Gordana Jovanovic Dolecek *Editor*

Advances in Multirate Systems

Springer

Advances in Multirate Systems

Gordana Jovanovic Dolecek
Editor

Advances in Multirate Systems

Springer

Editor
Gordana Jovanovic Dolecek
Department of Electronics
Institute National INAOE
Tonantzintla, Puebla
Mexico

ISBN 978-3-319-59273-2 ISBN 978-3-319-59274-9 (eBook)
DOI 10.1007/978-3-319-59274-9

Library of Congress Control Number: 2017949358

© Springer International Publishing AG 2018
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Multirate signal processing has become a key topic enabling efficient techniques in many areas of modern engineering such as wireless and satellite communication systems, software and cognitive radio, and image and video processing, among others. The main advantage of multirate systems is rooted in the computational efficiency based on the ability to use different sampling rates simultaneously in the same system.

Despite numerous existing techniques on multirate signal processing for digital information processing and different promising future applications, there are only a few edited books which include some of the applications of multirate systems in different fields. The reason for proposing a new book in this area is to overcome a gap existing between numerous results published in journals and conferences and no recently edited books. The goal is to include recent important results and promising future applications of multirate systems.

The text is divided into eight chapters written by experts in the field.

In the first chapter, "Implementation Studies of Multirate Systems," written by Y. Huang and C. Li, the implementation aspects of multirate systems are discussed. Two case studies are elaborated. The first study investigates the design challenge for high-speed and very high-bandwidth up-sampling filter for transmitter. The second study particularizes design challenge in wide frequency coverage for down-sampling filter for multi-standard radio receiver.

The next chapter, titled as "Advances in Multirate Filter Banks: A Research Survey," written by A. Kumar, B. Kuldeep, I. Sharma, G. K. Singh, and H. N. Lee, advances a research survey on filter banks including a general review of filter bank theory and the state of the art in filter bank design. The elaboration of future advancement in the field of optimal filter bank design is also presented.

The third chapter, "Methods for Improving Magnitude Characteristic of Comb Decimation Filters," written by G. Jovanovic Dolecek, presents some recent methods for improving the magnitude characteristic of comb decimation filters in stopband, in passband, as well as in both. First, the methods for aliasing rejection improvement, using comb zero rotation, based on exploring the characteristics of symmetrical polynomials, are presented. In the following the methods for comb passband droop compensation in a wideband are presented, based on a trigonometrical approach, in which the magnitude responses of compensators are in sinusoidal forms. Finally, the method based on multiplierless corrector filters, for the improvement of comb magnitude characteristic in passband, as well as in folding bands, is elaborated. The methods are illustrated with examples and MATLAB scripts are provided for presented methods.

The following chapter, "Design of Multi-Channel Filter Bank Using Minor Component Analysis and Fractional Derivative Constraints," written by B. Kuldeep, A. Kumar, G. K. Singh, and H. N. Lee, presents a new design technique for multichannel cosine-modulated filter banks, based on minor component analysis and fractional derivative constrains, using swarm optimization techniques. Problem formulation and proposed design methodology are elaborated and illustrated with examples.

Chapter 5, "Multiresolution Filter Banks for Pansharpening Application," written by H. Hallabia, A. Kallel, and A. B. Hamida, tackles the image signal processing application. Two channel filter banks are adopted to fuse remotely sensed imagery, also called a pansharpening. This process consists in transferring the spatial content of panchromatic (PAN) image at finer resolution into an image at coarse resolution, e.g., multispectral (MS) or hyper-spectral (HS) image. Experimental results, including datasets, the selected pansharpening algorithms, and the quality assessment metrics, are included.

The next chapter, "Video Signal Processing," written by Y. L. Huang, is devoted to video signal processing applications. The background knowledge, applications, and technical details of current multirate video systems are presented. First, the basic concept and knowledge of the video system are introduced. In addition, several examples of multirate video applications are shown. Afterward, the key techniques to achieve these applications from the fundamental frame rate conversion (FRC) and the advanced frame rate up-conversion (FRUC) are explained. The general flow diagram of the FRUC techniques is given, and the technical details are also discussed. Additionally, several evaluation methods are explained and the popular video datasets are shown. Finally, the requirement and recent researches of hardware implementation for FRUC techniques are discussed.

Chapter 7, "Multirate Systems in Cognitive Radio," written by S. C. Prema and K. S. Dasgupta, elaborates the use of filter banks for spectrum sensing in cognitive radio (CR), including cosine-modulated filter banks and DFT filter banks. Multirate filter bank techniques can reduce computational complexity and improve spectral analysis in cognitive radio applications. For a fractional utilization of spectrum, the center frequency and spectral edges of the primary user can also be estimated using filter banks.

Chapter 8, "Design of Nonuniform Linear-Phase Transmultiplexer System for Communication," written by A. Vishwakarma, A. Kumar, and H. N. Lee, presents an improved design technique for a nonuniform linear-phase transmultiplexer filter bank (FB) for communication system. The prototype filter is designed using different window functions that have high side-lobe falloff rate (SLFOR). Next, the filter coefficients are optimized to satisfy perfect reconstruction (PR) condition. The performances of the method are evaluated in terms of fidelity parameters such as inter-symbol interference (ISI), interchannel or inter-carrier interference (ICI), signal to inter-symbol interference ratio (SISI), signal to interchannel interference ratio (SICI), and signal to total interference ratio (SI). The simulation results demonstrate that very low values of ICI and ISI can be obtained using various adjustable windows.

Puebla, Mexico Gordana Jovanovic Dolecek

Contents

Implementation Studies of Multi-rate Systems

Yanxiang Huang and Chunshu Li

1 Introduction

This chapter discusses the implementation aspects of the advance in multi-rate signal processing. Two case studies of the multi-rate filters are presented: the first is the up-sampling filter for wireless transmitters, and the second is the down-sampling filter for receivers. The design challenge for the first up-sampling case is the ultimate high speed, since the data rate is very high. Therefore, in this case study, an important consideration is the relative sequence of different functionalities to ensure system requirements. The coexistence with other functionalities is also discussed, since they have cross effects on each other. For the second case, the design challenge is the wide frequency coverage. The focus of discussion is the filter architecture selection to meet a good trade-off of power consumption and area.

With the aggressive scaling of CMOS technology, the area utilization and power consumption of digital circuit are dropping rapidly. Therefore, more and more signal processing that was previously realized in the analog domain is now migrated to the digital domain. The most notable applications for multi-rate signal processing are up-sampling and down-sampling. Reducing the sampling rate requires an anti-aliasing filter prior to the decimation to a lower sampling rate. Increasing the sampling rate, on the other hand, also requires an anti-imaging filter after the interpolation. These two filters are specified using the original low-pass filter specification. To achieve any gain in computational efficiency, the two filters must run at the reduced sampling rates.

Y. Huang
IMEC, Heverlee, Belgium

KU Leuven, Leuven, Belgium

C. Li (✉)
Marvell, Santa Clara, CA, USA
e-mail: chunshu.li.zju@gmail.com

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9_1

These up-sampling and down-sampling functionalities are realized by FIR (finite impulse response) filters. For an FIR filter, let $x[n]$ be the discrete input, and c_i be the FIR coefficient; the output $y[n]$ are computed as

$$y[n] = \sum_{i=0}^{N} c_i x[n-i]$$

This computation is also known as discrete convolution.

Figure 1 depicts the implementation diagrams of an FIR filter. Figure 1a shows the default implementation of the decimation stage, which down-samples the signal after the low-pass FIR filter. The drawback is, however, half of the computation is wasted because they are discarded by the down-sampler. To ensure area and power efficiency, the high-frequency part should be kept as small as possible. The realistic implementations (see Fig. 1 structure (b) and (c)) perform the filtering after the down-sampler.

This chapter illustrates the design methodology with two case study filters. The first multi-rate filter case is the up-sampling filter used for polar transmitters, which exploits the mm-wave frequency band. The 60 GHz mm-wave frequency band

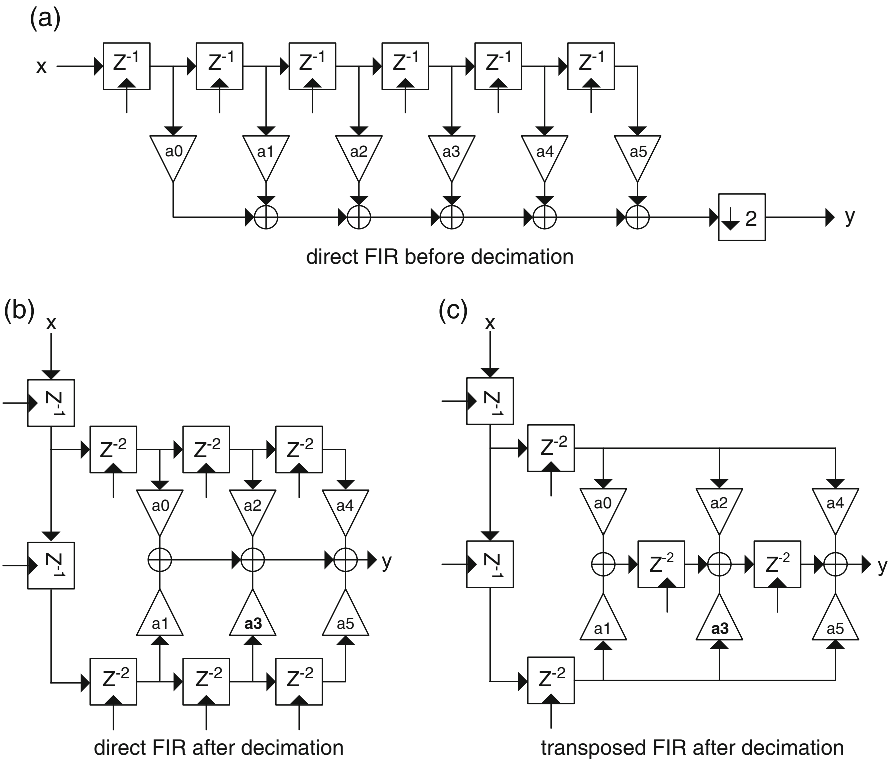


Fig. 1 FIR filter implementations

provides as wide as 7 GHz unlicensed bandwidth. The design challenge is the ultimate high speed, as the bandwidth and hence the sample rate is very high. This case study focuses on the high-level design aspects of the multi-rate system, for instance, the relative configuration of the up-sampling filter and order of the up-sampling filter with regard to other functionalities.

The second multi-rate filter case is the down-sampling filter for multi-standard radio receivers. The receiver supports HD Radio, DAB Radio, and DVB-T. The digital down-sampling decimation filter provides flexibility in those standards. The design challenge is the wide-frequency coverage, i.e., the input is as high as 4.096 Gsps and the output is as low as 1 Msps. Therefore, in this section, the emphasis is on the implementation details of the decimation down-sampling filter itself. For this design, the high-speed filters favor multiplier-less FIR filter to meet timing, and the low-speed filters employ one or few multipliers to perform all the FIR multiplications in sequence to save area cost.

2 Up-sampling System for Polar Transmitters

There is ample license-free bandwidth around the 60 GHz frequency (Fig. 2). This large spectrum still has little commercial utilization, which is in contrast to the over-crowded spectrum below 10 GHz. The 802.11ad [1] targets the throughput as

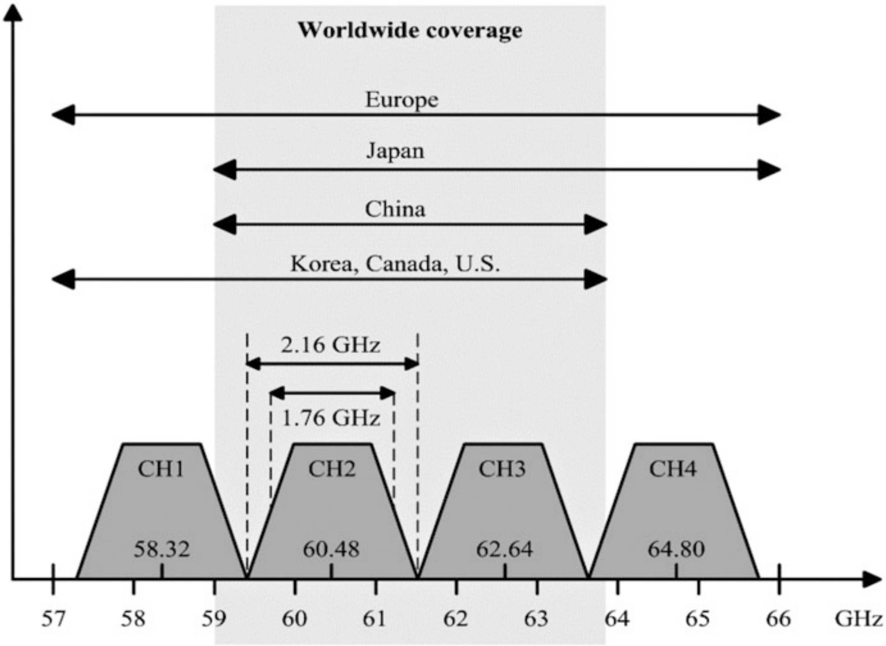


Fig. 2 Ample license-free bandwidth around 60 GHz

high as 6.75 GHz data rate in wireless personal area network (WPAN). However, despite the advantages in free bandwidth, the PA (power amplifier) in transmitters consumes a lot of power, especially in phased array transmitters where a lot of PAs are employed. The main reason is that transmission at 60 GHz covers much less distance for a given transmission power, mainly due to the high free-space path loss. To overcome the high signal losses [2], phased array antennas are typically employed. This will drastically increase the power consumption.

Improving the power efficiency of PA is critical in reducing the power cost of mm-wave transmitter. Most 60 GHz PAs operate in the class A linear mode [3, 4], due to the use of variable envelope modulations that are required for high data rates and high spectral efficiency. This causes the typical PA power efficiency of less than 5%.

In order to improve the PA power efficiency, the PA needs to work in its nonlinear region to utilize the peak efficiency. The polar architecture is one interesting solution that allows the PA to operate in saturation. In the polar transmission, the PH (phase) signal goes to the PA, while the AM (amplitude) is applied to the PA through a separate modulation path and combined with PH signal by modulating the supply.

Previously, the modulation and processing are mainly realized by analog circuits, mainly due to the high-speed requirement that is too costly for digital circuits. However, there are three design challenges for those analog-centric designs:

1. AM–AM distortion: the PA is subject to the effects of the supply voltage on linearity [5]. For CMOS devices, voltage gain is usually a strong function of the drain-source potential, which leads to PA gain changing with supply voltage. This contributes to the AM–AM distortion. Voltage-dependent capacitances in active devices exhibit high nonlinearity. The supply voltage can change the bias conditions on these capacitances, resulting in PH shift dependent on the overall output impedance, causing AM–PM distortion.
2. PA nonlinearity: the second concerns the linearity of the envelope detector. Precise reconstruction of the envelope signal is vital for the performance of polar transmitters. However, the analog envelope detector suffers from circuits' nonlinear behavior and produces spectral regrowth.
3. AM–PM distortion: the PA exhibits a phase shift that depends on the input amplitude of limiters at high frequencies. The phase shift increases as the input amplitude decreases using the analog limiter for constant envelope signal generation, which will distort the transmitted signal depending on the input signals' amplitude.

In recent years, due to the advance in digital CMOS scaling, more previously analog-dominated circuits are transformed into the digital-centric methodology. This also applies to the polar transmitter for mm-wave communication system.

For digital-intensive polar transmitters, the polar concept is expanded to the whole transmitter, rather than only in the RF domain [6]. The polar conversion is performed with digital signal processing. The AM signal can then digitally modulate a variable-size PA. This avoids modulating the supply and also eliminates the need for an additional RF limiter and AM detection circuits, which would introduce extra nonlinearity and bandwidth limitations.

Although the digital polar transmitter has many advantages, the design challenges on DSP (digital signal processing) front end need to be analyzed and tackled. For the 60 GHz application, the DSP usually works at a very high speed depending on the required oversampling factor. This complicates the timing closure and increases the power consumption.

In this section, the designing of an energy-efficient digital front end for such polar transmitter working in the 60 GHz band is analyzed. Section 2.1 depicts the digital-intensive polar transmitter and the functions of the digital front end, with the focus on the order of multi-rate system design. This is achieved by extensive system simulation, with regard to error vector magnitude and output spectrum. It allows to systematically optimize the design requirements on the DSP front end. Section 2.2 co-optimized the multi-rate system in algorithm and architecture level, to minimize the power consumption. Extra emphasis is paid to the multiplier-less filter implementation, as well as the multiplier-less Cartesian-to-polar coordination conversion. Section 2.3 demonstrates the design results.

2.1 *Up-sampling Multi-rate Polar System Architecture*

The polar radio transmitter (Fig. 3) comprises a digital front end and an analog front end [7]. The digital front-end system comprises DSP blocks which receive rectangular signals and output an AM signal and PH signals sin($\theta(t)$) and cos($\theta(t)$). The AM and the PH signals are then fed to the analog front end of the radio transmitter.

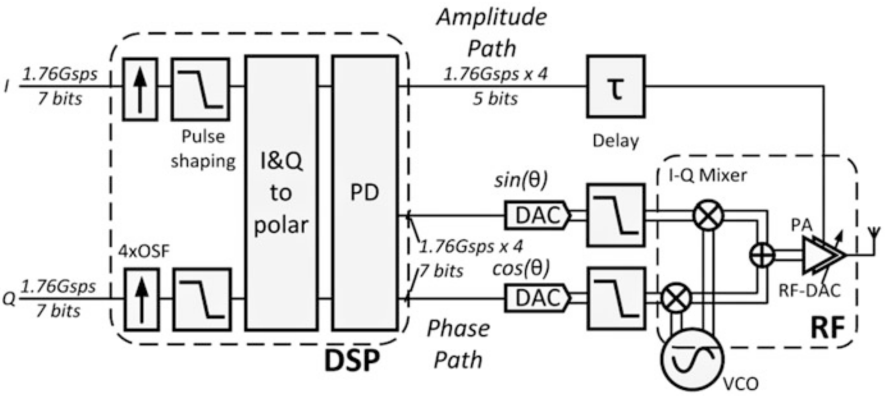


Fig. 3 Multi-rate 60 GHz polar transmitter. The DSP is a multi-rate system. The frequency (order of the filter and polar conversion) is yet to be discussed below

Each respective PH signal is first converted to an analog PH signal by DAC followed by analog baseband low-pass filters and then fed to the mixers to generate the up-converted, to radio frequency, and to PH signals. The up-converted PH signals are then summed and fed to the input of the PA. The digital AM signal is fed to the analog to directly control the operation of the PA. The PA comprises a number of amplifying unit cells, whose operation is directly controlled by the digital AM signal.

2.1.1 Multi-rate System Functionality Overview

This part gives a functional description of each multi-rate system block, i.e., the pulse-shaping filter, rectangular-to-polar conversion, pre-distortion block, and digital front-end of a polar transmitter.

Pulse-Shaping Filter The nonlinear transformation from rectangular signals to polar signals broadens the spectrum. Figure 4a depicts the PSD (power spectral density) of the rectangular signal, which is compliant with the spectrum mask of IEEE 802.11ad [1]. After nonlinear conversion to polar signals, the spectrum of the converted signal greatly expands, as shown in Fig. 4b. To avoid EVM (error vector magnitude) degradation from the spectrum overlap due to expansion after conversion, the rectangular signal needs to be firstly oversampled and digitally filtered

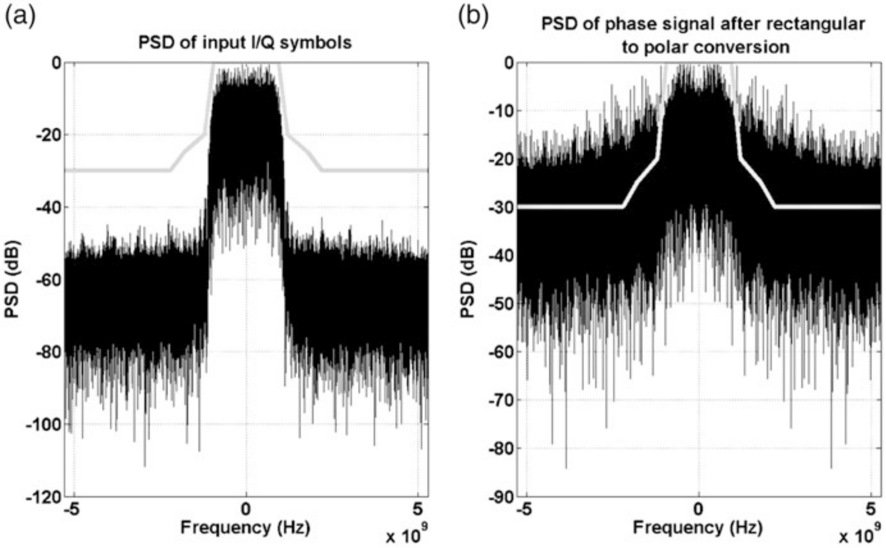


Fig. 4 The signal spectrum will expand, due to the nonlinear rectangular-to-polar conversion. (**a**) PSD of input *I*/*Q* symbols. (**b**) The PSD of the sine and cosine components after polar conversion

before converting to a polar signal. The first residual image after oversampling appears at an offset equal to the sampling frequency. For a symbol rate of, e.g., 1760 Msps in IEEE802.11ad application, an OSF (oversampling factor) of at least 6 is normally required to meet the spectrum mask requirements, by moving the first residual image out of the RF band of 802.11ad standard spanning from 57 to 66 GHz. However, even with a SOA (state-of-the-art) technology, the implementation of filters working at 10.56 (1.76*6) Gsps with reasonable power consumption is challenging.

Rectangular-to-Polar Conversion The digital rectangular-to-polar conversion can be mathematically computed from the Cartesian signals, i.e., an in-phase (I) and quadrature (Q) signal, as follows:

$$\begin{aligned} A &= \sqrt{I^2+Q^2} \\ \sin\theta &= \frac{Q}{\sqrt{I^2+Q^2}} \\ \cos\theta &= \frac{I}{\sqrt{I^2+Q^2}} \end{aligned}$$

The conversion equation involves multiple complex computations, e.g., square root, trigonometric, and division computations. Power-efficient implementation of these complex computations is challenging as well. Although COordinate Rotation DIgital Computer (CORDIC), first described in 1959 by Jack E. Volder [8], is an algorithm for these kinds of complex computations, its energy-efficient implementation for this high-speed conversion needs intensive optimization [9].

Pre-distortion Block The PA suffers from AM–AM and AM–PM distortions, as illustrated before. The distortion causes spectral regrowth and devastates the constellation diagram. In order to obtain high power efficiency while ensuring good linearity to fit high-order QAM signals, digital pre-distortion is needed. This PA linearization using lookup table (LUT) was widely used to linearize the PA, which simply uses the error feedback signal to construct the inverse function and generates the LUT. Considering that the AM–AM and AM–PM responses of the PA depend only on the input AM, the two-dimensional LUT in the mapping pre-distorter can actually be replaced by two one-dimensional LUTs. Indexed by the input amplitude $A(t)$, the one-dimensional LUTs specify the desired output AM, $A^{'}(t)$, and the phase shift, $\Delta\theta(t)$. Then the pre-distorted output is given by

$$Z(t) = A^{'}(t)e^{j(\theta(t)+\Delta\theta(t))}.$$

As the pre-distortion LUTs are in polar form, in practical implementations, the phase pre-distortion for each input complex point is computationally intensive, which involves multiple complex trigonometric and multiplication computations operating at oversampled throughput (e.g., 10.56 Gsps for OSF of 6).

The power budget for the DSP front-end was analyzed in [10]. In order for the DSP front-end to have a minor influence on the total power budget, a value of 10% of the total TX power consumption should be considered. This concludes an average of around 50 mW for the extra digital processing required for the polar operation [10].

The above analysis puts a challenging task on the optimization of algorithms and design techniques of the additional signal-processing circuitry. The 50 mW budget for signal-processing power consumption needs to cover 10,560 (1760 × 6) Msps complex mathematical computations analyzed above. Aggressive algorithms and architecture optimizations are explored to achieve this target.

2.1.2 Multi-rate System Design

Modeling with MATLAB, a complete multi-rate system for IEEE 802.11ad transmission chain is optimized, with the aim of output signals EVM and PSD. To be specific, the order of oversampling and rectangular-to-polar conversion blocks, the OSF in the DSP digital filtering block, and quantization accuracy of DSP conversion block are optimized.

IEEE 802.11ad standard specifies −21 dB EVM for single-carrier 16-QAM modulation. Taken as the design goal is −30 dB with a design margin of 9 dB, considering its application for deep-scaled CMOS implementation. The order of polar conversion and oversampling is studied. Various approaches are depicted in Fig. 5.

For the setup in Fig. 5b, the conversion is placed before the oversampling. This setup enjoys the advantage that the IQ–polar conversion is operating at the not-oversampled frequency. Therefore, the implementation cost is low.

However, as the EVM diagram shows (in Fig. 6), no matter how large the OSF is applied, the EVM stays around −20 dB, which is not acceptable. This result can be explained straightforwardly, since the rectangular-to-polar conversion without oversampling beforehand introduces significant spectrum overlap in both the amplitude and phase signals.

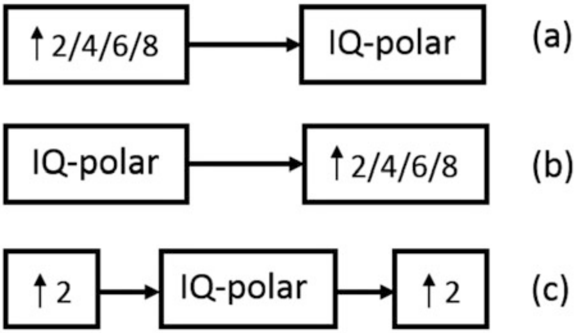


Fig. 5 Different orders of oversampling and polar conversion. (**a**) IQ-polar conversion after upsampling. (**b**) IQ-polar conversion before upsampling. (**c**) IQ-polar conversion between upsampling

Fig. 6 EVM is insufficient even with OSF8 for the setup in Fig. 5b

Figure 5c shows another approach by putting the oversampling partially before the conversion, combining with the effect of bandwidth limitation on phase path. The rectangular signal is first oversampled by a certain OSF and then converted to polar form. After this, a filter is put in the phase path to imitate the effect of bandwidth limitation.

As the results shown in Fig. 7, with no bandwidth limitation on phase path, the signal is hardly distorted, which means only the spectrum overlap due to conversion will not distort the signal. Spectrum overlapping with the filtering afterward is the main reason for performance degeneration. As shown in the figure, a combination of two OSF and polar conversion followed by a 7 GHz filter will distort the signal quality with final EVM higher than −30 dB. This means OSF2 conversion–OSF2 processing order, as shown in Fig. 5c, will neither work, as the OSF2 after conversion is equivalent to applying a low-pass filtering with a bandwidth of 1.76 GHz × 2.

Therefore, the only workable architecture is to have enough oversampling before rectangular-to-polar conversion (Fig. 5a). As explained above, an OSF of at least

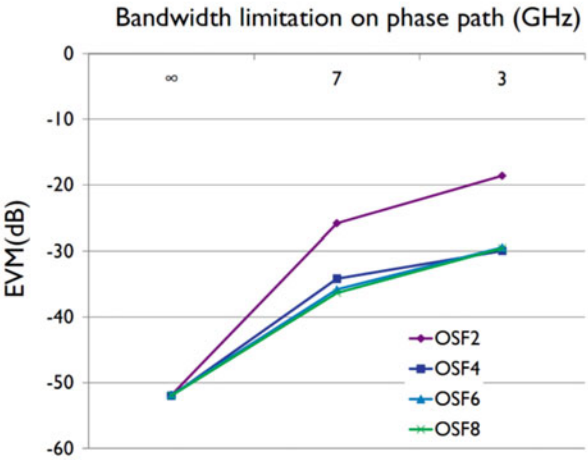


Fig. 7 Bandwidth limitation results of EVM make the setup of Fig. 5c unfavorable

6 is normally required to avoid the first alias located in the RF band of 802.11ad standard spanning from 57 to 66 GHz. To reduce the DSP front-end design challenge, this work explores the possibility of using OSF of 4 to suppress the alias below the spectrum mask, in combination with the suppression from the analog Butterworth baseband filter in the phase path.

The passband width and filter order of the analog baseband filter exhibit design trade-off between the output EVM and alias rejection ratio, wider passband or smaller filter order keeps more significant signals in the transmission output, which leads to better EVM. However, the alias is less suppressed which may violate the spectrum mask. Extensive simulations with different combinations of OSF, passband width, and filter order were conducted for trade-off.

Figure 8 presents the EVM results in terms of converted AM and PH quantization accuracies with (1) OSF of 4, (2) input I/Q signals of 7 bits, and (3) 2 GHz passband width of second-order analog Butterworth filter. Note that although there are multiple choices of quantization accuracies in Fig. 8 to achieve the −30 dB EVM, the ones with fewer bits of AM signal should be chosen to make layout easier when routing the digital AM bit wires to the variable-size PA. The point at the turning corner (5-bit AM and 7-bit PH) gives the best trade-off between AM and PH quantization accuracies. Simulation with this quantization accuracy shows −30 dB suppression on the first alias residing at 6.16 GHz. The resulting polar transmitter is shown in Fig. 3 with key design specs annotated.

2.2 Digital Front-End Implementation

This section discusses the detailed implementation of the multi-rate system, which is (1) the oversampling and phase-shaping filter, (2) the rectangular-to-polar converter, and (3) the pre-distortion unit for PA nonlinearity.

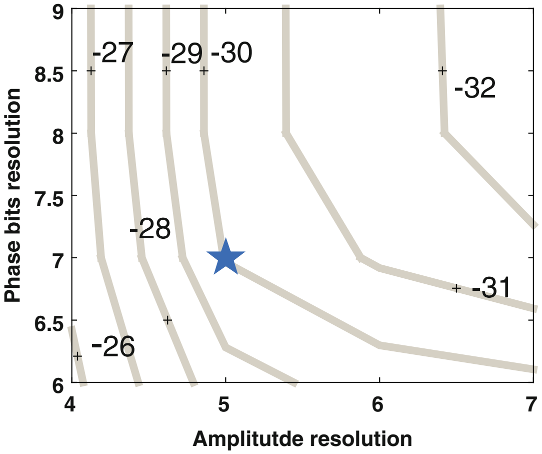


Fig. 8 Quantization of PH and AM with regard to the EVM

2.2.1 Multiplier-Less Phase-Shaping Filter

The function of the digital filter is to shape the frequency spectrum of the quadrature signals so that the signal at the output of the filter is compliant with the spectrum mask requirements. Instead of using computation-intensive-raised cosine filter for the pulse shaping, we use Cascaded Integrator-Comb (CIC) filter [11], which is a special case of FIR filter, to shape the spectrum compliant.

Figure 9 depicts the output spectrum of the signal at the output of an example implementation using three CIC-4 filters, of which the overall transfer function is shown in Fig. 9. The output spectrum is compliant with the 802.11ad spectrum mask.

The straightforward implementation with the choice of three CIC-4 is very challenging since it requires multipliers operating at 7.04 Gsps. This would require an implementation in carefully designed custom logic, and the power consumption could be very high. Since the pulse shaping will have to implement low-pass filtering operations by means of finite impulse response (FIR) filters following oversampling, the polyphase implementation of FIR filters is possible [12]. The transfer function of the equivalent filter is shown in Fig. 10, where these three CIC-4 tap filters are combined into a polyphase one.

Simplification of this filter is shown in Fig. 11. The transfer function of each of the FIR filter can be decomposed into a combination of two operations, i.e., an addition and a multiplication. Advantageously, the multiplication can be further simplified to multiplications with binary values; the latter can be implemented in digital domain simply with left shift operations. As a result, the digital filter can be implemented with a simple circuit comprising only of summation and left shift circuits working at 1.76 Gsps, which significantly reduces the area and power footprint of the digital front-end system.

As the coefficients of the decimation filters are predefined, the multiplications in those CIC filters are MCM (multiple constant multiplications). Instead of using

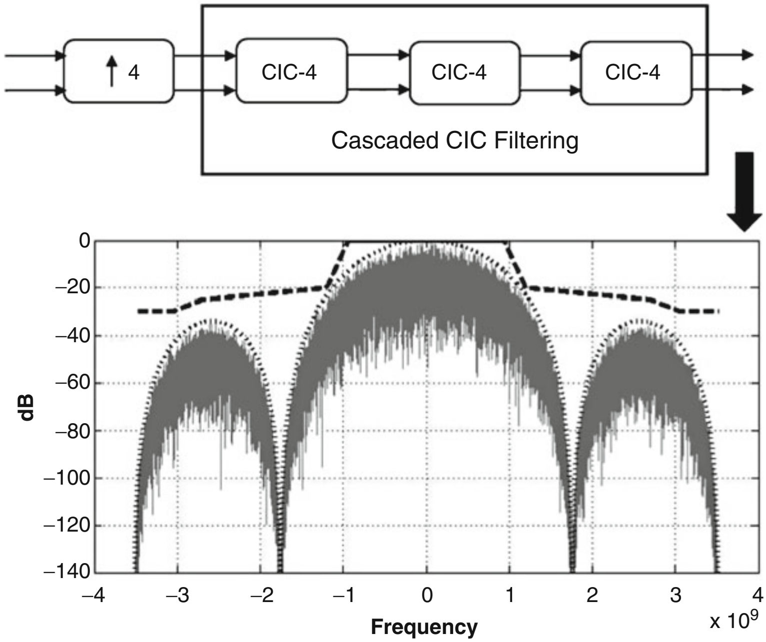


Fig. 9 CIC-4 filters shape the spectrum to be compliant to 802.11ad standard

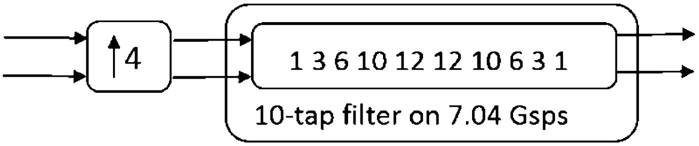


Fig. 10 Detailed transfer function of the three CIC-4 filters shown in Fig. 9

hardware multipliers, the multiplications are done by shift and adders. This forms a multiplier-less implementation of the CIC filter.

The shift-and-adder circuits are defined by the multiplication coefficients, which are encoded into canonical signed digit (CSD) format [13]. With CSD encoding, each coefficient bit is encoded into 1, −1, or 0. Therefore, most bits of the coefficients are zero, which reduces the number of addition operations for the MCM. Besides, some parts of two coefficients might be the same, so their shift-and-adder hardware can be shared, which further reduces area cost and power consumption.

Two different adder structures are considered, carry-ripple and carry-save adders. As the name indicates, the carry of carry-ripple adder propagates from the least significant bit (LSB) to the most significant bit (MSB). The carry-ripple addition cannot meet the timing constraint. One way to speed up the multiplication

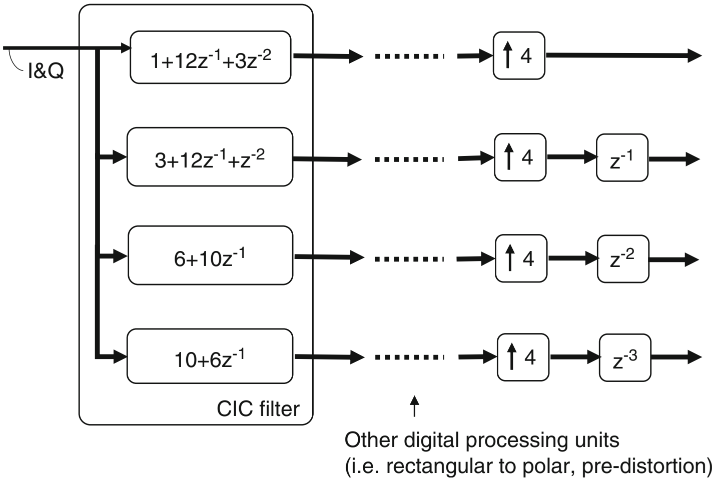


Fig. 11 Polyphase implementation of the phase-shaping filter

is to use carry-save arithmetic [14]. CSA (carry-save adder) is usually used to compute the sum of several numbers [15–18]. It differs from all other adders in that the outputs consist of two numbers of the same word length as the inputs, one contains the partial sum bits, and the other contains the carry bits. The idea is to route the carry signals vertically down to the next stage instead of horizontally within the same adder stage. The computation of CSA is faster, because the result is propagated down as soon as it becomes available, as opposed to propagating further within the stage. This scheme is, hence, called carry save and is used in this work.

Fundamental arithmetic operations are redesigned to preserve the signal in CS format. In total, 12 operations using standard cells are designed, which are binary (number) plus binary (number), binary minus binary, negative binary minus binary, carry save plus binary, carry save minus binary, negative carry save plus binary, negative carry save minus binary, carry save plus carry save, carry save minus carry save, negative carry save minus carry save, right shift, and CS tree addition. The outputs of all operations are in CS format. CS tree additions are designed with Dadda tree structure [19]. Moreover, they are designed smartly, with bit-level delay optimization capability that considers input delay for each bit to improve worst case delay [20]. The structure of the implementation details of the filters is further discussed in the second case study.

The output signals of the polyphase CIC filters are then fed to the conversion circuits and then to the pre-distortion circuits. After the pre-distortion circuits, at the very end of the digital front-end system, the signal paths are recombined. The implementation of the digital filter thus defines the number of signal paths of the digital front-end system.

2.2.2 Multiplier-Less Rectangular-to-Polar Convertor

CORDIC [8, 21–23] optimizes complex rectangular-to-polar conversion computations by decomposing the desired rotation angle into the sum of predefined elementary rotation angles such that the rotation through each of them can be accomplished with simple shift-and-add operations. The flow diagram of a circular CORDIC is shown in Fig. 12.

For polar conversion to get AM and PH, the circular CORDIC works in vectoring mode. In this mode, the direction of each rotation (d_i in Fig. 12) is determined by the sign of y_i. A positive y_i leads to a clockwise i-th rotation and vice versa. In this way, the vector is continuously approaching positive x-axis. The final x_i value will approximate the required AM signal, and the sum of each rotated angles (z_i column in Fig. 12) generates the PH value.

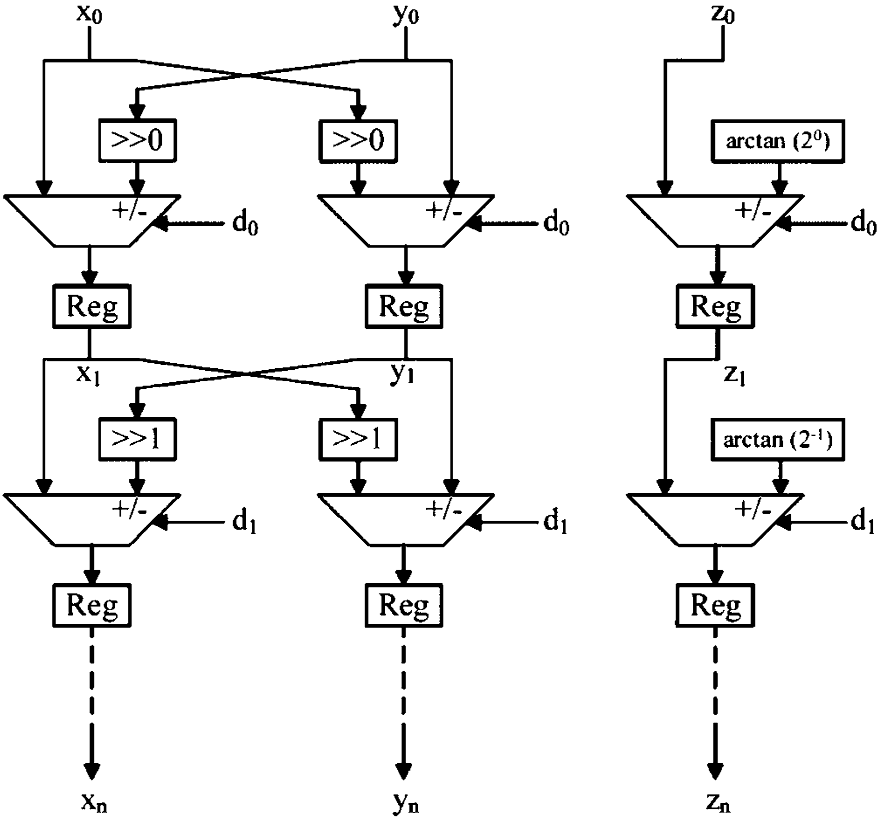


Fig. 12 Flowchart of a CORDIC

For the explored DSP conversion block, the PH signal itself is not required, but the sine and cosine functions of the PH value. Four candidate methods are explored in this work to get this. Figure 13 presents the first two candidate structures. Both candidates use a circular CORDIC working in vectoring mode to get the required AM signal and PH signal for further calculation. The following discussions are made based on the assumption that the resolutions for ($\sin(\theta)$ and $\cos(\theta)$) are both 7 bits.

Candidate 1 employs an additional circular CORDIC working in rotation mode to get ($\sin(\theta)$, $\cos(\theta)$). The working principle is shown in Fig. 14: whenever the vectoring CORDIC makes a rotation, the rotation-mode CORDIC starting with a unit vector rotates at the same angle but in an opposite direction. The rotation-mode CORDIC will realize a vector of ($\sin\theta$, $\cos\theta$).

Since the PH signal is hidden in the structure and no longer necessary, the z column in Fig. 12 for both CORDICs can be removed. This structure hence needs four calculation columns all in all. In reality, six (instead of seven) CORDIC

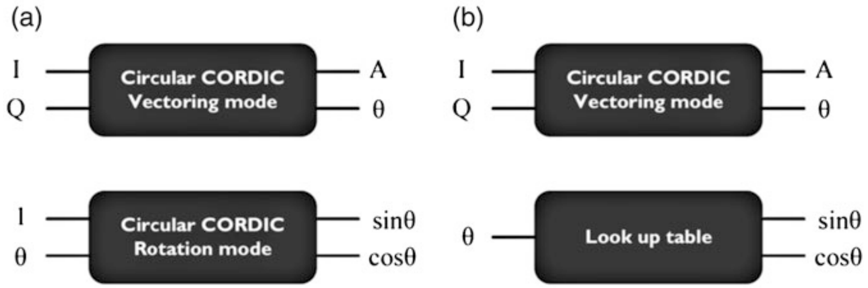


Fig. 13 (**a**) Candidate 1 and (**b**) candidate 2 for polar conversion

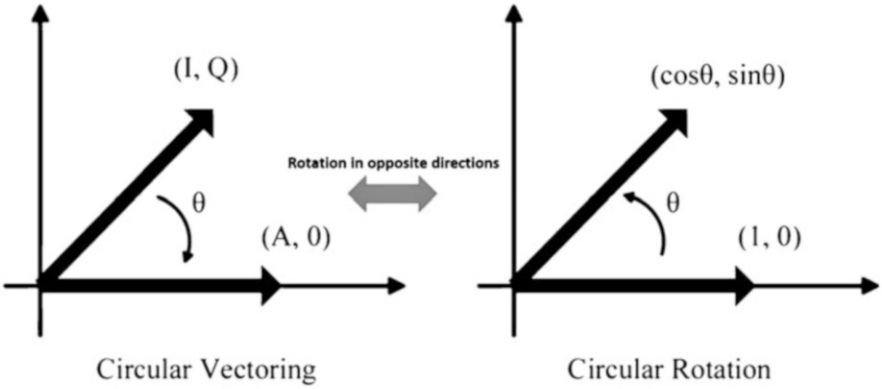


Fig. 14 Working mechanism of candidate 1

conversion stages are required, since the first conversion is substituted by folding the I and Q signals into the [0, $\pi/2$] domain. Each CORDIC stage utilizes a 7-bit adder for one column, although the exact adder width differs slightly in reality, to minimize quantization and overflow errors.

Therefore, candidate 1 employs $4 \times 6 \times 7 = 168$ full adders (FA). If the speed requirement is challenging, the adder microarchitecture will be changed and hence more FAs are utilized. Nevertheless, the number serves as a good estimation to compare the complexity with other candidates.

Candidate 2 uses a lookup table (LUT) with PH value from the vectoring CORDIC as the index to generate the required ($\sin\theta$, $\cos\theta$). The sign of the ($\sin\theta$, $\cos\theta$) can be derived easily from I and Q domain folding. Therefore, six multiplexer (MUX) stages are required to obtain ($\sin\theta$, $\cos\theta$). The data width for each stored ($\sin\theta$, $\cos\theta$) pair is $6 + 6 = 12$ bits. In sum, it requires 26–1 MUXs of 12 bits width. Besides the area cost, a bigger challenge for this LUT-based method comes from the required high throughput ($7.04 = 1.76 \times 4$ Gsps).

The read access time of a state-of-the-art ROM is reported to be 0.72 ns [24], which is still far away from the design requirement. It is admitted that the speed requirement can be met with a deep pipeline. However, that requires a huge amount of flip-flops to register the intermediate lookup values, which adds extra power and area cost to the system.

Another two candidates for the conversion block are presented in Fig. 15. These two candidates make use of CORDIC to do division operations. The linear CORDIC working in vectoring mode transforms (x, y, z) to (x, 0, $z + y/x$), which can be used to calculate a division. During the iteration, x remains a fixed value.

Therefore, two calculation columns are needed to calculate either $\sin\theta$ or $\cos\theta$. In total, six calculation columns are needed for candidate 3, which requires $6 \times 6 \times 7 = 252$ FAs. Compared to candidate 1, besides the extra two columns, another problem comes from the longer latency to generate the final ($\sin\theta$, $\cos\theta$), since

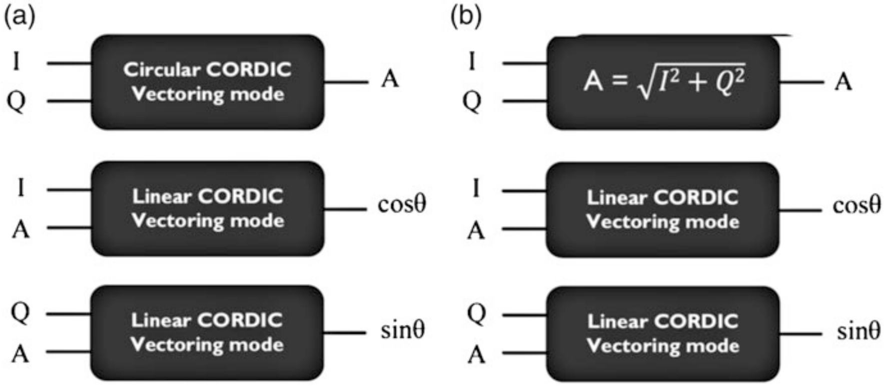


Fig. 15 (**a**) Candidate 3 and (**b**) candidate 4 for polar conversion

Table 1 Resource utilization of four candidate implementations of the polar converter, assuming 7-bit ($\sin\theta$, $\cos\theta$) resolution

Resources	FA	MUX	Others
Candidate 1	168	0	
Candidate 2	84	63× 6bits	
Candidate 3	252	0	
Candidate 4	$14 + 84 + 168 = 266$	0	2× 7-bit multiplier

the circular and linear CORDICs in candidate 3 have to work in series. Candidate 4 computes AM straightforwardly, which consists of two 7-bit multiplications, a 14-bit addition, and a hyperbolic CORDIC ($2 \times 6 \times 7 = 84$ FAs) to calculate the square root. Obviously, the complexity of candidate 4 is higher than the others.

Table 1 summarizes the complexity of those four candidates. Note that the area and power consumption of an FA are similar to a 1-bit MUX. Candidate 1 is superior in terms of resource utilization. Besides, it also has other advantages described above, e.g., speed and latency. Therefore, the first candidate is chosen for implementation.

Figure 16 shows a more detailed scheme of the conversion circuit. The conversion circuit comprises in total four computation columns, the first two columns forming the vectoring-mode CORDIC and the last two columns the rotation-mode CORDIC. Each computation column is characterized with a pipeline architecture, where the different pipeline stages are defined by the sequential logic circuits connecting the different stages to one another.

Each computation column of the first CORDIC (i.e., the first two computation columns) is provided with a folding circuit to fold the respective input samples to the first domain, followed by a number of computation circuits with each to perform a single rotation with a predetermined angle of rotation $\varnothing_i$. The direction of the rotation is defined by the most significant bit of y_i. Therefore, the first computation circuit rotates each respective sample with the same direction of rotation, d0, the second computation circuit with a direction of rotation d1, and so on. As shown in the Fig. 16, the vectoring-mode CORDIC comprises six computation circuits connected to one another by a sequential logic circuit.

Each computation column of the second CORDIC (i.e., the last two computation columns) comprises the same number of computation circuits as the vectoring-mode CORDIC, followed by an unfolding circuit arranged to unfold the resulting quadrature PH signals to the correct domain. For example, if in the vectoring-mode CORDIC, the input samples are folded from the second domain to the first domain, then in rotation-mode CORDIC, the output sample is unfolded from the first domain to the second domain. Each computation circuit performs a single rotation of its respective input signal with the same predetermined rotation angle di as in the vectoring-mode CORDIC. The angle of rotation φ in each respective computation circuit in both CORDIC processors at the same pipeline stage is the same; however, the direction of the rotation is opposite. At the end of the conversion, the overall rotation angle is

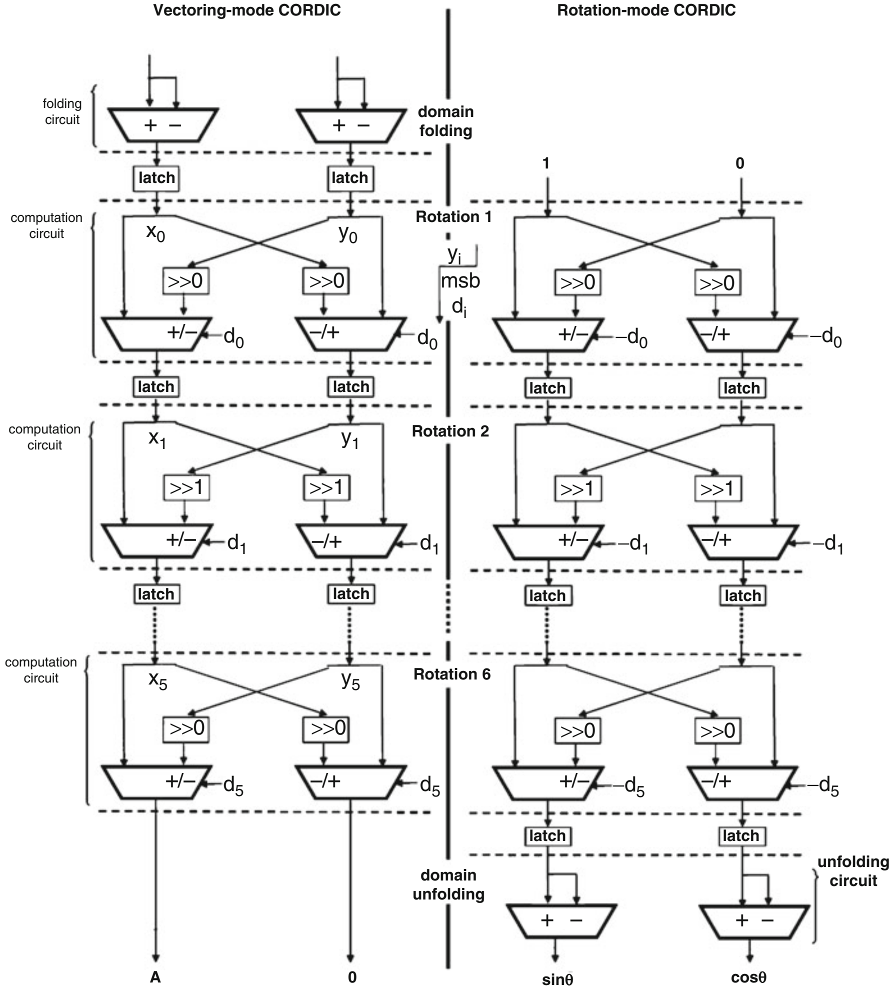


Fig. 16 Detailed architecture of the rectangular-to-polar conversion

$$\varnothing = \sum_{i=1}^{n} \varnothing_i$$

As a result, the vectoring-mode CORDIC will calculate a vector having (A, 0) spherical coordinates, while the rotation-mode CORDIC will calculate a vector with (sinθ, cosθ) spherical coordinates.

2.2.3 Pre-distortion of PA Nonlinearity

A pre-distortion circuit is necessary in case the PA's linearity is not sufficient enough. Although in a polar radio transmitter the PA can operate in saturation mode, which allows for high-power efficiency, the PA may suffer from AM–AM and AM–PM nonlinearities. These distortions cause spectral regrowth and devastate the constellation diagram.

As shown in Fig. 17, the pre-distortion signal is derived from an LUT based on the value of the AM signal. The value of the amplitude signal is used as an index to derive from the LUT two pre-distortion signals, one, $\Delta\theta(t)$, being a pre-distortion value for the phase signals and a second one, $\Delta A(t)$, being a pre-distortion value for the AM signal. The pre-distorted AM signal is created by summing the AM signal with the value $\Delta A(t)$ derived from the LUT. Similarly, the pre-distorted PH signals are created by operating on respective PH signal with and the value $\Delta\theta(t)$ derived from the LUT.

$$A(t)' = A(t) + \Delta A$$

$$\sin(\theta'(t)) = \sin\big(\theta(t) + \Delta\theta(t)\big) = \sin(\theta(t))\cos(\Delta\theta(t)) + \cos(\theta(t))\sin(\Delta\theta(t))$$

$$\cos(\theta'(t)) = \cos\big(\theta(t) + \Delta\theta(t)\big) = \cos(\theta(t))\cos(\Delta\theta(t)) - \sin(\theta(t))\sin(\Delta\theta(t))$$

To simplify the implementation complexities of the complex trigonometric computations in pre-distorting phase signals, the equation is simplified using Taylor's approximation [25]. Considering the fact that the distorted phase $\Delta\theta$ is limited within 10° from the analog circuit simulation, the equation is simplified as follows:

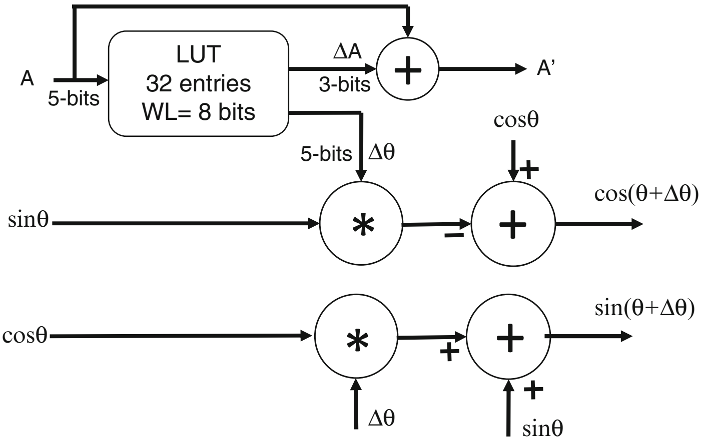


Fig. 17 Pre-distortion unit to handle nonlinearity

$$\sin(\theta'(t)) \approx \sin\left((\theta(t)\right) + \cos(\theta(t))\Delta\theta(t)$$
$$\cos(\theta'(t)) \approx \cos\left((\theta(t)\right) - \sin(\theta(t))\Delta\theta(t)$$

The LUT size can be limited to store the values of $\Delta A(t)$ and $\Delta\theta(t)$. In this implementation shown in Fig. 17, the LUT contains 32 entries indexed by the input AM, with each entry containing three bits for distortion amplitude $\Delta A(t)$ and five bits for $\Delta\theta(t)$. The power supply to the pre-distortion circuit can be dynamically shut down when the PA in the polar transmitter offers sufficient linearity. This allows to further reduce the power consumption.

2.3 Results Measurement

The multi-rate system for high-bandwidth polar transmitter was processed in a standard 28-nm CMOS technology [26]. A micrograph of the chip is shown in Fig. 18. The complete design area is as small as 0.036 mm^2, of which the pre-distortion unit utilizes 0.015 mm^2. The power consumption is 39 mW.

The power spectral density (PSD) of the DFE outputs is shown in Fig. 19. Both the in-band PSD and the alias rejection are confirmed to be compliant with the spectrum mask. The EVM of the produced signal is measured to be −30.5 dB.

Figure 20 plots the constellation for 16-QAM signals. The EVM is as good as −30.5 dB. The nice purity of the signals is demonstrated clearly. This ensures its feasibility of applying to polar transmitters.

Fig. 18 Die shot of the digital front-end processor

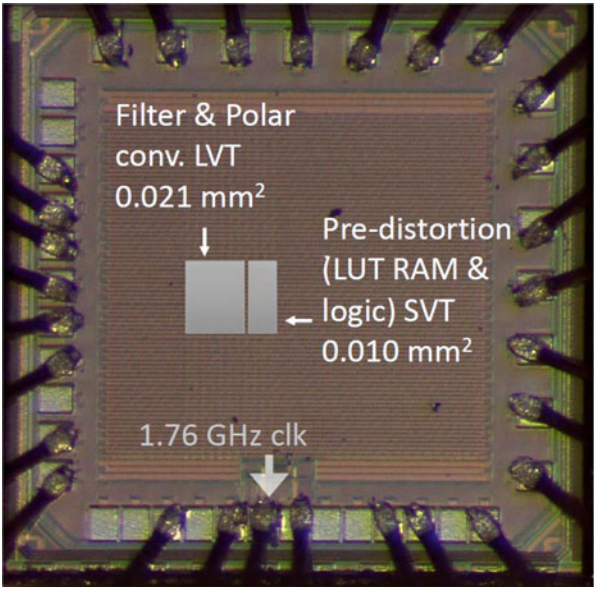


Fig. 19 Output PSD of combined PH and AM signals

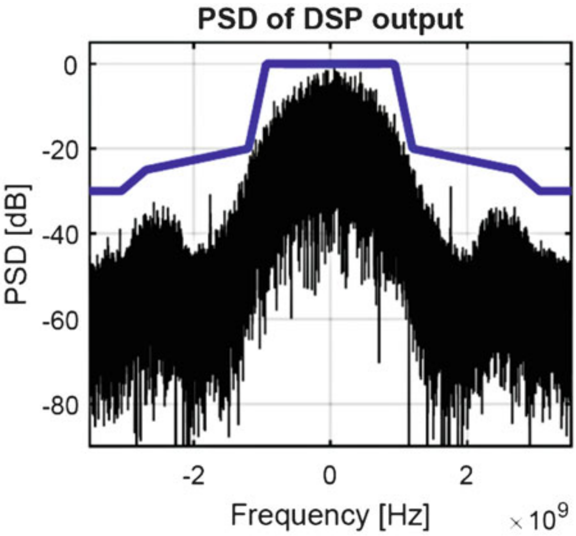
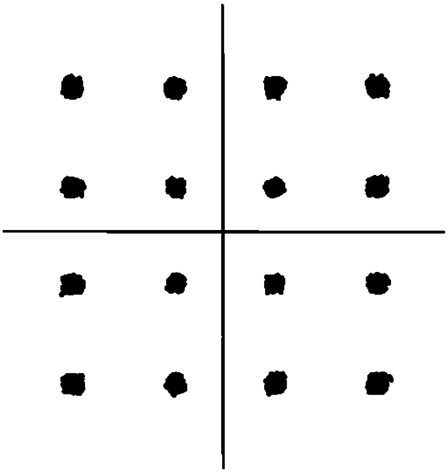


Fig. 20 Constellation of 16-QAM outputs of the digital front end



3 Down-Sampling System for Multi-rate Radio Receiver

Modern ADCs (analog-to-digital converters) are driven by the increasing demands for signal bandwidth and dynamic range from digital transceiver applications, e.g., multi-standard digital radio and advanced LTE (long-term evolution). The sigma–delta technique is a possible candidate to realize these ADCs with high bandwidth and high dynamic range.

For a sigma–delta ADC, the analog input signal is oversampled at a frequency much larger than the Nyquist frequency, to increase the signal-to-noise ratio.

Therefore, the first stage when it comes to digital domain is to down-sample the output signal from sigma–delta ADCs. The down-sampling is commonly realized using the well-known sample rate reduction concepts based on low-pass and polyphase decimation filters [27]. For a complete decimation chain, it is more efficient if the sample rate is reduced in multiple stages [28]. As a result, the decimation process is decomposed into a chain of decimation filters running at different frequencies, i.e., from 1.024 GHz to 1 MHz.

For the digital transceiver, the finite impulse response (FIR) filter design is typically challenging when the input is the oversampled high-speed signal. At such high frequencies, the filter circuits may operate at a speed close to the limit given by technology, which incurs large area cost and power consumption by gate upsizing. Considering that a number of commonly used algorithms in digital signal processing, e.g., FIR filters, infinite impulse response (IIR) filters, and fast Fourier transform (FFT) filters, can be simplified to basic arithmetic operations like multiplication, addition, and shifting operations; optimizing those arithmetic units is important to boost system performance.

3.1 Down-Sampling Multi-rate System-Level Design

A high-level model of a chain of decimation filters for wideband and multi-standard car entertainment digital radio receiver is considered as an input to this work. The structure of the whole receiver is shown in Fig. 21. The ADC, which is placed close to the antenna to simplify the analog front-end design, produces a 4.096 GHz signal for the decimation filter. The filter is divided into high-frequency (HF) and low-frequency (LF) parts. The output signals can be used for multiple applications: DVB-T, DAB Radio, or HD Radio. The input sample rate of the HF decimation filter is 4.096 GHz. Considering the requirements from the high-precision calibration circuit, the input signal resolution for the HF filter is as high as 13 bits.

To reduce the filter complexity, the whole decimation process is decomposed into 11 FIR decimation filters (Fig. 22). The first two filters (FIR1–FIR2) constitute the HF filter section, and FIR3–FIR11 constitute the LF filter section. The HF and

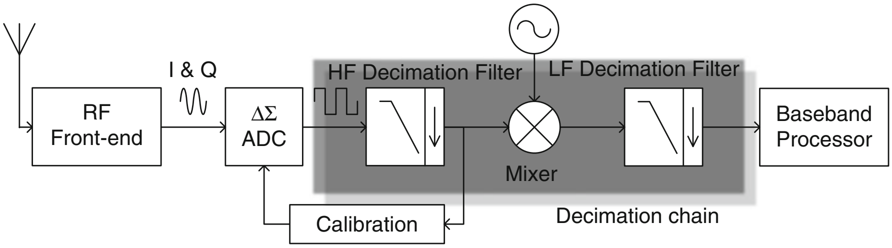


Fig. 21 Multi-rate down-sampling filter used in the receiver for multi-standard car entertainment radio

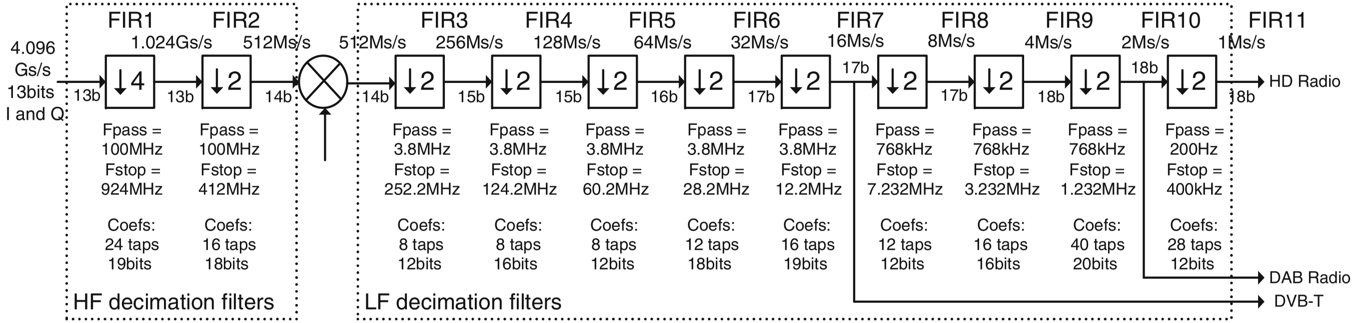


Fig. 22 Decimation strategy of the multi-rate down-sampling filter, decomposed into 11 stages

LF sections are separated by a frequency-shifting operation using CORDIC or multiplication.

The filter parameters of individual filters are determined by MATLAB simulation [17, 18]. Each filter of FIR2–FIR11 decimates the signal with a factor of two. Since a decimation filter is running at the speed of the output sample rate [29] and the input sample rate is 4096 Msps, the operating frequency of FIR1 would be as high as 2.048 GHz if its decimation factor is 2. That frequency requirement is extremely challenging for the targeted 45 nm technology. Therefore, FIR1 decimates the signal by a factor of 4, which enables it to run at 1.024 GHz, while at the cost of requiring more taps in the FIR filters.

After decomposing the decimation process into HF and LF sections, we explore the suitable architectures and for them in the subsequent sections.

3.2 Detailed Filter Architecture Designs

In this subsection, various detailed realizations of each FIR filter are illustrated.

For the HF (high-frequency) decimation FIR filters, the main design constraint is the extremely high data rate. Therefore, transposed form FIR filters (Fig. 1c) are chosen, because of their shorter critical path than direct form FIR filters.

The delay constraints of LF (low-frequency) filters are less stringent than HF filters. So the design challenge is to minimize the area. Therefore, we propose to employ one or two high-speed MAC units to execute the computation for every tap in the FIR filter to reduce area. In these circumstances, because of the symmetry coefficients, half computations are saved for a direct form FIR filter. Therefore, direct form FIR filters (Fig. 1b) are applied in LF filtering.

3.2.1 High-Frequency FIR Filters

As mentioned in the first case study, the coefficients are fixed, and therefore MCM computations are used. An MCM example is shown in Fig. 23. The MCM multiplies the same input signal with six different coefficients and outputs six corresponding products (G1–G6). Since the right-shift operations for binary numbers can be implemented using only wires in hardware without incurring any delay or area cost, they are not shown in this diagram. Note that in the diagram of the MCM, the addition symbol may also represent a subtraction operation, due to the CSD notation.

The transposed FIR architecture using MCMs in traditional binary format is called MCM-B. One example is shown in Fig. 24. The products for the same group (G) are added together by a tree adder.

An alternative solution is keeping numbers in CS format throughout the CIC filtering, which is called MCM-CS. It can reduce the delay compared with MCM-B.

