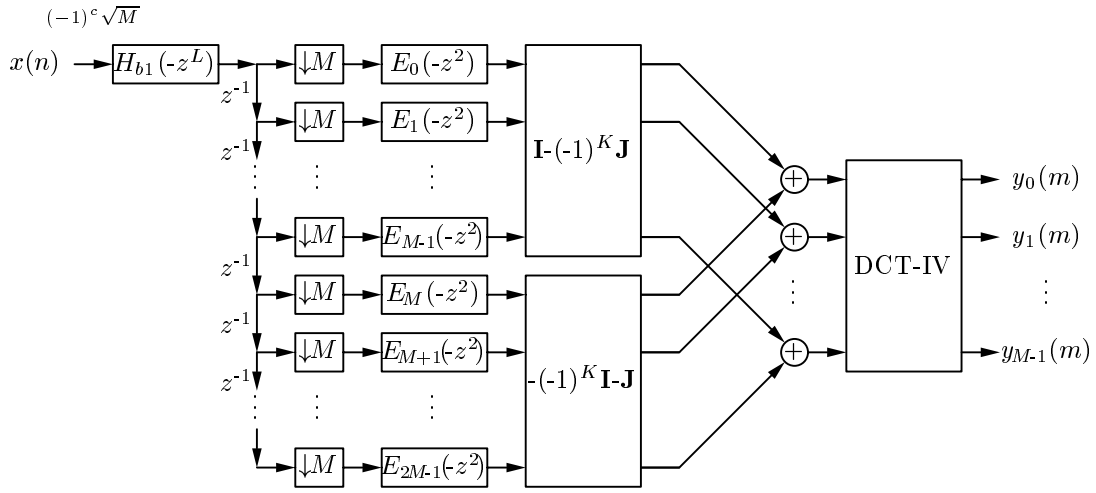


Figura 6.3: A estrutura em cascata (filtro interpolado seguido pelas decomposições polifásicas do filtro de máscara) do CMFRM para o caso  $L = 2K_1M$ .

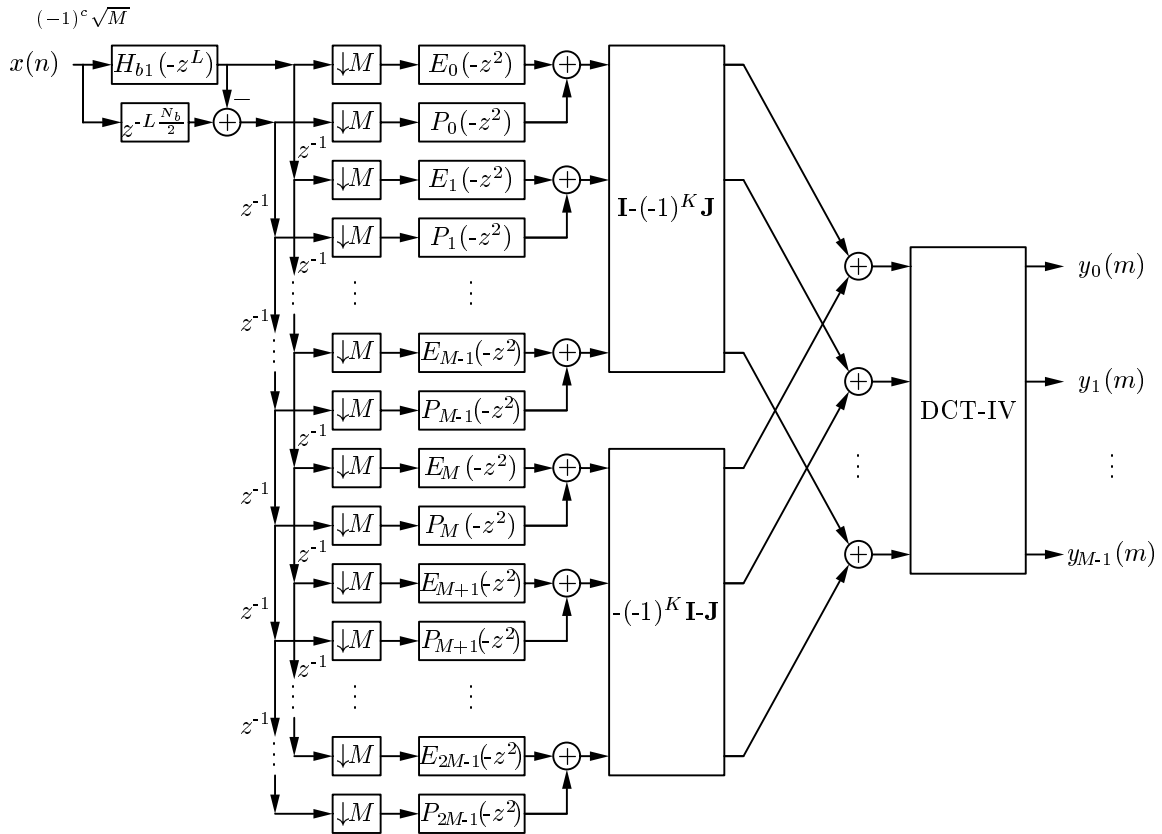


Figura 6.4: A estrutura completa (os dois ramos) do CMFRM para o caso em que  $L = 2K_1M$ .

Os valores de  $K$  nas matrizes pré-moduladoras  $\mathbf{I} - (-1)^K \mathbf{J}$  e  $-(-1)^K \mathbf{I} - \mathbf{J}$  e o valor da constante  $c$  na entrada do banco de filtros permanecerão inalterados, já que o banco de filtros nestes estágios se comporta como um CMFB padrão.

Em termos da complexidade computacional, teremos, para gerar um sinal  $x(n)$  na saída do filtro de síntese, as multiplicações da etapa moduladora e da constante na saída, ou seja,  $\alpha$ , mais  $K_2$  multiplicações em cada componente polifásica, e mais as  $N_b + 1$  multiplicações do filtro interpolado. Para uma estrutura completa (com os dois ramos do FRM), a complexidade computacional será

$$\mathcal{C} = \frac{4MK_2}{M} + N_b + 1 + \alpha = 4K_2 + N_b + 1 + \alpha \quad (6.8)$$

onde supomos que os dois filtros de mascaramento dos ramos positivo e negativo do CMFRM possuem o mesmo número de coeficientes  $2K_2M$ . O número de operações multiplicativas distintas, no banco de filtros, sem supor simetria de coeficientes, será

$$\mathcal{M} = \mathcal{M}_+ + \mathcal{M}_- + \mathcal{M}_b + \beta = 4K_2M + N_b + 1 + \beta \quad (6.9)$$

onde  $\beta$  será o número de multiplicações distintas nas matrizes de modulação, relacionado com  $\alpha$ . Note que, em relação à complexidade computacional do CMFB, dada pela equação (5.38), obtemos a redução apenas para uma combinação de  $K_2$  e de  $N_b$  que seja bem menor que  $K$ , enquanto que a partir da equação (6.9), chegamos a um fator de redução de multiplicadores nos filtros CMFRM idêntico ao fator de redução de um FRM substituindo um filtro FIR na implementação direta. A idéia de se calcular o número de multiplicações distintas é obter com isto o número de variáveis diferenciadas a serem otimizadas. Se o filtro apresentar redução de coeficientes, isto implica que o processo de otimização poderá possivelmente ser simplificado.

### 6.3.1 Análise de resposta em frequência

Se analisarmos o que acontece com a resposta em frequência do filtro protótipo para o caso em que  $L = 2K_1M$ , veremos que este filtro não é realizável. A Figura 6.5 nos permite analisar o filtro interpolado de acordo com as suas repetições de espectro. Uma vez que é necessário que  $\omega_{3\text{dB}} \approx \frac{1}{2M}\pi$ , será difícil obter um banco de filtros com reconstrução próxima da perfeita, pois  $\frac{1}{2M}$  é múltiplo de  $\frac{1}{2K_2M}$ , e, portanto, a

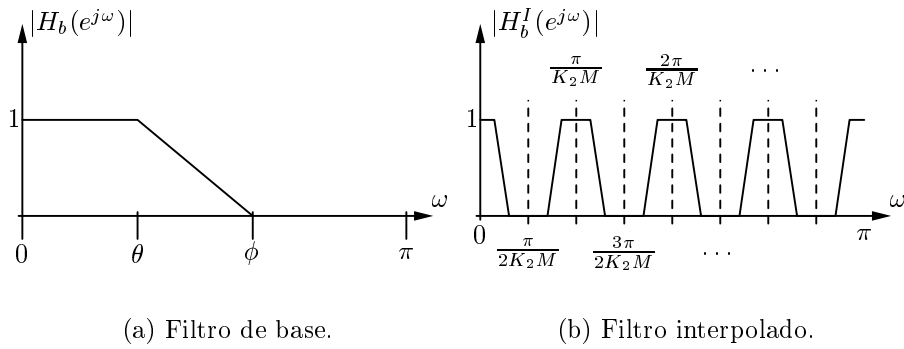


Figura 6.5: A interpolação por  $L = 2K_1M$  faz com que os centros das bandas do filtro interpolado caiam sobre  $\frac{k\pi}{2M}$ , não permitindo realizar a faixa de transição do filtro de base nesta frequência.

freqüência de queda de 3dB ocorre justamente sobre o centro de uma das bandas do filtro interpolado. Esta freqüência corresponde à freqüência 0 ou  $\pi$  da resposta do filtro de base. Portanto, não será possível realizar a faixa de transição no filtro de base. Esta estrutura, porém, pode se aproximar do caso de reconstrução próxima da perfeita se o fator de interpolação for alto, pois neste caso, uma freqüência próxima de zero ou de  $\pi$  no filtro de base tende a se aproximar desta freqüência. Outra possibilidade é que o sistema admita erros muito altos nas bandas passantes dos filtros de análise e de síntese.

## 6.4 CMFRM com amostragem igual ao número de canais

No caso especial em que  $L = M$ , isto é, em que o fator de interpolação do FRM é igual ao número de canais, ainda assim será possível realizar uma estrutura parecida com a da Figura 6.4, se os filtros de máscara apresentarem simetria de coeficientes. Entretanto, a simetria dos filtros de máscara não significa simetria nas decomposições polifásicas. Iremos analisar agora o caso mais simples em que  $K_2 = 1$  (já que neste caso, haverá apenas um coeficiente em cada função polifásica, permitindo a inserção de um sinal no sentido reverso do filtro de máscara). Para facilitar a análise, utilizaremos apenas o ramo superior do FRM, lembrando que para dois ramos a estrutura fica repetida, como no caso da Figura 6.4.



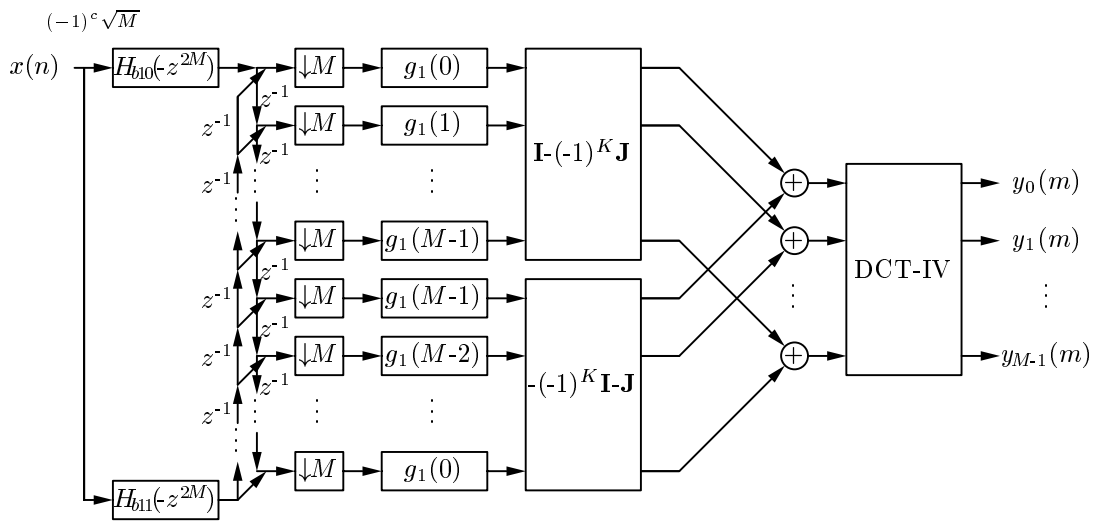


Figura 6.6: A estrutura do CMFRM com  $L = M$  e  $K_2 = 1$ .

Se o filtro  $G_1(z)$  possui simetria, teremos que

$$g_1(n) = g_1(N - n), \quad n = 0, 1, 2, \dots, \frac{N+1}{2} \quad (6.10)$$

sendo que  $N = 2K_2M - 1$ . Assim, se  $K_2 = 1$ , então cada componente polifásica  $E_j(z)$  será

$$E_j(z) = g_1(j), \quad j = 0, 1, 2, \dots, 2M - 1 \quad (6.11)$$

e, portanto,

$$E_j(z) = E_{2M-1-j}(z), \quad j = 0, 1, \dots, M - 1 \quad (6.12)$$

com isto, será possível separar duas componentes de  $H_{b1}^I(z)$  na forma da Figura 6.6.

A complexidade computacional deste filtro é idêntica a do caso anterior, ou seja, é dada pela equação (6.8), se utilizarmos dois ramos com  $K_2 = 1$ . O número de operações multiplicativas também é dado, assim como no caso anterior, pela equação (6.9). A decomposição de  $H_{b1}(-z^L)$  nas componentes  $H_{b10}(-z^{2L})$  e  $H_{b11}(-z^{2L})$ , que dão origem à estrutura da Figura 6.6, será explicada na seção seguinte. Esta decomposição não altera a complexidade computacional, uma vez que as componentes do filtro de base continuam totalizando o mesmo número de multiplicações, e que a operação de máscara continua idêntica ao caso anterior.

## 6.5 CMFRM com amostragem menor que o número de canais

Quando  $L$  for menor que  $M$ , a convolução na equação (6.5) não pode ser descrita pela equação (6.6), uma vez que  $L$  será uma fração de  $2M$ , e portanto os coeficientes de  $H_{b1}(z)$  serão não-nulos a cada  $2K_1Mi$ , para  $i = 0, 1, \dots$  com  $K_1$  fracionário, e portanto, em geral  $c_{m,n+2K_1Mi} \neq c_{m,n}$ . Para facilitar a notação, utilizaremos agora a seguinte relação entre  $L$  e  $K_1$ :

$$L = \frac{M}{K_1} \quad (6.13)$$

para  $K_1 = 1, 2, \dots$ . Neste caso,  $L = \frac{M}{K_1} < M$ , e devemos, portanto, decompor também o filtro interpolado em  $Q$  componentes polifásicas, onde  $Q = 2K_1$ . Assim, cada componente polifásica do filtro interpolado apresentará valores não-nulos apenas a cada  $2M$  amostras, e, portanto, será possível realizar a convolução. Ou seja, de acordo com a equação (6.5), teremos:

$$\begin{aligned} H_m(z) &= \sum_{n=0}^N h_m(n)z^{-n} \\ &= \sum_{n=0}^{2KM-1} c_{m,n}(h_{b1}^I * g_1)(n)z^{-n} \\ &= \sum_{i=0}^{N_b} \left[ h_{b1}(i)z^{-\frac{M}{K_1}i} \sum_{n=0}^{2K_2M-1} c_{m,(n+\frac{M}{K_1}i)}g_1(n)z^{-n} \right] \end{aligned} \quad (6.14)$$

Se a ordem do filtro de base for  $N_b = QK_3 - 1$ , para  $K_3 = 1, 2, \dots$ , e como  $\frac{MQ}{K_1} = 2M$ , poderemos reescrever a equação (6.14) como:

$$\begin{aligned} H_m(z) &= \sum_{k=0}^{K_3-1} \sum_{q=0}^{Q-1} \left( h_{b1}(q+kQ)z^{-\frac{M}{K_1}(q+kQ)} \sum_{n=0}^{2K_2M-1} c_{m,(n+\frac{M}{K_1}(q+kQ))}g_1(n)z^{-n} \right) \\ &= \sum_{k=0}^{K_3-1} \sum_{q=0}^{Q-1} \left( h_{b1}(q+kQ)z^{-\frac{M}{K_1}(q+kQ)} \sum_{n=0}^{2K_2M-1} c_{m,(n+\frac{M}{K_1}q+2kM)}g_1(n)z^{-n} \right) \end{aligned} \quad (6.15)$$

utilizando agora a relação entre  $c_{m,n}$  e  $c_{m,n+2kM}$  da equação (5.27), a equação (6.15) fica:

$$\begin{aligned} H_m(z) &= \sum_{k=0}^{K_3-1} \sum_{q=0}^{Q-1} \left( h_{b1}(q+kQ) z^{-\frac{M}{K_1}(q+kQ)} \sum_{n=0}^{2K_2M-1} (-1)^k c_{m,(n+\frac{M}{K_1}q)} g_1(n) z^{-n} \right) \\ &= \sum_{q=0}^{Q-1} \left\{ \left( \sum_{k=0}^{K_3-1} z^{-Lq} h_{b1}(q+kQ) (-1)^k z^{-kLQ} \right) \left( \sum_{n=0}^{2K_2M-1} c_{m,(n+Lq)} g_1(n) z^{-n} \right) \right\} \end{aligned} \quad (6.16)$$

Definindo as  $q$ -ésimas componentes polifásicas do filtro de base como sendo:

$$H_q(z) = \sum_{k=0}^{K_3-1} h_{b1}(q+kQ) z^{-k} \quad (6.17)$$

e lembrando que  $LQ = \frac{M}{K_1} 2K_1 = 2M$ , a equação (6.16) fica:

$$H_m(z) = \sum_{q=0}^{Q-1} \left\{ z^{-Lq} H_q(-z^{2M}) \sum_{n=0}^{2K_2M-1} c_{m,(n+Lq)} g_1(n) z^{-n} \right\} \quad (6.18)$$

e, portanto, em função das  $E_j(z)$  componentes polifásicas de  $G_1(z)$ , teremos:

$$H_m(z) = \sum_{q=0}^{Q-1} \left\{ z^{-Lq} H_q(-z^{2M}) \sum_{j=0}^{2M-1} c_{m,j+Lq} z^{-j} E_j(-z^{2M}) \right\} \quad (6.19)$$

Note que para o caso em que  $L = M$ , ocorre que  $Q = 2$ , e teremos a decomposição do filtro interpolado em duas componentes, de modo que cada componente agora será descrita como função de  $-z^{2M}$ , e que cada decomposição deve ser processada pelo filtro de máscara e pela etapa moduladora corretamente. No caso especial em que  $L = M$  e  $K_2 = 1$ , aproveitamos o fato de que  $g_1(n)$  é simétrico e obtivemos a estrutura da seção anterior.

