Design and Implementation of Discrete-Time Filters for Efficient Sampling Rate Conversion

by

Thomas A. Baran

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2007

Author .....
Department of Electrical Engineering and Computer Science
January 18, 2007

Certified by .....
Alan V. Oppenheim
Ford Professor of Engineering
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Design and Implementation of Discrete-Time Filters for Efficient Sampling Rate Conversion

by

Thomas A. Baran

Submitted to the Department of Electrical Engineering and Computer Science
on January 18, 2007, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering

Abstract

Rate-conversion systems are used in an array of applications, including the oversampled audio and video CODECs often found in entertainment and communications systems. It is common practice for many such systems to sample signals at rates which are much faster than the minimum required to represent some bandwidth of interest, and high-quality filters are often implemented at this fast rate. Therefore, their designs tend to be computationally-expensive. A number of structures have been proposed to address this, including polyphase implementations and folded structures for linear-phase FIR filters. In this thesis, techniques which combine benefits from both classes of structures are discussed, and an efficient class of structures is proposed. The Generalized Transposition Theorem is also reviewed to demonstrate that an efficient downsampling structure also implies an equally efficient, closely-related upsampling structure. Techniques are investigated for designing minimum-multiply filters for the class of structures presented, and methods are discussed for designing filters that, for a given set of frequency domain filter specifications, often require fewer multipliers and have smaller maximum error than Parks-McClellan designs.

Thesis Supervisor: Alan V. Oppenheim
Title: Ford Professor of Engineering

Acknowledgments

I would first like to thank my adviser, Al Oppenheim, for his mentorship and guidance while writing this thesis. Much of the document is a byproduct of the relaxed, open, and exacting research environment which characterizes the Digital Signal Processing Group (DSPG). I look forward to future research and collaboration.

I have had the fortune of working with a great group of people in the DSPG: Ross Bland, Petros Boufounos, Sourav Dey, Zahi Karam, Al Karbouch, Jon Paul Kitchens, Joonsung Lee, Charlie Rohrs, Melanie Rudoy, Maya Said, Joe Sikora, Eric Strattman, Archana Venkataraman, Dennis Wei, and Matthew Willsey. Thank you for your encouragement and eagerness to bounce around ideas. Special thanks go to Sourav, Melanie, and Dennis for your willingness to let me distract you so often with half-baked thoughts. Many of our discussions contributed significantly to this thesis. Eric, thank you for helping to keep the process running smoothly.

To the folks at Atwood's Tavern: thank you for the auxiliary research laboratory. To my roommate Matt Hirsch: our late-night research discussions have been a great help.

Finally, to my family: thank you for being so supportive along the way. Mary, your patience while listening to me try to explain the research has helped many ideas to crystallize. Mom, thank you for your encouragement; it can sometimes be easy to lose sight of past accomplishments when looking forward toward unsolved problems. Dad, your gift for intuitively explaining engineering concepts has been and will continue to be a lifelong influence. Thank you all for being a big part of this.

To my family

Contents

Chapter 1

Introduction

Rate conversion plays a central role in many signal processing settings. Oversampling CODECs and asynchronous signal processors make extensive use of rate conversion, and their implementations often rely on high-rate discrete-time filters. Efforts to reduce the computational cost of these filters have, broadly speaking, occurred on two fronts. One such effort concentrates on improvements in flow graph structures; key results include polyphase implementations and folded structures with time-symmetric impulse responses.[3][4][6][10][15] Another effort involves using filter design techniques, including the Parks-McClellan algorithm and the METEOR toolkit, as methods for choosing efficient filters from some set of permissible designs.[11][12] Drawing on these results, this thesis proposes a class of structures and corresponding FIR filter design techniques. The proposed structures are shown to require fewer multiplications per unit time than polyphase or folded implementations. The proposed design techniques are capable of designing filters which have smaller maximum error than Parks-McClellan designs and which also require fewer multiplications per unit time. Downsampling systems are discussed, and the Generalized Transposition Theorem[4][6] is reviewed as a method for obtaining upsampling systems from these downsampling structures.

For many applications encountered in practice, such as oversampling CODECs, rate conversion by a rational factor is desired, and the required input-output relationship is that of the system in Figure 1-1. Any system which implements the converter in Figure 1-1 will be referred to as a generalized rational rate conversion system. This thesis will discuss methods for efficiently implementing generalized rational rate conversion systems with integer L, M

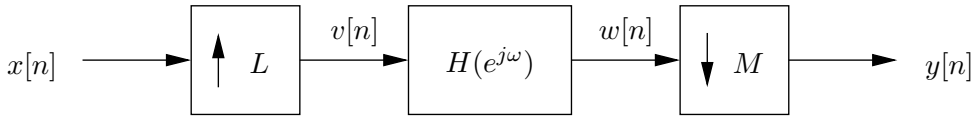


Figure 1-1: Generalized rational rate-conversion system.

and even, time-symmetric, FIR (ETSF) impulse responses $h[n] = \mathcal{F}^{-1}\{H(e^{j\omega})\}$.

The ETSF constraint on the impulse response of $H(e^{j\omega})$ is well-aligned with many applications in practice. This class of filters has linear phase, which is a desirable trait in certain audio and communications settings, for example. Restricting L/M to be rational is common in practice as well. For example, Figure 1-1 for $M = 1$ and integer L is characteristic of the upsampling stage found in many oversampling D/A converters.

1.1 Decomposing the design problem

The process of selecting the filter frequency response $H(e^{j\omega})$ and an appropriate flow graph implementation may be decomposed into two steps. First, a rate-conversion structure is selected which can implement a wide range of filters $H(e^{j\omega})$. This step also implies a correspondence between each implementable filter design and its computational cost. Second, a filter frequency response $H(e^{j\omega})$ is selected which meets the posed system specifications, including limits on computational cost. Exceptions to this two-step method (including techniques for designing multi-stage rate converters)[5][6][14] have been shown to give very efficient implementations, but decomposing the process into two stages has the advantage of resulting in systems which can be easily reconfigured to implement a wide range of $H(e^{j\omega})$.

1.1.1 Choosing a structure

A number of structures have been proposed for implementing systems of the form of Figure 1-1 for filters with ETSF impulse responses. One way to implement this system is to choose a direct-form implementation for $H(e^{j\omega})$. This implementation often requires the computation of many multiplications per unit time. Structures have been proposed to improve upon this, typically by taking advantage of a particular known property of the overall system or of the filter to achieve computational gains.

Folded structures take advantage of ETSF filters to reduce the number of required multiplications per unit time. Because every value taken on by the impulse response $h[n]$

of an ETSF filter occurs a minimum of two times (with the possible exception of the value at the point of symmetry), a folded structure can be used to reduce by approximately one half the required number of multiplications per unit time compared to a direct-form implementation. An example of a folded structure is shown in Figure 1-2. A generalized rational rate-converter may be obtained by adjoining an expander-by-L to the input of this system and a compressor-by-M to its output.

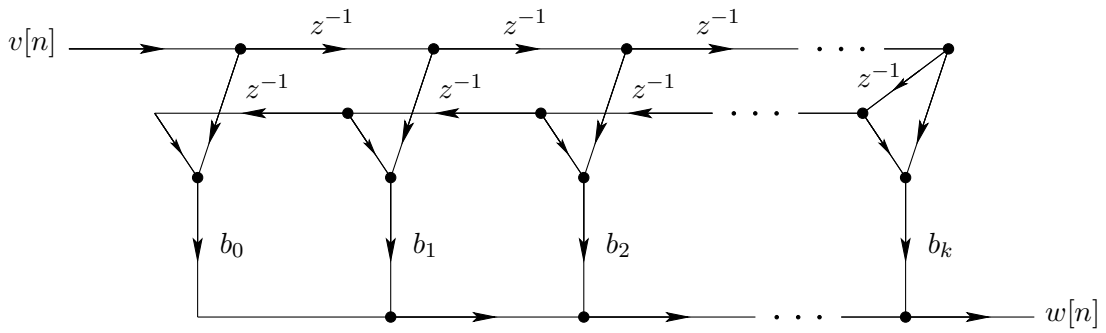


Figure 1-2: Folded implementation of an ETSF filter.

Polyphase structures take advantage of two features: the FIR property of the filter and the fact that the filter falls between an expander and a compressor block in Figure 1-1. When these conditions hold, the response of $H(e^{j\omega})$ can be decomposed into polyphase components $G_\ell(e^{j\omega})$, and the noble identities[15] can be applied to interchange multipliers with compressors or expanders. The resultant structure requires the same number of multipliers as a direct-form implementation, but all multipliers now operate at the sampling rate of either the input $x[n]$ or the output $y[n]$. It is often the case that the sampling rate of either $x[n]$ or $y[n]$ is slower than that of $v[n]$ in Figure 1-1. Consequently, this manipulation gives a system which requires fewer multiplications per unit time than a direct-form implementation. An example of a polyphase implementation for a rational rate converter is shown in Figure 1-3.

Folded implementations for rational rate converters take advantage of filters with time-symmetric impulse responses but do not take advantage of the fact that the filter falls in between an expander and a compressor. Polyphase structures take advantage of this fact, but do not take advantage of time-symmetry. The structures proposed in Chapter 4 take advantage of both properties. They also have the additional property of requiring exactly

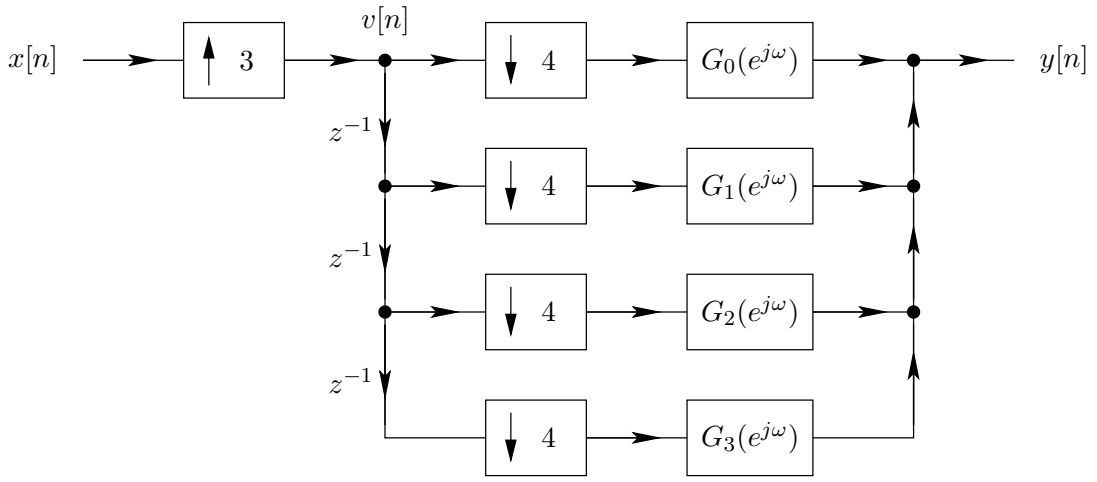


Figure 1-3: Polyphase implementation of a rational rate-conversion system.

one multiplier for each different value taken on by $h[n]$ for all n, independent of the number of times it may repeat.

1.1.2 Choosing a filter frequency response $H(e^{j\omega})$

The Parks-McClellan algorithm is often used for linear-phase FIR filter design, a practice which can perhaps be explained by the attractive error metric which the algorithm minimizes. The algorithm designs a zero-phase filter (or a linear-phase filter by introducing a delay) which minimizes the maximum frequency-domain error between the desired filter and the approximated filter in bands of interest. Specifically, the approximation to the desired filter which minimizes the maximum error is found by solving

$$\min_{\{h_e[n]:0\leq n\leq K\}} \left(\max_{\omega\in F} |E(\omega)| \right), \tag{1.1}$$

where F is the closed subset of $0 \leq \omega \leq \pi$ delineating the bands of interest and $h_e[n]$ is the positive-time part of the zero-phase approximated filter $h[n]$, with DTFT $H(e^{j\omega})$.[10] $E(\omega)$ is defined in terms of the desired response $H_d(e^{j\omega})$, the approximated response $H(e^{j\omega})$ and an error weighting function $W(\omega)$ as

$$E(\omega) = W(\omega)\left[H_d(e^{j\omega}) - H(e^{j\omega})\right]. \tag{1.2}$$

Two aspects of Equation 1.1 are of note here. To begin with, the equation minimizes

the error metric by choosing values of $h_e[n]$ for consecutive samples from $n = 0$ to $n = K$, inclusive. While this formulation does specify the number of required multiplications per unit time when using a given structure, the designed filter is not guaranteed to be the optimal filter with this computational cost across all structures. Using the structures proposed in Chapter 4, for instance, allows for the possibility of obtaining a filter with smaller peak error than a Parks-McClellan design while requiring the same or fewer multiplications per unit time. This possibility arises when the design process is allowed the freedom to pick K possibly nonconsecutive tap locations, in addition to their coefficient values.

Equation 1.1 also specifies that the Parks-McClellan algorithm minimizes the maximum error in bands of interest. For many applications the maximum error may serve as a more appropriate optimality criterion than, for instance, the mean-squared error, but in the common circumstance where a filter must be designed to meet some set of tolerances, the use of a maximum-error-minimizing algorithm less directly addresses the design problem. Furthermore, the fact that the Parks-McClellan algorithm designs a filter with minimal maximum error for a fixed, integer filter length leads to the common situation where filters are over-designed with respect to posed frequency specifications. Bounding frequency deviations instead of trying to minimize them is also an integral concept in formulations for optimization problems where the number of multiplications per unit time is minimized.

Both of these issues can be more directly addressed when the filter design problem is instead formulated in terms of a linear program. The application of linear programming to filter design has been investigated by Steiglitz, Parks, and Kaiser in [12], and the closely-related problem of using linear programming techniques to choose sensor weights for detector arrays has received attention as well.[8] The problem of designing minimum-multiply, constraint-based filters for rate conversion systems is addressed from a linear programming perspective in Chapter 5.

1.2 Outline of thesis

Before discussing methods for designing and implementing efficient rate-conversion systems, some key concepts are underscored. One important notion is the use of the Generalized Transposition Theorem as a method for finding a flow graph for an upsampling system from the flow graph for a downsampling system. This concept is reviewed in Chapter 2 and can

be applied throughout the thesis wherever a downsampling system is discussed. It is for this reason that efficient downsampling structures, only, are proposed in the thesis.

Chapter 3 discusses the relationship between filter designs, structures, and computational cost. This relationship gives rise to a number of subtle issues which become integral points in later chapters, so the topic first receives attention in this chapter.