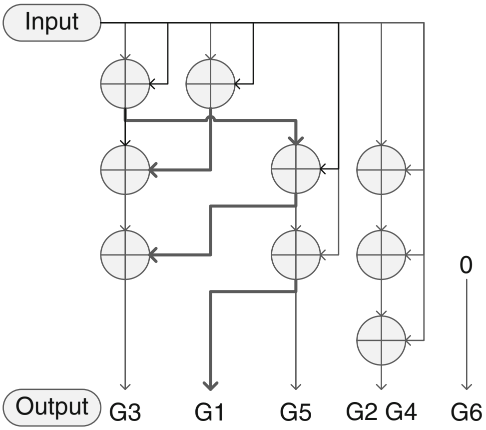


Fig. 23 An example implementation of the CIC filter. The input is multiplied with different coefficients and producing six different results (G1–G6)

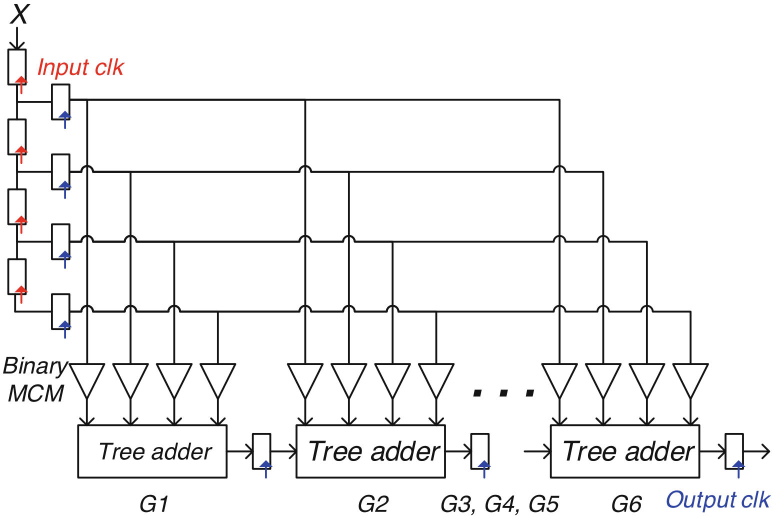


Fig. 24 MCM-B, one example of the FIR filter implantation. The intermediate results are stored by flip-flops in binary format. This enables timing closure for high speed

This implementation on FIR1 is shown in Fig. 25, in which a double-lined arrow represents a CS number. The CS number is converted back to binary number at the very end of the design by the VMA (vector-merging adder).

Compared to the binary architecture, the delay is smaller because of the elimination of CS to binary conversion. The CIC filter is thus implemented using two parts, i.e., the first computation circuit performing the carry-save addition and the second computation circuit implementing the vector-merging adder.

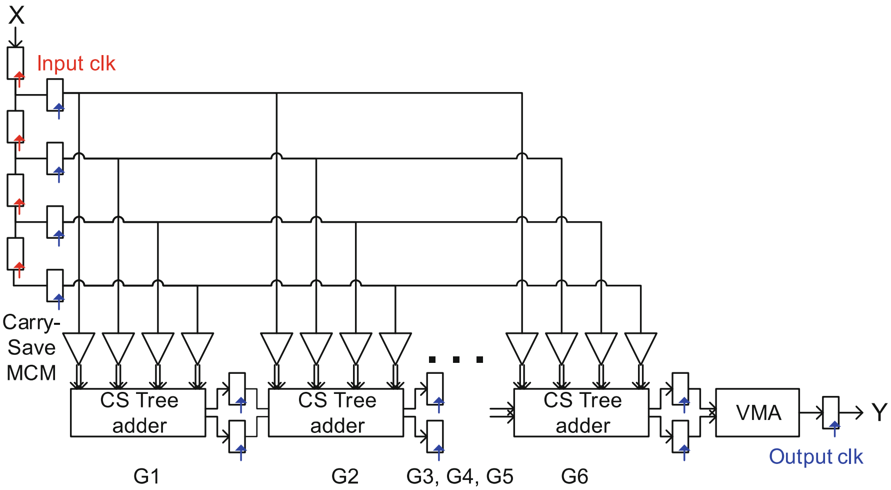


Fig. 25 MCM-CS, another example of the FIR filter implantation. The intermediate results are stored by flip-flops in CS (carry-save) format. The delay is further reduced

Modern synthesis tools are able to automatically replace binary additions in the MCM and the tree adder into CSA to reduce propagation delay. However, signals are still converted back to traditional binary format once they need right shifting or passed into pipeline registers. The conversion incurs a huge delay overhead.

3.2.2 Low-Frequency FIR Filters

For LF FIR filters, MACs are used to compute for many taps and to save area. Figure 26 shows this architecture using binary numbers and MACs (multiplier–accumulators). Therefore, this setup is called MAC-B. Because the sampling frequency is very low, the only MAC unit, controlled by the kernel clock (clk), is able to execute all the computation for each tap.

Replacing binary MAC with CS MAC eliminates the CS to binary conversion during multiplication, which decreases the delay. The FIR filter using CS-output MAC is called MAC-CS, which is shown in Fig. 27. The intermediate results are stored in the flip-flops as a carry-save (CS) format number. The CS number is finally converted back to a binary number by the VMA.

The shorter propagation delay property makes MAC-based designs possible for higher frequency filters, e.g., FIR2. This will reduce the area of the whole design, at the cost of power consumption increase.

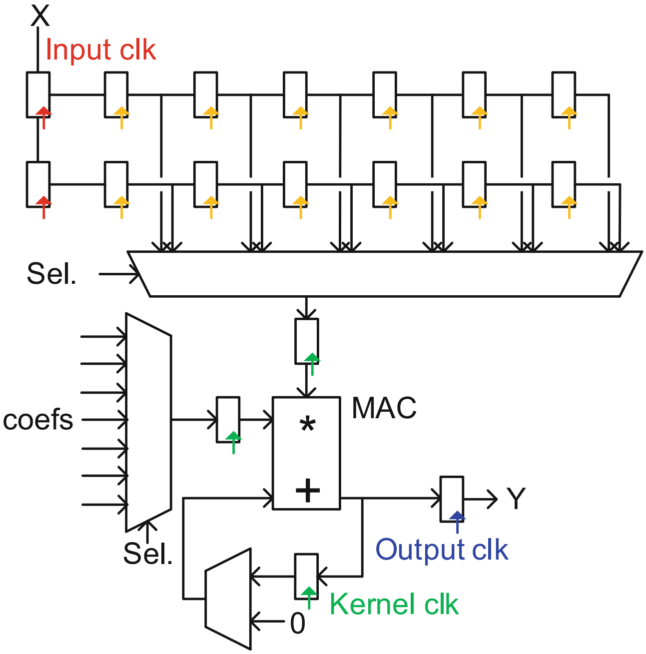


Fig. 26 FIR filter using binary MAC (MAC-B) for area saving

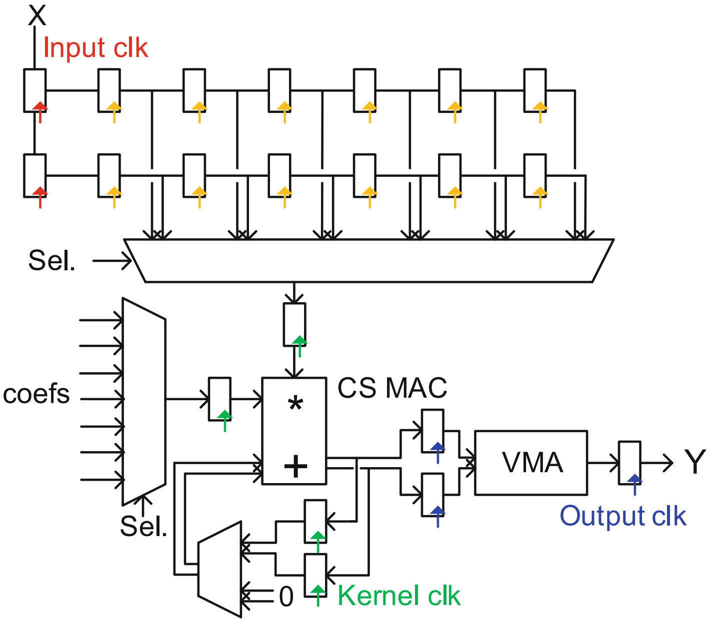
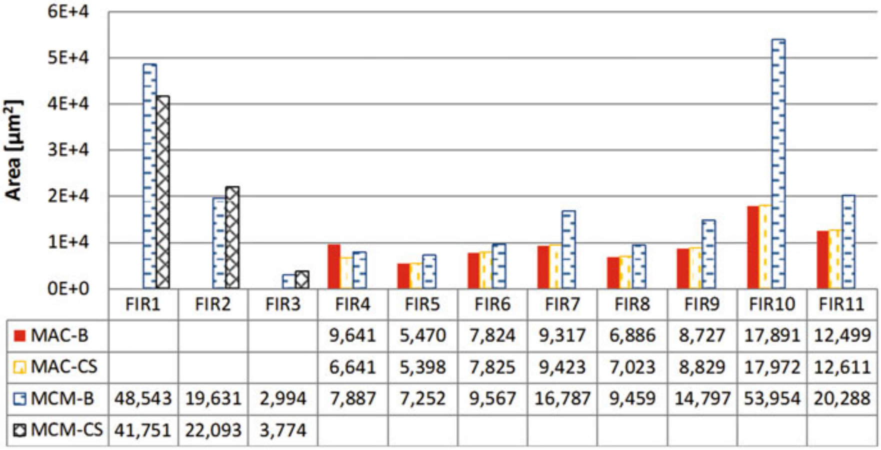


Fig. 27 FIR filter using CS MAC (MAC-CS). The advantage of the MAC-CS is reduced delay, with the cost of increase flip-flops to store the results



	FIR1	FIR2	FIR3	FIR4	FIR5	FIR6	FIR7	FIR8	FIR9	FIR10	FIR11
■ MAC-B				9,641	5,470	7,824	9,317	6,886	8,727	17,891	12,499
□ MAC-CS				6,641	5,398	7,825	9,423	7,023	8,829	17,972	12,611
□ MCM-B	48,543	19,631	2,994	7,887	7,252	9,567	16,787	9,459	14,797	53,954	20,288
☒ MCM-CS	41,751	22,093	3,774								

Fig. 28 Area comparison of the 11 FIR filters, with various FIR filter setup

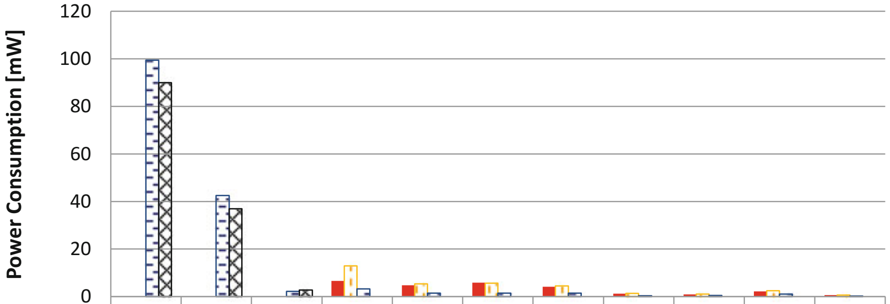
3.3 System-Level Selection of FIR Filter Architectures in the Multi-rate System

In this subsection, the selection of the four proposed FIR filter architectures is analyzed. The designs are first synthesized in 45-nm CMOS technology for quantitative analysis of the power and area trade-off of the design.

Two architectural options are possible for FIR1–FIR3. The reason is that for high-speed filters (FIR1–FIR3), the one (or several) MAC approach cannot meet the timing. For FIR4–FIR11, there are three options. This is because the MCM-CS approach is obviously too costly, compared with MCM-B, for low-speed designs.

Figure 28 shows the area of different FIR filters in the decimation chain. For FIR1, since the speed is as high as 1.024 GHz, MCM-B can only meet the speed requirements when three additional pipeline stages are inserted, which incurs large area and power consumption due to the inserted registers. Besides, gate upsizing is intensively used to reduce the signal propagation delay, which is also a reason for the large area cost. However, MCM-CS can meet the fast requirements without pipeline stages, which makes it smaller than MCM-CS approach. The situation is similar for other high-speed filters.

For lower-speed filters, where speed is not a concern, MCM-CS suffers from double registers, which makes the area larger. For very low-speed filters, MAC approach reduces the area by using just one or two fast MAC units. But the MAC approach cannot be applied to higher-speed filters. For FIR4, MAC-B requires two MAC units, which eventually makes its area larger than MCM-based designs. However, since MAC-CS is faster, it can realize FIR4 with just one MAC unit, which archives the lowest area.



	FIR1	FIR2	FIR3	FIR4	FIR5	FIR6	FIR7	FIR8	FIR9	FIR10	FIR11
🟥 MAC-B				6.65	4.85	5.88	4.16	1.25	0.99	2.29	0.64
🟨 MAC-CS				12.95	5.40	5.61	4.54	1.32	1.06	2.45	0.65
🟦 MCM-B	99.45	42.53	2.18	3.30	1.44	1.40	1.42	0.39	0.44	1.09	0.29
▩ MCM-CS	89.95	36.94	2.76								

Fig. 29 Power consumption comparison

The power consumption is summarized in Fig. 29. Because high-speed filters are running at a higher clock frequency, they consume most of the power. For FIR1, MCM-CS is more power efficient than MCM-B because of its lower area. For FIR2, although MCM-CS is larger than MCM-B, it consumes less power, because the toggling rate of the overhead from inserted pipeline stages is lower than that from the CS-to-binary conversion circuits.

MAC filters consume more power than MCM filters because the MAC and the connected registers are running at a much higher frequency than the sample rate. MAC-B is more power efficient than MAC-CS, simply because of fewer registers. For FIR4, the advantage is larger because MAC-B uses two MAC units that run at half the frequency compared to MAC-CS.

As each FIR filter can be realized by various architectures. The complete multi-rate decimation chain can be any combination of the options. The area, power consumption, and APP (area power product) are used for best area and power trade-off. The result is summarized in Table 2. As discussed before, all architectures should be considered numerically, for minimum power, area, or trade-off between those two. For the best power-area trade-off, the minimum APP is chosen as the final implementation for the decimation filter chain.

4 Conclusion

This chapter discusses designs of multi-rate systems. Two case studies are performed, with different focuses. The first case is the up-sampling filter for high data-rate wireless transmitters. The second case is the down-sampling filter for multi-standard wireless receivers.

Table 2 Lowest power, area, and APP designs for the multi-rate decimation filter chain

	Area [mm^2]	Power [mW]	Architecture selection of each FIR filter										
			FIR1	FIR2	FIR3	FIR4	FIR5	FIR6	FIR7	FIR8	FIR9	FIR10	FIR11
Min. power	0.21	139	MCM-CS		MCM-B								
Min. APP	0.15	144	MCM-CS		MCM-B				MAC-B				
Min. area	0.14	168	MCM-CS	MCM-B		MAC-CS		MAC-B					

For the high-bandwidth up-sampling filter, the multi-rate system usually works at a very high speed depending on the required oversampling factor. The design challenge on DSP front-end is analyzed and tackled. Firstly, system simulations with a complete transmission chain are conducted with regard to the error vector magnitude and the output spectrum, which allows to systematically optimize the design requirements on the DSP front-end. Secondly, algorithm and architecture co-optimization on the DSP front-end is explored to minimize the power consumption. The result shows the design is applicable for high-bandwidth usage and satisfies the speed requirement.

For the various frequency down-sampling filters, the data rate of the signal varies significantly. Therefore, different architectures are discussed. The advantages for each architecture are analyzed. In general, MCM-CS is the most suitable architecture for high-speed filters, e.g., running at 500 MHz or higher. For mid-speed filters, MCM-B is more attractive. MAC-B is the best choice for very low-speed (16 MHz or slower) FIR filters. If a minimum area is the objective, MCM-CS is useful for mid-frequency FIR filters (128 MHz). Note that although the analysis in this paper is dedicated for this decimation example, similar analysis can be generalized for other FIR applications.

In summary, the advance of digital CMOS scaling enables digital signal processing entering domains where it used to be believed as analog. In this scenario, careful system analysis and simulation, as well as various architecture implementation trade-off, should be carefully examined.

References

1. S. Association and others. (2012). IEEE standard for information technology—telecommunications and information exchange between systems—local and metropolitan area networks—specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 3: Enhancements for very high throughput in the 60 GHz Band. *IEEE*, 1–628.
2. Khalaf, K., et al. (2016). Digitally modulated CMOS polar transmitters for highly-efficient mm-wave wireless communication. *IEEE Journal of Solid-State Circuits, 99*, 1–14.
3. Nariman, M., Shirinfar, F., Pamarti, S., Rofougaran, M., Rofougaran, R., & De Flaviis, F. (2013). A compact millimeter-wave energy transmission system for wireless applications. In *Radio frequency integrated circuits symposium (RFIC), 2013 IEEE* (pp. 407–410). Seattle, WA: IEEE. doi:10.1109/RFIC.2013.6569617.
4. Chan, W. L., & Long, J. R. (2010). A 60-GHz band 2 2 phased-array transmitter in 65-nm CMOS. *IEEE Journal of Solid-State Circuits, 45*(12), 2682–2695.
5. Khalaf, K., Vidojkovic, V., Vaesen, K., Long, J. R., Van Thillo, W., & Wambacq, P. (2014). A digitally modulated 60GHz polar transmitter in 40nm CMOS. In *2014 IEEE. Radio Frequency Integrated Circuits Symposium* (pp. 159–162). Tampa, FL: IEEE. doi:10.1109/RFIC.2014.6851685.
6. Vidojkovic, V., et al. (2013). A low-power radio chipset in 40nm LP CMOS with beamforming for 60GHz high-data-rate wireless communication. In *International solid-state circuits conference digest of technical papers, 2013 IEEE* (pp. 236–237). San Francisco, CA: IEEE. doi:10.1109/ISSCC.2013.6487715.

7. Li, C., et al. (2016). Energy-efficient digital front-end processor for 60 GHz polar transmitter. *Journal of Signal Processing Systems, 1939*(8115), 1–13.
8. Volder, J. E. (1959). The CORDIC trigonometric computing technique. *Electronic Computers IRE Transactions, 3*, 330–334.
9. Li, C., et al. (2015). <30 mW rectangular-to-polar conversion processor in 802.11ad polar transmitter. In *2015 I.E. international conference on acoustics, speech and signal processing (ICASSP)* (pp. 1022–1026). Piscataway: IEEE.
10. Li, C., et al. (2015). Opportunities and challenges of digital signal processing in deeply technology-scaled transceivers. *Journal of Signal Processing Systems, 78*(1), 5–19.
11. Losada, R. A., & Lyons, R. (2006). Reducing CIC filter complexity. *IEEE Signal Processing Magazine, 23*(4), 124–126.
12. Aboushady, H., Dumonteix, Y., Louërat, M.-M., & Mehrez, H. (2000). Efficient polyphase decomposition of comb decimation filters in sigma delta analog-to-digital converters. *Circuits and Systems, 2000. Proceedings of the 43rd IEEE Midwest Symposium on, 2000, 1*, 432–435.
13. Hentschke, S., Herrfeld, A., Reifschneider, N,. Forster, D,. Heinemann, M., & Wicke, A. (1994). A flexible repetitive CSD code filter processor unit in CMOS. in *ASIC conference and exhibit, 1994. Proceedings seventh annual IEEE international ASIC conference and exhibit, Rochester, NY*, (pp. 261–264). doi: 10.1109/ASIC.1994.404562.
14. Noll, T. (1990). Carry-save arithmetic for high-speed digital signal processing. In *1990 IEEE international symposium on circuits and systems* (pp. 982–986). New Orleans: IEEE. doi:10.1109/ISCAS.1990.112267.
15. Koc, C. K., & Hung, C. Y. (1990). Multi-operand modulo addition using carry save adders. *Electronics Letters, 26*(6), 361–363.
16. Gustafsson, O., Dempster, A. G., & Wanhammar, L. (2004). Multiplier blocks using carry-save adders. *Circuits and Systems, 2004. ISCAS '04. Proceedings of the 2004 International Symposium on, 2004, 2*, I-473–I-476.
17. Huang, Y., Kapoor, A., Rutten, R., & Pineda de Gyvez, J. (2015). A 13bits 4.096GHz 45nm CMOS digital decimation filter chain with carry-save format numbers. *Microprocessors and Microsystems, 39*(8), 869–878.
18. Huang, Y., Kapoor, A., Rutten, R., & Pineda de Gyvez, J. (2013). *A 13 bits 4.096 GHz 45 nm CMOS digital decimation filter chain using Carry-Save format numbers* (pp. 1–4). Vilnius: 2013 NORCHIP. doi:10.1109/NORCHIP.2013.6702042.
19. Dadda, L. (1965). Some schemes for parallel multipliers. *Alta Frequency, 34*, 349–356.
20. Oklobdzija, V. G., Villeger, D., & Liu, S. S. (1996). A method for speed optimized partial product reduction and generation of fast parallel multipliers using an algorithmic approach. *IEEE Transactions on Computers, 45*(3), 294–306.
21. Huang, Y., Li, M., Li, C., Debacker, P., & Van der Perre, L. (2014). Computation-skip error resilient scheme for recursive CORDIC. In *2014 IEEE workshop on signal processing systems (SiPS)* (pp. 1–6). Belfast: IEEE. doi:10.1109/SiPS.2014.6986061.
22. Bi, Z., & Dai, Y. (2012). Full custom data path of 16-bit CORDIC. In *2012 IEEE fifth international conference on advanced computational intelligence (ICACI)* (pp. 993–998). Nanjing: IEEE. doi:10.1109/ICACI.2012.6463320.
23. Kwak, J. H., Piuri, V., & Swartzlander, E. E. (2000). Fault-tolerant high-performance CORDIC processors. In *Proceedings IEEE international symposium on defect and fault tolerance in VLSI systems* (pp. 164–172). Yamanashi: IEEE. doi:10.1109/DFTVS.2000.887154.
24. Umemoto, Y., et al. (2014). 28 nm 50% power-reducing contacted mask read only memory macro with 0.72-ns read access time using 2T pair bit cell and dynamic column source bias control technique. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 22*(3), 575–584.
25. Lee, B., & Burgess, N. (2003). Some results on Taylor-series function approximation on FPGA. *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on, 2003, 2*, 2198–2202.

26. Huang, Y., et al. (2016). A 28 nm CMOS 7.04 Gsps polar digital front-end processor for 60 GHz transmitter. In *2016 IEEE Asian solid-state circuits conference (A-SSCC)* (pp. 333–336). Toyama: IEEE. doi:10.1109/ASSCC.2016.7844203.
27. Vaidyanathan, P. P. (1990). Multirate digital filters, filter banks, polyphase networks, and applications: A tutorial. *Proceedings of the IEEE, 78*(1), 56–93.
28. Coffey, M. W. (2003). Optimizing multistage decimation and interpolation processing. *IEEE Signal Processing Letters, 10*(4), 107–110.
29. van den Enden, A. W. M. (2001). *Efficiency in multirate and complex digital signal processing*. Amerongen: Delta Press.

Advances in Multirate Filter Banks: A Research Survey

A. Kumar, B. Kuldeep, I. Sharma, G.K. Singh, and H.N. Lee

1 General Overview

Multirate filter banks (FBs) play a substantial role in numerous signal processing applications such as data compression, detection of harmonics, de-noising, subband decomposition, recognition of one and two dimensional (2-D) signals, adaptive filtering, design of wavelet bases, and wireless communication [1–12]. The specific idea of developing multirate systems is their ability to split original input signal into multiple signals or to combine multiple signals into a single composite signal in the frequency domain. Division of a signal into number of subband is also known as subband coding. Originally, the concept of subband coding was introduced to minimize the effect of quantization noise in speech coding. Later on, with the advancement in multirate signal processing, subband coding concept has been extensively used in several applications. Performance of a subband coding system or a multirate system in different applications relies on the optimal design methodologies that have used for a multirate system. Therefore, there is always a strong motivation to develop an efficient technique for designing a multirate system that can improve the performance for given specific applications. Basically, the

---

A. Kumar (✉) • I. Sharma
PDPM Indian Institute of Information Technology Design and Manufacturing,
Jabalpur, MP 482005, India
e-mail: anilkdee@gmail.com

B. Kuldeep
National institute of Technology, Hamirpur, HP 177005, India

G.K. Singh
Indian Institute of Technology Roorkee, Uttrakhand 247667, India

H.N. Lee
School of Electrical Engineering and Computer Science, Gwangju Institute of Science
and Technology, 123 Cheomdan-gwagiro, Buk-gu, Gwangju 61005, South Korea

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9_2

multirate system consists of a linear time-invariant (LTI) system and subsampling operations such as down-sampling or up-sampling. In general, multirate systems are classified in two types on the basis of their mode of operations. The first type corresponds to a filter bank structure in which original input signal is subdivided into different bands or channels, depending upon the requirement of given application, while the second type corresponds to a transmultiplexer (TMUX) structure, where several signals are combined and transferred through same channel in communication system. A generalized block diagram of these structures is graphically shown in Fig. 1. In Fig. 1a, the input signal is decomposed into different subbands using a bank of filters; here an input signal is analyzed, so these filters are also known as analysis filters, and at the receiver side, these subbands are synthesized into a reconstructed signal, so they are known as synthesis filters. Due to this operation, this structure is also known as analysis/synthesis multirate system. In Fig. 1b, several input signals are filtered by a set of filters, also termed as synthesis filters and combined into a composite signal, which is again filtered into several signals at receiver side by a bank of filters, also called as analysis filters.

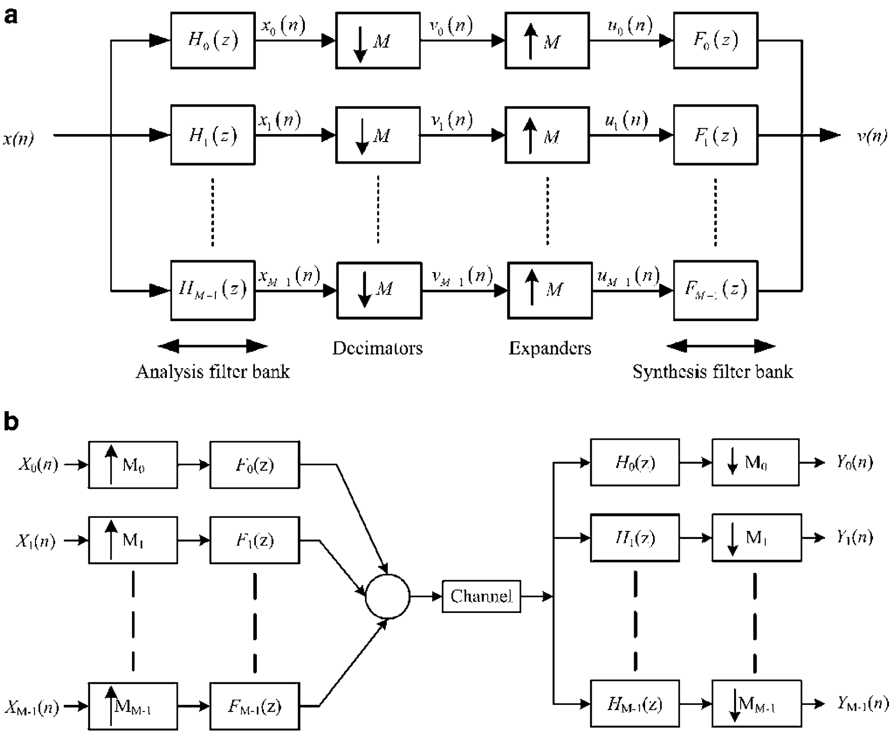


Fig. 1 (**a**) A block diagram of a filter bank [14, 15]. (**b**) A generalized block diagram of a transmultiplexer system [1, 2]

Therefore, this structure is also known as synthesis/analysis or transmultiplexer system. In absence of channel coding, a multirate system should be a pure delay system, it should not introduce any distortion to original input signal, and this type of a system is known as perfectly reconstructed (PR) multirate system [13–16-]. However, due to subsampling operations and non-ideal nature of LTI systems, multirate systems suffer from three different types of distortions during reconstruction of a signal. The first is aliasing distortion, which occurs due to sub-sampling operation; the next two distortions are amplitude and phase distortion, which occurs due to imperfect frequency response of the composing filters. A multirate system that suffers from these distortions is called nearly perfect reconstructed (NPR) filter bank. Therefore many researchers and scientists have proposed numerous methods to reduce or eliminate these distortions in multirate systems [14, 15].

Initially, the subband coding systems, introduced by Crochiere et al. [13, 17], were influenced by all three types of distortions, and cancellation of aliasing distortion in this multirate system was not completely achieved. Therefore, to remove aliasing distortion in a multirate system, a concept of quadrature mirror filter (QMF) bank was introduced by Croisier et al. [18]. In this context, firstly Esteban and Galand [19] have employed a two-channel filter bank structure in the voice-coding application and the aliasing distortion was reduced with the use of QMF [18, 19]. The phase distortion was eliminated by using linear-phase finite impulse response (FIR) filters. However, the amplitude distortion cannot be eliminated except for trivial case [14–16, 20]. It can be abated by using computer-aided techniques or proper optimization techniques. The amplitude distortion can be thoroughly removed by employing infinite impulse response (IIR) [14–16, 20–24]. But in that case, the amplitude distortion is eliminated at the cost of introducing phase distortion. Later on, Smith and Barnwell [25] and Mintzer [26] were able to design PR FB. Since then numerous techniques have been proposed and described for the design of PR multirate systems [14–16, 27–31], but for achieving the PR property, some important properties of analysis/synthesis filters have to be relinquished such as low stopband attenuation, narrow transition bands, and linear phase.

It is revealed from above discussion that the QMF banks are extensively used in a number of applications. However, in certain applications such as videos, communication systems, etc., the linear-phase QMF banks are strictly needed to avoid signal distortion and subband coding expansion problem [6, 11, 32]. Initially, the effort was concentrated on the efficient design of two-channel FBs, and later on, it was prolonged to multichannel FBs [14, 15, 20]. Among all multichannel FBs, cosine-modulated filter banks (CMFBs) are one of the most often used filter bank due to their simpler and more realizable design structure. As similar to QMF banks, several competent techniques have also been projected to design CMFBs efficiently [14, 15, 33–37], as the filter and FBs have become the most fundamental systems used in various applications among multirate systems.

2 Digital Filter Bank

The digital filter can be defined as a frequency selective network that picks and adjusts certain interlude of frequencies comparative to other frequencies, or it attenuates all the frequency content outside the anticipated interval [38, 39]. -Fundamentally, filtering operation is accomplished to enhance the quality of a signal. The characteristic of filtering is determined by the frequency response of filter. The frequency response of filter is computed by the system parameters or coefficients. Thus, by suitable assortment of system coefficients, appropriate frequency-selective network can be designed according to given applications [38]. Two classical types of digital filters used are finite impulse response (FIR) and infinite impulse response (IIR) filters, between which FIR digital filters are widely used as component of digital signal processing systems due to their ease of execution, stability, and linear design [16].

The digital FB is amalgamation of different band pass filters with a shared input and a summed output, which are mostly employed for synthesis and examination of different spectrums of signal. Therefore, FBs are grouping of analysis FBs and synthesis FBs as depicted in [14–16].

Analysis FB consists of sub-filters, which are known as analysis filters. Analysis filters are used to divide the input signal into dissimilar set of subband in frequency domain. Each subband comprises some frequency share of original signal. Similarly, the synthesis FB comprehends sub-filters called synthesis filters, which combine the subband signals and generate signal or reconstruct signal [38]. These FBs can be classified in two types; two-channel and *M*-channel FBs, based on the number of channels used for the signal separation.

2.1 Two-Channel Filter Bank

The two-channel FB or QMF bank consists of an analysis filters followed by down-samplers at transmission end, and up-samplers followed by synthesis filters at the receiving end. The block diagram of QMF bank is depicted in Fig. 2. This arrangement is termed as critically sampled FB, where decimation factor and interpolation factor are equal to the number of subbands. On the performance basis of reconstructed output, two-channel FBs can be further classified into nearly perfect reconstruction (NPR) and perfect reconstruction (PR) filter banks. The NPR two-channel FB is called conventional QMF banks, where the reconstructed output and input signal is not exact replica to each other. Usually, in NPR two-channel FB, the analysis and synthesis filters are in linear phase. In PR FBs, all the three types of distortions can be eliminated. These FBs have some delay, but the signal is an exact replica of original signal [38].

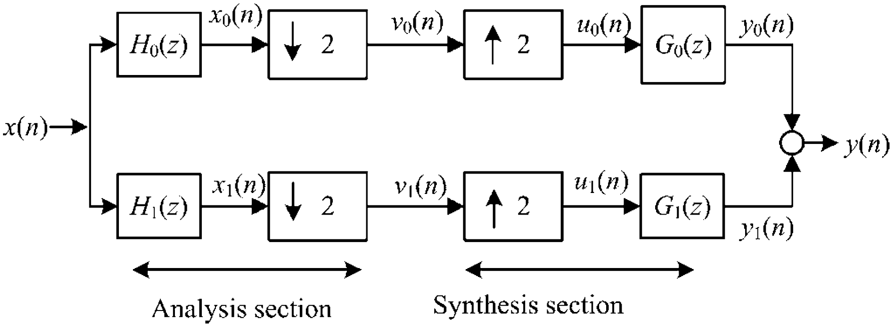


Fig. 2 A generalized block diagram of a two-channel filter bank [14, 15]

2.1.1 Analysis of Two-Channel Filter Bank

In general, the two-channel FB splits an input signal into two subbands and is made up of an analysis and synthesis filter as well as a processing unit in between the FBs. Hence, the elementary process of a two-channel FB involves two sequential steps in the absence of processing unit: operation of the analysis banks and synthesis banks. A general block diagram of two-channel FB is shown in Fig. 2. [14, 15, 38].

From the analysis of input and output relation, the output of two-channel FB is written as [14–16]

$$Y(z) = T(z)X(z) + A(z)X(-z), \tag{1}$$

where $T(z)$ is the overall distortion transfer function of FB, expressed as

$$T(z) = \frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)], \tag{2}$$

and $A(z)$ is the aliasing transfer function, defined as [14, 15]

$$A(z) = \frac{1}{2}[H_0(-z)G_0(z) + H_1(-z)G_1(z)]. \tag{3}$$

To cancel aliasing distortion, the second term of Eq. (1) must be cancelled, that is, $A(z) = 0$. Thus, the following choice cancels the aliasing error [14–16, 20]:

$$G_0(z) = H_1(-z) \text{ and } G_1(z) = -H_0(-z). \tag{4}$$

It is thus possible to completely cancel the aliasing distortion by this choice of synthesis filters. Therefore, using above relations, the input–output relation of the FB expressed by Eq. (1) can be further simplified as

$$Y(z) = T(z)X(z). \tag{5}$$

If $H_0(z)$ is a casual real coefficient FIR filter and $H_1(z) = H_0(-z)$ [14, 15], it conforms that $H_1(z)$ is a noble high-pass filter if $H_0(z)$ is a good low-pass filter and vice versa. By carefully examining these responses, it is perceived that one filter is the complementary or mirror of other filter at $\omega = \pi/2$, which is the cutoff frequency of both high-pass and low-pass responses and justifying the name Quadrature mirror filter (QMF) bank. Thus, by using $H_1(z) = H_0(-z)$ condition, the aliasing-cancellation condition given by Eq. (4) can be modified as

$$G_0(z) = H_0(z) \text{ and } G_0(z) = -H_0(-z). \tag{6}$$

Consequently, the above equations suggest that the analysis and synthesis filters of QMF bank can be generated by proper design of $H_0(z)$ called prototype FB. By using Eq. (6) and mirror-image condition $H_1(z) = H_0(-z)$, the distortion function given by Eq. (2) can be further rewritten as

$$T(z) = \frac{1}{2}\left[H_0^2(z) - H_1^2(z)\right] = \frac{1}{2}\left[H_0^2(z) - H_0^2(-z)\right]. \tag{7}$$

Consider a casual FIR filter $H_0(z)$ defined as

$$H_0(z) = \sum_{n=0}^{N-1} h_0(n)\, z^{-n}, \tag{8}$$

where, N is the order of filter. Given that $h_0(n) \neq 0$ and $h_0(N) \neq 0$, $H_0(z)$ has linear-phase characteristics if

$$h_0(n) = h_0(N - 1 - n). \tag{9}$$

For a linear-phase FIR filter, the frequency response $H_0(e^{j\omega})$ can be written as

$$H_0\left(e^{j\omega}\right) = \left|H_0\left(e^{j\omega}\right)\right| e^{-j\frac{N-1}{2}\omega}, \tag{10}$$

where $|H_0(e^{j\omega})| = H_0(\omega)$. Substituting (10) into (7) and using the fact that $|H_0(e^{j\omega})|$ is an even function, the FB transfer function reduces to

$$T\left(e^{j\omega}\right) = \frac{e^{-j(N-1)\omega}}{2}\left(\left|H_0\left(e^{j\omega}\right)\right|^2 - (-1)^{(N-1)}\left|H_0\left(e^{j(\pi-\omega)}\right)\right|^2\right). \tag{11}$$

If $N - 1$ is even, then above expression reduces to zero at $\omega/2$, resulting in severe amplitude distortion. Hence, even order filters cannot be used, since they allow the existence of nulls in overall system response. If the N is to be taken as odd, then, FB transfer function is reduced to

$$T\left(e^{j\omega}\right) = \frac{e^{-j(N-1)\omega}}{2}\left(\left|H_0\left(e^{j\omega}\right)\right|^2 + \left|H_0\left(e^{j(\pi-\omega)}\right)\right|^2\right). \tag{12}$$

For perfect reconstruction, a FB must have

$$\left|H_0\left(e^{j\omega}\right)\right|^2 + \left|H_0\left(e^{j(\pi-\omega)}\right)\right|^2 = 1. \tag{13}$$

But, this is not exactly possible because of amplitude distortion, which is defined as

$$e(\omega) = 1 - \left|H_0\left(e^{j\omega}\right)\right|^2 + \left|H_0\left(e^{j(\pi-\omega)}\right)\right|^2 \tag{14}$$

In linear-phase QMF bank design, the minimization of amplitude distortion will be a major consideration. Regarding this issue, several techniques for designing a linear-phase QMF bank have been reported in literature [14, 15, 20].

2.2 M-Channel Filter Bank

M-channel FB divides the signal into M number of subbands. On the basis of nature of channel bandwidth, M-channel FBs are categorized into two ways: uniform and nonuniform M-channel FBs [14, 15, 20, 38]. Uniform M-channel FBs are subgrouped into three different types: tree structure, independent M-channel (parallel FB), and modulation FBs [14, 15, 20, 38]. As compared to two-channel FB, the conditions for alias-cancellation and perfect reconstruction are much more complicated. The general theory of M-channel FB was developed by a number of researchers [14, 15, 20]. The M-channel filter banks are further classified as uniform and nonuniform filter bank on the basis of channel bandwidth, and so as the subsampling factors. When these sub-sampling factors are kept same throughout the design procedure, the M-channel filter bank is known as uniform as shown in Fig. 1a, and when these sub-sampling factors are changed, the M-channel filter bank is known as nonuniform filter bank as shown in Fig. 3. [14, 15, 20, 38]. It can be observed that from Fig. 3, a two-channel filter bank has been used as a basic building block for designing NUFB. With meticulous governance of two-channel filter banks, they are designed to be alias free, so as to eventually make NUFB alias free.

This section concentrates mainly on cosine modulation technique of M-channel FB design, because cosine modulated (CM) filter banks have become very popular in many applications of signal processing due to use of easy prototype filter design and fast discrete cosines transform (DCT).

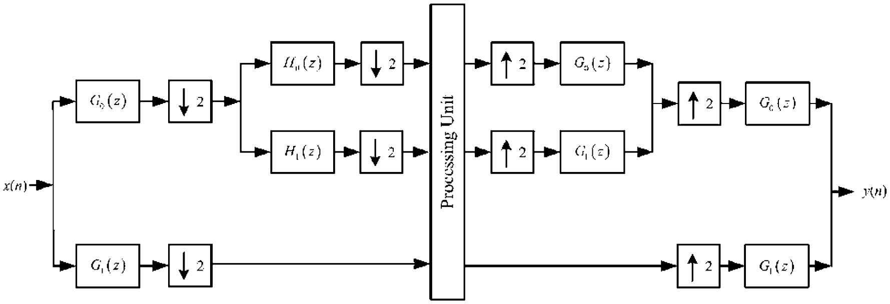


Fig. 3 Block diagram of tree structural NUFB [15–17]

2.2.1 Analysis of M-Channel CM Filter Bank

During the past, theory of multirate systems has been well developed and available in several research books [14, 15]. The generalized system structure of M-channel CMFB [14, 15, 20, 38], is graphically shown in Fig. 1a. In this architecture, ($h_k(n)$ and$H_k(z)$) and synthesis filters ($f_k(n)$ and $F_k(z)$) are derived from cosine modulation. Here, $h_k(n)$ and $H_k(z)$ are the impulse responses, and z-transforms of the analysis filters, while $f_k(n)$ and $F_k(z)$ represent the impulse responses, and z-transforms of the synthesis filters. In Fig. 1, M represents the number of channels in a multirate filter bank. In general, the basic components of a multirate system are either it is a filter bank (FB) or transmultiplexer (TMUX) are decimators, expander, and filters. Based on input/output relations of a decimator and expander, the reconstructed output of a considered structure in Fig. 1 [14, 15, 20, 38]:

$$Y(z) = T_0(z)X(z) + \sum_{l=1}^{M-1} T_l(z)X\left(ze^{-j2\pi l/M}\right), \tag{15}$$

where, $X(z)$ and $Y(z)$ is the input and output signals, respectively,

$$T_0(z) = \frac{1}{M}\sum_{k}^{M-1} F_k(z)H_k(z), \tag{16}$$

and

$$T_l(z) = \frac{1}{M}\sum_{k}^{M-1} F_k(z)H_k\left(ze^{-j2\pi l/M}\right) \text{ for } l = 1, 2, \ldots, M-1. \tag{17}$$

In Eq. (15), $T_0(z)$ stands for distortion transfer function, which decides the distortion, resulted from overall filter bank structure for un-aliased component of input signal ($X(z)$). Second term $T_l(z)$ decides aliasing error [14, 15, 20]. For perfect reconstruction, $T_0(z)$ is to be a delay function (z^{-K}), where K is an integer and aliasing error should be zero. If a filter bank structure satisfies these conditions, then it is known as perfect reconstructed (PR) cosine-modulated filter banks (CMFBs), in which the reconstructed output signal is an exact replica of original input signal. This structure is highly suitable for data compression application in lossless coding. If these conditions are partially satisfied, then it is known as nearly perfect reconstructed (NPR) CMFB with amplitude distortion and aliasing error [14, 15, 20]. In a CMFB, impulse responses of the analysis filters ($h_k(n)$) and synthesis filters ($f_k(n)$) are cosine-modulated version of a prototype filter $h_0(n)$ with following transfer function given by Eq. (18). The remaining analysis and synthesis filters are determined by

$$h_k(n) = 2h_0(n)\cos\left[\omega_k\left(n - \frac{N-1}{2}\right) + \theta_k\right], \tag{18}$$

$$f_k(n) = h_k(N-1-n) \tag{19}$$

with $k = 0, 1, \ldots, M-1$, and

$$\omega_k = \frac{(2k+1)\pi}{2M}, \tag{20}$$

where

$$\theta_k = (-1)^k \pi/4. \tag{21}$$

As discussed above, in a multirate filter bank, three types of distortions known as phase distortion, amplitude distortion, and aliasing distortion are encountered. Generally, the phase distortion is completely eliminated by using a finite impulse response (FIR) filter due to their exact linear-phase response characteristic. In NPR CMFB, distortion in amplitude response is computed as [14, 15, 20]

$$e_{\mathrm{am}} = \max_{\omega}\left(1 - \left|T_0\left(e^{j\omega}\right)\right|\right), \tag{22}$$

and aliasing distortion (e_{a}) is computed as

$$e_{\mathrm{a}} = \max_{l,\omega}\left(T_l\left(e^{j\omega}\right)\right) \text{ for } \omega \in [0, \pi], 1 \le l \le M-1. \tag{23}$$

3 Literature Review

In this section, a comprehensive review on multirate filter banks, M-channel filter banks, and two-channel filter banks, is presented.

3.1 Review on Multirate Filter Bank

In past few decades, an extensive research has been carried out in the area of multirate FB design, because of its significant importance in the field of signal processing. On the basis of number of channels employed for subband decomposing of a signal in frequency domain, the research and development of multirate FB belongs to two-channel and M-channel FB [40–105]. On the basis of reconstruction of signal, the investigation of multirate FB is further divided into NPR and PR FB design. Moreover, research in NPR FB is classified as linear and nonlinear phase FB on the basis of phase of compound filters. In linear-phase two-channel multirate FB or QMF banks, aliasing and phase distortions are removed by deploying appropriate design of the composite filters and linear-phase FIR filter, respectively, whereas an amplitude distortion can be minimized using computer-aided techniques or optimization techniques [14–16, 20, 40–42]. The exact reconstruction of signal with good frequency resolution is pretty difficult due to various distortions occuring in multirate FB.

In QMF bank, due to quadrature relationship between the analysis and synthesis filters, the design problem merely reduces to solitary prototype filter design [14–16, 20, 40–42]. In this context, there are numerous methods have been reported for designating prototype filter of two-channel multirate FB [43–50]. These design methods can be further categorized into optimization and non-optimization based methods. The design problem using optimization techniques is framed as nonlinear single or multi-objective functions, which is solved by numerous optimization techniques such as classical iterative, non-iterative [38, 47, 51–82] methods, and nature-inspired optimization techniques [99–105].

In 1980, the first symmetric iterative method for the design of two-channel FB was introduced by Johnston [47]. In this method, the objective function, which is framed as weighted sum of the prototype filter with stopband error energy and ripple energy, is optimized through Hooke and Jeaves optimization algorithm [47]. This method does not give optimum solution and needs manual initialization. Further, Jain and Crochiere [51] proposed a novel design approach with same objective function. In [51], manual initialization has been eliminated and solution becomes more optimal than that in [47], but it is well-suited to higher-order filter bank due to high degree of nonlinearity. However, in both these algorithms, reconstruction error is not Equiripple. Therefore, Chen and Lee [52] have introduced an iterative technique for the optimization of objective function in frequency domain that affects the Equiripple reconstruction error. Further, there are numerous algorithms introduced by the researchers based on [53–56]. These are superior from the algorithm given in [52] in term of peak reconstruction error and computational time, but all these algorithms are not acceptable for higher-order FB design. After that, Bregovic and Saramaki [57, 58] have developed a two-step method for designing two-channel FBs for those applications where Equiripple reconstruction error and less stopband energy are required.

It was found that the design problem, which is constructed in minimax logic, needs more efficient optimization techniques. Thus, the weighted least squares optimization technique is more appropriate than other conventional techniques, because of its smaller code and delivery of solution analytically. Therefore, several researchers used weighted least squares method in the field of FB design [47, 50–56, 59–64]. Fundamentally, the weighted minimax design is a nonlinear constrained optimization problem, which is complex to solve. In [54], weighted minimax problem is resolved by means of unconstrained weighted least squares (WLS) approach using two iterative methods. This method has been later on modified [53, 55–57, 59–61] in terms of computation time and reconstruction error. After that, WLS techniques have been successfully used by various researches with the modification in objective function, different parameters and constraints, or use of different modeling [62–65].

In this context, many classical iterative and WLS techniques have been introduced for the efficient design of QMF bank using constraints or unconstraint formulation. In [66], the researchers have used Lagrange multipliers method to optimize the nonlinear constrained problem. In [67–70], the unconstrained optimization methods have been used to optimize the objective functions, which have been framed as linear or nonlinear combination of reconstruction error, stopband residual energy or passband energy. Unfortunately, all the above discussed techniques were not much well organized in terms of peak reconstruction error (*PRE*) and computational complexity.

Recently, several efficient classical techniques have been explored for efficient NPR filter bank design. In the early stage of research, several gradient-based techniques such as Levenberg–Marquardt (LM) algorithm and quasi-Newton method [71–74] were developed to enhance the *PRE* and computational efficiency. A new methodology for the design of a two-channel FB by deploying Marquardt optimization technique is given in [71]. These design methodology was further enhanced by using quasi-Newton (QN) [75] and Levenberg–Marquardt (LM) method [76] and again modified as a hybrid technique based on LM and QN optimization techniques in [77]. As similar to linear-phase NPR QMF design, there are numerous classical techniques also reported in the stream of nonlinear phase and PR QMF bank [66, 75–81]. It is clear that the PR condition can be attained by surrendering the linear phase or some other properties of FB. Since this chapter quintessence on linear-phase NPR QMF design, therefore, here no detail on PR QMF bank is included. The detail on nonlinear and PR QMF bank is explained in [14–16, 20, 38, 82].

In this regard, primarily, the research effort was focused on two-channel FB, and thereafter, it was extended to multichannel FBs, particularly modulation-based multichannel FB, because of easy design and less complexity. Numerous modulation techniques are reported in literature such as modulated lapped transform, cosine modulation, a modified discrete Fourier transform (MDFT) technique, etc. [14, 15, 20, 83]. Cosine modulation technique is more popular in various requirements among different class of modulation techniques. Cosine-modulated FB is also categorized into PR and NPR cosine-modulated (CM) filter bank [14, 15, 20]. Initially, the concept of modulated FBs was presented by Nussbaumer [84–107]. The first NPR M-channel critically sampled FB employing cosine modulation on prototype filter was produced by Rottweiler [85]. In NPR CMFBs, PR state is relaxed to reduce the design complexity. After that, FBs with such property have been studied in detailed, and various approaches have been developed [86–105]. Several efficient methods have also been advanced, so that it can facilitate the efficient design of prototype filters for NPR CMFBs. In conventional design approaches [33, 34, 83], in overall, a relationship has to be made amongst the $2M$th band filter, and a group of quadratic constraints in impulse response coefficients of the prototype filter. These constraints and stopband attenuation of the prototype filter have to be reduced as much as possible using either constrained or unconstrained optimization techniques, which is quite time consuming. As a consequence of continued development in this field, several new procedures that have simpler minimizing cost functions have been established. To decrease the computational complexity in conservative designs [33, 34, 83], Creusere and Mitra [86] have presented the first systematic linear search optimization method for designing a prototype filter for CMFB, by varying passband frequency (ω_p) in each of iteration. This method yields improved performance of filter bank in terms of reconstruction. However, it also suffers from slow convergence. Therefore, these algorithms are not suitable for filter with larger taps. Thus, in search of efficiency and simplicity, a new efficient technique has been introduced by Lim and Vaidyanathan [87], in which Kaiser window function has been deployed for designing the prototype filters. This technique was further modified to include other windows [88, 89]. Furthermore, other efficient methods using different window techniques have been introduced [90–96] with improvement in [86–89]. In general, the FBs designed with windowing result in reduced stopband error, while passband ripple should be kept approximately equal to stopband ripple [97]. Therefore, a new method based on the weighted constraint least square (CLS) technique is devised for designing CMFB for given channel overlapping, and it was further improved in [98, 99]. Similarly, many other researchers [100–105] have also developed various methods for cosine-modulated FBs with NPR design. Similarly, various methods have also been proposed for M-channel PR FB design [14, 15, 36, 106]. These FBs are used in various applications such as lossless coding. Here, the detailed discussion on PR FB design is not given, because this chapter emphasizes on NPR M-channel CMFB design. The details of PR FB design can be found in [14, 15, 36, 106]. It is reflected from the above review that there are numerous conventional techniques have been proposed for two-channel and M-channel FB design. The general disadvantage of all these conventional techniques is that they may get stuck in local minima. To overcome this problem, recently various global optimization techniques are employed [107–113] in the field of FB design. Most of the global optimization techniques are driven by the natural phenomena's such as social behavior: flock of birds, schooling of fish, intelligence of swarm of bee; and biological process: reproduction, mutation, and interaction. Therefore, these algorithms are also termed as nature-inspired (NIO) or swarm optimization techniques (SOTs). Thus, many global optimization techniques are employed to design highly optimized FB such as

GA [113, 114], PSO and its different variants [115–118], ABC algorithm [116], and ADE algorithm [119]. In this context, a CORDIC genetic algorithm was used for a two-channel filter bank design in [107], which is based on coordinate rotation. A new design technique using nonlinear unconstraint optimization was presented for QMF banks based on PSO optimization and was further modified in [110]. Here, a new hybrid technique based on PSO and ABC algorithms was presented for a two-channel QMF bank design. This design was further improved by using ADE algorithm in [113]. Similarly, for M-channel FB design; a new technique based on genetic algorithm (GA) was devised by formulating a new cost function in [108], which is based on the weighed sum of aliasing error and amplitude distortion. Whereas, a novel design method was also presented for CMFB [111]. This design is based on Lagrange polynomial approximation and PSO, by considering stopband attenuation as a cost function, which was further modified using new variant of PSO [112], where quantum mechanics involved on particles of PSO is exploited and the reconstruction condition was formulated as a new cost function or objective function for optimization.

In recent years, several researchers have proposed the new design methodologies for optimal designs of digital filters and filter bank which are computationally efficient and require least hardware resources [115–119]. Here the continuous filter coefficients are quantized and converted into binary form (two's complement, maximal signed digit, canonical signed digit (CSD), etc.). These filter bank are also known as multiplierless multirate filter bank as all of the multipliers are realized in terms of adders and shifter. As the multipliers are the most resource-consuming component in hardwired realization, the removal of multipliers from the circuits reduces it overall resources cost. However, when these continuous filter coefficients are rounded in terms of binary or quantized CSD valued coefficients form, the performance of the filter is deteriorated due to quantization and conversion process, and, thus, the given specifications are no longer satisfied. Therefore, to overcome these problems, many global optimization techniques have been exploited to optimize the performance. In this context, authors in [120] have used genetic algorithm (GA) for designing digital filter with optimal performances. Since then, several designs for digital filter and filter banks have been proposed based on GA [121–123]. Recently, evolutionary algorithms (EA) such artificial bee colony (ABC), differential evolution (DE), harmony search algorithm (HSA), gravitational search algorithm (GSA), CSA and PSO have been developed and have become more promising optimization techniques in problem solving due to their multi dimension and multivariable steps for achieving a global solution [124, 125]. Therefore, researchers have exploited these techniques for designing optimal digital filter banks [123, 126, 127]. Multiplierless reconfigurable multichannel filters using metaheuristic algorithms have been proposed, where DE, HAS, gravitational search algorithm, and artificial bee colony algorithm were used to design the multiplierless reconfigurable nonuniform channel filters [93, 94, 128, 129]. Then, modified metaheuristic algorithms such as modified ABC algorithm, GA, and GSA have been employed to design frequency response masking (FRM)-based multiplierless uniform and non-uniform filter bank [132, 133], while authors in [130, 131] have used linear search technique for designing filter bank and represented the filter coefficients in CSD form and DE for digital filter, respectively.

From the literature review, it is quite evident that marked progress has been made in the field of FB design, and various conventional, nature-inspired, constraint optimization is incorporated in the field of FB design. Some methods are not good in computational time, and some do not give less distortion. Moreover, the new research trend in the field of multiplierless multirate filter banks has attracted many researchers to develop an efficient and resource-optimized designs for designing optimal multirate system. Therefore, there is still need to develop an efficient method, which can minimize the computational time, reconstruction error, and aliasing distortion and utilize least resources simultaneously.

4 Research Gap and Future Direction

The above discussion reveals that there are various literatures available in for efficient design of multirate filter bank. For the recent advancement in multirate system, some latest research aspects are discussed in this section, which are based on simultaneous utilization of nature inspire optimization techniques, fractional derivative, polyphase components, minor component analysis, multiplierless realization of filter, and filter bank. Recently, conventional integer derivative constraints have been used to design various FIR filters to improve design accuracy at the prescribed frequency point. However, a fractional calculus has shown improved performance in many engineering applications such as electrical networks, electromagnetic theory, biomedical applications, signal and image processing. The fractional derivative increases the possibility of improving control performance by reducing the convergence time in mentioned control problems. Therefore, a new technique for designing linear-phase FIR filters, based on fractional derivative constraints (FDCs), was devised in 2012. It gives smoother passband and stopband region than conventional least square and conventional derivative methods. But, it was not so efficient. The literature, available so far on fractional derivative, reveal that there is still need for an efficient technique, which shall use FDCs and evolutionary optimization technique for designing different types of linear-phase FIR filters and multirate FBs for higher order. Literature review also reveals that there is no method available for optimizing all the FDCs simultaneously. Fractional derivative has also been not employed in the field of multirate filter bank design.

The important advancement in signal processing is the polyphase representation of digital filter which reduces the computational complexity and data storage. But according to current literature, there is no method presented that has been used for optimized design of FIR filter and multirate filter bank, where polyphase components (PCs), FDCs, and swarm intelligent optimization algorithms are employed simultaneously.

As seen from the literature, nowadays, neural network plays a major role in several signal processing applications such as antenna, constraint beam forming, and biomedical processing [24, 56, 112, 113], due to fast convergence rate as compared to other reported techniques. In addition, it has parallel architecture, more suitable for higher order design, and real-time applications. Minor component analysis (MCA)-based neural learning technique is more preferred for digital filter design as compared to conventional Eigen filter design, in which an eigenvector corresponding to smallest eigenvalue of the associated matrix is computed. Recently, authors have applied MCA technique for the design of FIR and IIR FBs in [56, 113]. In literature, no method is given for optimized design of FIR filter and multirate filter bank, where PCs, MCA neural learning optimization, FDCs, and swarm intelligent optimization algorithms are employed simultaneously.

The literature available so far reveals that there is still need for an efficient technique, which shall enhance the filter response for quantized filter coefficient using different swarm-based techniques in design of different types of multiplierless FIR filters and multirate filter banks. Literature review also reveals that no such technique is available in the literature which can simultaneously design digital multiplierless FIR filter that can optimize the CSD coefficients and, at the same time, minimize the requirement of adders for MCM problems using single optimization. With these developments, there are strong motivations to undertake a thorough and systematic investigation toward the design and analysis of low complex multiplierless digital filter and filter bank using evolutionary algorithms with major focus on minimization resources.

Additionally, in several applications, it is needed to design digital filters and filter banks (FBs) with sophisticated design specifications. For example, high stopband attenuation (A_S), fast switching resolution and small channel, and overlapping are required for high-quality reconstruction of an audio signal, while for biomedical signal processing, fast switching resolution and adjustable A_S are highly desired. For software-defined radio (SDR) application, a flexible technology is required for multi-band, multi-standard, and multi-service, for that several important features of the channel filters such as low complexity, low power consumption, and reconfigurable are required. Thus, the sufficient and precise control of various frequencies such as passband cutoff frequency, stopband cutoff frequency, and transition width is required which is not possible with windowing technique.

In view of the above research gap recently, Kuldeep et al. [105–112] have presented the design of digital FIR filter and multirate filter bank using fractional derivative constraints, polyphase decomposition and nature-inspired optimization. Moreover, for efficient realization of multiplierless multirate system, Sharma et al. [134–136] have presented the design of multiplierless CMFB multirate filter bank, two-channel filter bank, etc. using evolutionary technique and sub-expression elimination algorithm in CSD space [137–139]. For further advancement in FIR filter and filter bank design, authors of this book proposed the design of digital FIR filter and M-channel multirate filter bank using MCA algorithm, fractional derivative constraints, polyphase decomposition, and nature-inspired optimization simultaneously (coming chapters in this book).

For further advancement in the field of optimal multirate filter bank design, following directions are suggested:

1. The optimal design of multirate filter can be achieved by improving the optimization algorithm. The hybrid combination of two conventional techniques such as gradient-based or swarm optimization methods can be employed to improve the optimization algorithm for filter bank.
2. Other form of fractional derivative like Riemann–Liouville, M. Caputo, can also be employed to achieve more optimal multirate filter bank.
3. The improved problem formulation like hybrid combination of L_2, L_1, and L_∞ error function can also improve the response of multirate filter bank.
4. The concept for sub-expression elimination technique (CSE) with CSD or other adder minimization techniques can be explored for designing digital multiplierless FIR filter and multiplierless multirate filter bank have not been introduced and analyzed since now.
5. The efficient design of multiplierless filter and filter bank having flexibility to satisfy the variety of sophisticated design specification has not been accomplished yet, which can be explored with the advancement of mathematical-based algorithms.
6. An efficient design of multirate system can be used as optimal trans-receiver system for 5G communication and other communication technology.
7. The concept of optimal filter bank structure can be utilized as filter bank-based sensing tool or may provide great aid in spectrum sharing and allocation for cognitive radio, smart antennas, and wireless sensor networks.

5 Conclusion

This chapter has presented a research survey on multirate filter banks design. A generalized theory of a multirate filter bank has been carried out. Research gap is also presented with various important aspects discussed between the reported methods and methods presented in this book in the field of digital FIR filter and multirate filter bank design. A future direction is also suggested for further advancements in the field of multirate filter bank design

References

1. Sohn, S. W., Bin Lim, Y., Yun, J. J., Choi, H., & Bae, H. D. (2012). A filter bank and a self-tuning adaptive filter for the harmonic and inter-harmonic estimation in power signals. *IEEE Transactions on Instrumentation and Measurement, 61*(1), 64–73.
2. Taskovski, D., & Koleva, L. (2012). Measurement of harmonics in power systems using near perfect reconstruction filter banks. *IEEE Transactions on Power Delivery, 27*(2), 1025–1026.

3. Le, T. K. (2012). Automated method for scoring breast tissue microarray spots using Quadrature mirror filters and Support vector machines. *15th International Conference Information Fusion (FUSION)*. pp. 1868–1875.
4. Chandran, S. (2003). Scheme for subband adaptive beamforming array implementation using quadrature mirror filter banks. *Electronics Letters, 39*(12), 891–892.
5. Chan, S. C., Pun, C. K. S., & Ho, K. L. (2004). New design and realization techniques for a class of perfect reconstruction two-channel FIR filterbanks and wavelets bases. *IEEE Transactions on Signal Processing, 52*(7), 2135–2141.
6. Afonso, V. X., Tompkins, W. J., Nguyen, T. Q., & Luo, S. (1999). ECG beat detection using filter banks. *IEEE Transactions on Biomedical Engineering, 46*(2), 192–202.
7. Chiang, H.-T., Phoong, S. M., & Lin, Y. P. (2007). Design of nonuniform filter bank transceivers for frequency selective channels. *EURASIP Journal on Advances in Signal Processing, 2007*, 061396.
8. Diniz P. S. R., Barcellos L. C. R., & Netto S. L. (2001). Design of cosine-modulated filter bank prototype filters using the frequency-response masking approach. *2001 I.E. International Conference on Acoustics, Speech, and Signal Processing Proceedings (Cat. No.01CH37221)*, Singapore, Singapore, *6*, 3621–3624.
9. Deng, Y., Mathews, V. J., & Farhang-Boroujeny, B. (2007). Low-delay nonuniform pseudo-QMF banks with application to speech enhancement. *IEEE Transactions on Signal Processing, 55*(5), 2110–2121.
10. Alaya, S., Zoghlami, N., & Lachiri, Z. (2014). Speech enhancement based on perceptual filter bank improvement. *International Journal of Speech Technology, 17*(3), 253–258.
11. Lian, J. A., & Wang, Y. (2014). Energy preserving QMF for image processing. *IEEE Transactions on Image Processing, 23*(7), 3166–3178.
12. Nawarathna, R., Oh, J., Muthukudage, J., Tavanapong, W., Wong, J., de Groen, P. C., & Tang, S. J. (2014). Abnormal image detection in endoscopy videos using a filter bank and local binary patterns. *Neurocomputing, 144*, 70–91.
13. Crochiere, R., Webber, S., & Flanagan, J. (1976). Digital coding of speech in sub-bands. *ICASSP '76. IEEE International Conference on Acoustics, Speech, and Signal Processing, 1*, 233–236.
14. Dolecek, G. J. (2002). *Multirate systems: Design and applications*. Hershey: Idea Group of Publishing.
15. Vaidyanathan, P. P. (1993). *Multirate systems and filter banks*. Delhi: Dorling Kindersley.
16. Mitra, S. K. (2006). *Digital signal processing: A computer based approach*. Singapore: Tata McGraw Hill Edition.
17. Crochiere, R. E. (1977). On the design of subband coders for low-bit-rate speech communication. *Bell System Technical Journal, 56*(5), 747–770.
18. Croisier A., Esteban D., & Galand C. (1976). Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques. in *International Conference on Information Sciences and Systems*. Patras, Greece.
19. Esteban, D., & Galand, C. (1977). Application of quadrature mirror filters to split band voice coding schemes. *ICASSP '77. International Conference on Acoustics, Speech, and Signal Processing, 2*, 191–195.
20. Fliege, N. J. (1994). *Multirate digital signal processing: Multirate systems, filter banks, wavelets*. New York, USA: *Wiley*.
21. Barnwell, T. (1982). Subband coder design incorporating recursive quadrature filters and optimum ADPCM coders. *IEEE Transactions on Acoustics, 30*(5), 751–765.
22. Swaminathan, K., & Vaidyanathan, P. (1986). Theory and design of uniform DFT, parallel, quadrature mirror filter banks. *IEEE Transactions on Circuits and Systems, 33*(12), 1170–1191.
23. Vaidyanathan, P., Regalia, P., & Mitra, S. (1987). Design of doubly-complementary IIR digital filters using a single complex all pass filter, with multirate applications. *IEEE Transactions on Circuits and Systems, 34*(4), 378–389.

24. Chen, L. W., Jou, Y. D., & Hao, S. S. (2014). Design of two-channel quadrature mirror filter banks using minor component analysis algorithm. *Circuits, Systems and Signal Processing, 34*(5), 1549–1569.
25. Smith, M., & Barnwell, T. (1984). A procedure for designing exact reconstruction filter banks for tree-structured subband coders. *ICASSP '84. IEEE International Conference on Acoustics, Speech, and Signal Processing, 9*, 421–424.
26. Mintzer, F. (1985). Filters for distortion-free two-band multirate filter Bank. *IEEE Transactions on Acoustics, Speech, and Signal Processing, 33*(3), 626–630.
27. Kim, C. W., & Ansari, R. (1991). FIR/IIR exact reconstruction filter banks with applications to subband coding of images. *Proceedings of 34th Midwest Symposium Circuits Systems*, Monterey, CA, USA. pp. 2–5.
28. Mitra, S. K., Creusere, C. D., & Babic, H. (1992). A novel implementation of perfect reconstruction QMF banks using IIR filters for infinite length signals. *Proceedings 1992 I. E. International Symposium on Circuits and Systems, 5*, 2312–2315.
29. Chen, S. G., Kao, M. C., & Chen, S. P. (1996). A new type of perfect-reconstruction QMF banks. *Conference Record of The Thirtieth Asilomar Conference on Signals, Systems and Computer, 70*(2), 1334–1338.
30. Chen, S. G., & Kao, M. C. (1998). Low-complexity, perfect reconstruction FIR QMF bank. *Electronics Letters, 34*(15), 1477–1478.
31. Hezar, R., & Madisetti, V. K. (1998). Efficient implementation of two-band PR-QMF filterbanks. *IEEE Signal Processing Letters, 5*(4), 92–94.
32. Smith, M. J. T., & Eddins, S. L. (1990). Analysis/synthesis techniques for subband image coding. *IEEE Transactions on Acoustics, 38*(8), 1446–1456.
33. Masson, J., & Picel, Z. (1985). Flexible design of computationally efficient nearly perfect QMF filter banks. *ICASSP '85. IEEE International Conference on Acoustics, Speech, and Signal Processing, 10*, 541–544.
34. Cox, R. (2003). The design of uniformly and non-uniformly spaced pseudo quadrature mirror filters. *IEEE Transactions on Acoustics, 34*(5), 1090–1096.
35. Vetterli, M. (1987). A theory of multirate filter banks. *IEEE Transactions on Acoustics, 35*(3), 356–372.
36. Koilpillai, R. D., & Vaidyanathan, P. P. (1992). Cosine-modulated FIR filter banks satisfying perfect reconstruction. *IEEE Transactions on Signal Processing, 40*(4), 770–783.
37. Koilpillai, R. D., & Vaidyanathan, P. P. (1993). A spectral factorization approach to pseudo-QMF design. *IEEE Transactions on Signal Processing, 41*(1), 82.
38. Kumar, A., Pooja, R., & Singh, G. K. (2016). An efficient closed-form design method for nearly perfect reconstruction of non-uniform filter bank. *ISA Transactions, 61*(1), 167–178.
39. Schlichthärle, D. (2015). *Digital filters – Basics and design*. Springer.
40. Galand, C., & Esteban, D. (1983). Design and evaluation of parallel quadrature mirror filters (PQMF). *IEEE International Conference on Acoustics, Speech, and Signal Processing, 8*, 224–227.
41. Nussbaumer, H. J., & Galand, C. (1983). Parallel filter banks using complex quadrature mirror filters (CQMF). In *Proceeding of Eusipco*, Erlangen, West Germany. pp. 69–72.
42. Crochiere, R. E., & Rabiner, L. R. (1983). *Multirate digital signal processing*, *Prentice-hall signal processing series*. Prentice-Hall: Englewood Cliffs.
43. Johnston, J. (1980). A filter family designed for use in quadrature mirror filter banks. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '80), 5*, 291–294.
44. Jain, V., & Crochiere, R. (1984). Quadrature mirror filter design in the time domain. *IEEE Transactions on Acoustics, 32*(2), 353–361.
45. Chen, C. K., & Lee, J. H. (1992). Design of quadrature mirror filters with linear phase in the frequency domain. *IEEE Transactions on Circuits and Systems II Analog and Digital Signal Processing, 39*(9), 593–605.