Para o caso mais geral, isto é, para  $L = M$  e para  $K_2 > 1$  (ou seja, a ordem das componentes polifásicas do filtro de máscara é maior que zero), ou para  $L < M$ , note que cada componente polifásica do filtro interpolado deve ser multiplicada primeiro pelos mesmos coeficientes de  $g_1(n)$ , e depois as respostas devem ser somadas convenientemente na entrada das matrizes pré-moduladoras. Para evitar a repetição da mesma estrutura da decomposição polifásica do filtro de máscara  $g_1(n)$ , uma estrutura multiplicativa do tipo da Figura 6.7 pode ser aplicada.

Nesta Figura, o filtro  $H(z)$  possui  $N$  coeficientes, e desejamos filtrar  $Q$  sinais  $x_q(n)$  e obter as respectivas saídas  $y_q(n)$  para  $q = 0, 1, \dots, Q - 1$ . Se analisarmos

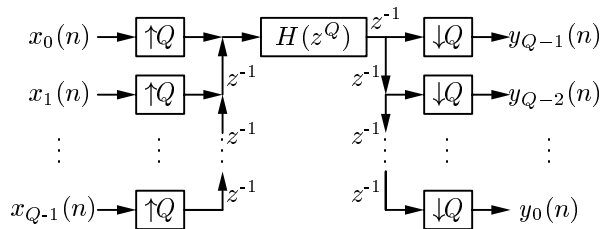


Figura 6.7: Filtragem de vários sinais a partir de um único filtro, utilizando o conceito de sistemas multitasas.

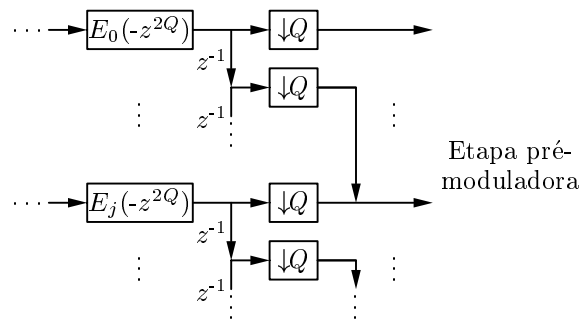
esta Figura, veremos que:

$$Y_q(z) = z^{-1} X_q(z) H(z) \quad (6.20)$$

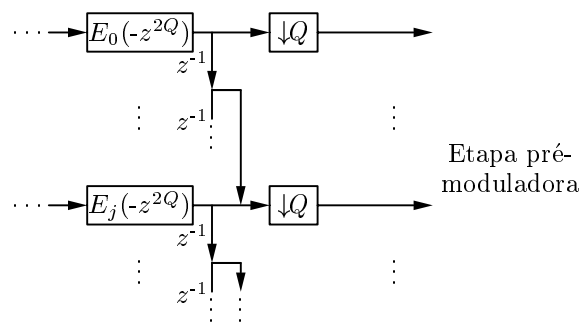
e que cada saída estará defasada de um ciclo em relação à sua entrada. Portanto, utilizar uma estrutura deste tipo é equivalente a introduzir um atraso em todos os sinais do banco de filtros. É possível eliminar o primeiro atraso após o filtro  $H(z^Q)$ , e neste caso a primeira saída agora será  $y_0(n)$  e não estará defasada em relação à sua entrada. Neste caso, é necessário introduzir um atraso extra na entrada desta componente polifásica, ou, se possível, reduzir um atraso de cada uma das outras componentes polifásicas do filtro de base, o que é possível, em certos casos.

Outro ponto importante a ser notado são os decimadores no final desta estrutura. Como os sinais da saída (após os decimadores da Figura 6.7) serão somados a outras componentes no banco de filtros, será possível simplificar a estrutura retirando alguns dos decimadores, como podemos ver na Figura 6.8. Na verdade, poderemos retirar  $Q - 1$  decimadores de cada estrutura deste tipo, e a estrutura final para o banco de análise será a apresentada na Figura 6.9.

Como podemos notar por esta Figura, ocorre primeiro uma decimação em  $M$  e em seguida uma interpolação em  $Q$ . Esta operação pode ser vista como uma chave comutadora com  $Q$  fases, sendo que em cada fase a chave liga uma das saídas das componentes polifásicas do filtro interpolado com a entrada de uma das componentes polifásicas do filtro de máscara. Esta chave pode ter a mesma taxa na entrada e na saída, se  $M = Q$ , assim como pode ter taxa de amostragem de entrada e de saída diferentes entre si, caso  $M \neq Q$ . Se observarmos a estrutura correspondente ao banco de síntese da Figura 6.10, veremos que a complexidade computacional agora



(a) Saída das componentes polifásicas do filtro de máscara.



(b) Retirada dos  $Q - 1$  decimadores de cada componente polifásica.

Figura 6.8: Redução dos decimadores em  $Q$  na saída das componentes polifásicas do filtro de máscara.

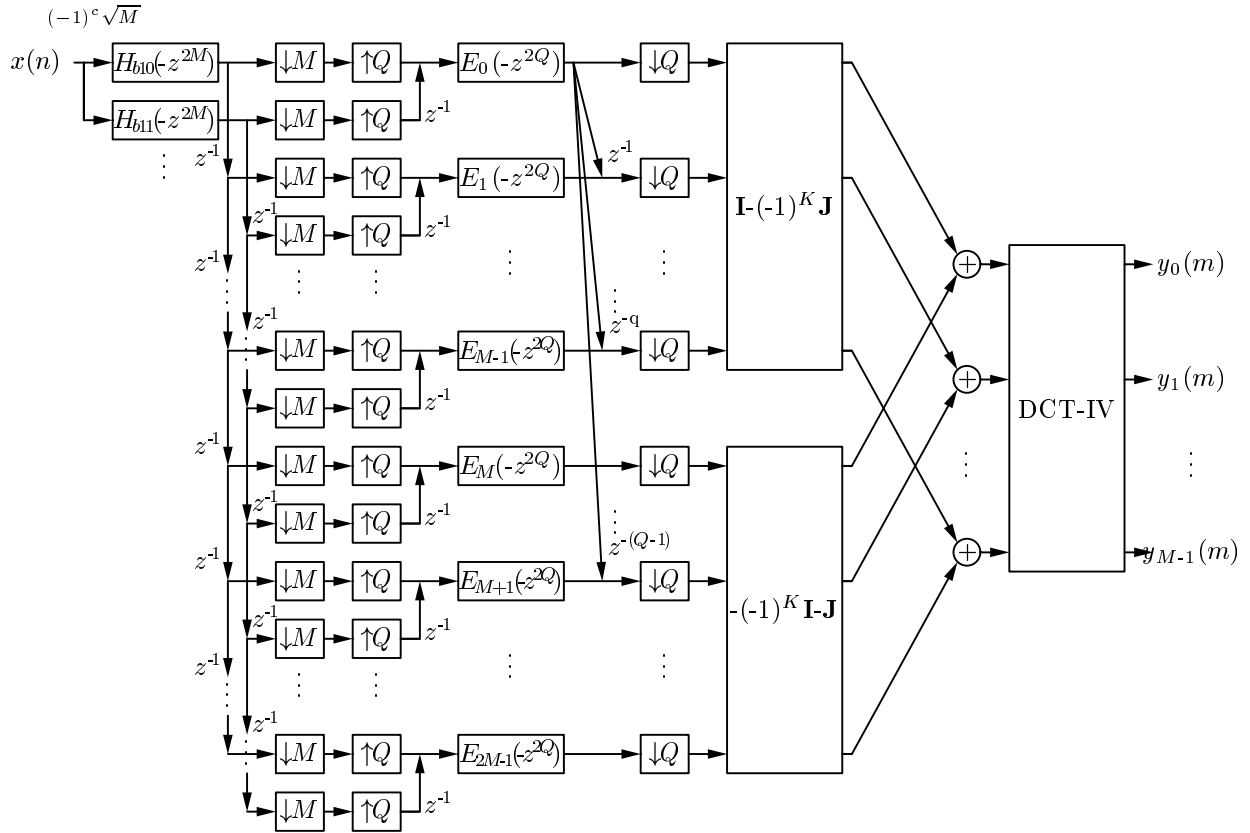


Figura 6.9: A estrutura do banco de análise do CMFRM para  $L < M$ . Para simplificação do desenho, apenas as saídas da componente polifásica  $E_0(-z^2)$  estão representadas. Estas saídas somam apenas a cada  $Lq$  componentes polifásicas.

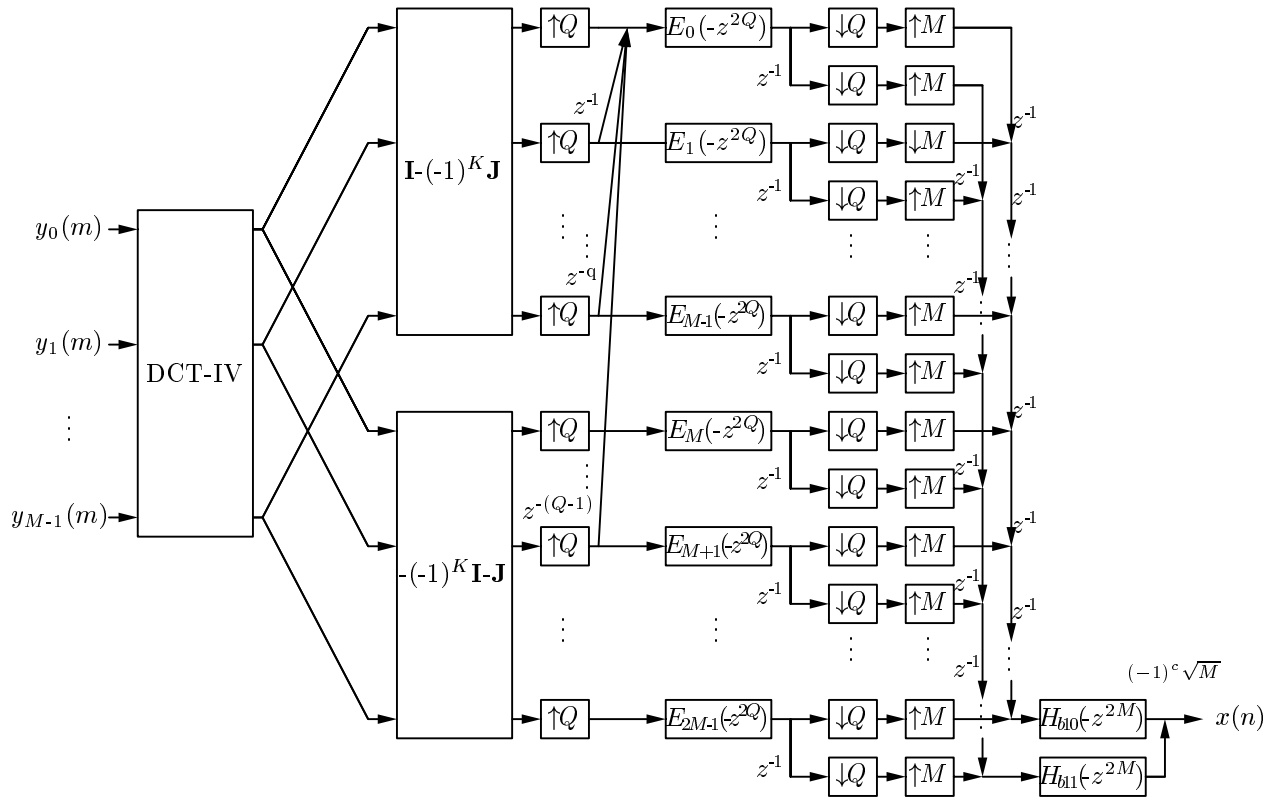


Figura 6.10: A estrutura do banco de síntese para o CMFRM com  $L < M$ .

será dada por

$$C = \frac{4MK_2Q}{M} + N_b + 1 + \alpha = 4K_2Q + N_b + 1 + \alpha \quad (6.21)$$

já que agora cada componente realiza a operação de  $Q$  sinais provenientes das decomposições polifásicas do filtro de base. Por outro lado, o número de operações multiplicativas distintas continua sendo dado pela equação (6.9).

## 6.6 CMFRM com fator de interpolação genérico

Nos casos anteriores, verificamos que é possível derivar estruturas para os casos em que  $L = 2K_1M$  e  $L = \frac{M}{K_1}$ , para  $K_1 = 1, 2, \dots$ . Entretanto, no primeiro caso, o filtro CMFRM não pode ser realizado como um filtro que atenda às condições PR ou NPR, pois uma das condições necessárias para se chegar à reconstrução perfeita ou a próxima da perfeita, ou seja,

$$\omega_{3dB} \approx \frac{1}{2M}\pi \quad (6.22)$$

não pode ser satisfeita, pois esta frequência cai no centro de uma das bandas do filtro interpolado ou de seu complementar (como já foi visto), a não ser que o fator de interpolação seja muito alto, mas neste caso o fator de *roll-off* tende a zero. Para evitar que isto ocorra, um valor de  $L$  que não seja exatamente múltiplo de  $2M$  deve ser utilizado.

Nesta seção aproveitaremos os resultados das seções anteriores para obter a estrutura para o caso mais geral em que definimos:

$$L = 2K_a M + \frac{M}{K_b} \quad (6.23)$$

ou seja, estamos utilizando os dois casos das seções anteriores. Com isto, a equação (6.5) fica:

$$H_m(z) = \sum_{i=0}^{N_b} \left[ h_{b1}(i) z^{-Li} \sum_{n=0}^{2K_2 M - 1} c_{m, [n + (2K_a M + \frac{M}{K_b})i]} g_1(n) z^{-n} \right] \quad (6.24)$$

e utilizando  $Q = 2K_b$ ,  $i = q + kQ$  e  $N_b = QK_3 - 1$ , teremos:

$$\begin{aligned} H_m(z) &= \sum_{k=0}^{K_3-1} \sum_{q=0}^{Q-1} \left( h_{b1}(q + kQ) z^{-L(q+kQ)} \sum_{n=0}^{2K_2 M - 1} c_{m, [n + (2K_a M + \frac{M}{K_b})(q+kQ)]} g_1(n) z^{-n} \right) \\ &= \sum_{k=0}^{K_3-1} \sum_{q=0}^{Q-1} \left( h_{b1}(q + kQ) z^{-L(q+kQ)} \sum_{n=0}^{2K_2 M - 1} c_{m, [n + 2K_a M(q+kQ) + q\frac{M}{K_b} + 2kM]} g_1(n) z^{-n} \right) \end{aligned} \quad (6.25)$$

Utilizando a relação da equação (5.27) teremos:

$$H_m(z) = \sum_{k=0}^{K_3-1} \sum_{q=0}^{Q-1} \left( h_{b1}(q + kQ) z^{-L(q+kQ)} (-1)^{k+K_a q + K_a k Q} \sum_{n=0}^{2K_2 M - 1} c_{m, (n + \frac{M}{K_b} q)} g_1(n) z^{-n} \right) \quad (6.26)$$

Lembrando que  $Q = 2K_b$  é sempre par, e realizando as decomposições polifásicas de  $G_1(z)$ , teremos:

$$H_m(z) = \sum_{k=0}^{K_3-1} \sum_{q=0}^{Q-1} \left( h_{b1}(q + kQ) z^{-L(q+kQ)} (-1)^{k+K_a q} \sum_{j=0}^{2M-1} c_{m, (n + \frac{M}{K_b} q)} z^{-j} E_j(-z^{2M}) \right) \quad (6.27)$$

Definindo as componentes polifásicas modificadas do filtro interpolado como:

$$H'_q(z) = \sum_{k=0}^{K_3-1} (-1)^{K_a q} h_{b1}(q + kQ) z^{-k}, \quad q = 0, 1, \dots, Q - 1 \quad (6.28)$$



a equação (6.27) pode ser escrita como:

$$H_m(z) = \sum_{q=0}^{Q-1} \left( z^{-2K_a M q} H'_q(-z^{2K_a M}) \sum_{j=0}^{2M-1} c_{m, (n + \frac{M}{K_b} q)} z^{-j} E_j(-z^{2M}) \right) \quad (6.29)$$

Note, entretanto que  $\frac{M}{K_b}$  é o valor do interpolador  $L$  da seção anterior. Definindo agora

$$L_1 = 2K_a M \quad (6.30)$$

$$L_2 = \frac{M}{K_b} \quad (6.31)$$

de modo que  $L = L_1 + L_2$ , e sendo  $L_1$  e  $L_2$  valores inteiros não nulos, teremos uma estrutura eficientemente realizável e, com isto, a equação (6.29) fica:

$$H_m(z) = \sum_{q=0}^{Q-1} \left( z^{-L_1 q} H'_q(z^{L_1}) \sum_{j=0}^{2M-1} c_{m, (n + L_2 q)} z^{-j} E_j(-z^{2M}) \right) \quad (6.32)$$

De acordo com esta última equação, a estrutura de análise será a da Figura 6.11. A diferença entre esta estrutura e a estrutura da Figura 6.9, é que o fator de interpolação das componentes polifásicas do filtro de base agora não é mais  $2M$ , mas sim um valor  $L_1 = 2K_a M$ .

A complexidade computacional desta estrutura será idêntica a da estrutura anterior, pois a alteração do fator de interpolação do filtro de base não modifica a complexidade computacional do sistema. Portanto, para a estrutura da Figura 6.11, teremos:

$$C = \frac{4MK_2Q}{M} + N_b + 1 + \alpha = 4K_2Q + N_b + 1 + \alpha \quad (6.33)$$

e o número de multiplicações distintas, assim como nos casos anteriores, será dado pela equação (6.9).

## 6.7 Exemplos numéricos

Nesta seção iremos exemplificar os vários casos descritos anteriormente, isto é, os casos em que  $L < M$ ,  $L = M$  e  $L > M$ .