Because the process of designing efficient rate-conversion systems is decomposed into the stages of selecting efficient structures and choosing efficient filter designs, a chapter is devoted to each. Efficient structures for rate-conversion systems are therefore proposed in Chapter 4, and efficient filter design techniques are discussed in Chapter 5.

Chapter 2

Complementary rate-conversion systems

LTI systems are commonly realized using one of a set of canonical structures, and a correspondence between certain of these structures with equivalent system functions is given by the Transposition Theorem.[9][10] Efficient implementations for rate-conversion systems, however, rely on linear subsystems which are not time-invariant. Many time-varying realizations have been proposed, including structures with time-varying multipliers,[4][6] multistage implementations,[5][6][14] and the well-known polyphase class of structures.[3][6][10][15] While the Transposition Theorem makes no general correspondence between time-varying structures, the Generalized Transposition Theorem[4][6] does relate a time-varying structure implementing a given input-output relationship to a structure implementing its generalized transpose. An arbitrary structure consisting of expanders, compressors, and LTI subsystems furthermore has a generalized transpose which requires the same number of multiplications per unit time as the original structure.[4] The generalized transpose of a rate converter is another rate converter with a reciprocal conversion factor, so the Generalized Transposition Theorem is utilized as a method for obtaining an efficient upsampling structure from a given efficient downsampling structure and vice-versa. The relationship between a rate-conversion system and its generalized transpose is discussed in Section 2.1, and a derivation of the Generalized Transposition Theorem is given in Section 2.2.

2.1 The generalized transpose for rate-conversion systems

An arbitrary linear system with input $x[n]$ and output $y[n]$ is fully characterized by the superposition sum

$$y[n] = \sum_{m=-\infty}^{\infty} g[n,m]x[m], \tag{2.1}$$

where $g[n,m]$ describes this linear homomorphism relating $x[n]$ to $y[n]$. The generalized transpose of $g[n,m]$ is defined as

$$g^T[n,m] \equiv g[-m,-n]. \tag{2.2}$$

Note that $g^{T^T}[n,m] = g[n,m]$.

Two systems with input-output relationships which are generalized transposes of one another are referred to as being complementary. To illustrate the implications of this definition within the context of rate-conversion systems, consider again the generalized rate-conversion system in Figure 1-1. Its respective input and output $x[n]$ and $y[n]$ are related by

$$y[n] = \sum_{m=-\infty}^{\infty} h[Mn-m]x_L[m], \tag{2.3}$$

where $h[n]$ is the LTI impulse response of the filter described by $H(e^{j\omega})$ and where

$$x_L[n] = \begin{cases} x[n/L], & n/L \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}.$$

Equation 2.3 is represented in the form of Equation 2.1 for $g[n,m]$ defined as

$$g[n,m] = \sum_{p=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} C_M[n,p]h[p-q]E_L[q,m], \tag{2.4}$$

where the expander function E_L and compressor function C_M are respectively defined as

$$E_L[n,m] = \begin{cases} 1, & m = \frac{n}{L} \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

and

$$C_M[n,m] = \begin{cases} 1, & m = Mn \\ 0, & \text{otherwise} \end{cases} . \tag{2.6}$$

The generalized transpose of $g[n,m]$ is therefore

$$\begin{aligned} g^T[n,m] &= \sum_{p=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} C_M[-m,p] h[p-q] E_L[q,-n] \\ &= \sum_{p=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} E_L[-q,-n] h[q-p] C_M[-m,-p]. \end{aligned}$$

Noting that $E_L[-m,-n] = C_L[n,m]$ and $C_M[-m,-n] = E_M[n,m]$, $g^T[n,m]$ becomes

$$\begin{aligned} g^T[n,m] &= \sum_{p=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} C_L[n,q] h[q-p] E_M[p,m] \\ &= \sum_{p=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} C_L[n,p] h[p-q] E_M[q,m]. \end{aligned} \tag{2.7}$$

Comparing Equations 2.4 and 2.7 illustrates that the complement to a generalized rate-conversion system differs from the original system only in that the compression and expansion factors are interchanged. Figure 2-1 summarizes this result.

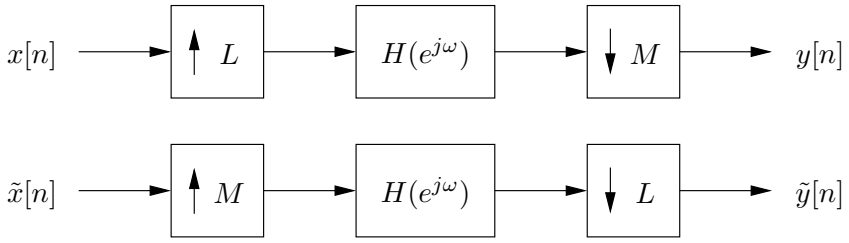


Figure 2-1: Complementary rational rate-conversion systems.

2.2 The Generalized Transposition Theorem

Because complementary rate-conversion systems have input-output relationships which are generalized transposes of one another, the Generalized Transposition Theorem can be used as a prescription for finding a structure for one system given the structure of its complement. This result is illustrated in [4], which as part of the proof represents branch functions in

terms of two-dimensional discrete-time Fourier transforms. In this section, an alternate proof for the Generalized Transposition Theorem is presented using time-domain arguments.

For an arbitrary linear flow graph network, the signal at a given node is equal to the summed contribution from all branches connecting to it, in addition to some potential contribution from an external source. Denoting the signal at node k as $w_k[n]$, the branch contribution from j to k as $v_{jk}[n]$, and the external input to k as $x_k[n]$, the signal at node k is written as

$$w_k[n] = x_k[n] + \sum_j v_{jk}[n]. \tag{2.8}$$

A given branch contribution $v_{jk}[n]$ from node j to node k relates, in turn, to node signal $w_j[n]$ in terms of a branch function $h_{jk}[n, m]$ as

$$v_{jk}[n] = \sum_m h_{jk}[n, m] w_j[m]. \tag{2.9}$$

The signal at node k is therefore related to signals at other nodes by

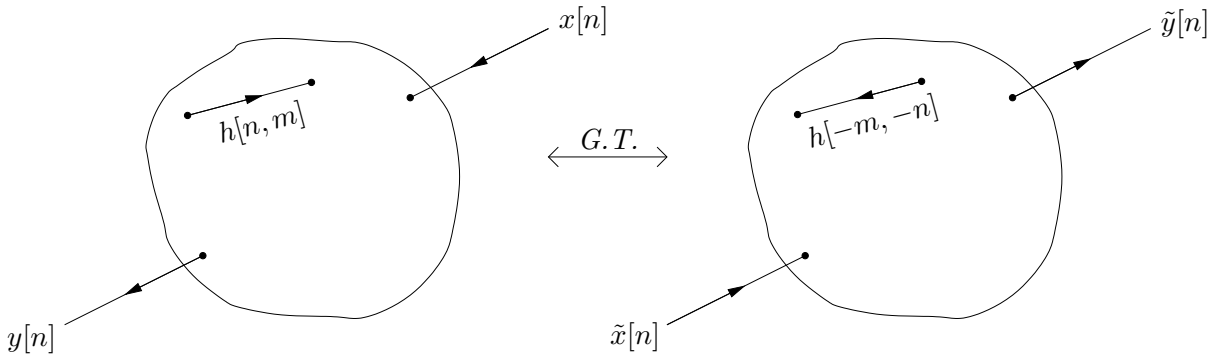
$$w_k[n] = x_k[n] + \sum_j \sum_m h_{jk}[n, m] w_j[m]. \tag{2.10}$$

The Generalized Transposition Theorem is stated with respect to these definitions, and the result is illustrated in Figure 2-2.

Theorem 2.2.1 (Generalized Transposition Theorem) *Given a flow graph for a single-input, single-output linear system, a flow graph for the complementary system is obtained by reversing the direction of every branch, replacing each branch function $h[n, m]$ with its generalized transpose $h[-m, -n]$, and exchanging the system input and output.*

In setting up the proof, consider a linear network with node variables $w_k[n]$, branch contributions $v_{jk}[n]$, external contributions $x_k[n]$, and branch functions $h_{jk}[n, m]$. Consider also a second network (which for convenience will be referred to as the "tilde network") with the same topology and with node variables $\tilde{w}_k[n]$, branch contributions $\tilde{v}_{jk}[n]$, external contributions $\tilde{x}_k[n]$, and branch functions $\tilde{h}_{jk}[n, m]$.[1]

[1]The constraint of topological equivalence is not particularly restrictive. An arbitrary number of nodes and branches with branch functions $h_{jk}[n, m] = 0$ and $\tilde{h}_{jk}[n, m] = 0$ are allowed in both networks, so any choice of original and tilde networks can be considered topologically equivalent. Unless otherwise stated, limits of summation are likewise assumed to extend from $-\infty$ to ∞ without loss of generality.



$$y[n] = \sum_{m=-\infty}^{\infty} g[n,m]x[m] \qquad \xleftrightarrow{G.T.} \qquad \tilde{y}[n] = \sum_{m=-\infty}^{\infty} g[-m,-n]\tilde{x}[m]$$

Figure 2-2: Illustration of the Generalized Transposition Theorem.

The proof begins by stating the following fact:

$$\sum_k \left(\sum_n \tilde{w}_k[n] w_k[-n] - \sum_n \tilde{w}_k[-n] w_k[n] \right) = 0. \tag{2.11}$$

Applying Equation 2.8 to Equation 2.11 for both the original and tilde networks gives

$$\sum_k \left[\sum_n \left(x_k[-n] + \sum_j v_{jk}[-n] \right) \tilde{w}_k[n] - \sum_n \left(\tilde{x}_k[-n] + \sum_j \tilde{v}_{jk}[-n] \right) w_k[n] \right] = 0.$$

This simplifies to

$$\sum_k \sum_j \sum_n \left(v_{jk}[-n]\tilde{w}_k[n] - \tilde{v}_{jk}[-n] w_k[n] \right) + \sum_k \sum_n \left(x_k[-n]\tilde{w}_k[n] - \tilde{x}_k[-n] w_k[n] \right) = 0. \tag{2.12}$$

The branch functions for the tilde network are defined in terms of the branch functions for the original network as

$$\tilde{h}_{jk}[n,m] \equiv h_{kj}[-m,-n]. \tag{2.13}$$

Re-stating Equation 2.9,

$$v_{jk}[-n] = \sum_m h_{jk}[-n,m] w_j[m], \tag{2.14}$$

and applying this to Equation 2.13 gives

$$\tilde{v}_{jk}[-n] = \sum_m \tilde{h}_{jk}[-n,m]\tilde{w}_j[m] = \sum_m h_{kj}[-m,n]\tilde{w}_j[m]. \tag{2.15}$$

Substituting Equations 2.14 and 2.15 into Equation 2.12 gives

$$\begin{aligned} &\sum_k \sum_j \sum_n \sum_m h_{jk}[-n,m]w_j[m]\tilde{w}_k[n] \\ &+\sum_k \sum_j \sum_n \sum_m -h_{kj}[-m,n]\tilde{w}_j[m]w_k[n] \\ &+\sum_k \sum_n (x_k[-n]\tilde{w}_k[n] - \tilde{x}_k[-n]w_k[n]) = 0. \end{aligned} \tag{2.16}$$

Rearranging indices of summation allows elimination of the quadruple sums, so Equation 2.16 becomes

$$\sum_k \sum_n (x_k[-n]\tilde{w}_k[n] - \tilde{x}_k[-n]w_k[n]) = 0. \tag{2.17}$$

Imposing the condition that only a single node a in the original network has contribution from an external input and that only a single node b in the tilde network has contribution from an external input, the summation over k in Equation 2.17 reduces to

$$\sum_n x_a[-n]\tilde{w}_a[n] = \sum_n \tilde{x}_b[-n]w_b[n].$$

Applying the naming convention from Figure 2-2,

$$\sum_n x[-n]\tilde{y}[n] = \sum_n \tilde{x}[-n]y[n]. \tag{2.18}$$

Relating $x[n]$ and $y[n]$ by Equation 2.1 therefore gives, without loss of generality,

$$\sum_n x[-n]\tilde{y}[n] = \sum_n \tilde{x}[-n] \sum_m g[n,m]x[m],$$

and manipulations on the indices of summation result in the relation

$$\sum_n x[-n]\tilde{y}[n] = \sum_n x[-n] \sum_m g[-m,-n]\tilde{x}[m]. \tag{2.19}$$

Because $\tilde{x}[n]$, $g[n,m]$, and $\tilde{y}[n]$ do not depend on $x[n]$ and because Equation 2.19 holds for

all $x[n]$,

$$\tilde{y}[n] = \sum_m g[-m,-n]\tilde{x}[m] = \sum_m g^T[n,m]\tilde{x}[m]. \tag{2.20}$$

Reversing the direction of all branches in a single-input, single-output network, replacing each branch function with its generalized transpose, and exchanging the system input and output therefore gives a complementary system.

When the network for a rate-conversion system consists only of expanders, compressors, and LTI subsystems, the prescribed structure for the complementary network is also defined in terms of these blocks.

Corollary 2.2.1 *For a single-input, single-output, linear flow graph which implements a generalized rate-conversion system and which consists only of expanders, compressors, and LTI branch functions, a complementary rate-conversion system can be obtained by reversing the direction of every branch, changing all expanders-by-L to compressors-by-L, changing all compressors-by-M to expanders-by-M, and exchanging the system input and output.*

The relationship between expander and compressor blocks in the original and transposed networks is verified by recalling $E_L^T[n,m] = E_L[-m,-n] = C_L[n,m]$ and $C_M^T[n,m] = C_M[-m,-n] = E_M[n,m]$. An LTI branch between nodes j and k with impulse response $f[n]$ relates the signal $w_j[n]$ to the branch contribution $v_{jk}[n]$ in terms of the convolution sum

$$v_{jk}[n] = \sum_m f[n-m]w_j[m],$$

so the corresponding branch function is $h[n,m] = f[n-m] = h[-m,-n] = h^T[n,m]$. LTI branch functions are therefore preserved under generalized transposition.

An important result discussed in [4] is that for a wide class of system architectures, any structure consisting of expanders, compressors, and LTI branch functions requires the same number of multiplications per unit time as its complementary structure. Briefly, this is shown by observing that a compressor-by-L connecting node j to k in the original network corresponds to an expander-by-L connecting node k to node j in the complementary network, and as long as the system is assumed to be running synchronously, the relative rates of nodes j and k are preserved.