46. Yang, R. H. (1994). Novel efficient approach for the design of equiripple quadrature mirror filters. *IEEE Proceedings Vision, Image, and Signal Processing, 141*(2), 95–100.
47. Lim, Y. C., Yang, R. H., & Koh, S.-N. (1993). The design of weighted minimax quadrature mirror filters. *IEEE Transactions on Signal Processing, 41*(5), 1780–1789.
48. Goh, C. K., Lim, Y. C., & Ng, C. S. (1996). Improved algorithm to design weighted minimax quadrature mirror filters. *1996 I.E. International Symposium on Circuits and Systems. Circuits and Systems Connecting the World (ISCAS 96), 2*, 381–384.
49. Goh, C. K., Lim, Y. C., & Ng, C. S. (1999). Improved weighted least squares algorithm for the design of quadrature mirror filters. *IEEE Transactions on Signal Processing, 47*(7), 1866–1877.
50. Bregovic, R., & Saramaki, T. (2003). A general-purpose optimization approach for designing two-channel FIR filterbanks. *IEEE Transactions on Signal Processing, 51*(7), 1783–1791.
51. Vinod, A. P., & Premkumar, A. B. (2000). A generalized design of the quadrature mirror filters. *WCC – ICSP 5th International Conference on Signal Processing Proceedings. 16th World Computer Congress, 1*, 126–129.
52. Premkumar, A. B., & Vinod, A. P. (2000). Modified design to eliminate passband anomaly in weighted minimax quadrature mirror filters. *IEEE Signal Processing Letters, 7*(8), 224–226.
53. Vinod, A. P., Premkumar, A. B., & Tong, L. C. (2001). Novel approach to the design of weighted minimax quadrature mirror filters. *6th International Symposium Signal Processing and Its Applications ISSPA 2001 - Proceedings; 6 Tutorials Communication Image Processing Signal Analysis, 1*, 33–35.
54. Sunder, S., & Ramachandran, V. (1994). Design of equiripple nonrecursive digital differentiators and Hilbert transformers using a weighted least-squares technique. *IEEE Transactions on Signal Processing, 42*(9), 2504–2509.
55. Goh, C. K., & Lim, Y. C. (1997). An efficient weighted Lp algorithm for the design of quadrature mirror filters. *IEEE Signal Processing Letters, 4*(3), 79–81.
56. Jou, Y. D. (2007). Design of two channel linear phase QMF bank based on neural networks. *Signal Processing, 87*(5), 1031–1044.
57. Selesnick, I. W., Lang, M., & Burrus, C. S. (1996). Constrained least square design of FIR filters without specified transition bands. *IEEE Transactions on Signal Processing, 44*(8), 1879–1892.
58. Horng, B. R., & Willson, A. N. (1992). Lagrange multiplier approaches to the design of two-channel perfect-reconstruction linear-phase FIR filter banks. *IEEE Transactions on Signal Processing, 40*(2), 364–374.
59. Tuncer, T. E., & Nguyen, T. Q. (1995). General analysis of two-band QMF banks. *IEEE Transactions on Signal Processing, 43*(2), 544–548.
60. Nguyen, T. Q. (1995). Digital filter bank design quadratic-constrained formulation. *IEEE Transactions on Signal Processing, 43*(9), 2103–2108.
61. Saghizadeh, P., & Wilson, A. N. (1994). Using unconstrained optimization in the design of two-channel perfect-reconstruction linear-phase FIR filter banks. *Proceedings of 1994 37th Midwest Symposium on Circuits and Systems, 2*, 1053–1056.
62. Bregovic, R., & Saramäki, T. (2000). Design of two-channel low-delay FIR filter banks using constrained optimization. *Journal of Computing and Information Technology, 4*, 341–348.
63. Sahu, O. P., Soni, M. K., & Talwar, I. M. (2006). Marquardt optimization method to design two-channel quadrature mirror filter banks. *Digital Signal Processing A Review Journal, 16* (6), 870–879.
64. Kumar, A., Singh, G. K., & Anand, R. S. (2010). An improved method for designing quadrature mirror filter banks via unconstrained optimization. *Journal of Mathematical Modelling and Algorithms, 9*(1), 99–111.
65. Kumar, A., Singh, G. K., & Anand, R. S. (2013). An improved method for the design of quadrature mirror filter banks using the Levenberg Marquardt optimization. *Signal, Image and Video Processing, 7*(2), 209–220.

66. Kumar, A., Rafi, S. M., & Singh, G. K. (2012). A hybrid method for designing linear-phase quadrature mirror filter bank. *Digital Signal Processing A Review Journal, 22*(3), 453–462.
67. Nayebi, K., Barnwell, T. P., & Smith, M. J. T. (1994). Low delay FIR filter banks: Design and evaluation. *IEEE Transactions on Signal Processing, 42*(1), 24–31.
68. Xu, H., Lu, W.-S., & Antoniou, A. (1998). An improved method for the design of FIR quadrature mirror-image filter banks. *IEEE Transactions on Signal Processing, 46*(5), 1275–1281.
69. Lu, W.-S., Xu, H., & Antoniou, A. (1998). A new method for the design of FIR quadrature mirror-image filter banks. *IEEE Transactions on Circuits and Systems II Analog and Digital Signal Processing, 45*(7), 922–926.
70. Abdel-Raheem, E., El-Guibaly, F., & Antoniou, A. (1996). Design of low-delay two-channel FIR filter banks using constrained optimization. *Signal Processing, 48*(3), 183–192.
71. Chao, H. C. (2000). Two-channel filter banks satisfying low-delay and perfect-reconstruction design. *Signal Processing, 80*(3), 465–479.
72. Kliewer, J., & Brka, E. (2006). Near-perfect-reconstruction low-complexity two-band IIR/FIR QMF banks with FIR phase-compensation filters. *Signal Processing, 86*(1), 171–181.
73. Kayabol, K., & Akman, E. T. (2005). The peak-constrained optimization of stable linear-phase IIR PRQMF bank. *AEU - International Journal of Electronics and Communications, 59*(1), 8–14.
74. Abo-Zahhad, M. (2003). Current state and future directions of multirate filter banks and their applications. *Digital Signal Processing, 13*(3), 495–518.
75. Nguyen, T. Q. (1994). Near-perfect-reconstruction pseudo-QMF banks. *IEEE Transactions on Signal Processing, 42*(1), 65–76.
76. Nussbaumer, H. (1981). Pseudo QMF Bank. *IBM Technical Disclosure Bulletin, 24*(6), 3081–3087.
77. Rothweiler, J. (1983). Polyphase quadrature filters – a new subband coding technique. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '83), 8*, 1280–1283.
78. Creusere, C. D., & Mitra, S. K. (1995). Simple method for designing high-quality prototype filters for M-band pseudo QMF banks. *IEEE Transactions on Signal Processing, 43*(4), 1005–1007.
79. Vaidyanathan, P. P. (1998). A Kaiser window approach for the design of prototype filters of cosine modulated filterbanks. *IEEE Signal Processing Letters, 5*(6), 132–134.
80. Cruz-Roldán, F., Amo-López, P., Maldonado-Bascón, S., & Lawson, S. S. (2002). An efficient and simple method for designing prototype filters for cosine-modulated pseudo-QMF banks. *IEEE Signal Processing Letters, 9*(1), 29–31.
81. Cruz-Roldán, F., Bravo-Santos, Á. M., Martín-Martín, P., & Jiménez-Martínez, R. (2003). Design of multi-channel near-perfect-reconstruction transmultiplexers using cosine-modulated filter banks. *Signal Processing, 83*(5), 1079–1091.
82. Jain, A., Saxena, R., & Saxena, S. C. (2006). An improved and simplified design of cosine-modulated pseudo-QMF filterbanks. *Digital Signal Processing, 16*(3), 225–232.
83. Datar, A., Jain, A., & Sharma, P. C. (2010). Design of Kaiser window based optimized prototype filter for cosine modulated filter banks. *Signal Processing, 90*(5), 1742–1749.
84. Kumar, A., Singh, G. K., & Kuldeep, B. (2011). An improved and simplified approach for designing cosine modulated filter Bank using window technique. *Journal of Mathematical Modelling and Algorithms, 10*(3), 213–226.
85. Kumar, A., & Kuldeep, B. (2012). Design of M-channel cosine modulated filter bank using modified exponential window. *Journal of the Franklin Institute, 349*(3), 1304–1315.
86. Kumar, A., & Kuldeep, B. (2014). Design of cosine modulated pseudo QMF bank using modified Dolph-Chebyshev window. *International Journal of Signal and Imaging Systems Engineering, 7*(2), 126–133.

87. Bergen, S. W. A., & Antoniou, A. (2007). An efficient closed-form design method for cosine-modulated filter banks using window functions. *Signal Processing, 87*(5), 811–823.
88. Kumar, A., Singh, G. K., & Anand, R. S. (2011). An improved closed form design method for the cosine modulated filter banks using windowing technique. *Applied Soft Computing Journal, 11*(3), 3209–3217.
89. Bergen, S. W. A. (2008). A design method for cosine-modulated filter banks using weighted constrained-least-squares filters. *Digital Signal Processing A Review Journal, 18*(3), 282–290.
90. Kumar, A., Singh, G. K., & Anand, R. S. (2011). A simple design method for the cosine-modulated filter banks using weighted constrained least square technique. *Journal of the Franklin Institute, 348*(4), 606–621.
91. Cruz-Roldán, F., Martín-Martín, P., Sáez-Landete, J., Blanco-Velasco, M., & Saramäki, T. (2009). A fast windowing-based technique exploiting spline functions for designing modulated filter banks. *IEEE Transactions on Circuits and Systems—I: Regular Papers, 56* (1), 168–178.
92. Cruz-roldán, F., Santamaría, I., Bravo, Á. M., & Member, S. (2004). Frequency sampling design of prototype filters for filter banks nearly perfect reconstruction cosine-modulated filter banks. *IEEE Signal Processing Letters, 11*(3), 397–400.
93. Kokare, M., Chatterji, B. N., & Biswas, P. K. (2004). Cosine-modulated wavelet based texture features for content-based image retrieval. *Pattern Recognition Letters, 25*(4), 391–398.
94. Argenti, F. (2002). Design of cosine-modulated filterbanks for partial spectrum reconstruction. *Signal Processing, 82*(3), 389–405.
95. Yin, S. S., Chan, S. C., & Tsui, K. M. (2008). On the design of nearly-PR and PR FIR cosine modulated filter banks having approximate cosine-roll-off transition band. *IEEE Transactions on Circuits and Systems II: Express Briefs, 55*(6), 571–575.
96. Hameed, K. M. A., & Elias, E. (2006). M-channel cosine modulated filter banks with linear phase analysis and synthesis filters. *Signal Processing, 86*(12), 3842–3848.
97. Duan, Z., Zhang, J., Zhang, C., & Mosca, E. (2006). A simple design method of reduced-order filters and its applications to multirate filter bank design. *Signal Processing, 86*(5), 1061–1075.
98. Malvar, H. S. (1990). Modulated QMF filter banks with perfect reconstruction. *Electronics Letters, 26*(13), 906–907.
99. Park, S. Y., & Cho, N. I. (2006). Design of signed powers-of-two coefficient perfect reconstruction QMF bank using CORDIC algorithms. *IEEE Transactions on Circuits and Systems—I: Regular Papers, 53*(6), 1254–1265.
100. Baicher, G. S. (2007). Optimal Design of a Class of M-channel uniform filter Bank using genetic algorithms. *IEEE International Conference on Signal Processing, Communications*, 24–27.
101. Upendar, J., Gupta, C. P., & Singh, G. K. (2010). Design of two-channel quadrature mirror filter bank using particle swarm optimization. *Digital Signal Processing A Review Journal, 20*(2), 304–313.
102. Rafi, S. M., Kumar, A., & Singh, G. K. (2013). An improved particle swarm optimization method for multirate filter bank design. *Journal of the Franklin Institute, 350*(4), 757–769.
103. Zhu, Y., Huang, C., & Tao, W. (2010). Frequency domain optimization design of linear phase cosine modulated filter banks. *2010 International Conference on Measuring Technology and Mechatronics Automation ICMTMA, 2*(4), 313–316.
104. Tan, F., Zhang, T., Gao, C., & Huang, L. (2011). Optimal design of cosine modulated filter banks using quantum-behaved particle swarm optimization algorithm. *Proceedings – 4th International Congress on Image and Signal Processing CISP 2011, 5*, 2280–2284.
105. Kuldeep, B., Singh, V. K., Kumar, A., & Singh, G. K. (2015). Design of quadrature mirror filter bank based on nature inspired optimization based fractional derivative constraints. *ISA Transactions (Elsevier), 54*, 101–116.

106. Kuldeep, B., Kumar, A., & Singh, G. K. (2015). Design of multi-channel cosine modulated filter bank based on fractional derivative constraints using cuckoo search algorithm. *Circuits, Systems, and Signal Processing (Springer), 34*(10), 3325–3351.
107. Kuldeep, B., Kumar, A., & Singh, G. K. (2015). Hybrid method for designing digital FIR filters based on fractional derivative constraints. *ISA Transactions, 58*, 493–508.
108. Baderia, K., Kumar, A., & Singh, G. K. (2015). Design of quadrature mirror filter bank using polyphase components based on optimal fractional derivative constraints. *AEU International Journal of Electronics and Communications (Elsevier), 69*(9), 1254–1264.
109. Kuldeep, B., Kumar, A., & Singh, G. K. (2015). Design of quadrature mirror filter bank using Lagrange multiplier method based on fractional derivative constraints. *Engineering Science and Technology, an International Journal (Elsevier), 18*(2), 235–243.
110. Kuldeep, B., Kumar, A., & Singh, G. K. (2015). Design of multi-channel filter bank using ABC optimized fractional derivative constraints. *IEEE Conference on Communication and Signal Processing (ICCSP), 2*, 490–494.
111. Kuldeep, B., Kumar, A., & Singh, G. K. (2015). PSO based optimized fractional derivative constraints for designing M-channel filter bank. *IEEE Conference on Signal Processing Computing and Control (ISPCC), 3*, 140–144.
112. Baderia, K., Kumar, A., & Singh, G. K. (2016). An improved method for designing cosine modulated filter bank using polyphase components. *IEEE conference on Signal Processing & Integrated Networks (SPIN), 9-13*, 11–12.
113. Jou, Y. D., & Chen, F. K. (2011). WLS design of FIR Nyquist filter based on neural networks. *Digital Signal Process. A Review Journal, 21*(1), 17–24.
114. Chen, L. W., Jou, Y. D., Chen, F. K., & Hao, S. S. (2014). Eigen filter design of linear-phase FIR digital filters using neural minor component analysis. *Digital Signal Processing, 32*, 146–155.
115. Mehendale, M., Sherlekar, S. D., & Venkatesh, G. (1995). Synthesis of multiplierless FIR filters with minimum number of additions. In *Proceedings of the 1995 IEEE/ACM International conference on computer-aided design* (pp. 668–671). Los Alamitos: IEEE Computer Society Press.
116. Sandhiya, V., Karthick, S., & Valarmathy, M. (2014). A survey of new reconfigurable architectures for implementing FIR filters with low complexity. *International Conference on Computer Communication and Informatics IEEE, 2014*, 1–9.
117. Parhi, K. K. (2007). *VLSI digital signal processing systems: Design and implementation*. Singapore: Wiley.
118. Hewlitt, R. M., & Swartzlantler, E. S. (2000), Canonical seigned digit representation for FIR digital filters. *IEEE Work. SiGNAL Processing System SiPS 2000 Design Implementation (Cat. No.00TH8528), IEEE*, Lafayette, LA, USA. pp. 416–26.
119. Aktan, M., Yurdakul, A., & Dundar, G. (2008). An algorithm for the design of low-power hardware-efficient FIR filters. *IEEE Transactions on Circuits and Systems—I: Regular Papers, 55*(6), 1536–1545.
120. Fuller, A. T. G., Nowrouzian, B., & Ashrafzadeh, F. (1998). Optimization of FIR digital filters over the canonical signed-digit coefficient space using genetic algorithms. *Midwest Symposium Circuits and Systems (Cat. No. 98CB36268), IEEE Computer Society*, Notre Dame, IN, USA. pp. 456–459.
121. Park, I.-C., & Kang, H.-J. (2002). Digital filter synthesis based on an algorithm to generate all minimal signed digit representations. *IEEE Transactions on Computer Design Integration Circuits and Systems, 21*(12), 1525–1529.
122. Goel, N., & Nandi, A. (2015), Design of FIR Filter Using FCSD Representation. In *Computational Intelligence & Communication Technology (CICT), 2015 I.E. International Conference on*, Ghaziabad, India. pp. 617–620.
123. Samadi, P., & Ahmad, M. (2003). Performance analysis of genetic algorithm for the design of linear phase digital filterbanks with CSD coefficients. *Third International Conference on Natural Computation (ICNC), 3*, 150–154.

124. Yang, X. S., & Deb, S. (2010). Engineering optimisation by cuckoo search. *International Journal of Mathematical Modelling and Numerical Optimisation, 1*(4), 330–343.
125. Yang, X. S., & Deb, S. (2013). Multiobjective cuckoo search for design optimization. *Computers & Operations Research, 40*(6), 1616–1624.
126. Civicioglu, P., & Besdok, E. (2013). A conceptual comparison of the cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. *Artificial Intelligence Review, 39*(4), 315–346.
127. Chiaradonna, S., Giandomenico, F. D., & Murru, N. (2015). On enhancing efficiency and accuracy of particle swarm optimization algorithms. *International Journal of Innovative Computing, Information and Control, 11*(4), 1165–1189.
128. Ratanavilisagul, C., & Kruatrachue, B. (2014). A modified particle swarms optimization with mutation and reposition. *International Journal of Innovative Computing, Information and Control, 10*(6), 2127–2142.
129. Bindiya, T. S., & Elias, E. (2013). Modified metaheuristic algorithms for the optimal design of multiplierless non-uniform channel filters. *Springer Journal of Circuits Systems and Signal Processing, 33*(3), 815–837.
130. Reddy, K. S., & Sahoo, S. K. (2015). An approach for FIR filters coefficient optimization using differential evolution algorithm. *International Journal of Electronics and Communications, 69*(12), 101–108.
131. Misra, D., Dhabal, S., & Venkateswaran, P. (2015), Quadrature mirror filterbank with canonical signed digit representation using linear optimization algorithm. *Third International Conference on Computer Communication Control and Information Technology (C3IT)*, Hooghly, India. pp. 1–6.
132. Kalathil, S., & Elias, E. (2015). Non uniform cosine modulated filterbanks using metaheuristic algorithms in CSD space. *Journal of Advanced Research, 6*(6), 839–849.
133. Elias, E. (2015). Design of multiplierless sharp transition width non-uniform filterbanks using gravitational search algorithm. *International Journal of Electronics, 102*(1), 48–70.
134. Sharma, I., Kumar, A., Singh, G. K., & Lee, H. N. (2017). Design of Multiplier less prototype filter for two-channel Filterbank using hybrid method in FCSD space. *IET Circuit Device and System, 11*(1), 29–40.
135. Sharma, I., Kumar, A., & Singh, G. K. (2017). An efficient method for designing multiplierless non-uniform filterbank based on hybrid method using CSE technique. *Circuit System and Signal Processing, 36*(3), 1169–1191.
136. Sharma, I., Kumar, A., & Singh, G. K. (2016). Adjustable window based design of multiplierless cosine modulated filterbank using swarm optimization algorithms. *AEU-International Journal of Electronics and Communications, 70*(1), 85–94.
137. Sharma, I., Kuldeep, B., Kumar, A., & Singh, V. K. (2016). Performance of swarm based optimization techniques for designing digital FIR filter: A comparative study. *Engineering Science and Technology, an International Journal, 19*(3), 1564–1572.
138. Sharma, I., Kumar, A., & Singh, G. K. (2016). Design of two channel multiplierless filterbank using common sub-expression elimination. *Signal and Information Processing (IConSIP), International Conference on. IEEE.* Vishnupuri, India
139. Sharma, I., Kumar, A., & Singh, G. K., & Lee H. N. (2016). Design of multiplierless cosine modulated filterbank using hybrid technique in sub-expression space. *Digital Signal Processing (DSP), 2016* IEEE International conference on. *IEEE.* Beijing, China.

Methods for Improving Magnitude Characteristic of Comb Decimation Filters

Gordana Jovanovic Dolecek

1 Introduction

Decimation is a process of decreasing the sample rate by an integer factor in a digital form. This process finds applications in subband coding, filter banks, communication systems, and oversampled A/D (analog/digital) converters, among others [1, 2]. Decimation introduces unwonted replicas of the original signal spectrum (also called aliasing). If the signal is not appropriately filtered before decimation, the aliasing deteriorates the decimated signal. Therefore, to prevent aliasing in the decimated signal, the signal must be first filtered by a low-pass filter, called anti-aliasing, or decimation filter. As a result, the process of decimation consists of two principal stages: filtering and down-sampling (decreasing the sampling rate by integer M), as shown in Fig. 1, where f_{i} and f_{o} are input and output sampling rate, respectively. The integer value M is also called the decimation factor.

The most simple decimation filter is comb filter, usually used in the first stage of decimation [3]. This filter has all coefficients equal to unity and thus does not require the multipliers.

Its system function can be presented either in a recursive form

$$H(z) = \left[\frac{1}{M}\frac{1 - z^{-M}}{1 - z^{-1}}\right]^{K}, \tag{1}$$

or in equivalent non-recursive form

G.J. Dolecek (✉)
Department of Electronics, Institute National INAOE, Tonantzintla, Puebla, Mexico
e-mail: gordana.dolecek@gmail.com; gordana@ieee.org

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9_3

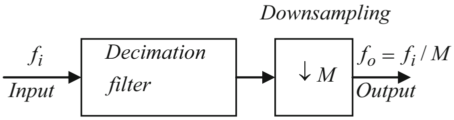


Fig. 1 Decreasing the sampling rate by integer factor *M*

$$H(z) = \left[\frac{1}{M}\sum_{k=0}^{M-1} z^{-k}\right]^K, \tag{2}$$

where K is the number of the cascaded filters (also called the order of the comb filter) and M is the decimation factor (length of the filter).

Comb filter naturally provides attenuations in bands around the comb zeros, called folding bands. It is important to have high attenuations in the folding bands, because the aliasing folds down into comb folding bands. Additionally, it is necessary to have a flat characteristic in the passband of interest, to avoid the deterioration of the decimated signal.

However, magnitude characteristic of comb filter

$$\left|H\left(e^{j\omega}\right)\right| = \left|\frac{1}{M}\frac{\sin(\omega M/2)}{\sin(\omega/2)}\right|^K, \tag{3}$$

does not provide enough attenuation in folding bands, and its magnitude characteristic in the passband is not flat. Attenuation can be increased by cascading comb filters. However, it increases droop of the passband characteristic, as shown in Fig. 2 for $M = 12$ and $K = 1, 2, 3$, and 4.

The comb passband is defined by the passband edge:

$$\omega_p = \pi/RM, \tag{4}$$

where R is the decimation factor of the stage that follows the comb decimation stage. Here we consider $R = 2$, and thus (4) for $R = 2$ defines a wideband.

Similarly, the comb folding bands are defined as:

$$\frac{2k\pi}{M} - \frac{\pi}{RM} \le \omega \le \frac{2k\pi}{M} + \frac{\pi}{RM}, \quad k = \begin{cases} 1, \ldots, M/2 & \text{for} \quad M(\text{even}) \\ 1, \ldots, (M-1)/2 & \text{for} \quad M(\text{odd}) \end{cases} \tag{5}$$

Different methods have been proposed to improve aliasing rejection in comb filters, compensate for the passband droop, and for both, simultaneously improving alias rejection and decreasing passband droop. The goal is to improve the magnitude comb characteristic without significantly increasing the overall complexity.

The objective of methods for increasing alias rejection is to increase the folding band widths and/or increase attenuation in the folding bands. This may be achieved by comb zero rotation, cosine filters, Chebyshev polynomials, cyclotomic polynomials, cascade of combs with different decimation factors, and changing specified filter coefficients, among others.

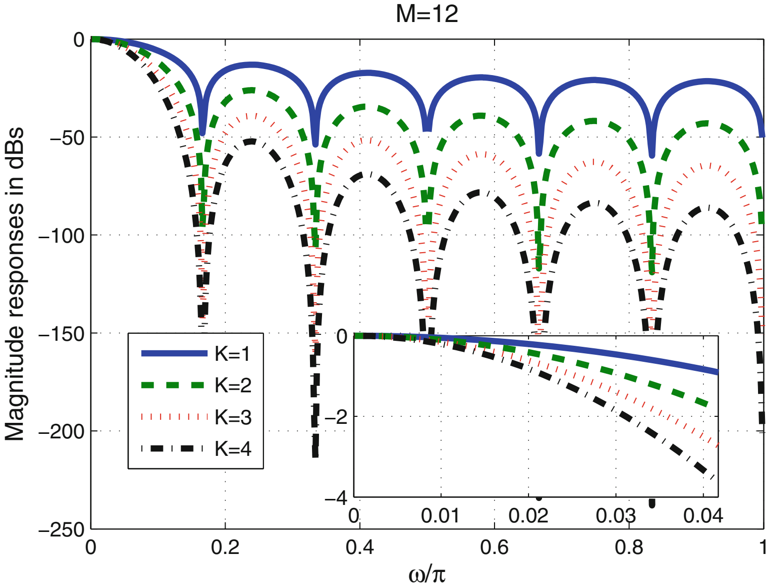


Fig. 2 Overall comb magnitude characteristics and passband zoom for $M = 12$ and $K = 1, 2, 3$, and 4

The comb zero rotation was introduced in [4] by applying both, a clockwise and opposite rotations of β radians to any zero of comb filter. The resulting filter is cascaded with comb filters and called *rotated sinc (RS) filter*. The method is generalized in [5] introducing a *generalized comb filter* (GCF) in which the parameter β is optimized to get the optimum alias rejection for a given M and K. However, the comb zero rotation introduces the following inconveniences:

- Two multipliers are introduced, one working at high input rate.
- Possible instability due to loss of pole – zero cancellation on the unit circle, in the case of the finite precision of filter coefficients.

As it is mentioned before, the popularity of comb decimation filter is due to its simplicity (comb filter is a multiplierless filter). Consequently, introducing multipliers for improvement of its characteristic degrades its principal characteristic expressed in its simplicity. As a result, it is desirable that the filter used to improve comb filter characteristic be also a multiplierless filter. Additionally, the solution to avoid possible instability is to eliminate the poles (cancelled by zeros for infinite precision of coefficients), on the unit circle using a non-recursive form for the rotation term. Different methods have been proposed to solve abovementioned problems [6–11]. In [12, 13], expanded cosine filters are used to introduce additional zeros in folding bands. The stopbands of a comb decimation filter can be improved by sharpening with a Chebyshev polynomial, using a comb variant [14]. As a result, comb multiple zeros are separated into an equiripple stopband. The advantage of the method in [14] is that the same controlled attenuations and width across all folding bands are obtained. The cascade of comb filters of different orders and with different decimation factors may result in higher attenuation than the equivalent comb filter as shown in [15–18].

Different methods for design of comb compensators are advanced in the literature. The compensator filters are usually finite impulse response (FIR) filters because FIR filters are always stable and can be designed to have a linear phase. The methods can be divided into two main groups: methods requiring multipliers [19–21] and methods with multiplierless designs [22–28].

The methods for simultaneous improvement of both passband and stopband can be categorized into two main groups: method based on sharpening technique [29], proposed in [30–35], and methods based on corrector filters [36, 37]. Additionally, there exist the methods that combine the methods for aliasing rejection and passband improvements [38].

In this chapter we present some recently introduced methods for alias rejection, passband droop compensation improvement, and method for simultaneous passband and stopband improvements.

2 Aliasing Rejection Improvement Based on Certain Characteristics of Symmetric Polynomials

2.1 Position of Zeros on Unit Circle for Symmetrical Polynomials

We consider a symmetric polynomial $f(z)$ with even degree $n = 2\,m$, $m > 0$

$$f(z) = a_0 + a_1 z + \ldots + a_{n-1} z^{(n-1)} + a_n z^n \tag{6}$$

where $a_k = a_{n-k}$, for $k = 0, \ldots m$.

The relative value of the middle coefficient of the polynomial in (6) with respect to the other coefficients is important for the position of its zeros on unit circle [39, 40].

Denoting

$$u = z + z^{-1}, \tag{7}$$

it is possible to uniquely associate a polynomial $g(u)$ to the polynomial $f(z)$, as

$$f(z) = z^{n/2} g(u) \tag{8}$$

with u defined in (7) [6, 7].

Theorem 1 [39, 40] *The zeros of the polynomial in (6) are all on the unit circle, if, and only if, the zeros of* g(u) *are all real and have values in the interval [−2, 2].*

The theorem is illustrated in the following example.

Example 1 Consider the polynomial (6) with $n = 4$ and coefficients $a_0 = a_1 = a_3 = a_4 = 1$. The coefficient a_2 should be determined according to Theorem 1, in order that all zeros of the polynomial are on the unit circle.

From (6) we have:

$$f(z) = 1 + z + a_2 z^2. + z^3 + z^4. \tag{9}$$

Using (7) and (8), we get

$$\begin{aligned} f(z) &= z^2\left(z^{-2} + z^{-1} + a_2 + z + z^2\right) \\ &= z^2\left[\left(z + z^{-1}\right) + \left(z^{-2} + 2 + z^2\right) + a_2 - 2\right]. \end{aligned} \tag{10}$$

Placing (7) into (10), we obtain:

$$f(z) = z^2\left[u + u^2 + a_2 - 2\right]. \tag{11}$$

Comparing (11) and (8), we arrive at:

$$g(u) = u + u^2 + a_2 - 2. \tag{12}$$

From (12) we easily find that the condition of the Theorem 1 is fulfilled for the following values of a_2: $0 \le a_2 \le 9/4$. Figure 3 illustrates pole-zero plots of the polynomial (9) for different values of a_2. Note that for the value of $a_2 = -1$ (which do not fulfill the condition of Theorem 1), two zeros are not on the unit circle.

2.2 How to Benefit from Theorem 1 for Improving Aliasing Rejection of Comb Filters?

The comb filter has all coefficients equal to unity (boxcar form), while the impulse response of the cascade of two combs has a triangular form. Omitting the normalization factor, the system function $F(z)$ of the comb cascade can be written in the form

$$\begin{aligned} F(z) &= \left[\sum_{k=0}^{M-1} z^{-k}\right]^2 \\ &= 1 + 2z^{-1} + \ldots + (M-1)z^{-(M-2)} + Mz^{-(M-1)} + (M-1)z^{-M} \ldots \\ &\quad + 2z^{-2M-3} + z^{-2(M-1)} = \frac{f(z)}{z^{2(M-1)}}, \end{aligned} \tag{13}$$

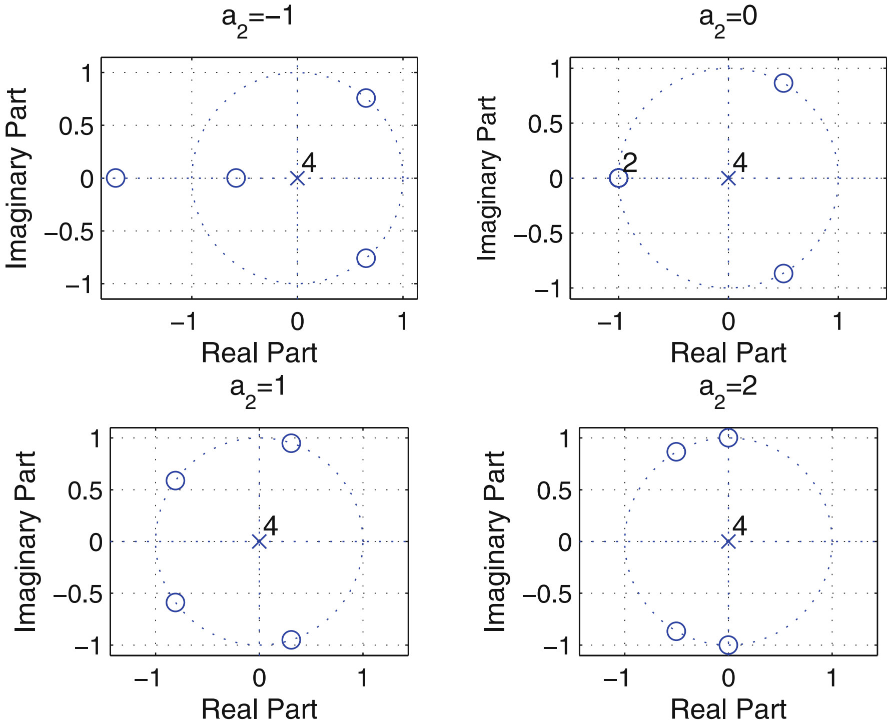


Fig. 3 Pole-zero plots for different values of coefficient a_2 in Example 1

where $f(z)$ is given in (6) with $m = M - 1$ and

$$a_0 = 1; \quad a_k = a_{k-1} + 1; \quad k = 1, \ldots, M-2; a_{M-1} = M \tag{14}$$

From (13) and (14), we have

$$\begin{aligned} f(z) = 1 + 2z + \ldots + (M-1)z^{(M-2)} + Mz^{(M-1)} + (M-1)z^{M} \ldots \\ + 2z^{2M-3} + z^{2(M-1)}. \end{aligned} \tag{15}$$

The zeros of the cascaded combs are doubled. According to the Theorem 1, changing the value of the middle coefficient in (15), we may get different distribution of zeros of (15) on the unit circle.

To this end, we consider the polynomial obtained from (15) by changing the middle coefficient M in (15) to the positive value of A:

$$\begin{aligned} p(z) = 1 + 2z + \ldots + (M-1)z^{(M-2)} + Az^{(M-1)} + (M-1)z^{M} \ldots + 2z^{2M-3} \\ + z^{2(M-1)}. \end{aligned} \tag{16}$$

The goal here is to separate the doubled zeros in (15) by appropriate choice of the value of A.

This idea is illustrated in the following example.

Example 2 Considering $M = 4$, we have

$$f(z) = 1 + 2z + 3z^2 + 4z^3 + 3z^4 + 2z^5 + z^6, \tag{17}$$

and

$$p(z) = 1 + 2z + 3z^2 + Az^3 + 3z^4 + 2z^5 + z^6. \tag{18}$$

In the following we show how Theorem 1 helps us to choose the appropriate value of A, to get the doubled zeros separated on the unit circle.

Using (18) we write:

$$\begin{aligned} p(z) &= 1 + 2z + 3z^2 + Az^3 + 3z^4 + 2z^5 + z^6 \\ &= z^3\left(z^{-3} + 2z^{-2} + 3z^{-1} + Az^3 + 3z + 2z^2 + z^3\right) \\ &= z^3 p_1(z). \end{aligned} \tag{19}$$

The expression in parenthesis in (19) is denoted as:

$$p_1(z) = z^{-3} + 2z^{-2} + 3z^{-1} + Az^3 + 3z + 2z^2 + z^3. \tag{20}$$

We rewrite (20) as:

$$p_1(z) - \left(z + z^{-1}\right)^3 = 2\left(z^{-2} + z^2\right) + A. \tag{21}$$

Similarly

$$p_1(z) - \left(z + z^{-1}\right)^3 - 2\left(z + z^{-1}\right) = A - 4. \tag{22}$$

Placing (7) into (22), we got

$$p_1(z) = u^3 + 2u + A - 4 = g(u). \tag{23}$$

Finally, from (19) and (23), we have

$$p(z) = z^3 g(u). \tag{24}$$

According to Theorem 1, all zeros of the polynomial $g(u)$ must be real and in the interval $[-2,2]$, in order that the polynomial $p(z)$ has all its zeros on the unit circle. We easily find that this condition is satisfied for the following values of A:

$$3 \le A \le 4. \tag{25}$$

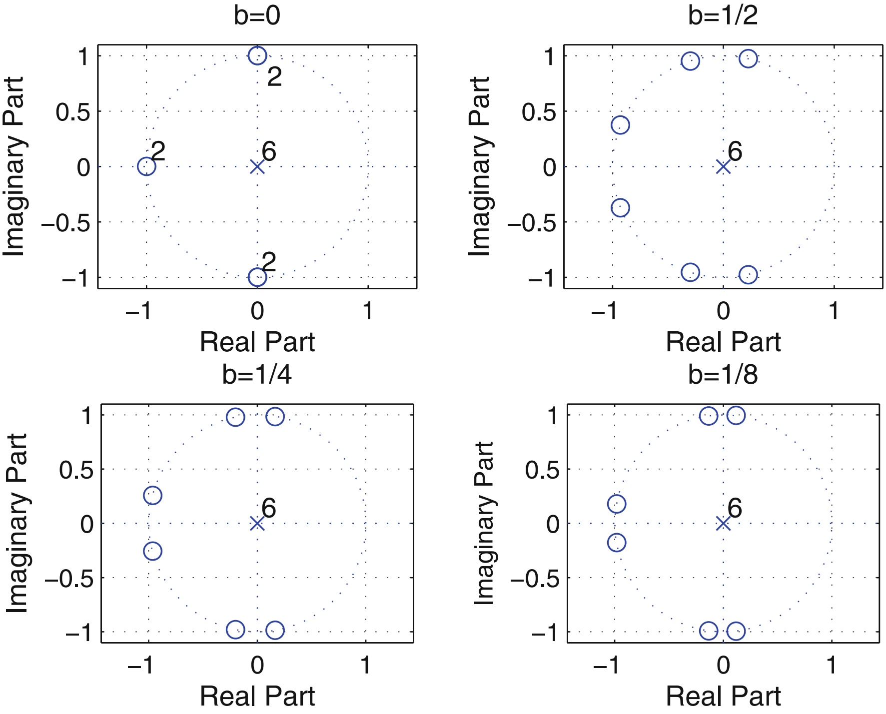


Fig. 4 Pole-zero plots of polynomial (19) for different values of *b*

Denoting $A = M - b = 3 - b$, where b is a positive value, the equivalent condition (25) is expressed in terms of the value of b, as

$$0 \leq b \leq 1. \tag{26}$$

Figure 4 shows pole-zero plots for the polynomial $p(z)$ for the values of b equal to 0, 1/2, ¼, and 1/8.

Note that, as a difference to the classical works on zero rotation in [4, 5], we get the comb zero rotation at low complexity.

2.3 Methods That Explore the Features of Symmetric Polynomials

Method in [41]

The features of symmetric polynomials, which are described in previous section, are explored in [41], and the *modified comb* was proposed as:

$$H_m(z) = \left[\sum_{k=0}^{M-1} z^{-k}\right]^2 - bz^{-(M-1)}, \tag{27}$$

where $b = 2^{-k}$, $k = 1,2,3,4$.

It is worth to mention that authors in [10, 11] proposed similar approaches, but from a different perspective; in [10], optimization is used to get the corresponding parameters, while in [11], the choice of parameters is not elaborated.

Finally, the proposed filter $G(z)$ is the cascade of K-2 combs (1) and the modified comb $H_m(z)$ (27):

$$G(z) = H^{K-2}(z)H_m(z) = H^{K-2}(z)\left[H^2(z) - bz^{-(M-1)}\right]. \tag{28}$$

The method is illustrated in the following example.

Example 3 Consider the comb parameters: $M = 12$ and $K = 4$ and the parameter of the modified comb, $b = 1/2$.

From (28), we have:

$$G(z) = H^2(z)\left[H^2(z) - 2^{-1}z^{-11}\right], \tag{29}$$

where $H(z)$ is given in (1) and (2).

Figure 5 shows the overall magnitude responses of the proposed and equivalent comb filters. The zoom in the first folding band is also shown.

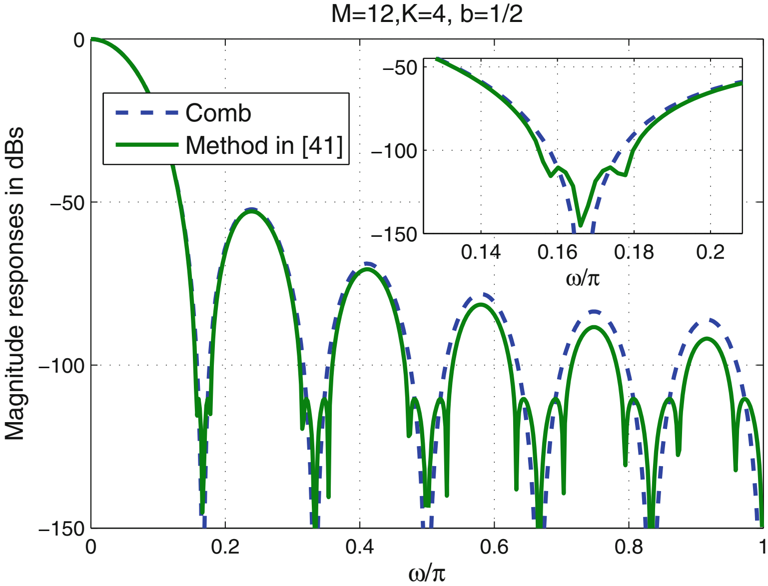


Fig. 5 Magnitude responses of comb and filter from [41]

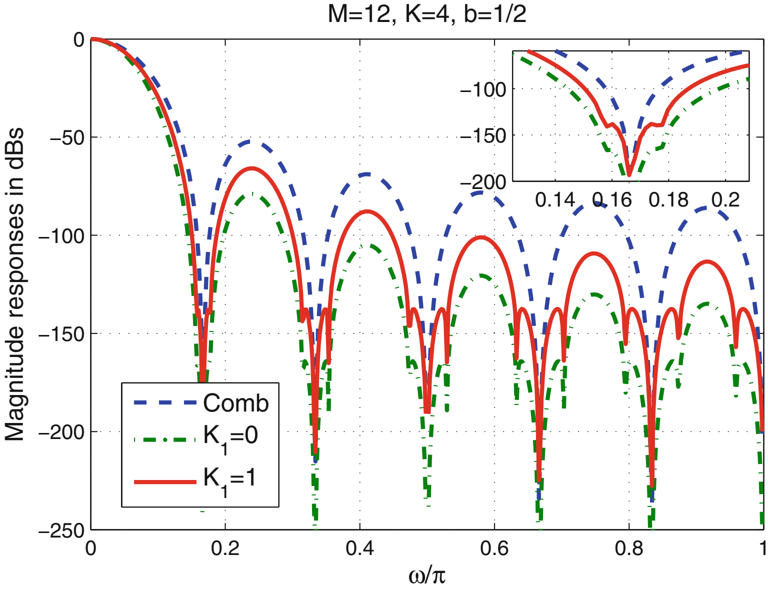


Fig. 6 Magnitude responses of comb and filter in [41] for values of $K_1 = 0$ and 1

In order to get more aliasing rejections in folding bands at the expense of an increased complexity, more general form of (28) is given as

$$G(z) = H^{K-K_1}(z)H_m(z) = H^{K-K_1}(z)\left[H^2(z) - bz^{-(M-1)}\right], \tag{30}$$

where $K_1 = 0$, 1, and 2. (The value $K_1 = 2$ is considered in (28)).

Figure 6 compares magnitude responses of comb filter and filter from [41] for $K_1 = 0$, and 1, for $M = 12$, $K = 4$, and $b = 1/2$. The zoom in the first folding band is also shown.

Modified Cosine Method [42]

In [42], two-stage filter for even decimation factors M is proposed. In the first stage is comb filter decimated by $M/2$,

$$H_1(z) = \left[\frac{1}{M/2}\frac{1 - z^{-M/2}}{1 - z^{-1}}\right]^K. \tag{31}$$

At the second stage is the cascade of a cosine filter

$$H_2(z) = \left[\frac{1}{2}\frac{1 - z^{-2}}{1 - z^{-1}}\right]^K = \left[\frac{1}{2}\left(1 + z^{-1}\right)\right]^K, \tag{32}$$

and the modified cosine filter:

$$F(z) = 1 + (2 - a)z^{-1} + z^{-2}, \tag{33}$$

decimated by 2.

The parameter a is chosen in a such way that the zeros of the expanded filter $F(z^{M/2})$

$$F\left(z^{M/2}\right) = 1 + (2 - a)z^{-M/2} + z^{-M}, \tag{34}$$

fall down in the proposed comb odd folding bands. More details are given in [42].

Using (31), (32), (33), and (34), the transfer function of the proposed filter at high input rate is given by:

$$H_{e.}(z) = H_1(z)H_2\left(z^{M/2}\right)\frac{1}{N}F\left(z^{M/2}\right) = \left[\frac{1}{M}\frac{1 - z^{-M}}{1 - z^{-1}}\right]^K \frac{1}{N}F\left(z^{M/2}\right), \tag{35}$$

where $N = 4\text{-}a$ is a normalization factor of F(z), necessary for the overall filter to have unity gain in the zero frequency, and K is the order of the comb filter.

The method is illustrated in the following example.

Example 4 Consider $M = 18$, $K = 3$, and the parameter $a = 1/2$. The first stage is decimated by $M_1 = 9$, while the second stage is decimated by 2. The overall system function is given as

$$H_{e.}(z) = \left[\frac{1}{18}\frac{1 - z^{-18}}{1 - z^{-1}}\right]^3 \frac{1}{N}F\left(z^9\right), \tag{36}$$

where $N = 4–1/2$ and $F(z)$ is given in (33).

The overall magnitude responses of comb and filter (36), along with the first folding band zoom, are given in Fig. 7.

Note that the alias rejection is increased in odd folding bands due to the additional zeros introduced by $F(z^9)$, as illustrated in Fig. 8.

3 Design of Compensators with a Magnitude Response Synthesized as Sinewave Functions

The compensators are designed to correct for the comb passband droop. Some desirable compensator characteristics are multiplierless design and a low absolute value of the passband deviation of compensated comb.

The magnitude response of the comb compensator has to approximate the inverse of the comb magnitude response in the comb passband. As a result, the cascade of comb and compensator should have the magnitude characteristic approximately equal to 1, in the passband.

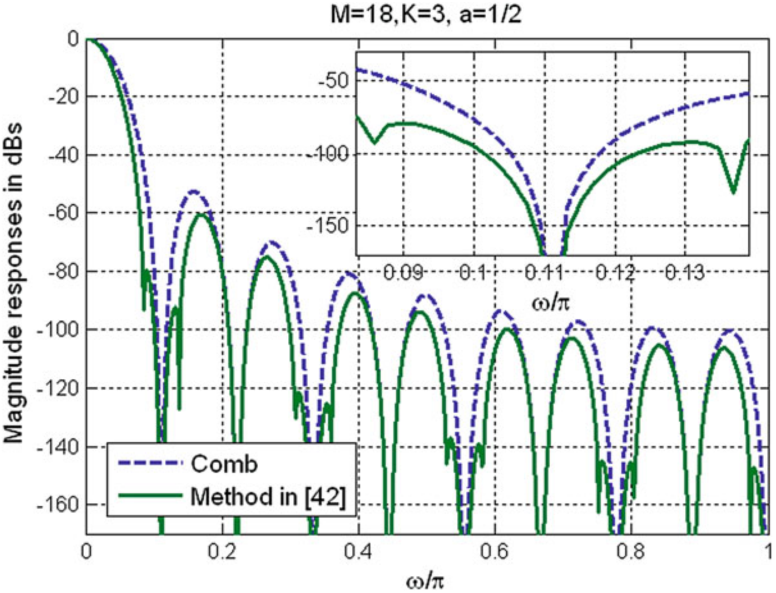


Fig. 7 Magnitude responses of comb and filter in [42]

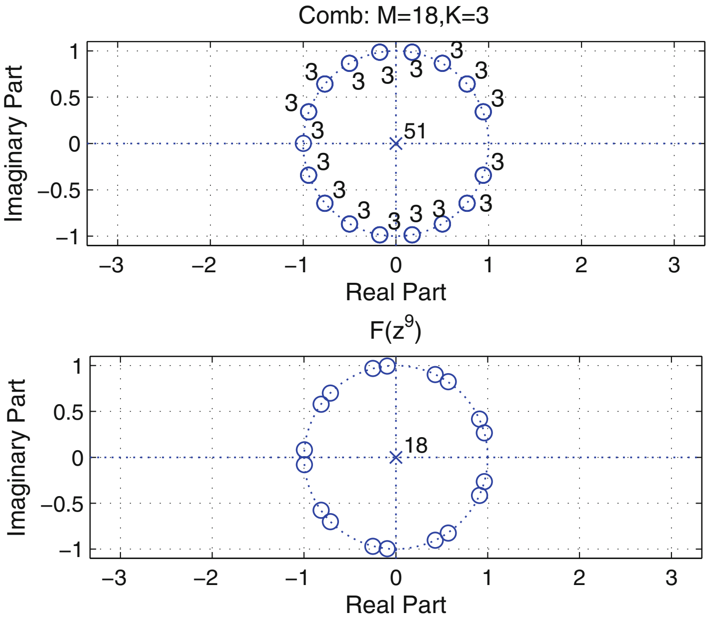


Fig. 8 Pole-zero plots for comb and expanded filter $F(z^9)$

3.1 Sine-Squared Method in [28]

The magnitude response of compensator has a sine-squared form

$$\left|G_2\left(e^{j\omega M}\right)\right| = 1 + B\sin^2(\omega M/2), \tag{37}$$

where B is amplitude of sine-squared function, M is a comb parameter, and subindex 2 denotes that a sine-squared function is used.

The magnitude characteristic of the compensated comb is

$$\left|H_c\left(e^{j\omega}\right)\right| = \left|H\left(e^{j\omega}\right)G_2\left(e^{j\omega M}\right)\right| = \left|\frac{1}{M}\frac{\sin(\omega M/2)}{\sin(\omega/2)}\right|^K \left[1 + B\sin^2(\omega M/2)\right], \tag{38}$$

where K is the order of comb filter.

Using a well-known trigonometrical relation,

$$\sin^2(\beta) = [1 - \cos(2\beta)]/2, \tag{39}$$

the system function of compensator, at low rate, is given as

$$\begin{aligned} G_2(z) &= 2^{-2}\left[-B + \left(2^2 + 2B\right)z^{-1} - Bz^{-2}\right] \\ &= 2^{-2}B\left[-1 + 2z^{-1} - z^{-2}\right] + z^{-1}. \end{aligned} \tag{40}$$

The values of parameter B depend on the given value of comb parameter K and do not depend on the comb parameter M, for $M > 10$, and are given in Table 1 [28].

The parameters B in Table 1 can be presented as adds and shifts resulting in a multiplierless design. The compensator needs 3 adders for $K = 1$, 2, and 4, and 4 adders for $K = 2$ and 5. In all cases the maximum absolute value of passband deviation is lower than 0.4 dB.

Next example illustrates the method.

Example 5 We chose two values of M (18 and 23) to show that the choice of the parameter B from Table 1 depends only on the value of K. The parameter $K = 4$ is used for both values of M. From Table 1 it follows $B = 1$. The overall magnitude responses and passband zooms, for comb and compensated comb, are given in Figs. 9 and 10, for $M = 18$ and 23, respectively.

We can observe that the compensator does not deteriorate comb folding bands.

Table 1 Values of parameter B for values of $K = 1,\ldots, 5$

K	B
1	1/4
2	1/2
3	3/4
4	1
5	5/4

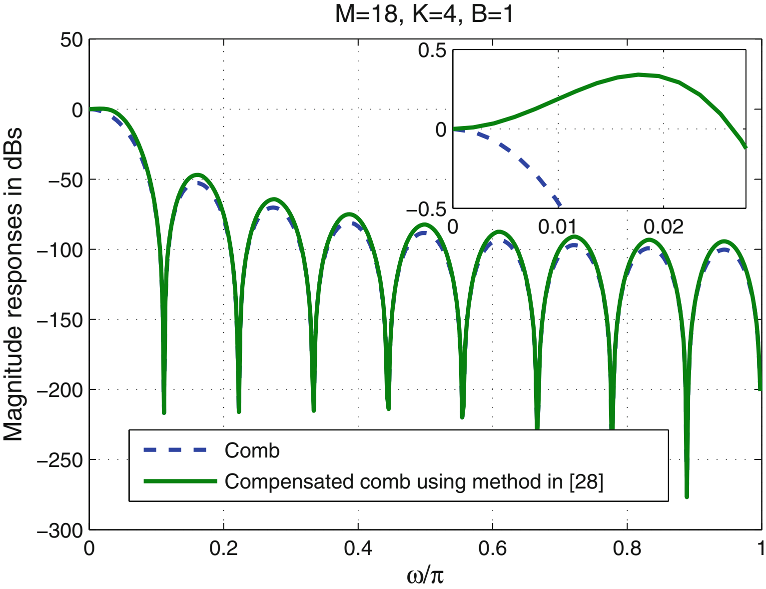


Fig. 9 Magnitude responses of comb and compensated comb for $M = 18$ and $K = 4$

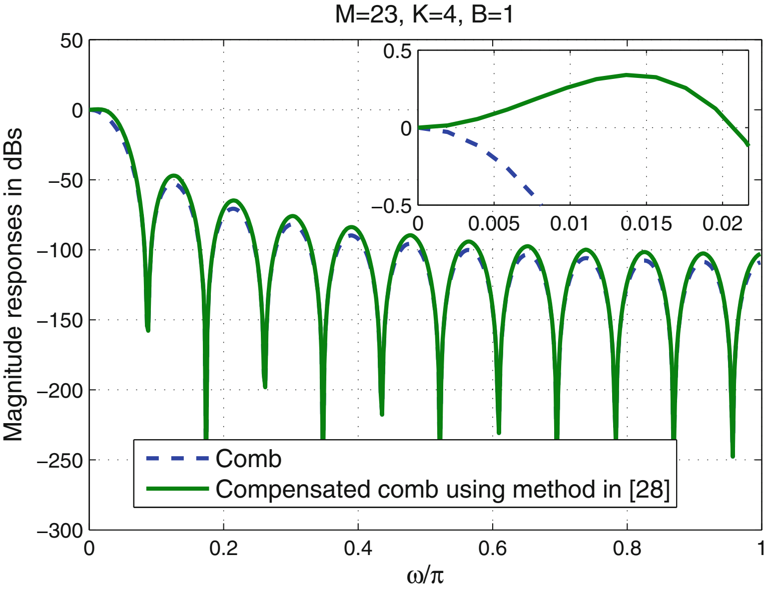


Fig. 10 Magnitude responses of comb and compensated comb for $M = 23$ and $K = 4$

3.2 Fourth-Order Sine-Based Magnitude Response Method in [43]

In order to get better approximation of the inverse comb magnitude characteristic, in [43] is proposed to cascade filter (37) with a filter with a fourth-order sine-based function magnitude response:

$$\left|G_4\left(e^{j\omega M}\right)\right| = 1 + A\sin^4(\omega M/2), \tag{41}$$

where subindex 4 means fourth-order sine function and A is the amplitude of fourth-order sine function.

Using well-known trigonometric relation

$$\sin^4(\beta) = [\cos(4\beta) - \cos(2\beta) + 3]/8, \tag{42}$$

the system function of the filter (41), at low rate, is expressed as [43]

$$\begin{aligned} G_4(z) &= 2^{-4}A[1 + z^{-4} - 4(z^{-1} + z^{-3})] + \left[2^{-3}3A + 1\right]z^{-2} \\ &= 2^{-4}A\left[1 + z^{-4} - 4(z^{-1} + z^{-3}) + \left(2^2 + 2\right)z^{-2}\right] + z^{-2}. \end{aligned} \tag{43}$$

From (37) and (41) the magnitude characteristic of compensator in [43] is given as

$$\begin{aligned} \left|G\left(e^{j\omega M}\right)\right| &= \left|G_2\left(e^{j\omega M}\right|\right|G_4\left(e^{j\omega M}\right| \\ &= \left[1 + A\sin^4(\omega M/2)\right]\left[1 + B\sin^2(\omega M/2)\right]. \end{aligned} \tag{44}$$

The system function of the compensator at low rate is given as

$$G(z) = G_2(z)G_4(z), \tag{45}$$

where $G_2(z)$and $G_4(z)$ are given in (40) and (43), respectively.

The values of parameters A and B are given in Table 2 [43]. The compensator (45) is multiplierless because the parameters A and B from Table 2 can be presented as adds and shifts. The compensator (45) provides absolute value of maximum passband deviation in a compensated comb lower than 0.1 dB.

The method is illustrated in Example 6.

Table 2 The values of parameters A and B for $K = 1,\ldots,6$

K	A	B
1	0	7/32
2	1/4	5/16
3	1/2	7/16
4	1/2	11/16
5	1	23/32
6	1	63/64

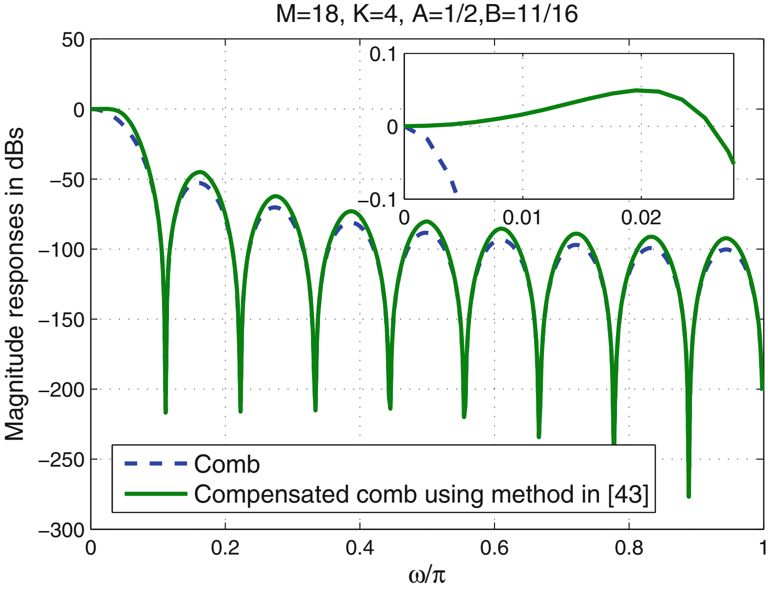


Fig. 11 Magnitude responses of comb and compensated comb using method in [43], for $M = 18$, $K = 4$

Example 6 We consider here the same comb parameters as in Example 5 in order to demonstrate the benefit of this method ($M = 18$ and 23, $K = 4$). The corresponding parameters in Table 2 are $A = 1/2$ and $B = 11/16$. Figures 11 and 12 show the overall magnitude responses and passband zooms for comb and compensated comb, respectively.

4 Simultaneous Passband and Stopband Improvements

4.1 Corrector Filters Method in [44]

The five novel simple multiplierless comb corrector filters, each for the given value of K, $K = 1,\ldots, 5$, and arbitrary even values of M are introduced in [44]. The filters are designed using the frequency sampling and IFIR methods and simultaneously compensate for the comb passband droop in the wideband passband region and increase the attenuations in the old folding bands. The corrector filter works at the rate which is $M/2$ lower than the high input rate, where M is the decimation factor.

To this end a two-stage comb structure is proposed. The first comb stage is decimated by $M/2$, while the second stage, which is the cascade of the combs and corrector $C(z)$, is decimated by 2.

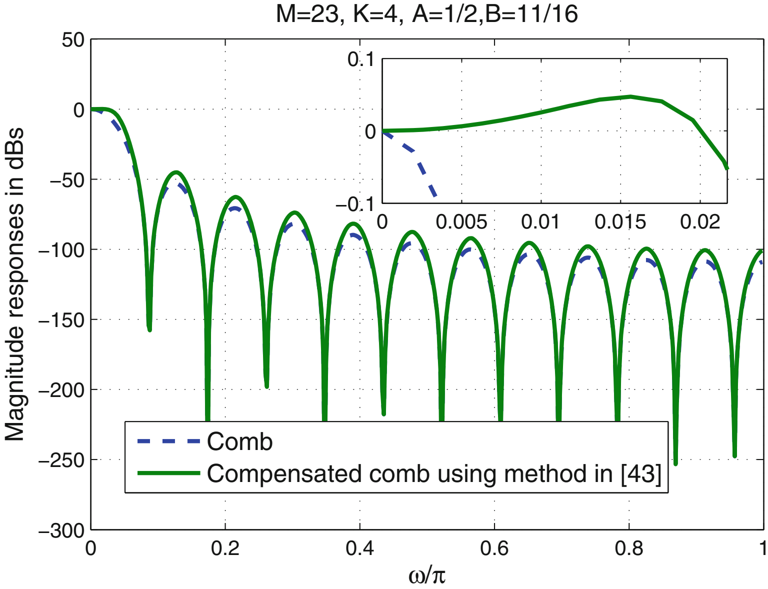


Fig. 12 Magnitude responses of comb and compensated comb using method in [43], for $M = 23$, $K = 4$

The system function of the overall filter is:

$$H_c(z) = H(z)C\left(z^{M/2}\right), \tag{46}$$

where

$$\begin{aligned} H(z) &= \left[\frac{1}{M}\left(\frac{1-z^{-M}}{1-z^{-1}}\right)\right]^K = \left[\frac{1}{M/2}\left(\frac{1-z^{-M/2}}{1-z^{-1}}\right)\right]^K \left[\frac{1}{2}\left(\frac{1-z^{-M}}{1-z^{-M/2}}\right)\right]^K \\ &= H_1(z)H_2\left(z^{M/2}\right), \end{aligned} \tag{47}$$

and

$$H_1(z) = \left[\frac{1}{M/2}\left(\frac{1-z^{-M/2}}{1-z^{-1}}\right)\right]^K; \quad H_2\left(z^{M/2}\right) = \left[\frac{1}{2}\left(\frac{1-z^{-M}}{1-z^{-M/2}}\right)\right]^K; \tag{48}$$

$$H_2(z) = \left[\frac{1}{2}\left(\frac{1-z^{-2}}{1-z^{-1}}\right)\right]^K = \left[\frac{1}{2}\left(1+z^{-1}\right)\right]^K. \tag{49}$$

Table 3 Coefficients of corrector filters for the values of $K = 1, \ldots, 5$, [44].

K	C
1	[−3,2,17,17,2,−3]
2	[1,−1,−5,3,18,18,3,−5,−1,1]
3	[1,1,−5,−5,12,24,12,−5,−5,1,1]
4	[1,1,−2,−8,1,24,24,1,−8,−2,1,1]
5	[1,2,−2,−11,0,27,27,0,−11,−2,2,1]

The comb filter (49) is also known as *cosine filter*, because its magnitude response has a cosine form. The non-normalized coefficients of corrector filters are shown in Table 3.

Method is illustrated in Example 7, for two different even values of M and the same value of K.

Example 7 We consider here the values of M equal to 18 and 22, and $K = 4$. The coefficients of the corrector filter are taken from Table 3. Figure 13a shows the overall magnitude responses of comb and the filter from [44], for $M = 18$ and $K = 4$. Similarly, Fig. 13b shows the zooms in the passband and the first folding band, to show the improvement in both bands.

From the other side, Fig. 14a, b show overall magnitude responses, and zooms in the passband and first folding bands, respectively.

5 Conclusions

We presented some recent low complexity methods, proposed for improvement of magnitude characteristic of comb decimation filter.

First, we elaborated the methods for increasing alias rejection based on certain characteristics of symmetric polynomials. Practically, one can get comb zero rotation just changing the middle coefficient of the impulse response of cascaded comb.

In the following we presented some recent proposed methods for the compensation for the comb passband droop, based on sinusoidal form of compensator magnitude characteristic. The compensators work at low rate and are multiplierless and may provide the maximum absolute value of passband deviation lesser than 0.1 dB, in the compensated comb.

Finally, a simultaneous improvement of comb magnitude characteristic was addressed in both passband and the folding bands. The method, based on simple multiplierless corrector filters, is described.

All presented methods are illustrated with examples and then compared with the corresponding comb filters.