### 1. Projeto com fator de interpolação igual ao número de canais ( $L = M$ )

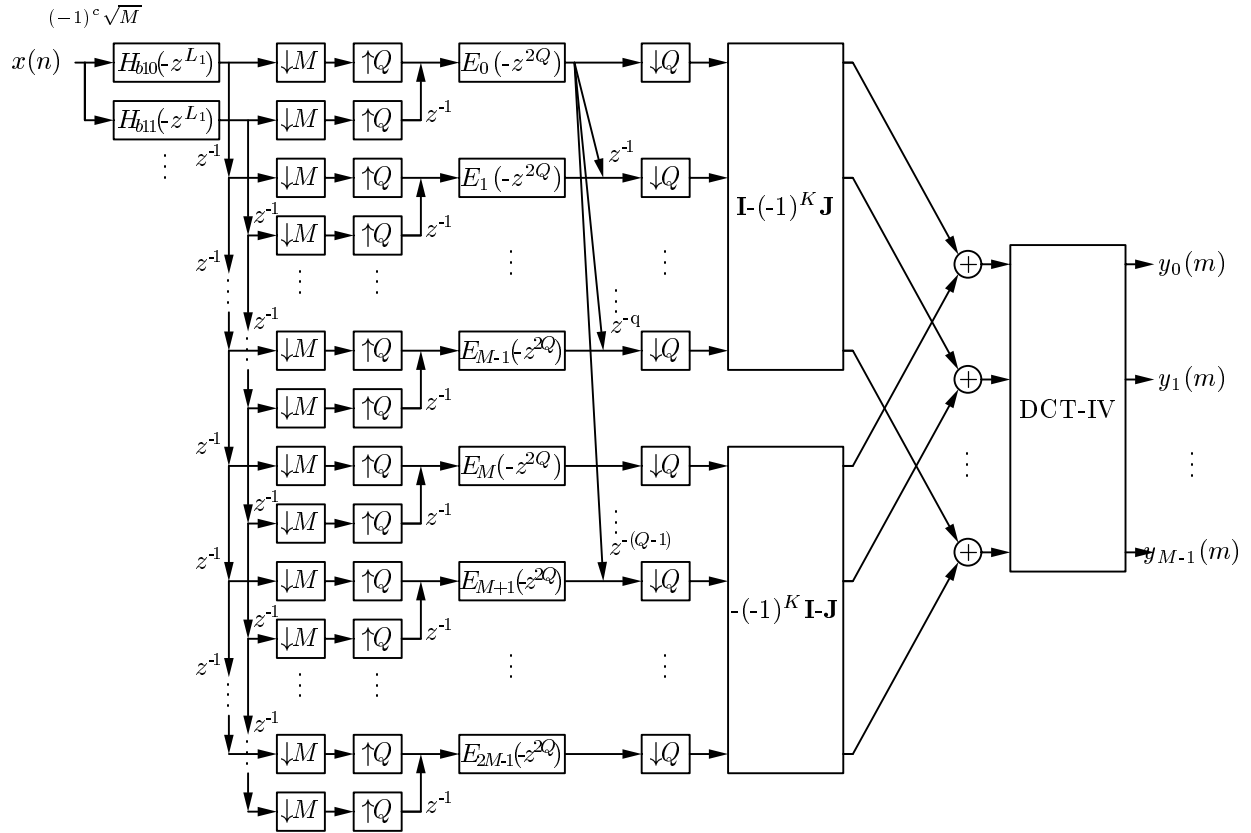


Figura 6.11: A estrutura do banco de análise do CMFRM para  $L = K_a M + \frac{M}{K_b}$ . Para simplificação do desenho, apenas as saídas da componente polifásica  $E_0(-z^2)$  estão representadas. Estas saídas somam apenas a cada  $L_2 q$  componentes polifásicas.

Tabela 6.1: Tabela com os melhores casos de redução para o exemplo 1.

$L$	$N_b$	$N_+$	$N_-$	$\mathcal{M}_{\text{tot}}$
6	100	30	-	132
7	86	38	-	126
8	74	50	-	126
9	66	64	-	132
10	60	82	-	144

Para este projeto, iremos adotar um fator de *roll-off*  $\rho$  igual a 0.1 e número de canais  $M = 8$ . Desejamos também que o filtro protótipo tenha um *ripple* máximo de banda passante de 0.05dB e atenuação mínima na banda de rejeição de 50dB. De acordo com as equações de reconstrução perfeita, devemos ter:

$$\omega_s = \frac{(1 + \rho)}{2M} \pi \quad (6.34)$$

$$\omega_{3\text{dB}} = \frac{1}{2M} \pi \quad (6.35)$$

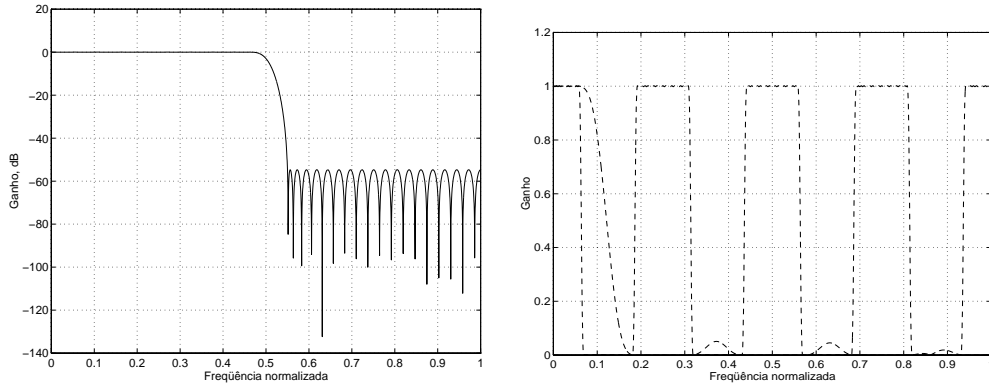
Uma boa aproximação inicial é supor que

$$\omega_p = \frac{(1 - \rho)}{2M} \pi \quad (6.36)$$

já que o fator de *roll-off* é pequeno. Utilizando estas especificações em um projeto de filtro FRM, obteremos os valores da Tabela 1, para  $L = 6, \dots, 10$ . Note que o fator de *roll-off* escolhido faz com que, para  $M = 8$  canais, a melhor redução seja para  $L$  igual a 7 ou 8, então poderemos escolher  $L = M = 8$ . Neste projeto, iremos também aumentar a ordem do filtro de máscara positiva para  $N_+ = 63$ , de modo que este filtro possa ser decomposto em  $2M = 16$  componentes polifásicas, cada uma de ordem  $K_2 - 1 = 3$  (ou seja,  $K_2 = 4$  coeficientes por componente polifásica). Note também que a melhor linha de redução só utiliza o ramo superior do FRM.

Projetando o filtro FRM, obtemos  $\omega_{3\text{dB}} = 0.0611\pi$ , um pouco abaixo de  $0.0625\pi = \frac{1}{2M}\pi$ . Alterando ligeiramente o valor de  $\omega_p$  para

$$\omega_p = \left[ \frac{(1 - \rho)}{2M} + 0.0025 \right] \pi \quad (6.37)$$



(a) Filtro de base.

(b) Operação de mascaramento.

Figura 6.12: Composição do filtro FRM para o exemplo 1.

e reprojetoando, com as ordens dos subfiltros acima obtidas, teremos os seguintes resultados para o filtro FRM:

$$\omega_{3\text{dB}} \approx 0.0625\pi = \frac{1}{2M}\pi \quad (6.38)$$

$$\delta_p = 0.03482\text{dB} \quad (6.39)$$

$$A_{\text{min}} = 54.79\text{dB} \quad (6.40)$$

A resposta do filtro de base e a operação de mascaramento do filtro interpolado podem ser vistas na Figura 6.12.

Realizando, agora o filtro CMFRM a partir dos filtros de base e de máscara obtidos, isto é, modulando o filtro protótipo FRM, obteremos os seguintes valores:

$$\delta_1 = 0.01381 \approx 0.12\text{dB} \quad (6.41)$$

$$\delta_2 = -60.44\text{dB} \quad (6.42)$$

$$A_{\text{rej}} = 54.9\text{dB} \quad (6.43)$$

$$\text{ISI} = -97.4\text{dB} \quad (6.44)$$

$$\text{ICI} = -110.8\text{dB} \quad (6.45)$$

O filtro FRM protótipo pode ser visto na Figura 6.13, juntamente com as funções  $T_i(z)$  que dão origem aos valores de  $\delta_1$  e de  $\delta_2$ . O valor máximo da função  $T_0(e^{j\omega})$  determina  $\delta_1$  e o valor máximo da função de pior caso de  $T_i(e^{j\omega})$ ,

que é  $T_1(e^{j\omega})$ , determina  $\delta_2$ . O valor máximo da função de pior caso  $T_{k,l}(e^{j\omega})$  que para este projeto são as funções de interferência entre canais adjacentes, isto é,  $T_{i,i\pm 1}(e^{j\omega})$  determina o valor de ICI.

Realizando um projeto deste tipo a partir de um filtro protótipo FIR, a ordem necessária para gerar um filtro aproximado (isto é, com banda de transição compatível com o fator de *roll-off*  $\rho = 0.1$ , *ripple* de 0.05dB na banda passante e atenuação de 50dB na banda de rejeição), seria de  $N = 383$ , que decompostos em  $2M = 16$  componentes polifásicas, resulta em componentes polifásicas de ordem  $K - 1 = 23$  (ou seja,  $K = 24$  coeficientes por componente polifásica). O fator de redução de complexidade computacional será de:

$$\mathcal{R}_c = \frac{\mathcal{C}_{\text{CMFRM}}}{\mathcal{C}_{\text{CMFB}}} = \frac{2K_2Q + N_b + 1}{2K} = \frac{16 + 74 + 1}{48} = 1.875 \quad (6.46)$$

e, portanto, este filtro não apresentará redução da complexidade computacional, embora apresente menor número de multiplicadores do que em um projeto CMFB padrão, pois

$$\mathcal{R}_M = \frac{\mathcal{M}_{\text{CMFRM}}}{\mathcal{M}_{\text{CMFB}}} = \frac{2K_2M + N_b + 1}{2KM} = \frac{64 + 74}{384} = 35.94\% \quad (6.47)$$

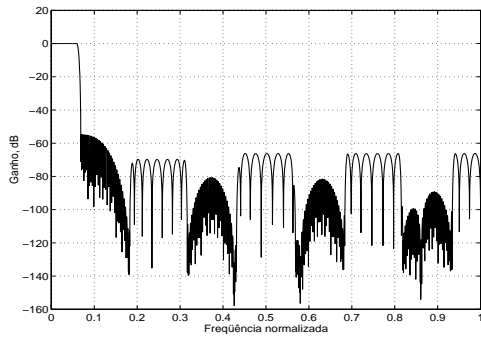
## 2. Projeto com fator de interpolação menor que o número de canais ( $L = \frac{M}{K_1} < M$ )

Para este segundo projeto, iremos adotar um fator de *roll-off*  $\rho$  igual a 0.7 e número de canais  $M = 8$ . Desejamos também que o filtro protótipo tenha um *ripple* máximo de banda passante de 0.1dB e atenuação mínima na banda de rejeição de 50dB. Calculando  $\omega_s$  e  $\omega_p$  do mesmo modo do exemplo anterior, teremos, para

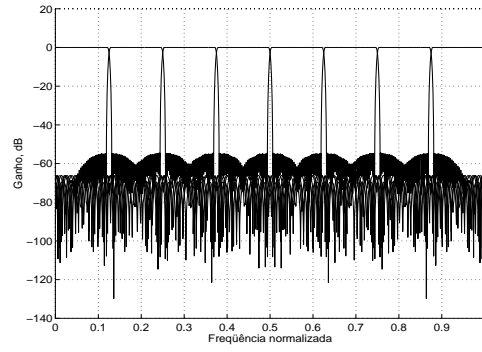
$$\omega_p = \left[ \frac{(1 - \rho)}{2M} + 0.0163 \right] \pi \quad (6.48)$$

o valor de  $\omega_{3\text{dB}} = 0.0625\pi$  satisfazendo a condição de reconstrução quase-perfeita. Com estas especificações, obteremos as linhas de redução da Tabela 6.2.

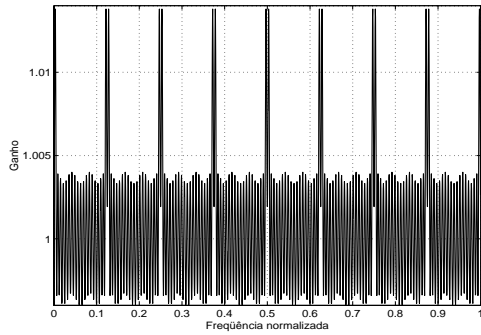
Neste caso, o fator de *roll-off* escolhido faz com que a melhor redução seja para  $L$  igual a 4, então escolhemos  $L = \frac{M}{2} = 4$ . Assim como no exemplo anterior, iremos ajustar a ordem do filtro de máscara de modo que se obtenha



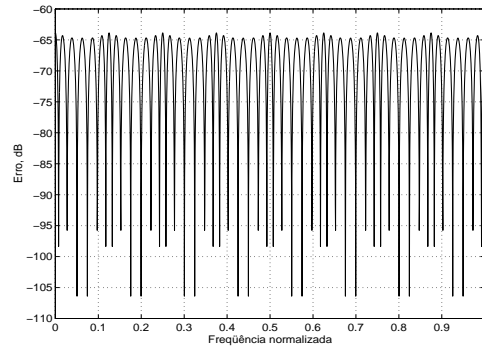
(a) Filtro protótipo FRM.



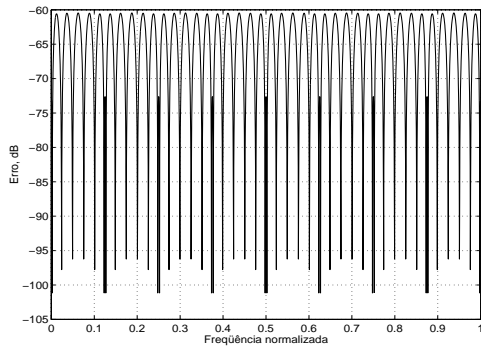
(b) Banco de filtros resultante.



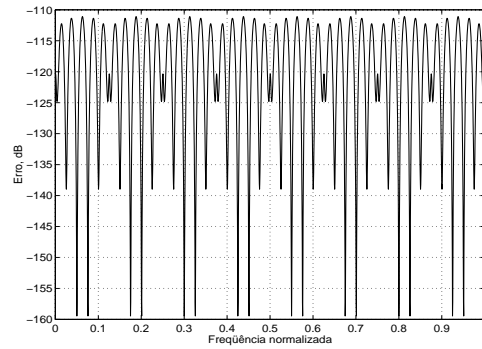
(c) Função  $T_0(e^{j\omega})$ .



(d) Função  $T_1(e^{j\omega})$ .



(e) Função  $T_2(e^{j\omega})$ .

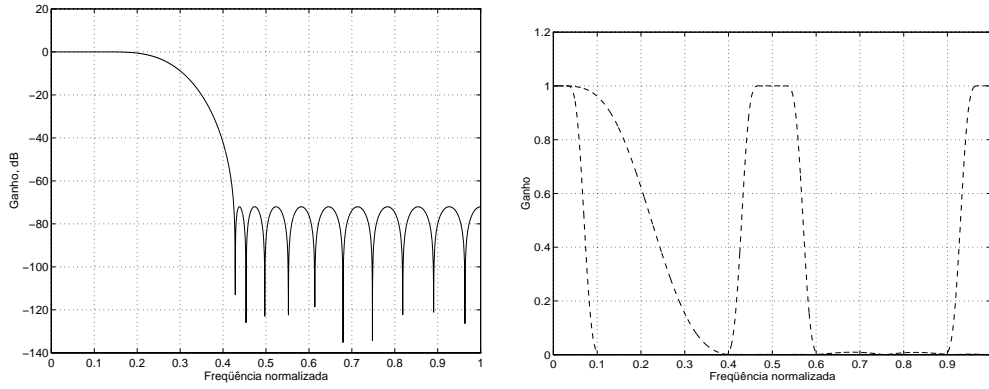


(f) Função  $T_{0,1}(e^{j\omega})$  e demais funções de interferência entre canais adjacentes.

Figura 6.13: Respostas em magnitude do filtro protótipo e das funções  $T_i(e^{j\omega})$  e  $T_{k,l}(e^{j\omega})$  responsáveis pelos valores de  $\delta_1$ ,  $\delta_2$  e pelos valores de ISI e de ICI para o exemplo 1.

Tabela 6.2: Tabela com os melhores casos de redução para o exemplo 2.

$L$	$N_b$	$N_+$	$N_-$	$\mathcal{M}_{\text{tot}}$
2	54	5	-	61
3	36	11	-	49
4	28	17	-	47
5	22	25	-	49
6	18	33	-	53



(a) Filtro de base.

(b) Operação de mascaramento.

Figura 6.14: Composição do filtro FRM para o exemplo 2.

$N_+ = 2K_2M - 1$ . Fazendo com que  $N_+ = 15$ , este filtro pode ser decomposto em  $2M = 16$  componentes polifásicas, cada uma com ordem  $K_2 - 1 = 0$  (ou seja,  $K_2 = 1$  coeficiente por decomposição polifásica). Note também que, neste caso, a melhor linha de redução só utiliza o ramo superior do FRM.

Projetando o filtro FRM, obteremos os seguintes valores:

$$\omega_{3\text{dB}} \approx 0.0625\pi = \frac{1}{2M}\pi \quad (6.49)$$

$$\delta_p = 0.0791\text{dB} \quad (6.50)$$

$$A_{\text{min}} = 52.8\text{dB} \quad (6.51)$$

A resposta do filtro de base e a operação de mascaramento do filtro interpolado para este exemplo podem ser vistas na Figura 6.14.

Realizando agora o filtro CMFRM a partir do protótipo FRM obtido, obteremos os seguintes valores:

$$\delta_1 = 0.005668 \approx 0.049\text{dB} \quad (6.52)$$

$$\delta_2 = -52.4\text{dB} \quad (6.53)$$

$$A_{\text{rej}} = 75.03\text{dB} \quad (6.54)$$

$$\text{ISI} = -103.2\text{dB} \quad (6.55)$$

$$\text{ICI} = -98.72\text{dB} \quad (6.56)$$

A resposta em magnitude do filtro FRM protótipo pode ser visto na Figura 6.15, juntamente com as funções  $T_i(e^{j\omega})$  que dão origem aos valores de  $\delta_1$  e  $\delta_2$ , e a função de pior caso de  $T_{k,l}(e^{j\omega})$ .