Chapter 3

Computational cost

As has been alluded to thus far, the term "efficient" will refer throughout the thesis to the number of multiplications per unit time. The process of evaluating the efficiency of multirate systems will therefore be reviewed in this chapter. The relationship between structures, filter designs, and computational cost will also be discussed.

3.1 Computational cost for multirate systems

The efficiency of a multirate system is determined by two properties of its flow graph: the number of multipliers and the clock rate at which each multiplier operates. While counting the number of multipliers is usually a straightforward task, determining the rate at which each one operates often requires additional work. In multirate systems, the sampling rates in different sections of a flow graph are often understood to be related by the presence of multipliers, expanders, and compressors. Specifically, these blocks define relationships between the sampling rates of the nodes to which they are connected, therefore introducing notions of synchronism in the system. A summary of these notions follows:

- A multiplier block produces 1 output sample per 1 input sample, and the sampling rate of its output is equivalent to the sampling rate of its input.
- An expander-by-L block produces L output samples per 1 input sample, and the sampling rate of its output is L times the sampling rate of its input.
- A compressor-by-M block produces 1 output sample per M input samples, and the sampling rate of its output is $1/M$ times the sampling rate of its input.

While these blocks impose rules which relate the sampling rates between nodes, they are not alone sufficient to find the sampling rate at any particular node. When a sampling rate is specified at some node in a flow graph consisting of expanders, compressors, and multipliers, however, these rules imply sampling rates for every node in the system. The specification of sampling rates therefore introduces notions of time in a flow graph consisting of compressors, expanders, and multipliers.

The efficiency metric, "required number of multiplications per unit time," depends explicitly on notions of time. Therefore, the computational cost of a flow graph cannot be evaluated unless the sampling rate at some node in the system is known. The process of minimizing the number of multiplications per unit time in a generalized rational rate-conversion system, however, can still be discussed as long as its input and output sampling rates are known to be fixed. To demonstrate this, consider two rational rate-conversion systems, System A and System B, which have identical flow graphs. Systems A and B have respective positive input sampling rates R_A and R_B. Denoting the required number of multiplications per unit time $\mathcal{M}_A$ and $\mathcal{M}_B$ for Systems A and B respectively, the computational costs for the two systems are related by

$$\mathcal{M}_B = \frac{R_B}{R_A}\mathcal{M}_A.$$

Minimizing $\mathcal{M}_A$ therefore also minimizes $\mathcal{M}_B$ for all fixed, positive R_A and R_B. It is for this reason that the design of efficient rate-conversion systems will be discussed without the specification of sampling rates.

When a quantification of computational cost is required for systems with unspecified sampling rates, "multiplications per output sample" will be used, and this value will generally be represented by the variable $\mathcal{C}$. The metric will be considered equivalent to "multiplications per unit time," where the time unit is one sample interval of the output signal. Minimizing the required number of multiplications per output sample for a rational rate-conversion system therefore also minimizes the required number of multiplications per unit time.

3.2 Effect of computational cost on filter design

This thesis approaches the design of rate-conversion systems by first proposing efficient structures, then investigating filter designs which have small computational cost when im-

plemented using these structures. The act of choosing a structure therefore imposes a correspondence between each filter design and its computational cost. The process of designing efficient filters depends, in turn, on the structure that is chosen. It is for this reason that the relationship between structures and the computational cost of filter designs is discussed in this section.

The relationship is developed somewhat abstractly, with the goal of illustrating issues which motivate later chapters. In particular, the correspondences between filter designs and computational cost which are imposed by polyphase and folded structures are determined, and the correspondence which is imposed by the efficient structure proposed later in the thesis is also discussed. Systems of the form of Figure 1-1 which use ETSF filter designs with integer group delay are considered.

While polyphase structures do not exploit possible time-symmetry in $h[n]$, they do have the advantage of requiring the same number of multiplier blocks as direct-form implementations for $H(e^{j\omega})$ while performing each of these multiplications once per sample period at the slowest rate in the system. An ETSF filter of the form

$$h[n] = b_{n_0}\delta[n-n_0] + \sum_{k=1}^{K} b_k \left\{\delta[n-(n_0+k)] + \delta[n-(n_0-k)]\right\} \tag{3.1}$$

which is implemented using a polyphase structure is therefore said to have computational cost

$$\mathcal{C}_{poly} = \mathcal{L} \min\left\{1, \frac{M}{L}\right\}, \tag{3.2}$$

where $\mathcal{L} = 2K+1$ is the filter length. $\mathcal{C}_{poly}$ represents the required number of multiplications per output sample of the system when using a polyphase implementation. (An implementation is assumed to be chosen in which all multiplications are performed at the slower of the output rate and the input rate.) Folded implementations take advantage of the ETSF property of the impulse response $h[n]$ but perform all multiplications at the highest rate in the system. $K+1$ multipliers are required, and the computational cost of a folded implementation is therefore

$$\mathcal{C}_{fold} = \left(\frac{\mathcal{L}-1}{2}+1\right) M \tag{3.3}$$

multiplications per output sample.

The structure proposed in this thesis requires one multiplier block for each different

value taken on by $h[n]$ for all n on the interval $n \in [n_0 - K, n_0 + K]$. As in the polyphase case, each multiplication is performed once per sample period and at the slowest rate in the system. The computational cost of the proposed structure, in terms of the number of required multiplications per output sample, is therefore

$$\mathcal{C}_{prop} = \mathcal{U} \min\left\{1, \frac{M}{L}\right\}, \tag{3.4}$$

where $\mathcal{U}$ is the number of unique values of $h[n]$ on the interval $n \in [n_0 - K, n_0 + K]$.

The form of Equation 3.4 raises the question of how to design filters with many recurring coefficient values, *i.e.,* with small $\mathcal{U}$. Broadly speaking, this process is related to quantization, since the coefficients of the resultant filter will be drawn from a small set of values. As an alternative to solving the quantization problem in general, results from sparse array design[8] and sparse approximation theory[7][13] can be used to reduce $\mathcal{U}$ by designing filters where the value $h[n] = 0$ often recurs. With this restriction of the design problem and for ETSF impulse responses $h[n]$, an upper bound $\bar{\mathcal{U}}$ on $\mathcal{U}$ is stated in terms of the number of time indices where $h[n]$ is nonzero, $||h[n]||_0$, as

$$\mathcal{U} \leq \bar{\mathcal{U}} = \left\lceil \frac{||h[n]||_0}{2} \right\rceil. \tag{3.5}$$

For ETSF filters, the computational cost of the proposed structure accordingly has an upper bound $\bar{\mathcal{C}}_{prop}$ which relates to $\mathcal{C}_{prop}$ as

$$\mathcal{C}_{prop} \leq \bar{\mathcal{C}}_{prop} = \left\lceil \frac{||h[n]||_0}{2} \right\rceil \min\left\{1, \frac{M}{L}\right\}. \tag{3.6}$$

Note that $\mathcal{C}_{poly} \geq \bar{\mathcal{C}}_{prop}$ and $\mathcal{C}_{fold} \geq \bar{\mathcal{C}}_{prop}$ for fixed L and M.

To summarize the interplay between structures and filter design techniques, consider first how the computational cost for each structure grows with increasing filter length and zero-norm for fixed L and M. From Equations 3.2, 3.3 and 3.6, the following trends are noted:

$$\mathcal{C}_{poly} \propto \mathcal{L} \tag{3.7}$$

$$\mathcal{C}_{fold} \propto \mathcal{L} + \eta_{fold}(\mathcal{L}) \tag{3.8}$$

$$\bar{\mathcal{C}}_{prop} \propto ||h[n]||_0 + \bar{\eta}_{prop}(||h[n]||_0), \tag{3.9}$$

where $\eta_{fold}(\mathcal{L})$ and $\bar{\eta}_{prop}(||h[n]||_0)$ vanish for large $\mathcal{L}$ and $||h[n]||_0$, respectively. Denote the set of all filter designs where $\mathcal{L} \leq \ell$ and $||h[n]||_0 \leq k$ as $\mathbb{H}[\ell, k]$. It is therefore true that

$$\mathbb{H}[\ell, k] \subseteq \mathbb{H}[\ell + \gamma, k + \varepsilon] \tag{3.10}$$

for arbitrary ℓ and k and all nonnegative γ and ε, so the existence of a feasible filter design with given upper limits on length and zero-norm also implies the existence of feasible designs for arbitrarily-increased upper limits on length and zero-norm. In other words, increasing the allowable filter length or number of nonzero coefficients in any filter design algorithm can never cause feasible designs to become infeasible.

While Equation 3.10 may seem to make an obvious point, it introduces interesting structure into the relationship between $\mathcal{C}_{poly}$, $\mathcal{C}_{fold}$, $\bar{\mathcal{C}}_{prop}$, and $\mathbb{H}[\ell, k]$. This relationship is illustrated in Figures 3-1 and 3-2. A single point (ℓ, k) on these plots represents a class of designs $\mathbb{H}[\ell, k]$; the triangular shape is a result of the fact that the zero-norm for a filter can be no greater than its length. Shading in Figure 3-1 represents the trend in computational cost which is exhibited by both $\mathcal{C}_{poly}$ and $\mathcal{C}_{fold}$, and shading in Figure 3-2 corresponds to the trend exhibited by $\bar{\mathcal{C}}_{prop}$. Hatched areas in both figures represent example regions where feasible designs exist, *i.e.,* where $\mathbb{H}[\ell, k]$ contains a feasible design. Note that because Equation 3.10 holds, the hatched region on each plot must extend toward increasing length $\mathcal{L}$ and zero-norm $||h[n]||_0$. The lower-left boundary of the hatched region, however, depends on the particular set of filter designs which will be considered feasible. The problem of determining the most efficient implementation for a filter given a set of feasible impulse responses and a class of flow graph structures therefore involves finding the lightest-shaded point which is contained in the hatched area on these plots.

To gain intuition for the structure of these plots, consider first the point in Figure 3-1 where the most efficient feasible design lies. Because the leftmost corner of the hatched region is the lightest-shaded point in the hatched region, this point represents the set of designs which contains the most efficient feasible filter. This point also falls along the dotted line, which represents filter designs where the upper bound on $||h[n]||_0$ is $\mathcal{L}$. Figure 3-2 indicates, however, that the leftmost point in the hatched region does not contain the most efficient design for the trend in computational cost which is exhibited by the proposed structure. Finding a more efficient design involves searching below and to the right of this

point, which indicates that increasing the filter length $\mathcal{L}$ while decreasing the zero-norm $||h[n]||_0$ has the potential to give more efficient designs. This observation can by intuitively justified by the argument that decreasing the number of nonzero filter coefficients while increasing the possible times to which they apply trades off degrees of freedom in terms of coefficient value for increased degrees of freedom in terms of coefficient location, thereby exploring a different design space.

Figures 3-1 and 3-2 also explain why the Parks-McClellan algorithm is well-suited to find efficient designs for folded and polyphase implementations but cannot, in general, find the most efficient filter designs for the proposed structure. In both figures, points along the dotted line correspond to sets of filter designs where the zero-norm $||h[n]||_0$ is no greater than the length $\mathcal{L}$. Design techniques which make no special restrictions on $||h[n]||_0$, such as the Parks-McClellan algorithm, can therefore be used to explore this space. Because the dotted line in Figure 3-1 contains the point where the most efficient feasible filter exists, the Parks-McClellan algorithm can be used to iteratively search along this line and find a design of minimal cost subject to frequency constraints, assuming that the filter will be implemented using a folded or polyphase structure. This is the technique that is used in [12]. The proposed structure relates $||h[n]||_0$ and $\mathcal{L}$ to computational cost as in Figure 3-2, so the dotted line is not guaranteed to contain filter designs which have minimum computational cost when implemented using this structure. It is therefore possible that for the proposed structure, there exist feasible filters with lower computational cost than ones which the Parks-McClellan algorithm can generate. The design techniques presented in this thesis consequently investigate the set of filters represented by the region below the dotted line.

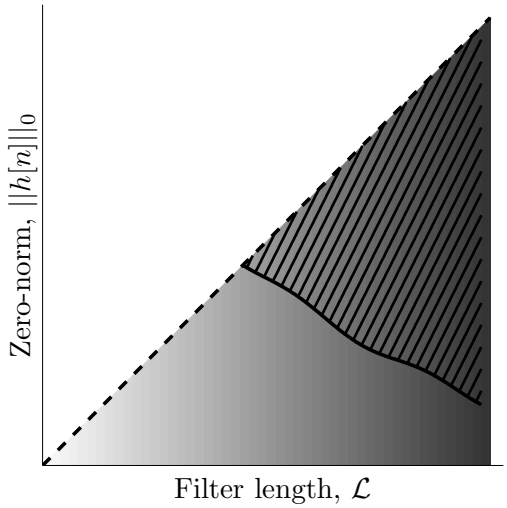


Figure 3-1: Example feasibility region, superimposed on trend in computational cost for polyphase and folded structures. Hatched area represents design problems which are feasible for a posed set of frequency constraints, and shading indicates computational cost. The Parks-McClellan algorithm operates at points along the dotted line.

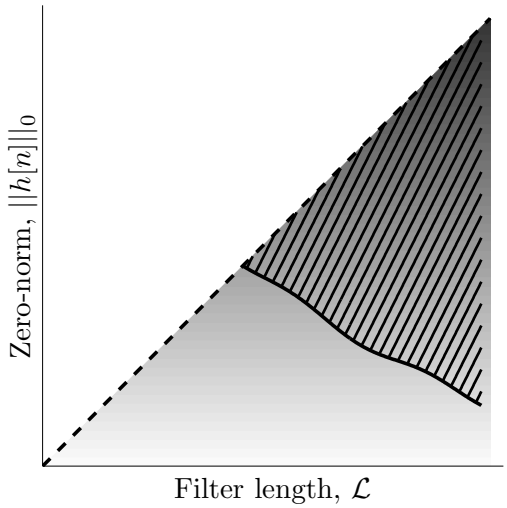


Figure 3-2: Example feasibility region, superimposed on trend in computational cost for presented implementation. Hatched area represents design problems which are feasible for a posed set of frequency constraints, and shading indicates the number of required multiplications per unit time. The Parks-McClellan algorithm operates at points along the dotted line, and the presented techniques search for feasible designs below it.