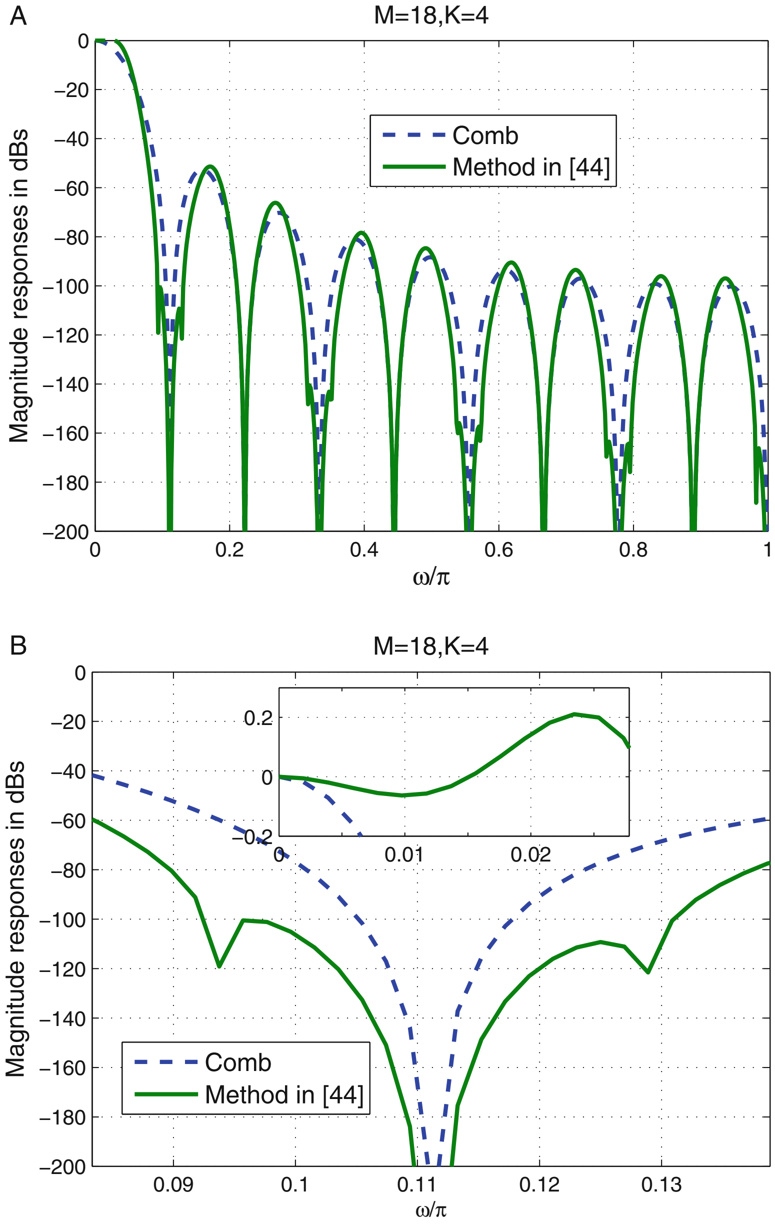


Fig. 13 Improving the comb magnitude characteristic using correctors from [44], for $M = 18$ and $K = 4$. (**a**) Overall magnitude responses. (**b**) Passband and first folding band zooms

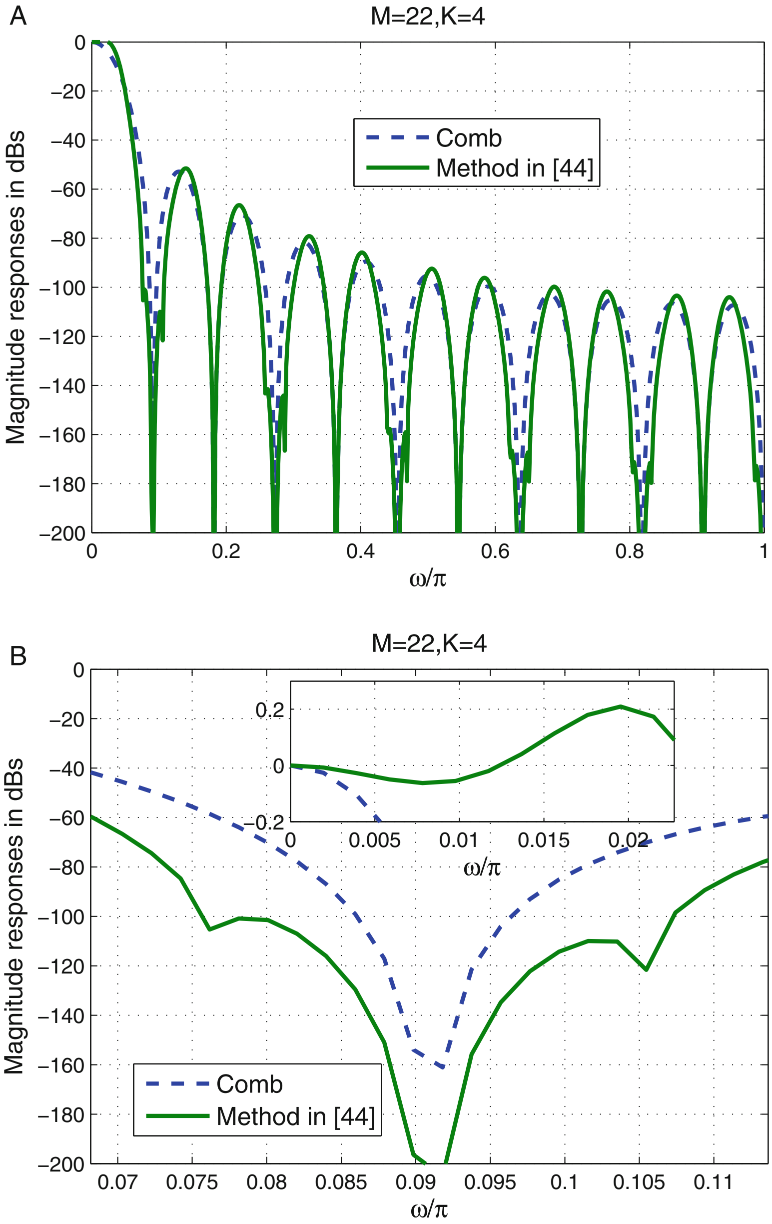


Fig. 14 Improving the comb magnitude characteristic using correctors from [44], for $M = 22$ and $K = 4$. (**a**) Overall magnitude responses. (**b**) Passband and first folding band zooms

References

1. Harris, F. J. (2004). *Multirate signal processing for communications systems*. New Jersey, USA: Prentice Hall.
2. Jovanovic Dolecek, G. (Ed.). (2001). *Multirate systems: Design and applications*. Hershey, PA, USA: IGI Global.
3. Hogenauer, E. (1981). An economical class of digital filters for decimation and interpolation. *IEEE Transactions on Acoustic, Speech, and Signal Processing, 29*(2), 155–162.
4. Presti, L. L. (2000). Efficient modified-sinc filters for sigma-delta A/D converters. *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, 47*(11), 1204–1213.
5. Laddomada, M. (2007). Generalized comb decimation filters for Sigma-Delta A/D converters: Analysis and design. *IEEE Transactions on Circuits and Systems-I: Regular papers, 54*(5), 994–1005.
6. Jovanovic Dolecek, G., & Mitra, S. K. (2004). Efficient multistage comb-modified rotated sinc (RS) decimator. In *Proceedings of european signal processing Conference EUSIPCO-2004* (pp. 1425–1428). Vienna, Austria: IEEEXplore.
7. Jovanovic Dolecek, G. (2010). Simplified rotated sinc (RS) filter for Sigma-Delta a/D conversion. In *Proceedings of International Conference on green Circuits and systems ICGCS 2010* (pp. 283–288). Shanghai, China: IEEEXplore.
8. Jovanovic Dolecek, G., & Fernandez-Vazquez, A. (2012). On non recursive rotated comb filter. In *Proceedings of Midwest Symposium MWSCAS 2012* (pp. 474–477). Boise, Idaho, USA: IEEEXplore.
9. Jovanovic Dolecek, G., & Laddomada, M. (2015). Design of two-stage non recursive rotated comb decimation filters with droop compensation and multiplier less architecture. *Journal of the Franklin Institute, 352*(3), 913–929.
10. Saramaki, T., & Ritoniemi, T. (1997). A modified comb filter structure for decimation. In *Proceedings of the IEEE International Symposium on Circuits and systems (ISCAS) 1997* (pp. 2353–2356). Hong Kong, China: IEEEXplore.
11. Harris, F. J., & Dolecek, G. (2016). Bifurcate repeated stop-band zeros of CIC filter. In *Proceedings of the IEEE international conference on digital signal processing (DSP) 2016* (pp. 365–369). Beijing, China: IEEEXplore.
12. Jovanovic Dolecek, G., & Laddomada, M. (2013). An improved class of multiplierless decimation filters: Analysis and design. *Digital Signal Processing, 23*(5), 1773–1782.
13. Garcia Robles, A., & Jovanovic Dolecek, G. (2016). On non-recursive comb-cosine decimation structures. In *Proceedings of the IEEE Latin American symposium on circuits and systems LASCAS 2016* (pp. 1–4). Florianapolis, Brazil: IEEEXplore.
14. Coleman, J. O. (2012). Chebyshev stopbands for CIC decimation filters and CIC- implemented array tapers in 1D and 2D. *IEEE Transactions on Circuits and Systems-I: Regular papers, 59* (12), 2956–2968.
15. Milic, D. N., & Pavlovic, V. D. (2014). A new class of low complexity lowpass multiplierless linear-phase special CIC FIR filters. *IEEE Signal Processing Letters, 21*(12), 1511–1515.
16. Stošić, B. P., Milić, D. N., & Pavlović, V. D. (2015). New CIC filter architecture: Design, parametric analysis and some comparisons. *IETE Journal of Research, 61*(3), 244–250.
17. Stošić, B. P., & Pavlović, V. D. (2016). Design of new selective CIC filter functions with passband-droop compensation. *Electronics Letters, 52*(2), 115–117.
18. Jovanovic Dolecek, G., & Fernandez-Vazquez, A. (2016). Multiplierless two-stage comb structure with an improved magnitude characteristic. In *Proceedings of 2016 I.E. Asia Pacific conference on circuits and systems (APCCAS) 2016, october, 2016*. Jeju, Korea: IEEEXplore.
19. Kim, S., Lee, W. C., Alm, S., & Choi, S. (2006). Design of CIC roll-off compensation filter in a W-CDMA digital receiver. *Digital Signal Processing, 16*(6), 846–854.

20. Molnar, G., & Vucic, M. (2011). Closed-form design of CIC compensators based on maximally flat error criterion. *IEEE Transactions on Circuits and Systems II: Express Brief, 58*(12), 926–930.
21. Jovanovic Dolecek, G., & Mitra, S. K. (2008). Simple method for compensation of CIC decimation filter. *Electronics Letters, 44*(19), 1162–1163.
22. Jovanovic Dolecek, G. (2009). Simple wideband CIC compensator. *Electronics Letters, 45* (24), 1270–1272.
23. Jovanovic Dolecek, G., & Harris, F. (2009). Design of wideband compensator filter for a digital IF receiver. *Digital Signal Processing, 19*(5), 827–837.
24. Jovanovic Dolecek, G., & Dolecek, L. (2010). Novel multiplierless wide-band CIC compensator. In *Proceedings of 2010 I.E. international symposium on circuits and systems (ISCAS) 2010* (pp. 283–288). Paris, France: IEEEXplore.
25. Fernandez-Vazquez, A., & Jovanovic Dolecek, G. (2012). Maximally flat CIC compensation filter: Design and multiplierless implementation. *IEEE Transactions on Circuits and Systems II: Express Brief, 54*(2), 113–117.
26. Pecotic, M., Molnar, G., & Vucic, M. (2012). Design of CIC compensators with SPT coefficients based on interval analysis. In *Proceedings of 35th IEEE International. Convention MIPRO 2012* (pp. 123–128). Opatija, Croatia: IEEEXplore.
27. Troncoso Romero, D. E., & Jovanovic Dolecek, G. (2013). Application of amplitude transformation for compensation of comb decimation filters. *Electronics Letters, 49*(16), 985–987.
28. Jovanovic Dolecek, G., & Fernandez-Vazquez, A. (2014). Trigonometrical approach to design a simple wideband comb compensator. *International Journal of Electronics and Communications, 68*, 437–441.
29. Kaiser, F., & Hamming, R. W. (1977). Sharpening the response of a symmetric nonrecursive filter by multiple use of the same filter. *IEEE Transactions Acoustic, Speech and Signal Processing, 25*(5), 415–422.
30. Kwentus, A., & Willson Jr., A. (1997). Application of filter sharpening to cascaded integrator-comb decimation filters. *IEEE Transactions on Signal Processing, 45*(2), 457–467.
31. Jovanovic Dolecek, G., & Mitra, S. K. (2003). Efficient sharpening of CIC decimation filter. In *Proceedings 2003 international conference on acoustics, speech, and signal processing (ICASSP) 2003* (Vol. 6, pp. 385–388). Hong Kong, China: IEEEXplore.
32. Laddomada, M., & Mondin, M. (2004). Decimation schemes for Sigma-Delta a/D converters based on kaiser and hamming sharpened filters. *IEE Proceedings of Vision, Image and Signal Processing, 151*(4), 287–296.
33. Jovanovic Dolecek, G., & Mitra, S. K. (2005). A new two-stage sharpened comb decimator. *IEEE Transactions on Circuits and Systems-I: Regular Papers, 52*(7), 1416–1420.
34. Nikolic, M., & Lutovac, M. (2011). Sharpening of the multistage modified comb filters. *Serbian Journal of Electrical Engineering, 8*(2), 281–291.
35. Molnar, G., Pecotic, M. G., & Vucic, M. (2013). Weighted least-squares design of sharpened CIC filters. In *Proceedings 36th international convention on information & communication technology electronics & microelectronics (MIPRO) 2013* (pp. 91–95). Opatija, Croatia: IEEEXplore.
36. Jovanovic Dolecek, G., & Fernandez-Vazquez, A. (2013). Novel droop-compensated comb decimation filter with improved alias rejections. *International Journal for Electronics and Communication, 67*(5), 387–396.
37. Molina Salgado, G., Jovanovic Dolecek, G., & de la Rosa, J. M. (2015). Novel two-stage comb decimator with improved frequency characteristic. In *Proceedings latin american Circuits and systems Symposium LASCAS 2015* (pp. 1–4). Montevideo, Uruguay: IEEEXplore.
38. Jovanovic Dolecek, G., & Mitra, S. K. (2010). Two-stage CIC-based decimator with improved characteristics. *IET Signal Processing, 4*(1), 22–29.
39. Konvalina, J., & Matache, V. (2004). Palindrome-polynomials with roots on the unit circle. *Comptes Rendus Mathematiques, 26*(2), 39–44.

40. Conrad, K. Roots on a circle. http://www.math.uconn.edu/~kconrad/blurbs/galoistheory/numbersoncircle.pdf.
41. Jovanovic Dolecek, G., & Dolecek, L. (2016). Exploiting features of symmetric polynomials for improved comb filter design. In *Proceedings of signal processing: algorithms, architectures, arrangements, and applications (SPA), 2016* (pp. 26–29). Poznan, Poland: IEEEXplore.
42. Jovanovic Dolecek, G., Garcia Baez, R., & Laddomada, M. (2017). Design of efficient multiplierless modified cosine-based comb decimation filters: Analysis and implementation. *IEEE Transactions on Circuits I: Regular papers, 64*(5), 1051–1063. Published online February 2017.
43. Jovanovic Dolecek, G., Garcia Baez, R., Molina Salgado, G., & de la Rosa, J. M. (2017). Novel multiplierless wideband comb compensator with high compensation capability. *Circuits, Systems and Signal Processing, 36*(5), 2031–2049.
44. Jovanovic Dolecek, G., & Fernandez-Vazquez, A. (2013). Novel droop-compensated comb decimation filter with improved alias rejections. *International Journal for Electronics and Communications, (AEUE), 67*(5), 387–396.

Design of Multichannel Filter Bank Using Minor Component Analysis and Fractional Derivative Constraints

B. Kuldeep, A. Kumar, G.K. Singh, and Heung-No Lee

1 Introduction

A substantial progress has been made toward multirate signal processing, especially design of digital filter banks for numerous applications such as coding of audio and video signal, biomedical signal processing, analog-to-digital converter, smart antenna, and cognitive radio (CR) [1]. Preliminarily, the quest was confined to two-channel filter banks, also called as quadrature mirror filter (QMF) banks, and, subsequently, directed to *M*-channel FBs, especially cosine-modulated filter banks (CMFBs) due to their simple design [1, 2]. The theory of CMFBs was evolved by several researchers and available in various books [1, 2]. The first systematic simple iterative technique for designing a prototype filter for CMFBs was proposed using a linear search optimization in [3], and this technique was further modified using different types of window functions [4–8]. Since windowing technique produces reduced stopband error with the passband ripple to be approximately equal to stopband ripple in CMFB design. Therefore, a new method based on the weighted constrained least square (WCLS) technique was proposed for designing a CMFB for given stopband attenuation, passband ripple, and channel overlapping [9] and

---

B. Kuldeep (✉)
National institute of Technology Hamirpur, Hamirpur, HP 177005, India

A. Kumar
PDPM Indian Institute of Information Technology Design and Manufacturing,
Jabalpur, MP 482005, India
e-mail: anilkdee@gmail.com

G.K. Singh
Indian Institute of Technology Roorkee, Uttrakhand 247667, India

H.-N. Lee
School of Electrical Engineering and Computer Science, Gwangju Institute of Science
and Technology, 123 Cheomdan-gwagiro, Buk-gu, Gwangju 61005, South Korea

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9_4

was further improved in [10]. A new closed form design method for CMFB was presented for the applications which are carried out in real time or quasi real time in [11] and has been further extended for two-channel filter bank [12]. Authors in [13, 14] have proposed an improved closed form method for designing CMFBs using spline function in transition band. Thus, it is evident from literature review that extensive work has been carried out for designing CMFBs based on either optimizing passband edge frequency (ω_p) or cutoff frequency (ω_c) using linear search optimization.

To overcome the differentiability and continuity problem of an objective function in an optimization problem, some gradient-free methods such as genetic algorithm (GA) were employed for the design of CMFB, in which the cost function has been constructed as weighted sum of amplitude distortion and aliasing distortion [15]. Subsequently, PSO and its variants were further employed for the design of CMFB using different objective functions [16]. A new technique using evolutionary algorithms was devised to design more optimized, and flexible multiplierless CMFBs, for the given stopband attenuation (A_s) and channel overlapping [17]. In [18], conventional integer derivative constraints have been exploited for designing digital finite impulse response (FIR) filters to improve design accuracy at the prescribed frequency point. However, the fractional calculus has shown improved performance in many engineering applications such as electrical networks, electromagnetic theory, biomedical applications, and signal and image processing as compared to integer derivative [19]. Fractional derivative increases the possibility of improving control performance by reducing the convergence time in mentioned control problems. Therefore, a new approach for designing linear-phase FIR filters based on fractional derivative constraints (FDCs) was devised in [20] and has been extended for two-dimensional FIR filters [21] and filter banks [22]. These techniques yield the improved designs; however, order of FDCs has been selected on trial and error method, which is time consuming. To overcome this problem, a new technique for determining optimal fractional derivative constraints using swarm-based optimizations for efficient design of FB was proposed in [23]. This method has been further improved by employing optimized PCs using Lagrange multiplier enforcing swarm optimized FDCs [24–28].

During the past decade, neural networks have been emerged as a very efficient and robust tool for solving linear and nonlinear signal processing applications, which are carried out in real and quasi real time, due to fast convergence, higher-order compatibility, and parallel structure [29]. Therefore, neural networks have been extensively used in various engineering applications such as antenna array and biomedical signal processing [30–31]. Authors in [32] have also exploited the neural networks for designing a two-channel filter bank. A new technique based on neural minor component analysis was proposed of designing digital FIR filters [33], and this was further utilized for solving phase optimization problem, constructed for the design of infinite impulse response (IIR) QMF bank [34]. Thus, it is evident from literature review on neural networks for signal processing that neural network-based methods are more efficient as compared to other iterative methods, and they are highly suitable for higher-order system design and real-time applications. Only few references have been reported in literature for designing either FIR filters or IIR filter-based two-channel filter bank, based on minor component analysis [33, 34]. So far, no technique has been reported for designing *M*-channel cosine-modulated filter bank, where PCs, MCA-based neural learning, and swarm optimization for fractional derivative constraints have been employed simultaneously.

In this context, therefore, this chapter presents a new technique for designing cosine-modulated filter banks, based on minor component analysis (MCA), and optimized fractional derivative constrained (FDCs) using different evolutionary techniques such as CS, MCS, ABC, and PSO techniques. The rest of the chapter is organized as follow: a brief overview on fractional derivative (FD) and multichannel cosine-modulated filter bank is given in Sects. 2 and 3, respectively. Section 4 describes the problem formulation of CMFB using MCA and FDCs. A brief overview of different evolutionary techniques such as CS, MCS, PSO, and ABC algorithm is presented in Sect. 5. The proposed methodology for CMFB is described in Sect. 6. Design results using the proposed method and the concluding remarks are given in Sects. 7 and 8, respectively.

2 Overview of Fractional Derivative (FD)

During the past decades, a substantial progress has been made toward fractional calculus and its applications in numerous science and engineering fields [19–34]. In the literature, three definitions of FD such as Riemann–Liouville, Grünwald–Letnikov, and Caputo have been explored. However, Grünwald–Letnikov derivative method has been most commonly exploited in signal processing applications due to low complexity, and ease of computation, defined as [19–21]:

$$D^v g(x) = \frac{d^v g(x)}{dx^v} = \lim_{\Delta \to 0} \sum_{k=0}^{\infty} \frac{(-1)^k I_k^v}{\Delta^v} g(x - k\Delta), \tag{1}$$

and the coefficient I_k^v is defined as

$$I_k^v = \frac{\Gamma(v+1)}{\Gamma(k+1)\Gamma(v-k+1)} = \begin{cases} 1, k = 0 \\ \dfrac{\left[\begin{array}{l} v(v-1)(v-2)\cdots \\ (v-k+1) \end{array}\right]}{1, 2, 3 \cdots k}, k \geq 1, \end{cases} \tag{2}$$

In Eq. (2), $\Gamma(\cdot)$ is the gamma function. Based on above definition, FD of sinusoidal functions is given as [19–21]:

$$D^v \beta_0 \sin(\omega x + \sigma) = \beta_0 \omega^v \sin\left(\omega x + \sigma + \frac{\pi}{2} v\right), \tag{3}$$

and

$$D^{v}\beta_0 \cos(\omega x + \varphi) = \beta_0 \omega^{v} \cos\left(\omega x + \varphi + \frac{\pi}{2} v\right). \tag{4}$$

A detailed discussion on fractional derivative and its applications are given in [19] and the references therein.

3 Overview of *M*-Channel Cosine-Modulated Filter Bank

The theory of *M*-channel FBs has been extensively developed and available in literature [1, 2]. In multichannel filter bank, the input signal is divided into multiple subbands so that each subband can be independently processed, as illustrated in Fig. 1. These filter banks can be further classified in three types: parallel structure, tree structured, and based on cosine modulation or modified discrete Fourier transform (MDFT). Among all *M*-channel filter banks, cosine-modulated filter banks (CMFBs) are the most frequently used filter banks in several fields such as subband processing of audio, image and video signals, and analog-to-digital converters, due to their simple design, and more realizable as compared to other filter banks [1, 2]. In this filter bank structure, only single prototype filter is to be designed, and the rest of other composing filters are generated from this filter with the aid of cosine modulation or MDFT [1, 2].

Consider a generalized architecture of an *M*-channel CMFB, where $H_0(z)$, $H_1(z), \ldots, H_{M-1}(z)$ and $F_0(z), F_1(z), \ldots, F_{M-1}(z)$ are the analysis and synthesis filters, respectively, and M stands for number of subbands in a filter bank. $x(n)$ is the input to *M*-channel FB, and $x_0(n), x_1(n), \ldots, x_{M-1}(n)$ are the outputs of analysis filters. While $v_0(n), v_1(n), \ldots., v_{M-1}(n)$ are the output signals after decimation,

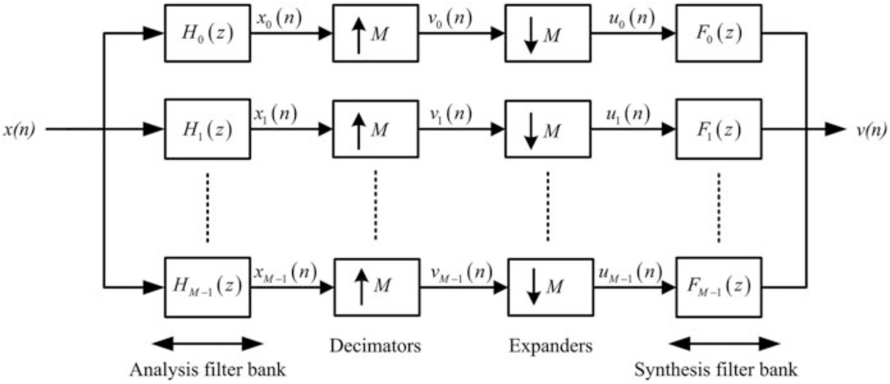


Fig. 1 A generalized block diagram of *M*-channel CMFB

and $u_0(n), u_1(n), \ldots, u_{M-1}(n)$ are the output signals of synthesis filters. The final reconstructed output of a CMFB in z-transform is given as [1, 2]

$$Y(z) = T_0(z)X(z) + \sum_{l=1}^{M-1} T_l(z)X\left(ze^{-j2\pi l/M}\right), \tag{5}$$

where $X(z)$ and $Y(z)$ are the input and output signals in z-domain, respectively, and $T_0(z)$ is a distortion transfer function, defined as

$$T_0(z) = \frac{1}{M}\sum_{k}^{M-1} F_k(z)H_k(z). \tag{6}$$

In Eq. (5), $T_l(z)$ is the aliasing error, defined as

$$T_l(z) = \frac{1}{M}\sum_{k}^{M-1} F_k(z)H_k\left(ze^{-j2\pi l/M}\right) \text{ for } l = 1, 2, \ldots, M-1. \tag{7}$$

For perfect reconstruction (PR), the distortion transfer function $T_0(z)$ is to be a delay function z^{-K} with K being an integer and $T_l(z) = 0$ for $l = 1, 2, \ldots, M-1$. If these conditions are satisfied, the reconstructed output signal is an exact replica of the original input signal with some delay, that is, $y(n) = x(n-k)$. Such multirate systems are known as perfect reconstructed (PR) FBs and are highly useful in case of lossless coding [1, 2]. If these conditions are partially satisfied, then filter bank structure is called as nearly perfect reconstructed (NPR) FB, having amplitude distortion and aliasing error. NPR CMFBs are suitable for lossy coding, where the effect of these distortions is being less than those caused by the coding distortions [1, 2].

In a CMFB, the impulse responses of the analysis filters ($h_k(n)$) and synthesis filters ($f_k(n)$) are cosine-modulated version of a prototype filter $h_0(n)$ with following transfer function:

$$H_0(z) = \sum_{n=0}^{N-1} h_0(n)z^{-n}, \tag{8}$$

where $N-1$ is the order of a prototype filter and the remaining analysis and synthesis filters are computed as

$$h_k(n) = 2h_0(n)\cos\left[\omega_k\left(n - \frac{N-1}{2}\right) + \theta_k\right], \tag{9}$$

and

$$f_k(n) = h_k(N-1-n) \tag{10}$$

with $k = 0, 1, \ldots, M-1$, and $\omega_k = \frac{(2k+1)\pi}{2M}$. In Eq. (9), $\theta_k = (-1)^k\pi/4$.

Therefore, the design problem of a CMFB is reduced to design of a single prototype filter that yields either perfect reconstruction (PR) or nearly perfect reconstruction (NPR). As it is evident from the literature, three types of errors such as phase distortion, amplitude distortion, and aliasing distortion are introduced in filter banks due to nonideal nature of analysis and synthesis filters and subsampling operation. In an NPR CMFB, phase distortion is eliminated through use of linear-phase FIR filters. Therefore, these filter banks are characterized by the error in amplitude response, given as [1, 2]

$$e_{\mathrm{am}} = \max_{\omega} \left(1 - \left|T_0\left(e^{j\omega}\right)\right|\right), \tag{11}$$

and the aliasing distortion (e_{a}), defined as

$$e_a = \max_{l,\omega} \left(T_l\left(e^{j\omega}\right)\right) \text{ for } \omega \in [0, \pi], 1 \le l \le M - 1. \tag{12}$$

From analysis of M-channel CMFBs, it is evident that perfect reconstruction is possible if Eq. (13) is satisfied, which is reduced to $|H_0(e^{j\pi/2M})| = 0.707$, when computed at $\omega_{\mathrm{c}} = \pi/2M$.

$$\left|H_0\left(e^{j\omega}\right)\right|^2 + \left|H_0\left(e^{j(\omega - 2\omega_{\mathrm{c}})}\right)\right|^2 = 1, \text{ for } 0 < \omega < 2\omega_{\mathrm{c}} \tag{13}$$

In this chapter, the filter coefficients of a prototype filter for CMFB are optimized using MCA, FDCs, and evolutionary techniques. The constraint is derived by applying the fractional derivative at prescribed frequency point ω_0 on the prototype filter response. After that, filter coefficients of a prototype filter for CMFB are derived by MCA method, based on optimized FDCs through swarm optimization techniques (SOTs).

4 Design Problem Formulation of CMFB Using MCA and FDC

From literature review, it is evident that an NPR CMFB is simply designed by developing a suitable algorithm for designing a low-pass prototype filter that can satisfy the perfect reconstruction criterion, and other design issues. Therefore, in this chapter, a new method using Type 1 polyphase decomposition is devised for optimal design of a prototype filter for CMFB using MCA and optimized FDCs, which are determined using different evolutionary techniques. For this purpose, assume a low-pass FIR prototype ($H_0(z)$), whose transfer function is defined as [1, 2]

$$H_0(z) = \sum_{n=0}^{N} h_0(n) z^{-n}, \tag{14}$$

and its polyphase realization is given as [35, 36]

$$H_0(z) = \sum_{l=0}^{P-1} z^{-l} E_l\left(z^L\right), \tag{15}$$

where $E_l(z)$ is the l^{th} polyphase component of a prototype filter, defined as

$$E_l(z) = \sum_{n=0}^{N} e_l(n) z^{-n}. \tag{16}$$

In Eq. (16), $e_l(n) = h_0(nP + l)$, and P stands for number of PCs. For simplicity and low computational complexity, in this work, P is assumed to 2. Due to this, the polyphase realization of a prototype filter is reduced to

$$H_0(z) = E_0\left(z^2\right) + z^{-1} E_1\left(z^2\right). \tag{17}$$

To have an exact linear-phase response, the impulse response of an FIR filter exhibits either symmetry or antisymmetric condition ($h(n) = \pm h(N - n)$). Here, Type 2 FIR filter is considered as a low-pass prototype filter, whose impulse response follows the symmetry condition ($h_0(n) = h_0(N - n)$). Due to this, e_0 and e_1 become mirror image to each other, and their relations in time and z-domains are, respectively, described as

$$e_0(n) = e_1\left(\frac{N-1}{2} - n\right), \tag{18}$$

and

$$E_1(z) = z^{-(N-1)/2} E_0\left(z^{-1}\right). \tag{19}$$

Using Eqs. (18) and (19), $H_0(z)$ can be rewritten as [35, 36]

$$H_0(z) = E_0\left(z^2\right) + z^{-N} E_0\left(z^2\right), \tag{20}$$

and the frequency response of a prototype filter in term of PCs is defined as

$$H_0\left(e^{j\omega}\right) = \sum_{n=0}^{(N-1)/2} e_0(n) e^{-2j\omega n} + e^{-j\omega N} \sum_{n=0}^{(N-1)/2} e_0(n) e^{2j\omega n}. \tag{21}$$

After further simplification, frequency response is reduced to

$$H_0\left(e^{j\omega}\right) = e^{-j\omega\frac{N}{2}}\left(\sum_{n=0}^{(N-1)/2} 2e_0(n)\cos\omega\left(2n-\frac{N}{2}\right)\right) = e^{-j\omega\frac{N}{2}}H_0(\omega). \tag{22}$$

In Eq. (22), $H_0(\omega)$ is the amplitude response of a prototype filter, rewritten in a matrix form as

$$H_0(\omega) = {e_0}^T \times C(\omega) = C^T(\omega) \times e_0, \tag{23}$$

where

$$e_0 = \left[e_0(0), e_0(1), e_0(2), \ldots\ldots\ldots e_0\left(\frac{N-1}{2}\right)\right]^T, \tag{24}$$

and

$$C(\omega) = 2\left[\cos\left(0-\frac{N}{2}\right)\omega,\ \cos\left(2-\frac{N}{2}\right)\omega,\ \cos\left(4-\frac{N}{2}\right)\omega, \ldots \cos\left(\frac{N}{2}-1\right)\omega\right]^T. \tag{25}$$

For designing a CMFB, in this work, a new algorithm is developed to optimize the polyphase component e_0 so that actual response $H_0(\omega)$ and ideal response $H_\mathrm{d}(\omega)$ are approximately same. For this purpose, an objective function is constructed as a mean integral square error between $H_0(\omega)$ and $H_\mathrm{d}(\omega)$, using suitable constraint on polyphase component ${e_0}^T e_0 = 1$, mathematically formulated as

$$\phi = \frac{1}{\pi}\left(\int_{\omega\in R} \left(H_0(\omega) - H_\mathrm{d}(\omega)\right)^2 d\omega\right). \tag{26}$$

In Eq. (26), $H_\mathrm{d}(\omega)$ is defined as $H_\mathrm{d}(\omega) = H(0)$ for $\omega \in [0\ \omega_\mathrm{p}]$ and 0 for $\omega \in [\omega_\mathrm{s}\ \pi\ ]$, and R is the interested band, which is $R \in [0, \omega_\mathrm{p}] \cup [\omega_\mathrm{s}, \pi]$. The objective function, defined by Eq. (26), can be rewritten as a function of polyphase component e_0,

$$\begin{aligned}\phi &= {e_0}^T\left[\frac{1}{\pi}\int_0^{\omega_\mathrm{p}} \left(C(\omega) - C(0)\right)\left(C(\omega) - C(0)\right)^T d\omega + \frac{1}{\pi}\int_{\omega_\mathrm{s}}^{\pi} \left(C(\omega)\right)\left(C(\omega)\right)^T d\omega\right] e_0 \\ &= {e_0}^T Q e_0,\end{aligned} \tag{27}$$

where $Q = Q_\mathrm{p} + Q_\mathrm{s}$. Here, Q_p and Q_s are the passband and stopband specification matrixes, respectively, and can be expressed as

$$Q_{\mathrm{p}} = \frac{1}{\pi} \int_{0}^{\omega_{\mathrm{p}}} (C(\omega) - C(0))\,(C(\omega) - C(0))^{T} d\omega, \tag{28}$$

and

$$Q_{\mathrm{s}} = \frac{1}{\pi} \int_{\omega_{\mathrm{s}}}^{\pi} (C(\omega))\,(C(\omega))^{T} d\omega. \tag{29}$$

To achieve perfect reconstruction and more accurate response of an M-channel CMFB, following constraints are imposed:

$$H_0(\omega)\big|_{\omega=\omega_{\mathrm{c}}} = 0.707 H_{\mathrm{d}}(\omega_0), \tag{30}$$

and

$$D^{u} H_0(\omega)\big|_{\omega=\omega_0} = 0. \tag{31}$$

In the above equations, ω_0 is the prescribed passband frequency; ω_{c} is the cutoff frequency, equal to $\pi/2M$; and $u \in \{u_1, u_2, u_3, \ldots, u_L\}$ is the order of fractional derivatives. Total number of constraints imposed are equal to $L + 1$. Eq. (30) corresponds to a perfect reconstruction of a M-channel CMFB, while Eq. (31) implies that fractional derivative of $H_0(\omega)$ at ω_0 becomes zero, which increases the flatness or accuracy of $H_0(\omega)$ at ω_0. The FD of $H_0(\omega)$ is computed as [19]:

$$\begin{aligned} D^{u} H_0(\omega) &= \frac{d^{u}\left(\sum_{n=0}^{(N-1)/2} 2e_0(n) \cos \omega \left(2n - \frac{N}{2}\right)\right)}{d\omega^{u}} \\ &= \sum_{n=0}^{\frac{N-1}{2}} 2e_0(n) \times \frac{d^{u} \cos \omega \left(2n - \frac{N}{2}\right) \omega}{d\omega^{u}} = {e_0}^{T} c(\omega, u), \end{aligned} \tag{32}$$

where vector $c(\omega, u)$ is defined as

$$c(\omega, u) = \begin{bmatrix} 2 \times (2 \times 0 - (N/2))^{u} \cos\left((2 \times 0 - (N/2))\omega + (\pi u)/2\right) \\ 2 \times (2 \times 1 - (N/2))^{u} \cos\left((2 \times 1 - (N/2))\omega + (\pi u)/2\right) \\ 2 \times (2 \times 2 - (N/2))^{u} \cos\left((2 \times 2 - (N/2))\omega + (\pi u)/2\right) \\ . \\ . \\ . \\ 2 \times (2 \times ((N-1)/2) - (N/2))^{u} \cos\left((2 \times ((N-1)/2) - (N/2))\omega + (\pi u)/2\right) \end{bmatrix}. \tag{33}$$

The constraints, given by Eqs. (30) and (31), can be rewritten in more compact form using matrix representation defined as

$$C^T(\omega_c)e_0 = 0.707H_d(\omega_0), \tag{34}$$

and

$$c^T(\omega_0, u_k)e_0 = 0. \tag{35}$$

Above equations can be combined as

$$S \times e_0 = g, \tag{36}$$

where

$$S = \begin{bmatrix} C^T(\omega_c) \\ c^T(\omega_0, u_1) \\ \vdots \\ c^T(\omega_0, u_L) \end{bmatrix} \text{and } \; g = \begin{bmatrix} 0.707H_d(\omega_0) \\ D^{u_1}H_d(\omega)|_{\omega=\omega_0} \\ \vdots \\ D^{u_L}H_d(\omega)|_{\omega=\omega_0} \end{bmatrix} \tag{37}$$

Now, design problem of a low-pass prototype filter for CMFB is diminished to

$$\text{Optimize : } \; \phi(e_0) = {e_0}^T Q e_0, \text{subject to } \; {e_0}^T e_0 = 1 \text{ and S} \times e_0 = g. \tag{38}$$

For solving above design problem, Eq. (36) is modified using $H_0(\omega)|_{\omega=\omega_0} = H_d(\omega_0)$, given as

$$S \times e_0 = g \times \frac{H_0(\omega_0)}{H_d(\omega_0)} = g \times \frac{C^T(\omega_0) \times e_0}{H_d(\omega_0)}, \tag{39}$$

and further can be rewritten as

$$\overset{\wedge}{\mathrm{S}} \times e_0 = 0, \tag{40}$$

where

$$\hat{S} = S - \frac{g \times C^T(\omega_0)}{H_d(\omega_0)}. \tag{41}$$

Using Eq. (40), the design problem given in Eq. (38), is reduced to

$$\mathit{Optimize} : \phi(e_0) = {e_0}^T Q e_0, \; \text{ Subject to } {e_0}^T e_0 = 1 \text{ and } \hat{S} \times e_0 = 0 \tag{42}$$

In Eq. (42), $\hat{S} \times e_0 = 0$ can be rewritten as $e_0 = B \times r$, where columns of B form an orthonormal basis of the null space of a matrix $\hat{S}$ [37, 38]. Due to this, Eq. (42) is modified as

$$\textit{Optimize} \quad \phi(e_0) = r^T Q_{\mathrm{B}} r, \ \ \text{Subject to } r^T r = 1, \tag{43}$$

where $Q_{\mathrm{B}} = B^T Q B$ contains FDCs and the optimal solution r_{opt} is a Eigen vector of a matrix Q_{B} corresponding to smallest Eigen value and the optimal PCs are given by

$$e_{0\mathrm{opt}} = B r_{\mathrm{opt}}. \tag{44}$$

By using QR decomposition, orthonormal basis of the null space of matrix $\hat{S}$ is computed. For solving the above optimization problem, a minor component analysis-based neural learning is employed.

4.1 Minor Component Analysis (MCA)-Based Learning Algorithm

If $x(t) \in R^M$ is an input, $y(t)$ is the output, and $w(t)$ is a weight vector of neurons of a neural network illustrated in Fig. 2, then input/output relation is defined as [33, 34]

$$y(t) = w^T(t)x(t) = x^T(t)w(t) = \sum_{i=1}^{M} w_i(t)x_i(t). \tag{45}$$

In the proposed work, MCA-based neural learning is exploited to compute the Eigen vector or weight vector, corresponding to the smallest Eigen value of Q_{B}. For this purpose, the output power of a neural network model is considered as an objective function, defined as [33, 34]

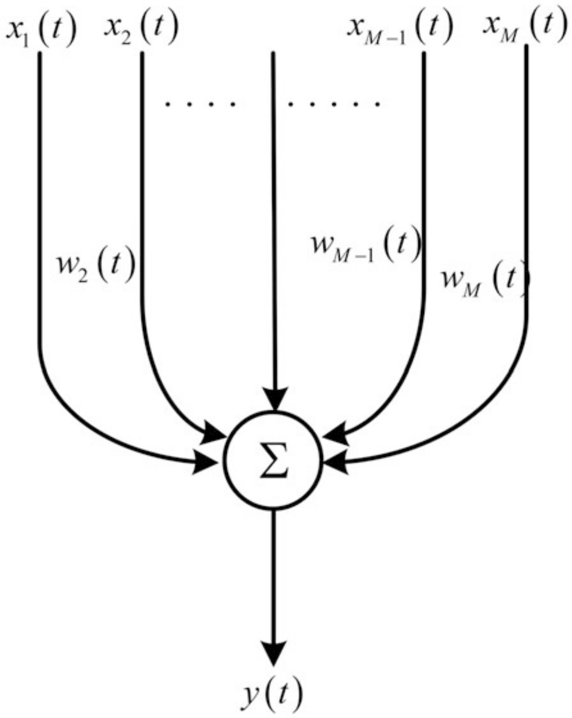
$$J(w(t)) = \frac{1}{2}E\left[y^2(t)\right] + \frac{1}{2}\lambda\left(w^T(t)w(t) - 1\right), \tag{46}$$

where $E[y^2(t)]$ is the power of neuron's output and λ is the Lagrange multiplier exploited so that $w^T(t)w(t) = 1$. For solving optimization problem, $J(w(t))$ is differentiated with respect to $w(t)$, given as

$$\partial J/\partial(w(t)) = \frac{1}{2}E[y(t)x(t)] + \frac{1}{2}\lambda w(t), \tag{47}$$

and this can be further simplified for computing optimal value of λ, given as [33, 34]

Fig. 2 Basic diagram of a linear neural network [33, 34]



$$w^T(t)\partial J/\partial(w(t)) = E\left[y^2(t)\right] + \lambda - \sigma\left(w^T(t)w(t) - 1\right), \tag{48}$$

where σ is an arbitrary constant and the optimal value of λ is given as

$$\lambda_o = -E\left[y^2(t)\right] + \sigma\left(w^T(t)w(t) - 1\right). \tag{49}$$

Computational complexity of MCA algorithm can be further enhanced by formulizing neural learning algorithm, given as

$$\begin{aligned} dw(t)/dt &= -(\partial J/\partial w(t)) \\ &= -E[y(t)x(t) - y^2(t)w(t)] - \sigma(w^T(t)w(t) - 1)w(t). \end{aligned} \tag{50}$$

Above neural learning rule can be further simplified in discrete time domain with very small time period (T), defined as $\Delta w/T$, where $\Delta w = w((k+1)T) - w(kT)$. Therefore, Eq. (50) is modified as

$$\Delta w = -\eta E\left[yx - y^2 w\right] - \eta\sigma\left(w^T w - 1\right)w, \tag{51}$$

where η is the learning step size, which decides the convergence rate of MCA algorithm.

For designing cosine-modulated filter bank, the input vector x and weight vector w of a neural model can be considered as the frequency specification ($BC(\omega)$) of a prototype filter, and polyphase component equivalent (r), respectively, defined as

$$x_i(k) = \begin{cases} {B_i}^T(C_i(\omega) - C_i(0)) & \omega \text{ in passband} \\ {B_i}^T C_i(\omega) & \omega \text{ in stopband} \end{cases}, \tag{52}$$

and

$$w_i(k) = r_i(k). \tag{53}$$

An equivalent relationship between a linear neural model and a prototype filter is given as

$$y(k) = H(\omega) - H_{\mathrm{d}}(\omega), \tag{54}$$

and the neural output with respect to frequency response is defined as

$$y(k) = \begin{cases} r^T(k)B(C(\omega) - C(0)) & \omega \text{ in passband} \\ r^T(k)B(C(\omega)) & \omega \text{ in stopband} \end{cases}. \tag{55}$$

As in a linear neural network model, neuron's output power is same as an objective function is to be optimized. Therefore, neuron's output power is computed as

$$E\left[y^2(k)\right] = w^T(k)E\left[x(k)x^T(k)\right]w(k) \tag{56}$$

and can be further simplified as

$$E\left[y^2(k)\right] = r^T B^T \begin{bmatrix} \frac{1}{\pi}\int\limits_0^{\omega_{\mathrm{p}}} (C(\omega) - C(0))\,(C(\omega) - C(0))^T d\omega \\ + \frac{1}{\pi}\int\limits_{\omega_{\mathrm{s}}}^{\pi} (C(\omega))\,(C(\omega))^T d\omega \end{bmatrix} Br \tag{57}$$

$$E\left[x(k)x^T(k)\right] = Q_{\mathrm{B}} = B^T Q B \tag{58}$$

is a covariance matrix of a neural network system and contains FDCs, where $E[x] = \int x \, d\omega$. Now, MCA-based neural learning rule derived in Eq. (51) can be exploited for determining r, corresponding to polyphase component (e_0). For next iteration, updating equation is

$$r(k+1) = r(k) + \mu\left\{-\left[Q_{\mathrm{B}} r(k) - \left(r^T(k) Q_{\mathrm{B}} r(k)\right) r(k)\right] - \sigma\left(r^T(k) r(k) - 1\right) r(k)\right\}, \tag{59}$$

and the stopping criterion is set to $\|r(k+1) - r(k)\| \leq \xi$, where $\xi = e - 10$ and $\eta = 0.002$ [33, 34]. When this is satisfied, the weight vector is conversed to the Eigen vector, corresponding to smallest Eigen value of a matrix Q_{B}, which is the optimal filter coefficients, obtained similar to Eigen filter design approach. Finally, MCA-optimized polyphase component is computed as

$$e_{0\mathrm{opt}}(k) = B \times r_{\mathrm{opt}}(k). \tag{60}$$

During MCA neural learning algorithm, FDC matrix S is not optimized. So for optimizing, several swarm-based techniques such as CS, MCS, PSO, and ABC algorithms are exploited by considering an objective function given by Eq. (61), which is the peak reconstruction error (PRE) in the filter bank.

$$\psi(\{u_1, u_2, u_3 \ldots . u_L\}) = \mathrm{PRE} = \max\left\{10\log\left(\left|H_0\left(e^{j\omega}\right)\right|^2 + \left|H_0\left(e^{j(\omega - 2\omega_{\mathrm{c}})}\right)\right|^2\right)\right\}. \tag{61}$$

After optimization, the optimized constraint matrix is S_{S}, and corresponding B matrix is B_{s}. Finally, optimized polyphase component $(e_0)_{\mathrm{Sopt}}$ is computed as

$$(e_0)_{\mathrm{Sopt}}(k) = B_{\mathrm{s}} \times r_{\mathrm{Sopt}}(k), \tag{62}$$

which can be used for determining a prototype filter response, and other composing filters response are derived using cosine modulation.

5 Overview on Employed Swarm Optimization Techniques

In this section, a brief overview on swarm optimization techniques is presented, which are employed in this work.

5.1 Cuckoo Search Algorithm

Cuckoo search (CS) algorithm [39] is a very efficient global search algorithm based on coercion of progeny parasitic demeanor of some cuckoo spices in conjunction with Levy flight demeanor [39]. Levy flights are random move, whose step size and direction is decided by Levy flight distribution [39]. Cuckoo bird's reproduction approach is one of the most attacking as compared to other bird species [39]. Intraspecific brood parasitism, cooperative breeding, and nest takeover are three basic progeny parasitism behaviors. Cuckoos can directly struggle from the host birds. Either host birds can chuck the unknown eggs away from the nest or they discard the nest and make another one. The *Tapera*, a genus of cuckoo bird, engages with such evolution; they can emulate the properties (color and pattern) of the eggs of selected few hosts [39]. Cuckoo chick can also emulate the call of host chicks to gain access to more nourishing possibilities [39]. There are three basic assumptions of CS algorithm [39]:

(i) At a time, each cuckoo lays one egg and drops it in a conjecturally selected nest.
(ii) The best nests will be passed to the next generation.
(iii) The number of existing host nests are constant, and discovering probability of host bird for cuckoos egg is $p_a \in [0, 1]$.

Now, there is a chance that host bird can totally dumb the nest and construct a new nest. For ease, it is assumed that the last supposition can be approximated by a probability p_a of the n nest and is swapped by new nests (with random new solutions). Each egg in a nest denotes a solution, and a cuckoo egg denotes a new solution, then plan is to exploit new and potentially better solutions (cuckoos) to replace an unfit solution in the nests. For producing new solution $x_i(t+1)$ for cuckoo i, a Levy flight is executed using

$$x_i(t+1) = x_i(t) + \alpha \oplus \text{Levy}(\lambda), \tag{63}$$

where α is the step size, which is associated to the dimension of optimization problem. Levy(λ) is a Levy flight distribution. Cuckoo search is basically regulated by the following factors: number of host nests (NS), probability (p_{a}), number of iterations, and step size (α). A detailed discussion on CS and Levy flight distribution is available in [39] and the references therein.

5.2 Modified Cuckoo Search Algorithm

MCS technique is amended version of CS algorithm with two basic amendments [40]. The first amendment is that the Levy flight step size α becomes variable. The value of α in MCS decreases as the number of generation (G) increases. Due to the amendment, search becomes more localized. Therefore, the adapted Levy flight step size α_m is expressed by

$$\alpha_{\text{m}} = \alpha/\sqrt{G}, \tag{64}$$

where α and G are Levy flight step size used in CS and generation number, respectively. The knowledge exchange between the eggs is introduced as a second amendment so that convergence of algorithm accelerated. Unlike CS, a group of top eggs formatted which is made by fraction of eggs have best fitness values in MCS. In the group for each top egg, a reference egg is selected as random manner, and new egg is generated by on the line connecting the top egg and reference egg. The new egg position along the line is calculated by using golden ratio $\varphi = \left(1 + \sqrt{5}\right)/2$ [40], such that the position of new egg is nearer to the egg with best fitness value. The new egg is generated at the mid of both egg, if both have same fitness value. There is a chance that, in the selection process, same egg is picked up twice. At that point, a local Lévy flight search is executed from randomly selected nest with step size $\alpha_{\mathrm{m}} = \alpha/G^2$. So, there are two parameters, fraction of nests to be abandoned (p_{a1}) and fraction of nests to make up the top nests ($p_{a2} = 1 - p_{a1}$), which need to be adjusted in MCS. A detailed knowledge of MCS can be found in [40] and the references therein.

5.3 *Artificial Bee Colony (ABC)*

The food procuring process of honey bees is the main motivation of artificial bee colony (ABC) [41] optimization. For given optimization problem which is solved by ABC, the possible solutions are represented by the food source positions. In the begging, the positions of food source are randomly generated in the virtual space. In ABC, there are three types of honey bees employed for search, which are employed bees, onlooker bees, and scout bees. Initially, all the presented bees are equally divided into employed and onlooker bees. The nectar amount of food source (solutions) is calculated by placed employed and onlooker bees on food source. Scout bees walk around the whole colony devoid of any guidance. The vicinal food position is calculated by

$$\mathrm{SP}_i(c+1) = \mathrm{SP}_i(c) + \chi_i(\mathrm{SP}_i(c) - \mathrm{SP}_k(c)), \tag{65}$$

where χ_i is randomly produced from interval [−1,+1], c is the cycle, and k is randomly produced index, which is different from i. If the nectar amount or fitness value $F(\mathrm{SP}_i(c+1))$ is higher than $F(\mathrm{SP}_i(c))$, then employed bee stores $\mathrm{SP}_i(c+1)$ and shares her information with onlooker bees, and the position $\mathrm{SP}_i(c)$ of food source i is replaced by $\mathrm{SP}_i(c+1)$; otherwise, $\mathrm{SP}_i(c)$ is kept as it is. More precisely, the probability of i^{th} source selected with $F(\mathrm{SP}_i(c))$ fitness value is given by [41]

$$\mathrm{Prob}_i = \frac{F(\mathrm{SP}_i)}{\sum_{k=1}^{\mathrm{FS}} F(\mathrm{SP}_k)}. \tag{66}$$

Every food source has only one employed bee (employed bees = food sources). If the position $\mathrm{SP}_i(c)$ of food source i cannot be improved within predefined number of cycles called *limit*, then the food source i is deselected, and employed bee becomes scout bee. The scout bee begins search for a new food source randomly, and after finding suitable one, the new position is admitted to be $\mathrm{SP}_i(c+1)$. All three steps conducted by the employed bees, onlooker bees, and scout bees are repeated until the termination criteria are satisfied. It can be concluded that ABC algorithm is controlled by three parameters: number of employed bees, iteration number, and limit value. A detailed discussion on ABC optimization algorithm is given in [41].

5.4 Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) [41] is encouraged by the unified group demeanors of living species that show complicated social behaviors. Unlike GA and DE algorithms, there are no crossover and mutation operators in PSO. In PSO algorithm, for searching new location of particles, both cognitive factors of relative particle and social factor generated by the swarm are used. Due to these factors PSO is effectively used for finding global optimal solution. The possible solutions are termed as particle position and collection of possible solutions termed as swarm, respectively. The PSO is administrated by two basic updating equations for particle position i, first is velocity, whose updating equation is defined by

$$V_i(k+1) = w \times V_i(k) + c_1\phi_1(P_{\mathrm{ibest}}(k) - P_i(k)) + c_2\phi_2(G_{\mathrm{best}}(k) - P_i(k)), \quad (67)$$

and second is position updating equation, defined by

$$P_i(k+1) = P_i(k) + V_i(k+1), \quad (68)$$

where w is the inertia weight factor and vary between 0 and 1 linearly; c_1 and c_2 are the cognitive and social acceleration factors, respectively; ϕ_1 and ϕ_2 are the random numbers, which are uniformly distributed in between 0 and 1; P_{ibest} and G_{best} are the updated velocity and position in next iteration called particle best and global best, respectively; and V_i is the velocity of particle. The next velocity $V_i(k+1)$ and position $P_i(k+1)$ of i^{th} particle is measured by Eqs. (67) and (68), respectively. Basically, the particle position P_i represents one possible solution of optimization problem. At every iteration, the objective function (fitness function) is measured by the position vector $P_i(k)$. The position vector corresponding to best fitness is known as *pbest*, and the overall best outcome of all the particles in population is called *gbest*; PSO depends on initial values of the control parameters (w, c_1 and $c_{2,}\phi_1$ and ϕ_2), the size of swarm value (s), and the maximum iteration number. Detailed description of PSO is given in [42].

6 Proposed Method for CMFB Based on MCA and Optimized FDCs

In the proposed work, swarm intelligent based optimization algorithms such as CS, MCS, ABC and PSO are exploited to determine the optimized FDC by minimizing Eq. (61). Here, x_i for CS and MCS, P_i for PSO and SP_i for ABC are commonly replaced by solution of problem (sol_i). The following steps are to be taken for designing M-channel CMFB:

Step 1: Specify the design parameters of a prototype filter for CMFB such as length of filter (N), passband edge frequency (ω_{p}), and stopband edge frequency(ω_{s}).
Step 2: State the controlling parameters of CS, MCS, PSO, and ABC algorithm.
Step 3: Create initial population of host nests $\mathrm{sol}_i(t)$ having dimension equal to$u(L)$.
Step 4: Compute the fitness value ψ for each $\mathrm{sol}_i(t)$ using Eqs. (60) and (61).
Step 5: Determine a new solution $\mathrm{sol}_i(t+1)$ using different SOTs updating equations.
Step 6: Calculate the fitness value $\psi(\mathrm{sol}_i(t+1))$ at $\mathrm{sol}_i(t+1)$ using Eqs. (60) and (61).
Step 7: Check the fitness of new solution $\psi(\mathrm{sol}_i(t+1))$ with other possible solutions fitness $\psi(\mathrm{sol}_i(t))$. If $\psi(\mathrm{sol}_i(t+1)) < \psi(\mathrm{sol}_i(t))$, use different criteria depending upon the employed SOT to select best solution denoted by $\mathrm{best}(\mathrm{sol}_i(t+1))$ [39–41]. If $\psi(\mathrm{sol}_i(t+1)) > \psi(\mathrm{sol}_i(t))$, check whether maximum number of iterations is reached to specified limit. If yes, optimization process gets terminated, and again start process. If no, go to step 5, and follow the next steps.
Step 8: Check $\psi(\mathrm{best}(\mathrm{sol}_i(t+1))) < \mathrm{Tol}$. If yes, get optimized constraints matrixes B_{s} and r_{Sopt} using QR decomposition, and optimized PCs $(e_0)_{\mathrm{Sopt}}$ using Eq. (62) corresponding to $\mathrm{best}(\mathrm{sol}_i(t+1))$. Finally, design filter bank using Eqs. (9 and 10). If no, check whether maximum number of iterations is reached. If yes, optimization process gets terminated, and again start process. If no, go to step 5 and follow the next steps.

The flowchart for proposed method is depicted in Fig. 3, and the values of controlling parameters for CS, MCS, ABC, and PSO are taken from [25, 27, 28].

7 Result and Discussion

In this section, several simulated design examples for M-channel CMFB using the proposed method are presented. The effectiveness and efficiency of this method is assessed by:

Amplitude distortion:

$$e_{\mathrm{am}} = \max_{\omega}\left(1 - \left|T_0\left(e^{j\omega}\right)\right|\right), \tag{69}$$

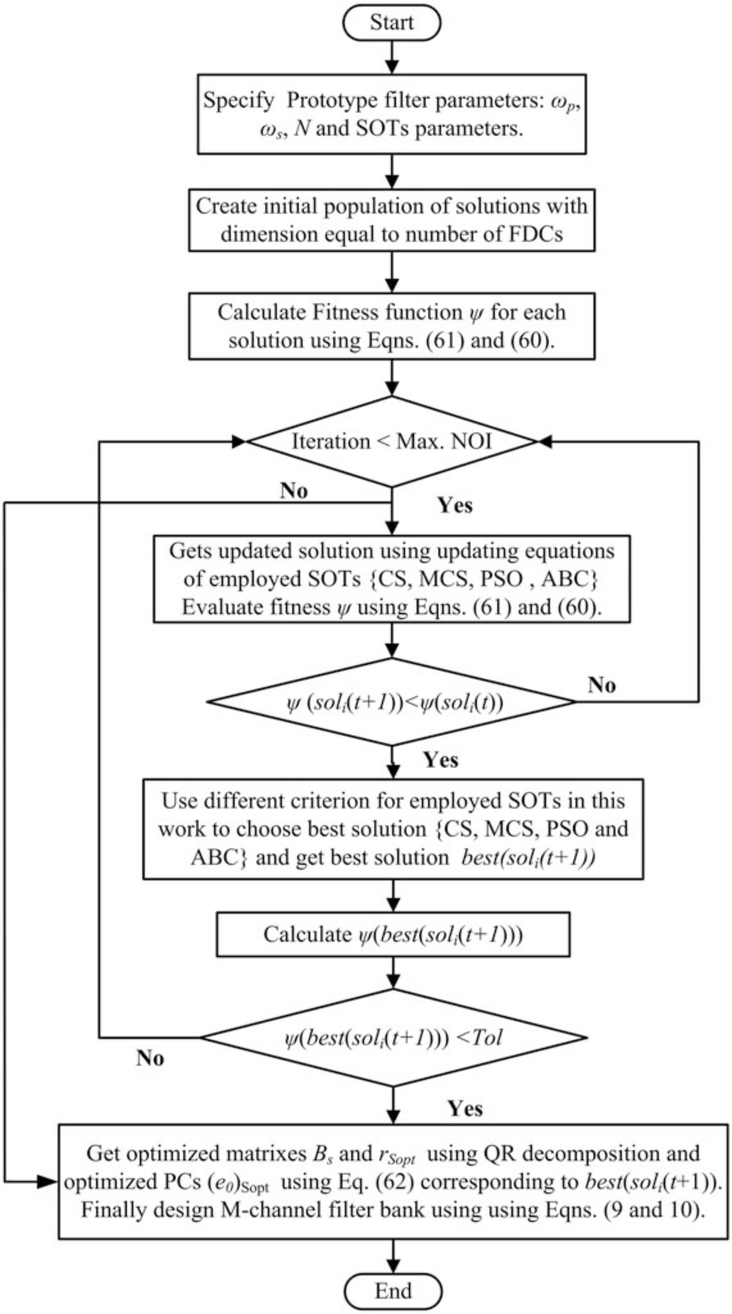


Fig. 3 Proposed methodology for desiging CMFB based on MCA and SOT

Aliasing distortion:

$$e_{\mathrm{a}} = \max_{l,\omega} \left(T_l\left(e^{j\omega}\right)\right) \text{ for } \omega \in [0, \pi], 1 \leq l \leq M - 1, \tag{70}$$

Peak reconstruction error (PRE) in dB:

$$\mathrm{PRE} = \max_{\omega} \left\{ 10\log\left(\sum_{i=1}^{M} \left|H_i\left(e^{j\omega}\right)\right|^2\right)\right\}, \tag{71}$$

Stopband attenuation:

$$A_{\mathrm{s}} = -20\log_{10}|H_0(\omega)|, \quad \text{at } \omega = \omega_{\mathrm{s}}, \tag{72}$$

Passband Error:

$$\phi_{\mathrm{p}} = \frac{1}{\pi} \int_0^{\omega_{\mathrm{p}}} (H_0(\omega) - 1)^2 d\omega, \tag{73}$$

Stopband Error:

$$\phi_{\mathrm{s}} = \frac{1}{\pi} \int_{\omega_{\mathrm{s}}}^{\pi} (H_0(\omega) - 0)^2 d\omega, \tag{74}$$

Transition band Error:

$$\phi_{\mathrm{t}} = (H_0(\omega) - 0.707H_0(0))^2 \text{at } \ \omega = \omega_{\mathrm{c}} = \pi/2M \tag{75}$$

and computational time (CPU time in seconds for each iteration).

7.1 Design Examples

To examine the performance of proposed method, 8-, 16-, and 32-channel CMFBs are designed using this methodology. Design parameters for these filter banks are $N = 160$, $\omega_{\mathrm{p}} = 0.4\pi/8$, and $\omega_{\mathrm{s}} = 0.6\pi/8$ for $M = 8$; $N = 256$, $\omega_{\mathrm{p}} = 0.4\pi/16$, $\omega_{\mathrm{p}} = 0.4\pi/16$, and $\omega_{\mathrm{s}} = 0.6\pi/16$ for $M = 16$; and $N = 512$, $\omega_{\mathrm{p}} = 0.4\pi/32$, and $\omega_{\mathrm{s}} = 0.6\pi/32$ for $M = 32$. Initial guess value of polyphase component is taken $e_0(n) = [0, 0, 0, 0, \ldots, 1]$. The best design results obtained for CMFB using proposed method based on CS, MCS, ABC, and PSO for 8 channels are

$$\begin{aligned}\mathrm{CS}: \phi_{\mathrm{p}} &= 1.69e - 7, \phi_{\mathrm{s}} = 2.47e - 5, \phi_{\mathrm{t}} = 1.23e - 32, \mathrm{PRE} = 0.0011\mathrm{dB}, \\ e_{\mathrm{am}} &= 6.81\text{e-}4, \quad e_{\mathrm{a}} = 5.44\text{e-}9, \text{and } \ A_{\mathrm{s}} = 21.08\mathrm{dB};\end{aligned}$$

$$\begin{aligned} MCS: {} & \phi_{\mathrm{p}} = 9.06e-7, \phi_{\mathrm{s}} = 3.73e-5, \phi_{\mathrm{t}} = 3.08e-31, \mathrm{PRE} = 0.0034\mathrm{dB}, \\ & e_{\mathrm{am}} = 0.0089,\ e_{\mathrm{a}} = 8.14e-7, \text{and } A_{\mathrm{s}} = 18.62\mathrm{dB}; \\ ABC: {} & \phi_{\mathrm{p}} = 1.46e-6, \phi_{\mathrm{s}} = 5.23e-5, \phi_{\mathrm{t}} = 2.39e-21, \mathrm{PRE} = 0.0037\mathrm{dB}, \\ & e_{\mathrm{am}} = 0.0079, e_{\mathrm{a}} = 7.79e-7, \text{and } A_{\mathrm{s}} = 17.55\mathrm{dB}; \\ \mathrm{PSO}: {} & \phi_{\mathrm{p}} = 2.29e-6, \phi_{\mathrm{s}} = 1.51e-5, \phi_{\mathrm{t}} = 1.23e-32, \mathrm{PRE} = 0.0061\mathrm{dB}, \\ & e_{\mathrm{am}} = 0.0020,\ e_{\mathrm{a}} = 2.10e-9, \text{and } A_{\mathrm{s}} = 21.46\mathrm{dB}. \end{aligned}$$

The best results obtained in case16-channel CMFB using proposed method are

$$\begin{aligned} \mathrm{CS}: {} & \phi_{\mathrm{p}} = 1.96e-6, \phi_{\mathrm{s}} = 7.06e-5, \phi_{\mathrm{t}} = 7.36e-27, \mathrm{PRE} = 0.0029\mathrm{dB}, \\ & e_{\mathrm{am}} = 3.96e-4,\ e_{\mathrm{a}} = 6.07e-7, \text{and } A_{\mathrm{s}} = 14.22\mathrm{dB}; \\ \mathrm{MCS}: {} & \phi_{\mathrm{p}} = 2.02e-6, \phi_{\mathrm{s}} = 7.12e-5, \phi_{\mathrm{t}} = 7.36e-27, \mathrm{PRE} = 0.0034\mathrm{dB}, \\ & e_{\mathrm{am}} = 4.29e-4,\ e_{\mathrm{a}} = 5.78e-7, \text{and } A_{\mathrm{s}} = 14.30\mathrm{dB}; \\ \mathrm{ABC}: {} & \phi_{\mathrm{p}} = 4.60e-6, \phi_{\mathrm{s}} = 1.13e-4, \phi_{\mathrm{t}} = 1.81e-24, \mathrm{PRE} = 0.0053\mathrm{dB}, \\ & e_{\mathrm{am}} = 0.0014,\ e_{\mathrm{a}} = 5.92e-7, \text{and } A_{\mathrm{s}} = 12.48\mathrm{dB}, \\ \mathrm{PSO}: {} & \phi_{\mathrm{p}} = 1.20e-6, \phi_{\mathrm{s}} = 7.31e-5, \phi_{\mathrm{t}} = 2.81e-28, \mathrm{PRE} = 0.0024\mathrm{dB}, \\ & e_{\mathrm{am}} = 7.79e-4,\ e_{\mathrm{a}} = 4.51e-7, \text{and } A_{\mathrm{s}} = 14.28\mathrm{dB}, \end{aligned}$$

and in case of 32-channel CMFB are

$$\begin{aligned} \mathrm{CS}: {} & \phi_{\mathrm{p}} = 3.91e-7, \phi_{\mathrm{s}} = 2.75e-5, \phi_{\mathrm{t}} = 1.10e-31, \mathrm{PRE} = 0.0019\mathrm{dB}, \\ & e_{\mathrm{am}} = 5.15e-4,\ e_{\mathrm{a}} = 1.45e-8, \text{and } A_{\mathrm{s}} = 16.48\mathrm{dB}, \\ \mathrm{MCS}: {} & \phi_{\mathrm{p}} = 9.56e-7, \phi_{\mathrm{s}} = 5.07e-5, \phi_{\mathrm{t}} = 6.56e-29, \mathrm{PRE} = 0.0043\mathrm{dB}, \\ & e_{\mathrm{am}} = 0.0037,\ e_{\mathrm{a}} = 1.66e-7, \text{and } A_{\mathrm{s}} = 12.93\mathrm{dB}, \\ \mathrm{ABC}: {} & \phi_{\mathrm{p}} = 2.11e-6, \phi_{\mathrm{s}} = 7.22e-5, \phi_{\mathrm{t}} = 2.62e-23, \mathrm{PRE} = 0.0048\mathrm{dB}, \\ & e_{\mathrm{am}} = 1.33e-4,\ e_{\mathrm{a}} = 6.9e-8, \text{and } A_{\mathrm{s}} = 13.82\mathrm{dB}. \\ \mathrm{PSO}: {} & \phi_{\mathrm{p}} = 2.31e-6, \phi_{\mathrm{s}} = 3.77e-5, \phi_{\mathrm{t}} = 1.08e-27, \mathrm{PRE} = 0.0043\mathrm{dB}, \\ & e_{\mathrm{am}} = 0.0011,\ e_{\mathrm{a}} = 1.57e-7, \text{and } A_{\mathrm{s}} = 14.08\mathrm{dB}. \end{aligned}$$

Performance for different channels using the proposed method based on CS, MCS, ABC, and PSO is also summarized in Tables 1a, 1b, 1c, 1d, 1e, 1f, 2a, 2b, 2c, 2d, 2e, 2f, 3a, 3b, 3c, 3d, 3e, 3f, 4a, 4b, 4c, 4d, 4e, and 4f (Table 1 for CS, Table 2 for MCS, Table 3 for ABC, and Table 4 for PSO). The frequency response obtained for 32-channel CMFB is shown in Figs. 4, 5, 6, and 7 for CS, MCS, ABC, and PSO, respectively.