Realizando um projeto deste tipo a partir de um filtro protótipo FIR, a ordem necessária para gerar um filtro aproximado é de  $N = 79$ , que decomposto em  $2M = 16$  componentes polifásicas, resultam em componentes de ordem  $K - 1 = 4$  (ou seja,  $K = 5$  coeficientes por componente polifásica). O fator de redução de complexidade computacional será de:

$$\mathcal{R}_c = \frac{\mathcal{C}_{\text{CMFRM}}}{\mathcal{C}_{\text{CMFB}}} = \frac{2K_2Q + N_b + 1}{2K} = \frac{8 + 28 + 1}{10} = 3.7 \quad (6.57)$$

e, novamente, neste caso, não obteve-se redução da complexidade do banco de filtros, devido ao fato dos subfiltros possuírem ordens relativamente baixas. O fator de redução do número de multiplicadores distintos, neste caso, é de

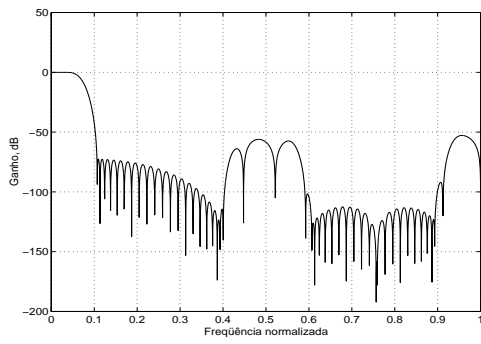
$$\mathcal{R}_M = \frac{\mathcal{M}_{\text{CMFRM}}}{\mathcal{M}_{\text{CMFB}}} = \frac{2K_2M + N_b + 1}{2KM} = \frac{16 + 29}{80} = 56.25\% \quad (6.58)$$

### 3. Projeto com fator de interpolação maior que o número de canais ( $L = 2K_aM + \frac{M}{K_b}$ )

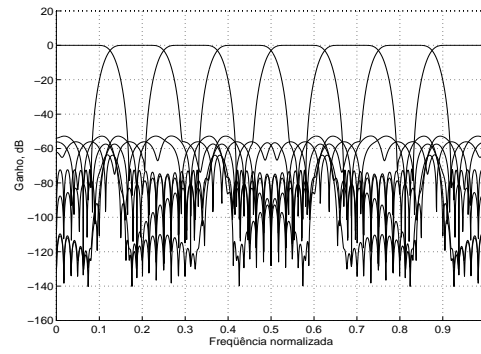
Neste terceiro exemplo, iremos utilizar um fator de *roll-off* igual a 0.015 e número de canais  $M = 8$ . Desejamos também que o filtro tenha um *ripple* máximo de banda passante de 0.05dB e atenuação mínima na banda de rejeição de 50dB. Utilizando o valor de

$$\omega_p = \left[ \frac{(1 - \rho)}{2M} + 0.00023 \right] \pi \quad (6.59)$$

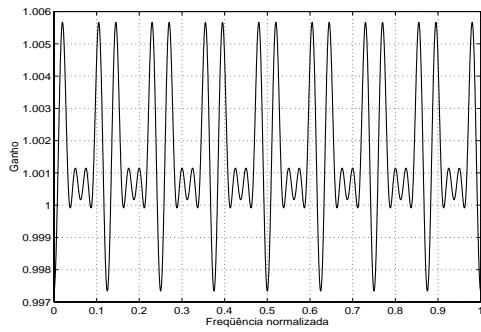




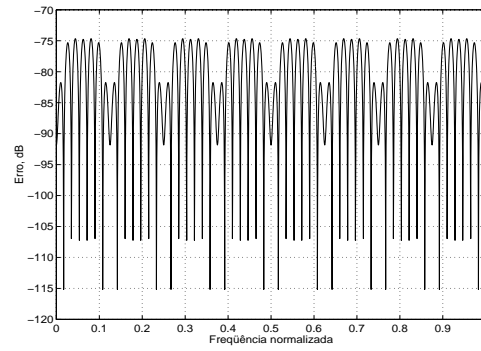
(a) Filtro protótipo FRM.



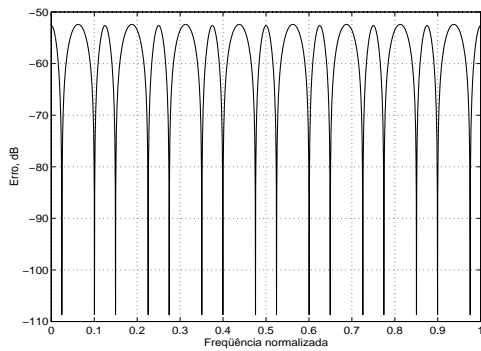
(b) Banco de filtros resultante.



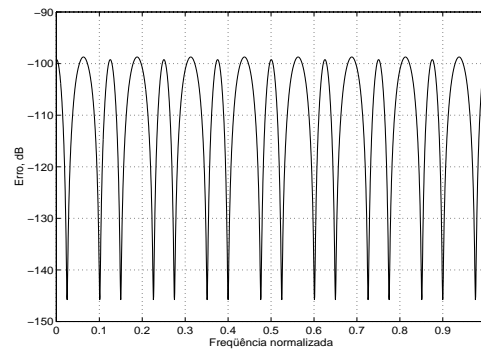
(c) Função  $T_0(e^{j\omega})$ .



(d) Função  $T_1(e^{j\omega})$ .



(e) Função  $T_2(e^{j\omega})$ .



(f) Função de pior caso de  $T_{k,l}(e^{j\omega})$ .

Figura 6.15: Respostas em magnitude do filtro protótipo e das funções  $T_i(e^{j\omega})$  e  $T_{k,l}(e^{j\omega})$  responsáveis pelos valores de  $\delta_1$  e  $\delta_2$  e pelos valores de ISI e ICI para o exemplo 2.

Tabela 6.3: Tabela com os melhores casos de redução para o exemplo 3.

$L$	$N_b$	$N_+$	$N_-$	$\mathcal{M}_{\text{tot}}$
22	204	179	109	495
23	194	161	127	485
24	186	147	149	485
25	180	137	177	497
26	172	128	214	517

obteremos  $\omega_{3\text{dB}} \approx 0.0625\pi = \frac{1}{2M}\pi$ . Utilizando estas especificações para o projeto do filtro reduzido, obteremos as linhas de redução para o FRM indicadas na Tabela 6.3, para  $L = 22, \dots, 26$ .

Neste caso, o fator de *roll-off* levou a um fator ótimo de redução de  $L = 23$  ou  $L = 24$ , então poderemos escolher  $L = M = 2K_aM + \frac{M}{K_b} = 24$ , se  $K_a = 1$  e  $K_b = 1$ . Neste projeto, iremos reduzir ambas as ordens do filtro de máscara positiva e negativa para  $N_+ = N_- = 143$ , de modo que estes filtros possam ser decompostos em  $2M = 16$  componentes polifásicas, cada uma com ordem  $K_2 - 1 = 8$  (ou seja,  $K_2 = 9$  coeficientes por componente polifásica). Note também que neste caso necessitamos dos dois ramos do FRM.

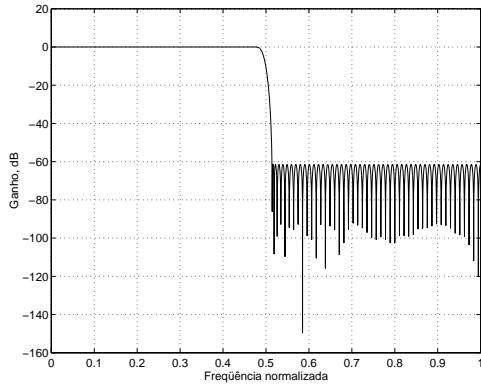
Assim, para  $L = 24$ , teremos os seguintes resultados para o filtro FRM:

$$\delta_p = 0.0468\text{dB} \quad (6.60)$$

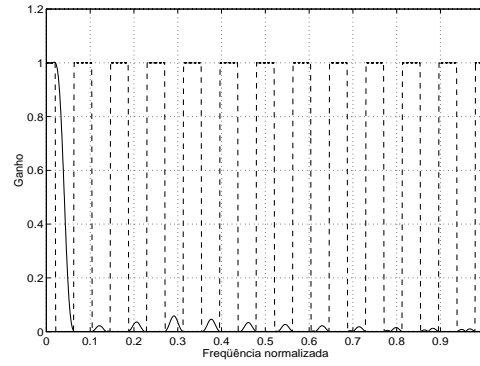
$$A_{\text{min}} = 56.48\text{dB} \quad (6.61)$$

e a resposta do filtro de base e as operações de mascaramento podem ser vistas na Figura 6.16.

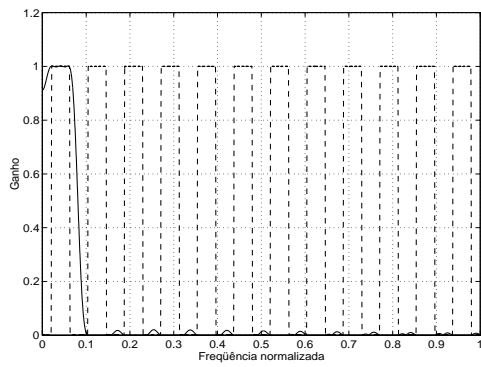
Realizando agora o filtro CMFRM a partir dos filtros de base e os de máscara



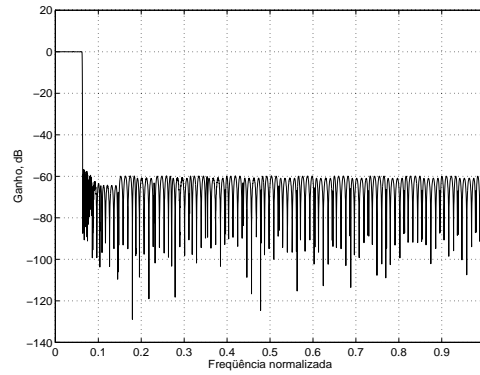
(a) Filtro de base.



(b) Operação de mascaramento positivo.



(c) Operação de mascaramento negativo.



(d) Filtro FRM resultante.

Figura 6.16: Composição do filtro FRM para o exemplo 4, e o filtro protótipo obtido.

calculados, obteremos os seguintes valores:

$$\delta_1 = 0.0087 \approx 0.076\text{dB} \quad (6.62)$$

$$\delta_2 = -56.09\text{dB} \quad (6.63)$$

$$A_{\text{rej}} \approx 55\text{dB} \quad (6.64)$$

$$\text{ISI} = -100.2\text{dB} \quad (6.65)$$

$$\text{ICI} = -83\text{dB} \quad (6.66)$$

A resposta em magnitude do banco de filtros resultante pode ser vista na Figura 6.17, juntamente com as funções  $T_i(e^{j\omega})$  que originam os valores de  $\delta_1$  e de  $\delta_2$ , e com a função de pior caso de  $T_{k,l}(e^{j\omega})$ .

O valor máximo de  $T_0(e^{j\omega})$  determina  $\delta_1$  e o valor máximo da função de pior caso de  $T_i(e^{j\omega})$ , que neste caso é  $T_1(e^{j\omega})$ , determina  $\delta_2$ . O valor máximo da função de pior caso de  $T_{k,l}(e^{j\omega})$ , que neste caso são as funções entre os canais  $i$  e  $i + 2$ , isto é,  $T_{i,i\pm 2}(e^{j\omega})$  determina o valor de ICI.

Realizando um projeto deste tipo a partir de um filtro protótipo FIR, a ordem necessária para gerar um filtro aproximado seria de  $N = 3215$ , que decomposto em  $2M = 16$  componentes polifásicas resultaria em componentes de ordem  $K - 1 = 200$  (ou seja,  $K = 201$  coeficientes por decomposição polifásica). O fator de redução de complexidade computacional será de

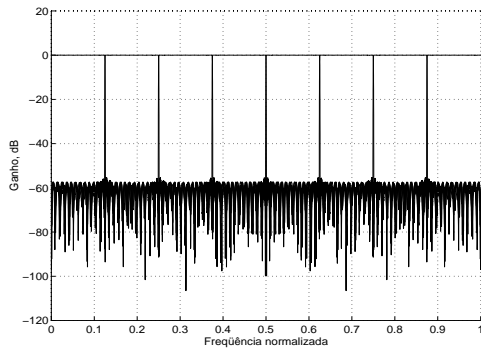
$$\mathcal{R}_c = \frac{\mathcal{C}_{\text{CMFRM}}}{\mathcal{C}_{\text{CMFB}}} = \frac{4K_2Q + N_b + 1}{2K} = \frac{72 + 186 + 1}{402} = 55.4\% \quad (6.67)$$

e, como podemos notar, o fator de redução é bem abaixo dos obtidos para os exemplos anteriores, uma vez que os filtros envolvidos possuem ordens relativamente mais altas. Também aqui obtém-se a redução do número de coeficientes dos filtros componentes que fica

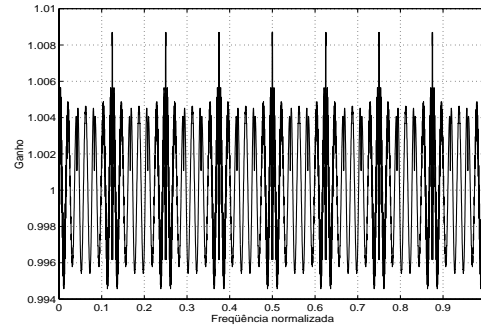
$$\mathcal{R}_M = \frac{\mathcal{M}_{\text{CMFRM}}}{\mathcal{M}_{\text{CMFB}}} = \frac{4K_2M + N_b + 1}{2KM} = \frac{288 + 187}{3216} = 14.77\% \quad (6.68)$$

## 6.8 Conclusões

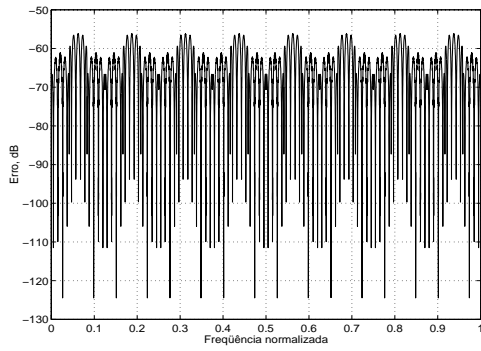
Como foi visto neste capítulo, o filtro protótipo do CMFB pode ser substituído por um filtro FRM desde que se atendam algumas restrições de fator de interpolação e número de coeficientes. Verificando a convolução do filtro interpolado com



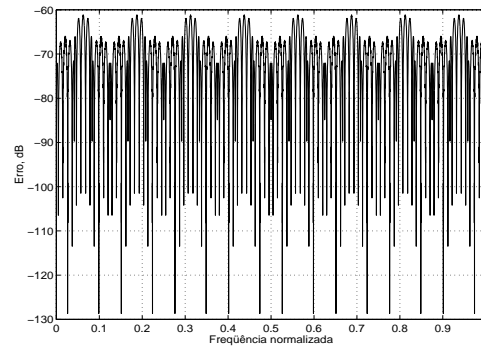
(a) Banco de filtros resultante.



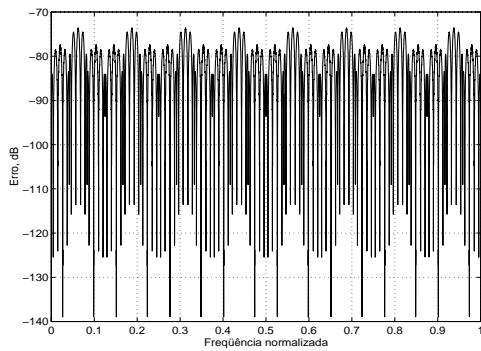
(b) Função  $T_0(e^{j\omega})$ .



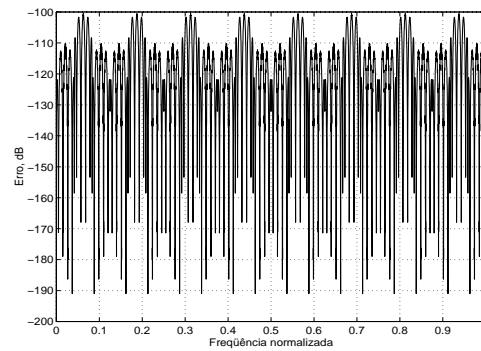
(c) Função  $T_1(e^{j\omega})$ .



(d) Função  $T_2(e^{j\omega})$ .



(e) Função  $T_3(e^{j\omega})$ .



(f) Função  $T_{0,2}(e^{j\omega})$ .

Figura 6.17: Respostas em magnitude do banco de filtros e das funções  $T_i(e^{j\omega})$  e  $T_{k,l}(e^{j\omega})$  responsáveis pelos valores de  $\delta_1$ ,  $\delta_2$  e pelos valores de ISI e ICI para o exemplo 3.

o filtro de máscara, foi possível derivar estruturas eficientes para certos fatores de interpolação. Para  $L = M$ , teremos apenas duas decomposições polifásicas do filtro de base, e se o filtro de máscara possuir  $2M$  coeficientes, então a estrutura mais simplificada pode ser utilizada. Uma estrutura generalizada para o caso em que  $L = K_a M + \frac{M}{K_b}$ , com  $K_a = 0, 1, \dots$  e  $K_b = 1, 2, \dots$  pôde ser derivada. Calculando o fator de redução da complexidade computacional, verificamos que para altos fatores de  $L$ , a estrutura apresenta alto fator de redução de complexidade computacional.

# Capítulo 7

## Conclusões

Como foi visto na primeira parte, o método WLS-Chebyshev pode ser utilizado para otimizar os coeficientes de um filtro FRM. A sua versatilidade faz com que seja possível a modificação da função objetiva a ser minimizada. Utilizamos as restrições de pico e a tomada da função de erro na saída do filtro FRM, demonstrando a facilidade em se empregar o algoritmo.

A otimização do filtro FRM também pode ser feita, além das maneiras mencionadas, em dois passos: primeiro, otimizando os filtros de máscara, supondo já conhecido o filtro de base (o primeiro chute para o filtro de base é um minimax). Em seguida, a otimização do filtro de base, mantendo os filtros de máscara obtidos. Ambos os passos utilizam a otimização WLS-Chebyshev com tomada da função de erro na saída do FRM. Aplicando este algoritmo recursivamente, é possível chegar a reduções ligeiramente mais baixas que as obtidas aqui. Consideramos suficiente, porém, demonstrar a versatilidade do WLS-Chebyshev, e a facilidade de implementação, obtendo ainda um fator de redução cerca de 5% abaixo do obtido para um algoritmo minimax com bandas de *don't care*.