Chapter 4

Efficient structures

Flow graphs for rate-conversion systems often make use of compressor and expander blocks, and the noble identities can be used to exchange expanders and compressors with certain LTI branch functions. A number of structures have been proposed to take advantage of this fact, a broad class of which relate to the well-known polyphase structure.[3][6][10][15] Another property commonly encountered in FIR-based rate conversion systems is coefficient redundancy, where the impulse response of the filter takes on the same value for multiple time indices. Because coefficient redundancy necessarily occurs in ETSF filters, the issue can be exploited for a large class of desirable rate-conversion systems. Efficiencies which take advantage of coefficient redundancy can, in a general way, be combined with polyphase techniques to yield further improvements, and the structures proposed in this chapter address this. Drawing on the Generalized Transposition Theorem presented in [4] and discussed in Chapter 2, the techniques in this chapter will be applied to the downsampling case with the understanding that the Generalized Transposition Theorem can be invoked to find a complementary upsampling system which is optimally efficient in the same sense.

4.1 Exploiting coefficient redundancy

In this section the implementation of FIR filters with redundant coefficients is explored. A class of structures which implements such filters using fewer multipliers than corresponding direct-form implementations is presented, and the result is illustrated within the context of linear-phase FIR filters.

4.1.1 General technique for FIR filters

For the implementation of FIR filters with redundant coefficients, efficiencies can be introduced in a straightforward way. Specifically, when a filter with impulse response $h[n]$ and z-transform $H(z)$ is written in the form

$$H(z) = \sum_{k=0}^{\mathcal{U}-1} c_k \sum_{p=n_{min}}^{n_{max}} d_{k,p} z^{-p},\ d_{k,p} \in \{0,1\}, \tag{4.1}$$

where

$$h[n] = 0 \quad \forall n > n_{max},\ n < n_{min},$$

and where the coefficients c_k represent the $\mathcal{U}$ unique values taken on by the filter's impulse response, $\mathcal{U}$ multiplications are required per input sample to the filter. In particular, implementing the system using the structure in Figure 4-1 takes advantage of this representation.

Writing $H(z)$ as

$$H(z) = \sum_{p=n_{min}}^{n_{max}} b_p z^{-p}$$

and matching polynomial terms with those in Equation 4.1 gives

$$b_p = \sum_{k=0}^{\mathcal{U}-1} c_k d_{k,p}. \tag{4.2}$$

Equation 4.2 therefore relates the structure in Figure 4-1 to the feed-forward coefficients for the FIR filter it implements.

4.1.2 Application to linear-phase FIR filters

Since ETSF filters have impulse responses which are symmetric about some integer or half-integer point n_0, $2n_0 \in \mathbb{Z}$, their coefficients b_p can be represented as

$$b_{n_0-\ell} = b_{n_0+\ell}, \quad 2n_0,\ n_0+\ell \in \mathbb{Z},$$

which from Equation 4.2 gives

$$b_{n_0-\ell} = \sum_{k=0}^{\mathcal{U}-1} c_k d_{k,n_0-\ell} = \sum_{k=0}^{\mathcal{U}-1} c_k d_{k,n_0+\ell}, \quad 2n_0,\ n_0+\ell \in \mathbb{Z}, \quad d_{k,p} \in \{0,1\}. \tag{4.3}$$

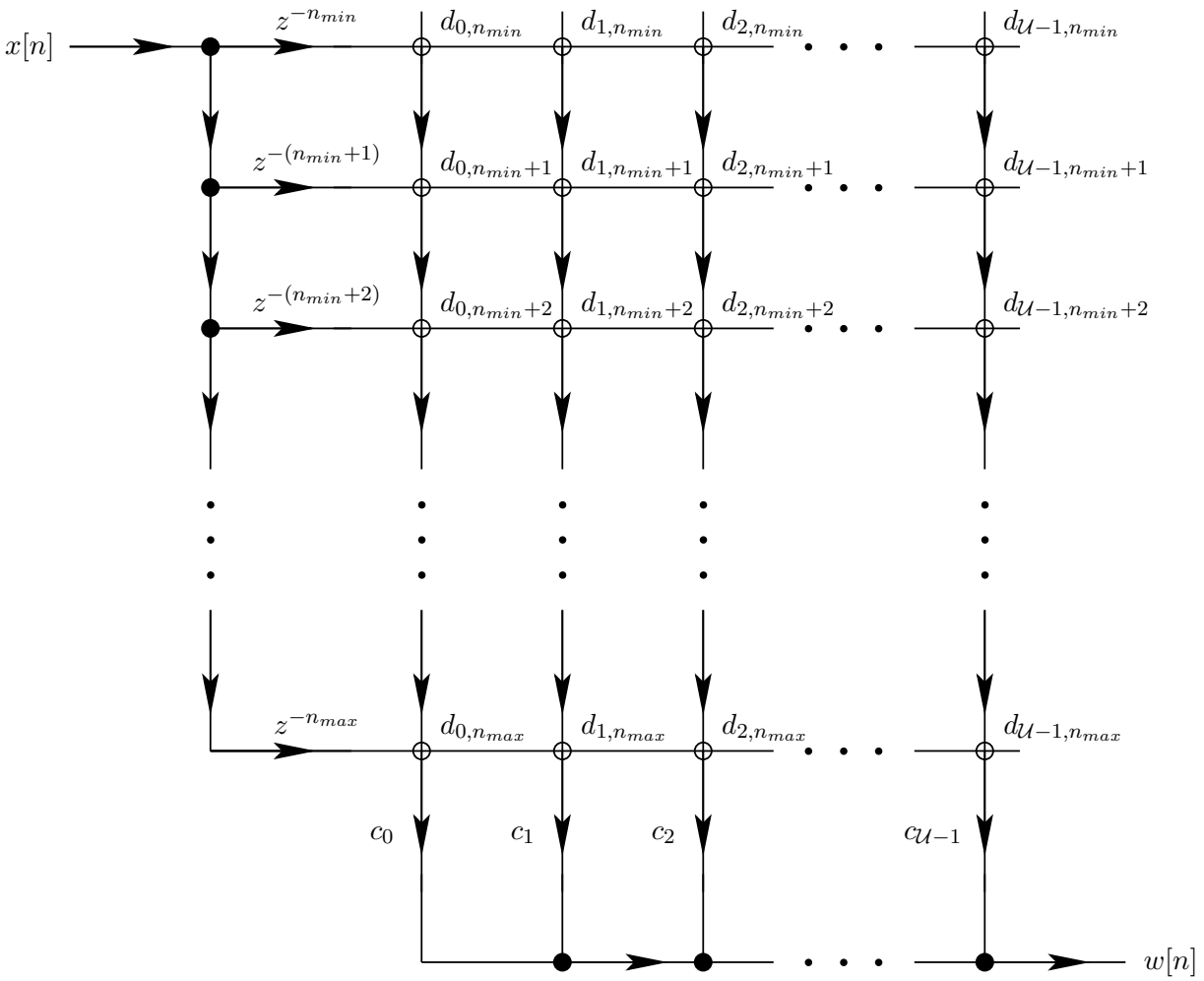


Figure 4-1: General structure for exploiting coefficient redundancy. Coefficients $d_{k,p}$ with value $d_{k,p} = 1$ represent connections at corresponding nodes.

Choosing the multiplier coefficients c_k as

$$c_k = \begin{cases} b_{n_0+k}, & n_0 \in \mathbb{Z} \\ b_{n_0+k+\frac{1}{2}}, & n_0 + \frac{1}{2} \in \mathbb{Z} \end{cases}, \quad k \geq 0$$

implies from Equation 4.3 that

$$b_{n_0-\ell} = \begin{cases} \sum_{k=0}^{\mathcal{U}-1} b_{n_0+k} d_{k,n_0-\ell} = \sum_{k=0}^{\mathcal{U}-1} b_{n_0+k} d_{k,n_0+\ell}, & n_0 \in \mathbb{Z} \\ \sum_{k=0}^{\mathcal{U}-1} b_{n_0+k+\frac{1}{2}} d_{k,n_0-\ell} = \sum_{k=0}^{\mathcal{U}-1} b_{n_0+k+\frac{1}{2}} d_{k,n_0+\ell}, & n_0 + \frac{1}{2} \in \mathbb{Z} \end{cases}, \quad n_0+\ell \in \mathbb{Z}, \quad d_{k,p} \in \{0,1\}$$

which is satisfied by

$$d_{k,p} = \begin{cases} \delta[k - |n_0 - p|], & n_0 \in \mathbb{Z} \\ \delta[k + \frac{1}{2} - |n_0 - p|], & n_0 + \frac{1}{2} \in \mathbb{Z} \end{cases} . \tag{4.4}$$

This choice of $d_{k,p}$ gives the well-known folded structure discussed in [10]. It is illustrated for a typical $n_0 + \frac{1}{2} \in \mathbb{Z}$ case in Figure 4-2.

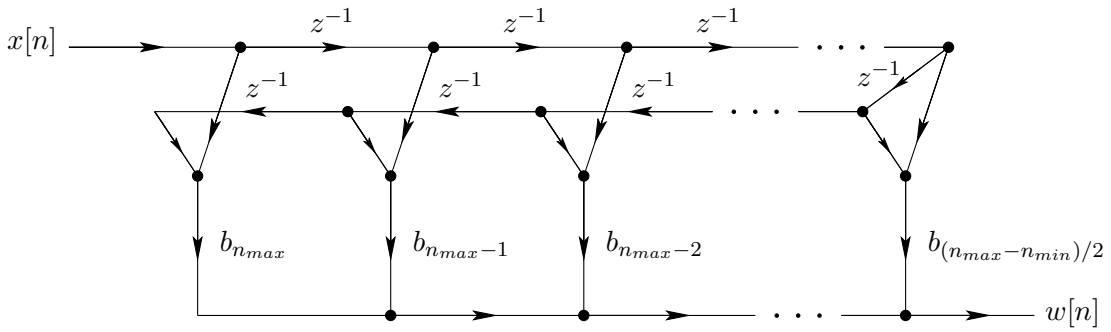


Figure 4-2: Folded implementation of an ETSF filter.

4.2 Polyphase implementation

Polyphase structures are widely used in rate-conversion systems to efficiently implement FIR filter designs. With this motivation, a polyphase implementation of the structure in Figure 4-1 is presented. Rate conversion systems consisting of an anti-aliasing filter $H(e^{j\omega})$ followed by a compressor-by-M are considered, and this arrangement is depicted in Figure 4-3. It is shown that for any system represented in this form, where $H(z)$ is a length-$\mathcal{L}$ FIR

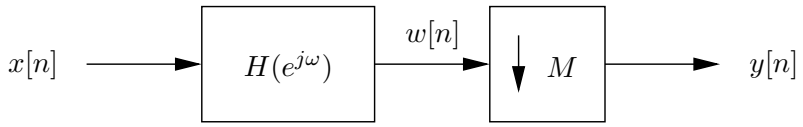


Figure 4-3: General downsampling arrangement for which efficiencies are discussed.

filter whose impulse response takes on $\mathcal{U}$ unique values, the polyphase implementation of Figure 4-1 provides an implementation requiring $\mathcal{U}$ multiplications per output sample and employing M compressor blocks. Because all multiplications in this structure are performed at the rate of the output signal $y[n]$, the system can be prepended by an expander-by-L to

implement a generalized rational rate converter. If the input rate of this resulting system is slower than that of the output rate, the generalized transpose of the system in Figure 4-3 can instead be implemented first and then cascaded with an appropriate compressor block. Using these manipulations, the polyphase implementation of Figure 4-1 can be used to realize generalized rational rate converters where all of the multiplications are performed at the slowest rate in the system, whether this rate occurs at the system input or output.

Equation 4.1 implies the following input-output relationship between the input $x[n]$ and the output $w[n]$ for the system in Figure 4-1:

$$w[n] = \sum_{p=n_{min}}^{n_{max}} x[n-p] \sum_{k=0}^{\mathcal{U}-1} c_k d_{k,p},\ d_{k,p} \in \{0,1\}. \tag{4.5}$$

Compressing $w[n]$ by a factor of M results in a new signal $y[n]$ which is related to $w[n]$ by $y[n] = w[Mn]$. Applying this relationship to Equation 4.5 gives

$$y[n] = \sum_{p=n_{min}}^{n_{max}} x[Mn-p] \sum_{k=0}^{\mathcal{U}-1} c_k d_{k,p},\ d_{k,p} \in \{0,1\}. \tag{4.6}$$

By decomposing $x[n]$ into its polyphase components

$$x_p[n] = x[Mn-p], \tag{4.7}$$

Equation 4.6 becomes

$$y[n] = \sum_{p=n_{min}}^{n_{max}} x_p[n] \sum_{k=0}^{\mathcal{U}-1} c_k d_{k,p},\ d_{k,p} \in \{0,1\}. \tag{4.8}$$

The relationship between $x[n]$ and $y[n]$ is therefore represented as a cascade of a single-input, multiple-output system (Equation 4.7) with a multiple-input, single-output system (Equation 4.8). The relationship between $x[n]$ and $x_p[n]$ in Equation 4.7 can be expressed in terms of the following equations:

$$v_p[n] = x[n-p] \tag{4.9}$$

$$x_p[n] = v_p[Mn]. \tag{4.10}$$

This separates one branch of the single-input, multiple-output system into a cascade of an

LTI delay-by-p with a compressor-by-M. The transfer function of Equation 4.9 is therefore

$$\frac{V_p(z)}{X(z)} = z^{-p} \ ,$$

and invoking the identity

$$p = M\lfloor p/M \rfloor + ((p))_M \quad \forall p \in \mathbb{Z}, M \in \mathbb{Z}^+$$

gives

$$\frac{V_p(z)}{X(z)} = z^{-p} = z^{-\{M\lfloor p/M \rfloor + ((p))_M\}} = z^{-M\lfloor p/M \rfloor} z^{-((p))_M}. \tag{4.11}$$

The relationship between $x[n]$ and $v_p[n]$ in Equation 4.9 can, in turn, be expressed using the following two equations:

$$u_p[n] = x[n - ((p))_M] \tag{4.12}$$

$$v_p[n] = u_p[n - M\lfloor p/M \rfloor]. \tag{4.13}$$

The relationship between $x[n]$ and a given $x_p[n]$ is therefore described by Equations 4.10, 4.12, and 4.13, which represent a cascade of three single-input, single-output systems.