It is evident from Tables 1a, 1b, 1c, 1d, 1e, 1f, 2a, 2b, 2c, 2d, 2e, 2f, 3a, 3b, 3c, 3d, 3e, 3f, 4a, 4b, 4c, 4d, 4e, and 4f that the least value of error and distortions parameters like *PRE*, ϕ_{p}, ϕ_{s}, ϕ_{t}, e_{a}, and e_{am} are resulted from the proposed methodology. Their respective values obtained in the case of CS-optimized FDCs are 0.00109, 1.69e-7, 2.48e-5, 1.23e-32, 1.87e-9, and 6.81e-4 for 8-channel CMFB; 0.0024, 1.07e-6, 3.23e-5, 1.10e-31, 3.26e-8, and 3.96e-4 for 16-channel CMFB; and 0.0019, 3.91e-7, 1.73e-5, 1.23e-32, 8.47e-9, and 2.99e-4 for 32-channel CMFB.

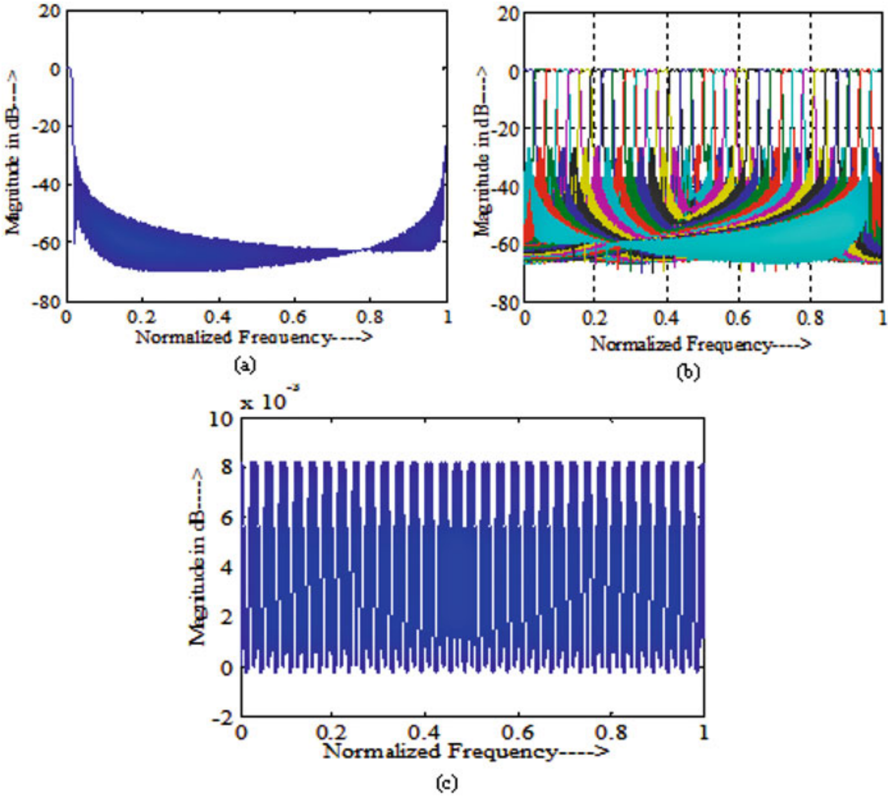


Fig. 4 CS-based 32-channel CMFB design with $N + 1 = 512$: (**a**) Prototype filter amplitude response in dB; (**b**) amplitude response of analysis filters in dB; (**c**) reconstruction error in dB

Table 1a Performance parameters obtained using proposed method based on CS algorithm for 8-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_s	ϕ_p	ϕ_S	ϕ_t	e_a	e_{am}
1	0.00474	21.18	8.75e-7	1.74e-5	1.23e-32	7.37e-9	0.0131
2	0.00247	20.96	1.05e-6	2.20e-5	1.23e-32	1.32e-8	0.0074
3	0.00109	21.08	1.69e-7	2.48e-5	1.23e-32	5.44e-9	6.81e-4
4	0.00136	21.09	1.89e-7	2.50e-5	1.59e-29	1.87e-9	0.0026
5	0.00367	21.34	2.71e-6	3.79e-5	1.10e-31	7.72e-9	0.0068
6	0.00211	20.55	3.48e-7	2.27e-5	4.44e-30	1.80e-7	0.0079
7	0.00140	19.61	6.22e-7	3.89e-5	2.08e-30	7.12e-7	0.0015
8	0.00150	18.43	3.80e-7	4.87e-5	3.20e-29	1.13e-6	0.0044
9	0.00690	18.85	3.16e-6	3.46e-5	1.64e-17	7.44e-7	0.0210
10	0.00600	18.23	2.54e-6	3.81e-5	8.67e-17	3.98e-9	0.0200

Table 1b The optimized FDCs in case 8-channel CMFB for $N + 1 = 160$ and $M = 8$ using CS algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	6.13									
2	8.65	1.21								
3	4.18	6.41	1.25							
4	4.12	1.18	6.43	22.94						
5	15.40	8.11	13.49	3.58	1.30					
6	8.42	3.92	68.94	40.92	32.90	1.06				
7	73.66	1.08	5.17	2.48	49.09	11.24	63.16			
8	7.61	80.71	43.75	10.87	1.16	77.92	10.29	47.85		
9	57.35	60.20	66.19	8.78	68.76	66.19	61.95	57.20	18.73	
10	2.90	52.23	68.87	61.69	43.64	40.87	54.18	65.25	49.13	49.14

Table 1c Performance parameters obtained using proposed method based on CS algorithm for 16-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0054	16.61	1.30e-6	3.23e-5	1.10e-31	3.26e-8	0.0166
2	0.0026	16.61	1.23e-6	4.63e-5	1.10e-31	4.69e-8	0.0016
3	0.0024	16.57	1.07e-6	3.46e-5	1.10e-31	4.61e-8	0.0014
4	0.0055	15.73	1.51e-6	4.22e-5	3.08e-31	1.98e-7	0.0158
5	0.0034	15.71	2.00e-6	4.58e-5	1.23e-30	1.98e-7	0.0016
6	0.0042	15.39	2.31e-6	5.86e-5	2.12e-27	3.40e-7	0.0058
7	0.0038	15.21	2.41e-6	6.78e-5	1.10e-31	3.58e-7	0.0014
8	0.0029	14.22	1.96e-6	7.06e-5	7.36e-27	6.07e-7	3.96e-4
9	0.0046	13.64	5.96e-6	9.76e-5	5.33e-26	6.38e-7	0.0020
10	0.0044	14.26	3.53e-6	7.27e-5	5.32e-26	6.22e-7	0.0011

Table 1d The optimized FDCs in case 8-channel CMFB for $N + 1 = 256$ and $M = 8$ using CS algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	4.28									
2	1.12	5.76								
3	3.85	1.76	40.96							
4	41.03	69.27	71.75	7.54						
5	7.51	1.60	9.05	29.12	59.71					
6	59.70	52.96	33.25	56.38	1.14	17.46				
7	18.92	65.29	69.20	1.08	45.01	68.94	12.50			
8	29.23	70.10	53.79	56.28	13.61	69.79	65.69	1.58		
9	17.60	72.15	61.02	33.10	1.30	33.00	62.76	10.35	27.38	
10	71.47	33.79	20.83	71.65	1.49	0.99	46.92	29.01	41.39	25.90

Table 1e Performance parameters obtained using proposed method based on CS algorithm for 32-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0054	16.63	6.50e-7	1.61e-5	1.23e-32	8.47e-9	0.0168
2	0.0019	16.48	3.91e-7	2.75e-5	1.10e-31	1.45e-8	5.15e-4
3	0.0023	16.57	5.30e-7	1.73e-5	4.93e-32	1.16e-8	6.95e-4
4	0.0025	15.04	7.14e-7	2.75e-5	3.08e-31	1.04e-7	3.67e-4
5	0.0034	14.34	5.94e-7	3.31e-5	5.51e-27	1.16e-7	0.0043
6	0.0041	14.95	1.21e-6	2.82e-5	3.47e-25	1.27e-7	0.0058
7	0.0062	15.53	2.02e-6	2.34e-5	9.32e-29	9.89e-8	0.0176
8	0.0051	13.96	2.83e-6	6.23e-5	5.45e-23	1.47e-7	2.99e-4
9	0.0046	14.24	3.07e-6	4.10e-5	1.00e-23	1.23e-7	6.81e-4
10	0.0053	14.74	1.60e-6	4.12e-5	3.37e-23	1.14e-7	0.0047

Table 1f The optimized FDCs in case 8-channel CMFB for $N + 1 = 512$ and $M = 32$ using CS algorithm

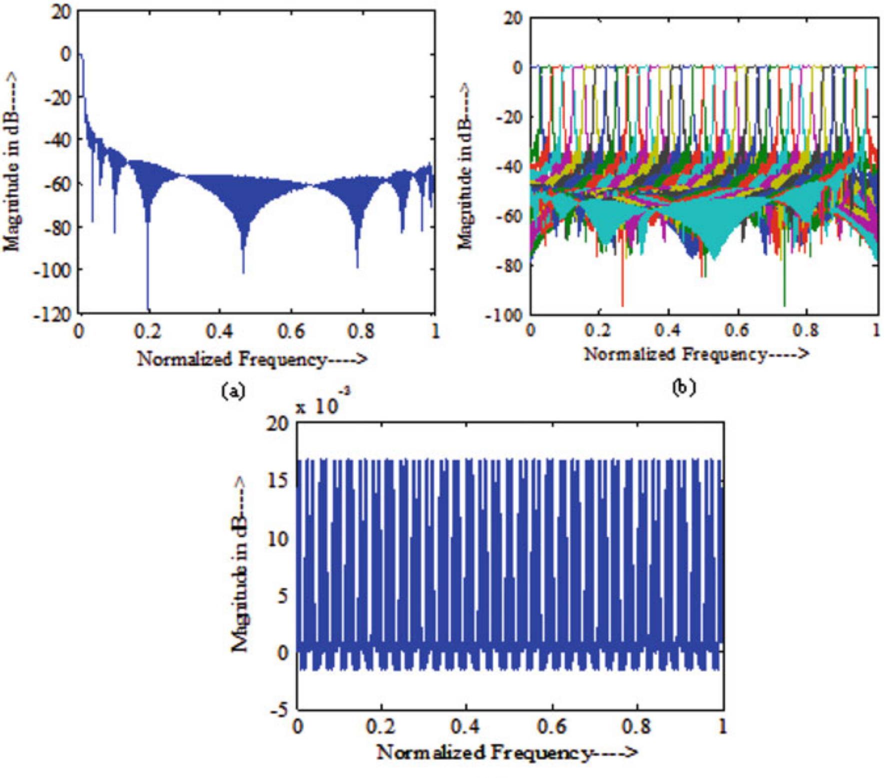
No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	4.42									
2	4.73	0.96								
3	1.78	3.66	26.99							
4	45.78	1.60	54.10	11.35						
5	3.98	22.47	10.60	8.33	5.95					
6	19.54	41.18	36.16	41.09	42.69	1.51				
7	1.44	52.38	34.82	21.79	39.26	22.70	50.84			
8	38.26	41.00	1.06	33.06	41.21	33.20	29.54	2.80		
9	6.92	55.42	36.50	50.64	36.62	48.31	36.58	1.40	48.00	
10	44.10	37.11	30.29	0.96	62.61	50.70	44.43	54.57	21.09	43.55

Table 2a Performance parameters obtained using proposed method based on MCS algorithm for 8-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0041	21.18	8.75e-7	1.74e-5	1.23e-32	7.49e-9	0.0131
2	0.0039	21.13	8.02e-7	1.76e-5	1.23e-32	1.24e-8	0.0126
3	0.0037	20.58	9.13e-7	2.09e-5	4.58e-29	1.77e-7	0.0118
4	0.0042	20.04	1.10e-6	2.40e-5	1.23e-32	4.91e-7	0.0138
5	0.0034	18.62	9.06e-7	3.73e-5	3.08e-31	8.14e-7	0.0089
6	0.0057	19.93	1.99e-6	2.45e-5	2.08e-30	5.94e-7	0.0190
7	0.0037	18.56	1.07e-6	3.93e-5	2.49e-29	8.25e-7	0.0099
8	0.0044	18.75	1.43e-6	3.60e-5	9.74e-27	8.51e-7	0.0121
9	0.0058	18.95	2.10e-6	3.28e-5	9.96e-21	8.92e-7	0.0189
10	0.0058	18.15	2.43e-6	4.09e-5	2.39e-20	8.69e-7	0.0192

Table 2b The optimized FDCs in case 8-channel CMFB for $N + 1 = 160$ and $M = 8$ using MCS algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	6.13									
2	4.09	2.00								
3	76.79	38.80	8.35							
4	40.84	71.85	9.71	7.43						
5	77.50	14.50	58.45	38.50	4.50					
6	55.72	34.84	20.30	70.88	56.27	11.35				
7	24.17	79.50	25.00	27.75	77.95	11.50	7.50			
8	34.83	74.50	50.91	40.20	22.79	2.54	40.60	27.03		
9	41.67	17.04	70.75	69.05	78.92	60.41	46.88	51.13	44.67	
10	71.97	13.03	65.16	57.24	41.21	58.27	43.66	44.10	55.60	77.48



(a)

(b)

(c)

Fig. 5 MCS-based 32-channel CMFB design with $N + 1 = 512$: (**a**) Prototype filter amplitude response in dB; (**b**) amplitude response of analysis filters in dB; (**c**) reconstruction error in dB

Table 2c Performance parameters obtained using proposed method based on MCS algorithm for 16-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0054	16.62	1.30e-6	3.23e-5	1.10e-31	3.26e-8	0.0166
2	0.0049	16.51	3.96e-6	4.12e-5	1.10e-31	5.19e-8	0.0030
3	0.0055	16.61	4.81e-6	3.73e-5	7.88e-31	5.26e-8	0.0031
4	0.0034	15.56	2.14e-6	5.15e-5	3.08e-31	2.21e-7	0.0018
5	0.0066	15.26	2.16e-6	4.87e-5	4.43e-31	3.67e-7	0.0199
6	0.0039	14.97	2.30e-6	6.64e-5	1.97e-31	4.67e-7	0.0027
7	0.0073	14.89	2.61e-6	5.44e-5	5.96e-26	4.89e-7	0.0223
8	0.0034	14.30	2.02e-6	7.12e-5	7.32e-26	5.78e-7	4.29e-4
9	0.0087	15.01	3.66e-6	5.31e-5	2.87e-21	4.51e-7	0.0284
10	0.0049	13.53	2.45e-6	8.44e-5	1.60e-18	5.62e-7	0.0017

Table 2d The optimized FDCs in case 8-channel CMFB for $N + 1 = 256$ and $M = 16$ using MCS algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	4.30									
2	1.50	4.00								
3	1.50	5.50	55.01							
4	62.50	4.50	33.05	1.50						
5	57.55	12.50	11.50	71.50	56.50					
6	1.13	59.78	32.49	32.04	16.03	47.57				
7	66.85	19.52	41.38	58.11	32.32	62.68	22.77			
8	40.14	43.91	37.50	14.50	25.50	69.50	42.50	1.50		
9	68.06	23.33	70.55	56.70	60.41	68.80	54.07	13.12	61.89	
10	10.50	48.76	60.22	2.50	25.41	65.50	35.50	73.15	67.28	64.10

Table 2e Performance parameters obtained using proposed method based on MCS algorithm for 32-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0054	16.63	6.50e-7	1.61e-5	1.23e-32	8.47e-9	0.0168
2	0.0054	16.53	6.36e-7	1.66e-5	1.23e-32	2.61e-9	0.0164
3	0.0050	16.14	1.41e-6	1.95e-5	1.10e-31	3.91e-8	0.0137
4	0.0065	14.93	1.13e-6	2.69e-5	1.10e-31	1.11e-7	0.0181
5	0.0072	15.08	1.28e-6	2.56e-5	2.27e-29	1.08e-7	0.0223
6	0.0066	14.91	1.17e-6	2.72e-5	2.64e-27	1.14e-7	0.0187
7	0.0078	13.87	1.91e-6	3.71e-5	3.26e-25	1.46e-7	0.0252
8	0.0043	12.93	9.56e-7	5.07e-5	6.56e-29	1.66e-7	0.0037
9	0.0110	12.17	3.55e-6	8.12e-5	3.09e-13	6.43e-7	0.0045
10	0.0135	12.64	7.12e-6	5.14e-5	2.20e-22	7.95e-8	0.0533

Table 2f The optimized FDCs in case 32-channel CMFB for $N + 1 = 512$ using MCS algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	4.42									
2	55.01	5.19								
3	1.50	60.50	10.50							
4	58.50	37.91	46.50	13.35						
5	14.77	45.57	40.72	37.04	49.97					
6	53.62	39.09	14.38	23.03	43.76	48.39				
7	28.50	44.81	22.50	16.83	49.84	42.50	50.50			
8	8.30	6.50	34.50	39.47	5.50	29.50	56.50	46.49		
9	08.37	26.04	59.99	37.82	55.90	53.97	50.04	27.59	54.83	
10	29.60	16.51	50.74	34.95	10.87	55.50	27.46	54.91	30.67	49.94

Table 3a Performance parameters obtained using proposed method based on ABC algorithm for 8-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0053	21.39	1.63e-6	1.59e-5	1.23e-32	7.12e-9	0.0172
2	0.0041	21.18	9.09e-7	1.73e-5	1.23e-32	1.08e-8	0.0134
3	0.0058	20.13	2.09e-6	2.29e-5	4.93e-32	5.41e-7	0.0195
4	0.0052	20.11	1.66e-6	2.31e-5	1.23e-32	5.26e-7	0.0175
5	0.0045	19.70	1.33e-6	2.62e-5	4.00e-29	6.23e-7	0.0151
6	0.0054	19.71	1.77e-6	2.60e-5	2.08e-30	7.12e-7	0.0178
7	0.0045	18.53	1.36e-6	3.74e-5	1.49e-30	9.61e-7	0.0141
8	0.0065	17.78	3.52e-6	4.46e-5	9.26e-23	8.49e-7	0.0230
9	0.0037	17.55	1.46e-6	5.23e-5	2.39e-21	7.79e-7	0.0079
10	0.0067	16.19	4.20e-6	7.44e-5	9.67e-25	1.33e-7	0.0215

In case of MCS optimized FDCs, these parameters are 0.0037, 8.02e-7, 1.74e-5, 1.23e-32, 7.49e-9, and 0.0089 for 8-channel CMFB; 0.0034, 1.30e-6, 3.23e-5, 1.10e-31, 3.26e-8, and 4.29e-4 for 16-channel CMFB; and 0.0043, 6.36e-7, 1.61e-5, 1.23e-32, 2.61e-9, and 0.0037 for 32-channel CMFB. While in case of ABC optimized FDCs, these performance indices are 0.0037, 9.09e-7, 1.59e-5, 1.23e-32, 7.12e-9, and 0.0079 for 8-channel CMFB; 0.0046, 1.63e-6, 2.86e-5, 1.23e-32, 2.78e-9, and 0.0014 for 16-channel CMFB; and 0.0048, 8.03e-7, 1.44e-5, 1.23e-32, 8.21e-9, and 1.33e-4 for 32-channel CMFB. In PSO optimized FDCs, these indices are 0.0047, 1.21e-6, 1.51e-5, 1.23e-32, 2.10e-9, and 0.0020 for 8-channel CMFB; 0.0021, 7.44e-7, 3.23e-5, 1.23e-32, 1.40e-8, and 7.79e-4 for 18-channel CMFB; and 0.0043, 1.46e-6, 1.41e-5, 1.10e-31, 2.54e-8, and 0.0011 for 32-channel CMFB. These performance indices evidence that the proposed method is very suitable for multiband multirate FB design. During the experiments, it is found that for $M = 32$ and $N + 1 = 512$, the maximum CPU times for each

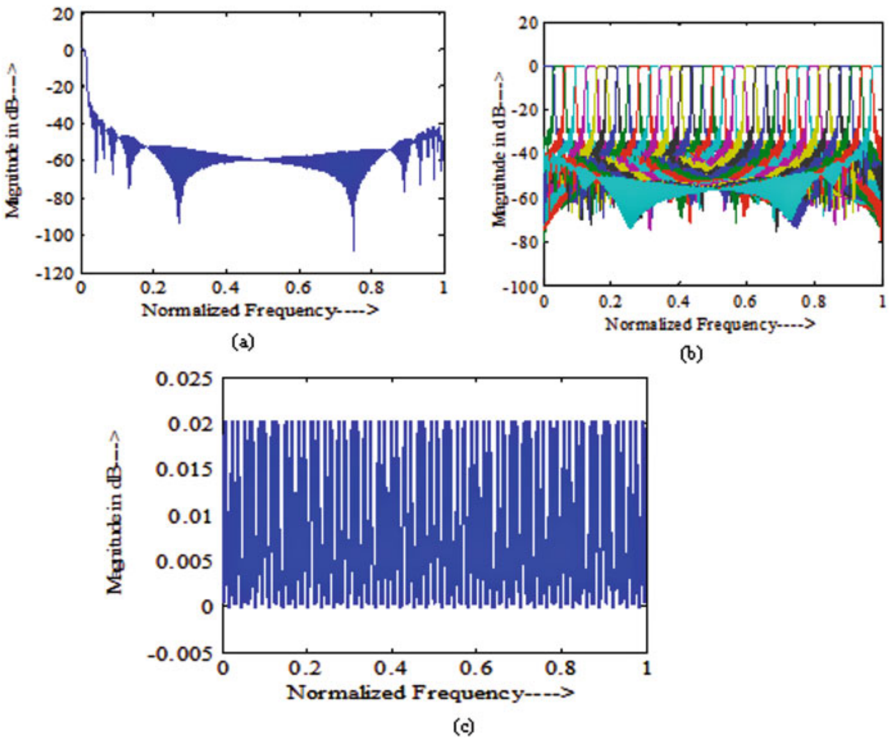


Fig. 6 ABC-based 32-channel CMFB design with $N + 1 = 512$: (**a**) Prototype filter amplitude response in dB; (**b**) amplitude response of analysis filters in dB; (**c**) reconstruction error in dB

Table 3b The optimized FDCs in case 8-channel CMFB for $N + 1 = 160$ and $M = 8$ using ABC algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	9.93									
2	8.86	6.39								
3	38.21	54.10	22.15							
4	61.06	52.44	16.15	66.89						
5	6.73	55.70	11.88	36.99	68.83					
6	65.37	53.06	68.53	79.15	20.46	11.15				
7	18.47	6.68	74.15	29.08	32.15	61.64	8.71			
8	38.20	67.53	54.19	28.86	31.49	30.65	52.332	67.68		
9	66.96	16.79	34.80	6.46	21.30	21.22	71.31	58.89	34.80	
10	8.10	77.38	38.61	57.12	46.09	36.98	19.68	42.85	30.59	27.88

Table 3c Performance parameters obtained using proposed method based on ABC algorithm for 16-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0102	16.72	5.51e-6	2.86e-5	1.10e-31	2.98e-8	0.0391
2	0.0061	16.56	1.63e-6	3.28e-5	1.10e-31	2.78e-9	0.0203
3	0.0072	15.44	2.47e-6	4.61e-5	1.23e-32	3.33e-7	0.0235
4	0.0072	15.19	2.48e-6	4.95e-5	1.23e-32	4.11e-7	0.0226
5	0.0058	15.37	1.70e-6	4.72e-5	5.41e-27	3.03e-7	0.0172
6	0.0074	14.18	3.28e-6	6.88e-5	7.88e-31	5.23e-7	0.0226
7	0.0079	15.00	3.06e-6	5.25e-5	3.22e-22	4.89e-7	0.0255
8	0.0053	12.48	4.60e-6	1.13e-4	1.81e-24	5.92e-7	0.0014
9	0.0064	11.62	1.10e-5	1.67e-4	1.46e-25	5.68e-7	0.0580
10	0.0046	11.51	2.92e-6	1.63e-4	5.74e-27	4.37e-7	0.0063

Table 3d The optimized FDCs in case 16-channel CMFB for $N + 1 = 256$ using ABC algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	57.10									
2	7.46	45.04								
3	14.21	68.06	38.31							
4	68.33	15.84	31.52	52.40						
5	37.59	50.15	9.56	41.78	33.44					
6	67.65	58.30	9.12	65.40	11.29	28.26				
7	45.54	32.90	38.11	69.48	35.84	35.31	32.60			
8	19.75	3.99	72.40	65.79	53.06	48.91	62.34	7.98		
9	12.40	2.94	15.55	38.28	4.22	16.23	40.79	2.57	46.59	
10	20.73	65.10	10.91	36.08	2.57	3.25	14.32	11.05	67.04	4.95

Table 3e Performance parameters obtained using proposed method based on ABC algorithm for 32-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0123	16.49	4.23e-6	1.44e-5	4.93e-32	2.90e-8	0.0484
2	0.0061	16.67	8.03e-7	1.58e-5	1.10e-31	8.21e-9	0.0197
3	0.0073	15.97	1.23e-6	1.96e-5	1.10e-31	4.35e-8	0.0248
4	0.0064	15.28	1.06e-6	2.43e-5	1.23e-32	8.17e-8	0.0186
5	0.0087	15.50	1.85e-6	2.21e-5	1.77e-30	9.13e-8	0.0307
6	0.0090	13.76	2.72e-6	3.77e-5	8.61e-25	1.43e-7	0.0323
7	0.0089	15.18	1.91e-6	2.49e-5	4.10e-23	1.07e-7	0.0299
8	0.0081	13.90	2.03e-6	3.70e-5	1.84e-21	1.38e-7	0.0259
9	0.0048	13.82	2.11e-6	7.22e-5	2.62e-23	6.9e-8	1.33e-4
10	0.0189	13.50	3.12e-5	1.42e-4	7.38e-27	1.12e-7	0.0052

Table 3f The optimized FDCs in case 32-channel CMFB for $N + 1 = 512$ using ABC algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	56.70									
2	6.76	5.51								
3	11.42	61.64	27.10							
4	35.26	13.11	30.87	11.59						
5	57.17	36.02	21.52	62.06	35.35					
6	28.58	13.54	16.39	39.95	43.87	47.83				
7	56.38	47.15	39.90	62.18	53.96	26.57	46.85			
8	55.98	49.58	63.83	37.59	41.51	16.59	38.20	54.89		
9	4.55	15.85	38.96	37.20	25.23	55.46	31.76	55.62	28.74	
10	59.66	27.11	1.17	15.85	38.05	34.87	1.33	10.00	16.93	11.94

Table 4a Performance parameters obtained using proposed method based on PSO algorithm for 8-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0061	21.46	2.29e-6	1.51e-5	1.23e-32	2.10e-9	0.0020
2	0.0047	21.00	1.21e-6	1.81e-5	1.23e-32	1.04e-7	0.0154
3	0.0053	19.81	1.91e-6	2.51e-5	4.93e-32	5.34e-7	0.0182
4	0.0069	20.28	4.17e-6	2.30e-5	1.23e-32	3.54e-7	0.0239
5	0.0098	20.46	7.64e-6	2.14e-5	1.23e-32	3.78e-7	0.0340
6	0.0108	17.46	1.11e-5	4.50e-5	2.88e-26	4.80e-7	0.0401
7	0.0063	19.44	2.44e-6	2.86e-5	1.51e-24	7.30e-7	0.0205
8	0.0071	16.08	5.36e-7	7.91e-5	8.11e-19	2.58e-7	0.0028
9	0.0058	17.58	2.47e-6	5.12e-5	5.07e-23	5.14e-7	0.0178
10	0.0060	16.74	2.81e-6	1.17e-4	2.26e-11	6.17e-7	0.0117

Table 4b The optimized FDCs in case 8-channel CMFB for $N + 1 = 160$ using PSO algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	16.75									
2	7.40	55.04								
3	7.58	56.04	19.13							
4	10.74	1.53	27.46	12.09						
5	26.89	77.24	74.59	1.47	19.15					
6	68.32	76.90	39.03	33.64	15.78	40.72				
7	63.76	50.81	32.74	69.49	52.18	35.24	36.04			
8	56.17	21.39	78.51	53.50	08.57	65.67	02.91	58.82		
9	33.51	31.60	73.65	41.59	74.56	54.36	64.89	7.65	29.47	
10	24.48	69.24	22.74	49.49	69.65	67.33	27.87	68.72	23.98	01.54

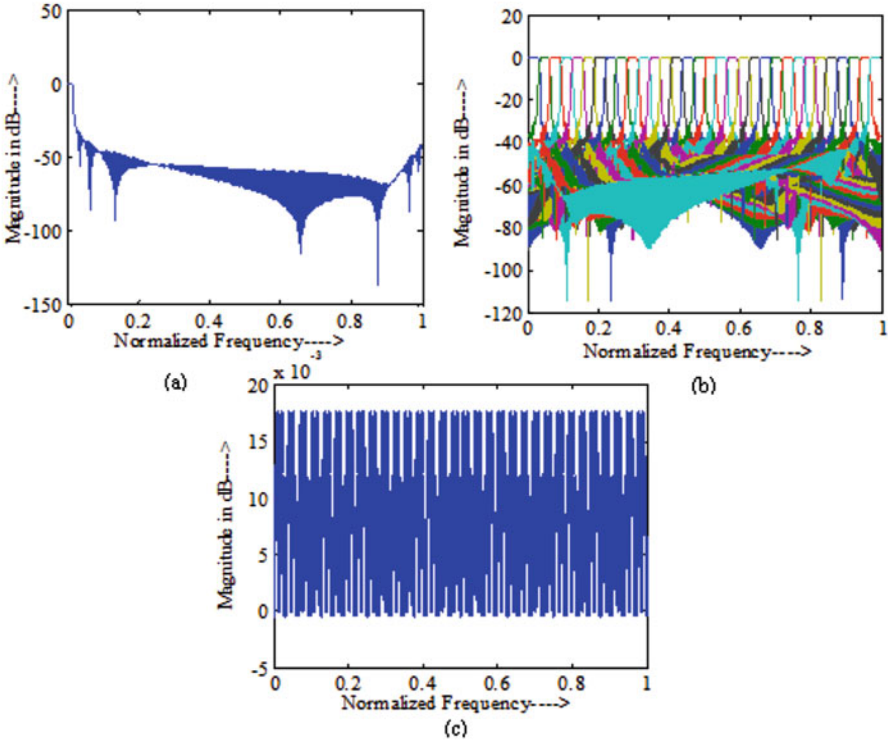


Fig. 7 PSO-based 32-channel CMFB design with $N + 1 = 512$: (**a**) Prototype filter amplitude response in dB; (**b**) amplitude response of analysis filters in dB; (**c**) reconstruction error in dB

Table 4c Performance parameters obtained using proposed method based on PSO algorithm for 16-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0054	16.61	1.30e-6	3.23e-5	1.97e-31	3.26e-8	0.0166
2	0.0050	16.42	1.11e-6	3.49e-5	1.10e-31	1.40e-8	0.0144
3	0.0045	16.48	1.12e-6	3.39e-5	1.23e-32	2.97e-8	0.0112
4	0.0055	15.50	1.56e-6	4.55e-5	9.98e-31	2.53e-7	0.0161
5	0.0073	14.94	2.64e-6	5.36e-5	2.20e-27	4.76e-7	0.0224
6	0.0024	14.28	1.20e-6	7.31e-5	2.81e-28	4.51e-7	7.79e-4
7	0.0074	14.97	2.69e-6	5.30e-5	3.97e-26	4.77e-7	0.0229
8	0.0039	13.41	1.61e-6	8.86e-5	6.95e-26	6.04e-7	0.0026
9	0.0021	13.15	7.44e-7	1.00e-4	6.25e-26	7.03e-7	0.0086
10	0.0075	14.05	3.13e-6	7.08e-5	2.78e-19	5.82e-7	0.0231

Table 4d The optimized FDCs in case 16-channel CMFB for $N + 1 = 256$ using PSO algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	4.28									
2	6.72	40.90								
3	8.92	22.96	6.88							
4	9.47	35.40	39.52	13.22						
5	33.90	48.35	23.79	40.29	40.42					
6	9.48	5.27	16.28	35.11	54.35	39.93				
7	23.24	48.39	50.62	25.08	7.00	58.81	28.66			
8	52.31	67.37	57.42	26.25	9.94	9.74	45.53	4.33		
9	63.54	53.06	9.35	61.32	3.84	53.25	42.38	2.43	21.16	
10	44.09	31.28	44.43	14.99	55.68	52.01	24.97	65.91	30.31	49.58

Table 4e Performance parameters obtained using proposed method based on PSO algorithm for 32-channel CMFB by varying different number of FDCs

No. of constraints	PRE	A_S	ϕ_p	ϕ_s	ϕ_t	e_a	e_{am}
1	0.0126	16.55	4.41e-6	1.41e-5	1.10e-31	2.54e-8	0.0493
2	0.0129	16.48	4.65e-6	1.42e-5	1.10e-31	3.08e-8	0.0507
3	0.0126	16.32	4.40e-6	1.54e-5	1.49e-30	4.23e-8	0.0492
4	0.0123	15.31	7.74e-7	3.39e-5	4.93e-30	1.41e-7	0.0485
5	0.0121	14.15	5.01e-6	3.10e-5	3.56e-30	1.20e-7	0.0480
6	0.0069	15.66	2.38e-6	2.23e-5	8.72e-28	9.38e-8	0.0218
7	0.0043	14.08	2.31e-6	3.77e-5	1.08e-27	1.57e-7	0.0011
8	0.0053	11.78	1.63e-6	7.42e-5	1.52e-21	5.76e-8	0.0041
9	0.0080	13.78	1.99e-6	3.81e-5	8.42e-22	1.43e-7	0.0257
10	0.0050	12.02	1.46e-6	6.94e-5	5.44e-23	7.19e-8	0.0031

Table 4f The optimized FDCs in case 32-channel CMFB for $N + 1 = 512$ using PSO algorithm

No. of constraints	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	39.14									
2	50.12	22.64								
3	27.82	48.01	43.87							
4	5.97	88.62	62.08	95.88						
5	46.94	6.13	20.83	51.41	32.63					
6	58.81	1.43	37.13	43.29	31.91	29.60				
7	39.50	12.50	33.50	50.50	13.50	1.50	28.50			
8	7.36	19.50	26.03	35.66	31.94	19.79	33.73	18.01		
9	26.74	12.97	32.33	28.60	46.21	57.94	51.09	47.10	50.75	
10	13.32	57.31	5.60	27.87	48.34	43.41	36.49	38.12	4.84	42.47

iteration taken by CS, MCS, PSO, and ABC are 76.51 *s*, 81.81 *s*, 80.41 *s*, and 76.51 *s*, respectively. It has been also found that 100 iterations are sufficient for convergence of CS, MCS, PSO, and ABC for CMFB design.

7.2 Comparison of Proposed Method with Other Existing Methods

Table 5 compares the proposed method with recently reported methods [5–13, 17, 24, 26] for M-channel CMFB design in terms of aliasing distortion and amplitude distortion. The comparison is carried out for $M = 8$-, 16- and 32-channel CMFB. It is apparent from the comparative statistical analysis that in terms of amplitude and

Table 5 Comparative study with recently proposed methods for M-channel CMFB design

Type of algorithm	M	$N + 1$	e_{am}	e_a
Algorithm [5]	32	460	2.9e-3	1.14e-7
Algorithm [6]	8	144	4.1e-3	2.55e-7
	8	160	4.1e-3	1.68e-7
	16	224	4.2e-3	2.96e-7
	32	512	4.0e-3	5.27e-8
Algorithm [7]	8	160	4.8e-3	7.15e-7
	32	512	4.7e-3	9.51e-7
Algorithm[8]	32	512	1.00e-3	4.67e-8
Algorithm [9]	32	440	3.42e-3	2.60e-7
Algorithm [10]	32	512	9.99e-4	5.68e-7
Algorithm [11]	8	144	1.8e-3	3.38e-7
	8	160	1.9e-3	7.98e-7
	16	260	4.2e-3	3.15e-7
	32	512	1.0e-3	8.82e-7
Algorithm [12]	32	512	2.01e-3	2.63e-7
Algorithm [13]	8	144	2.7e-3	3.75e-6
	8	160	3.4e-3	1.22e-6
	16	224	3.1e-3	4.24e-7
	32	512	2.1e-3	4.00e-8
Algorithm [17]	8	128	–	2.40e-6
	16	256	–	2.34e-6
	32	512	–	2.29e-6
Algorithm [24]	8	160	1.4e-3	7.14e-7
	16	256	9.6e-4	4.16e-7
	32	512	1.4e-3	8.08e-8
Algorithm [26]	8	160	1.33e-4	7.86e-9
	16	256	1.75e-4	3.22e-8
	32	512	1.81e-4	9.46e-9

(continued)

Table 5 (continued)

Type of algorithm	M	$N+1$	e_{am}	e_{a}
Proposed (CS) + MCA	**8**	**160**	**6.81e-4**	**5.44e-9**
Proposed (CS) + MCA	**16**	**256**	**3.96e-4**	**6.07e-7**
Proposed (CS) + MCA	**32**	**512**	**5.15e-4**	**1.45e-8**
Proposed (MCS) + MCA	**8**	**160**	**8.90e-3**	**8.14e-7**
Proposed (MCS) + MCA	**16**	**256**	**4.29e-4**	**5.78e-7**
Proposed (MCS) + MCA	**32**	**512**	**3.70e-3**	**1.66e-7**
Proposed (ABC) + MCA	**8**	**160**	**7.90e-3**	**7.79e-7**
Proposed (ABC) + MCA	**16**	**256**	**1.40e-3**	**5.92e-7**
Proposed (ABC) + MCA	**32**	**512**	**1.33e-4**	**6.90e-8**
Proposed (PSO) + MCA	**8**	**160**	**2.00e-3**	**2.10e-9**
Proposed (PSO) + MCA	**16**	**256**	**7.79e-4**	**4.51e-7**
Proposed (PSO) + MCA	**32**	**512**	**1.10e-3**	**1.57e-7**

aliasing distortion, optimized PCs obtained after applying CS optimized FDCs give excellent performance in terms of amplitude distortion, and CS and PSO optimized FDCs give excellent performance in terms of aliasing distortion. Overall, CS-based PCs give excellent performance for M-channel CMFB design.

8 Conclusion

In this chapter, a new design technique based on minor component analysis (MCA) in conjunction with optimized PCs of filter and swarm optimization (CS, MCS, PSO and ABC) FDCs are employed for multirate FB design. It can be concluded, from the simulation results, that the proposed method gives better performance in terms of passband error, stopband error, and transition band error and overall performance parameters of M-channel FB like amplitude distortion, *PRE*, aliasing distortion. The comparison with other recently published methods shows that the proposed techniques give excellent performance. Compared to CS, MCS, PSO, and ABC, in terms of amplitude and aliasing distortion, CS gives better performance than other SOTs. In terms of *PRE*, CS and MCS give better performance as compared to PSO and ABC. In terms of computational time, again CS and ABC consume less time as compared to MCS and PSO because of using different optimized population size.

References

1. Dolecek, G. J. (2002). *Multirate systems: Design and applications*. Hershey PA: Idea Group of Publishing.
2. Vaidyanathan, P. P. (1993). *Multirate systems and filter banks*. Englewood Cliffs, NJ: Prentice-Hall.

3. Creusere, C. D., & Mitra, S. K. (1995). A simple method for designing high quality prototype filters for M-band pseudo QMF banks. *IEEE Transaction of Signal Processing, 43*, 1005–1007.
4. Cruz-Roldán, F., Lopez, P. A., Bascon, S. M., & Lawson, S. S. (2002). An efficient and simple method for designing prototype filters for cosine modulated pseudo QMF banks. *IEEE Signal Processing Letters, 9*(1), 29–31.
5. Cruz-Roldán, F., Martín-Martín, P., Sáez-Landete, J., Blanco-Velasco, M., & Saramäki, T. (2009). A fast windowing-based technique exploiting zz of Kaiser window based optimized prototype filter for cosine modulated filter banks. *Signal Processing, 90*, 1742–1749.
6. Kumar, A., Singh, G. K., & Kuldeep, B. (2011). An improved and simplified approach for designing cosine modulated filter Bank using window technique. *International Journal of Mathematical Modelling and Algorithm, 10*, 213–226.
7. Kumar, A., & Kuldeep, B. (2012). Design of M-channel cosine modulated filter bank using modified exponential window. *Journal of the Franklin Institute, 349*, 1304–1315.
8. Bergen, S. W. A. (2008). A design for cosine modulated filter banks using weighted constrained least squares filters. *Digital Signal Processing, 18*, 282–290.
9. Kumar, A., Singh, G. K., & Anand, R. S. (2011). A simple design method for the cosine-modulated filter banks using weighted constrained least square technique. *Journal of the Franklin Institute, 348*, 606–621.
10. Berger, S. W. A., & Antoniou, A. (2007). An efficient closed form design method for cosine modulated filter banks using window function. *Signal Processing, 87*, 811–823.
11. Kumar, A., Singh, G. K., & Anand, R. S. (2011). A closed form design method for the two channel quadrature mirror filter banks. *Signal Image and Video Processing, 5*(1), 121–131.
12. Kumar, A., Singh, G. K., & Anand, R. S. (2011). An improved closed form design method for the cosine modulated filter banks using windowing technique. *Applied Soft Computing, 11*, 3209–3217.
13. Kumar, A., Pooja, R., & Singh, G. K. (2014). Design and performance of closed form method for cosine modulated filter bank using different windows functions. *International Journal of Speech Technology, 17*(4), 427–441.
14. Baicher, G. S. (2007). Optimal design of a class of M-channel uniform filter bank using genetic algorithms. In *Proceedings of the ICSPC* (pp. 24–27). Dubai: IEEE.
15. Tan, F., Zhang, T., Gao, C., & Huang, L. (2011). Optimal Design of Cosine Modulated Filter Banks using quantum-behaved particle swarm optimization algorithm. In *Proceedings of the CISP* (pp. 2280–2284). Shanghai: IEEE.
16. Sharma, I., Kumar, A., & Singh, G. K. (2016). Adjustable window based Design of Multiplier-Less Cosine Modulated Filter Bank using swarm optimization algorithms. *AEUE - International Journal of Electronics and Communications, 70*, 85–94.
17. Pei, S. C., & Tseng, C. C. (2001). A new Eigen filter based on total least squares error criterion. *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, 48*(6), 699–709.
18. Das, S., & Pan, I. (2012). *Fractional order signal processing: Introductory concepts and applications*. Heidelberg/New York: Springer.
19. Tseng, C. C., & Lee, S. L. (2012). Design of linear phase FIR filters using fractional derivative constraints. *Signal Processing, 92*, 1317–1327.
20. Tseng, C. C., & Lee, S. L. (2013). Designs of two-dimensional linear phase FIR filters using fractional derivative constraints. *Signal Processing, 93*, 1141–1151.
21. Kuldeep, B., Kumar, A., & Singh, G. K. (2015). Design of Quadrature Mirror Filter Bank using Lagrange multiplier method based on fractional derivative constraints. *Engineering Science and Technology, an International Journal, 18*(2), 235–243.
22. Kuldeep, B., Singh, V. K., Kumar, A., & Singh, G. K. (2015). Design of two-channel filter bank using nature inspired optimization based fractional derivative constraints. *ISA Transaction, 54*, 101–116.

23. Kuldeep, B., Kumar, A., & Singh, G. K. (2015). Design of Multi-Channel Cosine Modulated Filter Bank Based on fractional derivative constraints using cuckoo search algorithm. *Circuits, Systems, and Signal Processing, 34*(10), 3325–3351.
24. Baderia, K., Kumar, A., & Singh, G. K. (2015). Design of quadrature mirror filter bank using polyphase components based on optimal fractional derivative constraints. *AEUE – International Journal of Electronics and Communications, 69*(9), 1254–1264.
25. Baderia, K., Kumar, A., Singh, G. K. (2016). An improved method for designing cosine modulated filter bank using polyphase components. *IEEE conference on Signal Processing & Integrated Networks (SPIN)*, pp. 9–13. Noida, India.
26. Baderia, K., Kumar, A., Singh, G. K. (2015). Design of multi-channel filter bank using ABC optimized fractional derivative constraints. *IEEE Conference on Communication and Signal Processing (ICCSP)*, pp. 492–496. Melmaruvathur, India.
27. Kuldeep, B., Kumar, A. Singh, G. K. (2015). PSO based optimized fractional derivative constraints for designing m-channel filter bank. *IEEE Conference on Signal Processing Computing and Control (ISPCC)*, pp. 140–144. Solan, India.
28. Du, K. L., Lai, A. K. Y., Cheng, K. K. M., & Swamy, M. N. S. (2002). Neural methods for antenna array signal processing: A review. *Signal Processing, 82*, 547–561.
29. Fiori, S. (2003). Neural minor component analysis approach to robust constrained beamforming. *IEE Proceedings, Vision, Image and Signal Processing, 150*(4), 205–218.
30. Castells, F., Laguna, P., Sornmo, L., Bollmann, A., & Roig, J. M. (2007). Principal component analysis in ECG signal processing. *EURASIP Journal on Advances in Signal Processing, 2007*, 074580. http://dx.doi.org/10.1155/2007/74580.
31. Jou, Y. D. (2007). Design of two channel linear phase QMF bank based on neural networks. *Signal Processing, 87*(5), 1031–1044.
32. Chen, L. W., Jou, Y. D., Chen, F. K., & Hao, S. S. (2014). Eigen filter design of linear-phase FIR digital filters using neural minor component analysis. *Digital Signal Processing, 32*, 146–155.
33. Chen, L. W., Jou, Y. D., & Hao, S. S. (2015). Design of two-channel quadrature mirror filter banks using minor component analysis algorithm. *Circuits System Signal Processing, 34*(5), 1549–1569.
34. Agrawal, S. K., & Sahu, O. P. (2013). Two-channel quadrature mirror filter bank design using FIR polyphase component. *ACEEE International Journal on Signal Image Processing, 4*(1), 24–28.
35. Agrawal, S. K., & Sahu, O. P. (2014). Design of alias-free linear phase quadrature mirror filter banks using Eigen value- Eigen vector approach. *An International Journal on Electrical and Electronics Engineering, 3*(2), 111–120.
36. Pei, S. C., Tseng, C. C., & Yang, W. S. (1998). FIR filter designs with linear constraints using the Eigen filter approach. *IEEE Transaction on Circuits and Systems-II: Analog and Digital Signal Processing, 45*(2), 232–237.
37. Golub, E. H., & Van Loan, C. (1983). *Matrix computations*. Baltimore, MD: Johns Hopkins University Press.
38. Yang, X. S., & Deb, S. (2010). Engineering optimization by cuckoo search. *International Journal on Mathematical Modelling and Numerical Optimization, 1*(4), 330–343.
39. Walton, S., Hassan, O., Morgan, K., & Brown, M. R. (2011). Modified cuckoo search: A new gradient free optimization algorithm. *Chaos, Solitons & Fractals, 44*, 710–718.
40. Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *Journal of Global Optimization, 39*(3), 459–471.
41. Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization: An overview. *Swarm Intelligence, 1*(1), 33–57.

Multiresolution Filter Banks for Pansharpening Application

Hind Hallabia, Abdelaziz Kallel, and Ahmed Ben Hamida

1 Introduction

Multirate filter banks were introduced by Croisier et al. (1976) [2] and Esteban and Galand (1977) [3], which have been applied to subband coding, image compression, speech coding, and signal denoising and in pansharpening [1, 4]. Numerous multirate filter banks have been discussed in literature, including the multichannel and in particular the two-channel filter banks [5]. They can be regrouped into quadrature mirror filter (QMF) [2], orthogonal [6] and biorthogonal filter banks [5], which are mainly based on three basic operations: linear filtering, down-sampling, and up-sampling [7].

Among existing multirate filter banks, we focus in this chapter on the two-channel one [4–6]. The latter has been recently adopted to fuse remotely sensed imagery in Hallabia et al. (2016) [1]. Such application is called pansharpening [8]. It consists in transferring the spatial content of panchromatic (PAN) image at finer resolution into an image at coarse resolution, e.g., multispectral (MS) or hyper-spectral (HS) image.

During the last two decades, numerous approaches are introduced in the literature, which can be classified mainly into three large categories [8]. The first class, called Component Substitution (CS) [13], is based on a spectral transformation (e.g., using the intensity–hue–saturation, principal component analysis, Gram–Schmidt

---

H. Hallabia (✉)
Advanced Technologies for Medicine and Signals (ATMS), Sfax University, Sfax, Tunisia

Digital Research Center of Sfax (CRNS), Sfax University, Sfax, Tunisia
e-mail: hindhallabia@yahoo.fr

A. Kallel
Digital Research Center of Sfax (CRNS), Sfax University, Sfax, Tunisia

A.B. Hamida
Advanced Technologies for Medicine and Signals (ATMS), Sfax University, Sfax, Tunisia

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9_5

transform). Methods based Multiresolution Analysis [10] use spatial frequency decomposition which is usually performed by means of high-pass filtering (e.g., the Laplacian pyramid or wavelet transform). The model-based methods constitute the third class, which are based on compressing sensing, sparse representation, and Bayesian approach.

The rest of this chapter is organized as follows. Section 2 recalls the analysis/synthesis configuration for the two-channel filter bank. The different solutions, existing in the literature, for filter banks are discussed in Sect. 3 (the mono-dimensional signal is considered). Then, the case of image is introduced in Sect. 4 Section 5 presents a short overview about CS and MRA pansharpening techniques. Experimental results are discussed in Sect. 6, including datasets, the selected pansharpening algorithms, and the quality assessment metrics. Finally, conclusion and some perspectives are shown in Sect. 7.

2 Two-Channel Filter Banks

In this section, the analysis/synthesis configuration is discussed. The block diagram of the two-channel filter banks is reported in Fig. 1. It consist of an analysis stage [low-pass filter $h[n]$ and high-pass filter $g[n]$ followed by down-sampling operators by a factor two (↓2)] and a synthesis stage [low-pass filter $h_r[n]$ and high-pass filter $g_r[n]$ followed by an up-sampling operator (↑2)].

Let us recall some definitions according to filter banks. If the subsampling and up-sampling operations have similar factors (i.e., equals two in our case), the filter bank is uniform [4]. Moreover, a uniform filter bank is critically sampled, if the number of branches is equal to the up-sampling factor [4].

In the following, the analysis/synthesis of an input signal is introduced in discrete and in the Z-transform domain.

2.1 Discrete Domain

The input signal $x[n]$ is divided into low-frequency and high-frequency subbands $(v_1[n], v_2[n])$. Each subband $v_m[n], (m = 1, 2)$ has a rate twice as low as the original

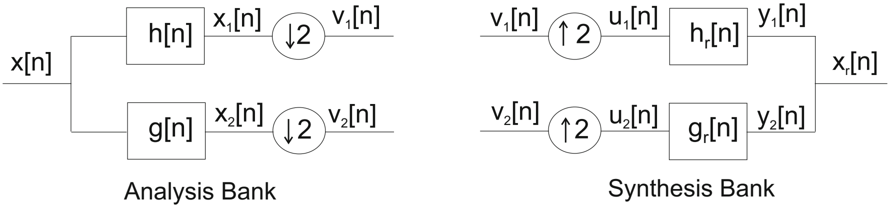


Fig. 1 Two-channel filter banks: analysis (in the *left*) and synthesis configuration (in the *right*)

signal $v_1[n]$. The two components $v_1[n]$ and $v_2[n]$ are obtained by applying a low-pass and high-pass filtering operation of impulse response $h[n]$ and $g[n]$, respectively, given by

$$\begin{cases} v_1[n] = \sum_{k=-\infty}^{+\infty} x[k]h[2n-k], \\ v_2[n] = \sum_{k=-\infty}^{+\infty} x[k]g[2n-k]. \end{cases} \tag{1}$$

The synthesis bank allows to reconstruct a signal $x_r[n]$ from two components $v_1[n]$ and $v_2[n]$. After applying an interpolation step with a factor of two, these two subbands are convolved, respectively, by a low-pass $h_r[n]$ and high-pass $g_r[n]$ filter. The two output results of the interpolator filters are summed together to create the signal $x_r[n]$ having the same rate as the original signal $x\ [n]$.

The obtained signal $x_r[n]$ is expressed as

$$x_r[n] = \sum_{k=-\infty}^{+\infty} v_1[k]h_r[n-2k] + v_2[k]g_r[n-2k]. \tag{2}$$

Substituting (2) in (1), we obtain

$$x_r[n] = \sum_{k=-\infty}^{+\infty} \sum_{p=-\infty}^{+\infty} x[p][h[2k-p]h_r[n-2k] + g[2k-p]g_r[n-2k]]. \tag{3}$$

Considering

$$e[n,p] = [h[2k-p]h_r[n-2k] + g[2k-p]g_r[n-2k]] \tag{4}$$

The reconstructed signal is then given as

$$x_r[n] = \sum_{p=-\infty}^{+\infty} x[p]e[n,p]. \tag{5}$$

In this case, the filter bank under perfect reconstruction is considered, which is obtained if the reconstructed signal is a delayed version of the input signal [5], in other words, when the signal $x_r[n]$ is equal to $x[n]$ with a delay$D$.

Taking in consideration the relation (4), the perfect reconstruction condition is verified if

$$e[n,p] = \delta[n-p-D] \tag{6}$$

Combining (4) and (6), we obtain

$$\sum_{k=-\infty}^{+\infty} h[2k-p]h_{\mathrm{r}}[n-2k]+g[2k-p]g_{\mathrm{r}}[n-2k]=\delta[n-p-D]. \tag{7}$$

The Eq. (7) is known as the biorthogonality property or the perfect reconstruction [5].

Since the filter bank contains subsampling operations, the phenomenon of aliasing artifact could be present in the reconstructed signal $x_{\mathrm{r}}[n]$. Therefore, to be perfectly equal to the input signal $x[n]$, the following relation must be satisfied:

$$\sum_{k=-\infty}^{+\infty} h[2k-p]h_{\mathrm{r}}[n-2k]+g[2k-p]g_{\mathrm{r}}[n-2k]=1, \forall p=n-D. \tag{8}$$

The Eq. (8) is known as the aliasing suppression property [5].

The critically decimated filter bank allows to decompose a discrete signal into two subbands and to recover it under the perfect reconstruction condition (7) and the aliasing-free condition (8).

2.2 Z-Transform Domain

In the Z-transform domain, the reconstructed signal is given by

$$X_{\mathrm{r}}(z)=\frac{1}{2}[E(z)X(z)+F(z)X(-z)] \tag{9}$$

where

$$E(z)=\frac{1}{2}\,[H(z)H_{\mathrm{r}}(z)+G(z)G_{\mathrm{r}}(z)\,], \tag{10}$$

is the *distortion transfer function*, and

$$F(z)=\frac{1}{2}\,[H(-z)H_{\mathrm{r}}(z)+G(-z)G_{\mathrm{r}}(z)\,] \tag{11}$$

is called the aliasing transfer function.

In the relation (9), the component containing $X\,(z)$ represents the desired signal, and that containing $X\,(-z)$ is the alternate signal which causes the aliasing artifacts due to the subsampling operation.

The aliased component X$(-z)$ can be very disturbing particularly in audio applications signal [4]. Aliasing artifact is viewed as a nonharmonic distortion. Indeed, new sinusoidal components appear which are not harmonically related to the input signal [4].

Furthermore, the output signal $Xr\ (z)$ must be a delayed version of the input signal $X\ (z)$, in order to verify the perfect signal reconstruction constraint [4–6]. Based on (9) and (11), this condition leads to

$$\frac{1}{4}[H(z)G(-z) - H(-z)G(z)] = z^{-D} \tag{12}$$

where D is the delay.

3 Filter Bank Solutions

In the previous section, the analysis/synthesis configuration is shown. As mentioned above, three categories of filter banks are existing, which are the quadrature mirror filter (QMF) [2], orthogonal [6] and biorthogonal filter banks [5]. Here, we will present the different solutions to the Eq. (12) related to filters design under the perfect reconstruction condition.

3.1 Quadrature Mirror Filter Banks

In Croisier and Esteban (1976) [2], $H(z)$ and $G(z)$ are chosen as follows: $(z) = H(-z)$, which is the alternating of the low-pass magnitude $H(z)$. The resulting filter bank was called quadrature mirror filter (QMF).

In the frequency domain, this relation leads to $G(e^{jw}) = H(e^{jw+\pi})$. Indeed, the high-pass response $|G(e^{jw})|$ is a mirror image of the low-pass magnitude $|H(e^{jw})|$ with respect to the middle frequency $\frac{\pi}{2}$. The responses are symmetric.

Choosing the QMF filters, the relation (9) is expressed as

$$X_{\mathrm{r}}(z) = \frac{1}{2}\left[H^2(z) - G^2(-z)\right]X(z) \tag{13}$$

When the condition $x_{\mathrm{r}}[n] = x[n-D]$ is desired, the relation $H^2(z) - G^2(-z) = 2z^{-D}$ must be verified. In this case, the aliasing cancelation propriety is achieved.

In the discrete domain, under the QMF constraint and given the impulse response of the analysis low-pass filter $h[n]$, the filters are expressed as follows:

$$\begin{cases} g[n] = (-1)^n h[n], \\ h_{\mathrm{r}}[n] = h[n], \\ g_{\mathrm{r}}[n] = -(-1)^n h[n]. \end{cases} \tag{14}$$

3.2 Orthogonal Filter Banks

A further solution to QMF, according to Smith and Barnwell (1984) [9], the high-pass response $G(z)$ is considered as the alternating flip of the low-pass magnitude $H(z)$. In this case, $H(z)$ and $G(z)$ are related as follows:

$$G(z) = -z^{-D}\, H\left(-z^{-1}\right) \tag{15}$$

when $H(z)$ is correctly chosen, (15) leads to the orthogonal filter banks.

For orthogonal filter banks, the aliasing cancelation propriety (10) is satisfied. In order to verify the perfect reconstruction condition (12), the *complementary power propriety* [6, 10] is considered. Such filters (solutions) must verify the following relation:

$$\tilde{H}(z)\, H(z) + \tilde{H}(-z)\, H(-z) = 1, \tag{16}$$

where $\tilde{H}(z) = H_*(z^{-1})$; the subscript * indicates a complex number.

Considering $G(z) = -z^{-D}\, \tilde{H}(-\mathrm{z})$ and for an even number D, the distortion transfer function (11) is given by

$$T(z) = \frac{z^{-D}}{2}\left[\tilde{H}(z)\, H(z) + \tilde{H}(-z)\, H(-z)\right] \tag{17}$$

As solution to (16), when $H(z)$ verifies the *complementary power propriety,* the perfect reconstruction propriety is achieved. Therefore, the filter solutions are defined by

$$\begin{aligned} G(z) &= -z^{-D}\tilde{H}(-z) \\ H_{\mathrm{r}}(z) &= \ \ z^{-D}\tilde{H}(z) \\ G_{\mathrm{r}}(z) &= z^{-D}\tilde{G}(z) \end{aligned} \tag{18}$$

In the discrete domain, the orthogonality condition is derived from (16) and (18):

$$\begin{aligned} 2\sum_n h[n]h[n-2k] &= \delta[k] \\ 2\sum_n h[n]g[n-2k] &= 0, \\ 2\sum_n g[n]g[n-2k] &= \delta[k] \end{aligned} \tag{19}$$

3.3 Biorthogonal Filter Banks

In this case, the low-pass analysis filter $H(z)$ is firstly chosen. Then, the corresponding high-pass analysis filter is constructed from $G(z)$. In (10), the product $H(z)G(z)$ is a halfband filter, which gives *biorthogonality*. Moreover, in order to eliminate the term containing aliasing problem (cf., Eq. 9), $F(z)$ must be canceled. As solution, the two synthesis filters are chosen as follows:

$$\begin{cases} H_r(z) = \frac{1}{2} G(-z) \Rightarrow h_{\mathrm{r}}[n] = \frac{1}{2} (-1)^n g[n], \\ G_{\mathrm{r}}(z) = -\frac{1}{2} H(-z) \Rightarrow g_{\mathrm{r}}[n] = -\frac{1}{2} (-1)^n h[n]. \end{cases} \tag{20}$$

Defining the product filter as $P(z)=H(z)H_{\mathrm{r}}(z)$ and taking into account the condition of aliasing cancelation, the product $G(z)G_{\mathrm{r}}(z)=-P(-z)$ is satisfied. Relation (10) becomes

$$E(z) = \frac{1}{2} \left[P(z) - P(-z) \right] \tag{21}$$

That makes the perfect reconstruction condition expressed as follows:

$$P(z) - P(-z) = 2z^{-D} \tag{22}$$

The biorthogonal filter banks are designed with respect to the biorthogonality condition and deduced from (20) and (21) as

$$\begin{aligned} 2 \sum_n h[n]\, h_r[n-2k] &= \delta[k] \\ 2 \sum_n h[n] g_r[n-2k] &= 0, \\ 2 \sum_n g[n] h_r[n-2k] &= 0, \\ 2 \sum_n g[n] g_r[n-2k] &= \delta[k]. \end{aligned} \tag{23}$$

The biorthogonal filter bank is considered in our application in fusing the remotely sensed data (i.e., pansharpening).

4 Tree-Structured Filter Banks

In this section, the tree-structured filter banks [4] are introduced in image case. Using the separability property, the subband decomposition is straightforward generalized to multiple dimensions. In fact, separable decompositions consist in applying down-sampling and filtering operations on rows and columns, at each stage of the decomposition.

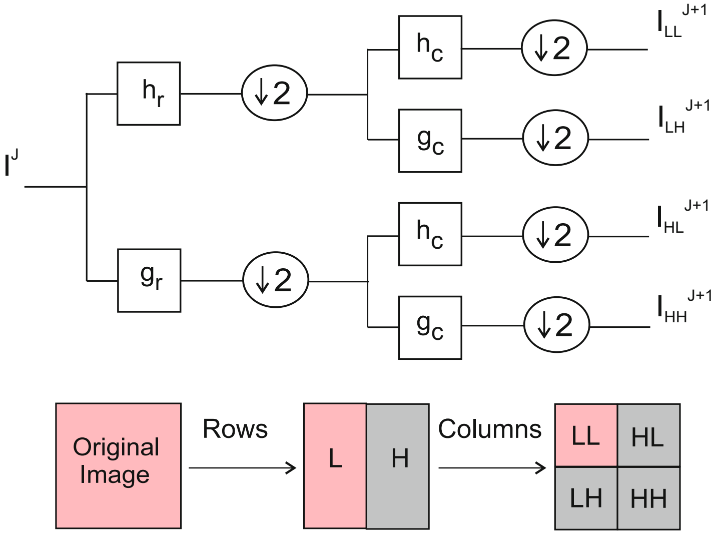


Fig. 2 Multiresolution analysis filter banks for image decomposition (in the *top*) and image decomposition example (in the *bottom*). The subscripts *r* and c indicate that such operation is applied on rows and columns, respectively. The superscript *j* indicates the decomposition level

4.1 Image Decomposition

Given an image I^j, the filtering and down-sampling operations $(\downarrow 2)$ are performed firstly on rows (i.e., vertically) using low-pass filter *h* and high-pass filter *g*. Then, they are applied on columns (i.e., horizontally), creating then four sub-images. The decomposition of the image gives in each level of decomposition (*j*) four sub-images: a decimated version of the input image (called a low-frequency approximation, I^{j+1}_{LL}) and three high-frequency sub-images (I^{j+1}_{LH}, I^{j+1}_{HL}, I^{j+1}_{HH}) containing the details of the input image I^j oriented in horizontal, vertical, and diagonal directions.

A schematic diagram of multiresolution image decomposition is shown in Fig. 2. After the decomposition, coarse resolution images of half size are obtained: a smoothed decimated version (LL) as well as a three difference or detail images (LH, HL, and HH).

4.2 Image Reconstruction

The original image, I^j, can be reconstructed using up-sampling and interpolation operations, under the perfect reconstruction and aliasing cancelation conditions. In fact, the low-frequency (I^{j+1}_{LL}) and the high-frequency subbands (I^{j+1}_{LH}, I^{j+1}_{HL}, I^{j+1}_{HH}) at coarse resolution are up-sampled by a factor of two $(\uparrow 2)$, using the

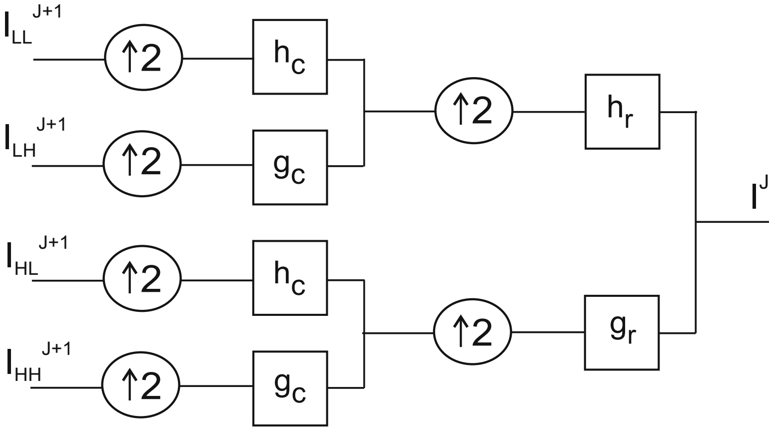


Fig. 3 Multiresolution synthesis filter banks for image reconstruction

zero-padding operation. The size of these subbands is then doubled. After that, a filtering operation is performed on these resulting subbands, in order to eliminate the imaging effect which is created by up-sampling. The two operations are firstly performed on columns then on rows (cf., Fig. 3).

In some applications in image processing, such as in the pansharpening context [1, 8], the scale ratio between PAN and MS imagery (e.g., Pléaides, GeoEye-1, QuickBird) is generally four. The 2D decomposition filter bank (cf., Fig. 2) with two levels is then needed. It is applied twice: firstly on the input image I^j and secondly on the low-frequency subband $I^{j+1}{}_{LL}$. As result, seven subbands are obtained (a low-pass approximation and six details).

5 Overview About CS and MRA Methods

Pansharpening algorithms allow to enhance the spatial resolution of the MS bands by injecting the high-frequency structures derived from the PAN image while preserving its original spectral content. Firstly, an up-sampling is performed to the original MS bands at coarse resolution. Spatial details are extracted through a spectral transform (i.e., CS-based method) or a multi-scale transform (i.e., MRA-based method). Then, they are transferred to the different interpolated MS bands by a simple addition or modulated by gains in order to obtain the high-resolution MS image (at finer resolution). Generally, a pansharpening approach is given by the following relation [8]:

$$\widehat{MS}_k = \widetilde{MS}_k + g_k P_{\mathrm{D}}; k = 1, \ldots, N \tag{23}$$

where k indicates the k^{th} MS band, $\{g_k\}_{k=1,\ldots,N}$ defines the gain vector, and P_{D} is the high-frequency detail.

Based on the manner of extracting spatial details, pansharpening methods are classified into three major categories: the component substitution (CS), multiresolution analysis (MRA), and the model-based methods.

In this section, we focus mainly on the CS and MRA methods. The general principle of the CS techniques is summarized as follows: *(1)* the original MS bands are interpolated in order to be aligned to the PAN image; *(2)* the intensity component ($\tilde{I}$) is estimated from MS bands; *(3)* a histogram matching is performed between PAN and $\tilde{I}$; and *(4)* the geometrical details, obtained by the difference between the PAN image and the $\tilde{I}$ component, are injected to the interpolated MS image (cf. relation (24)). A CS-based pansharpening can be expressed by the following relation [8]:

$$\widehat{\mathrm{MS}}_k = \widetilde{\mathrm{MS}}_k + g_k \left(P - \tilde{I}\right); k = 1, \ldots, N \tag{24}$$

where $\tilde{I}$ is the component to be substituted, defined as

$$\tilde{I} = \sum_{i=1}^{N} w_i \, \widetilde{MS}_i \tag{25}$$

where $\{w_i\}_{\; k=1,\ldots,N}$ are the corresponding weights indicating the spectral response of each MS band to those of the PAN image [11].