O WLS-Chebyshev também permite variar o número de picos a serem restritos, e portanto, apesar de aumentar a sua versatilidade, o problema de otimização fica com uma variável a mais. Para a escolha deste valor, deve-se ter em mente o tipo de aplicação do filtro FRM, pois pode ser desejável filtros com características minimax, bem como características *least-squares*, devendo o projetista escolher a melhor relação entre ambas.

Na segunda parte deste trabalho, abordamos os filtros modulados por cosseno,

utilizando um filtro FRM como protótipo do banco de filtros, dando origem a uma nova classe de banco de filtros – CMFRMs. Para certos fatores de interpolação, conseguimos derivar estruturas eficientes que realizam as respostas do banco, utilizando filtros em cascata que são decompostos em componentes polifásicas. As complexidades computacionais de cada caso foram avaliadas, e pode-se notar que os filtros CMFRM podem apresentar redução de complexidade computacional, desde que as ordens dos filtros sejam relativamente altas. Em contrapartida, o número de multiplicadores no sistema sempre se reduz do mesmo modo que um filtro FRM substituindo um filtro FIR na forma direta. Particularmente, filtros com  $L = M$  e com  $K_2 = 1$  coeficiente por decomposição polifásica do filtro de máscara apresentam melhores possibilidades de redução, pois neste caso não é preciso implementar os interpoladores em  $Q$ , já que as saídas das componentes polifásicas do filtro de base são combinadas nas linha de atrasos que precedem as componentes polifásicas do filtro de mascaramento.

A convolução do filtro de base com o filtro de mascaramento representa ainda um problema quanto às condições de reconstrução perfeita. Particularmente, se o número de canais for ímpar, não é possível, por exemplo, utilizar apenas um filtro de máscara para se obter a reconstrução perfeita. Entretanto, para  $M$  par, é possível ao menos derivar filtros CMFRMs com reconstrução perfeita, com uma variável a ser otimizada. É necessário agora verificar se é possível obter-se um filtro CMFRM com número de variáveis a otimizar que seja ao mesmo tempo suficientes para realizar um bom filtro protótipo, e serem em menor número que as variáveis em um CMFB padrão. Com menos variáveis a otimizar, o processo de otimização poderá, talvez, ser realizado em dois passos, isto é, otimizando-se o filtro de base e depois os filtros de mascaramento.

Algumas propriedades também devem ser levantadas, tais como verificar se as componentes polifásicas do filtro de base e as dos filtros de máscara possuem, independente umas das outras, reconstrução perfeita. Se sim, seria então possível tratar os filtros de máscara separadamente do filtro de base, impondo primeiramente a reconstrução perfeita para os filtros de máscara, e depois para o filtro de base.

Enfim, as estruturas obtidas aqui para filtros modulados em cosseno nos permitem estudar os CMFBs por outro ponto de vista.



# Referências Bibliográficas

- [1] Lim, Y. C., *Frequency-Response Masking Approach for the Synthesis of Sharp Linear Phase Digital Filters*, IEEE Trans. on Circuits and Systems, vol. CAS-33, n. 4, pp. 357-364, Abril 1986.
- [2] Yang, R., Lim, Y. C., Bellanger, M., *Design of Sharp FIR Bandstop Filters Using Quadrature Masking Filters*, ICASSP, 1995.
- [3] Neuvo, Y., Rajan, G., Mitra, S. K., *Design of Narrow-Band FIR Bandpass Digital Filters with Reduced Arithmetic Complexity*, IEEE Trans. on Circuits and Systems, vol. CAS-34, n. 4, pp. 409-419, Abril 1987.
- [4] Yang, R., Liu, B., Lim, Y. C., *A New Structure of Sharp Transition FIR Filters Using Frequency-Response Masking*, IEEE Trans. on Circuits and Systems, vol. 35, n. 8, pp. 955-966, Agosto 1988.
- [5] Rajan, G., Neuvo, Y., Mitra, S. K., *On the Design of Sharp Cutoff Wide-Band FIR Filters with Reduced Arithmetic Complexity*, IEEE Trans. on Circuits and Systems, vol. 35, n. 11, pp. 1447-1454, Novembro 1988.
- [6] Saramäki, T., Lim, Y. C., *Use of the Remez Algorithm for Designing FIR Filters Utilizing the Frequency-Response Masking Approach*, ???, vol. III, pp. 449-455, ???
- [7] Lim, Y. C., Lian, Y., *Frequency-Response Masking Approach for Digital Filter Design: Complexity Reduction via Masking Filter Factorization*, IEEE Trans. on Circuits and Systems-II Analog and Digital Signal Processing, vol. 41, n. 8, pp. 518-525, Agosto 1994.

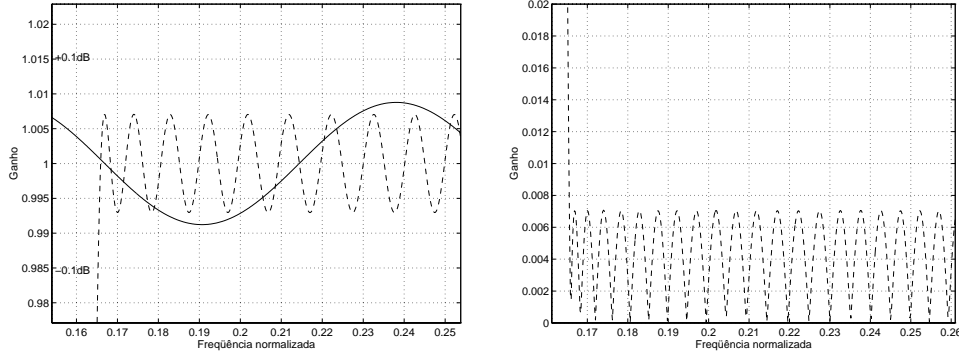
- [8] Lim, Y. C., Lee, J. H., Chen, C. K., Yang, R. H., *A Weighted Least Squares Algorithm for Quasi-Equiripple FIR and IIR Digital Filter Design*, IEEE Trans. on Signal Processing, vol. 40, n. 3, pp. 551-558, Março 1992.
- [9] Rice, J. R., Usow, K. H., *The Lawson Algorithm and Extensions*, Mathematics of Computation, vol. 22, pp. 118-127, 1968.
- [10] Diniz, P. S. R., Netto, S. L., *On WLS-Chebyshev FIR Digital Filters*, Journal of Circuits, Systems and Computers, vol. 9, nos. 3&4, pp. 155-168, 1999.
- [11] Adams, J. W., *FIR Digital Filters with Least-Squares Stopbands Subject to Peak-Gain Constraints*, IEEE Trans. on Circuits and Systems, vol. 39, n. 4, Abril 1991.
- [12] Barcellos, L. C. R. de, Netto, S. L., Diniz, P. S. R., *Design of FIR Filters Combining the Frequency-Response Masking and the WLS-Chebyshev Approaches*, ISCAS, 2001, Sydney, Australy.
- [13] Saramäki, T., *A Generalized Class of Cosine-Modulated Filter Banks*, Workshop on Transforms and Filter Banks, Tampere, Finland 1998.
- [14] Alhava, J., Viholainen, A., *Implementation of Nearly Perfect-Reconstruction Cosine-Modulated Filter Banks*, FinSig, Oulu, 1999.
- [15] Viholainen, A., Saramäki, T., Renfors, M., *Nearly Perfect-Reconstruction Cosine-Modulated Filter Bank Design for VDSL Modems*, ICECS, 1999, Paphos, Cyprus.
- [16] Barcellos, L. C. R. de, Neto, S. L., Diniz, P. S. R., *Design of Cosine-Modulated Filter Bank Prototype Filters Using the Frequency-Response Masking Approach*, ICASSP, 2001, Salt Lake City, USA.
- [17] Alhava, J., Viholainen, A., *Coefficient Quantization in Nearly Perfect-Reconstruction Cosine-Modulated Filter Banks*, ???, Tampere, Finland.
- [18] Alhava, J., Viholainen, A., *Coefficient Quantization in Perfect-Reconstruction Cosine-Modulated Filter Banks*, ???, Tampere, Finland.

# Apêndice A

## Estudo das margens de desvios de ganho do FRM

De acordo com a estrutura básica de um filtro FRM, podemos comprovar, de acordo com Lim [1] que os filtros de mascaramento exercem seus efeitos apenas sobre certas faixas da resposta em frequência do filtro resultante, por isto podemos utilizar o conceito de bandas não críticas no projeto. O mesmo se aplica à margem; em algumas bandas estaremos realizando uma composição entre a resposta do filtro base interpolado e a máscara positiva, e neste caso apenas as margens do filtro de base e o filtro de máscara positiva são relevantes. Nas outras bandas teremos o resultado do filtro interpolado complementar e o filtro de máscara negativa atuando, e neste caso apenas as margens do filtro interpolado complementar e do filtro de máscara negativa são relevantes. Portanto podemos dizer que as margens dos filtros de máscara são independentes umas das outras. Isto pode ser comprovado nas equações descritas a seguir.

Seja o filtro FRM final com resposta  $G(\omega) + \delta(\omega)$ , onde  $G(\omega)$  é a resposta desejada do filtro, e  $\delta(\omega)$  é o desvio máximo permitido. Sejam ainda  $G_m(\omega) + \delta_m(\omega)$ ,  $G_{mc}(\omega) + \delta_{mc}(\omega)$  e  $G_b(\omega) + \delta_b(\omega)$ , as respostas desejadas, com os seus respectivos desvios, dos filtros de máscara positiva, máscara negativa, e interpolado (igual ao do filtro base), respectivamente. Podemos analisar o comportamento destes filtros para as três faixas do filtro final, banda passante, de rejeição e de transição, aplicando,



(a) Desvios de ganho dos filtros de máscara positiva (linha contínua) e do filtro interpolado (linha tracejada).

(b) Desvio de ganho do filtro interpolado complementar. O ganho do filtro de máscara negativa é unitário.

Figura A.1: Detalhes dos desvios de ganho dos filtros componentes, para a faixa (1).

em cada caso, a fórmula do filtro FRM final:

$$\begin{aligned}
 G(\omega) + \delta(\omega) &= [G_m(\omega) + \delta_m(\omega)][G_b(\omega) + \delta_b(\omega)] \\
 &+ [G_{mc}(\omega) + \delta_{mc}(\omega)][1 - G_b(\omega) - \delta_b(\omega)]
 \end{aligned}
 \tag{A.1}$$

Assim, teremos as seguintes faixas a considerar:

1. quando  $G_m(\omega) = 1$  e  $G_{mc}(\omega) = 1$ , ou seja, a faixa de frequência em que o filtro desejado apresenta ganho unitário, devido à soma das respostas do filtro interpolado e seu complementar. Neste caso,  $G(\omega) = 1$ , e temos:

- (a) Se  $G_b(\omega) = 1$ : temos que

$$\begin{aligned}
 1 + \delta(\omega) &= [1 + \delta_m(\omega)][1 + \delta_b(\omega)] + [1 + \delta_{mc}(\omega)][-\delta_b(\omega)] \\
 1 + \delta(\omega) &= 1 + \delta_m(\omega) + \delta_b(\omega) + \delta_b(\omega)\delta_m(\omega) - \delta_b(\omega) - \delta_{mc}(\omega)\delta_b(\omega)
 \end{aligned}
 \tag{A.2}$$

e, portanto

$$\delta(\omega) = \delta_m(\omega) + \delta_m(\omega)\delta_b(\omega) - \delta_{mc}(\omega)\delta_b(\omega)
 \tag{A.3}$$

O pior caso para a função acima, será se  $\delta_{mc}(\omega)$  for mínimo e  $\delta_m(\omega)$  for máximo, ou seja, os desvios  $\delta_b$ , modulados por  $\delta_m$  e  $\delta_{mc}$  não se anulam parcialmente, mas se somam. Assim, aplicando a margem de desvio para

cada filtro componente, isto é, fazendo com que  $\delta_m(\omega)$ ,  $\delta_{mc}(\omega)$  e  $\delta_b(\omega)$  sejam iguais a  $\alpha\delta_{PROJ}$ , onde  $\delta_{PROJ}$  seja o desvio máximo permitido, especificado ao filtro final e  $\alpha = (1 - \text{margem de } ripple)$ , teremos:

$$\delta_{PROJ} = \alpha\delta_{PROJ} + 2\alpha^2[\delta_{PROJ}]^2 \Rightarrow \alpha = \frac{\sqrt{1 + 8\delta_{PROJ}} - 1}{4\delta_{PROJ}} \quad (\text{A.4})$$

ou seja, para o valor de desvio máximo  $\delta_{PROJ}$  especificado para a banda de passagem, obtemos a margem necessária para que o filtro final atenda às especificações nesta faixa de frequência, que vale:  $\text{margem} = (1 - \alpha)$ . Assim, para um desvio máximo desejado de 0.1 dB na banda de passagem ( $\delta_{PROJ} = 0.0114$ ), por exemplo, teremos:

$$1 - \alpha = 1 - \frac{\sqrt{1 + 8 \times 0.0114} - 1}{4 \times 0.0114} = 0.022 \quad (\text{A.5})$$

ou seja, a margem deverá ser de 2,2% para que o filtro final obtido atenda às especificações. Note que o valor de margem é muito baixo, pois nesta faixa considerada, o desvio do filtro de máscara positiva domina sobre os outros. Portanto, se especificássemos o filtro de máscara com o mesmo desvio do filtro final desejado, o comportamento do filtro final estaria bem próximo da especificação nesta faixa, e a margem calculada acima, portanto, se refere ao erro cometido pelos termos de segunda ordem ( $\delta_m(\omega)\delta_b(\omega)$  e  $\delta_{mc}(\omega)\delta_b(\omega)$ ) no cálculo acima.

Para um desvio especificado  $\delta_{PROJ} = 0.1$ , teremos, pela fórmula acima, que:  $1 - \alpha = 14.7\%$ , ou seja, mesmo para uma especificação alta de desvio máximo, como esta, a margem a ser deixada é de cerca de 15% do *ripple* total.

(b) Se  $G_b(\omega) = 0$ : Neste caso, teremos que

$$\begin{aligned} 1 + \delta(\omega) &= [1 + \delta_m(\omega)][\delta_b(\omega)] + [1 + \delta_{mc}(\omega)][1 - \delta_b(\omega)] \\ 1 + \delta(\omega) &= \delta_b(\omega) + \delta_m(\omega)\delta_b(\omega) + 1 + \delta_{mc}(\omega) - \delta_b(\omega) - \delta_{mc}(\omega)\delta_b(\omega) \end{aligned} \quad (\text{A.6})$$

ou seja,

$$\delta(\omega) = \delta_{mc}(\omega) + \delta_m(\omega)\delta_b(\omega) - \delta_{mc}(\omega)\delta_b(\omega) \quad (\text{A.7})$$

Analogamente ao caso (a), a margem, neste caso, só é devida aos termos de segunda ordem. Assim, para este caso, o *ripple* do filtro de mascaramento negativo é o que domina na função acima, e teremos os mesmos limites do caso (a).

- (c) Se  $0 < G_b(\omega) < 1$  : Neste caso, o *ripple* máximo será a soma de uma parte de  $\delta_m$  e de uma parte de  $\delta_{mc}$ , resultando no maior valor entre estes dois desvios.

2. Para  $G_m(\omega) = 0$  e  $G_{mc}(\omega) = 0$ , isto é, para a faixa de freqüência em que o filtro desejado apresenta ganho zero, devido à soma das respostas do filtro interpolado e seu complementar. Neste caso,  $G(\omega) = 0$ , e para  $G_b(\omega) = 1$ , teremos

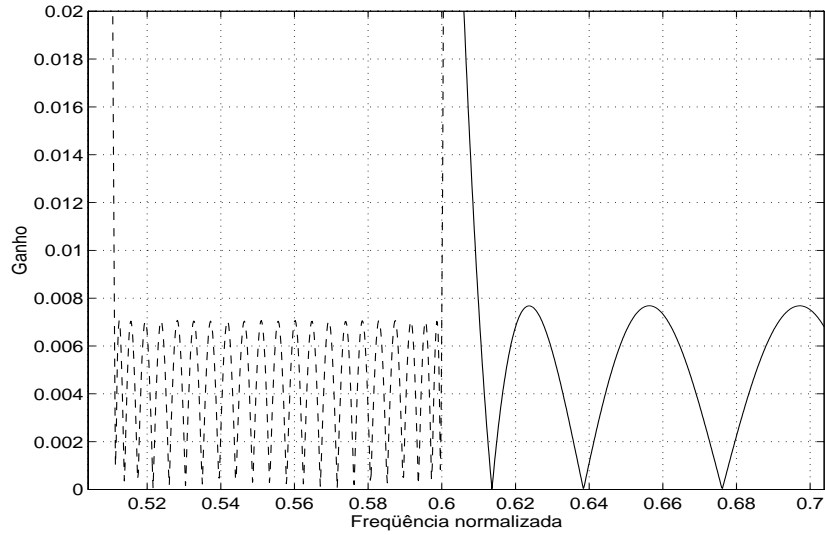
$$\begin{aligned}\delta(\omega) &= [\delta_m(\omega)][1 + \delta_b(\omega)] + [\delta_{mc}(\omega)][-\delta_b(\omega)] \\ \delta(\omega) &= \delta_m(\omega) + \delta_b(\omega)\delta_m(\omega) - \delta_{mc}(\omega)\delta_b(\omega)\end{aligned}\tag{A.8}$$

que é o mesmo caso do item (a), anterior. Analogamente, para  $G_b(\omega) = 0$ , teremos o mesmo caso do item (b), e, para  $0 < G_b(\omega) < 1$ , o mesmo caso do item (c).

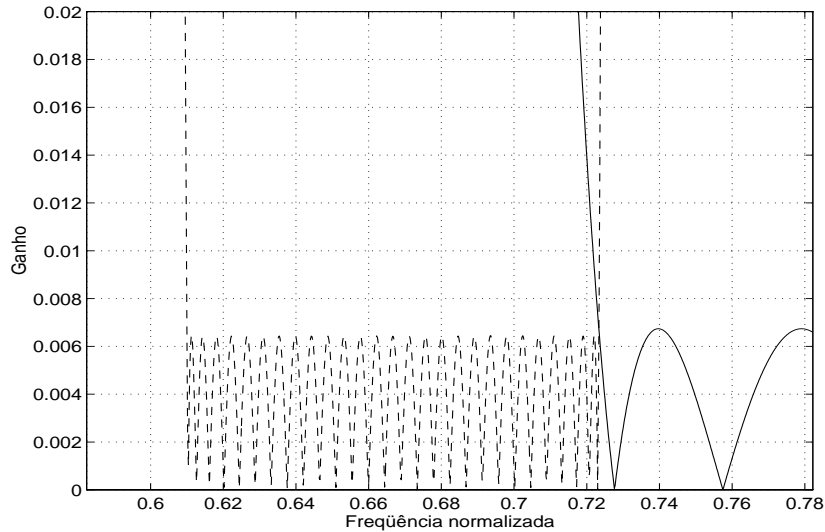
3. Para o caso em que  $0 < G_m(\omega) < 1$  e  $G_{mc}(\omega) = 1$ , e  $G_m(\omega) = 0$  e  $0 < G_{mc}(\omega) < 1$ , que representa o final da banda passante do filtro (última banda do filtro interpolado ou de seu complementar que não é descartada pela operação de mascaramento), e o início da banda de rejeição do filtro final (até que o filtro de máscara de freqüência de corte mais alta esteja no corte). Neste caso, estamos admitindo que  $G_m(\omega)$  possui freqüência de corte inferior a  $G_{mc}(\omega)$ , mas se considerarmos o contrário, chegaremos ao mesmo resultado.