Because the system from $u_p[n]$ to $x_p[n]$ (Equations 4.13 and 4.10) consists of an LTI block whose z-transform contains only integer powers of z^M, followed by a compressor-by-M, the downsampling noble identity can be applied. This manipulation results in a new set of equations which equivalently describe the relation between $u_p[n]$ and $x_p[n]$:

$$r_p[n] = u_p[Mn] \tag{4.14}$$

$$x_p[n] = r_p[n - \lfloor p/M \rfloor]. \tag{4.15}$$

The system from $x[n]$ to $x_p[n]$ can now be implemented as a cascade of Equations 4.12, 4.14, and 4.15. As a final manipulation, the order of summation in Equation 4.8 is changed, resulting in

$$y[n] = \sum_{k=0}^{\mathcal{U}-1} c_k \sum_{p=n_{min}}^{n_{max}} x_p[n] d_{k,p}, \ d_{k,p} \in \{0, 1\}. \tag{4.16}$$

Equation 4.16 demonstrates the explicit use of $\mathcal{U}$ multipliers. Note that there are, however, at most M unique systems from $x[n]$ to $r_p[n]$ (described by a cascade of Equations

4.12 and 4.14), since Equation 4.12 gives at most M unique results $u_p[n]$ for all $p \in \mathbb{Z}$ and Equation 4.14 has a dependence on p which is only involved in choosing the appropriate input $u_p[n]$. Because Equation 4.14 implements compression by a factor of M and there are at most M unique signals $r_p[n]$, at most M compressors are required when implementing the overall system from $x[n]$ to $y[n]$ using Equations 4.12, 4.14, 4.15, and 4.16, and $\mathcal{U}$ multipliers are required. Figures 4-4 and 4-5 depict an implementation which uses M compressors and $\mathcal{U}$ multipliers, all of which operate at the output rate.

4.3 Comparison of results

A comparison between computational requirements for different implementations of the system in Figure 4-3 is summarized in Table 4.1. $H(z)$ in this system represents a length-$\mathcal{L}$ filter whose impulse response takes on $\mathcal{U}$ unique values. The number of required multiplications per output sample are compared, in addition to the number of required compressor blocks.

Structure	Multiplications per output sample	Compressors
Direct-form	$\mathcal{L} \cdot M$	1
Polyphase	$\mathcal{L}$	M
Presented	$\mathcal{U}$	M

Table 4.1: Comparison of computational requirements for different implementations of a downsampler-by-M based around a length-$\mathcal{L}$ FIR filter with $\mathcal{U} \leq \mathcal{L}$ unique coefficients b_p.

A similar comparison is illustrated in Table 4.2 for the case where $H(z)$ is a length-$\mathcal{L}$ ETSF filter and $h[n_0 - \ell] = h[n_0 + \ell]$ $\forall (n_0 + 1/2), (n_0 + \ell) \in \mathbb{Z}$ ($h[n]$ is symmetric about a half-integer point n_0). This case is discussed in Subsection 4.1.2, where the number of unique values taken on by $h[n]$ is at most $\mathcal{U} = \mathcal{L}/2$.

Another set of FIR filters which take advantage of the class of structures discussed in this chapter have impulse response samples with known value $h[n] = 0$ for certain values of n in the range $n_{min} \leq n \leq n_{max}$. Chapter 5 considers design algorithms for generating filters with this property.

Structure	Multiplications per output sample	Compressors
Direct form	$\mathcal{L} \cdot M$	1
Folded delay line	$\mathcal{L} \cdot M/2$	1
Polyphase	$\mathcal{L}$	M
Presented	$\mathcal{L}/2^{\dagger}$	M

Table 4.2: Comparison of computational requirements for different implementations of a downsampler-by-M based around a length-$\mathcal{L}$ linear-phase FIR filter. †This assumes $\mathcal{U} = \mathcal{L}/2$. It is also possible, however, that $\mathcal{U} < \mathcal{L}/2$. In this case, the computational cost of the presented structure is further reduced.

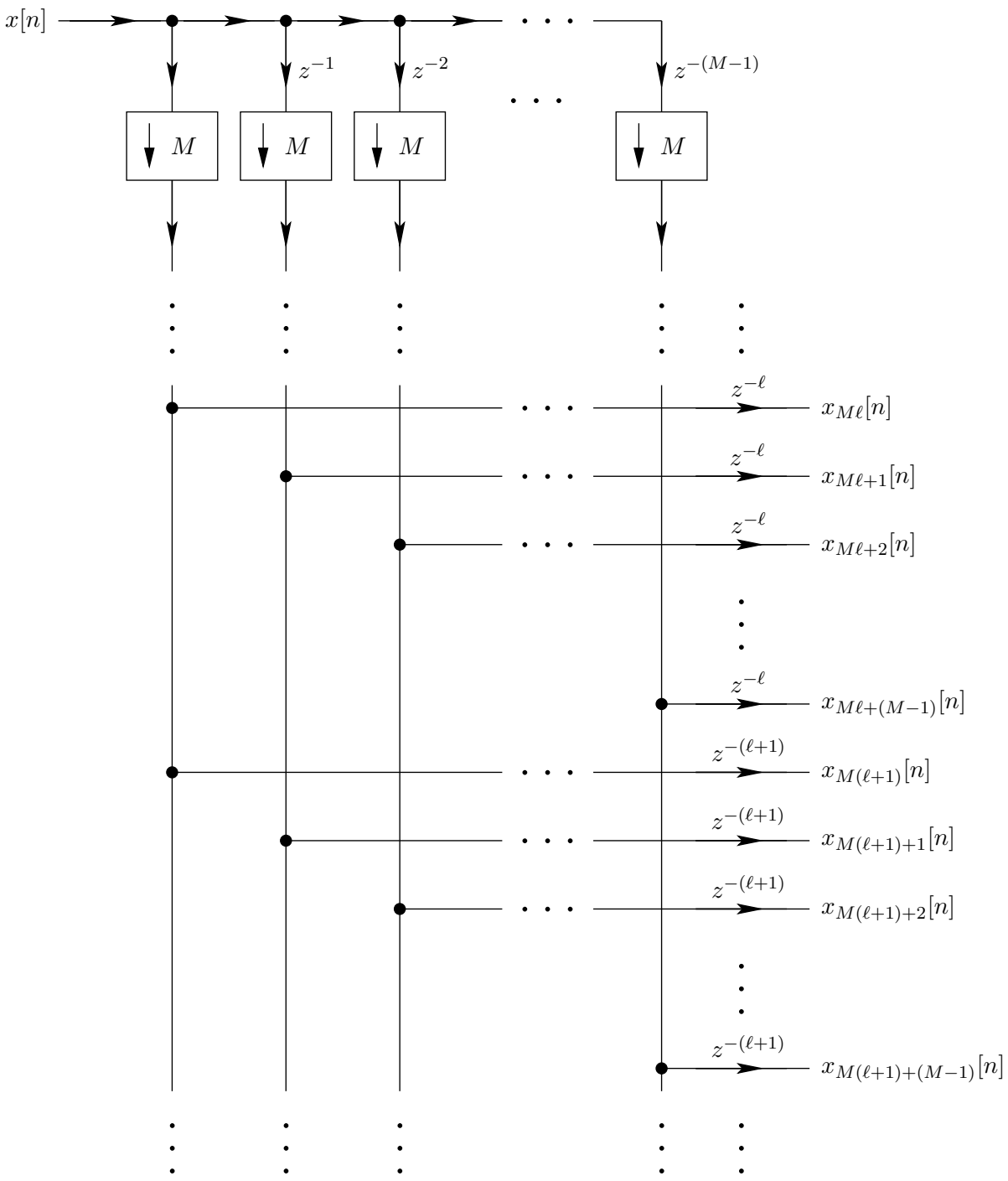


Figure 4-4: Delay and compressor section of polyphase structure for exploiting coefficient redundancy. Typical outputs $x_p[n]$ are shown for p in the range $n_{min} < M\ell \leq p \leq M(\ell + 1) + (M - 1) < n_{max}$, for some $\ell \in \mathbb{Z}$.

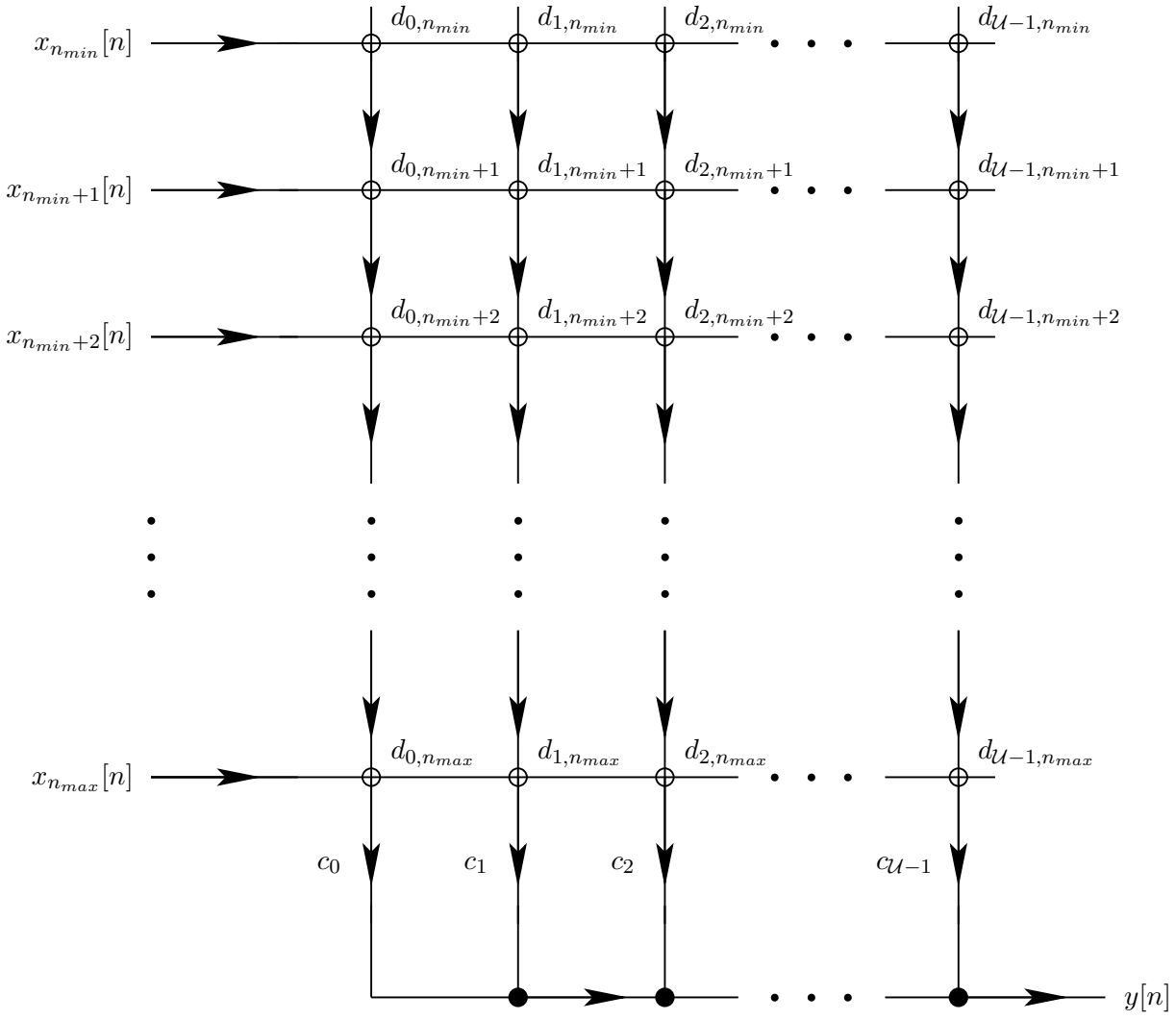


Figure 4-5: Multiplier section of polyphase structure for exploiting coefficient redundancy. Coefficients $d_{k,p}$ with value $d_{k,p} = 1$ represent connections at corresponding nodes.

Chapter 5

Design of efficient filters

In this chapter the problem of designing filters for efficient rate conversion systems is investigated. The filters discussed in this chapter will be restricted to the widely-applicable ETSF class of designs. In particular, special attention will be paid to filters which lend themselves to efficient implementations when using the structures presented in Chapter 4. It was demonstrated in Chapter 3 that the Parks-McClellan algorithm is not, in general, capable of generating filters with minimum computational cost when implemented using these structures. Alternative design methods will therefore be investigated. Since many of the techniques will be centrally formulated in terms of linear programs, discussions of the applicability of linear programming to filter design will be included throughout.

5.1 Design constraints and error metrics

With the understanding that the presented arguments extend easily to the general ETSF case, the discussion focuses on methods for designing ETSF filters whose point of symmetry is $n = 0$. Specifically, impulse responses of the form

$$h[n] = b_0\delta[n] + \sum_{k=1}^{K} b_k \left(\delta[n-k] + \delta[n+k]\right), \quad b_0, \ldots, b_K \in \mathbb{R} \tag{5.1}$$

with associated DTFT

$$H(e^{j\omega}) = b_0 + \sum_{n=1}^{K} 2b_n \cos\left(\omega n\right), \quad b_0, \ldots, b_K \in \mathbb{R} \tag{5.2}$$

are considered. Note that from this definition, the length of the filter is $\mathcal{L} = 2K + 1$, and the filter necessarily has a real DTFT.

Within the scope of Equation 5.1, the filter design process can be generally formulated as finding the design parameters $K, b_0, \ldots, b_K$ which satisfy

$$\min_{K,b_0,\ldots,b_K} f(K, b_0, \ldots, b_K) \;\text{ s.t. } h[n] \in C,\; K \in \mathbb{Z}^+ \tag{5.3}$$

where C denotes some set of feasible impulse responses and $f(K, b_0, \ldots, b_K)$ is an optimization metric over the design parameters. The design problems that will be addressed are taken from Table 5.1, which partitions the respective sets of feasible designs C and optimization metrics f that will be considered. In all cases, the output of the optimization is $h[n]$, and Equation 5.1 relates $h[n]$ to the design parameters. The ideal desired response is denoted $h_d[n]$, and the frequency error of the designed filter is defined as $E(e^{j\omega}) \equiv H(e^{j\omega}) - H_d(e^{j\omega})$. $W(\omega)$ is a weighting function which pre-emphasizes the error so that it is minimized to a greater extent for larger values of $W(\omega)$. N_z is the set of time locations on the interval $[0, K]$ where the impulse response $h[n]$ is constrained to be zero. Note that since the discussion is restricted to zero-phase filters, $n \in N_z \Rightarrow b_n = 0 \Leftrightarrow h[n] = 0 \Leftrightarrow h[-n] = 0$. Prior work is referenced for problems which are related to, but do not constitute, the main focus of the chapter.

min ⇓ s.t. ⇒	K fixed	$b_n = 0\ \forall n \in N_z$, K fixed	$\max_{\omega \in F} \lvert W(\omega)E(e^{j\omega})\rvert \le \delta$, K fixed
$\int_{-\pi}^{\pi} \lvert E(e^{j\omega})\rvert^2 d\omega$	[10]	[10]	[1],[2]
$\max_{\omega \in F} \lvert W(\omega)E(e^{j\omega})\rvert$	(5.2.1)	(5.2.2)	(5.2.1)
$\lvert\lvert h[n]\rvert\rvert_1$	**T**	**T**	(5.2.3)
$\lvert\lvert h[n]\rvert\rvert_0$	**T**	**T**	(5.2.4)

Table 5.1: Optimization metrics and feasibility constraints for different filter design problems. Designs marked (k) are addressed in Subsection k, those denoted $[j]$ are discussed in reference j, and those marked **T** represent optimization problems which have the trivial solution $h[n] = 0$.