MRA approaches are generally described by the following steps: *(1)* MS bands are interpolated; *(2)* the low-pass version of PAN image (P_L) is estimated through an un-decimated (cf. relation 27) or decimated (cf. relation 28) scheme; (3) the injection gains $\{g_k\}_{\; k=1,\ldots,N}$ are computed for each band; and *(4)* the spatial details are then incorporated into the different interpolated MS bands according to (26)

$$\widehat{\mathrm{MS}}_k = \widetilde{\mathrm{MS}}_k + g_k \left(P - P_{\mathrm{L}}\right); k = 1, \ldots, N \tag{26}$$

where P_{L} is the low-pass version of original PAN image (i.e., approximation).

Considering the un-decimated case, the low-pass approximation P_{L} is expressed by [12]

$$P_{\mathrm{L}} = P \bigotimes h, \tag{27}$$

where $\bigotimes$ is the convolution operator, whereas, in the decimated one [12]

$$P_{\mathrm{L}} = (((P^* \, h) \downarrow R) \uparrow R)^* \, \tilde{h} \tag{28}$$

where h and $\tilde{h}$ are the analysis and the synthesis low-pass filters, respectively. ($\downarrow R$) and ($\uparrow R$) represent the down-sampling and the up-sampling operations by a factor R, which indicates the scale ratio between MS and PAN data.

For the MRA concept, the missing information in MS bands (i.e., high-frequency details) are contained in the PAN image [12]. For this reason, the spectral content of original MS bands is preserved while adding geometrical structures by spatial filtering. However, CS methods preserve better the spatial content comparing to MRA ones, and the spectral quality can be distorted due to the transformation step [13].

6 Experiments and Comparisons

In this section, we introduce the datasets and the algorithms selected for comparisons and finally the quality metrics assessment used for evaluation.

6.1 Datasets

The databases used to evaluate the different fusion approaches were acquired on a semi-urban (Data 1) and an urban (Data 2) areas over Melbourne (Australia) from Pléaides-A1 sensor. They are taken on 4 May 2012. They contain MS images with four spectral bands (blue, green, red, and PIR) at a resolution of 2 m and a gray level PAN image at a resolution of 0.5 m. Dataset specifications are shown in Table 1.

In the different experiments, we consider a subpart of the set of MS images with 300 × 400 pixels and 1200 × 1600 pixels for PAN image (c.f., Figs. 4 and 5). The resolution ratio between the PAN and MS modalities is equal to four.

6.2 Selected Algorithms for Pansharpening

Numerous approaches for pixel-level remote sensing image fusion are available in the literature allowing the enhancement of the geometrical information of MS images using high-resolution PAN image [8]. In the following, we present some pansharpening techniques used in our study, which belongs mainly into the two popular families: the component substitution (CS) [13] and multiresolution analysis

Table 1 Dataset specification

Datasets	Pléaides-A1 (Data 1)	Pléaides-A1 (Data 2)
Location	Melbourne (Australia)	Melbourne (Australia)
Type	Suburban	Urban
Acquisition date	04 May 2012	04 May 2012
PAN/MS resolution	0.5 m/2 m	0.5 m/2 m
PAN size	1200 * 1600	1200*1600
MS size	300 * 400	300 * 400

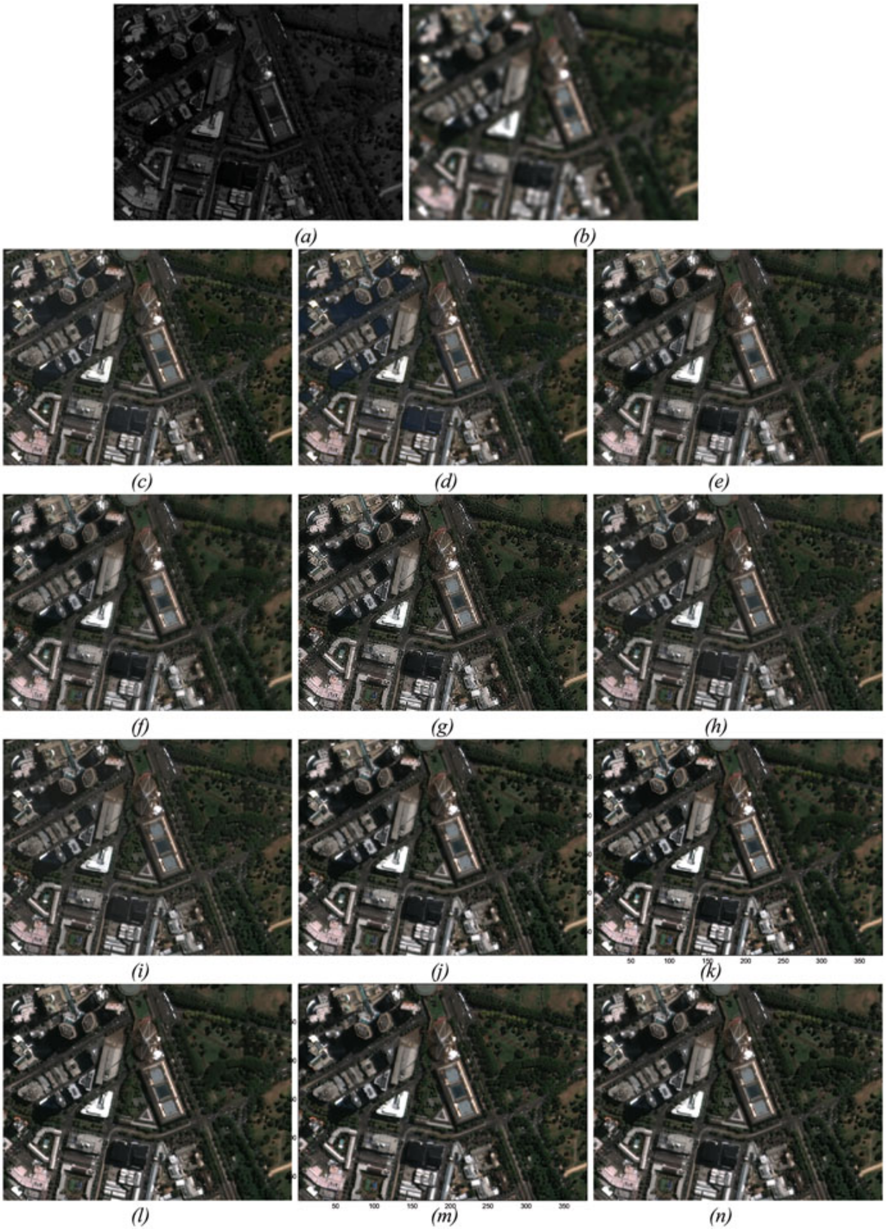


Fig. 4 Pansharpening results at reduced scale (2 m) for the suburban dataset: (**a**) PAN image, (**b**) expanded MS image, (**c**) GIHS, (**d**) Brovey, (**e**) GSA, (**f**) GS2-MTF, (**g**) HPF-Box, (**h**) -HPF-Laplacian, (**i**) SFIM, (**j**) AWT-MTF, (**k**) GLP-MTF, (**l**) GLP-SDM-MTF, (**m**) FB-MTF, and (**n**) FB-SDM-FTM

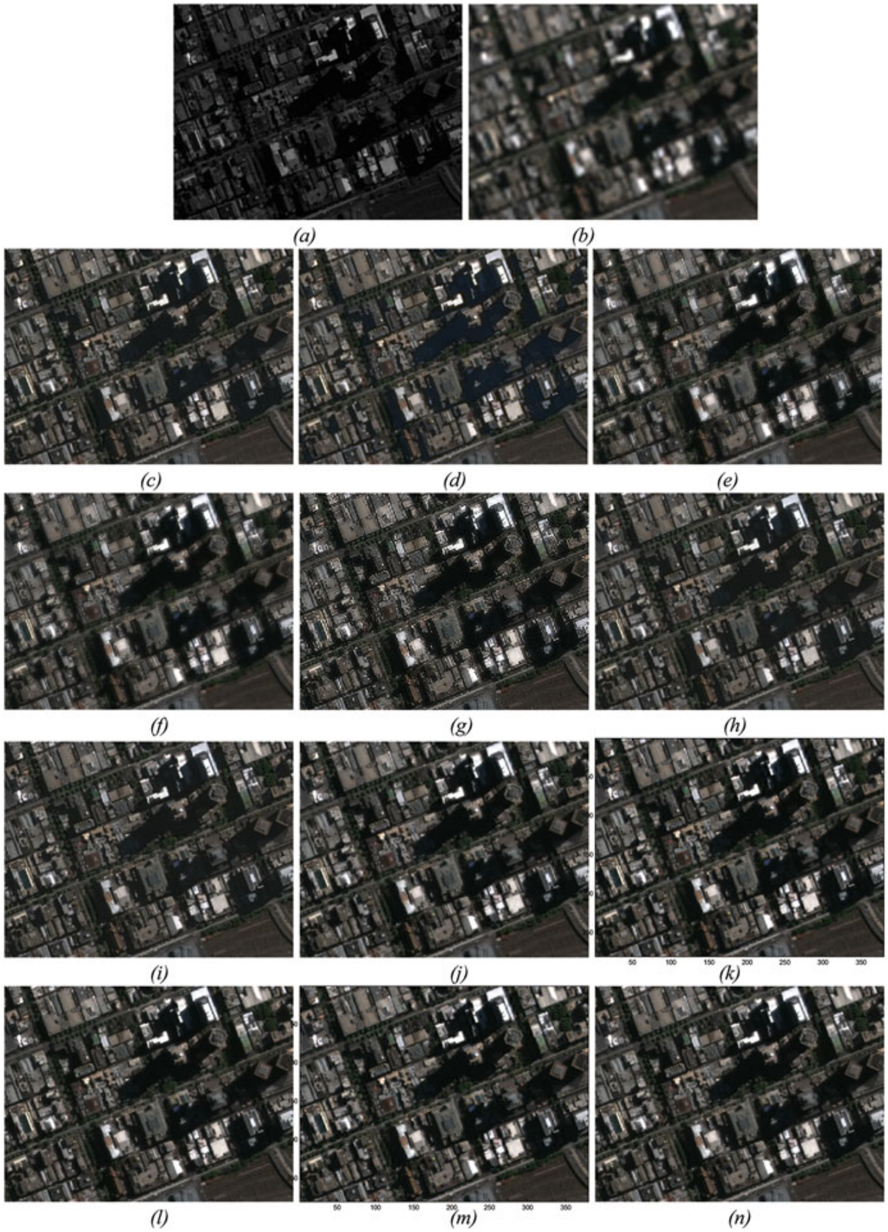


Fig. 5 Pansharpening results at reduced scale (2 m) for the urban dataset: (**a**) PAN image, (**b**) expanded MS image, (**c**) GIHS, (**d**) Brovey, (**e**) GSA, (**f**) GS2-MTF, (**g**) HPF-Box, (**h**) HPF-Laplacian, (**i**) SFIM, (**j**) AWT-MTF, (**k**) GLP-MTF, (**l**) GLP-SDM-MTF, (**m**) FB-MTF and (**n**) FB-SDM-FTM

Table 2 Pansharpening algorithms used for experiments

Method	Name
GIHS [11, 14]	Generalized intensity, hue, and saturation
Brovey [14, 15]	Brovey Transform
GSA [13]	Gram–Schmidt adaptive
GS2 [16]	Gram–Schmidt (mode 2)
HPF-Box [18]	High-pass filtering with 5 × 5 box filter
HPF-Laplacian [18]	High-pass filtering with Laplacian filter
SFIM [19]	Smoothing filter-based intensity modulation
AWT-MTF [21]	Additive wavelet transform "à trous" [21] with MTF-adjusted filter [20]
GLP-MTF [10, 20]	Generalized Laplacian pyramid (GLP) [10] with MTF-adjusted filter [20] and unitary injection model
GLP-SDM-MTF [10, 20]	GLP with MTF-adjusted filter and spectral distortion minimization (SDM) injection model [20]
FB-MTF [1]	Filter banks with MTF-adjusted filter and unitary injection model [1]
FB-SDM-MTF [1]	Filter banks with MTF-adjusted filter [1] and SDM injection model [20]

(MRA) [10]. Table 2 summarizes all pansharpening techniques used for the comparative study.

Generalized Intensity, Hue, and Saturation (GIHS) [14] The standard intensity (I), hue (H) and saturation (S) concept [Carper, 1990], which is limited to RGB color images, is generalized to images with more than three bands (called GIHS, for generalized IHS) [11, 14]. The GIHS pansharpening method allows to substitute the intensity component, obtained by averaging the MS bands, with the histogram-matched PAN image. Then, a backward transformation is performed from HIS space color to multispectral one, producing the high-resolution MS image.

Brovey Transform [14, 15] Brovey transform presumes that the degraded PAN image is a linear combination of different MS bands [15]. In other terms, it means that the range covered by PAN image is similar to the one covered by the totality of MS bands. The pansharpened MS image is obtained as follows: for each pixel, every up-sampled MS band is multiplied by the ratio of the corresponding PAN divided by the sum of MS bands.

Gram–Schmidt (GS) Pansharpening Technique [13, 16] Using GS transform [16], a synthetic low-resolution version of PAN image and MS bands are jointly transformed, creating a new set of GS components. Then, a modified PAN image is created through histogram matching in order to have the same mean and variance as the first component in GS transform. This histogram-matched PAN image replaces therefore the first GS component. Finally, the pansharpening process is accomplished by applying the inverse GS transformation.

According to the manner how to generate the intensity component, several modalities are proposed in the literature. The first one is called GS-mode 1 (GS21). In this case, the intensity component is obtained as a pixel average of all MS bands [16]. For the second modality, called GS-mode 2 (GS2), the low-resolution intensity is generated by performing a low-pass filter to the original PAN image, leading to a hybrid approach. GS2 is considered as MRA-based method [16].

An enhanced version of GS1 is proposed [13], called Gram–Schmidt adaptive (GSA). The intensity component is obtained as a weighted average of the MS channels, using optimal weights, which are estimated as the minimum MSE with respect to the low-pass filtered version of PAN image.

High-Pass Filtering (HPF) [17, 18] For HPF technique, a low-pass approximation version of PAN image is firstly obtained by applying a low-pass filter [17]. Secondly, spatial details are extracted as pixel difference between the original PAN image and its low-pass smoothed version, and they are incorporated into the corresponding up-sampled MS bands.

Several low-pass filters are used in the literature for obtaining the low-pass smoothed version of PAN image: box [19], Gaussian, Laplacian [18], and the modulation transfer function (MTF) matched filters [20].

Smoothing Filter-Based Intensity Modulation (SFIM) [19] SFIM is a pansharpening method with a multiplicative injection model. It permits to transfer the spatial information from PAN image into the corresponding interpolated MS bands without altering its spectral properties. Details are then modulated by a factor called gains, which are given as a ratio computed between the up-sampled MS bands and the low-pass filtered version of PAN image.

Additive Wavelet Transform (AWT) [21] The un-decimated “à trous” technique belongs to MRA class. It is based on redundant wavelet decomposition [21]. This method permits the separation of the low-frequency component (called approximation) from the high-frequency one (wavelet coefficients). The high-resolution MS image is synthetized by adding spatial details, derived from the original PAN image and computed at the different scales, to the coarser approximation (from the low-resolution MS image). Due to its capability to be matched to the sensor MTF [22], this method produces more accurate pansharpened images.

Generalized Laplacian Pyramid (GLP) [10] The GLP approach is used to merge MS and PAN images [8, 10, 23]. Firstly, MS bands are interpolated at the same scale of PAN image. Then, a low-pass filtering and down-sampling operations are performed to original PAN image leading to the approximation (Aiazzi et al. 2002). Geometrical details are computed as pixel-to-pixel difference between PAN image and the interpolated version of its approximation. Finally, they are injected into the interpolated MS images using an additive injection model (by a simple addition between pixel values) or a multiplicative injection model (modulated by gain factors) [10, 20, 24].

Filter-Bank-Based Pansharpening (FB) [1] Hallabia et al. (2016) [1] presented a new concept of multi-modality fusion based on the principle of filter banks matched to the sensor response (i.e., MTF), applied in the field of remote sensing. Under the perfect reconstruction, the low-pass and high-pass filter banks are used for decomposition and reconstruction processing.

The fusion strategy is based mainly on this assumption: the original MS image is considered as a low-pass approximation in order to preserve the spectral quality; and geometrical details are derived from the PAN image using a decomposition filter bank (approximating the MTF). In fact two cases are imposed: The first one is that images are assumed perfectly aligned and the MTF value at the Nyquist frequency is known in order to minimize artifacts (e.g., blur, aliasing and ringing) in the pansharpened products. The second one is to take into account the case of misregistration between PAN and MS images and/or the case that the MTF value is not precisely given. In this case, a preprocessing algorithm is proposed in order to increase the coherence between the MS and PAN images, by generating a new image MS using the GLP algorithm.

6.3 Quality Assessment Metrics

In context of pansharpening, estimating the quality of the merged MS image is a challenging task, since the ground truth is not usually available. Quality assessment can be performed visually (subjective) or quantitatively (objective), including full-reference and no-reference metrics.

Subjective evaluation is a fundamental tool for judging the quality of synthesized images [23, 25]. It consists in comparing visually the fused product with the reference image. Visual comparison can be carried not only on band-to-band reference/fused products but also on the RGB color image. It allows identifying the spatial and spectral artifacts (e.g., pixelization, blur, ringing, and color saturation) in the pansharpened products.

For the objective quality measures, numerous metrics are proposed in the literature, which are divided into two categories: full-reference indexes using Wald protocol [26] or Zhou protocol [27] and no-reference indexes based on Quality with No reference (QNR) protocol [28]. In the first case, the quality of fused products is evaluated using the reference image (i.e., the original image is considered as a reference), which is not available in the second one.

Full-Reference Quality Assessment Using the original reference MS image, full-reference indexes give information about the fidelity and the consistency of such a fusion method. For judging quantitatively the performances of different pansharpening methods, the following quality metrics are utilized:

- Spectral angle mapper (SAM) [26] quantifies the spectral distortion of the merged product. It is defined as the absolute value of the angle between the reference and the pansharpened vectors. Its ideal value is equal to zero.
- Relative dimensionless global error in synthesis (ERGAS) [26] measures the radiometric distortion. It should be as low as possible (i.e., ideal value is zero which indicates the similarity between MS data).

- The quaternion-based coefficient Q4 is proposed by [29] indicating the global quality of the merged MS data. It quantifies the spectral and the radiometric quality. The value of Q4 index should be as high as possible (ideally one) [26].
- The spatial correlation coefficient index (SCC) is proposed by [27] in order to measure the spatial quality. It is used to quantify the amount of the high-frequency details which are injected into the interpolated MS bands during the merging process. This index compares the spatial details extracted from original and pansharpened MS images by means of a Laplacian filter.

No-Reference Quality Assessment As mentioned above, no-reference quality assessment does not require the reference MS image. In fact, the authors [28] proposed the QNR protocol in order to evaluate the quality of pansharpened products at full scale. It involves two indexes to measure separately the spectral distortion (D_λ) and the spatial distortion (D_s), given MS image at coarse resolution and PAN image at finer resolution. D_λ index is estimated between original MS bands at coarse resolution and the pansharpened MS bands at finer resolution (at the same scale of PAN image). D_s index is computed between each MS band and PAN image at low and high resolution. These two indexes can be combined together obtaining a global similarity quality index, denoted *QNR*. The ideal value for both indexes D_λ and D_s is 0, while it is 1 for *QNR*.

7 Results

In this section, we will present the experimental results using full-resolution reference (at reduced scale) and no-reference quality metrics (at full scale).

7.1 Reduced Scale Results

Here, we present the quality scores computed on different pansharpened MS, in order to evaluate the performance of the fusion algorithms based on full-reference quality assessment. In fact, we use the Wald protocol [26] (SAM, ERGAS, and Q_4 indices) for estimating the spectral quality and the SCC index proposed by [27] for spatial quality.

Tables 3 and 4 report comparative quality indexes computed between 2-m fused MS and 2-m reference original MS data using full-reference quality metrics. In addition to quantitative indices, the fused results at reduced scale (2 m) for the suburban and the urban dataset are illustrated in Figs. 4 and 5, respectively, in order to evaluate the visual quality of the final products.

For evaluating the spectral and spatial quality of pansharpened data, we follow firstly the validation protocols proposed in [26] and [27] at reduced scale. These two

Table 3 Quality scores between the original reference 2-m MS and the enhanced MS bands obtained from 2-m PAN and 8-m MS (suburban dataset)

	ERGAS	SAM	Q4	SCC
Reference	0	0	1	1
GIHS [11, 14]	4.258	5.015	0.940	0.731
Brovey [15]	4.160	4.457	0.942	0.749
GSA [13]	4.187	4.932	0.943	0.734
GS2-MTF [16]	3.563	4.664	0.964	0.720
BOX-HPF [18]	4.010	4.501	0.951	0.682
HPF-Laplacian [18]	7.832	6.600	0.853	0.452
SFIM [19]	3.906	4.085	0.954	0.705
AWT-MTF [21]	3.355	4.403	0.965	0.725
GLP-MTF [10]	3.346	4.433	0.966	0.727
GLP-MTF-SDM [10, 20]	3.142	3.863	0.972	0.751
FB-MTF [1]	2.951	3.652	0.974	0.731
FB-MTF-SDM [1]	**2.802**	**3.435**	**0.978**	**0.754**

Table 4 Quality scores between the original reference 2-m MS and the enhanced MS bands obtained from 2-m PAN and 8-m MS (urban dataset)

	ERGAS	SAM	Q4	SCC
Reference	0	0	1	1
GIHS [11, 14]	4.734	4.280	0.940	0.751
Brovey [15]	4.757	3.925	0.939	0.748
GSA [13]	4.702	4.283	0.942	0.755
GS2-MTF [16]	3.748	3.003	0.970	0.744
BOX-HPF [18]	4.489	3.408	0.952	0.713
HPF-Laplacian [18]	8.114	4.753	0.869	0.480
SFIM [19]	4.432	3.322	0.954	0.717
AWT-MTF [21]	3.560	3.004	0.972	0.757
GLP-MTF [10]	3.490	2.962	**0.973**	0.727
GLP-MTF-SDM [20]	3.383	2.938	0.975	0.751
FB-MTF [1]	3.111	**2.616**	0.979	0.731
FB-MTF-SDM [1]	**3.072**	2.780	0.980	**0.754**

procedures require a reference image in order to validate the pansharpening methods. To this end, the original MS image is considered as a reference and the pansharpening algorithms are performed using simulated data (i.e., a spatially degraded version of the original PAN and MS data). Therefore, according to [26], the obtained fusion product must be as identical as possible to the original MS image in terms of spectral quality.

Comparing the two categories (i.e., CS and MRA methods), we notice that globally MRA methods present a better spectral quality than to those belonging to CS category. Moreover, CS methods are affected by spectral and radiometric distortions, since they present high values of SAM and ERGAS indexes.

FB-SDM-MTF presents the best spatial and spectral quality, thanks to the control of the amount of spatial structures injected into the up-sampled MS bands, followed by the techniques based on MTF (GLP-SDM-MTF, GLP-MTF, GS2-MTF and AWT-MTF) and those based on the high filtering injection (HPF-BOX, HPF-Laplacian, and SFIM). These results are explained by the fact that the algorithms take into account the MTF response approximating the sensor response in the frequency domain. However, the box filter is characterized by the presence of ripples in the frequency domain that is confirmed by the lowest value of SCC index (i.e., 0.695).

Finally, it is shown in Tables 3 and 4 that the fusion methods based on adaptive injection scheme (i.e., SDM model) present better results comparing to global ones.

7.2 Full-Scale Results

In this section, the full-resolution dataset images are considered. In this case, reference image is not available. The no-reference quality metrics are used in order to evaluate the performance of the different fused MS images. Therefore, the QNR protocol [28] allows estimating the spectral quality and the spatial quality.

Tables 5 and 6 show the estimated quality indexes of different pansharpening algorithms at full-scale protocol, using no-reference quality metrics. As mentioned above, $\boldsymbol{D_\lambda}$ and $\boldsymbol{D_s}$ indexes are used to quantify the spectral and spatial distortions, respectively. The ideal value for $\boldsymbol{D_\lambda}$ and $\boldsymbol{D_s}$ is 0, while it is 1 for QNR.

Notice that the method FB-MTF-SDM provides the best results in terms of ***QNR*** indices. Furthermore, it presents the lowest spectral and spatial distortion, mainly in the suburban dataset. This result is explained by the best control of the similarity between MS and PAN when passing from coarse to fine resolutions [1].

Assessing the quality on different image contents (i.e., suburban and urban data), we note that the quantitative results according to suburban data are better compared to those on urban one. In fact, the urban area presents very high resolution details (i.e., building edges) which are difficult to recover.

Table 5 Full-resolution quality indexes estimated for the suburban dataset

	D_λ	D_s	QNR
GIHS [11, 14]	0.069	0.054	0.881
Brovey [15]	0.037	0.048	0.917
GSA [13]	0.076	0.052	0.876
GS2-MTF [16]	0.074	0.038	0.891
BOX-HPF [18]	0.051	0.028	0.922
HPF-Laplacian [18]	0.087	0.041	0.875
SFIM [19]	0.052	0.033	0.917
AWT-MTF [21]	0.067	0.044	0.891
GLP-MTF [10]	0.069	0.047	0.887
GLP-MTF-SDM [20]	0.070	0.039	0.893
FB-MTF [1]	0.064	0.034	0.904
FB-MTF-SDM [1]	**0.049**	**0.018**	**0.934**

Table 6 Full-resolution quality indexes estimated for the urban dataset

	D_λ	D_s	QNR
GIHS [11, 14]	0.065	0.046	0.892
Brovey [15]	0.021	0.038	0.942
GSA [13]	0.084	0.037	0.881
GS2-MTF [16]	0.065	0.017	0.919
BOX-HPF [18]	0.047	0.067	0.890
HPF-Laplacian [18]	0.085	0.013	0.904
SFIM [19]	0.040	0.074	0.888
AWT-MTF [21]	0.059	0.006	0.935
GLP-MTF [10]	0.061	0.008	0.931
GLP-MTF-SDM [20]	0.054	0.006	**0.941**
FB-MTF [1]	0.060	**0.007**	0.933
FB-MTF-SDM [1]	**0.045**	0.019	0.937

Finally, the results using no-reference quality metric (i.e., at full resolution) results are in agreement with those obtained using full-reference one (i.e., at reduced scale).

8 Discussion

Pansharpening techniques belonging to CS family (e.g., GIHS, Brovey, GSA) are generally fast and easy to implement. In addition, the synthesized MS images are characterized by a high fidelity in terms of the spatial content [13]. These approaches are well suited for mapping applications [30]. However, spectral distortions can be produced in the fused product, which are due to the existence of local dissimilarities between the two PAN and MS modalities [13, 18]. This is explained physically by the non-correspondence between the spectral ranges of the different MS bands and that of the PAN image [31]. On the other hand, they suffer from radiometric distortion due to a modification of the low frequencies of the original MS image. This can alter the spectral signatures (i.e., colors) of the synthesized MS modalities [32]. Indeed, they are inadequate for studying vegetated areas (e.g., agricultural regions).

Regarding AMR-based approaches, they preserve a better spectral content because they affect only the high-frequency components, and those at low frequencies are retained [10]. However, they are affected by spatial distortions, especially on images of highly textured urban areas [32]. In fact, the fusion product may be affected by artifacts such as blur or aliasing or the phenomenon of Gibbs [10] due to resampling and spatial filtering operations.

In addition, the pansharpening methods based on the injection of the high-frequency components into the up-sampled MS bands demonstrated their superiority by comparing to the CS family, which is confirmed in [33]. Moreover, adaptive injection methods are used to refine the fused results compared to the global methods (i.e., same detail or gain applied to the interpolated MS bands) [24]. They take into account the homogeneity or the texture of the considered data.

9 Conclusions

In this chapter, the two-channel filter bank configuration is presented. Under the perfect reconstruction and aliasing cancelation, different solutions are presented. Based on this theory, pansharpening is then considered as an application, since the authors have been proposed an algorithm for fusing remotely sensed imagery.

In the context of pansharpening, a comparative study is studied. Indeed, the quality of several methods has been studied on high-resolution Pléaides-A1 dataset. For validating the fused images, full-reference (using Wald and Zhou protocol) and no-reference (QNR protocol) quality assessment metrics have been considered. Experimental results show that CS and MRA methods present complementary results in the context of spectral and spatial quality.

References

1. Hallabia, H., Kallel, A., Ben Hamida, A., & Le Hegarat-Mascle, S. (2016). High spectral quality’ pansharpening approach based on MTF-matched filter banks. *Multidimensional Systems and Signal Processing, 27*(4), 831–861.
2. Croisier, A., Esteban, D., & Galand, C. (1976). Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques. In *International Conference on Information Sciences and Systems*, Patras.
3. Esteban, D., & Galand, C. (1977). Application of quadrature mirror filters to split band voice coding schemes. *IEEE international conference on acoustics, speech, and signal processing, ICASSP ’77, 2*, 191–195.
4. Vetterli, M., & Kovacevic, J. (1995). *Wavelets and subband coding*, *Prentice-Hall signal processing series*. Englewood Cliffs, NJ: Prentice-Hall.
5. Vetterli, M. (1986). Filter banks allowing perfect reconstruction. *Signal Processing (Elsevier), 10*(3), 219–244.
6. Vaidyanathan, P. (1990). Multirate digital filters, filter banks, polyphase networks, and applications: A tutorial. *Proceedings of the IEEE, 78*(1), 56–93.
7. Oppenheim, A. V., & Schafer, R. W. (2009). *Discrete-time signal processing* (3rd ed.). Upper Saddle River: Pearson Higher Education, Inc.
8. Vivone, G., Alparone, L., Chanussot, J., Dalla Mura, M., Garzelli, A., Licciardi, G. A., Restaino, R., & Wald, L. (2015). A critical comparison among pansharpening algorithms. *IEEE Transactions on Geoscience and Remote Sensing, 53*(5), 2565–2586.
9. Smith, M., & Barnwell, T. (1987). A new filter bank theory for time-frequency representation. *IEEE Transactions on Acoustics, Speech, and Signal Processing, 35*(3), 314–327.
10. Aiazzi, B., Alparone, L., Baronti, S., & Garzelli, A. (2002). Context-driven fusion of high spatial and spectral resolution images based on oversampled multiresolution analysis. *IEEE Transactions on Geoscience and Remote Sensing, 40*(10), 2300–2312.

11. Tu, T. M., Huang, P. S., Hung, C. L., & Chang, C. P. (2004). A fast intensity-hue-saturation fusion technique with spectral adjustment for ikonos imagery. *IEEE Geoscience and Remote Sensing Letters, 1*(4), 309–312.
12. Alparone, L., Baronti, S., & Aiazzi, B. G. A. (2016). Spatial methods for multispectral pansharpening: Multiresolution analysis demystified. *IEEE Transactions on Geoscience and Remote Sensing, 54*(5), 2563–2576.
13. Aiazzi, B., Baronti, S., & Selva, M. (2007). Improving component substitution pansharpening through multivariate regression of MS +pan data. *IEEE Transactions on Geoscience and Remote Sensing, 45*(10), 3230–3239.
14. Tu, T. M., Su, S. C., Shyu, H. C., & Huang, P. S. (2001). A new look at IHS-like image fusion methods. *Information Fusion, 2*(3), 177–186.
15. Gillespie, A. R., Kahle, A. B., & Walker, R. E. (1987). Color enhancement of highly correlated images. Ii. Cannel ratio and chromaticity transformation techniques. *Remote Sensing of Environment, 22*(3), 343–365.
16. Laben, C. A., & Brower, B. V. (2000). Process for enhancing the spatial resolution of multispectral imagery using pansharpening, US Patent 6,011,875.
17. Schowengerdt, R. A., & Sensing, R. (2007). *Models and methods for image processing* (3rd ed.). Burlington: Academic Press.
18. Amro, I., Mateos, J., Vega, M., Molina, R., & Katsaggelos, A. K. (2011). A survey of classical methods and new trends in pansharpening of multispectral images. *EURASIP Journal on Advances in Signal Processing, 2011*(1), 79. 1–79:22.
19. Liu, J. G. (2000). Smoothing filter-based intensity modulation: A spectral preserve image fusion technique for improving spatial details. *International Journal of Remote Sensing, 21* (18), 3461–3472.
20. Aiazzi, B., Alparone, L., Baronti, S., Garzelli, A., & Selva, M. (2006). MTF-tailored multiscale fusion of high resolution MS and pan imagery. *Photogrammetric Engineering and Remote Sensing, 72*(5), 591–596.
21. Nunez, J., Otazu, X., Fors, O., Prades, A., Pala, V., & Arbiol, R. (1999). Multiresolution-based image fusion with additive wavelet decomposition. *IEEE Transactions on Geoscience and Remote Sensing, 37*(3), 1204–1211.
22. Aiazzi, B., Alparone, L., Baronti, S., Garzelli, A., & Selva, M. (2012). Advantages of Laplacian pyramids over “à trous” wavelet transforms for pansharpening of multispectral images. *Proceedings of SPIE The International Society for Optical Engineering, 8537*(10), 853704.
23. Alparone, L., Wald, L., Chanussot, J., Thomas, C., Gamba, P., & Bruce, L. M. (2007). Comparison of pansharpening algorithms: Outcome of the 2006 GRSS-S data-fusion contest. *IEEE Transactions on Geoscience and Remote Sensing, 45*(10), 3012–3021.
24. Aiazzi, B. B. S., Lotti, F., & Selva, M. (2009). A comparison between global and context-adaptive Pansharpening of multispectral images. *IEEE Geoscience and Remote Sensing Letters, 6*(2), 302–306.
25. Thomas, C., Ranchin, T., Wald, L., & Chanussot, J. (2008). Synthesis of multispectral images to high spatial resolution: A critical review of fusion methods based on remote sensing physics. *IEEE Transactions on Geoscience and Remote Sensing, 46*(5), 1301–1312.
26. Wald, L., Ranchin, T., & Mangolini, M. (1997). Fusion of satellite images of different spatial resolutions: Assessing the quality of resulting images. *Photogrammetric Engineering and Remote Sensing, 63*(6), 691–699.
27. Zhou, J., Civco, D. L., & Silander, J. A. (1998). A wavelet transform method to merge landsat tm and spot panchromatic data. *International Journal of Remote Sensing, 19*(4), 743–757.
28. Alparone, L., Aiazzi, B., Baronti, S., Garzelli, A., Nencini, F., & Selva, M. (2008). Multispectral and panchromatic data fusion assessment without reference. *Photogrammetric Engineering and Remote Sensing, 74*(2), 193–200.
29. Alparone, L., Baronti, S., Garzelli, A., & Nencini, F. (2004). A global quality measurement of pan-sharpened multispectral imagery. *IEEE Geoscience and Remote Sensing Letters, 1*, 313–317.

30. Yocky, D. A. (1996). Artifacts in wavelet image merging. *Optical Engineering, 35*, 2094–2101.
31. Xu, Q., Zhang, Y., & Li, B. (2014). Recent advances in pansharpening and key problems in applications. *International Journal of Image and Data Fusion, 5*(3), 175–195.
32. Aiazzi, B., Alparone, L., Baronti, S., Garzelli, A., & Selva, M. (2011). Twenty-five years of pansharpening: A critical review and new developments. In *Signal and image processing for remote sensing* (pp. 533–548). Boca Raton, FL: Taylor and Francis Books.
33. Wang, Z., & Bovik, A. C. (2002). A universal image quality index. *IEEE Signal Processing Letters, 9*, 81–84.

Video Signal Processing

Yung-Lin Huang

1 Introduction

Nowadays, multimedia information systems become more popular. People love to watch images and videos on different devices such as television (TV), personal computer (PC), and mobile phone. The variety of display formats rapidly increase. This fact has resulted in a demand for efficiently converting between various video formats.

We briefly explain the format of the video signal. An image consists of two-dimensional (2D) signals called *pixels* which represent colors in a point of the picture, and a video is composed of a sequence of images called frames. That is, the consecutive frames form a video as shown in Fig. 1. Therefore, a video is a three-dimensional (3D) signal including x-direction, y-direction, and temporal domain. Frame rate, expressed in frame per second (FPS), is the frequency that the consecutive frames are shown on the display device. The frame rate differs between lots of video formats. For example, movie films are at 24 FPS, and the videos recorded by mobile phone are often at 30 or 60 FPS.

In addition to the frame rate, the refresh rate of display devices should be took into consideration. Liquid crystal display (LCD), which is one of the most important display techniques, is widely used in many display devices. To provide high visual quality videos, LCD is capable of displaying high frame rate videos at 120 FPS or more. However, the video frame rate differs from many sources and is often lower than the refresh rate.

Y.-L. Huang
National Taiwan University, Taipei City, Taiwan
e-mail: cary@media.ee.ntu.edu.tw

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9_6

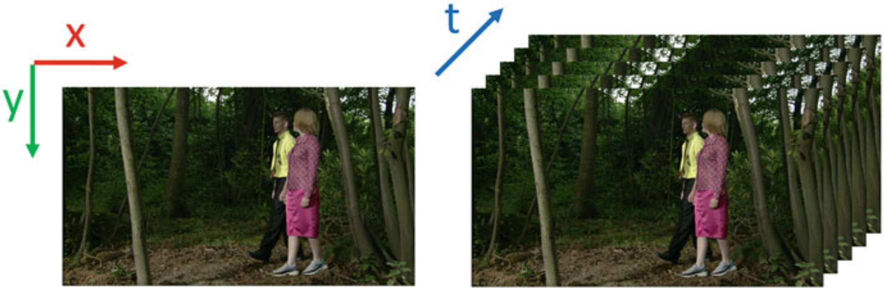


Fig. 1 Illustration of image and video signal. A video signal consists of a sequence of images. It is a three-dimensional signal including x-direction, y-direction, and temporal domain

Frame rate conversion (FRC) is a post-processing technique to convert the frame rate. The technique aims to increase or decrease the video frame rate for different applications. Because increasing the frame rate introduces smoother motion in videos, frame rate up-conversion (FRUC) is often operated to convert the frame rate from a lower number to a higher one. The technique is useful for lots of applications that are stated in the following section.

This chapter is organized as follows. First, we introduce several multirate video applications in Sect. 2 to show the importance of the multirate video techniques. Several fundamental FRC techniques are first presented in Sect. 3. Then, Sect. 4 explains the motivation of higher frame rate, which gives the reason why the FRUC techniques are most widely adopted. Then, we present the flow diagram and details in each part of state-of-the-art FRUC techniques in Sect. 5. The evaluation methods for the FRUC techniques are then introduced in Sect. 6. Moreover, Sect. 7 demonstrates the importance of hardware architecture design and recent research results. Finally, a conclusion and further discussion are given in Sect. 8.

2 Multirate Video Applications

There are several video applications that require the FRC techniques. Here we list and introduce four of them in the following.

2.1 Video Format Conversion

As mentioned previously, videos are required to be displayed on many different devices. In different digital video standards, the specification of frame rate differs. There are many variations, and new specifications are also introduced by emerging standards. It varies from 24, to 30, to 60, and even to 120 or higher FPS. Converting the frame rate between different standards is essential.

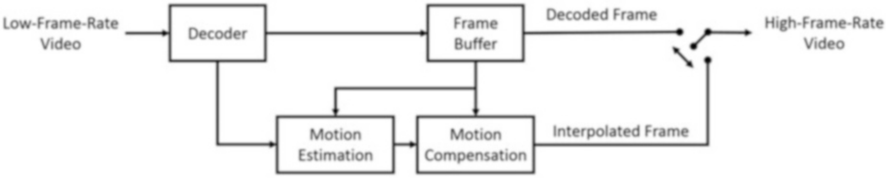


Fig. 2 The frame rate up-conversion technique for low bit rate video coding. The frames are skipped before transmission and interpolated after reception.

2.2 Low Bit Rate Video Coding

To efficiently transmit a video sequence, video coding such as H.264 [1] and high efficient video coding (HEVC) [2] should be applied. Moreover, decreasing the frame rate sufficiently lower the transmitted bit rate because some frames are skipped. After receiving the video, its frame rate can be up-converted to preserve the visual quality when displaying. The decoded and interpolated frames are alternatively displayed as shown in Fig. 2.

2.3 Video Editing

When editing a video, changing the frame rate creates impressive visual effects in a scene with moving objects. To be more specific, decreasing and increasing the frame rate creates time-lapse and slow-motion videos, respectively. Note that the edited video should be displayed at the original frame rate. Take slow-motion effects as an example; increasing the frame rate from 30 to 60 FPS means that the numbers of frame are doubled. The motion lasting 1 s becomes 2 s when displaying the video at the original 30 FPS. This indicates that the motion is slowed down.

2.4 High Refresh Rate Display Device

To provide better visual quality with smooth motion, some off-the-shelf display devices are able to display videos at 120 FPS and higher. Since most videos are at lower frame rate, the FRUC technique should be operated to fill the gap. Figure 3 shows an example of the application. Several frames are interpolated before the video is displayed.

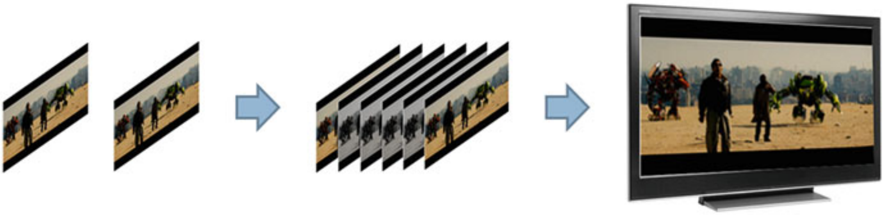


Fig. 3 The frame rate up-conversion technique increases video frame rate for high refresh rate display devices. The four intermediate frames are interpolated to achieve five times up-conversion

3 Fundamental Frame Rate Conversion

We first present several fundamental FRC techniques. Since implementing these techniques is simple, they are still employed in some systems with limited resources.

3.1 Frame Duplicate and Skipping

Different sampling methods introduce different frame rates. When up-converting the frame rate, a straightforward way is to duplicate frames. That is, sampling a frame more than once in the same period to achieve the target higher frame rate. For example, duplicating each frame once creates a video at doubled frame rate. On the contrary, skipping an appropriate number of frames down-converts the frame rate. Frames are often automatically skipped when the resources are limited.

3.2 Three-Two Pull Down

A popular method for changing the sample rate on an original video to achieve a different frame rate is the three-two pull down technique [3]. It converts videos from movie film at 24 FPS to the widely used 30 FPS as shown in Fig. 4. The frames are split into two fields noted as *a* and *b* in Fig. 4. One field contains odd rows and the other contains even rows of the frame. Then, the fields are recombined into frames for video at 30 FPS. The three-two pull down technique causes interlacing problems which is not discussed here.

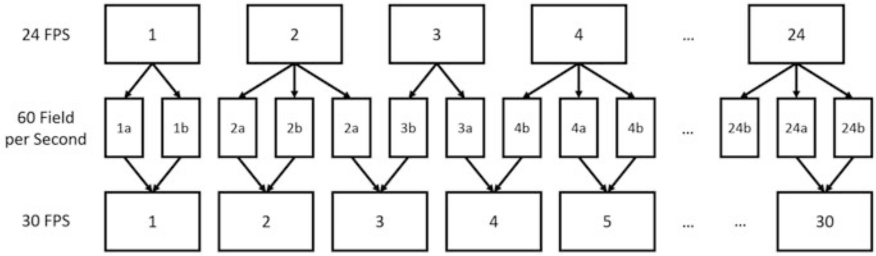


Fig. 4 Illustration of three-two pull down to convert video at 24 FPS to 30 FPS. Four frames are converted to five frames. For each original frame, it is separated into two fields noted as *a* and *b*. The fields are then recombined

3.3 Frame Insertion

Increasing the frame rate by inserting meaningful frames benefits visual quality. Some systems apply black frame insertion or frame blending. *Black frame insertion* means that a frame filled with black color is inserted. The purpose is to reduce the motion blur problem introduced in the next section. *Frame blending* inserts a frame $f(t)$ between frame $t - 1$ and frame $t + 1$. It utilizes the blending function $f(x, y, t)$ for the pixels of inserted frames,

$$f(x, y, t) = \frac{\lambda_1 f(x, y, t-1) + \lambda_2 f(x, y, t+1)}{2}$$

The (x, y, t) mean the spatial and temporal position of a pixel. It blends the pixels at the same position in the previous and next frames. The parameters λ_1 and λ_2 can be adjusted to be adaptive weighting.

Since videos always present moving objects or scene, blending the pixels at the same position often causes blurred pixels. Therefore, first estimating the motion in videos plays an important role for frame insertion. Motion-compensated FRUC, abbreviated as MC-FRUC or simply FRUC, is the essential technique. In the following section, we state more details of the motivation before introducing the FRUC techniques.

4 Motivation of Higher Frame Rate

As mentioned previously, the FRUC technique is important because video at higher frame rate can display smoother motion. This also indicates better visual quality. We explain this in the aspects of LCD motion blur problems. The visual quality of LCDs suffers from motion blur due to the physical property of the liquid crystal. In general, two types of motion blur occur in LCDs [4].

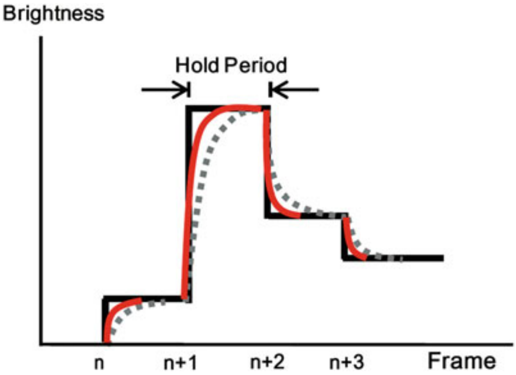


Fig. 5 Hold-type display with slow response. The black solid, the dotted, and the red solid lines represent the target, displayed, and overdrived brightness, respectively

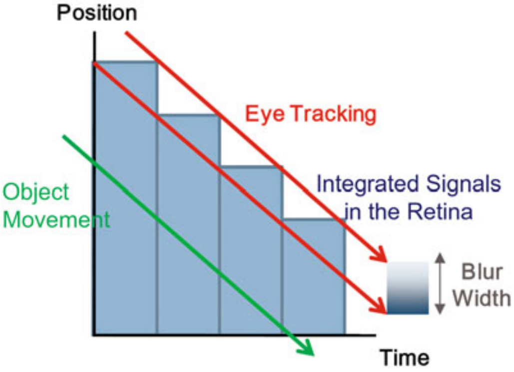


Fig. 6 Direct evaluation of the blur width. The blur width is caused by the integrated signals in the retina

The first motion-blur type is caused by the slow response of the liquid crystal. Figure 5 shows that the black solid line indicates the target brightness and the dotted line indicates the actual displayed brightness. The smooth variation in brightness appears blurred as perceived by human eyes. To overcome this problem, a popular solution called overdrive is applied, i.e., the voltage is first set higher or lower to approach the target brightness and is then set back to the ordinary value after the brightness approaches the target. The red solid line in Fig. 5 shows the resulting brightness, which demonstrates a reduction in the smooth variation in brightness.

The second type is called the hold-type motion blur. Figure 5 shows that maintaining the brightness is termed as the "hold period," which is equal to the inverse of the frame rate. Figure 6 shows that when human eyes track the movement of an object with velocity v, the intensity is continuously integrated in the retina, but the actual intensity discretely changes. This divergence causes the integrated signals in the object boundary in the retina to smoothly decrease or increase. The range of the decrease or increase is called the blur width and can be directly expressed as

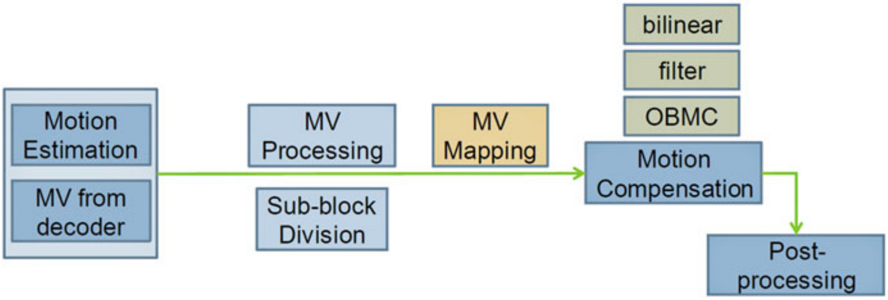


Fig. 7 General flow diagram of the FRUC technique. The MVs are estimated using the ME or retrieved from a video decoder, after which the MVs are refined and mapped to interpolate the intermediate frames using the MC

$$\text{Blur width}_{\text{direct}} = \frac{v}{\text{frame rate}}$$

Another method of evaluating the hold-type motion blur is based on the sampling and reconstruction theory of integrated signals in the retina [5]. In the case of an idle display, the blur width is equal to

$$\text{Blur width}_{\text{theoretical}} = 0.8 \times \frac{v}{\text{frame rate}}$$

Therefore, blur width is inversely proportional to the frame rate. Among the solutions for the hold-type motion blur, increasing the frame rate is regarded as the most efficient method because it can directly reduce the effect of motion blur without drop in visual quality [4].

5 Frame Rate Up-Conversion

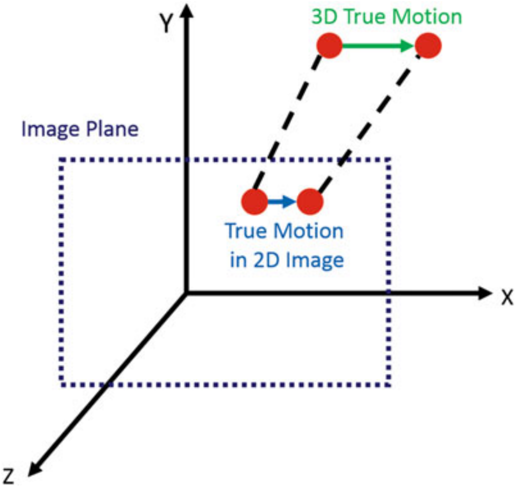
In this section, we give an overview of FRUC techniques and the details in each part. Figure 7 shows the general FRUC flow. Initially, motion vectors (MVs) between existing frames are required. In general, MVs are derived from a video decoder or by performing motion estimation (ME). The MVs can be estimated block-based, sub-block-based, or even pixel-based. Then, all derived MVs in a frame form an MV field (MVF). To display a more realistic and detailed motion, further MV processing may be performed on the MVFs. After the MVFs are retrieved, they must be mapped from the existing frames to the target intermediate frames because of temporal mismatch. Thereafter, the intermediate frames are interpolated according to the mapped MVFs using motion compensation (MC) techniques. Finally, the interpolated frames are post-processed to achieve better visual quality.

The discussions of the functions related to each part are listed below. Five parts are included: ME, MV processing, MV mapping, MC, and post-processing. In each part, we first discuss related works and then introduce several representative and state-of-the-art methods.

5.1 Motion Estimation

Illustration of motion in an image is shown in Fig. 8. A motion in 3D space is projected into the image plane and becomes a 2D vector. The 2D vector corresponding to the 3D true motion is called true MV. True ME is the method aiming to estimate the true MVs between two frames. On the other hand, conventional ME required for residual minimization is the key technique in video encoder to find out the most similar blocks for video compression. As shown in Fig. 9, both the MVs are able to be used for compression. However, unlike the ME in a video encoder, performing true ME between frames aims to determine the true motion

Fig. 8 Illustration of 3D motion in real world projected into image space



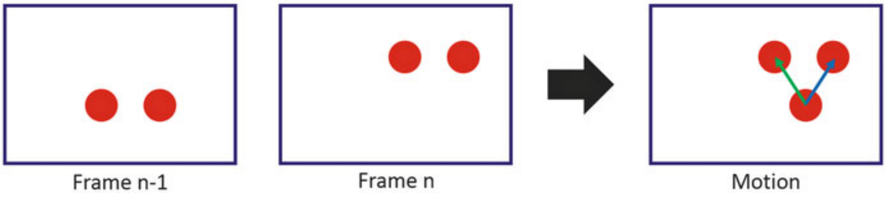


Fig. 9 Illustration of two consecutive frames with two moving red balls. This describes different goals of ME. For compression, ME using the two MVs performs the same because they both match the pixel values. However, only one MV represents the true motion of this object

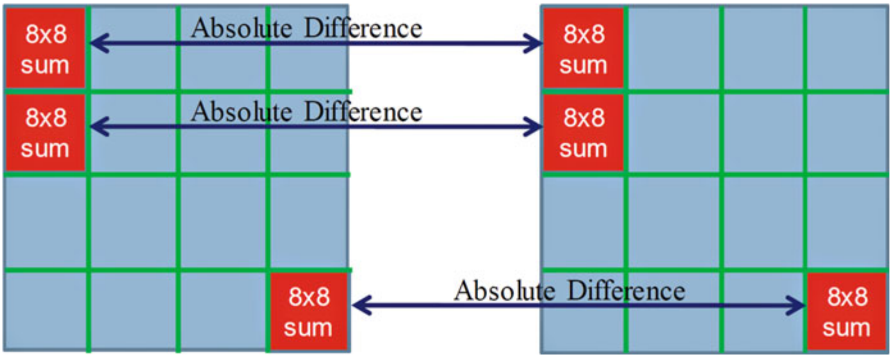


Fig. 10 An example of the 8 × 8 MSEA computation. The 32 × 32 block is composed of 16 sub-blocks, and each sub-block has 8 × 8 pixels. The absolute difference between the sub-block pairs are computed and then accumulated

that demonstrates the object movement [6], instead of reducing only the residual energy.

Many related works approximate the true MVFs using block-based ME with spatial and temporal predictions [6–9]. To achieve more accurate MVFs, Min and Sim [10] introduces a confidence level of blocks, Liu et al. [11] uses a multiple hypotheses Bayesian scheme, and Kaviani and Shirani [12] adopts optical flow estimation method. However, the true MVF is difficult to estimate, and the computational cost is usually high. On the other hand, we can possibly retrieve the MVF from a video decoder [13–15] and then perform MV processing to optimize the rough MVFs. Nevertheless, the decoding information is not always available for FRUC in the current LCD systems. We introduce three types of ME techniques in the following.

Block matching ME is a memory-efficient and hardware-friendly technique. Separating a frame into several blocks and then computing the difference between current block and neighboring blocks in the other frame. The conventional matching criterion is the sum of absolute difference (SAD). For current block A at (x, y), discover the block B at $(x + s, y + t)$ minimizing the SAD computed as

$$\mathrm{SAD} = \sum_{\text{All pixels}} \left|A_{x,y} - B_{x+s,y+t}\right|$$

The vector (s, t) indicates the MV of the current block. To efficiently compute the difference, the down-sampled version of the SAD, multilevel successive elimination algorithm (MSEA) [16], can be adopted. An example of computing 8 × 8 MSEA between two 32 × 32 blocks is shown in Fig. 10.

To keep the robustness and consistency between neighboring blocks, the prediction and special search patterns are usually applied. A ME technique with median prediction and hybrid search pattern is introduced in [17]. The ability to reject the predictor and reestimate from the origin can prevent estimation of incorrect MVs due to incorrect predictors. In addition, the method is cost-efficient owing to its early convergence. The pseudocode is shown as follows. Here *SP* means square pattern, and ε means minimum distortion of the applied *SP*. *MV* is used as the center of *SP*, and the *threshold* is equal to 1024 in the implementation.

```
Step 1    Set $MV$ = median of three neighboring blocks' $MVs$
Step 2    Apply 4-step SP on $MV$
          If $\varepsilon$ is found at the center or $\varepsilon < threshold$
              Apply 2-step and 1-step SPs for converge
          Else
              Set $MV$ = origin, go to Step. 3
Step 3    Apply 8-step SP on $MV$
          If $\varepsilon$ is not found at the center
              Set $MV$ = the position with $\varepsilon$, repeat Step. 3
          Else
              Apply 4-step, 2-step and 1-step SPs for converge
```

This search strategy is similar to a hybrid search algorithm with four-step search (4SS) [18] and three-step search (3SS) [19]. The square search pattern contains the centering block and eight neighboring blocks, and the distance between two blocks is represented by step-size. At first, a predictor is given by the median of three neighboring (left, up, and upper-right) MVs. Then a 4-step square search pattern centering on the predictor is employed. If the minimum distortion appears at the center or its value is smaller than the threshold, the predictor will be regarded as good and proceed to apply 2-step and 1-step square patterns for converge, like 3SS. Otherwise, it go back to the origin and search MV like 4SS but with an 8-step square pattern. If the minimum distortion is found at the center of an 8-step square pattern, 4-step, 2-step, and 1-step square patterns are employed for converge .

Optical flow estimation, a pixel-based ME method first proposed by Lucas and Kanade [20], can be used in place of block matching ME. This avoids blocky artifacts of block matching algorithms, but generates salt-and-pepper artifacts [21]. Recently, Lee et al. [22] proposed a FRUC framework which estimates four sets of MVF using a modified optical flow algorithm. Multiple intermediate frames are reconstructed and fused to obtain the final frame. The MVF $\boldsymbol{V_t}$ between frame $\boldsymbol{F_{t\text{-}1}}$ and frame $\boldsymbol{F_{t+1}}$ is estimated by minimizing the optical flow energy function

$$E_{\mathrm{OF}} = E_{\mathrm{OF,D}}(\boldsymbol{V_t}) + \alpha E_{OF,S}(\boldsymbol{V_t})$$

where the data term $E_{\mathrm{OF,D}}$ and smooth term $E_{\mathrm{OF,S}}$ are defined as

Fig. 11 An example of MVF estimated by optical flow estimation [23]. The MVF includes MVs of each pixel. The color coding of the MVF is explained in the evaluation method section

$$E_{\mathrm{OF,D}}(\boldsymbol{V_t}) = \int_{\Omega} \varphi\Big(|\boldsymbol{F_{t-1}}(\boldsymbol{x}) - \boldsymbol{F_{t+1}}(\boldsymbol{x})|^2 + \gamma \|\nabla \boldsymbol{F_{t-1}}(\boldsymbol{x}) - \nabla \boldsymbol{F_{t+1}}(\boldsymbol{x})\|_2^2 \Big) d\boldsymbol{x}$$

$$E_{\mathrm{OF,S}}(\boldsymbol{V_t}) = \int_{\Omega} \varphi\Big(\|\nabla u_t\|_2^2 + \|\nabla v_t\|_2^2 \Big) d\boldsymbol{x}$$

Here, ∇ denotes the gradient operator, and the α and γ denote the weighting parameters, respectively. Moreover, $\boldsymbol{V_t} = (u_t, v_t)$ and $\varphi(\mathrm{s}) = \sqrt{s + \varepsilon}$, where ε is a tiny constant. The data term $E_{\mathrm{OF,D}}$ preserves the intensity and gradient constancy of the data, while the smoothness term $E_{\mathrm{OF,S}}$ smooths the optical flow field. Both terms are measured in a regularized L1-norm via function φ. That is, the data term can improve the robustness to outliers, and the smoothness term can preserve motion boundaries. According to [22], the optical flow energy function is minimized by using a multilevel pyramidal scheme to cope with large displacements. However, the optical flows obtained in the previous level are often over-smoothed near motion boundaries. Therefore, the initial optical flow near motion boundaries should be refined at each level before the optimization. There are also many implementations available. Figure 11 shows an example of the pixel-based MVF estimated by [23].

Decoder-side MV extraction is an efficient way to retrieve MVF when the decoder information is available. This is often true when FRUC techniques are employed to the abovementioned low bit rate video coding application. [24] suggests that the decoder-side MVF is also useful for the FRUC techniques. Figure 12 shows an example of MVF extracted from H.264 decoder [25]. The color coding of MVF and the ground truth MVF are explained in the evaluation method section. Three different ME strategies are shown: full search (FS), fast full search (Fast FS), and enhanced predictive zonal search (EPZS). The general FS algorithm with two different block size is also shown for comparison. We can see that the MVFs extracted from H.264 decoder include accurate MVs in most regions. The further MV processing can be operated to achieve a MVF closer to the ground truth one.

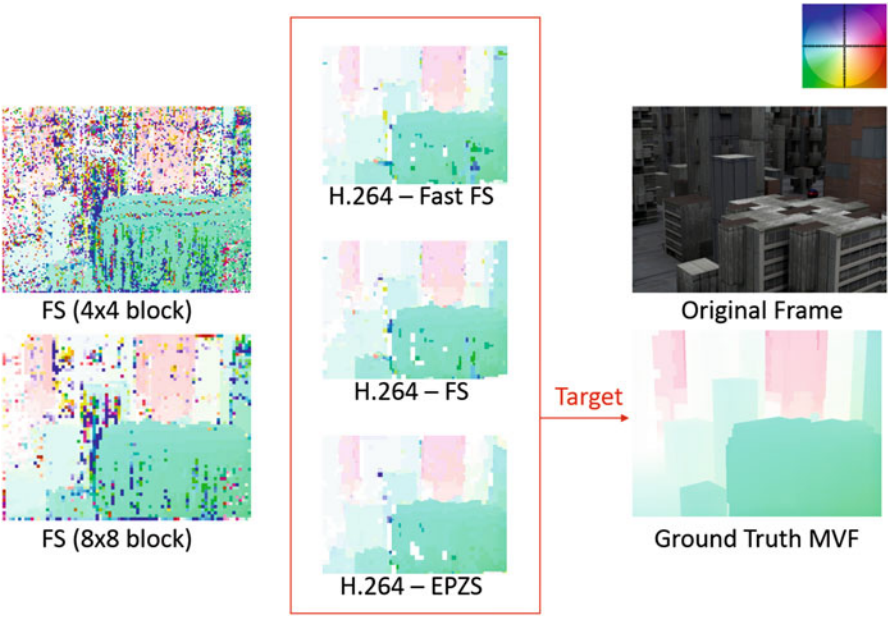


Fig. 12 An example of MVF extracted from H.264 decoder. The left two MVFs are computed using conventional full-search ME with two different block sizes. The middle three MVFs are extracted from H.264 decoder with different ME strategies. The target is to enhance the MVFs to approach the ground truth one

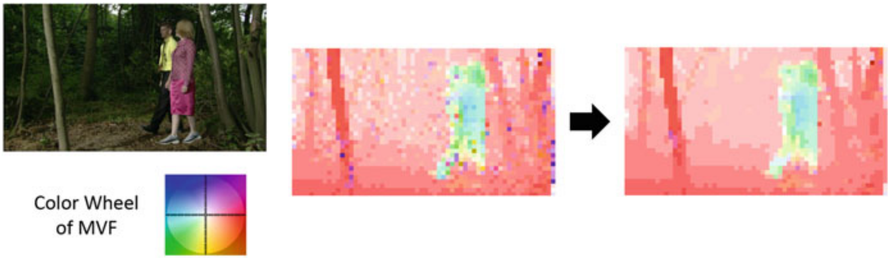


Fig. 13 Illustration of the MVF before and after MV processing. The outlier MVs are removed using MRF modeling

5.2 *Motion Vector Processing*

To restore the MV outliers which are misestimated as shown in Fig. 13, a further MV processing is often performed on the MVF. For example, a vector median filter [26] is often adopted owing to the important spatial and temporal coherence in true MVFs. In addition, more processing methods such as prediction-based MV

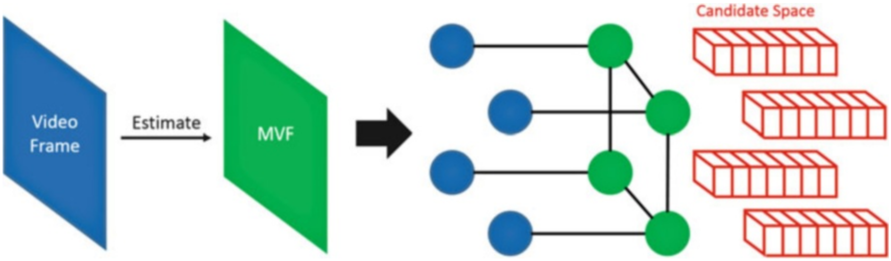


Fig. 14 Illustration of MRF modeling on MVF estimation. Estimating the MVF while observing the video frame. The edges between nodes show the connection modeled in the MRF energy function. The candidate space size depends on the algorithm

smoothing [9], motion smoothing via global energy minimization [27], and 3D Markov random field (MRF) modeling [28] are proposed to approximate the true MVF. A hardware-friendly MRF modeling method is also introduced in [17].

Vector median filter is a widely used technique to remove outlier vectors. Since MVs are vectors containing x-direction and y-direction, the filter can be applied straightforward. The median filter is a simple but effective choice. Using a 3 × 3, 5 × 5, or larger window for a block, its median MV is often highly consistent with its neighboring MVs. The median vector $\boldsymbol{v}_m$ can be derived by

$$\boldsymbol{v}_m = \underset{\boldsymbol{v}_m \in S_i}{\operatorname{argmin}} \sum_{i=1}^{N} \|\boldsymbol{v}_m - \boldsymbol{v}_i\|_L$$

where i is the summation index, N is the number of members in the set, and $S_i = \{\boldsymbol{v}_i\}$ is the set of vectors. L denotes the order of the norm, and any proper norm is eligible to be used.

MRF modeling is a theoretical modeling method based on the Bayesian framework, and this modeling method has been applied to computer-vision algorithms for many years [29]. The framework can also be adopted to estimate MVF in the previous ME part as shown in Fig. 14. A node represents a pixel or block and its corresponding MV. The MVF is estimated with observed video frame. For each node, searching in the MV candidate space and choosing the MV can locally or globally minimize the MRF energy function.

A good reason to apply the framework after the ME is to reduce MV candidates to avoid heavy computation. [30] chooses only neighboring MVs as candidates to perform optimization, which can prevent over-smoothing and keep the computational complexity lower. For a block, eight neighboring MVs and the MV itself are chosen as the nine MV candidates. Thus, the corresponding MRF energy function for each MV candidate is calculated as follows,

$$\text{energy}_{\text{candidate}} = \text{cost}(\text{MV}_{\text{candidate}}) + \text{weight} \times \sum_{\forall\text{neighbor}} \left|\text{MV}_{\text{candidate}} - \text{MV}_{\text{neighbor}}\right|$$

The cost of a MV candidate can be measured in many ways such as the MSEA and optical flow data term previously introduced. The MV candidate that can minimize the MRF energy function is selected as the refined MV of the processed block as follows:

$$\text{Refined MV} = \underset{\text{MV}_{\text{candidate}}}{\text{argmin}}\ \text{energy}_{\text{candidate}}$$

5.3 Motion Vector Mapping

The abovementioned MVF generally represents the motion relation between two existing frames. To convert the frame rate to a higher one, new frames are interpolated between the two existing frames. The frame rate is doubled when an additional frame is interpolated as shown in Fig. 15. In this illustration, the MVF is divided by two to map the position from existing frames to the middle frame.