Para  $0 < G_{mc}(\omega) < 1$  e  $G_m(\omega) = 1$  (consideramos  $\delta_{mc}(\omega) = 0$ , neste caso, já que  $G_{mc}$  está em transição). Neste caso,  $G_b(\omega) = 1$  (e, portanto, o filtro interpolado complementar apresenta resposta igual a  $1 - (G_b(\omega) + \delta_b(\omega)) = -\delta_b(\omega)$ ),  $G(\omega) = 0$ ,  $\delta(\omega)$  representa a atenuação do filtro final e  $\delta_{PROJ}$  é a atenuação máxima de rejeição permitida, e temos:

$$\begin{aligned}\delta(\omega) &= \delta_m(\omega)[1 + \delta_b(\omega)] - G_{mc}(\omega)\delta_b(\omega) \\ \delta(\omega) &= \delta_m(\omega) + \delta_m(\omega)\delta_b(\omega) - G_{mc}(\omega)\delta_b(\omega)\end{aligned}\tag{A.9}$$

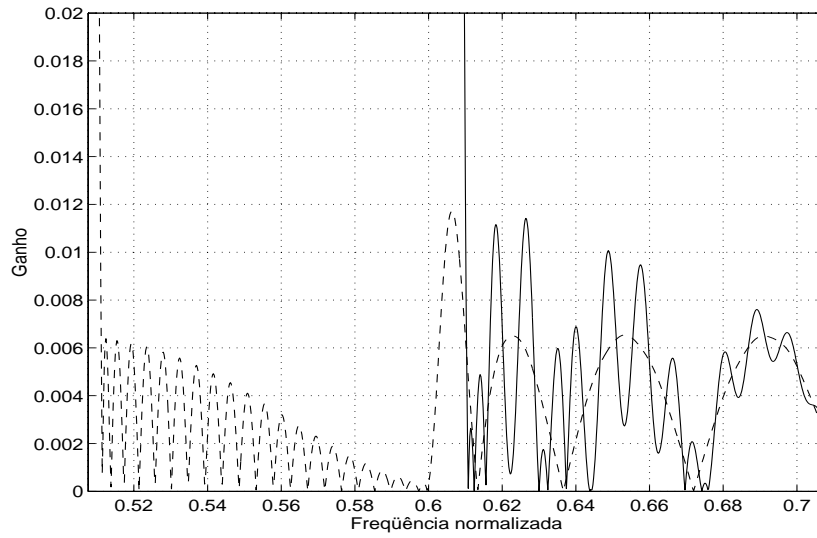


(a) Faixa de corte do filtro de mascaramento anterior. Este filtro de máscara começa a cortar em  $\omega_p = 0.5111$ .

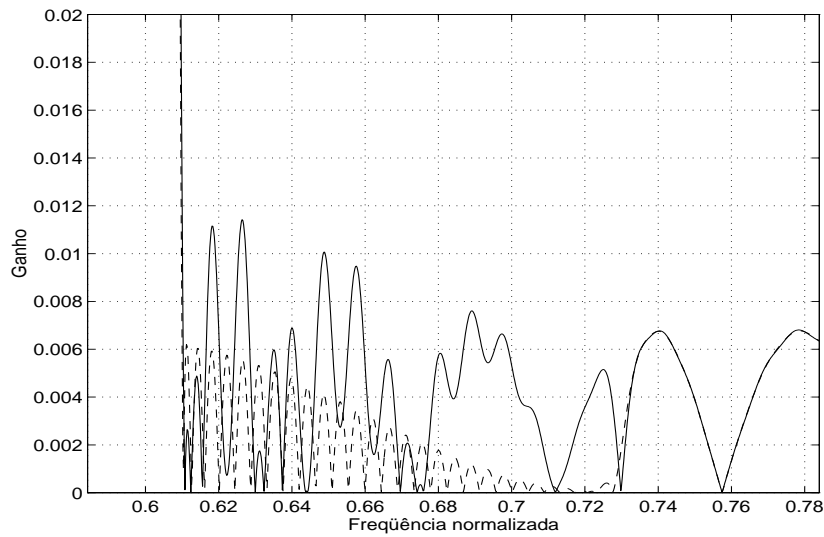


(b) Faixa de corte do filtro de mascaramento posterior. Este filtro de máscara começa a cortar em  $\omega_p = 0.61$ .

Figura A.2: Detalhe das faixas de transição dos filtros de mascaramento. As linhas pontilhadas representam os filtros interpolado e interpolado complementar, e as linhas contínuas representam os filtros de (a) máscara positiva e (b) máscara negativa.



(a) Faixa de transição do filtro de mascaramento anterior (de 0.5111 a 0.6).



(b) Faixa de transição do filtro de mascaramento posterior (de 0.61 a 0.73).

Figura A.3: Resposta de cada ramo dos filtros (linhas tracejadas). A linha contínua representa a resposta do filtro final. Note que no começo da faixa 0.61 a 0.73, o *ripple* dos ramos nos dois gráficos é alto, necessitando de maior margem. O mesmo acontece no início da faixa de 0.5111 a 0.6, no *ripple* da faixa de passagem do filtro final.



O pior caso desta função ocorrerá se  $G_{mc}(\omega)$  for igual a um e se  $\delta_b$  for mínimo, e assim teremos:

$$\delta(\omega) = \delta_m(\omega) + \delta_m(\omega)\delta_b(\omega) - \delta_b(\omega) \quad (\text{A.10})$$

aplicando a margem aos desvios (lembrando que se  $\delta_b(\omega)$  é mínimo, então será igual a  $-\alpha\delta_{PROJ}$ , no pior caso), temos:

$$\delta_{PROJ} = \alpha\delta_{PROJ} - \alpha^2[\delta_{PROJ}]^2 + \alpha\delta_{PROJ} \Rightarrow \alpha = \frac{1 - \sqrt{1 - \delta_{PROJ}}}{\delta_{PROJ}} \quad (\text{A.11})$$

Assim, para  $\delta_{PROJ} = 0.01$  (-40 dB) a margem será de 50,1%; Para  $\delta_{PROJ} = 0.1$  (-20 dB), a margem de ripple será 51%, e para  $\delta_{PROJ} = 0.0001$  (-80 dB), a margem de ripple será de 50%. Portanto, este valor é sempre aproximadamente 50%, qualquer que seja o desvio máximo especificado, como já foi visto anteriormente. Na verdade, se ignorarmos o termo de segunda ordem  $\delta_m(\omega)\delta_b(\omega)$  nas equações acima, chegaremos ao resultado  $\alpha = \frac{1}{2} = 50\%$ . O mesmo ocorre para  $G_{mc}(\omega) = 1$  e  $0 < G_{mc}(\omega) < 1$ . Entretanto, calculando a margem, mas levando em consideração o valor  $G_{mc}(\omega) = k$ , no cálculo acima, temos que

$$\alpha = \frac{(1 + k) - \sqrt{(1 + k)^2 - 4\delta_{PROJ}}}{2\delta_{PROJ}} \quad (\text{A.12})$$

ou seja, podemos calcular o quanto de margem seria necessária para  $k$  variando de 0 a 1. Assim, notamos que, com  $k = 0.5$ , por exemplo, a margem necessária seria de apenas 33%, segundo o cálculo; isto é, após  $G_m(\omega)$  ter decrescido para 0.5, a margem utilizada pelos desvios é de 33% do desvio total deixado no projeto. Portanto, a margem alta só ocorre no início das faixas de transição dos filtros de mascaramento (como mostram as Figuras A.2 e A.3). Note que a soma das respostas dos dois ramos é alta, na faixa  $0.61 \leq \omega \leq 0.66$ , o que compromete a performance do filtro final obtido. Para compensarmos o efeito dessa margem extra necessária apenas quando  $0.5 < G_{mc}(\omega) < 1$ , podemos fazer uma compensação nos filtros de mascaramento, de modo que o início da banda de rejeição de cada um dos filtros de mascaramento apresentem a margem necessária (50%) para que o filtro atenda às especificações. Isto pode ser realizado aumentando-se o peso de cálculo nesta banda, em relação às outras, nos filtros de mascaramento. Assim, suas respostas apresentarão um *ripple* menor nesta banda, e apresentarão maior *ripple* nas outras faixas, em

que a margem requerida é de 10 a 15%. Esta compensação funciona melhor no projeto com bandas não críticas, pois é muito simples alterar o peso da primeira banda de corte dos filtros de mascaramento por um peso maior.

# Apêndice B

## Exemplo de construção de sinais nos filtros CMFRMs

Iremos exemplificar aqui a construção no tempo dos sinais em um CMFRM. Para estes exemplos, iremos adotar  $L = M = 2$ , resultando que  $Q = 2$ , e, portanto, estaremos tratando das estruturas das Figuras 6.6, 6.9 e 6.11. Primeiramente iremos demonstrar os sinais em um exemplo de CMFB, e em seguida iremos aplicar o mesmo sinal de entrada em um CMFRM equivalente. Na seção B.1, analisaremos os sinais na estrutura generalizada, e na seção B.2 analisaremos os sinais para a estrutura simplificada, que ocorre quando  $L = M$ .

### B.1 Análise da estrutura CMFRM generalizada

Para a análise da estrutura CMFRM generalizada, utilizaremos os filtros componentes da Subseção B.1.1, seguido das análises das Subseções B.1.2, para o CMFB, e B.1.3, para o CMFRM equivalente. Verificaremos, com isto, que os sinais de entrada das matrizes pré-moduladoras são equivalentes, e portanto os filtros CMFB e CMFRM são intercambiáveis. As demonstrações para o banco de síntese são similares e não serão apresentadas aqui.

### B.1.1 Filtros componentes

Para este exemplo de filtro CMFRM com  $L = M = 2$ , o filtro base é dado pela resposta impulsiva

$$h_b(n) = \{1 \ 2 \ 3\} \quad (\text{B.1})$$

e, portanto o filtro interpolado por um fator  $L = 2$  será

$$h_b^I(n) = \{1 \ 0 \ 2 \ 0 \ 3\} \quad (\text{B.2})$$

O filtro de máscara positiva é dado pela resposta impulsiva

$$g_1(n) = \{1 \ 2 \ 3 \ 4\} \quad (\text{B.3})$$

e o filtro de máscara negativa é nulo. O filtro protótipo equivalente deste filtro FRM é dado pela convolução das seqüências  $g_1(n)$  e  $h_b^I(n)$  acima, e será, portanto

$$h_p(n) = \{1 \ 2 \ 5 \ 8 \ 9 \ 14 \ 9 \ 12\} \quad (\text{B.4})$$

Com isto, teremos para o projeto CMFB padrão, o filtro  $h_p(n)$  acima, que possui  $2KM = 8$  coeficientes, com  $K = 2$ . Para o filtro CMFRM, usaremos os filtros  $h_b(n)$  e  $g_1(n)$ . Para as análises, estaremos utilizando o sinal de entrada  $x(n)$  dado por

$$x(n) = \{0 \ 1 \ 1 \ 0 \ \dots\} \quad (\text{B.5})$$

ou seja, possui dois elementos não nulos, seguidos de zeros.

### B.1.2 Projeto CMFB

A estrutura do CMFB pode ser vista nas Figuras B.1 e B.2, onde incluímos as indicações dos sinais nos vários nós do sistema. De acordo com a decomposição polifásica do filtro  $H_p(z)$  em  $2M = 4$  componentes, teremos as componentes  $E_0(z)$  a  $E_3(z)$

$$E_0(z) = 1 + 9z^{-1} \quad (\text{B.6})$$

$$E_1(z) = 2 + 14z^{-1} \quad (\text{B.7})$$

$$E_2(z) = 5 + 9z^{-1} \quad (\text{B.8})$$

$$E_3(z) = 8 + 12z^{-1} \quad (\text{B.9})$$

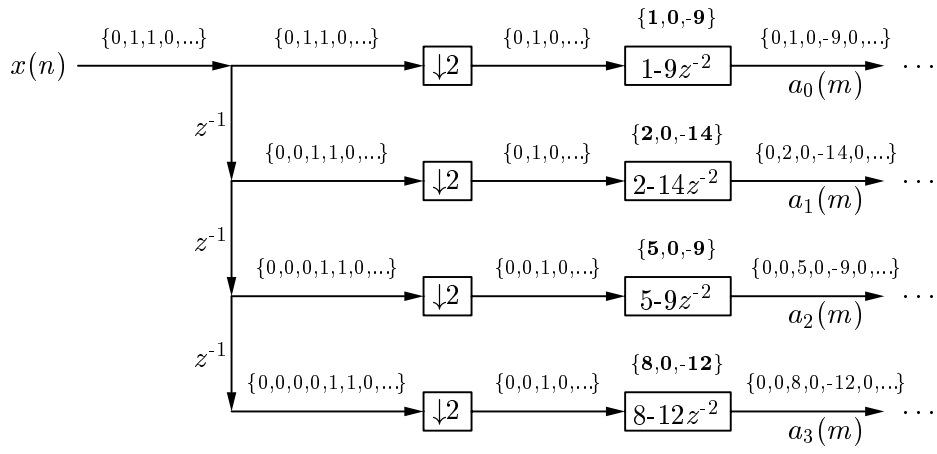


Figura B.1: Sinais nas várias componentes polifásicas para o exemplo de filtro CMFB.

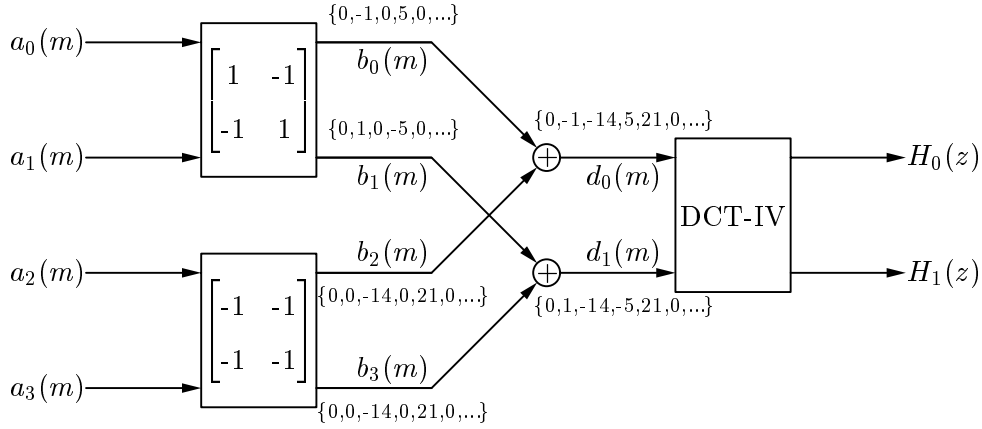


Figura B.2: Sinais nas matrizes moduladoras para o exemplo do CMFB.

e, portanto, os filtros após os decimadores serão dados por:

$$E_0(-z^2) = 1 - 9z^{-2} \quad (\text{B.10})$$

$$E_1(-z^2) = 2 - 14z^{-2} \quad (\text{B.11})$$

$$E_2(-z^2) = 5 - 9z^{-2} \quad (\text{B.12})$$

$$E_3(-z^2) = 8 - 12z^{-2} \quad (\text{B.13})$$

e, portanto, teremos as seqüências de sinais mostradas nas Figuras B.1 e B.2, excluindo-se a constante multiplicativa na entrada do banco de filtros. Note que para este exemplo,  $K = 2$ , resultando nas matrizes pré-moduladoras da Figura B.2.

### B.1.3 Projeto CMFRM equivalente

Neste caso, teremos a estrutura da Figura 6.9 ou 6.11 com  $Q = 2$ , e o sinal de entrada  $x(n)$  é aplicado às duas componentes polifásicas do filtro de base na entrada do filtro. Estas componentes são dadas por:

$$h_{b0}(n) = \{1 \ 3\} \Rightarrow h'_{b0}(n) = \{1 \ 0 \ -3\} \quad (\text{B.14})$$

$$h_{b1}(n) = \{2\} \Rightarrow h'_{b1}(n) = \{2\} \quad (\text{B.15})$$

e, portanto, as versões interpoladas destes filtros serão:

$$h'_{b0}{}^I(n) = \{1 \ 0 \ 0 \ 0 \ -3\}; \quad h'_{b1}{}^I(n) = \{2\} \quad (\text{B.16})$$

O filtro de máscara, neste caso, é decomposto em  $2M = 4$  componentes polifásicas, cada uma com apenas um único coeficiente, ou seja

$$\begin{aligned} E_0(-z^{2Q}) = E_0(z) = 1; \quad E_1(-z^{2Q}) = E_1(z) = 2; \\ E_2(-z^{2Q}) = E_2(z) = 3; \quad E_3(-z^{2Q}) = E_3(z) = 4 \end{aligned} \quad (\text{B.17})$$

e, assim, teremos os sinais mostrados nas Figuras B.3 e B.4. Note que os sinais de entrada para as matrizes pré-moduladoras, isto é,  $a_0(m), \dots, a_3(m)$  são idênticos, a menos de um atraso, resultante da estrutura multiplicativa da Figura 6.7. Portanto, para esta estrutura, as matrizes pré-moduladoras e a DCT-IV serão idênticas às da estrutura CMFB equivalente, ou seja, o valor de  $K$  equivalente, nesta estrutura, é 2. Igualmente, se utilizarmos a constante multiplicativa na entrada do banco de filtros, esta seria idêntica à constante da estrutura CMFB.