The number of unique nonzero sample values $\hat{\mathcal{U}}$ taken on by an ETSF impulse response $h[n]$ is bounded as

$$\hat{\mathcal{U}} \le \left\lceil \frac{||h[n]||_0}{2} \right\rceil.$$

Making the fairly general assumption that all nonzero values of $h[n]$ for $0 \leq n \leq K$ are unique,

$$\hat{\mathcal{U}} = \left\lceil \frac{||h[n]||_0}{2} \right\rceil . \tag{5.4}$$

Minimizing $||h[n]||_0$ therefore corresponds to minimizing the number of multipliers required by the structures in Chapter 4 when all nonzero values of $h[n]$ for $0 \leq n \leq K$ are unique. While it is possible to further reduce computational cost by finding filters with recurring nonzero values, formulating the problem in terms of finding filters with small $||h[n]||_0$ allows the use of a number of results from sparse array design[8] and sparse approximation theory.[7][13]

Although only one non-trivial entry from Table 5.1 explicitly minimizes $||h[n]||_0$, the different design constraints on $h[n]$ impose relevant bounds on $||h[n]||_0$ for each problem considered. Upper limits on $||h[n]||_0$ are listed in Table 5.1 for each design constraint.

Constraint:	K fixed	$b_n = 0 \; \forall n \in N_z$ K fixed	$\max_{\omega \in F} \left\lvert W(\omega) E(e^{j\omega}) \right\rvert \leq \delta$ K fixed
Norm limit:	$\|\|h[n]\|\|_0 \leq 2K+1$	$\|\|h[n]\|\|_0 \leq 2K+1-\lvert N_z \rvert$	$\|\|h[n]\|\|_0 \leq 2K+1$

Table 5.2: Bounds on $||h[n]||_0$ for each design constraint.

In an attempt to interrelate and summarize the non-trivial problems from Table 5.1, entries are compared in terms of their optimization metrics and design constraints.

5.1.1 Minimum squared-error designs (row 1)

The squared-error optimization metric has the convenient property of being representable in terms of the induced inner product on the Hilbert space l^2, so a rich body of theory can be invoked to find analytic solutions for many of the corresponding design problems. The first entry in row 1 represents the optimization problem

$$\min_{b_0,\ldots,b_K} \sum_{k=-K}^{K} (h[k] - h_d[k])^2, \tag{5.5}$$

with solution

$$h[n] = \begin{cases} 0, & |n| > K \\ h_d[n], & \text{otherwise} \end{cases} , \tag{5.6}$$

and the second entry in row 1 corresponds to

$$\min_{b_0,\ldots,b_K} \sum_{\substack{k=-K \\ k\notin N_z}}^{K} (h[k] - h_d[k])^2, \tag{5.7}$$

with solution

$$h[n] = \begin{cases} 0, & |n| > K \text{ or } |n| \in N_z \\ h_d[n], & \text{otherwise} \end{cases} . \tag{5.8}$$

The third entry, which represents the class of problems where the total energy in $E(e^{j\omega})$ is minimized, subject to constraints on the peak value of $\left|W(\omega)E(e^{j\omega})\right|$, does not in general have a closed-form solution. It does, however, draw a connection between minimum squared-error designs and notions of peak error. This entry also points to the common practice of specifying filter designs in terms of inequality constraints on $H(e^{j\omega})$, which for many minimum squared-error filters effectively mitigates the impact of Gibbs' phenomenon on $E(e^{j\omega})$. For the remainder of the non-trivial problems in Table 5.1, similar notions of specifying constraints on, and often minimizing, the peak value of $\left|W(\omega)E(e^{j\omega})\right|$ will dominate problem formulations as discussion of the total energy in $E(e^{j\omega})$ is dropped.

5.1.2 Minimum peak error designs (row 2)

The first entry in row 2 represents the class of optimization problems which is addressed by the well-known Parks-McClellan algorithm. Within the context of this chapter, however, a close approximation to the problem will be cast into the framework of linear programming. This formulation will also be useful in addressing closely-related subsequent problems.

The second entry in row 2 will be the first such problem to draw on this framework. For designs with equivalent $||h[n]||_0$, the second entry can, but does not necessarily, give filters with smaller peak frequency error than the first entry. This property is tied to the fact that for this design problem, the alternation theorem (on which the Parks-McClellan algorithm is based) does not apply. This formulation also characterizes many related engineering problems, including min-max filter design for architectures where multipliers are known to be broken with output value forced to zero, as well as beamformer design for array processing applications with fixed but nonuniform transducer locations.

Because $\max_{\omega\in F} |W(\omega)E(e^{j\omega})|$ is minimized in this row and because the respective sets of feasible responses $h[n]$ for the first and third entries are related by $\left\{h[n] \;\middle|\; K \text{ fixed}\right\}$

$\supseteq \left\{h[n] \;\middle|\; \max_{\omega \in F} |W(\omega)E(e^{j\omega})| \leq \delta,\ K \text{ fixed}\right\}$, the third entry is discussed in the same section as the first entry. This is specifically done because an optimal solution to the first entry is guaranteed to be the corresponding optimal solution to the third entry, as long as a solution to the third entry exists. The computational methods reviewed in Section 5.2.1 can therefore be used to solve this design problem.

5.1.3 Constrained peak error designs (column 3)

Many design specifications are, in practice, formulated in terms of the peak allowable error as $\max_{\omega \in F} |W(\omega)E(e^{j\omega})| \leq \delta$. Filter design techniques have therefore been developed for a number of such problems over an array of optimality criteria. The minimum error energy and minimum peak error problems (corresponding to column three's first two entries) received attention in the previous two subsections.

Since minimizing $||h[n]||_0$ also minimizes the number of required multipliers for the class of design problems presented, it would be convenient if there were a straightforward way to obtain a minimal zero-norm filter, given a set of design specifications. This problem unfortunately relies in general on a combinatoric search whose complexity grows exponentially with K. The third entry in column three therefore represents a proposed relaxation of this optimization metric which is often used for sparse approximation and can be formulated in terms of a single linear program.[7][13] While minimizing the one-norm of $h[n]$ does not explicitly return a filter with the minimum number of nonzero coefficients, it does tend to result in filters which have very small coefficients, and zeroing these small coefficients often results in filters which very nearly meet the original design requirements.

The fourth entry explicitly designs filters with minimal zero-norm, subject to a set of frequency-domain specifications. While computationally expensive, the combinatoric search involved in the design process can be expedited by taking advantage of the hierarchical nature of the underlying search tree. The economy of using a true minimal zero-norm design procedure is still questionable, however, given the often marginal reduction in error over near-optimal approaches.

5.2 Minimum-multiply designs

The following filter specifications characterize the design problems which will be considered for the remainder of the chapter. These specifications were chosen to demonstrate the behavior of each design technique using filters of relatively short length.

- Filter type: lowpass
- Passband edge: $0.2\pi \quad (H_d(e^{j\omega}) = 1,\ \omega \in [0, 0.2\pi])$
- Stopband edge: $0.25\pi \quad (H_d(e^{j\omega}) = 0,\ \omega \in [0.25\pi, \pi])$
- Error in passband: $\left|E(e^{j\omega})\right| \leq 0.01\ \forall \omega \in [0, 0.2\pi]$
- Error in stopband: $\left|E(e^{j\omega})\right| \leq 0.1\ \forall \omega \in [0.25\pi, \pi]$

The specifications bounding passband and stopband error can be equivalently formulated as

$$\max_{\omega \in F} |W(\omega)E(e^{j\omega})| \leq 0.1$$

where

$$F = [0, 0.2\pi] \cup [0.25\pi, \pi]$$

and

$$W(\omega) = \begin{cases} 10, & 0 \leq \omega \leq 0.2\pi \\ 1, & 0.25\pi \leq \omega \leq \pi \end{cases} .$$

Entries from Table 5.1 are compared for this set of specifications, with the goal of illustrating the sense in which each design technique minimizes the number of required multipliers.

5.2.1 Filters with constrained length and minimum peak frequency error

The Parks-McClellan algorithm obtains solutions to the class of optimization problems

$$\text{minimize } \max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right| \text{ subject to } K \text{ fixed} \tag{5.9}$$

The following related problem differs from 5.9 only in that its set of feasible impulse responses is further restricted by an additional constraint:

$$\text{minimize } \max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right| \text{ subject to } \max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right| \leq \delta,\ K \text{ fixed} \tag{5.10}$$

Denoting the set of feasible impulse responses for 5.9 as

$$C_{5.9} = \left\{ h[n] \;\middle|\; K \text{ fixed} \right\} \tag{5.11}$$

and those for 5.10 as

$$C_{5.10} = \left\{ h[n] \;\middle|\; \max_{\omega \in F} |W(\omega)E(e^{j\omega})| \leq \delta,\ K \text{ fixed} \right\} = C_{5.9} \cap \left\{ h[n] \;\middle|\; \max_{\omega \in F} |W(\omega)E(e^{j\omega})| \leq \delta \right\}, \tag{5.12}$$

it becomes clear that $C_{5.10} \subseteq C_{5.9}$. This implies that if no solution exists to 5.9, no solution exists to 5.10. An optimal solution to 5.9 will furthermore be an optimal solution to 5.10 as long as a solution to 5.10 exists, and any optimal (and feasible) solution to 5.10 will also be an optimal (and feasible) solution to 5.9. A recap of these implications follows.

(a) No solution to 5.9 exists $\implies$ no solution to 5.10 exists

(b) An optimal solution $h[n]$ to 5.9 satisfies $\max_{\omega \in F} |W(\omega)E(e^{j\omega})| > \delta \implies$ no solution to 5.10 exists

(c) An optimal solution $h[n]$ to 5.9 satisfies $\max_{\omega \in F} |W(\omega)E(e^{j\omega})| \leq \delta \implies h[n]$ is an optimal solution to 5.10

A solution to 5.10 can therefore be found by solving 5.9 and evaluating (a)-(c).

While the Parks-McClellan algorithm prescribes an efficient method for solving 5.9, the focus will be on a close linear-programming approximation which has been used by Steiglitz, Parks and Kaiser[12] and which will be built upon for the remainder of the chapter. In

standard linear program form, this approximation is described by

$$
\begin{aligned}
&\text{minimize } \delta \\
&\text{subject to } \delta W(\omega_0) - b_0 - \sum_{n=1}^{K} 2b_n \cos(\omega_0 n) \geq -H_d(e^{j\omega_0}) \\
&\qquad \delta W(\omega_1) - b_0 - \sum_{n=1}^{K} 2b_n \cos(\omega_1 n) \geq -H_d(e^{j\omega_1}) \\
&\qquad\qquad \vdots \\
&\qquad \delta W(\omega_p) - b_0 - \sum_{n=1}^{K} 2b_n \cos(\omega_p n) \geq -H_d(e^{j\omega_p}) \\
&\qquad \delta W(\omega_0) + b_0 + \sum_{n=1}^{K} 2b_n \cos(\omega_0 n) \geq H_d(e^{j\omega_0}) \\
&\qquad \delta W(\omega_1) + b_0 + \sum_{n=1}^{K} 2b_n \cos(\omega_1 n) \geq H_d(e^{j\omega_1}) \\
&\qquad\qquad \vdots \\
&\qquad \delta W(\omega_p) + b_0 + \sum_{n=1}^{K} 2b_n \cos(\omega_p n) \geq H_d(e^{j\omega_p}) \\
&\qquad\qquad \delta \geq 0
\end{aligned}
$$

Here, $\omega_0, \ldots, \omega_p \in F$ represent a dense grid of frequency samples over which the metric $\max_{\omega \in \{\omega_0, \ldots, \omega_p\}} |W(\omega)E(e^{j\omega})|$ is minimized.

The MATLAB linear program solver `linprog` was used to solve this optimization problem. With the goal of finding the minimum-multiply filter within this framework, the design constraint K was varied and the program was repeatedly run until the minimum-K filter satisfying the design constraints was found. A convenient property of designing filters according to this problem statement is that the minimum K which falls below some maximum value $K_{max} \geq K$ can be found by solving approximately $\log_2 K_{max}$ linear programs.[12] For the set of design specifications assumed throughout this section, acceptable filters are obtainable when $K \geq 26$. The magnitude response and impulse response of the designed filter for $K = 26$ are respectively depicted in Figures 5-1 and 5-2.

The number of required multipliers when implementing this design using a direct-form structure is $2 \cdot 26 + 1 = 53$, and the number of required multipliers when using the structure in Figure 4-1 is $26 + 1 = 27$.

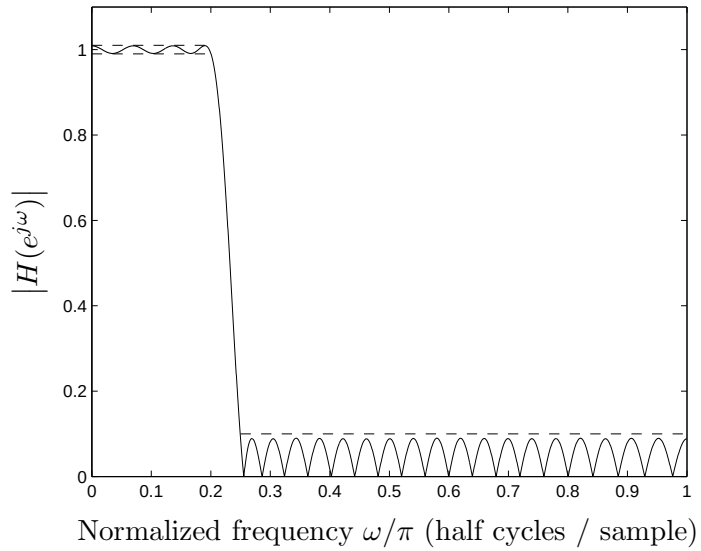


Figure 5-1: Magnitude response of filter which minimizes $\max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right|$ subject to $K = 26$. Dotted lines represent design specifications.