To our knowledge, three MV mapping methods are mainly available, and these are shown in Fig. 16. The first one is called the *traditional MV mapping* method,

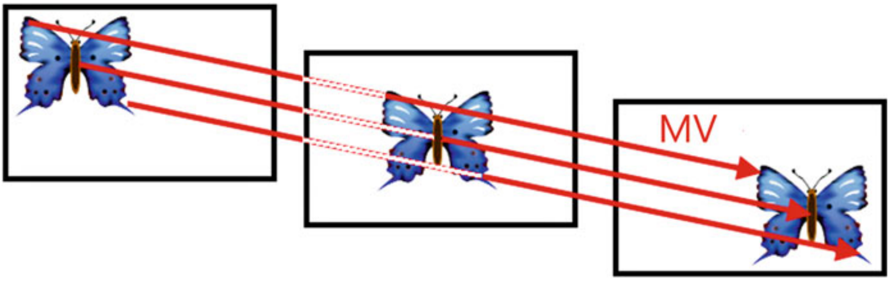


Fig. 15 Illustration of mapping MVs to the interpolated frames. Dividing the MVF by two maps the position to the interpolated frames for the twice up-converted video

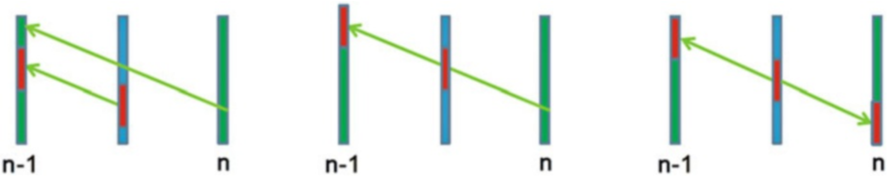


Fig. 16 Three general MV mapping methods. From left to right: traditional, forward, and bilateral MV mapping. Note that green, blue, and red represent existing frames, intermediate frames, and processing pixels/blocks, respectively

Fig. 17 The problems on overlaps and holes. The black regions are not pointed by reference pixels, which generate the holes

which maps a block MV in existing frames to a block at the same position in the intermediate frames. However, these two blocks exist at the same position but at different times; hence, their MVs are not exactly the same. The second one is called the *forward MV mapping* [31], which maps through the direction of an MV to the block where it is pointed. No temporal mismatch occurs in the use of the forward MV mapping, but in this case, some positions may be pointed out by many MVs or no MV at all, thus introducing problems on overlaps and holes. The third one is called the *bilateral MV mapping*, which performs ME on the intermediate frames [32, 33]. No problem on overlaps and holes, as mentioned earlier, but it usually fails to estimate the true MVs in flat regions. Recent researches often improve their MV mapping based on the abovementioned three mapping methods.

Forward MV mapping can avoid temporal mismatch but introduce problems on overlaps and holes as mentioned above. Figure 17 shows an example of the problems. Since the mapping operation is temporally correct and straightforward, it is widely adopted. However, to solve the problems on overlaps and holes, the further MC and post-processing techniques play important roles.

Bilateral MV mapping operation is shown in Fig. 18. The mapping method can avoid the problems on overlaps and holes because it operates on the target intermediate frame. For each block in the intermediate frame, it searches a pair of matched blocks in previous and next frames. The searching directions on two frames are opposite to match temporal domain.

5.4 Motion Compensation

Within the mapped MVF, MC is performed to interpolate the intermediate frame. The appropriate MVs are calculated as shown in Fig. 19. The adopted MC technique often depends on the ME and MV mapping techniques. For forward MV

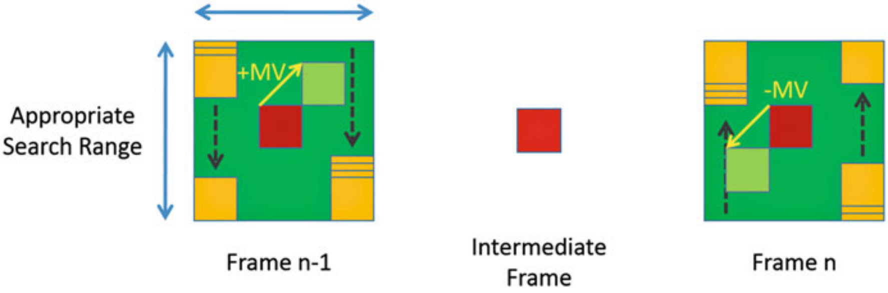


Fig. 18 The operation of bilateral MV mapping. In an appropriate search range of the previous and next frame, the blocks in the intermediate frame search for a best-matched pair of blocks

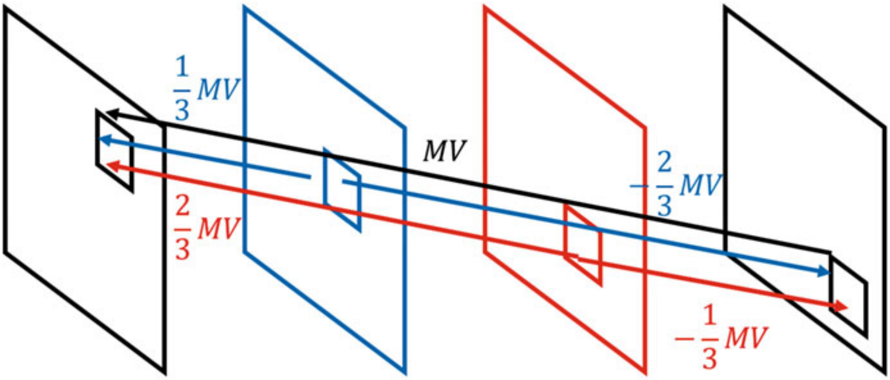


Fig. 19 MC is performed to interpolate the intermediate frame using the appropriate ratio of MVs

mapping, the problems of overlap and hole should be taken into account. For example, [34] uses mesh-based method, [12] employs patch-based reconstruction, and [30] applies block-based forward MC. For block-based operations, block artifacts should be avoided. Overlapped block MC (OBMC) [35] is a conventional method in this part and the further post-processing. Applying adaptive weighted interpolation for occlusion handling is also proposed [36]. In addition, because MC should be performed for each interpolated frame, the computational effort and hardware bandwidth consumption become significant problems when the higher frame rate is required and the number of interpolated frames increases.

Conventional *bilinear* and *adaptive weighting MC* calculate a pixel value in an interpolated frame. The pixel at the position $\boldsymbol{p}$ in frame t is computed as

$$f(\boldsymbol{p}, t) = \frac{\lambda_1 f(\boldsymbol{p} - \boldsymbol{MV}, t-1) + \lambda_2 f(\boldsymbol{p} + \boldsymbol{MV}, t+1)}{2}$$

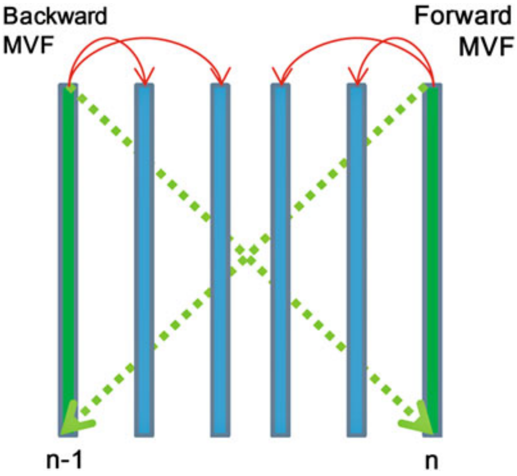


Fig. 20 Both forward and backward MVFs are estimated to achieve multirate up-conversion. The red arrows show that unidirectional interpolation is adopted for these four intermediate frames

That is, blending the pixels after taking the object motion into account. The parameters λ_1 and λ_2 can be adjusted to be adaptive weighting.

To achieve higher frame rates, the MVF should be divided by a larger number, and the MC technique should be performed more than once. However, the visual quality of up-converted frames is limited by the precision of MVF and computational complexity of the following MC. Therefore, twice up-conversion mapping is most adopted.

Estimating more MVFs creates possibilities to achieve higher frame rate. To achieve five times up-conversion, [30] performs low-complexity ME twice to retrieve the forward and backward MVFs, as indicated by the green dotted arrows in Fig. 20. Moreover, unidirectional interpolation is adopted to prevent blur and reduce complexity. In other words, the first and second intermediate frames are interpolated using the pixels in frame $n - 1$ with mapped backward MVF. Similarly, the third and fourth intermediate frames are interpolated using the pixels in frame n with mapped forward MVF.

5.5 *Post-processing*

The visual quality of interpolated frames always suffers due to unstable results from all of the above factors such as incorrect MVs and interpolation mismatch. Therefore, many studies choose to refine the interpolated frames via post-processing [36, 37]. Nevertheless, determining where the artifacts are and how to interpolate the blocks to obtain better visual quality are difficult.

The artifacts often happen near moving object boundaries because there exists occluded and uncovered regions. These regions cause misestimated MVs due to unmatched pixel values between two frames. [12] utilizes structure similarity and

Fig. 21 An example of possible regions with visual artifacts detected by [30]. These regions mostly reside near the boundaries of moving objects

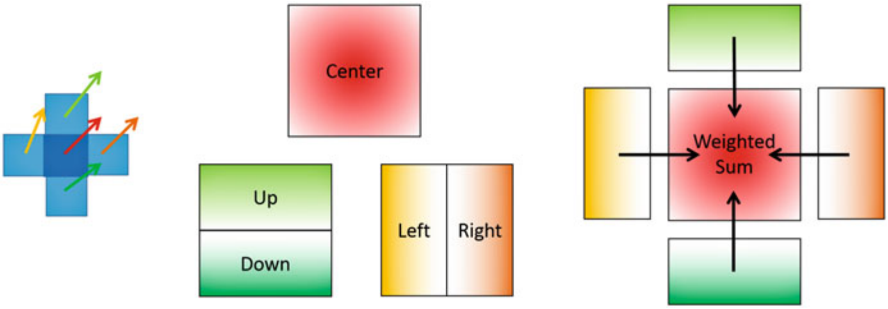


Fig. 22 Illustration of the OBMC operation. Five different colors represent five blocks. Each neighboring block contributes its half part near the processed block for blending. The more saturated colors represent the larger weighting numbers

edge information to generate a mismatch mask for further visual enhancement. [30] divides blocks into sub-blocks and takes the boundaries of the MV discontinuity as the possible regions with visual artifacts. Figure 21 shows an example of the detected result. [38] computes residual energy for blocks and then enhance their visual quality instead of finding occlusion regions.

A local refinement is often performed to enhance the visual quality of these regions. OBMC [35] and its variants are often adopted. The concept is to blend the regions with their neighboring regions by weighted sum. As shown in Fig. 22, four neighboring blocks are selected to perform OBMC on one block. We use colors to

Fig. 23 Results of post-processing to eliminate artifacts near boundaries of moving objects. The left and right images of each pair are before and after processing, respectively

represent the blocks and their weighted number, and the larger weighted numbers show more saturated color. The pixel values in these five blocks are blended by weighted sum. Performing weighted sum can achieve smooth visual quality between blocks because the pixel values across block boundaries are blended. Figure 23 shows several results before and after post-processing.

6 Evaluation Methods

To evaluate the performance of each technique, a universal dataset and measurement play important roles. In this section, we introduce the popular video datasets and evaluation criterion.

6.1 Test Video Sequences

Video coding techniques have been studied for many years. Therefore, several test video sequences are widely used in the research field [39, 40]. Figure 24 shows some snapshots of the well-known test video sequences. FRUC techniques are usually performed on these video sequences for evaluation. Since recording a video is a simple task nowadays, many researchers also have their own video sequences.

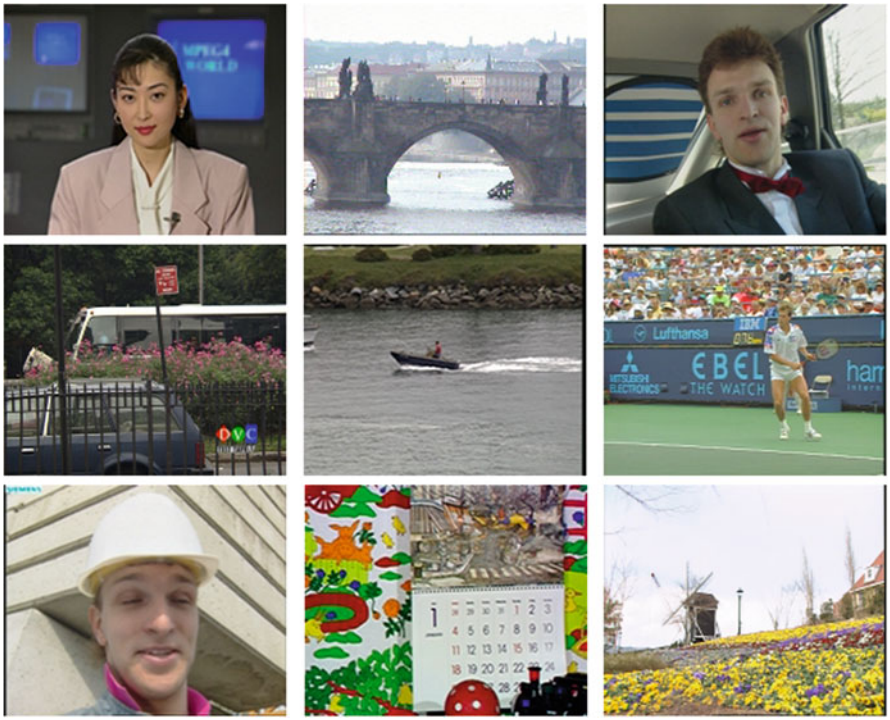


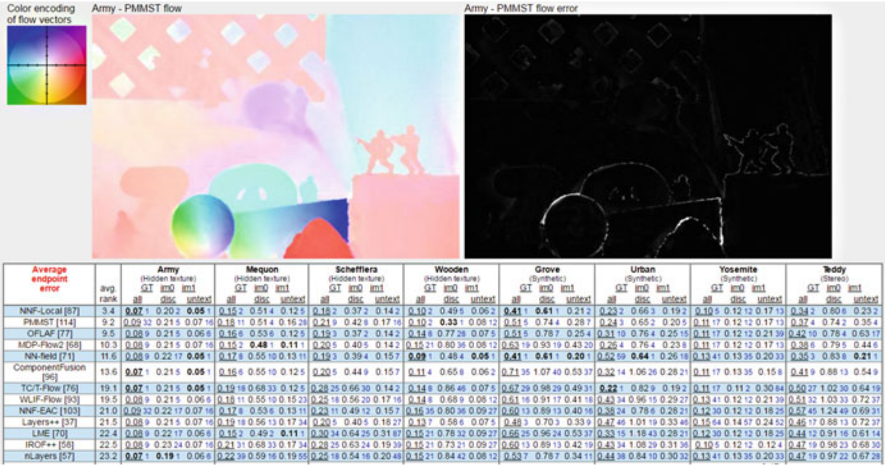
Fig. 24 An example of popular test video sequences in video coding

6.2 Motion Visualization and Evaluation

Since the estimated MVFs are crucial for FRUC, visualizing and evaluating MVFs are helpful. [41] provides a website [42] containing tools for visualizing MVF and several image pairs with ground truth MVF for evaluation. A snapshot of [42] is shown in Fig. 25. The MVFs are visualized using color coding. More specifically, a color wheel is utilized, and its center representing zero motion is white color. The ticks on the x-axe and y-axe denote a motion unit of one pixel. Therefore, the representation can visualize pixel-based MVF in floating-point precision. Note that the maximum magnitudes of the visualized MVF depend on different datasets and can be manually set.

6.3 Evaluation Criterion

The most popular evaluation method is to compute the difference between original and interpolated frames. The evaluation method using frame skipping is illustrated in Fig. 26. For example, the even frames in original video sequence are deleted, and



Average endpoint error	avg. rank	Army (Hidden texture) GT im0 im1			Mequon (Hidden texture) GT im0 im1			Schefflera (Hidden texture) GT im0 im1			Wooden (Hidden texture) GT im0 im1		
		all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext
NNF-Local [87]	3.4	**0.07** 1	0.20 2	**0.05** 1	0.15 2	0.51 4	0.12 5	0.18 2	0.37 2	0.14 2	0.10 2	0.49 5	0.06 2
PMMST [114]	9.2	0.09 32	0.21 5	0.07 16	0.18 11	0.51 4	0.16 28	0.21 9	0.42 8	0.17 16	0.10 2	**0.33** 1	0.08 12
OFLAF [77]	9.5	0.08 9	0.21 5	0.06 6	0.16 6	0.53 6	0.12 5	0.19 3	0.37 2	0.14 2	0.14 8	0.77 28	0.07 5
MDP-Flow2 [68]	10.3	0.08 9	0.21 5	0.07 16	0.15 2	**0.48** 1	**0.11** 1	0.20 5	0.40 5	0.14 2	0.15 21	0.80 36	0.08 12
NN-field [71]	11.6	0.08 9	0.22 17	**0.05** 1	0.17 8	0.55 10	0.13 11	0.19 3	0.39 4	0.15 7	**0.09** 1	0.48 4	**0.05** 1
ComponentFusion [96]	13.6	**0.07** 1	0.21 5	**0.05** 1	0.16 6	0.55 10	0.12 5	0.20 5	0.44 9	0.15 7	0.11 4	0.65 8	0.06 2
TC/T-Flow [76]	19.1	**0.07** 1	0.21 5	**0.05** 1	0.19 18	0.68 33	0.12 5	0.28 25	0.66 30	0.14 2	0.14 8	0.86 46	0.07 5
WLIF-Flow [93]	19.5	0.08 9	0.21 5	0.06 6	0.18 11	0.55 10	0.15 23	0.25 18	0.56 20	0.17 16	0.14 8	0.68 9	0.08 12
NNF-EAC [103]	21.0	0.09 32	0.22 17	0.07 16	0.17 8	0.53 6	0.13 11	0.23 11	0.49 12	0.15 7	0.16 35	0.80 36	0.09 27
Layers++ [37]	21.5	0.08 9	0.21 5	0.07 16	0.19 18	0.56 13	0.17 34	0.20 5	0.40 5	0.18 27	0.13 7	0.58 6	0.07 5
LME [70]	22.4	0.08 9	0.22 17	0.06 6	0.15 2	0.49 2	**0.11** 1	0.30 34	0.64 25	0.31 87	0.15 21	0.78 32	0.09 27
IROF++ [58]	22.5	0.08 9	0.23 24	0.07 16	0.21 31	0.68 33	0.17 34	0.28 25	0.63 24	0.19 39	0.15 21	0.73 21	0.09 27
nLayers [57]	23.2	**0.07** 1	**0.19** 1	0.06 6	0.22 39	0.59 16	0.19 55	0.25 18	0.54 16	0.20 48	0.15 21	0.84 42	0.08 12

Average endpoint error	avg. rank	Grove (Synthetic) GT im0 im1			Urban (Synthetic) GT im0 im1			Yosemite (Synthetic) GT im0 im1			Teddy (Stereo) GT im0 im1		
		all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext
NNF-Local [87]	3.4	**0.41** 1	**0.61** 1	0.21 2	0.23 2	0.66 3	0.19 2	0.10 5	0.12 12	0.17 13	0.34 2	0.80 6	0.23 2
PMMST [114]	9.2	0.51 5	0.74 4	0.28 7	0.24 3	0.65 2	0.20 5	0.11 17	0.12 12	0.17 13	0.37 4	0.74 2	0.35 4
OFLAF [77]	9.5	0.51 5	0.78 7	0.25 4	0.31 10	0.76 4	0.25 15	0.11 17	0.12 12	0.21 39	0.42 10	0.78 4	0.63 17
MDP-Flow2 [68]	10.3	0.63 19	0.93 19	0.43 20	0.26 4	0.76 4	0.23 8	0.11 17	0.12 12	0.17 13	0.38 6	0.79 5	0.44 6
NN-field [71]	11.6	**0.41** 1	**0.61** 1	**0.20** 1	0.52 59	**0.64** 1	0.26 18	0.13 41	0.13 35	0.20 33	0.35 3	0.83 8	**0.21** 1
ComponentFusion [96]	13.6	0.71 35	1.07 40	0.53 37	0.32 14	1.06 26	0.28 21	0.11 17	0.13 35	0.15 8	0.41 9	0.88 13	0.54 9
TC/T-Flow [76]	19.1	0.67 29	0.98 29	0.49 31	**0.22** 1	0.82 9	0.19 2	0.11 17	0.11 2	0.30 84	0.50 27	1.02 30	0.64 19
WLIF-Flow [93]	19.5	0.61 16	0.91 17	0.41 18	0.43 34	0.96 15	0.29 27	0.13 41	0.12 12	0.21 39	0.51 32	1.03 33	0.72 37
NNF-EAC [103]	21.0	0.60 13	0.89 13	0.40 16	0.38 24	0.78 6	0.28 21	0.12 30	0.12 12	0.18 25	0.57 45	1.24 49	0.69 31
Layers++ [37]	21.5	0.48 3	0.70 3	0.33 9	0.47 46	1.01 19	0.33 46	0.15 64	0.14 57	0.24 52	0.46 17	0.88 13	0.72 37
LME [70]	22.4	0.66 25	0.96 24	0.53 37	0.33 15	1.18 43	0.28 21	0.12 30	0.12 12	0.18 25	0.44 12	0.91 16	0.61 14
IROF++ [58]	22.5	0.60 13	0.89 13	0.42 19	0.43 34	1.08 29	0.31 36	0.10 5	0.12 12	0.12 4	0.47 19	0.98 23	0.68 30
nLayers [57]	23.2	0.53 7	0.78 7	0.34 11	0.44 38	0.84 10	0.30 32	0.13 41	0.13 35	0.20 33	0.47 19	0.97 22	0.67 28

Fig. 25 An example of visualizing and evaluating MVFs. The vectors are visualized according to their positions in the color wheel. The results by selected algorithm are shown, and the ranked algorithms are listed below

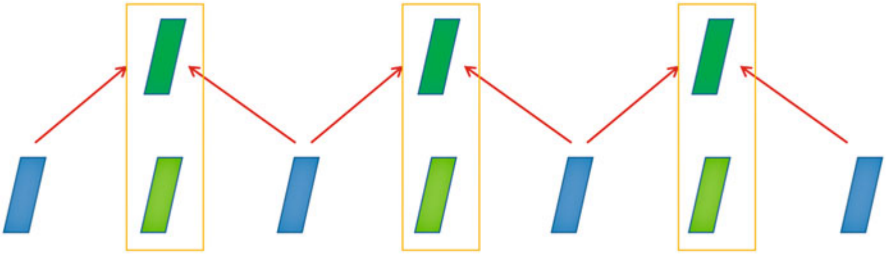


Fig. 26 Illustration of skipping frames for FRUC evaluation. The difference between skipped and interpolated frames is computed for evaluation

FRUC techniques are then performed to interpolate new even frames. The FRUC technique is evaluated based on the difference between the original and interpolated even frames.

Heinrich et al. [43] suggests an evaluation method called double interpolation without altering the original frame rate of the test video sequence. The first interpolation takes place between original frames and the second one on the interpolated result as illustrated in Fig. 27. Since the interpolation errors caused by the evaluated FRUC technique are possible to be amplified, it allows a better performance to discriminate between different FRUC methods.

The difference between frames is usually measured by the peak signal-to-noise ratio (PSNR). The PSNR between two images I^A and I^B is computed as

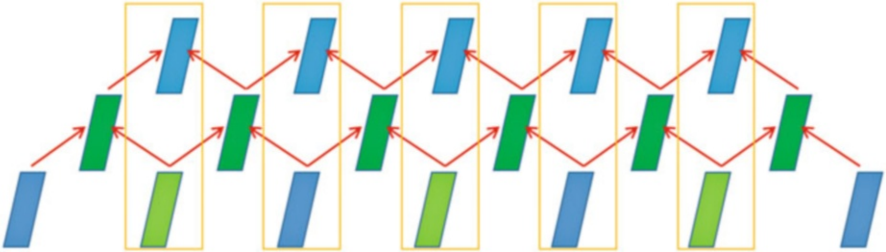


Fig. 27 Illustration of double interpolating frames for FRUC evaluation. The frames are interpolated twice and compared to the original frames for evaluation

$$\mathrm{PSNR}\left(I^{\mathrm{A}}, I^{\mathrm{B}}\right) = 10\log_{10}\left(\frac{255^2}{\mathrm{MSE}}\right)$$

$$\mathrm{MSE} = \frac{\sum_{n=1}^{\text{Frame Size}} \left(I_n^{\mathrm{A}} - I_n^{\mathrm{B}}\right)^2}{\text{Frame Size}}$$

Although the PSNR values cannot totally represent the perceptual visual quality, frames with higher PSNR values still appear better at most times. In addition to the objective evaluation, some researchers also perform subjective evaluation to measure the perceptual visual quality. Displaying the original and processed videos, several people rate the videos. Instead, some researchers perform the structural similarity index (SSIM) [44] to evaluate the results since the index shows more relation with perceptual visual quality. The SSIM between two blocks B_1 and B_2 is defined as

$$\mathrm{SSIM}(B_1, B_2) = \frac{2(m_1 m_2 + C_1)(2\sigma_{1,2} + C_2)}{\left(m_1^2 + m_2^2 + C_1\right)\left(\sigma_1^2 + \sigma_2^2 + C_2\right)}$$

where m_i and σ_i^2 are the mean and variance of the luminance value in the corresponding block B_i, respectively. $\sigma_{1,2}$ is the covariance between values in two blocks B_1 and B_2. C_1 and C_2 are constants which stabilize the division.

7 Hardware Implementation Issues

Since FRUC becomes a crucial technique in display devices, integrating it into display systems is an efficient solution for consumer electronics. However, the required amounts of computational cost and bandwidth consumption are massive. The higher video resolution also introduces new challenges. Three challenges are encountered for the hardware architecture design. The first challenge is the requirement of a large on-chip SRAM. The on-chip SRAM arrangement is significant to

Table 1 Comparison of the FRUC hardware architecture specifications

	Kang [45]	Cetin [46]	Wang [48]	Hsu [49]	Lee [47]	Huang [30]
Technology	Xilinx Virtex 4	FPGA 90-nm	UMC 90-nm	UMC 90-nm	Altera Cyclone III	TSMC 90-nm
Clock rate (MHz)	168.33	63	200	133	148.5	300
Total gate count	6899 slices	89,000 slices	292,732	1,627,900	28,091 LEs	410,356
SRAM size (Bytes)	N/A	14,070	3036	12,365	336,444	9984
FRUC mode (FPS)	60–120	168–336	60–120	60–120	60–120	24–120 60–120
Frame size	352 × 288	1280 × 720	1920 × 1080	1920 × 1080	1920 × 1080	3840 × 2160

support a larger resolution operation. Second, because interpolating multiple frames is essential to achieve multirate FRUC, the required bandwidth consumption for the ME and MC becomes larger. Third, the cycles for data fetching must be as few as possible to achieve multirate up-conversions. Consequently, efficiently utilizing the available hardware resources is very important.

We list six hardware implementations of FRUC techniques for reference in Table 1. Three [45–47] are implemented using a field-programmable gate array (FPGA), and three [30, 48, 49] are implemented using an application-specific integrated circuit (ASIC). Despite the fact that comparison between hardware implementations is difficult, it shows that the requirement of hardware resource becomes higher. [48, 30] lower the requirement but maintain the competitive performance because they apply more efficient hardware utilization. To support the larger frame size and multirate video mode for current display devices, recent researchers should address more hardware implementation issues.

8 Conclusion and Discussion

In this chapter, we present several applications and techniques of multirate video systems. This shows that multirate property for video systems is essential, and the related research topics are important issues. We start from the fundamental FRC techniques, and selected conventional methods are presented. Then, we divide the FRUC techniques into five parts and state several popular methods of each part. The evaluation methods and hardware implementation issues are also discussed. We give a thorough presentation on the whole stage of the FRC techniques. However, many video processing techniques such as scene change detection, videotext localization, and perceptual video processing can also be applied to enhance the visual quality of converted videos. Moreover, the FRC techniques can be cooperated with other techniques such as video coding, super resolution, and virtual view synthesis. We hope the relative techniques and implementations become well developed and enhance the video viewing experience for people.

References

1. Wiegand, T., Sullivan, G. J., Bjontegaard, G., & Luthra, A. (2003). Overview of the H. 264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology, 13*(7), 560–576.
2. Sullivan, G. J., Ohm, J., Han, W. J., & Wiegand, T. (2012). Overview of the high efficiency video coding (HEVC) standard. *IEEE Transactions on Circuits and Systems for Video Technology, 22*(12), 1649–1668.
3. Poynton, C. (2012). *Digital video and HD: Algorithms and interfaces*. Amsterdam: Elsevier.
4. Pan, H., Feng, X., & Daly, S. (2005). 51.4: Quantitative analysis of LCD motion blur and performance of existing approaches. In *SID symposium digest of technical papers* (Vol. 36, No. 1, pp. 1590–1593). Blackwell Publishing Ltd, UK.
5. Pan, H., Feng, X. F., & Daly, S. (2005). LCD motion blur modeling and analysis . In *Proceedings of IEEE international conference of image processing* (*ICIP*). (Vol. 2, pp. II–21).
6. De Haan, G., Biezen, P. W., Huijgen, H., & Ojo, O. A. (1993). True-motion estimation with 3-D recursive search block matching. *IEEE Transactions on Circuits and Systems for Video Technology, 3*(5), 368–379.
7. Wang, J., Wang, D., & Zhang, W. (2003). Temporal compensated motion estimation with simple block-based prediction. *IEEE Transactions on Broadcasting, 49*(3), 241–248.
8. Tourapis, A. M. (2002, January). Enhanced predictive zonal search for single and multiple frame motion estimation. In *Proceedings of SPIE visual communications and image processing (VCIP)* (pp. 1069–1079).
9. Kim, U. S., & Sunwoo, M. H. (2014). New frame rate up-conversion algorithms with low computational complexity. *IEEE Transactions on Circuits and Systems for Video Technology, 24*(3), 384–393.
10. Min, K. Y., & Sim, D. G. (2013). Confidence-based adaptive frame rate up-conversion. *EURASIP Journal on Advances in Signal Processing, 2013*(1), 1–12.
11. Liu, H., Xiong, R., Zhao, D., Ma, S., & Gao, W. (2012). Multiple hypotheses Bayesian frame rate up-conversion by adaptive fusion of motion-compensated interpolations. *IEEE Transactions on Circuits and Systems for Video Technology, 22*(8), 1188–1198.
12. Kaviani, H., & Shirani, S. (2016). Frame rate up-conversion using optical flow and patch-based reconstruction. *IEEE Transactions on Circuits and Systems for Video Technology, 26*(9), 1581–1594.
13. Yang, Y. T., Tung, Y. S., & Wu, J. L. (2007). Quality enhancement of frame rate up-converted video by adaptive frame skip and reliable motion extraction. *IEEE Transactions on Circuits and Systems for Video Technology, 17*(12), 1700–1713.
14. Huang, A. M., & Nguyen, T. Q. (2008). A multistage motion vector processing method for motion-compensated frame interpolation. *IEEE Transactions on Image Processing, 17*(5), 694–708.
15. Liu, Y. N., Wang, Y. T., & Chien, S. Y. (2011). Motion blur reduction of liquid crystal displays using perception-aware motion compensated frame rate up-conversion. In *Proceedings of IEEE workshop on signal processing systems (SiPS)* (pp. 84–89).
16. Gao, X. Q., Duanmu, C. J., & Zou, C. R. (2000). A multilevel successive elimination algorithm for block matching motion estimation. *IEEE Transactions on Image Processing, 9*(3), 501–504.

17. Chen, F. C., Huang, Y. L., & Chien, S. Y. (2012). Hardware-efficient true motion estimator based on Markov Random Field motion vector correction. In *Proceedings of IEEE international symposium on VLSI design, automation, and test (VLSI-DAT)* (pp. 1–4).
18. Li, R., Zeng, B., & Liou, M. L. (1994). A new three-step search algorithm for block motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology, 4*(4), 438–442.
19. Po, L. M., & Ma, W. C. (1996). A novel four-step search algorithm for fast block motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology, 6*(3), 313–317.
20. Lucas, B. D., & Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2 (IJCAI'81)* (Vol. 2, pp. 674–679). San Francisco, CA: Morgan Kaufmann Publishers Inc.
21. Tang, C. W., & Au, O. C. (1998, May). Comparison between block-based and pixel-based temporal interpolation for video coding. In *Proceedings of IEEE international symposium on circuits and systems (ISCAS)*, (Vol. 4, pp. 122–125).
22. Lee, W. H., Choi, K., & Ra, J. B. (2014). Frame rate up conversion based on variational image fusion. *IEEE Transactions on Image Processing, 23*(1), 399–412.
23. Liu, C. (2009). *Beyond pixels: exploring new representations and applications for motion analysis*. Doctoral dissertation, Massachusetts Institute of Technology.
24. Huang, Y. L., Liu, Y. N., & Chien, S. Y. (2010, October). MRF-based true motion estimation using H. 264 decoding information. In *IEEE workshop on signal processing systems (SIPS)* (pp. 99–104).
25. H.264/AVC Software Coordination. http://iphome.hhi.de/suehring/tml/
26. Astola, J., Haavisto, P., & Neuvo, Y. (1990). Vector median filters. *Proceedings of the IEEE, 78*(4), 678–689.
27. Dane, G., & Nguyen, T. Q. (2004). Smooth motion vector resampling for standard compatible video post-processing. In *Proceedings of IEEE Asilomar conference on signals, systems and computers* (Vol. 2, pp. 1731–1735).
28. Wang, D., Zhang, L., & Vincent, A. (2010). Motion-compensated frame rate up-conversion—Part I: Fast multi-frame motion estimation. *IEEE Transactions on Broadcasting, 56*(2), 133–141.
29. Li, S. Z. (1994, May). Markov random field models in computer vision. In *European conference on computer vision* (pp. 361–370). Berlin/Heidelberg: Springer.
30. Huang, Y. L., Chen, F. C., & Chien, S. Y. (2016). Algorithm and architecture design of multirate frame rate up-conversion for ultra-HD LCD systems. *IEEE Transactions on Circuits and Systems for Video Technology, PP*(99), 1–1. doi:10.1109/TCSVT.2016.2596198.
31. Jeon, B. W., Lee, G. I., Lee, S. H., & Park, R. H. (2003). Coarse-to-fine frame interpolation for frame rate up-conversion using pyramid structure. *IEEE Transactions on Consumer Electronics, 49*(3), 499–508.
32. Choi, B. T., Lee, S. H., & Ko, S. J. (2000). New frame rate up-conversion using bi-directional motion estimation. *IEEE Transactions on Consumer Electronics, 46*(3), 603–609.
33. Choi, B. D., Han, J. W., Kim, C. S., & Ko, S. J. (2007). Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation. *IEEE Transactions on Circuits and Systems for Video Technology, 17*(4), 407–416.
34. Min, K. Y., Ma, J. H., Sim, D. G., & Bajic, I. V. (2015). Bidirectional mesh-based frame rate up-conversion. *IEEE Multimedia, 22*(2), 36–45.
35. Orchard, M. T., & Sullivan, G. J. (1994). Overlapped block motion compensation: An estimation-theoretic approach. *IEEE Transactions on Image Processing, 3*(5), 693–699.
36. Ling, Y., Wang, J., Liu, Y., & Zhang, W. (2008). A novel spatial and temporal correlation integrated based motion-compensated interpolation for frame rate up-conversion. *IEEE Transactions on Consumer Electronics, 54*(2), 863–869.
37. Hsu, K. Y., & Chien, S. Y. (2008). Frame rate up-conversion with global-to-local iterative motion compensated interpolation. In *Proceedings of IEEE international conference on multimedia and expo (ICME)*, (pp. 161–164).

38. Huang, A. M., & Nguyen, T. (2009). Correlation-based motion vector processing with adaptive interpolation scheme for motion-compensated frame interpolation. *IEEE Transactions on Image Processing, 18*(4), 740–752.
39. Xiph.org Video Test Media. https://media.xiph.org/video/derf/
40. YUV Video Sequences. http://trace.eas.asu.edu/yuv/
41. Baker, S., Scharstein, D., Lewis, J. P., Roth, S., Black, M. J., & Szeliski, R. (2011). A database and evaluation methodology for optical flow. *International Journal of Computer Vision, 92*(1), 1–31.
42. Middlebury Optical Flow Dataset. http://vision.middlebury.edu/flow/
43. Heinrich, A., de Haan, G., & Cordes, C. N. (2008). A novel performance measure for picture rate conversion methods. In *Proceedings of IEEE international conference on consumer electronics (ICCE)*, (pp. 1–2).
44. Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing, 13*(4), 600–612.
45. Kang, S. J., Yoo, D. G., Lee, S. K., & Kim, Y. H. (2008, November). Hardware implementation of motion estimation using a sub-sampled block for frame rate up-conversion. In *IEEE international SoC design conference* (Vol. 2, pp. II-101).
46. Cetin, M., & Hamzaoglu, I. (2011). An adaptive true motion estimation algorithm for frame rate conversion of high definition video and its hardware implementations. *IEEE Transactions on Consumer Electronics, 57*(2), 923–931.
47. Lee, G. G., Chen, C. F., Hsiao, C. J., & Wu, J. C. (2014). Bi-directional trajectory tracking with variable block-size motion estimation for frame rate up-convertor. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS), 4*(1), 29–42.
48. Wang, Y.T. (2010). *Algorithm and hardware architecture design of perception-aware motion compensated frame rate up-conversion*. Master's thesis, National Taiwan University.
49. Hsu, K. Y., & Chien, S. Y. (2011). Hardware architecture design of frame rate up-conversion for high definition videos with global motion estimation and compensation. In *Proceedings of IEEE workshop on signal processing systems (SiPS)*, (pp. 90–95).

Multirate Systems in Cognitive Radio

S. Chris Prema and K.S. Dasgupta

1 Introduction

Spectrum sensing is an important function of cognitive radios (CRs), in finding spectrum access opportunities and obtaining noninterfered spectrum for reliable communication. The purpose of cognitive radio techniques is to allow secondary users (unlicensed) to utilize the spectrum which is not occupied by the primary users (licensed) [1]. Multirate systems can perform key signal processing applications for CR systems. In wideband spectrum sensing, the wideband channel is divided into multiple nonoverlapping narrowband channels and is sensed for opportunities which are referred as multiband sensing in literature [2, 3]. The multirate signal processing techniques are useful in wideband spectrum sensing for multiband spectrum detection by the use of filter bank techniques. In cases where wideband spectrum sensing requires high sampling rates and high power consumption, multirate filter banks become a better solution [4].

In general for wideband spectrum sensing, the radio frequency (RF) front end requires wideband architecture, and the spectrum is estimated by using power spectral density (PSD). The basic method used for PSD is the periodogram spectrum estimator (PSE). The PSE is limited due to the trade-off between spectrum resolution and dynamic range because of the sidelobes of the PSE window. To overcome these limitations, multi-taper (MT) method and filter bank-based methods were utilized. The advantage of filter bank method is that it enables efficient implementation of band-pass filters using polyphase decomposition of prototype filters. A comparison between filter bank method and MT has shown that the filter bank methods are more promising compared with MT method in terms of lower computational complexities.

S. Chris Prema (✉) • K.S. Dasgupta
Indian Institute of Space Science and Technology, Trivandrum, India
e-mail: chrisprema@iist.ac.in; ksd@iist.ac.in

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9_7

In wideband multichannel spectrum sensing, FFT or filter bank-based spectrum analyzer has been considered by averaging the output samples of each subbands for detecting multiple spectral gaps. FFT and filter bank techniques have been used for sensing Wireless Local Area Networks (WLAN) such as OFDM based on IEEE 802.11 system and Wireless Personal Area Network (WPAN) with bluetooth designated to operate on 2.4 GHz ISM band [5]. Multi-resolution filter banks based on fast filter bank design with varying spectral bands have been used for sensing military radio receivers [6]. Tree-structured DFT filter bank was applied for estimating the center frequencies and spectral edges of primary user signals [7]. - Filter banks are also useful for the detection of wireless microphones in IEEE 802.22 Wireless Regional Area Network (WRAN) and estimation of center frequency to fractionally utilize the available bandwidth [8, 9]. Progressive decimation filter bank techniques (PDFB) are applicable for variable sensing resolutions to detect different bandwidths in wideband spectrum. The theoretical framework for the analysis and design of filter bank-based detectors for spectrum sensing applications in cognitive radios is also discussed in literature [10]. In this chapter we discuss the spectrum sensing techniques in CR and the application of multirate filter banks in CR applications.

2 Cognitive Radio

Cognitive radio system has been proposed as a promising solution to improve the spectrum utilization. The concept of cognitive radio was proposed by Joseph Mitola [11]. CR systems have intelligent mechanism for monitoring the radio spectrum to detect spectral holes and, thereby, allocate the same to secondary users without causing any harmful interference to the primary users in wideband spectrum. Particularly, CR is considered for obtaining spectrum usage characteristics across multiple dimensions such as time, space, frequency, and code. CR comprises of determining the type of signals in addition to parameters such as modulation, waveform, bandwidth, and carrier frequency occupying the spectrum [12].

FCC defines CR as *A radio or system that senses its operational electromagnetic environment and can dynamically and autonomously adjust its radio operating parameters to modify system operation, such as maximize throughput, mitigate interference, facilitate inter-operability, access secondary markets.*

CRs are considered to be the most promising future wireless communication technology that may potentially mitigate the problem of spectrum scarcity using dynamic spectrum access techniques [13]. The underutilization of spectrum is due to the extremely low spectrum utilization in some localized temporal and geographical spectrum bands. Spectral opportunities have to be detected without any assistance from primary users. The primary users do not have any constraints to share or change the operating parameters for sharing spectrum with cognitive radio networks [14].

A CR system consists of the following entities:

Primary User: The users who have higher priority or legacy rights on the usage of a specific part of the spectrum are defined as primary users.

Secondary User: The unlicensed user, who transmits and receives signals over the licensed spectra or portions of it when primary users are inactive, is called secondary users [12].

Spectral hole: A band of frequencies assigned to a primary user, which is unused at a particular time and at a specific geographic location is called a spectrum hole [1].

The spectral holes are classified into two types: (1) temporal and (2) spatial spectral holes. A temporal spectral hole appears when a primary user is not transmitting for a certain period of time. When the primary user transmission is confined within an area, a spatial spectral hole appears, and the secondary users can use the spectrum outside that area [15].

Cognitive radios have two main features which distinguish them from the conventional radio devices; they are *cognitive capability* and *reconfigurability* [16].

The cognitive ability allows a CR system to sense and capture the information from the surrounding radio environment. This feature allows a cognitive user to be aware of different parameters such as transmitted waveform, radio frequency spectrum, and geographical information. The gathered information are analyzed to identify any unused spectrum at a specific time and location [17]. The interaction between CR and radio environment is known as cognitive cycle. The cognitive ability of CR explained through cognitive cycle is shown in Fig. 1. A cognitive cycle consists of the following three components namely: spectrum sensing, spectrum analysis, and spectrum decision [1].

Spectrum Sensing In spectrum sensing, a cognitive radio observes the frequency band and gathers necessary information regarding its surrounding radio environment. Based on the information captured, the cognitive radio is able to detect spectrum holes.

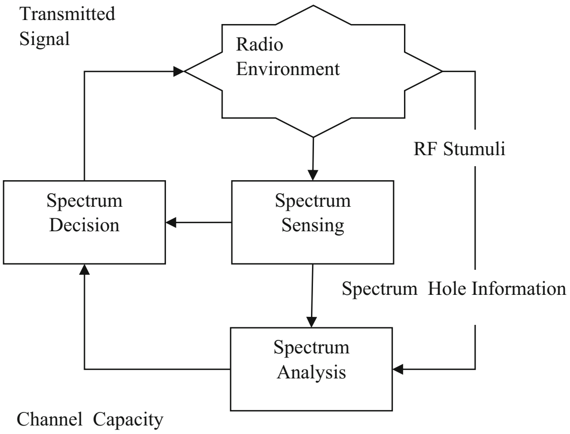


Fig. 1 Cognitive cycle

Spectrum Analysis. Once the spectrum holes are detected using spectrum sensing, each of the spectrum band is characterized based on the local observation of the cognitive radio as well as the statistical information of primary user network. Moreover, characteristics of spectrum holes are also analyzed and estimated.

Spectrum Decision. Depending on the spectrum analysis, the cognitive radio determines the operating parameters such as the data rate, the transmission mode, and the bandwidth available for transmission. The most appropriate spectrum band is selected based on the spectrum band characterization and the user requirements.

As mentioned earlier, the second key feature of a cognitive radio that distinguishes it from a traditional radio is reconfigurability. The ability of a cognitive radio to intelligently adapt to the radio environment by adjusting its operating parameters, according to the sensed environmental variations, in order to achieve the optimal performance, is referred to as reconfigurability. Cognitive wireless networks are capable of reconfiguring their infrastructure in order to adapt to the continuously changing environment. The reconfiguration actions take place in the PHY/MAC layers for the selection of appropriate technology and spectrum for operation. Different transmission access technologies can be supported by its hardware design such that transmission and reception are possible in a variety of frequencies [16, 18].

3 Spectrum Sensing Methods

Spectrum sensing is an inevitable part of cognitive radio systems that allows us to use the available spectrum efficiently. The different spectrum sensing methods provide the key to monitor and reuse the spectrum without interference. One of the major tasks of CR is to obtain underutilized and noninterfered spectrum for allocation of secondary users. The channel conditions keep changing due to the noise uncertainty, multipath fading, and shadowing effects in wireless channels. Therefore, there exists a need for monitoring and cooperation among secondary users for efficient spectrum utilization. The usefulness of the spectrum sensing techniques is based on the sensing performance and complexity in implementation.

Spectrum sensing techniques can be classified as noncooperative and cooperative methods. The cognitive radio acts on its own in noncooperative spectrum sensing, while in cooperative spectrum sensing, multiple CRs work together, which results in an increase of accuracy in spectrum detection and spectrum awareness. Cooperative spectrum sensing is further classified into three categories depending on how cooperating CR users share the sensing data: (1) centralized, (2) distributed, and (3) relay assisted [19, 20]. In multipath fading and shadowing environment, cooperative spectrum sensing is considered to be an effective approach. The common spectrum sensing techniques are energy detection (ED), matched filter (MF), and cyclostationary feature detection (CFD), which are discussed in subsequent sections. In filter banks, subband-based energy detection is applied for detection of spectral holes. Apart from common spectrum sensing methods, other techniques existing in literature include eigenvalue-based methods, covariance matrix method, and wavelet-based methods [21, 22]. After spectrum sensing, the secondary users are allowed to access the spectrum holes. In order to access the spectrum holes effectively, spectrum sharing and spectrum allocation techniques are important [23]. The common spectrum sensing methods are briefly explained in the following sections.

3.1 Energy Detection Method

Energy detection method is further classified as traditional energy detection and subband-based energy detection.

3.1.1 Traditional Energy Detection

The most widely used method of spectrum sensing is the traditional energy detection due to its low computational complexity [24]. The receiver does not require any prior knowledge of the primary user signal as energy detection is a noncoherent method of detection. The primary user is detected by measuring the energy and comparing it with a predetermined threshold. The threshold λ is computed using the assumed noise variance σ_{w}^2 and probability of false alarm P_{fa}, which generally depends on the channel characteristics. The problem of detecting the presence and absence of signal in spectrum sensing is typically formulated by the following binary hypothesis test [25],

$$\begin{aligned} H_0 &: y[n] = w[n] \\ H_1 &: y[n] = x[n] + w[n] \end{aligned}$$

where $y[n]$ represents the received signal, $x[n]$ is the transmitted wireless signal, and $w[n]$ is the zero mean complex circularly symmetric additive white Gaussian noise (AWGN). Further, $x[n] = s[n] \bigotimes h[n]$ where $s[n]$ denotes the primary user signal and $h[n]$ the channel impulse response [26]. Hypothesis H_0 represents the absence of a primary user signal and consists only the noise $w[n]$. On the other hand, hypothesis H_1 represents the presence of primary user signal $x[n]$ along with noise $w[n]$. The test statistic is computed as the energy of the received signal as given in Eq. (1),

$$T(y) = \frac{1}{N_{\mathrm{s}}} \sum_{n=0}^{N_{\mathrm{s}}-1} |y[n]|^2, \tag{1}$$

where N_{s} is the total number of samples sensed at the receiver. The test statistic follows a chi-square distribution. However, in practical cases, the test statistic can be approximated to a Gaussian distribution for large number of samples according to the central limit theorem (CLT) [27]. According to CLT any independent and identically distributed (IID) random variable with finite mean and variances approaches is a normal distribution when the number of samples N_{s} is large enough. Therefore, the distribution of test statistics can be accurately approximated with a normal distribution for sufficiently large number of samples. The above hypothesis can be written as in,

$$T(y) \sim N\left(\sigma_{\mathrm{w}}^2, \frac{1}{N_{\mathrm{s}}}\sigma_{\mathrm{v}}^2\right); \text{for hypothesis } H_0$$

$$T(y) \sim N\left(\sigma_{\mathrm{x}}^2 + \sigma_{\mathrm{w}}^2, \frac{1}{N_{\mathrm{s}}}\left(\sigma_{\mathrm{x}}^2 + \sigma_{\mathrm{w}}^2\right)^2\right); \text{for hypothesis } H_1$$

where σ_{x}^2 is the signal variance and σ_{w}^2 is the noise variance. The presence of an active signal is determined by comparing the energy (test statistics) with a predetermined threshold. The threshold λ is calculated using the knowledge of probability of false alarm P_{fa} and the assumed noise variance σ_{w}^2 of the received signal. The probability of false alarm P_{fa} is given as

$$p_{\mathrm{fa}} = Q\left(\frac{\lambda - \sigma_{\mathrm{w}}^2}{\sqrt{1/N_{\mathrm{s}}\sigma_{\mathrm{w}}^2}}\right) \tag{2}$$

and the probability of detection can be expressed as

$$p_{\mathrm{d}} = Q\left(\frac{\lambda - \left(\sigma_{\mathrm{w}}^2 + \sigma_{\mathrm{s}}^2\right)}{\sqrt{1/N_{\mathrm{s}}}\left(\sigma_{\mathrm{w}}^2 + \sigma_{\mathrm{s}}^2\right)}\right) \tag{3}$$

The threshold λ is determined from Eq. (2) as

$$\lambda = \Big(Q^{-1}(P_{\mathrm{fa}})\sqrt{{}^{1}/_{N_{\mathrm{s}}}{+1)}\sigma_{\mathrm{w}}^2} \tag{4}$$

The minimum number of samples required for spectrum sensing is obtained using Eqs. (2) and (3) [28],

$$N_{\min} = 2\left[Q^{-1}(p_{\mathrm{fa}}) - Q^{-1}(p_{\mathrm{d}})(1 + \mathrm{SNR})\right]^2 \mathrm{SNR}^{-2} \tag{5}$$

3.1.2 Subband-Based Energy Detection

When the available wideband is split into nonoverlapping subbands, the subband-based energy detection is performed at the output of the individual subbands. The energy is computed as the test statistic at the output of each subband and compared with a predetermined threshold. Filter bank-based methods are robust and efficient for multiband spectrum sensing where energy detection is performed at the subband level at the output of the FFT or analysis filter bank (AFB). The wideband signal is split into narrow signal bands using FFT or AFB. Similar to traditional energy detection, the subband signal can be expressed as follows:

$$H_0 : y_{\mathrm{k}}[m] = w_{\mathrm{k}}[m]$$
$$H_1 : y_{\mathrm{k}}[m] = x_{\mathrm{k}}[m] + w_{\mathrm{k}}[m]$$

where $y_k[m]$ is the received signal at the k^{th} subband ($k = 1, 2, \ldots, M$), M is the total number of subbands with $x_k[m] = H_k s_k[m]$, H_{k} represents the complex gain of the subbands, $s_k[m]$ is the input signal, and $w_k[m]$ is the noise samples of the subbands. Similar to traditional energy detection, noise follows the distribution $w_k[m] \sim N\left(0, \sigma^2_{\mathrm{w},k}\right)$ and signal $x_k[m] \sim N\left(0, \sigma^2_{\mathrm{x},k}\right)$ with $\sigma^2_{\mathrm{w},k}$ being the noise variance and $\sigma^2_{\mathrm{x},k}$ the signal variance [26]. If $\sigma^2_{\mathrm{w},k}$ is the noise variance of the wideband channel, the subband noise variance is $\frac{\sigma^2_{\mathrm{w}}}{M}$. The energy at the output of individual subbands is considered as the test statistic,

$$Y_k = \frac{1}{L} \sum_{m=0}^{L-1} y_k[m]^2 \tag{6}$$

where $L = \left(\frac{N_{\mathrm{s}}}{M}\right)$ is the number of samples in each subband with M number of subbands for sensing and N_{s} total number of samples received. The presence and absence of a primary user signal is written in terms of the following two hypotheses:

$$y_k(m) \sim N\left(\sigma^2_{\mathrm{w},k}, \frac{1}{L}\sigma^4_{\mathrm{w},k}\right); \text{for hypothesis } H_0$$
$$y_k(m) \sim N\left(\sigma^2_{\mathrm{x},k} + \sigma^2_{\mathrm{w},k}, \frac{1}{L}\left(\sigma^2_{\mathrm{x},k} + \sigma^2_{\mathrm{w},k}\right)^2\right); \text{for hypothesis } H_1$$

The number of samples for each stage needs to be large enough to perform energy detection even in low SNR. The minimum number of samples required in each stage can be calculated using the relation in Eq. (5).

3.2 Matched Filter

Matched filter (MF) is a non-blind spectrum sensing technique with coherent detection. Prior knowledge of the primary user signal is required in MF. The known primary user information is correlated with the received signal to detect the presence of primary user signal and maximize the signal-to-noise ratio (SNR). The matched filter requires short sensing time and achieves good detection performance with low probability of missed detection and false alarm. The drawback of this method is that it requires knowledge about primary user signal such as operating frequency, bandwidth, modulation type, and packet format. Therefore, the technique is not applicable when the information regarding the primary users are unknown [23].

3.3 Cyclostationary Feature Detection

Cyclostationary feature detection (CFD) technique exploits the cyclostationary features of the signal for spectrum sensing. A signal is considered to be cyclostationary if its statistical properties vary cyclically with time. When the modulated signals are combined with sinusoidal signals and pulse trains, they exhibit periodicity. The cyclostationary features are exploited from the periodicity using signal statistics such as mean and autocorrelation. The cyclic autocorrelation function (CAF) of the received signal $x(t)$ can be expressed as

$$R_{\mathrm{x}}^{(\alpha)} = E[x(t)x^{*}(t-\tau)\exp(-j2\pi\alpha\tau)], \tag{7}$$

where α is the cyclic frequency, $E[.]$ is the expectation operation, and $*$ denotes complex conjugation. Using Fourier series expansion, CAF can be expressed as cyclic spectral density (CSD) [23].

When the cyclic frequency α and fundamental frequencies become equal, CSD exhibits peaks. Therefore, under hypothesis H_0, the noise alone is present, and the CSD function does not exhibit peaks as the noise is nonstationary. On the other hand, in hypothesis H_1, peaks occur due to the signal and presence of noise. Therefore, CFD distinguishes the noise from the PU signal and can also be used for the detection of weak signal in case of very low SNR. CFD does not require prior knowledge of primary user waveform. The performance of CFD can be improved at a given SNR by increasing the number of samples, however at the cost of sensing time. The limitation of cyclostationary feature detection is that it requires longer processing time compared to the energy detection and matched filter detection techniques.

4 Wideband Spectrum Sensing

An important challenge in CR is sensing of multiple narrowband channels over a wideband spectrum. Most of the existing spectrum sensing algorithms discussed above are suitable for narrowband spectrum sensing, which exploits the spectral opportunities over narrow frequency range. To achieve higher throughput, CR needs to exploit spectral opportunities over a wide range of frequencies, from hundreds of megahertz to several gigahertz [14]. In cases where spectral opportunities are to be identified in ultra-high frequency (UHF) TV band (between 300 MHz and 3 GHz), wideband spectrum sensing techniques are to be employed. Narrowband spectrum sensing techniques cannot be applied in this scenario as they can make only binary decision on the whole spectrum and the spectral opportunities within the wideband cannot be identified. The benefits of multichannel/wideband spectrum sensing for CR networks are the secondary user throughput capacity can be maximized and aggregate interference of primary user networks can be reduced [29]. Multiband joint detection techniques have also been utilized to maximize the secondary user throughput capacity and reduce interference of primary users [3]. The multiband spectrum sensing has a few challenges due to the following reasons as discussed in [2].

- The available wideband for spectrum sensing may not be contiguous.
- A small portion of bandwidth may be occupied by a wireless device, and the entire bandwidth may be considered unavailable. (For example, in IEEE 802.22, wireless microphone occupies only 200 kHz of a 6 MHz TV channel, and the entire TV channel would be considered occupied.)
- If a portion of signal is in deep fade, the subbands may consider that portion as a spectral hole. Therefore, if a secondary user is allocated to that portion of the spectrum, interference would occur with the existing primary user.

The multiband spectrum sensing is categorized into serial-based detectors, parallel-based detectors, and wideband-based detectors. Serial sensing is simple to implement; however, the technique is slow and undesirable when the subbands are more. Parallel sensing provides faster detection at the expense of RF components and complex signal processing. The common multiband sensing techniques use reconfigurable band-pass filters, tunable oscillators, filter banks, wavelets, and blind sensing. A comparison between the different multiband spectrum sensing methods is provided in [2]. A detailed review and comparison between the different spectrum sensing methods along with advantages, disadvantages, and challenges are also provided in [12].

Further, wideband spectrum sensing techniques are broadly classified into two types:

- Nyquist wideband SS
- Sub-Nyquist wideband SS

In Nyquist wideband spectrum sensing, digital signals are sampled at or above the Nyquist rate, and in sub-Nyquist technique the signals are sampled below the Nyquist rate. Standard analog-to-digital converters (ADC) and digital signal processing techniques are used in Nyquist wideband spectrum sensing. After the received signals are sampled, serial to parallel conversions are required for further processing of the signals. A widely used Nyquist wideband spectrum sensing technique is the filter bank-based spectrum sensing. In filter bank-based techniques, fast Fourier transform (FFT) is used to convert the signal to a series of narrowband spectra. The spectral opportunities were identified by applying the binary hypothesis test to the individual subbands. In most of the filter bank techniques, energy detection was chosen as the test statistic. The threshold for detection was jointly chosen using optimization techniques.

The sub-Nyquist approach overcomes the limitations of Nyquist approach resulting from high sampling rate and computation complexity. In sub-Nyquist sensing, the wideband signal is acquired by using sampling rates lower than Nyquist rate. The compressed sensing techniques are applied for sub-Nyquist sampling.

5 Filter Bank Techniques for Spectrum Sensing

The concept of multirate filter banks was proposed for spectrum sensing initially by Farhang [30]. Filter banks are implemented by shifting a low-pass prototype filter. The first subband is estimated using the prototype filter, and other subbands are obtained by modulating the prototype filter. The total bandwidth is split into narrow nonoverlapping subbands using multiple band-pass filters. Multicarrier techniques were also suggested for spectrum sensing, where OFDM was the first multicarrier technique proposed for CR [31]. OFDM was considered as a suitable candidate for CR as FFT can be used for spectral analysis and demodulator for OFDM signal. However, the limitation of using the OFDM for CR application is the presence of large sidelobes in the response of the subband filters due to 13 dB attenuation of FFT, which may lead to interference between different users because of the spectral leakage. Moreover, OFDM techniques lack high spectral dynamic range and are not suitable for detection of low-power primary users. To overcome this issue, the rectangular pulse shape in OFDM was replaced with a smooth edge pulse shape filters called filtered OFDM. Filter bank multicarrier (FBMC) and filtered OFDM become alternate solutions to overcome the above limitations. FBMC reduces the spectrum leakage compared to cyclic prefixed OFDM systems and is capable of identifying multiple users with different center frequencies and spectral gaps between users efficiently with flexibility [10]. Different FBMC schemes reported in literature include staggered modulated multitone (SMT), filtered multitone (FMT), and cosine-modulated multitone (CMT). A comparison of filter bank multicarrier methods in cognitive radio systems is presented in [32].

The spectrum efficiency can be increased by designing prototype filters with acceptable subband attenuation. Therefore, filter banks are considered to be an alternate solution for wideband spectrum sensing. To achieve high spectral dynamic range in filter banks, the length of the prototype filter also needs to be adjusted. Multi-taper method (MT) is shown as a near optimal sensing method, even though MT has high computational complexity [1, 33]. However, similar performance can be achieved with filter banks using prolate filters with lower computational complexity [30]. Discrete Fourier transform (DFT) and modified DFT filter bank with root-Nyquist filter have also been exploited for spectrum sensing in wideband cognitive radios.

Different types of filter banks have been used in CR system for varied applications. Multistage polyphase filter bank techniques were used for the detection of center frequency of primary users with low computational complexity and higher precision. FFT-based filter bank techniques have been used for sensing Wireless

Local Area Networks (WLAN) such as OFDM based on IEEE 802.11 system and Wireless Personal Area Network (WPAN) with bluetooth designated to operate on 2.4 GHz ISM band. Multi-resolution filter banks based on fast filter bank design with varying spectral resolution were applied for spectrum sensing in military radio receivers. Tree-structured DFT based filter banks have also been used for estimating the center frequencies and spectral edges of primary user signals.

5.1 Sensing Architecture Based on Filter Banks

Filter banks consist of an analysis filter bank (AFB) and synthesis filter bank (SFB). Synthesis filter banks are sufficient to extract the signal components of each subband from the wideband RF signals. The basic filter bank spectrum sensing architecture is illustrated in Fig. 2. The RF module is followed by wideband ADC to sample the RF signal. Different filter bank structures like cosine-modulated filter bank (CMFB), DFT, and polyphase DFT can be considered. In case of complex modulated filter banks, the complete filter bank structure can be realized using complex modulation of a single prototype filter.

In general, multiband sensing utilizes energy detection techniques due to the reduced computational complexity. Different methods such as periodogram method, multi-taper method (MTM), and filter bank methods have investigated energy detection for spectrum sensing in literature. Farhang has shown in [30] that DFT filter banks based on energy detection are more promising in terms of accuracy if noise variance is known. Energy detection is the most common method as it has low computational and implementation complexities. Energy (power) is computed at the output of individual subband and considered as the test statistic. The presence and absence of the signal is detected by comparing the energy with a predefined

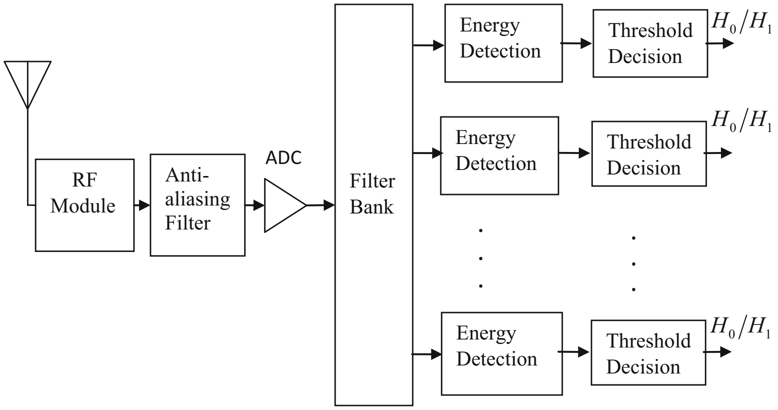


Fig. 2 Sensing Architecture Based on Filter Banks

threshold as explained in Sect. 3. The threshold is a function of probability of false alarm and noise variance of the channel.

Different filter bank structures have been used in CR for spectrum sensing like cosine-modulated filter banks and DFT-\FFT-based filter banks.

6 Multirate Filter Banks

The multichannel filter banks can be implemented using cosine modulation, FFT, DFT, or modified DFT filter banks. The filter banks can be implemented using complex modulation of a single prototype filter [34]. Multirate filter banks are designed with basic multirate signal processing techniques such as decimation and interpolation [4]. The polyphase representation in multirate is useful for computationally efficient implementation of polyphase filter banks. In general the analysis and synthesis subbands of the filter bank are simultaneously generated by applying an appropriate modulation scheme to the linear phase finite impulse response (FIR) prototype filter. The significance of prototype filter design in the implementation of filter banks to improve the overall performance is well proven. Therefore, the design of prototype filter is vital in the implementation of filter bank structures.

In general, filter bank designs can be categorized into two types:

- Perfect reconstruction (PR)
- Near-perfect reconstruction (NPR) or quadrature mirror filters (QMF)

Perfect reconstruction filters are alias-free filters, where the output is a delayed version of input. However, the implementation of PR filters is computationally complex, and for practical applications, NPR filters are adequate. The filter bank implementation focuses on NPR as they provide improved alias suppression in the subbands by relaxing PR constraint. As the same prototype filter is employed in the analysis and synthesis banks, the NPR filters have polyphase matrices which are paraunitary and, hence, have favorable numerical properties.

Perfect reconstruction (PR) filter banks satisfy the condition that the reconstructed signal

$\widehat{x}(n)$ need to be a scaled and delayed version of the input $x(n)$.

$$\widehat{x}(n) = cx(n - n_0) \tag{8}$$

The reconstructed signal can be represented using z-transform as

$$\widehat{X}(z) = T(z)X(z) \tag{9}$$

The PR condition indicates that aliasing is canceled, and distortion function $T(z)$ is forced to be a delay. The optimization of the prototype coefficients for perfect reconstruction is highly nonlinear. In case of NPR or approximate reconstruction, the analysis and synthesis filters $H_k(z)$ and $F_k(z)$, respectively, are chosen in such a way that the adjacent subband aliasing gets cancelled. The distortion function $T(z)$ is approximately a delay. The approximate systems mentioned are called pseudo-QMF banks and are acceptable for practical applications.

Extensive research has been carried out to find an optimal prototype filter for complex modulated filter banks. The optimization techniques for prototype filter design can be categorized into three types:

- Frequency sampling techniques
- Window-based techniques
- Direct optimization of filter coefficients

In order to overcome the limitation of having M different transfer functions, which provide perfect reconstruction, the complex modulated filter banks can be realized from a single low-pass prototype filter. The complex modulated filter banks generally use the basic pseudo-quadrature mirror filter principle. The filter banks are realized by equidistant frequency shifts of a prototype filter. For near perfect reconstruction in the filter banks, the low-pass prototype filters have to satisfy the following conditions [35]:

1. Prototype filter has to be band limited.

$$\left|H\left(e^{j\omega}\right)\right| \approx 0 \; |\omega| > \frac{\pi}{M} \tag{10}$$

2. Frequency response of prototype filter has to be pairwise power complementary.

$$\left|H\left(e^{j\omega}\right)\right|^2 + \left|H\left(e^{\left(\frac{\pi}{M}-\omega\right)}\right)\right|^2 \approx 1, 0 \le \omega \le \frac{\pi}{M} \tag{11}$$

The advantages of such filters are twofold:

- The cost of implementing M analysis filter bank includes the cost of one prototype filter and modulation overhead. The cost of M synthesis filter is similar to an analysis filter.
- Optimization of prototype filter alone is required for the implementation of filter bank structure.

The prototype filters should have sufficient stopband attenuation to suppress the aliasing components. The realization of DFT, cosine modulation, and DFT-based polyphase filter banks are discussed in the following subsections.

6.1 DFT Filter Banks

In DFT filter bank, the M analysis filter banks, $H_k(z), k = 0, 1, \ldots, M-1$, are realized by frequency shifting the transfer function $H(z)$ of the prototype filter $h(n)$. The impulse response of the prototype filter is multiplied by a factor $e^{jn\Omega_0}$.