O atraso nos sinais  $a_0(m), \dots, a_3(m)$  pode ser retirado, se notarmos que podemos deslocar o atraso em  $z^{-2}$ , resultante da estrutura multiplicativa, na Figura B.4, para a entrada do sistema (isto é, antes da filtragem por  $H_{b0}(z)$ ). Este atraso e o atraso da segunda componente polifásica,  $H_{b1}(z)$ , podem ser vistos como um único atraso de  $z^{-2}$  na entrada do sistema, e pode, portanto, ser retirado. Para o caso geral, o atraso  $z^{-Q}$  sempre pode ser passado para a entrada do sistema e pode ser retirado, juntamente com o mesmo atraso  $z^{-Q}$  da primeira linha de atrasos, isto é, da linha de atrasos da decomposição polifásica do filtro de base.

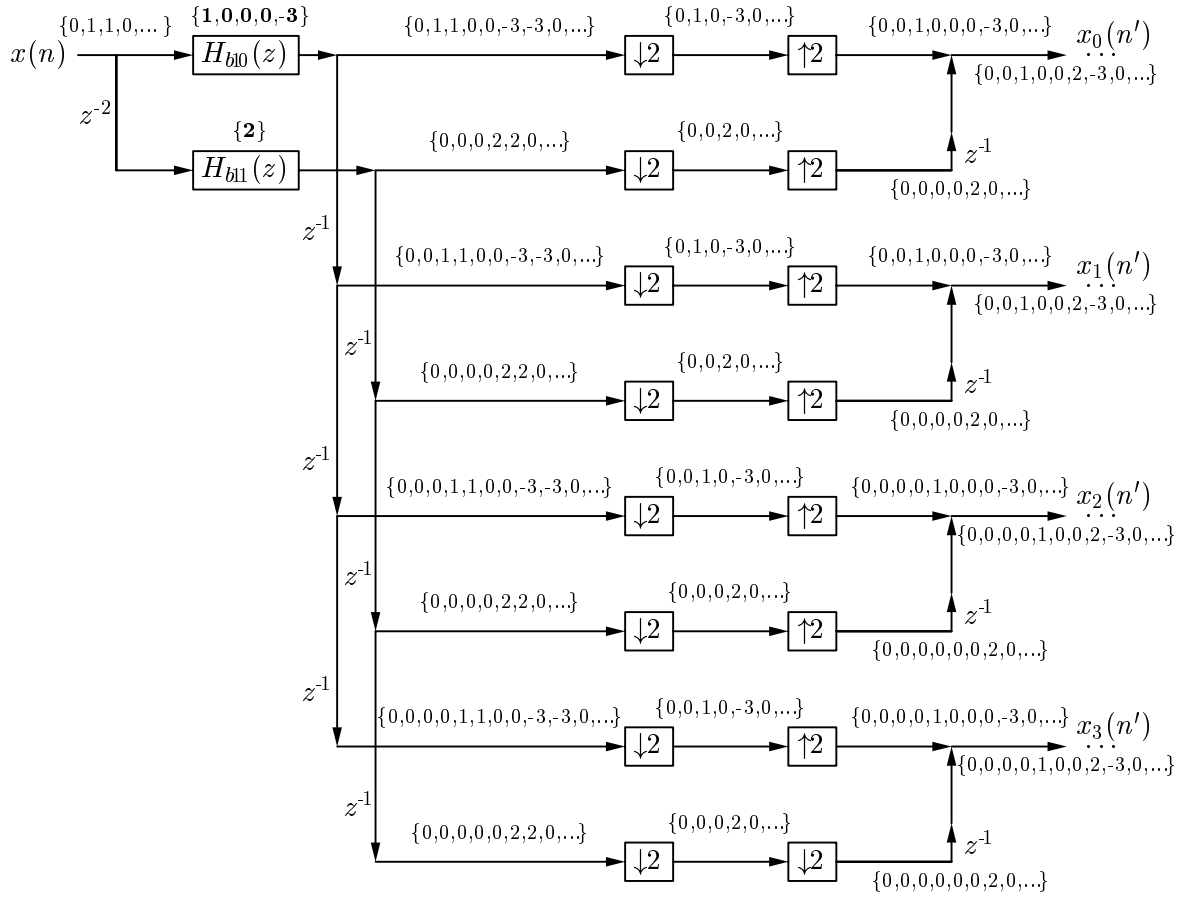


Figura B.3: Sinais na estrutura CMFRM equivalente.

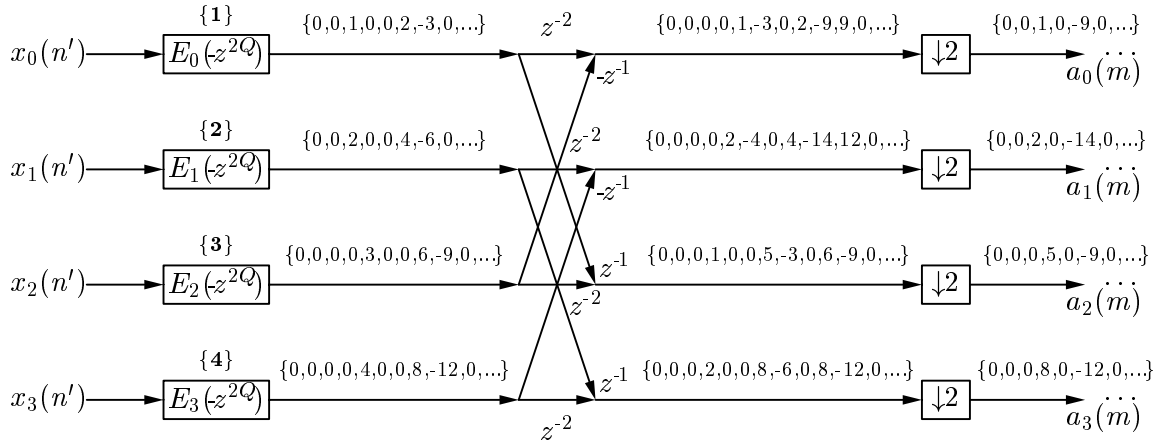


Figura B.4: Sinais preparados para a entrada das matrizes pré-moduladoras na estrutura CMFRM equivalente.

## B.2 Análise da estrutura simplificada para o caso

$$L = M$$

Para a análise da estrutura CMFRM simplificada, utilizaremos os filtros descritos na Subseção B.2.1, e as análises das Subseções B.2.2, para o CMFB, e B.2.3 para o CMFRM equivalente. Neste caso, os sinais de saída das matrizes pré-moduladoras (isto é, antes da operação da DCT-IV) serão equivalentes, e portanto os filtros CMFB e CMFRM serão intercambiáveis. As demonstrações para o banco de síntese são similares e não serão apresentadas aqui.

### B.2.1 Filtros componentes

Para este exemplo de filtro CMFRM com  $L = M = 2$ , o filtro base é dado pela resposta impulsiva

$$h_b(n) = \{1 \ 2 \ 3\} \quad (\text{B.18})$$

e, portanto o filtro interpolado por um fator  $L = 2$  será

$$h_b^I(n) = \{1 \ 0 \ 2 \ 0 \ 3\} \quad (\text{B.19})$$

O filtro de máscara positiva necessita ter resposta impulsiva simétrica para o uso da estrutura simplificada, sendo, portanto

$$g_1(n) = \{1 \ 2 \ 2 \ 1\} \quad (\text{B.20})$$

e o filtro de máscara negativa é nulo. O filtro protótipo equivalente deste filtro FRM é dado pela convolução das seqüências  $g_1(n)$  e  $h_b^I(n)$  acima, e será dado por

$$h_p(n) = \{1 \ 2 \ 4 \ 5 \ 7 \ 8 \ 6 \ 3\} \quad (\text{B.21})$$

Com isto, teremos para o projeto CMFB padrão, o filtro  $h_p(n)$  acima, que possui  $2KM = 8$  coeficientes, com  $K = 2$ . Para o filtro CMFRM, usaremos os filtros  $h_b(n)$  e  $g_1(n)$ . Para as análises, estaremos utilizando o sinal de entrada  $x(n)$  dado por

$$x(n) = \{0 \ 1 \ 1 \ 0 \ \dots\} \quad (\text{B.22})$$

ou seja, possui dois elementos não nulos, seguidos de zeros.



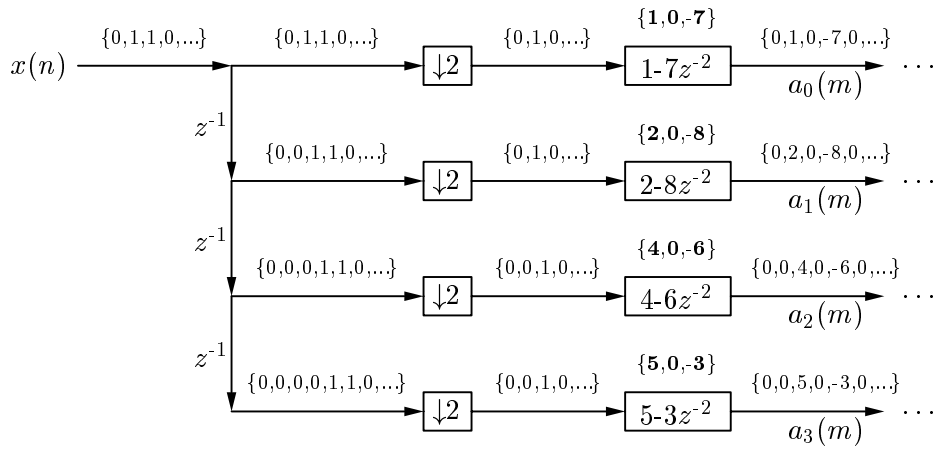


Figura B.5: Sinais nas várias componentes polifásicas para o exemplo de filtro CMFB.

## B.2.2 Projeto CMFB

A estrutura do CMFB pode ser vista nas Figuras B.5 e B.6, onde incluímos as indicações dos sinais nos vários nós do sistema. De acordo com a decomposição polifásica do filtro  $H_p(z)$  em  $2M = 4$  componentes, teremos as componentes  $E_0(z)$  a  $E_3(z)$

$$E_0(z) = 1 + 7z^{-1} \quad (\text{B.23})$$

$$E_1(z) = 2 + 8z^{-1} \quad (\text{B.24})$$

$$E_2(z) = 4 + 6z^{-1} \quad (\text{B.25})$$

$$E_3(z) = 5 + 3z^{-1} \quad (\text{B.26})$$

e, portanto, os filtros após os decimadores serão dados por:

$$E_0(-z^2) = 1 - 7z^{-2} \quad (\text{B.27})$$

$$E_1(-z^2) = 2 - 8z^{-2} \quad (\text{B.28})$$

$$E_2(-z^2) = 4 - 6z^{-2} \quad (\text{B.29})$$

$$E_3(-z^2) = 5 - 3z^{-2} \quad (\text{B.30})$$

e, portanto, teremos as seqüências de sinais mostradas nas Figuras B.5 e B.6, excluindo-se a constante multiplicativa na entrada do banco de filtros. Note novamente que para este exemplo,  $K = 2$ .

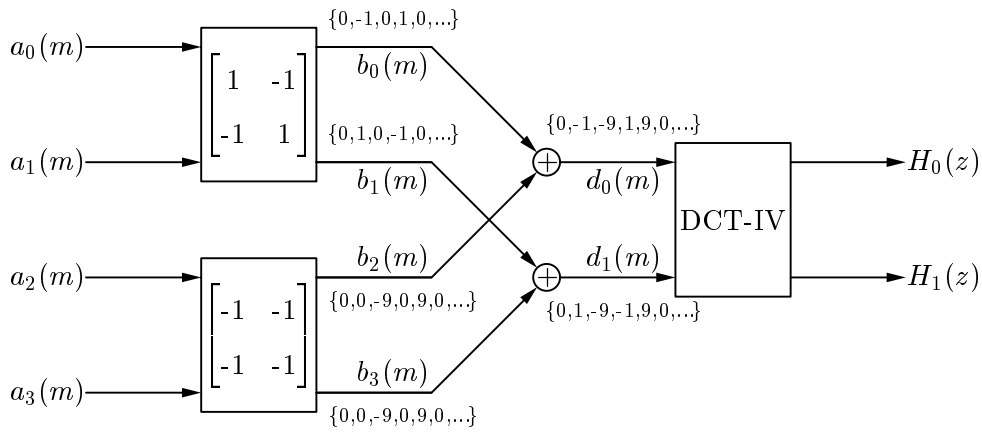


Figura B.6: Sinais nas matrizes moduladoras para o exemplo do CMFB.

### B.2.3 Projeto CMFRM equivalente

Neste caso, teremos a estrutura da Figura 6.6, e o sinal de entrada  $x(n)$  é aplicado às duas componentes polifásicas do filtro de base na entrada do filtro. Estas componentes são dadas por:

$$h_{b10}(n) = \{1 \ 3\} \Rightarrow h'_{b10}(n) = \{1 \ 0 \ -3\} \quad (\text{B.31})$$

$$h_{b11}(n) = \{2\} \Rightarrow h'_{b11}(n) = \{2\} \quad (\text{B.32})$$

e, portanto, as versões interpoladas destes filtros serão:

$$h'_{b10}{}^I(n) = \{1 \ 0 \ 0 \ 0 \ -3\}; \quad h'_{b11}{}^I(n) = \{2\} \quad (\text{B.33})$$

O filtro de máscara, neste caso, é decomposto em  $2M = 4$  componentes polifásicas, cada uma com apenas um único coeficiente, ou seja

$$\begin{aligned} E_0(-z^2) = E_0(z) = 1; \quad E_1(-z^2) = E_1(z) = 2; \\ E_2(-z^2) = E_2(z) = 2; \quad E_3(-z^2) = E_3(z) = 1 \end{aligned} \quad (\text{B.34})$$

ou seja, as componentes polifásicas do filtro de máscara apresentam simetria e, assim, teremos os sinais mostrados nas Figuras B.7 e B.8. Note que os sinais de entrada para a matriz moduladora DCT-IV, isto é,  $d_0(m)$  e  $d_1(m)$  são idênticos, e portanto a estrutura CMFRM é equivalente à do CMFB.

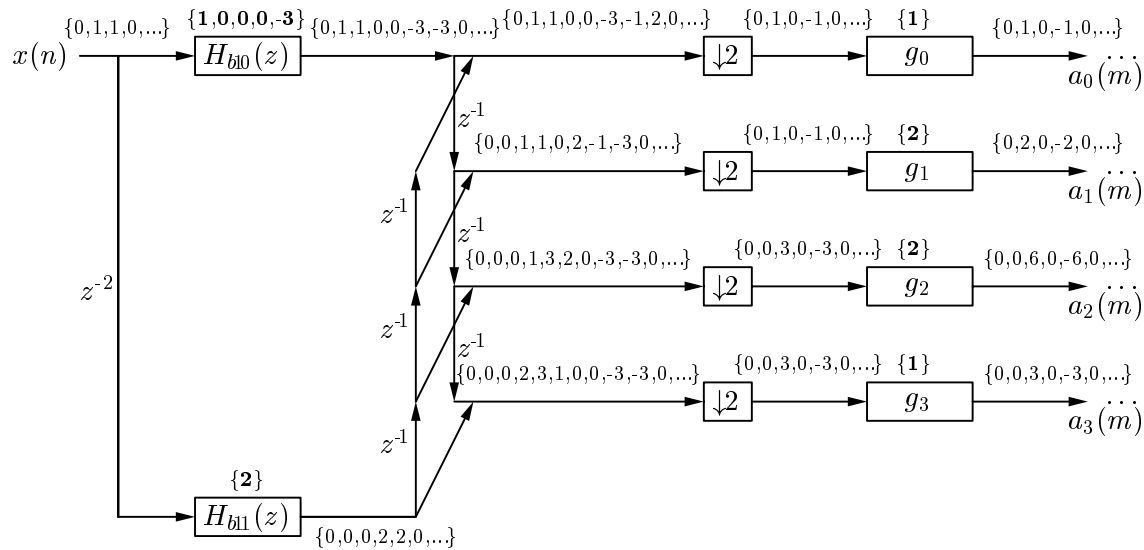


Figura B.7: Sinais na estrutura CMFRM simplificada.

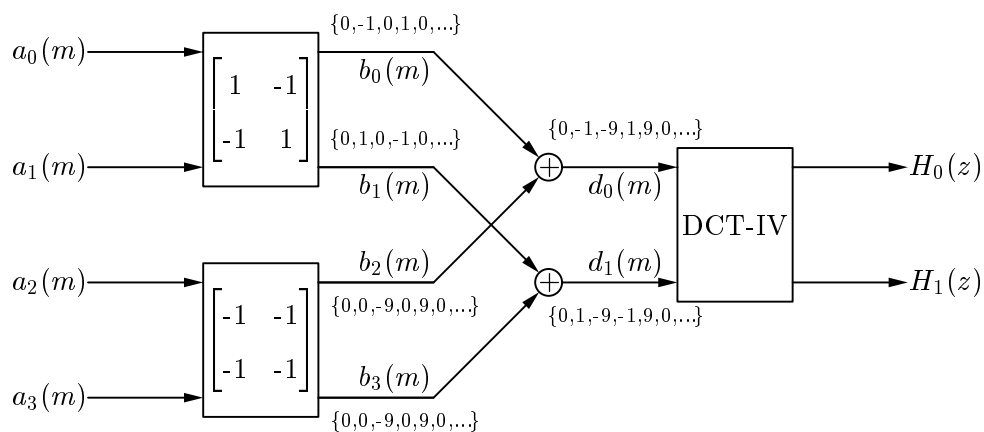


Figura B.8: Sinais nas matrizes moduladoras da CMFRM simplificada.

# Apêndice C

## Reconstrução perfeita para o CMFRM utilizando uma variável a ser otimizada

Uma das questões importantes a ser levantada quando se trabalha com o CMFRM, é a de saber se o banco de filtros, uma vez realizado por um filtro protótipo que seja uma cascata de dois filtros, como é no caso do FRM, permite reconstrução perfeita, ou seja, se as restrições necessárias à reconstrução perfeita podem ser mantidas mesmo sabendo que o filtro protótipo possui suas próprias restrições devido ao método de interpolação e de mascaramento. Por exemplo, um filtro com número ímpar de canais não permite que um FRM com apenas um filtro de máscara (um único ramo na estrutura FRM) seja um filtro protótipo cujo banco apresente reconstrução perfeita. Isto pode ser visto facilmente se levarmos em conta que o coeficiente central necessita ser  $1/(2M)$  e que os coeficientes a cada  $kM$  amostras necessitam ser nulos no caso de número ímpar de canais no sistema [13]. Ou seja, não seria possível obter-se um filtro interpolado e um de máscara, tal que quando cascateados respeitem esta restrição.