5.2.2 Filters with constrained coefficient values and minimum peak frequency error

Using the fact that any multiplier with coefficient value zero can be removed (and therefore costs nothing, according to the assumed efficiency metric), and noting that the impulse response in Figure 5-2 takes on several near-zero values, the question arises as to whether it is possible to set coefficients to zero and obtain a filter which still meets the design specifications. The alternation theorem, which provides a necessary and sufficient condition for having a unique, optimal design according to 5.9, does not preclude this possibility. Paraphrasing from [10], the alternation theorem states that given a filter $h[n]$ in the form of Equation 5.1, $h[n]$ is the unique filter which minimizes $\max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right|$ if and only if $W(\omega)E(e^{j\omega})$ exhibits at least $(K+2)$ alternations over the set $\omega \in F$. It does not, however, imply that there exist no other filters which, for fixed K, are feasible with respect to a fixed set of inequalities in frequency. In Figure 5-2, for instance, $h[13]$ and $h[-13]$ are very close to zero. As it turns out, explicitly setting to zero $h[13]$ and $h[-13]$ results in a filter which does not have minimum $\max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right|$ but which does still meet the stated design specifications.

Additionally, the alternation theorem makes no comparison between a solution $h[n]$ to

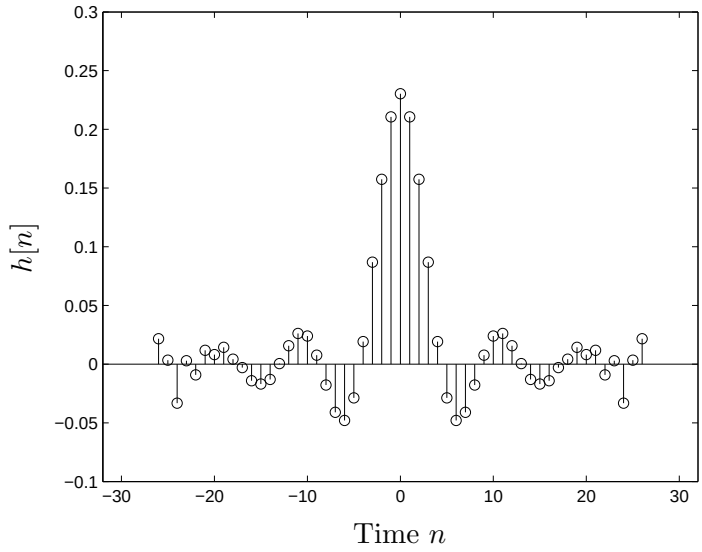


Figure 5-2: Impulse response of filter which minimizes $\max_{\omega \in F} \left| W(\omega) E(e^{j\omega}) \right|$ subject to $K = 26$.

5.9 and some other filter $\hat{h}[n]$ which satisfies $||\hat{h}[n]||_0 \leq ||h[n]||_0$ but which in the form of Equation 5.1 requires $\hat{K} > K$. This point is investigated by considering the case where a minimum peak error filter $h[n]$ is designed with fixed K and where certain values of $h[n]$ are constrained to be zero. The problem is more compactly stated as follows:

$$\text{minimize } \max_{\omega \in F} \left| W(\omega) E(e^{j\omega}) \right| \text{ subject to } b_n = 0 \ \forall n \in N_z, \ K \text{ fixed}, \tag{5.13}$$

where $N_z \subseteq [0, K]$. The associated set of feasible impulse responses is described by

$$C_{5.13} = \left\{ h[n] \ \middle| \ b_n = 0 \ \forall n \in N_z, \ K \text{ fixed} \right\}. \tag{5.14}$$

A close linear program approximation to this optimization problem which builds on the

formulation from the previous subsection is described by

$$
\begin{aligned}
&\text{minimize } \delta \\
&\text{subject to } \delta W(\omega_0) - b_0 - \sum_{n=1}^{K} 2b_n \cos(\omega_0 n) \geq -H_d(e^{j\omega_0}) \\
&\qquad \delta W(\omega_1) - b_0 - \sum_{n=1}^{K} 2b_n \cos(\omega_1 n) \geq -H_d(e^{j\omega_1}) \\
&\qquad\qquad \vdots \\
&\qquad \delta W(\omega_p) - b_0 - \sum_{n=1}^{K} 2b_n \cos(\omega_p n) \geq -H_d(e^{j\omega_p}) \\
&\qquad \delta W(\omega_0) + b_0 + \sum_{n=1}^{K} 2b_n \cos(\omega_0 n) \geq H_d(e^{j\omega_0}) \\
&\qquad \delta W(\omega_1) + b_0 + \sum_{n=1}^{K} 2b_n \cos(\omega_1 n) \geq H_d(e^{j\omega_1}) \\
&\qquad\qquad \vdots \\
&\qquad \delta W(\omega_p) + b_0 + \sum_{n=1}^{K} 2b_n \cos(\omega_p n) \geq H_d(e^{j\omega_p}) \\
&\qquad\qquad \delta \geq 0 \\
&\qquad\qquad b_n = 0 \quad \forall n \in N_z
\end{aligned}
$$

Denote $K_{5.9}$ as the variable K from 5.9 and Equation 5.11, and $K_{5.13}$ as the variable K from 5.13 and Equation 5.14. Furthermore, denote $\delta_{5.9} = \max_{\omega \in F} \left| W(\omega) E(e^{j\omega}) \right|$ for the optimal solution to 5.9 and $\delta_{5.13} = \max_{\omega \in F} \left| W(\omega) E(e^{j\omega}) \right|$ for an optimal solution to 5.13. If $K_{5.13} = K_{5.9}$, $C_{5.13} \subseteq C_{5.9}$, so $\delta_{5.13} \geq \delta_{5.9}$. In other words, 5.13 can give no smaller maximum weighted error than 5.9 for the same fixed K.

For $K_{5.13} > K_{5.9}$, however, it is not generally impossible that $\delta_{5.13} < \delta_{5.9}$. The relationship between $\delta_{5.13}$ and $\delta_{5.9}$ is therefore specifically governed for fixed $K_{5.9}$, $K_{5.13}$ ($> K_{5.9}$), $W(\omega)$, and $H_d(e^{j\omega})$ by choice of N_z. Furthermore, relating the zero-norm of a solution to 5.13, $||h_{5.13}[n]||_0$, with that of a solution to 5.9, $||h_{5.9}[n]||_0$, reveals little about how $\delta_{5.13}$ and $\delta_{5.9}$ generally compare. While this point may seem somewhat disparaging within the context of minimum-multiply filter design, it does at least leave open the possibility of finding filters from 5.13 which use fewer multipliers than designs from 5.9 and which have $\delta_{5.13} < \delta_{5.9}$. It is in fact possible to have any combination of $||h_{5.13}[n]||_0 < ||h_{5.9}[n]||_0$ or

$||h_{5.13}[n]||_0 > ||h_{5.9}[n]||_0$ with $\delta_{5.13} < \delta_{5.9}$ or $\delta_{5.13} > \delta_{5.9}$, for $K_{5.13} > K_{5.9}$. Since the cases where $||h_{5.13}[n]||_0 < ||h_{5.9}[n]||_0$ with $\delta_{5.13} < \delta_{5.9}$ and where $||h_{5.13}[n]||_0 > ||h_{5.9}[n]||_0$ with $\delta_{5.13} > \delta_{5.9}$ are perhaps the most interesting, examples of these situations follow.

Consider first the favorable scenario where $||h_{5.13}[n]||_0 < ||h_{5.9}[n]||_0$ and $\delta_{5.13} < \delta_{5.9}$ for $K_{5.13} > K_{5.9}$. The solution to 5.9 for $K_{5.9} = 26$ was presented in the previous subsection and is depicted in Figures 5-1 and 5-2. This design has $\delta_{5.9} = 0.088$ with zero norm $||h_{5.9}[n]||_0 = 53$. Solving 5.13 for $N_z = \{4, 9, 13, 17, 18, 22, 25, 26, 27, 29, 30, 31\}$ and $K_{5.13} = 32$, however, gives a design with $\delta_{5.13} = 0.075 < 0.088$ and with $||h_{5.13}||_0 = 39 < 53$, as illustrated in Figures 5-3 and 5-4. This solution meets the posed set of design specifications.

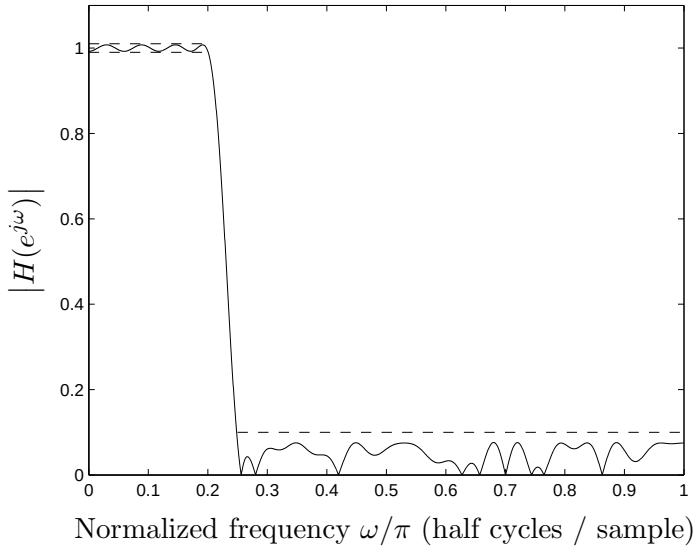


Figure 5-3: Magnitude response of filter which minimizes $\max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right|$ subject to $b_n = 0 \; \forall n \in \{4, 9, 13, 17, 18, 22, 25, 26, 27, 29, 30, 31\}$, $K = 32$. Dotted lines represent design specifications.

Consider now the unfavorable scenario where $||h_{5.13}[n]||_0 > ||h_{5.9}[n]||_0$ and $\delta_{5.13} > \delta_{5.9}$ for $K_{5.13} > K_{5.9}$. Solving 5.13 for $N_z = \{6, 22, 30\}$ and $K_{5.13} = 32$ gives a design with $\delta_{5.13} = 0.11 > 0.088$ and with $||h_{5.13}||_0 = 59 > 53$. This solution is shown in Figures 5-5 and 5-6 and does not meet the stated set of design specifications.

It is clear that this optimization problem is capable of producing filters with smaller peak error and zero norm than the Parks-McClellan algorithm. It has also been demonstrated, however, that this formulation can return filters with greater peak error and larger zero norm than a comparable Parks-McClellan design. The next two subsections therefore investigate

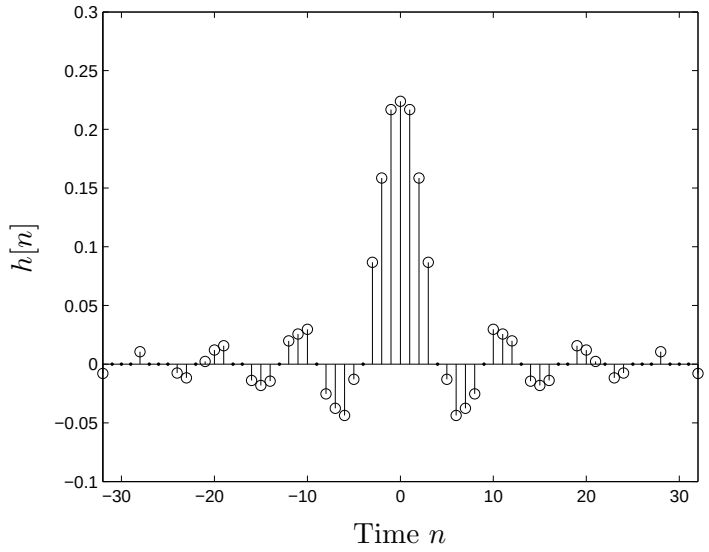


Figure 5-4: Impulse response of filter which minimizes $\max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right|$ subject to $b_n = 0 \ \forall n \in \{4, 9, 13, 17, 18, 22, 25, 26, 27, 29, 30, 31\}$, $K = 32$. Solid dots represent values of $h[n]$ which are constrained to be zero.

formulations which address this concern.

5.2.3 Filters with constrained peak frequency error and minimum $||h[n]||_1$

One way to obtain a filter with no greater zero norm than a corresponding Parks-McClellan result is to explicitly find a minimum zero norm filter subject to a set of feasible designs which is a superset of feasible Parks-McClellan results. This formulation is unfortunately nonlinear in the cost metric, so it cannot be described in terms of a single linear program. Because minimizing the one-norm of a function often yields results with small coefficients b_k, and because one-norm minimization can be cast as a single linear program, the following optimization problem is considered:

$$\text{minimize } ||h[n]||_1 \text{ subject to } \max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right| \leq \delta, \ K \text{ fixed} \tag{5.15}$$

This formulation is often used in the area of sparse approximation as a relaxation of minimizing $||h[n]||_0$,[7][13] and the corresponding close approximation can be cast in terms of a linear program as

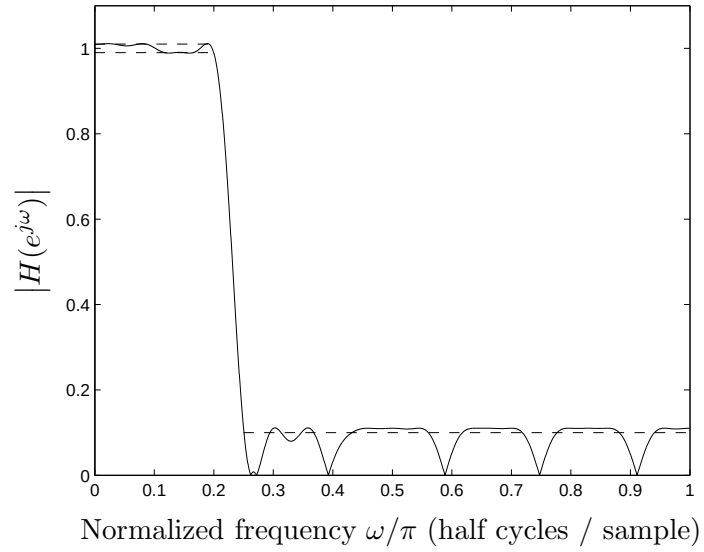


Figure 5-5: Magnitude response of filter which minimizes $\max_{\omega \in F} \left| W(\omega) E(e^{j\omega}) \right|$ subject to $b_n = 0 \ \forall n \in \{6, 22, 30\}, \ K = 32$. Dotted lines represent design specifications.