Embora nesta tese não tenhamos abordado o problema de otimização nem os casos em que se obtenha a reconstrução perfeita, iremos agora exemplificar como, em certos casos, é possível obter bancos de filtros CMFRM com reconstrução perfeita. Os exemplos dados aqui são simples (para uma única variável a ser otimizada), e entretanto mostram como é possível obter-se a redução do número de variáveis a

serem otimizadas. Assim, na Seção C.1, exemplificaremos um filtro de reconstrução perfeita com 2 canais e  $K = 2$ , enquanto que na Seção C.2, exemplificaremos um filtro com 2 canais, derivado do caso anterior, mas com  $K = 3$ .

## C.1 Reconstrução perfeita com 2 canais e $K = 2$

Para a reconstrução perfeita com 2 canais, e utilizando  $K = 2$  no projeto CMFB padrão, teremos os polinômios  $P(z)$  e  $Q(z)$  das equações (5.47) e (5.48) da reconstrução perfeita dados por

$$P^{(1)}(z) = \sin(\phi_1) \quad (\text{C.1})$$

$$Q^{(1)}(z) = \cos(\phi_1) \quad (\text{C.2})$$

$$P^{(2)}(z) = \sin(\phi_2) \sin(\phi_1) + z^{-1} \cos(\phi_2) \cos(\phi_1) \quad (\text{C.3})$$

$$Q^{(2)}(z) = \cos(\phi_2) \sin(\phi_1) - z^{-1} \sin(\phi_2) \cos(\phi_1) \quad (\text{C.4})$$

e, portanto, utilizando as equações (5.53) a (5.56), teremos

$$g_p(1) = p(1) = \cos(\phi_2) \cos(\phi_1) \quad (\text{C.5})$$

$$g_p(2) = q(0) = \cos(\phi_2) \sin(\phi_1) \quad (\text{C.6})$$

$$g_p(3) = q(1) = -\sin(\phi_2) \cos(\phi_1) \quad (\text{C.7})$$

$$g_p(4) = p(0) = \sin(\phi_2) \sin(\phi_1) \quad (\text{C.8})$$

e os  $2KM = 8$  coeficientes do filtro protótipo serão dados por

$$h_p(0) = h_p(7) = g_p(4)/(M\sqrt{2}) \quad (\text{C.9})$$

$$h_p(1) = h_p(6) = g_p(3)/(M\sqrt{2}) \quad (\text{C.10})$$

$$h_p(2) = h_p(5) = g_p(2)/(M\sqrt{2}) \quad (\text{C.11})$$

$$h_p(3) = h_p(4) = g_p(1)/(M\sqrt{2}) \quad (\text{C.12})$$

e, portanto, em um CMFB padrão, teremos duas variáveis a otimizar no banco de filtros,  $\phi_1$  e  $\phi_2$ . Se utilizarmos um banco CMFRM, deveremos ter os  $2KM$  coeficientes no filtro protótipo equivalente, ou seja, o comprimento da convolução do filtro interpolado e o do filtro de máscara deve ser igual a 8. Podemos, portanto, utilizar um filtro de máscara simétrico que seja

$$g_1 = \left\{ a \quad b \quad b \quad a \right\} \quad (\text{C.13})$$

e um filtro interpolado dado por

$$h_b^I = \left\{ A \quad 0 \quad B \quad 0 \quad A \right\} \quad (\text{C.14})$$

Realizando a convolução e igualando os elementos a  $h_p(n)$ , teremos

$$Aa = \sin(\phi_2) \sin(\phi_1) \quad (\text{C.15})$$

$$Ab = -\sin(\phi_2) \cos(\phi_1) \quad (\text{C.16})$$

$$Ab + Ba = \cos(\phi_2) \sin(\phi_1) \quad (\text{C.17})$$

$$Aa + Bb = \cos(\phi_2) \cos(\phi_1) \quad (\text{C.18})$$

Uma boa escolha será

$$a = \sin(\phi_1) \quad (\text{C.19})$$

$$b = -\cos(\phi_1) \quad (\text{C.20})$$

e, com isto, resolvendo as equações (C.15) e (C.16), teremos

$$A = \sin(\phi_2) \quad (\text{C.21})$$

e, resolvendo as equações (C.17) e (C.18) obteremos

$$B = \frac{\sin(\phi_1 + \phi_2)}{\sin(\phi_1)} = -\frac{\cos(\phi_2 + \phi_1)}{\cos(\phi_1)} \quad (\text{C.22})$$

Para a equação (C.22) seja válida, é necessário que

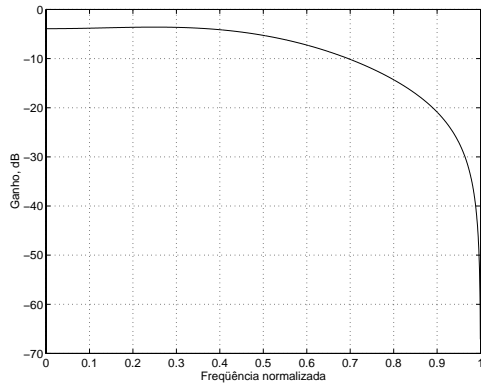
$$\phi_2 = -2\phi_1 \quad (\text{C.23})$$

e, neste caso,  $B = 1$ . Portanto, devido à estrutura FRM, uma das variáveis a serem otimizadas agora se torna dependente da outra, e assim reduzimos o número de variáveis a serem otimizadas. O filtro interpolado e o de máscara obtidos são

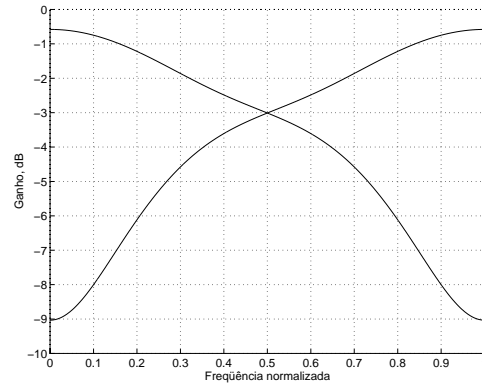
$$h_b^I(n) = \left\{ \sin(-2\phi_1) \quad 0 \quad 1 \quad 0 \quad \sin(-2\phi_1) \right\} \quad (\text{C.24})$$

$$g_1(n) = \left\{ \sin(\phi_1) \quad -\cos(\phi_1) \quad -\cos(\phi_1) \quad \sin(\phi_1) \right\} \quad (\text{C.25})$$

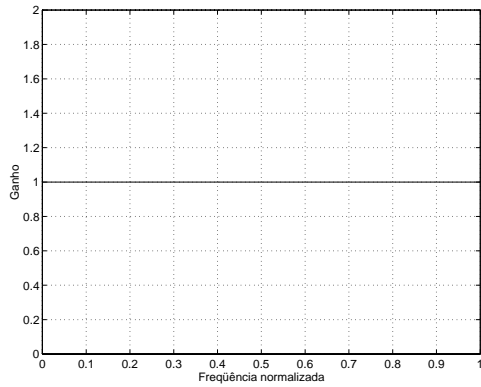
Ajustando a variável  $\phi_1$  para  $0.001\pi$ , obteremos os resultados do banco de filtros CMFRM da Figura C.1. Os valores obtidos para este caso são:



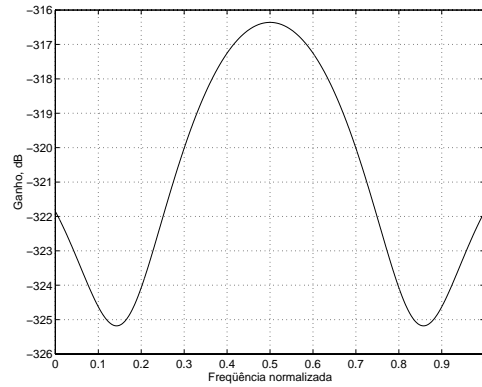
(a) Filtro protótipo obtido.



(b) Resposta dos filtros de análise ou de síntese.



(c) Função  $T_0(z)$ .



(d) Função  $T_{01}(z)$ .

Figura C.1: Resultados obtidos com o filtro CMFRM de reconstrução perfeita utilizando  $M = L = 2$ ,  $K = 2$  e uma única variável a ser otimizada.

$$\omega_{3\text{dB}} = 0.2501\pi \approx 0.25\pi \quad (\text{C.26})$$

$$\delta_1 = 7.6 \times 10^{-14} \approx 0 \quad (\text{C.27})$$

$$\delta_2 = -316 \text{ dB} \approx 0 \quad (\text{C.28})$$

$$A_r = 1.159 \text{ dB} \quad (\text{C.29})$$

$$\text{ISI} = -625 \text{ dB} \approx 0 \quad (\text{C.30})$$

$$\text{ICI} = -625.6 \text{ dB} \approx 0 \quad (\text{C.31})$$

## C.2 Reconstrução perfeita com 2 canais e $K = 3$

Para a reconstrução perfeita com 2 canais, e utilizando  $K = 3$  no projeto CMFB padrão, os polinômios  $P^{(3)}(z)$  e  $Q^{(3)}(z)$  de ordem dois são obtidos a partir dos polinômios  $P^{(2)}(z)$  e  $Q^{(2)}(z)$  do exemplo anterior e são dados por

$$\begin{aligned} P^{(3)}(z) = & \sin(\phi_3) \sin(\phi_2) \sin(\phi_1) + z^{-1}[\sin(\phi_3) \cos(\phi_2) \cos(\phi_1) \\ & + \cos(\phi_3) \cos(\phi_2) \sin(\phi_1)] - z^{-2} \cos(\phi_3) \sin(\phi_2) \cos(\phi_1) \end{aligned} \quad (\text{C.32})$$

e

$$\begin{aligned} Q^{(3)}(z) = & \cos(\phi_3) \sin(\phi_2) \sin(\phi_1) + z^{-1}[\cos(\phi_3) \cos(\phi_2) \cos(\phi_1) \\ & - \sin(\phi_3) \cos(\phi_2) \sin(\phi_1)] + z^{-2} \sin(\phi_3) \sin(\phi_2) \cos(\phi_1) \end{aligned} \quad (\text{C.33})$$

e, portanto, utilizando as equações (5.57) a (5.60), teremos

$$g_p(1) = q(1) = \cos(\phi_3) \cos(\phi_2) \cos(\phi_1) - \sin(\phi_3) \cos(\phi_2) \sin(\phi_1) \quad (\text{C.34})$$

$$g_p(2) = p(1) = \sin(\phi_3) \cos(\phi_2) \cos(\phi_1) + \cos(\phi_3) \cos(\phi_2) \sin(\phi_1) \quad (\text{C.35})$$

$$g_p(3) = p(2) = -\cos(\phi_3) \sin(\phi_2) \cos(\phi_1) \quad (\text{C.36})$$

$$g_p(4) = q(0) = \cos(\phi_3) \sin(\phi_2) \sin(\phi_1) \quad (\text{C.37})$$

$$g_p(5) = q(2) = \sin(\phi_3) \sin(\phi_2) \cos(\phi_1) \quad (\text{C.38})$$

$$g_p(6) = p(0) = \sin(\phi_3) \sin(\phi_2) \sin(\phi_1) \quad (\text{C.39})$$



e os  $2KM = 12$  coeficientes do filtro protótipo serão dados por

$$h_p(0) = h_p(11) = g_p(6)/(M\sqrt{2}) \quad (\text{C.40})$$

$$h_p(1) = h_p(10) = g_p(5)/(M\sqrt{2}) \quad (\text{C.41})$$

$$h_p(2) = h_p(9) = g_p(4)/(M\sqrt{2}) \quad (\text{C.42})$$

$$h_p(3) = h_p(8) = g_p(3)/(M\sqrt{2}) \quad (\text{C.43})$$

$$h_p(4) = h_p(7) = g_p(2)/(M\sqrt{2}) \quad (\text{C.44})$$

$$h_p(5) = h_p(6) = g_p(1)/(M\sqrt{2}) \quad (\text{C.45})$$

e, portanto, em um CMFB padrão, teremos três variáveis a otimizar no banco de filtros,  $\phi_1$ ,  $\phi_2$  e  $\phi_3$ . Se utilizarmos um banco CMFRM, deveremos ter o comprimento da convolução do filtro interpolado e o do filtro de máscara igual a 12. Podemos, portanto, utilizar um filtro de máscara simétrico que seja

$$g_1 = \{a \quad b \quad b \quad a\} \quad (\text{C.46})$$

e um filtro interpolado dado por

$$h_b^I = \{A \quad 0 \quad B \quad 0 \quad C \quad 0 \quad B \quad 0 \quad A\} \quad (\text{C.47})$$

Realizando esta convolução e igualando os elementos a  $h_p(n)$ , teremos

$$Aa = g_p(6) \quad (\text{C.48})$$

$$Ab = g_p(5) \quad (\text{C.49})$$

$$Ab + Ba = g_p(4) \quad (\text{C.50})$$

$$Aa + Bb = g_p(3) \quad (\text{C.51})$$

$$Bb + Ca = g_p(2) \quad (\text{C.52})$$

$$Ba + Cb = g_p(1) \quad (\text{C.53})$$

e poderemos escolher o mesmo filtro de máscara do exemplo anterior, isto é

$$a = \sin(\phi_1) \quad (\text{C.54})$$

$$b = \cos(\phi_1) \quad (\text{C.55})$$

e, com isto, resolvendo as equações (C.48) e (C.49), teremos

$$A = \sin(\phi_3) \sin(\phi_2) \quad (\text{C.56})$$

Resolvendo as equações (C.50) e (C.51) e igualando os valores de  $B$  obteremos

$$\frac{\sin(\phi_1 - \phi_3)}{\sin(\phi_1)} = -\frac{\cos(\phi_1 - \phi_3)}{\cos(\phi_1)} \quad (\text{C.57})$$

e, resolvendo as equações (C.52) e (C.53) e igualando os valores de  $C$ , teremos

$$\frac{\sin(\phi_1 + \phi_2)}{\sin(\phi_1)} = -\frac{\cos(\phi_1 + \phi_2)}{\cos(\phi_1)} \quad (\text{C.58})$$

Para a equação (C.57) seja válida, é necessário que

$$\phi_3 = 2\phi_1 \quad (\text{C.59})$$

e para que a equação (C.58) seja válida, é necessário que

$$\phi_2 = -2\phi_1 \quad (\text{C.60})$$

Portanto, novamente, teremos a redução das três variáveis do CMFB em apenas uma do CMFRM. Ajustando a variável  $\phi_1$  para  $0.066\pi$ , obteremos os resultados do banco de filtros CMFRM da Figura C.2. Os valores obtidos neste caso são:

$$\omega_{3\text{dB}} = 0.2473\pi \approx 0.25\pi \quad (\text{C.61})$$

$$\delta_1 = 9.64 \times 10^{-14} \approx 0 \quad (\text{C.62})$$

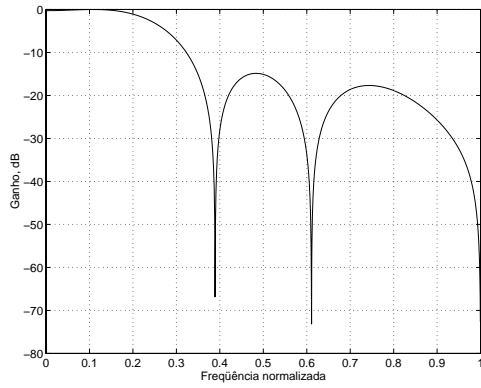
$$\delta_2 = -316 \text{ dB} \approx 0 \quad (\text{C.63})$$

$$A_r \approx 15 \text{ dB} \quad (\text{C.64})$$

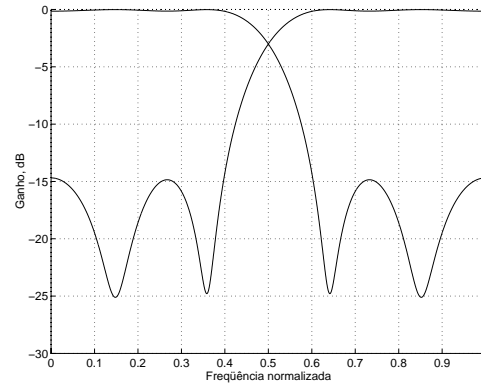
$$\text{ISI} = -606 \text{ dB} \approx 0 \quad (\text{C.65})$$

$$\text{ICI} = -629.1 \text{ dB} \approx 0 \quad (\text{C.66})$$

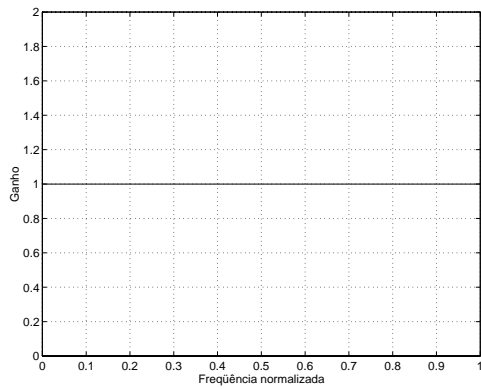
Como podemos notar, com apenas uma variável livre, ou seja, uma única variável a ser otimizada, conseguimos para este exemplo uma atenuação na banda de rejeição de cerca de 15 dB para os filtros do banco, o que é uma atenuação razoável para um CMFB com 2 canais e  $K = 3$ . Concluimos, portanto, que a redução de variáveis pode levar a projetos com boa atenuação, e com maior facilidade na otimização destes filtros.



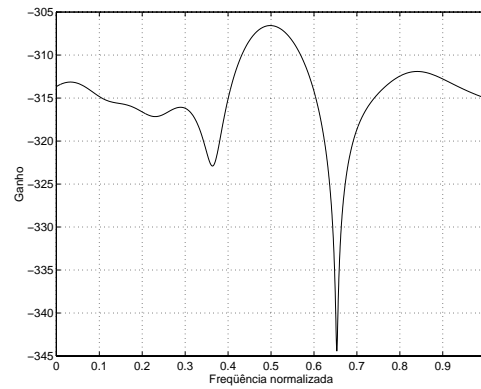
(a) Filtro protótipo obtido.



(b) Resposta dos filtros de análise ou de síntese.



(c) Função  $T_0(z)$ .



(d) Função  $T_{01}(z)$ .

Figura C.2: Resultados obtidos com o filtro CMFRM de reconstrução perfeita utilizando  $M = L = 2$ ,  $K = 3$  e uma única variável a ser otimizada.