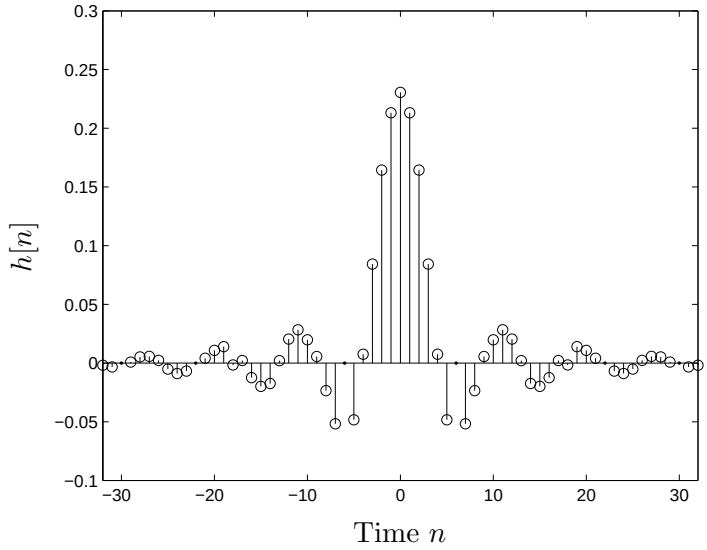


Figure 5-6: Impulse response of filter which minimizes $\max_{\omega \in F} \left| W(\omega) E(e^{j\omega}) \right|$ subject to $b_n = 0 \ \forall n \in \{6, 22, 30\}, \ K = 32$. Solid dots represent values of $h[n]$ which are constrained to be zero.

$$
\begin{aligned}
\text{minimize } & \varepsilon_0 + \sum_{n=1}^{K} 2\varepsilon_n \\
\text{subject to } & -b_0 - \sum_{n=1}^{K} 2b_n \cos(\omega_0 n) \geq -H_d(e^{j\omega_0}) - \delta W(\omega_0) \\
& \vdots \\
& -b_0 - \sum_{n=1}^{K} 2b_n \cos(\omega_p n) \geq -H_d(e^{j\omega_p}) - \delta W(\omega_p) \\
& b_0 + \sum_{n=1}^{K} 2b_n \cos(\omega_0 n) \geq H_d(e^{j\omega_0}) - \delta W(\omega_0) \\
& \vdots \\
& b_0 + \sum_{n=1}^{K} 2b_n \cos(\omega_p n) \geq H_d(e^{j\omega_p}) - \delta W(\omega_p) \\
& \varepsilon_0 + b_0 \geq 0 \\
& \vdots \\
& \varepsilon_K + b_K \geq 0 \\
& \varepsilon_0 - b_0 \geq 0 \\
& \vdots \\
& \varepsilon_K - b_K \geq 0 \\
& \varepsilon_0 \geq 0 \\
& \vdots \\
& \varepsilon_K \geq 0
\end{aligned}
$$

While solutions to 5.15 often have a number of small coefficients b_k and therefore come close to designs where many coefficients are set to zero, some method must be used after the optimization to determine which multipliers will be explicitly removed. The technique used here is simple thresholding; coefficients b_k with magnitude $|b_k| < \Gamma$ are, for some fixed Γ, set to zero. After this process, the resulting filter is no longer a minimal one-norm design, nor is it generally a minimal zero-norm design, nor is it even guaranteed to meet the posed set of design constraints. Because this approach is capable of designing filters which very nearly satisfy the requirements, often with smaller zero-norms than comparable Parks-McClellan

designs, the technique is presented as a valuable heuristic method.

As an example, consider the case where 5.15 is solved for $K = 32$ and threshold $\Gamma = 0.002$. The resultant filter, depicted in Figure 5-7, has zero-norm $||h[n]||_0 = 41$. This filter nearly meets the stated specifications, but violates the stopband constraint by a small amount, as illustrated in Figure 5-8. The impulse response of the final design is depicted in Figure 5-9. While this filter does not strictly meet the set of design specifications, it is an acceptable design with respect to a slightly-relaxed set of tolerances.

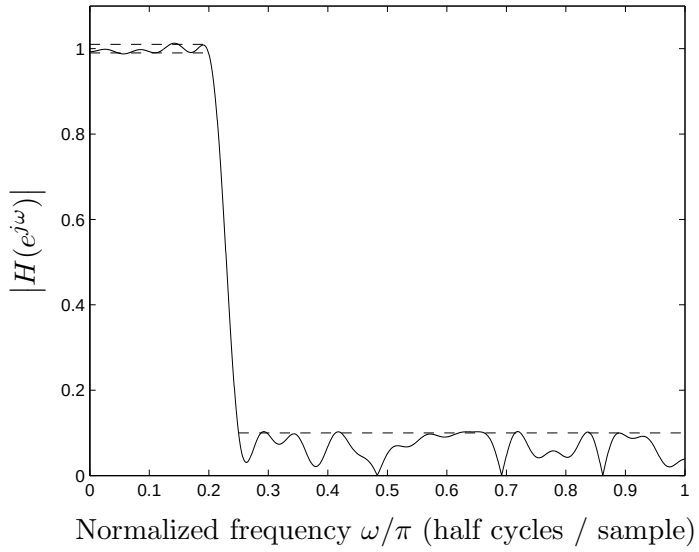


Figure 5-7: Magnitude response of filter designed by minimizing $||h[n]||_1$ subject to $\max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right| \leq \delta,\ K = 32$, followed by setting coefficients b_k with $|b_k| < 0.002$ to zero. Dotted lines represent design specifications.

5.2.4 Filters with constrained peak frequency error and minimum $||h[n]||_0$

As was mentioned in the previous subsection, explicitly finding a minimal zero-norm filter which meets the posed set of design constraints is equivalent to obtaining a minimum-multiply filter under the assumption that multipliers with value zero can be removed. The optimization is more formally stated as

$$\text{minimize } ||h[n]||_0 \text{ subject to } \max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right| \leq \delta,\ K \text{ fixed} \tag{5.16}$$

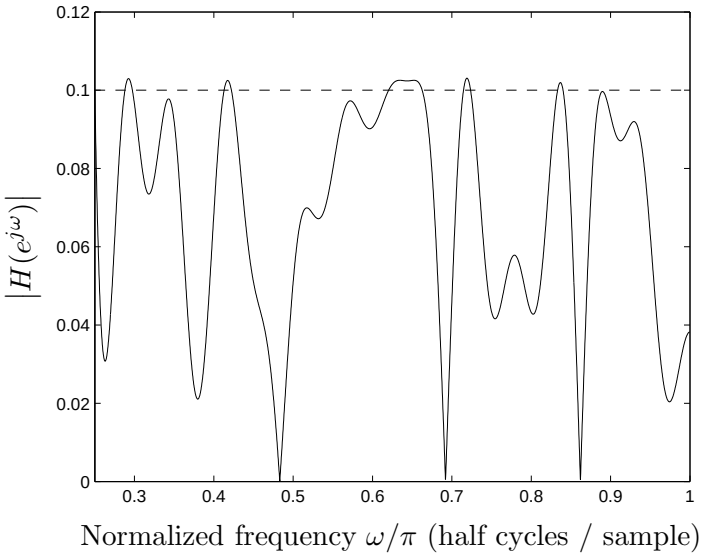


Figure 5-8: Detail of stopband in Figure 5-7, illustrating that this filter violates the stopband design constraint by a small amount.

While this problem cannot be solved in terms of a single linear program, an algorithm can be prescribed in terms of multiple linear programs. Specifically, solving 5.13 2^{K+1} times, where each time N_z is chosen from the set of all subsets of $[0, K]$, and picking the overall feasible solution with smallest zero-norm, results in a minimal zero norm design. While this problem may be computationally tractable for very small K, setting $K = 32$, for instance, requires over 4 billion iterations, and using 33 multipliers is already well within the limits of many practical design budgets. Still, certain properties associated with this formulation are worth noting. In particular, the following two points relate iterations of 5.13 for the same K and can be used to arrive at a pruned search scheme for solving 5.16 more efficiently:

- Solution to 5.13 for $N_z = A$ meets the posed design constraints $\Rightarrow$ solutions to all problems 5.13 for $N_z \subseteq A$ meet the posed design constraints
- Solution to 5.13 for $N_z = A$ violates the posed design constraints $\Rightarrow$ solutions to all problems 5.13 for $N_z \supseteq A$ violate the posed design constraints

Since the topology of the pruned search tree depends largely on the particular choice of design constraints, it is difficult in general to predict what kinds of gains in efficiency may be expected. A solver which uses these techniques was implemented in MATLAB and run on a PowerMac G5 for $K = 32$. The program was manually terminated after three weeks

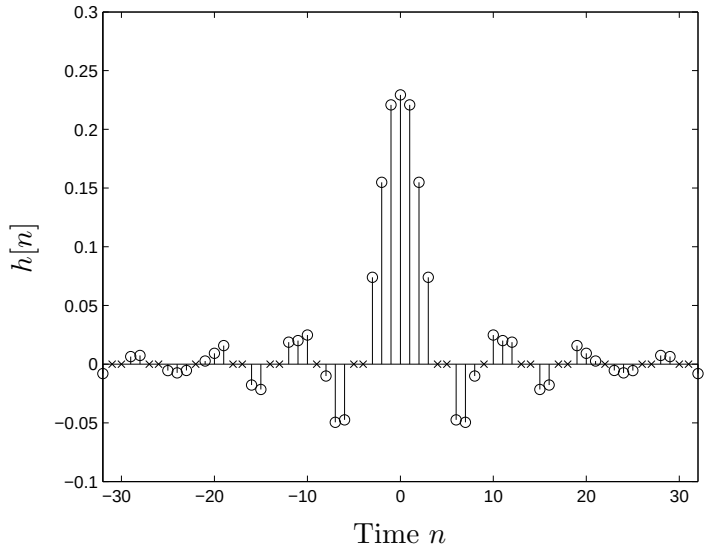


Figure 5-9: Impulse response of filter designed by minimizing $||h[n]||_1$ subject to $\max_{\omega \in F} \left|W(\omega)E(e^{j\omega})\right| \leq \delta,\ K = 32$, followed by setting coefficients b_k with $|b_k| < 0.002$ to zero. Symbols $\times$ represent values of $h[n]$ which were zeroed.

of runtime without arriving at an optimal solution, underscoring the relevance of 5.15 as a relaxation of 5.16.

5.3 Comparison of results

In this section, the computational cost of the downsampler-by-4 in Figure 5-10 is evaluated for the different filter designs and structures discussed in this thesis. Table 5.3 summa-

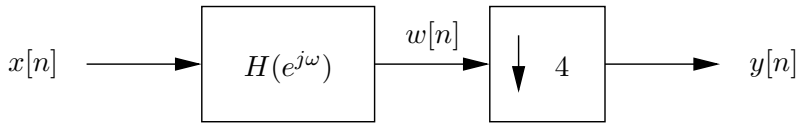


Figure 5-10: Downsampling system for comparing computational cost of filter designs.

rizes results from each design method. The variable $\mathcal{C}$ represents the number of required multiplications per output sample. Contrast, in particular, the cost of implementing a Parks-McClellan design (Figure 5-1) using a direct-form structure with that of implementing the design in Figure 5-3 using the presented structure. Both systems meet the stated design constraints, but the latter has significantly reduced computational cost.

Figure	Optimization problem	$\mathcal{C}$, direct-form structure	$\mathcal{C}$, folded structure	$\mathcal{C}$, polyphase structure	$\mathcal{C}$, presented structure
5-1	$\min \max_{\omega \in F} \left\lvert W(\omega)E(e^{j\omega}) \right\rvert$ s.t. K fixed	212	108	53	27
5-3	$\min \max_{\omega \in F} \left\lvert W(\omega)E(e^{j\omega}) \right\rvert$ s.t. $b_n = 0 \ \forall n \in N_z$, K fixed	260	132	65	21
5-7	$\min \lvert\lvert h[n] \rvert\rvert_1$ s.t. $\max_{\omega \in F} \left\lvert W(\omega)E(e^{j\omega}) \right\rvert \leq \delta$, K fixed; $\lvert b_k \rvert < 0.002$ zeroed	260	132	65	21

Table 5.3: Required number of multiplications per output sample for different filter designs and structures in a downsampler-by-4 configuration.

Bibliography

[1] John W. Adams. FIR digital filters with least-squares stopbands subject to peak-gain constraints. *IEEE Transactions on Circuits and Systems*, 39(4):376–388, April 1991.

[2] John W. Adams and James L. Sullivan. Peak-constrained least-squares optimization. *IEEE Transactions on Signal Processing*, 46(2):306–321, February 1998.

[3] Maurice Bellanger. *Digital Processing of Signals: Theory and Practice*. John Wiley & Sons, Chichester, 1984.

[4] T. A. Classen and W. F. Mecklenbräuker. On the transposition of linear time-varying discrete-time networks and its application to multirate digital systems. *Philips J. Res.*, 23:78–102, 1978.

[5] Ronald E. Crochiere and Lawrence R. Rabiner. Optimum FIR digital filter implementations for decimation, interpolation, and narrow-band filtering. *IEEE Transactions on Signal Processing*, (5):444–456, October 1975.

[6] Ronald E. Crochiere and Lawrence R. Rabiner. *Multirate Digital Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1983.

[7] Lorenzo Granai and Pierre Vandergheynst. Sparse approximation by linear programming: Measuring the error with the ℓ_1 norm. Technical Report TR-ITS-2005.028, ITS, June 2005.

[8] Sverre Holm, Bjrnar Elgetun, and Geir Dahl. Weight- and layout-optimized sparse arrays. *Proceedings of the 1997 Workshop on Sampling Theory and Applications, SampTA-97*, pages 97–102, June 1997.

[9] Alan V. Oppenheim and Ronald W. Schafer. *Digital Signal Processing*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1975.

[10] Alan V. Oppenheim, Ronald W. Schafer, and John R. Buck. *Discrete-Time Signal Processing*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1999.

[11] T. W. Parks and J. H. McClellan. Chebyshev approximation for nonrecursive digital filters with linear phase. *IEEE Transactions on Circuit Theory*, pages 189–194, March 1972.

[12] K. Steiglitz, T. W. Parks, and J. F. Kaiser. METEOR: a constraint-based FIR filter design program. *IEEE Transactions on Signal Processing*, 40(8):1901–1909, August 1992.

[13] Joel A. Tropp. Just relax: Convex programming methods for identifying sparse signals. *IEEE Transactions on Information Theory*, 51(3):1030–1051, March 2006.

[14] Daniel B. Turek. Design of efficient digital interpolation filters for integer upsampling. Master's thesis, Massachusetts Institute of Technoloy, June 2004.

[15] P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1993.