Fig. 3 Frequency response of prototype filter

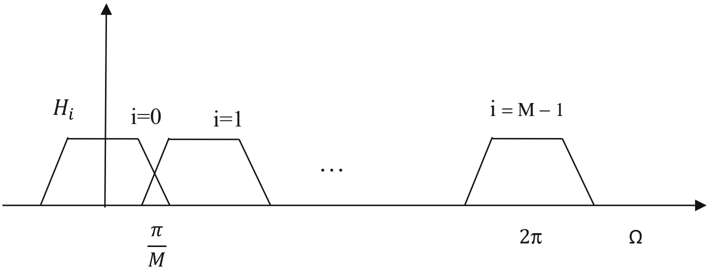


Fig. 4 Frequency shifted version of the prototype filter for analysis filter banks

Further, the frequency response of the prototype filter $H(e^{j\Omega})$ is shifted right as $H\left(e^{j(\Omega-\Omega_0)}\right)$. The M analysis filter response can be expressed as

$$H_i\left(e^{j\Omega}\right) = H\left(e^{j\left(\Omega-\frac{2\pi i}{M}\right)}\right), i = 1, 2, \ldots, M-1 \tag{12}$$

In case, $W_M = e^{j2\pi/M}$, the Z-transform of the analysis filters can be written as

$$H_i(z) = H\left(zW_M^i\right) \tag{13}$$

The frequency response of the prototype filter and the shifted versions of the prototype filter for the generation of subbands filter are shown in Figs. 3 and 4, respectively.

6.2 Cosine-Modulated Filter Banks

The cosine-modulated filter banks (CMFB) are also pseudo-QMF and satisfy the NPR conditions. Cosine-modulated filters can easily maintain maximally decimated NPR condition. Among the NPR FIR filter banks, CMFB is considered to be simple both in terms of design and implementation complexities. Initially, the prototype filter is designed satisfying the power complementary and band-limiting conditions specified in Eqs. (10) and (11). In an M channel CMFB, the impulse responses of the analysis and synthesis filters are $h_k(n)$ and $f_k(n)$, respectively. The filter banks are cosine-modulated versions of the prototype filter $h(n)$. The prototype filter is linear phase FIR Type I filter. The analysis and synthesis filters are given by Eqs. (14) and (15) for $0 \le n \le N-1$ and $0 \le k \le M-1$ as in [4],

$$h_k(n) = 2h(n)\cos\left[\frac{\pi}{M}\left(k+\frac{1}{2}\right)\left(n-\frac{N-1}{2}\right)+(-1)^k\frac{\pi}{4}\right] \tag{14}$$

$$f_k(n) = 2h(n)\cos\left[\frac{\pi}{M}\left(k+\frac{1}{2}\right)\left(n-\frac{N-1}{2}\right)-(-1)^k\frac{\pi}{4}\right] \tag{15}$$

where $k=0,1,2,\ldots M-1$ is the number of subbands in the filter bank. By choosing a linear phase FIR Type I filter, the phase distortion can be eliminated completely. The amplitude distortion is reduced, when the band-limiting condition stated in Eq. (10) is satisfied and if the filters are pairwise power complementary as in Eq. (11), the aliasing error can also be reduced.

6.3 Spectrum Sensing with Cosine-Modulated Filter Bank

Among the different filter bank-based methods such as orthogonal multiplexed quadrature amplitude modulation (OQAM), cosine-modulated multitone (CMT), and filtered multitone (FMT), CMT is more desirable as it provides higher bandwidth efficiency compared to FMT and lower sidelobes than OQAM. The performance of spectrum estimation is characterized by different parameters such as frequency resolution, spectrum leakage, and estimation time. The above three parameters can be regulated using CMFB with a proper design of prototype filters. In wideband spectrum sensing, signals are filtered using CMFB followed by power spectrum estimation [36]. CMFB can detect primary users over contiguous channel having different bandwidths. A transceiver framework based on cosine-modulated filter bank was proposed in [37] for cognitive access to TV white spaces. The spectrum sensing is performed using the system model described in Sect. 3 for subband-based energy detection.

6.4 Polyphase Filter Banks

Polyphase filter bank structure reduces the complexity of the filter bank implementation using the noble identities of multirate systems. Polyphase filter banks are efficiently designed using FFT when the number of subbands M is a power of two. The DFT filters can be modified to get a better stopband attenuation compared to 13 dB of DFT at the cost of one prototype filter. The polyphase decomposition of the prototype filter enables to implement the filter bank in an efficient manner [4, 38]. The transfer function of a FIR prototype filter $h(n)$ is given by

$$H(z) = \sum_{n=-\infty}^{\infty} h(n) z^{-n} \tag{16}$$

The transfer function in Eq. (16) can be decomposed into polyphase components as in Eq. (17)

$$\begin{aligned} H(z) = \sum_{n=-\infty}^{\infty} h(nM) z^{-nM} + z^{-1} \sum_{n=-\infty}^{\infty} h(nM+1) z^{-nM} + \ldots \\ + \mathrm{z}^{-(M-1)} \sum_{n=-\infty}^{\infty} h(nM+M-1) z^{-nM} \end{aligned} \tag{17}$$

Eq. (17) can be written in short as in Eq. (18).

$$H(z) = \sum_{l=0}^{M-1} z^{-l} E_l\left(z^M\right) \tag{18}$$

The above equation represents a Type-1 polyphase filter. Similarly, l^{th} polyphase component of the filter bank is defined as

$$E_l(z) = \sum_{n=-\infty}^{\infty} e_l(n) z^{-n} \tag{19}$$

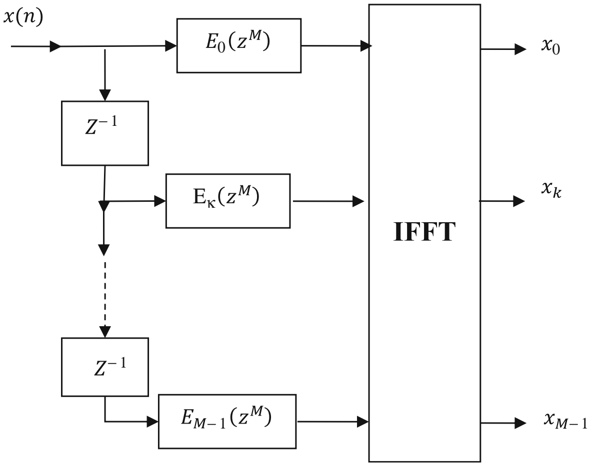
where, $e_l(n) = h(Mn + l)$. The Type-2 polyphase decomposition of the Eq. 10 can be expressed as

$$H(z) = \sum_{l=0}^{M-1} z^{-(M-1-l)} R_l\left(z^M\right) \tag{20}$$

Polyphase implementation simplifies the theoretical results, and computationally efficient filter banks can be realized. The filter bank implementation with uniform DFT bank using polyphase decomposition is shown in Fig. 5. Using noble identities the polyphase uniform DFT filter bank structure with decimators can be redrawn as shown in Fig. 6. Since the downsampler are shifted toward the input side, the polyphase subband filters are computed at a low sampling rate, which reduces the computational complexity by a factor of M. Due to the polyphase decomposition of prototype filter, the polyphase subband filters are shorter compared to the original filters by a factor of M.

The computational complexity of polyphase filter banks is $N + M\log_2 M$, where N is the length of the prototype filter. Polyphase filter bank reduces the computational complexity to a large extent compared to the complexity NM of direct

Fig. 5 Uniform DFT bank using polyphase decomposition



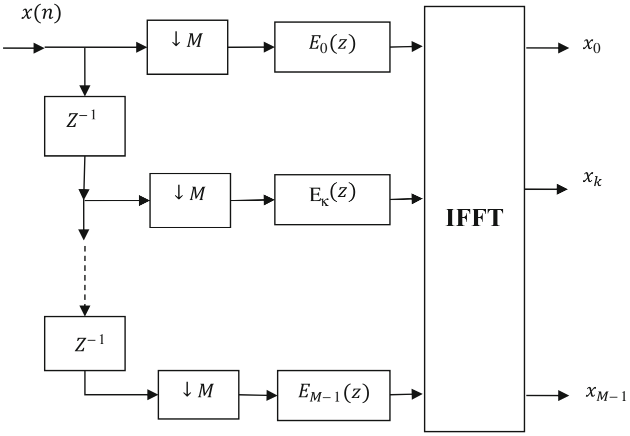


Fig. 6 Polyphase filter banks with efficient implementation using noble identities

implementation. The important advantage of multirate polyphase filter banks is that it allows efficient implementation of filter bank structure due to the polyphase decomposition. Moreover, the entire computational complexity of the polyphase filter bank is reduced to the design of a prototype filter and M point FFT.

7 System Model for Filter Bank Spectrum Sensing

The wideband signal for spectrum sensing is localized to various subband frequencies using filter bank structures. The available spectrum band is divided into *M* nonoverlapping uniform subbands, where *M* is the number of subbands in the filter bank. The output of each subband, $x_k(n)$, is assumed to be a random process, obtained from a random process input $s_k(n)$ passing through a linear subband filter of frequency response H_k, where $x_k(n) = H_k s_k(n)$. The received signal $y_k(n)$ can be modeled as [26],

$$y_k(n) = x_k(n) + w_k(n), k = 0, 1, 2, \ldots, M - 1,$$

where $x_k(n)$ is the active signal and $w_k(n)$ is the additive white Gaussian noise with zero mean and variance σ_w^2. In order to detect the presence of primary user signal, a binary hypothesis is defined as [40],

$$\begin{aligned} H_{o,k} &: y_k(n) = w_k(n) \text{ absence of signal} \\ H_{1,k} &: y_k(n) = x_k(n) + w_k(n) \text{ presence of signal} \end{aligned}$$

We consider the test statistic as the energy at the output of each subband as given by

$$y_k(n) = \frac{1}{L} \sum_{n=0}^{L-1} x_k^2(n)$$

where $L = \left(\frac{N_s}{M}\right)$, is the number of samples in each subband and $k = 0, 1, 2, \ldots, M - 1$.When the number of samples is increased, the chi-square distribution approximate to a normal distribution from the central limit theorem as discussed in Sect. 3.

The presence of an active signal in a specified subband can be determined by comparing the energy in that subband with a predetermined threshold. The available wideband is divided into M nonoverlapping subbands using filter banks. Energy detection is performed on the output of each subband and compared with a predefined threshold. Depending on the threshold decision, a subband is considered to have the presence of primary user referred as ’1’ or absence of primary user referred as ’0’ (spectral hole). This is referred to as binary detection and illustrated in Fig. 7.

7.1 Calculation of Threshold

The threshold λ can be calculated using the knowledge of probability of false alarm P_{fa} and noise variance σ_{w}^2 of the received signal as explained in Sect. 3. The energy detector gives the best performance with known noise variance and performance deteriorates when the noise variance is uncertain. To improve the detection performance, the noise variance can be estimated at the receiver before spectrum detection using noise variance estimation techniques.

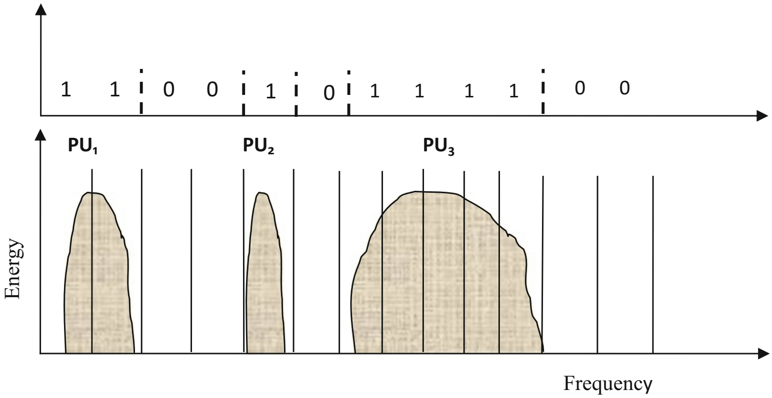


Fig. 7 Illustration of binary detection in wideband spectrum sensing

8 Multistage Filter Bank Spectrum Sensing

Spectrum sensing can be performed using filter bank structures for varying granularity

bands. The number of subbands M determines the granularity of sensing bandwidth. For spectrum sensing with finer granularity bands, M needs to be increased, whereas for spectrum sensing with coarser granularity bands, M needs to be decreased. The advantage of varying granularity band is that the spectrum utility and re-usability can be effectively increased. Moreover, the same structure can be used for spectrum reallocation due to the flexibility offered by the filter bank structure.

The granularity of the filter banks can be chosen specifically if the bandwidth of the primary users is known apriori. The energy is computed at the output of individual subbands as the test statistics. The threshold is calculated for specified probability of false alarm P_{fa} and known noise variance σ_{w}^2. The bandwidth efficiency could be effectively increased using finer granularity bands for spectrum sensing. When the number of subbands is increased, the spectral resolution of the filter banks gets increased, and better detection performance is achieved. The computational complexity can be reduced with efficient structures using polyphase filter bank discussed in subsequent sections. However, finer granularity bands increase the computational complexity of the filter bank structure used for spectrum sensing. In order to reduce the computational complexity, multistage filter bank structures can be considered. In the subsequent sections, we discuss multistage CMFB and multistage polyphase filter banks [9, 39].

8.1 Spectrum Detection with Multistage CMFB

Better sensing performance can be achieved in filter banks with finer resolution. However, this would increase the computational complexity of the filter bank structure. Therefore, to overcome the computational complexity, multistage filter banks and tree-structured DFT filter banks are investigated for sensing from coarser to finer resolution [7, 8]. In multistage or multi-resolution filter banks, at the initial stage, the total bandwidth is sensed using coarser spectral resolution (smaller number of subbands). The bandwidth of interest is identified depending on the sensing decision, and only those frequency bands are further sensed with finer spectral resolution. Multi-resolution filter bank techniques include fast filter bank (FFB) based on frequency-response masking (FRM), coarser to finer spectrum sensing using wavelet transforms, and FFT-based multi-resolution spectrum sensing using multiple antennas [6].

In case of multistage filter bank spectrum sensing, the available bandwidth is initially divided into nonoverlapping subbands with coarser spectral resolution of M_1 subbands as illustrated in Fig. 8. When narrowband users appear in wideband spectrum and the bandwidth of sensing is sparse as shown in Fig. 8, the subbands of interest can be detected in the first stage. The detected subbands can be sensed further with finer spectral resolution in the next stage with a spectral resolution of M_2 subbands. As the narrowband users are identified in coarser resolution, only the detected subbands are sensed further with finer resolution. Therefore, the computational complexity is reduced, and better sensing performance can be achieved.

Multistage spectrum sensing can be performed by defining two thresholds based on different probability of false alarm P_{fa} depending on the channel conditions. Energy detection is performed using the predefined thresholds with different probability of false alarms, calculated as discussed in Sect. 3. Two thresholds, λ_1 and λ_2, are the calculated based on different probability of false alarm ($\lambda_2 > \lambda_1$). If the energy is above the threshold λ_2, it can be concluded as the presence of primary

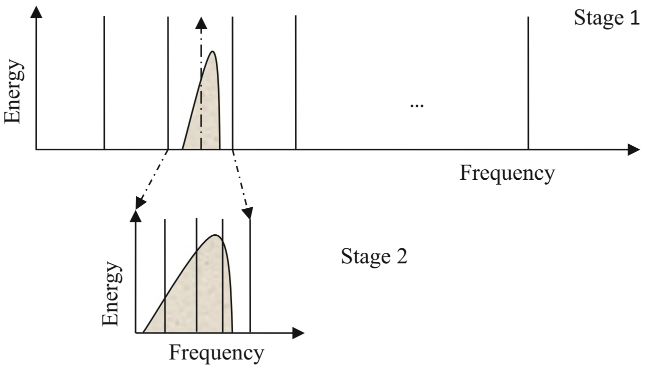


Fig. 8. Illustration for multistage filter bank spectrum sensing from coarser to finer spectral resolution

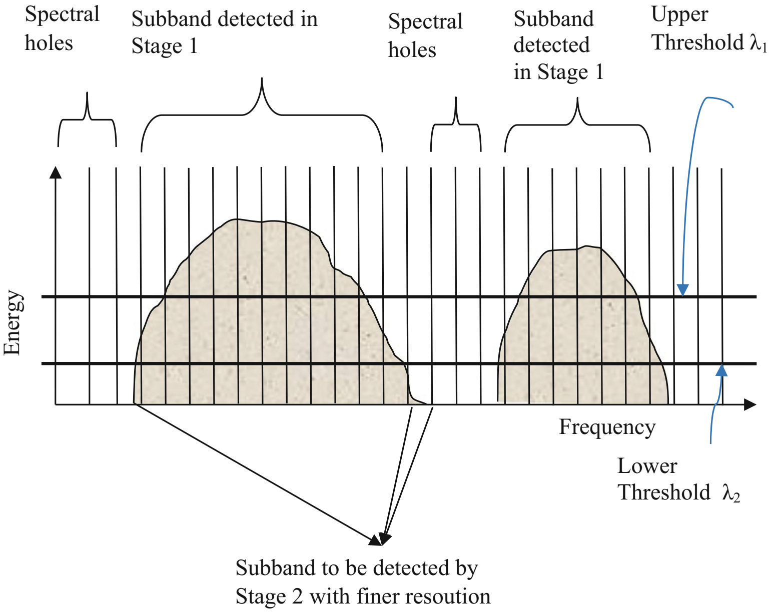


Fig. 9 Illustration of threshold decision with multistage spectrum sensing

user. If the energy is below λ_1, it is decided as the (absence of primary user) presence of a spectral hole or absence of primary user, and if the energy is between λ_1 and λ_2, there is a possibility of spectral hole within the subband.

The multistage spectrum sensing with energy distribution and thresholds are explained in Fig. 9. Only the subbands having energy between λ_1 and λ_2 have to be sensed in the next level with finer granularity. Multiple spectral gaps can be identified in an efficient and flexible way using the multistage methods with reduced complexity since the whole band need not be sensed with finer granularity.

The significance of multistage spectrum sensing is summarized as follows:

1. The probability of detection is improved with finer granularity bands.
2. Multistage filter banks reduce the computational complexity as the whole band need not be sensed with the finer granularity.
3. Spectrum sensing can be performed from a coarser to finer spectral resolution.

8.2 *Spectrum Sensing with Polyphase Filter Banks*

Wideband spectrum sensing using filter banks has proved to be robust and efficient. Filter bank-based physical layer design for CR systems was introduced to perform simultaneous spectrum sensing and transmission. Filter bank techniques can reduce computational complexity and improve spectral analysis in cognitive radio applications. For fractional utilization of spectrum, the center frequency and spectral edges of the primary user need to be estimated which can be done using polyphase filter banks. The complexity in filter bank implementation can be reduced to a large extent using polyphase filter bank structures. Polyphase filter banks are efficiently designed using FFT when the number of subbands M is a power of two. The problem of estimating the center frequency and spectral edges of primary users in a wideband spectrum using polyphase filter banks is discussed below.

8.3 Multistage Polyphase Filter Banks

Multistage polyphase filter bank method detects the presence of primary user and identifies the spectral holes. In addition, for fractional utilization of spectrum bands, the center frequency of the primary user can be estimated with higher precision using filter banks and subband-based energy detection along with centroid/center of mass method [8, 9]. It is well known that the detection accuracy depends on the number of subbands M in the filter bank. The computational complexity of the filter bank increases with higher values of M. However, the complexity is reduced by using multistage polyphase filter bank structure. The primary users are detected by computing the signal energy (power) at output of the individual subbands. The algorithm for the detection of unused spectrum (spectrum holes) starts from a coarser spectral resolution (smaller number of subbands) at the first stage to reduce computational complexity. Single user and multiuser scenarios are considered in wideband for spectrum sensing using multistage polyphase filter banks. The detection of single and multiple users in widebands is elaborated in the following subsections.

8.4 Single User Detection in Wideband Spectrum

Polyphase filter are useful in detection of narrowband single user in a wideband spectrum [9, 38]. For example, consider the detection of wireless microphone (WM) in the presence of a signal that follows IEEE 802.22 WRAN standard. In IEEE 802.22 WRAN standard, spectrum sensing has to be done to allow television (TV) services and wireless microphones to coexist. WMs are low-power licensed users and are allowed by Federal Communications Commission (FCC) to operate on vacant TV channels without causing interference. The detection of WM is difficult due to the low power transmission (typically 50 mW for 100 m coverage) and small bandwidth occupancy (200 kHz). In IEEE 802.22 WRAN standard, when a WM appears anywhere in the TV channel, the whole channel of 6 MHz has to be evacuated to avoid interference. However, TV channels can be utilized fractionally when the exact position of the WM is detected [8, 41]. Hence, there are several challenges in the detection of WM signals/narrowband users.

Multistage polyphase filter bank method can detect the presence of WM and estimate the center frequency of the WM with better precision by using the centroid method. The centroid-based technique can detect the presence of WM in the first stage itself, when spectrum of WM lies partly in one subband and partly in adjacent subband. When the WM is detected in the first stage, it reduces the computational complexity and latency and achieves fast sensing. However, if WM appears exclusively within a single subband, an additional stage is required to detect and estimate the center frequency of WM with finer spectral resolution. In such cases, WM can be detected in the second stage without ambiguity.

The multistage polyphase filter banks can be designed to detect the presence of WM anywhere within a TV channel (6 MHz) and to estimate the center frequency of WM taking into account the following scenarios:

Case 1: If the signal spectrum of WM lies partly in one subband and partly in the adjacent subband as shown in Fig. 10, the center frequency of WM can be either in one of the subbands or between two subbands. The center frequency in such a case is estimated using the centroid method as described in the subsequent sections.

Case 2: If the signal spectrum of WM is in the middle of two adjacent subbands as shown in Fig. 11, the energy at the output of two subbands will be equal. That is, the center frequency of WM is at the midpoint of the two subbands. Therefore, finer level of detection is not necessary, which in turn reduces the computational complexity and latency.

Case 3: If the signal spectrum of WM appears exclusively within a subband as shown in Fig. 12, the output of first stage is passed to the input of the next stage filter

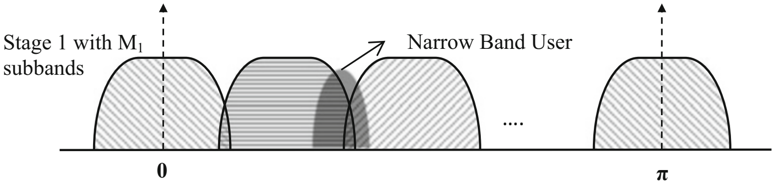


Fig. 10 Case 1: Narrowband user appears anywhere between two consecutive subbands

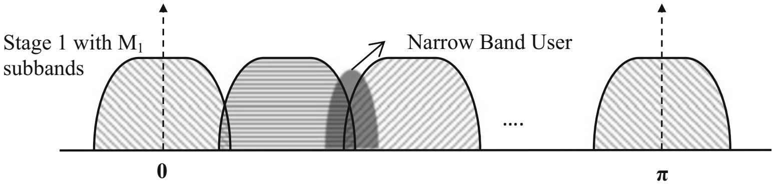


Fig. 11 Case 2: Narrowband user appears exactly between two subbands

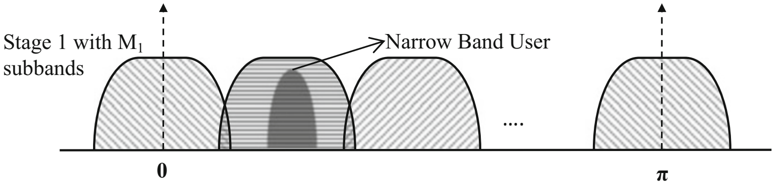


Fig. 12 Case 3: Narrowband user appears exclusively within a subband

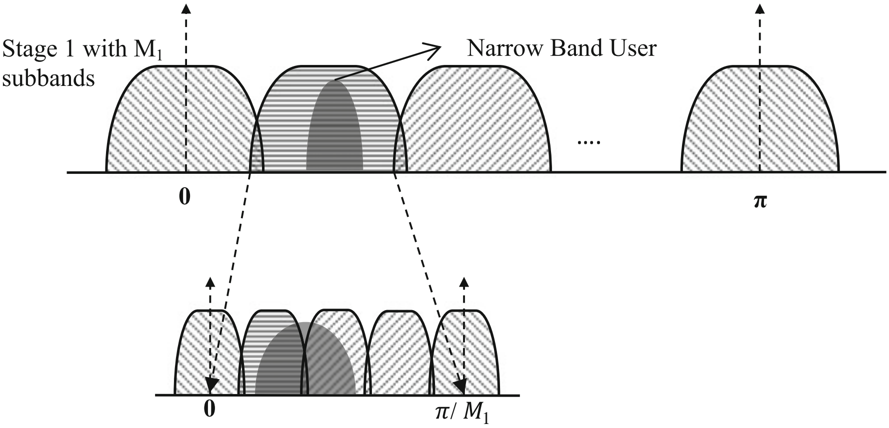


Fig. 13 Detection of narrowband users using two-stage filter bank

bank to estimate the center frequency with a finer spectral resolution. The process is illustrated in Fig. 13 where only two stages are required to detect the presence of the WM and to accurately estimate its center frequency. The center frequency is estimated in the second stage using DFT polyphase filter bank and centroid method.

The procedure followed for multistage spectrum sensing is briefed in the following two steps and illustrated in Fig. 13:

Step 1: The bandwidth of sensing is divided coarsely into M_1 subbands and sensed through the M_1 subband DFT polyphase filter bank. Energy detection is performed at the output of each subband, considering energy (power) as the test statistic to decide the presence or absence of the WM in the subbands. The detection and estimation of center frequency of WM as per Case 1 or Case 2 is done in the first stage. If the signal spectrum of WM is as per Case 3, the output of the sensed subband is further processed with finer resolution as in Step 2.

Step 2: The output of first stage is sensed in the next level with M_2 subbands. The signal energy (power) at the output of the subband is considered as the test statistic. At this level, the spectrum is sensed with a spectral resolution of π/M_1M_2.

The proposed method can be summarized as:

(i) If WM appear anywhere within consecutive subbands (Case 1 and Case 2), the center frequency of WM is estimated accurately using the centroid method in single stage.
(ii) If WM appears anywhere exclusively within any subband (Case 3), the output of the sensed subband is further processed with finer resolution as in Step 2.

8.5 Center Frequency Detection Using Centroid Method

The center frequency of the narrowband frequency can be detected when the narrowband user appears between the two subbands. The center of each subband represents the energy in that subband resolution as shown in Fig. 14. The energies can be modeled as a trapezoid, and the center frequency can be calculated from the centroid of the trapezoid. The top edge of the distribution can be defined using a linear function as,

$$f(x) = b + \frac{x}{h}(a-b).$$

The area of the trapezoid is given as $A = \frac{h}{2}(a-b)$.
The centroid in the x direction is computed as

$$\begin{aligned} A\bar{x} &= \int_0^h x f(x) dx \\ &= \int_0^h x\left(b + \frac{x}{h}(a-b)\right) dx \\ &= \frac{h^2}{6}(2a+b) \end{aligned}$$

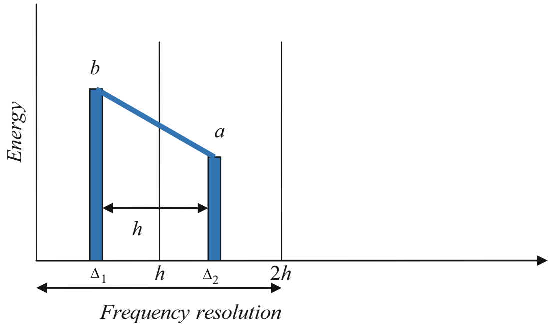


Fig. 14 Illustration of centroid method

where $\bar{x} = \frac{h}{3}\left(\frac{2a+b}{a+b}\right)$, and A is the area of the trapezoid. Here, $\bar{x}$ represents the centroid of the trapezoid.

In case of equal energy at the output of individual subband, i.e., when $a = b$, the midpoint can be verified as $\bar{x} = \frac{h}{2}$. The center frequency of the narrowband user is related to h which represents the granularity of filter bank M, a and b are related to the energies E_1 and E_2 of the adjacent subbands. The minimum of two subband energies are represented as a, i.e., $a = \min(E_1, E_2)$ and maximum as $b = \max(E_1, E_2)$. Thus, the estimated center frequency $\widehat{f}_c$ can be expressed as follows:

$$\widehat{f}_c = \frac{M}{3}\left(\frac{2a+b}{E_1 + E_2}\right)$$

A generalized expression is obtained by considering the energies of subband E_i and adjacent subband E_{i+1}.

$$\widehat{f}_c = \frac{M}{3}\left(\frac{2\min(E_i, E_{i+1}) + \max(E_i, E_{i+1})}{E_i + E_{i+1}}\right)$$

The centroid method provides better accuracy in center frequency estimation when the number of detected subbands is almost two. If the number of detected subbands is beyond two, the centroid method does not provide accurate estimation of center frequency. Since the energy distribution can no longer be modeled as a trapezoid, the top edge cannot be written as a linear function. In such cases, a center of mass method is used for estimation of center frequency.

8.6 Center Frequency Detection Using Center of Mass

The center of mass method can also be used to estimate the center frequency. The energy of the subbands is related to the mass, and the distance is related to the frequency. Consider the energy in two subbands as E_1 and E_2, let Δ_1 and Δ_2 be the center point of spectral resolution in the subband bins as shown in Fig. 15. The center frequency for the energy bins can be obtained using the law of center of mass as

$$\widehat{f}_c = \frac{E_1\Delta_1 + E_2\Delta_2}{E_1 + E_2}$$

The expression can be extended for different subbands, and the center frequency can be calculated using the relation,

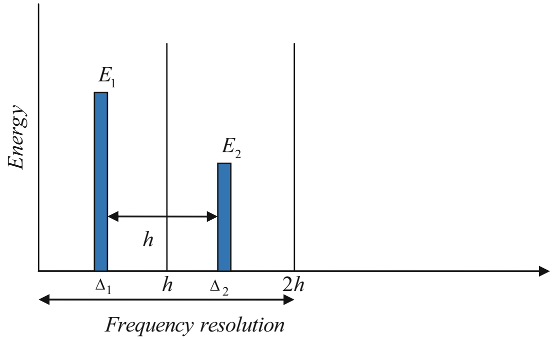


Fig. 15 Illustration of center of mass

$$\hat{f}_c = \frac{\sum_{k=1}^{n} E_k \Delta_k}{\sum_{k=1}^{n} E_k}$$

Thus the polyphase filter banks using the multirate system can be utilized for wideband spectrum sensing in CR. The filter banks are useful for varied applications such as detection of spectral holes and estimation of center frequencies of different primary users. The polyphase filter banks reduce the computational complexity and provide efficient realization of filter bank structure for wideband spectrum sensing.

9 Conclusions

The multirate filter bank-based spectrum analysis is applicable for multiband spectrum sensing in cognitive radio applications. Different filter banks have been analyzed for spectrum sensing in CR such as FFT, cosine-modulated filter banks, and polyphase filter banks. From the performance analysis of the different filter banks, the polyphase filter banks exhibit more reliable and efficient detection performance. Polyphase filter banks take advantage of the low spectral leakage property of the different subbands in the filter bank structure, which enables them to enhance the performance of multiband spectrum sensing in cognitive radio networks. Moreover, the computational complexity of polyphase filter banks is reduced when compared with other filter banks. In addition, the same filter bank sensing architecture can be applied for transmission and reception of signals in vacant subbands using channel adaptation techniques.

References

1. Haykin, S. (2005). Cognitive radio: Brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications, 23*(2), 201–220.
2. Hattab, G., & Ibnkahla, M. (2014). Multiband spectrum access: Great promises for future cognitive radio networks. *Proceedings of the IEEE, 102*(3), 282–306.
3. Quan, Z., Cui, S., Sayed, A. H., & Poor, H. V. (2009). Optimal multiband joint detection for spectrum sensing in cognitive radio networks. *IEEE Transactions on Signal Processing, 57*(3), 1128–1140.
4. Vaidyanathan, P. P. (1993). *Multirate systems and filter banks*. India: Pearson Education.
5. Dikmese, S., Srinivasan, S., Shaat, M., Bader, F., & Renfors, M. (2014). Spectrum sensing and resource allocation for multicarrier cognitive radio systems under interference and power constraints. *EURASIP Journal on Advances in Signal Processing*. doi:10.1186/1687-6180-2014-68.
6. Smitha, K. G., & Vinod, A. P. (2012). A multi-resolution fast filter bank for spectrum sensing in military radio receivers. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 20*(7), 1323–1327.
7. Narendar, M., Vinod, A. P., Madhukumar, A., & Krishna, A. K. (2013). A tree-structured dft filter bank based spectrum detector for estimation of radio channel edge frequencies in cognitive radios. *Physical Communication, 9*, 45–60.
8. Zhao, Z., Zhang, H., & Cui, Y. (2010). An efficient filter banks based multicarrier system in cognitive radio networks. *Radioengineering, 19*(4), 479–487.
9. Chris Prema, S., & Dasgupta, K. S. (2016). A low-complexity multistage polyphase filter bank for wireless microphone detection in CR. *Circuits Systems and Signal Processing, 36*(4), 1671–1685. doi:10.1007/s00034-016-0358-8.
10. Maliatsos, K., Adamis, A., & Kanatas, A. G. (2014). Elaborate analysis and design of filter-bank-based sensing for wideband cognitive radios. *EURASIP Journal on Advances in Signal Processing*. doi:10.1186/1687-6180-2014-82.
11. Mitola, J., & Maguire, G. Q. (1999). Cognitive radio: Making software radios more personal. *IEEE Personal Communications, 6*(4), 13–18.
12. Yucek, T., & Arslan, H. (2009). A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Communications Surveys & Tutorials, 11*(1), 116–130.
13. Biglieri, E., Goldsmith, A. J., Greenstein, L. J., Mandayam, N. B., & Poor, H. V. (2012). *Principles of cognitive radio*. New York: Cambridge University Press.
14. Sun, H., Nallanathan, A., Wang, C. X., & Chen, Y. (2013). Wideband spectrum sensing or cognitive radio networks: A survey. *IEEE Wireless Communications, 20*(2), 74–81.
15. Ghasemi, A., & Sousa, E. S. (2008). Spectrum sensing in cognitive radio networks: Requirements, challenges and design trade-offs. *IEEE Communications Magazine, 46*(4), 32–39.
16. Akyildiz, I. F., Lee, W. Y., Vuran, M. C., & Mohanty, S. (2006). Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks, 50*(13), 2127–2159.
17. Wang, B., & Liu, K. R. (2011). Advances in cognitive radio networks: A survey. *IEEE Journal of Selected Topics in Signal Processing, 5*(1), 5–23.
18. Jondral, F. K. (2005). Software-defined radio: Basics and evolution to cognitive radio. *EURASIP Journal on Wireless Communications and Networking, 2005*(3), 275–283.
19. Unnikrishnan, J., & Veeravalli, V. V. (2008). Cooperative sensing for primary detection in cognitive radio. *IEEE Journal of Selected Topics in Signal Processing, 2*(1), 18–27.
20. Ganesan, G., & Li, Y. (2007). Cooperative spectrum sensing in cognitive radio, part ii: Multiuser networks. *IEEE Transactions on Wireless Communications, 6*(6), 2214–2222.
21. Zeng, Y., & Liang, Y. C. (2009). Eigenvalue-based spectrum sensing algorithms for cognitive radio. *IEEE Transactions on Communications, 57*(6), 1784–1793.
22. Rao, S., & Singh, G. (2012). Wavelet based spectrum sensing techniques in cognitive radio. *Procedia Engineering, 38*, 880–888.

23. Lu, L., Zhou, X., Onunkwo, U., & Li, G. Y. (2012). Ten years of research in spectrum sensing and sharing in cognitive radio. *EURASIP Journal on Wireless Communications and Networking*. doi:10.1186/1687-1499-2012-28.
24. Urkowitz, H. (1967). Energy detection of unknown deterministic signals. *Proceedings of the IEEE, 55*(4), 523–531.
25. Steven, M. K. (1998). *Fundamentals of statistical signal processing, Vol. 11, detection theory*. Upper Saddle River: Prentice-Hall.
26. Dikmese, S., Sofotasios, P. C., Ihalainen, T., Renfors, M., & Valkama, M. (2015). Efficient energy detection methods for spectrum sensing under non-flat spectral characteristics. *IEEE Journal on Selected Areas in Communications, 33*(5), 755–770.
27. Atapattu, S., Tellambura, C., & Jiang, H. (2014). *Energy detection for spectrum sensing in cognitive radio*. New York: Springer.
28. Cabric, D., Tkachenko, A., Brodersen, R.W., (2006). Experimental study of spectrum sensing based on energy detection and network cooperation. Proceedings of the first international workshop on technology and policy for accessing spectrum ACM.
29. Paysarvi-Hoseini, P., & Beaulieu, N. C. (2012). On the benefits of multichannel/wideband spectrum sensing with non-uniform channel sensing durations for cognitive radio networks. *IEEE Transactions on Communications, 60*(9), 2434–2443.
30. Farhang-Boroujeny, B. (2008). Filter bank spectrum sensing for cognitive radios. *IEEE Transactions on Signal Processing, 56*(5), 1801–1811.
31. Farhang-Boroujeny, B., & Kempter, R. (2008). Multicarrier communication techniques for spectrum sensing and communication in cognitive radios. *IEEE Communications Magazine, 46*(4), 80–85.
32. Amini, P., Kempter, R., Farhang-Boroujeny, B., (2006). A comparison of alternative filter bank multicarrier methods for cognitive radio systems. In proceedings of the SDR technical conference and product exposition.
33. Thomson, D. J. (1982). Spectrum estimation and harmonic analysis. *Proceedings of the IEEE, 70*(9), 1055–1096.
34. Fliege, N. J. (1994). *Multirate digital signal processing: Multirate systems, filter banks, wavelets*. Chichester: John Wiley & Sons Inc.
35. Creusere, C. D., & Mitra, S. K. (1995). A simple method for designing high-quality prototype filters for m-band pseudo qmf banks. *IEEE Transactions on Signal Processing, 43*(4), 1005–1007.
36. Zhao, N., Pu, F., Xu, X., & Chen, N. (2015). Cognitive wideband spectrum sensing using cosine-modulated filter banks. *International Journal of Electronics, 102*(11), 1890–1901.
37. Zhao, N., Pu, F., Xu, X., & Chen, N. (2012). Cosine-modulated transceivers for tv white space cognitive access. *China Journal of Electronics, 21*(2), 362–366.
38. Zhang, H., Ruyet, D. L., Roviras, D., & Sun, H. (2014). Polyphase filter bank based multi-band spectrum sensing in cognitive radio systems. *International Journal of Communication Systems*. doi:10.1002/dac.2798.
39. Chris Prema, S., Sudha Rani, D., & Dasgupta, K. S. (2017). Spectral detection with multistage granularity bands using filter bank techniques for CR applications. *International Journal of Wireless and Mobile Computing Processing, 12*(1), 62–67.
40. Kay, S. M. (2013). *Fundamentals of statistical signal processing: Practical algorithm development*. New Jersey: Pearson Education.
41. Kim, C. J., et al. (2006). Fractional bw usage for wran systems. IEEE 802.22-06/0117r0.

Design of Nonuniform Linear-Phase Transmultiplexer System for Communication

A. Vishwakarma, A. Kumar, and Heung-No Lee

1 Introduction

The multirate systems are extensively used in numerous applications, which consist of two different structures: filter bank (FB) and transmultiplexer (TMUX) structures [1, 2]. Depending upon on required applications, multirate systems can be either used in analysis /synthesis mode that corresponds to a filter bank (FB) structure or in synthesis /analysis mode, which corresponds to a transmultiplexer system, shown in Fig. 1. Filter bank systems are used in numerous signal processing applications such as graphic equalizers, signal compression, and as vocoders [1–4], whereas TMUX systems are employed as the basic building blocks in many wireless communication techniques such as time domain multiple access (TDMA), frequency domain multiple access (FDMA), code domain multiple access (CDMA), and space domain multiple access [1, 2].These systems are basically exploited to convert time division multiplexed signals to frequency division multiplexed signals at the synthesis section and then back to time division multiplexed signals at the analysis section [1, 2]. TMUX systems can be derived from a filter bank structure, just by exchanging the role of analysis and synthesis filters. These structures can be further categorized into uniform TMUX and nonuniform TMUX. Uninform TMUX systems are used to transmit the signals having same bandwidth, while nonuniform TMUX are exploited to transmit the composite signals such as video signals and text signals having different sampling rates [1, 2]. Transmission of a video signal

---

A. Vishwakarma • A. Kumar (✉)
PDPM – Indian Institute of Information Technology Design and Manufacturing, Jabalpur, MP 482005, India
e-mail: anilkdee@gmail.com

H.-N. Lee
School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, Korea

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9_8

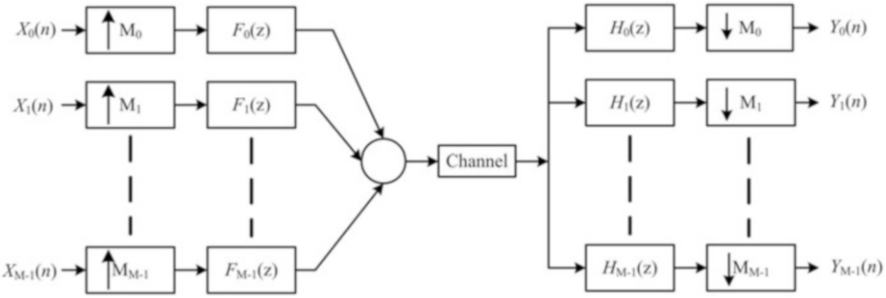


Fig. 1 A generalized block diagram of a transmultiplexer system [1, 2]

needs much larger bandwidth as compared to transmission of text messages, and most of the secure applications such as military need much larger bandwidth compared to other less important applications [5, 6]. Therefore, during the past, several efficient attempts have made to improve design of a TMUX system [5–9].

In TMUX system, perfect reconstruction (PR) is not possible due to structural incompatibility obtained from using samplers and synthesis/analysis filter bank [1, 2]. Due to structural incompatibility, three types of distortions such as *ICI* (aliasing distortion), *ISI*, and phase distortion are introduced in a TMUX system [1, 2]. In all multirate systems, phase distortion is eliminated by using a linear-phase finite impulse response (FIR) filter [1, 2]. In case of *PR* condition, *ICI* and *ISI* are negligible, but in case of nearly perfect reconstruction (*NPR*) condition, *ICI* and *ISI* are not negligible [1, 2]. Therefore, several methods for designing *PR* and *NPR* transmultiplexers have been proposed in literature [10–18]. Authors in [10] have proposed a new method to design nonuniform TMUX systems, based on general dual-rate structures to obtain PR property by providing extra design liberty. In [11], authors have used a quadratic semi-infinite programming (QSP) technique for designing of nonuniform transmultiplexer, which reduces the amplitude and aliasing distortions by reducing the sum of ripple energy of a single filter of synthesis/analysis section. In [12], authors have used a nonuniform transmultiplexer structure for flexible frequency-band reallocation (FFBR) networks. In this method, a nonuniform PR TMUX system consists of finite impulse response filters as well as a farrow configuration, which is used for up-sampler and down-sampler operation [13, 14]. Improvement in the design of a nonuniform TMUX system was carried out using artificial bee colony (ABC), genetic algorithm (GA) and particle swarm optimization (PSO) in [15–17]. Another method for the design of a nonuniform TMUX system with canonical signed digit (CSD) filter coefficients has been presented in [18]. Due to CSD representation, hardware complexity is decreased, which improves computational complexity, and performance of TMUX systems for communication application. Based on the reviewed literature on filter banks or transmultiplexer systems, for efficient design of a TMUX system, firstly a prototype filter is designed with the help of any optimization technique to minimize an objective function such as *ICI* or sum of *ICI* and *ISI*.

Several designs [19–28] were presented for efficient design of filter banks and TMUX systems based on linear search optimization proposed by Creusere and Mitra [29]. In these techniques, either passband edge frequency or 3 dB cutoff frequency is optimized using a linear search optimization. Recently, authors have used windowing technique for the design of cosine modulated uniform TMUX system [30–33].

In above context, this chapter presents an efficient method for the design of *NPR* nonuniform TMUX system using different window functions that have high side-lobe falloff rate (SLFOR), which reduce the interferences. The rest of the chapter is organized as follows: a brief introduction on filter banks and transmultiplexer systems is given in Sect. 1. In Sect. 2, an overview of a TMUX system is given, while Sect. 3 presents an improved design method for non-uniform TMUX system using different adjustable window functions such as Kaiser, Saramaki, Dolph–Chebyshev, and Transitional windows. Section 4 discusses different design examples using the devised methodology. Finally, concluding remarks are incorporated in Sect. 5.

2 Overview of Transmultiplexer System

During the past, several researchers have developed theory of multirate filter banks or transmultiplexer systems, available in [1]. Consider a generalized structure of a multichannel nonuniform tree structure TMUX system, given in Fig. 2a [1]. The decimation/interpolation factors ($M_a = M_0, M_1, \ldots, M_{M-1}$) for each band are chosen to satisfy Eq. (1) for perfect reconstruction of the input signals at synthesis section to the output signals of analysis section [1, 2]:

$$\sum_{a=0}^{M-1} \frac{1}{M_a} = 1, \tag{1}$$

where M stands for number of channels in a TMUX system. The parallel structure of this system is graphically illustrated in Fig. 2b.

In any tree-structured multirate system, a two-channel quadrature mirror filter (QMF) bank is used as the basic building blocks. At the synthesis side, input signals are interpolated and then filtered by the synthesis filters. After that, signals are united and transmitted through a common communication channel [1]. At the analysis side, united signal is filtered and decomposed by the decimation process. 2^{C-1} numbers of two-channel QMF banks are required in synthesis section or in analysis section, where C is the total number of stage of decomposition or combination. The corresponding parallel form of a nonuniform TMUX system is shown in Fig. 2b; the following relations are used [1, 2]:

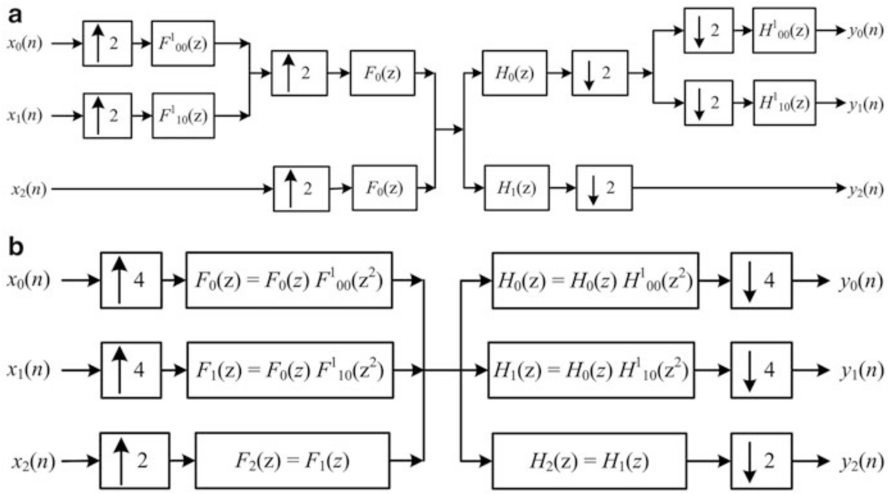


Fig. 2 (**a**) A block diagram of a tree-structured nonuniform transmultiplexer [1] (**b**) equivalent parallel structure of (**a**) [1]

$$H_0(z) = H_L(z)H_L\left(z^2\right), F_0(z) = F_L(z)F_L\left(z^2\right), \tag{2}$$

$$H_1(z) = H_L(z)H_H\left(z^2\right), F_1(z) = F_L(z)F_H\left(z^2\right), \tag{3}$$

and

$$H_2(z) = H_H(z), F_2(z) = F_H(z). \tag{4}$$

In general, from Fig. 1, the reconstructed output signal at i^{th} channel due to an input at the l^{th} channel is given as [1, 2]:

$$Y_i\left(z^{M_\text{a}}\right) = \frac{1}{M_\text{a}} \sum_{k=0}^{M_a-1} H_i\left(zW_{M_a}{}^{k}\right) . \sum_{l=0}^{M-1} F_l\left(zW_{M_a}{}^{k}\right) . X_l\left(z^{M_\text{a}}\right), \tag{5}$$

where $W_{M_\text{a}} = e^{-j2\pi/M_\text{a}}$ and $k = 0, .., M_\text{a} - 1$. Transfer function, which amid output signal $Y_i(z^{M_\text{a}})$ and input signal $X_r(z^{M_\text{a}})$, is defined as [1, 2, 29–33]

$$T_{i,r}\left(z^{M_\text{a}}\right) = \sum_{k=0}^{M_\text{a}-1} H_i\left(zW_{M_\text{a}}{}^{k}\right) . F_r\left(zW_{M_\text{a}}{}^{k}\right). \tag{6}$$

In Eq. (6), $T_{i,r}(z^M)$, $i \neq r$ stands for the cross talk between different nonuniform TMUX system channels. In case of crosstalk-free nonuniform TMUX system, Eq. (8) is zero. *ICI* is a measure of signal leak from one sub-channel to adjacent sub-channel, mathematically expressed as [1, 29–33]

$$E_{\mathrm{ICI}} = \max_{\omega,\, i}\left(\sum_{r=0,\, i\neq r}^{M-1} \left|T_{i,r}\left(e^{j\omega}\right)\right|^2\right). \tag{7}$$

ISI distortion in TMUX system occurs due to non-ideal magnitude response of synthesis/analysis filters in their passbands [31–33] and defined as

$$E_{\mathrm{ISI}} = \frac{1}{\pi}\int_0^{\pi}\left(\left|T_{ll}\left(e^{j\omega}\right) - 1\right|^2\right)d\omega. \tag{8}$$

Signal to interchannel interference ratio (*SICI*) [33]:

$$\mathrm{SICI} = \frac{E_k}{E_{\mathrm{ICI}}}, \tag{9}$$

where

$$E_k = \frac{1}{\pi}\int_0^{\pi}\left(\left|T_{kk}\left(e^{j\omega}\right)\right|^2\right)d\omega. \tag{10}$$

Signal to intersymbol interference ratio (SISI) [33]:

$$\mathrm{SISI} = \frac{E_k}{E_{\mathrm{ICI}}}. \tag{11}$$

Signal to total interference ratio in the k^{th} channel (SI_k) [33]:

$$\mathrm{SI}_k = \frac{E_k}{E_{\mathrm{ICI}(k)} + E_{\mathrm{ISI}(k)}} = \frac{1}{(\mathrm{SICI})^{-1} + (\mathrm{SISI})^{-1}}. \tag{12}$$

A detailed discussion on filter banks and TMUX systems is found in [1, 2] and the references therein.

3 Improved Design Method for Nonuniform TMUX System

In this chapter, different window functions such as Kaiser, Saramaki, Dolph–Chebyshev, and Transitional windows [1, 22–25] are employed for the design of a prototype filter for nonuniform TMUX system due to closed form solution and low complexity. These window functions are defined as follow [22–25, 34, 35].

3.1 Kaiser Window

In discrete time (DT) domain, Kaiser window is defined as [34, 35]

$$w_k(n) = \begin{cases} \dfrac{I_o\left(\alpha_k\sqrt{1-\left(\frac{2n}{N-1}\right)^2}\right)}{I_o(\alpha_k)} & \text{for } \mid n \mid \leq \dfrac{N-1}{2}, \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

where α_k is the adjustable parameter. $I_0(x)$ is the modified Bessel function of the first kind of zero order, given as

$$I_0(x) = 1 + \sum_{k=1}^{\infty}\left[\frac{1}{k!}\left(\frac{x}{2}\right)^k\right]^2. \tag{14}$$

In Eq. (13), the parameter α_k can be computed as

$$\alpha_k = \begin{cases} 0.1102(A_s - 8.7); & \text{for}, A_s > 50 \\ 0.5842(A_s - 21)0.4 + 0.07886(A_s - 21); & \text{for}, 21 \leq A_s \leq 50 \\ 0; & \text{for}, A_s < 21 \end{cases} \tag{15}$$

Order (N-1) of a filter is calculated as [34, 35].

$$N - 1 = \frac{(A_s - 7.95)}{14.95\Delta f}, \tag{16}$$

where A_s is the stopband attenuation and $\Delta f = (\omega_s - \omega_p)/2$.

3.2 Saramaki Window [35]

A Saramaki window is given as

$$w(n) = v_o(n) + 2\sum\nolimits_{k=1}^{N} v_k(n), 0 \leq n \leq N-1, \tag{17}$$

where

$$v_o(n) = \begin{cases} 1, & n = 0 \\ 0; & \text{otherwise} \end{cases}, \tag{18}$$

$$v_1(n) = \begin{cases} \gamma - 1, & n = 0 \\ \gamma/2, & |n| = 10, \qquad \text{otherwise} \end{cases}", \tag{19}$$

and

$$v_k(n) = \begin{cases} 2(\gamma - 1)v_{k-1}[n] - v_{k-2}[n] + \gamma[v_{k-1}[n-1] + v_{k-1}[n+1]], & -k \le n \le k \\ 0, & \text{otherwise} \end{cases}. \tag{20}$$

In Eqs. (24) and (25), γ is defined as

$$\gamma = \frac{1 + \cos\frac{2\pi}{2N+1}}{1 + \cos\frac{2\beta\pi}{2N+1}}, \tag{21}$$

where β is the adjustable parameter, computed as

$$\beta = \begin{cases} 0.000121(A_{\mathrm{s}} - 21)^2 + 0.0224(A_{\mathrm{s}} - 21) + 1; & \text{for } 21 \le A_{\mathrm{s}} \le 65 \\ 0.033A_{\mathrm{s}} + 0.062; & \text{for } 65 < A_{\mathrm{s}} \le 110 \\ 0.0342A_{\mathrm{s}} - 0.064; & \text{for } A_{\mathrm{s}} > 110 \end{cases}, \tag{22}$$

Order (N-1) of a filter is obtained using

$$N - 1 = \frac{A_{\mathrm{s}} - 8.15}{14.36(\omega_{\mathrm{s}} - \omega_{\mathrm{p}})/\pi}. \tag{23}$$

3.3 Dolph–Chebyshev Window [34, 35]

A Dolph–Chebyshev window function $w(n)$ is defined as

$$w(n) = \frac{1}{N}\left[\frac{1}{r} + 2\sum_{i=1}^{(N-1)/2} T_{N-1}\left(y_o \cos\frac{i\pi}{N}\right)\cos\frac{2ni\pi}{N}\right], \tag{24}$$

where

$$y_o = \cosh\left(\frac{1}{N-1}\cosh^{-1}\frac{1}{r}\right). \tag{25}$$

and r is the required ripple ratio, and

$$T_k(y) = \begin{cases} \cos\left(k\cos^{-1}y\right) & |\, y \,| \le 1 \\ \cosh\left(k\cosh^{-1}y\right) & |\, y \,| > 1 \end{cases}. \tag{26}$$

Order (N-1) of a filter is obtained using

$$N - 1 = \frac{A_{\mathrm{s}} - 8.15}{14.36(\omega_{\mathrm{s}} - \omega_{\mathrm{p}})/\pi}. \tag{27}$$

3.4 Transitional Window [35]

A Transitional window function is defined as

$$w(n) = \begin{cases} \left(\dfrac{\sin\left[2\pi n/(N+1)\right]}{2\pi n/(N+1)}\right)^{\eta}; & \text{for } 0 \le n \le N-1 \\ 0; & \text{otherwise} \end{cases}, \tag{28}$$

where η is the shape factor, given as

$$\eta = \begin{cases} 0; & \text{for}, A_{\mathrm{s}} < 21 \\ \dfrac{13}{1+(126/A_{\mathrm{s}})^{1.6}} - 0.7; & \text{for}, 21 \le A_{\mathrm{s}} < 120 \\ 0.5/\left(1+\left[A_{\mathrm{s}} - 120/20\right]^{1.6}\right) + 0.063(A_{\mathrm{s}} - 120) + 5.06; & \text{or}, A_{\mathrm{s}} \ge 120 \end{cases}. \tag{29}$$

Order $(N-1)$ of a filter is determined using

$$N - 1 = \begin{cases} \dfrac{\pi}{\Delta\omega}\left(\dfrac{24.3}{1+(149/A_{\mathrm{s}})1.6} - 0.085\right) - 1; & \text{for}, 21 \le \mathrm{A_s} < 120\ \mathrm{dB} \\ \dfrac{\pi}{\Delta\omega}\left(-0.00075(200.3 - A_{\mathrm{s}})^2 + 14.74\right) - 1; & \text{for}, 120 \le \mathrm{A_s} < 150\ \mathrm{dB} \\ \dfrac{\pi}{\Delta\omega}\left(0.00001087(A_{\mathrm{s}} + 245.6)^2 - 3.1\right) - 1; & \text{for}, \mathrm{A_s} \ge 150\ \mathrm{dB} \end{cases} \tag{30}$$

A detailed discussion on these window functions and their properties can be found in [34, 35] and the references therein.

From the analysis of a nonuniform tree-structured filter bank, and TMUX system [2, 34, 35], *PR* is achieved by satisfying [2, 34, 36]

$$\sum_{k=0}^{M-1} \left|F_k\left(e^{j\omega}\right)\right|^2 = 1, \ \ \text{for } 0 < \omega < \frac{\pi}{M}, \tag{31}$$

where $F_k(e^{j\omega})$ is the k^{th} synthesis filter in a TMUX system. In case of three-channel nonuniform TMUX, Eq. (31) is reduced to

$$\left|F_0\left(e^{j\omega}\right)\right|^2 + \left|F_1\left(e^{j\omega}\right)\right|^2 + \left|F_2\left(e^{j\omega}\right)\right|^2 = 1, \text{for } 0 \le \omega \le \frac{\pi}{3}. \tag{32}$$

Using relations given in Eqs. (1) and (2), it is further simplified as [36]

$$\left|F_L(z)F_L\left(z^2\right)\right|^2 + \left|F_L(z)F_H\left(z^2\right)\right|^2 + |F_H(z)|^2 = 1. \tag{33}$$

Since up-sampled and original signals have same energy content, then above equation, using $H_H(z) = H_L(-z)$, can be modified as

$$|F_L(z)|^4 + \mid F_L(z)|^2 F_L(-z)\big|^2 + |F_L(-z)|^2 = 1, \tag{34}$$

and if it is computed at $\omega = 0.5\pi$ in frequency domain [36],

$$2\left|F_L\left(e^{j\pi/2}\right)\right|^4 + \left|F_L\left(e^{j\pi/2}\right)\right|^2 = 1 \tag{35}$$

Let $F_L(e^{j\pi/2}) = h$, then Eq. (16) reduced to

$$2h^4 + h^2 - 1 = 0. \tag{36}$$

The roots of the above polynomial are $h = \mp 0.7071$ and $h = \mp i$. Imaginary and negative roots are not allowed for the filter design due to implementation [36], so

$$h = \mid F_L\left(e^{j\pi/2}\right) \mid = 0.7071. \tag{37}$$

In case of a four-channel nonuniform TMUX system or FB [34, 36], PR condition is resulted if

$$\left|F_0\left(e^{j\omega}\right)\right|^2 + \left|F_1\left(e^{j\omega}\right)\right|^2 + \left|F_2\left(e^{j\omega}\right)\right|^2 + \left|F_3\left(e^{j\omega}\right)\right|^2 = 1, \text{ for } 0 \le \omega \le \frac{\pi}{4}, \tag{38}$$

where synthesis filters, mathematically, are described as [34, 36]

$$F_0(z) = F_L(z)F_L\left(z^2\right)F_L\left(z^4\right), \tag{39}$$

$$F_1(\mathrm{z}) = F_{\mathrm{L}}(\mathrm{z})F_{\mathrm{L}}\left(\mathrm{z}^2\right)F_{\mathrm{H}}\left(\mathrm{z}^4\right), \tag{40}$$

and

$$F_2(\mathrm{z}) = F_{\mathrm{L}}(\mathrm{z})F_{\mathrm{H}}\left(\mathrm{z}^2\right), F_3(\mathrm{z}) = F_{\mathrm{H}}(\mathrm{z}). \tag{41}$$

Using above relations given in Eqs. (39) and (40), Eq. (38) is modified as

$$\begin{aligned}&\left|F_L(z)F_L\left(z^2\right)F_L\left(z^4\right)\right|^2 + \left|F_L(z)F_L\left(z^2\right)F_H\left(z^4\right)\right|^2 + \mid F_L(z)F_H\left(z^2\right)\\&\quad \mid + |H_H(z)|^2 = 1.\end{aligned} \tag{42}$$

Similar to 3-channel TMUX system:

$$\left|F_L\left(z^4\right)\right|^2 = \left|F_H\left(z^2\right)\right|^2 =\mid H_L(-z) \mid \tag{43}$$

and

$$F_H\left(z^4\right)\right|^2 = \left|F_H\left(z^2\right)\right|^2 = |H_H(z)|^2. \tag{44}$$

Using the above equations, Eq. (42) can be written as [36]

$$|F_L(z)|^6 + |F_L(z)|^4 + |F_L(-z)|^2 + |F_L(z)|^2|F_L(-z)|^2 + |F_L(-z)|^2 = 1, \tag{45}$$

and in frequency domain, if computed at $\omega = 0.5\pi$,

$$2\left|F_L\left(e^{j\pi/2}\right)\right|^6 + \left|F_L\left(e^{j\pi/2}\right)\right|^4 + \left|F_L\left(e^{j\pi/2}\right)\right|^2 = 1, \tag{46}$$

and in polynomial form, it is

$$2h^6 + h^4 + h^2 - 1 = 0. \tag{47}$$

The roots of above equation are found using a simple factorization approach, and these are x = ± 0.7071, 0.5 ± 0.866i, and 0.5 ± 0.866i. For filter design, only $h = |F_L(e^{j\pi/2})| = 0.7071$ is considered due to implementation. Similarly, this analysis has been also repeated for different channel nonuniform TMUX systems; in each case, h is reduced to 0.707, which is a perfect reconstruction condition or magnitude response of a prototype filter at quadrature frequency.

In this chapter, filter coefficients of a prototype filter for a nonuniform TMUX are optimized to 0.7071 by iteratively varying 3 dB cutoff frequency (ω_c). A flowchart of the proposed algorithm for designing a prototype filter for nonuniform *FB* is shown in Fig. 3. The steps to be followed in this case are summarized below [22–25, 36]:

Step 1: Specify design specifications of a prototype filter such as A_s, A_p, ω_s, and ω_p.

Step 2: Initialize different parameters such as counter, step size, error limit, *target* $=(|H_o(e^{j\pi/2})| = 0.707)$.

Step 3: Compute filter coefficients using different window functions discussed in this chapter using order of filter (N), A_s, and ω_c. Also compute magnitude response of a designed (*MRD*) filter at $\omega = 0.5\pi$ and also *deviation or error* from target limit by

$$\text{Error} = \text{Target} - \text{MRD} \tag{48}$$

Step 4: If $/Error/ \leq$ *error limit* satisfies, then stop algorithm. Otherwise move to the next step.

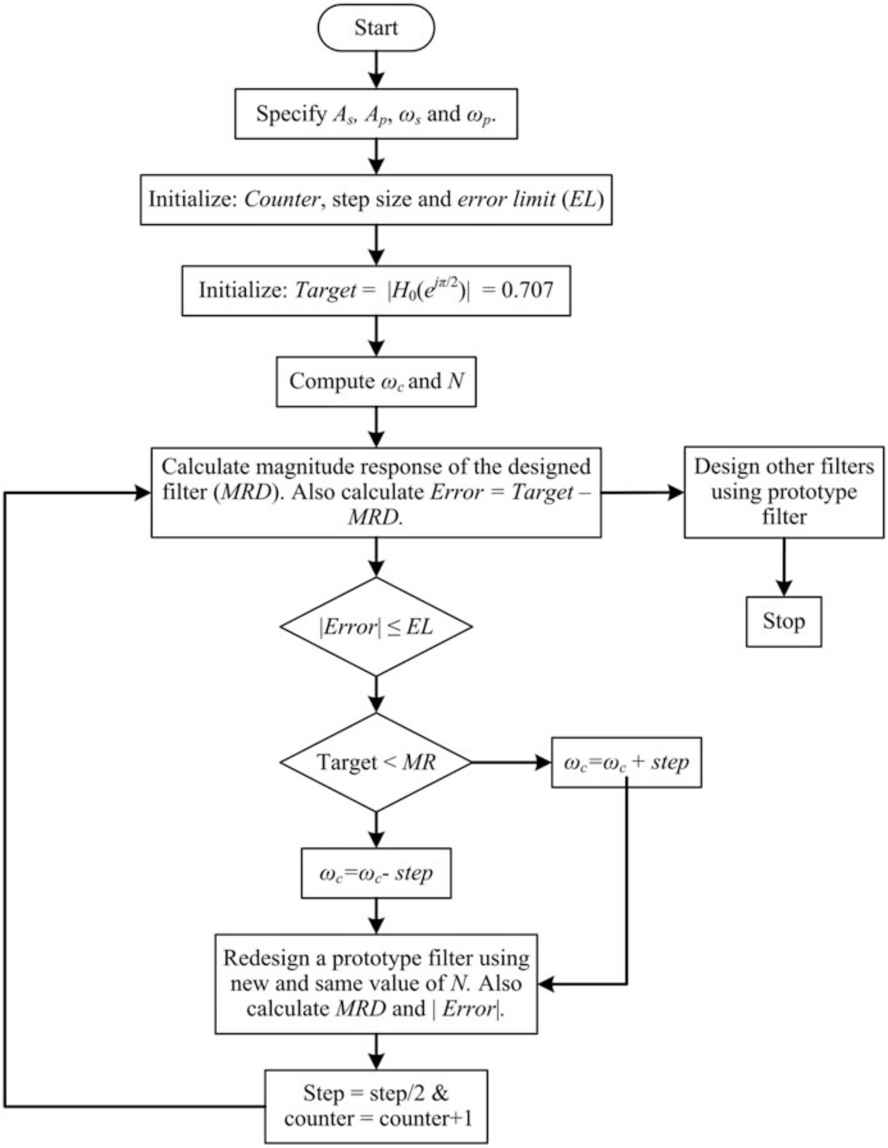


Fig. 3 A flowchart of the proposed optimization algorithm [22–25, 36]

Step 5: If $MRD < Target$, then adjust cut-off frequency by $\omega_c = \omega_c + step$, else $\omega_c = \omega_c - step$.

Step 6: Redesign the filter with new ω_c and with same *N*, then calculate *MRD* and |*Error*|, at preceding halved step size. Also, increase counter value by one.

Step 7: Go to step 4 till error limit is not satisfied.

4 Design Examples

In this section, the proposed method is exploited for the design of *NPR* nonuniform TMUX systems with different design specifications.

Design Example I In these examples, a three-channel nonuniform TMUX system has been designed with decimation factor 4, 4, and 2, with different filter taps using different window functions such as Kaiser, Saramaki, Dolph–Chebyshev and Transitional windows. Figure 4 summarizes the simulation results obtained with the design specifications of $A_s = 110\,dB, A_p = 0.00024$ dB, $\omega_s = 0.45\pi$, and $\omega_p = 0.38\pi$. Table 1 lists the fidelity parameters obtained in each design of nonuniform TMUX system. It is evident from Table 1 that Transitional window gives best performance in terms of lower *ICI*, *ISI*, and higher *SISI*, *SICI*, and *SI*. As shown in Fig. 5, *ICI* decreases with the increase in filter length, but high filter length also increases the implementation and design complexity. Variation of *ICI*, in case of Kaiser window, is different as compared to other three adjustable windows.

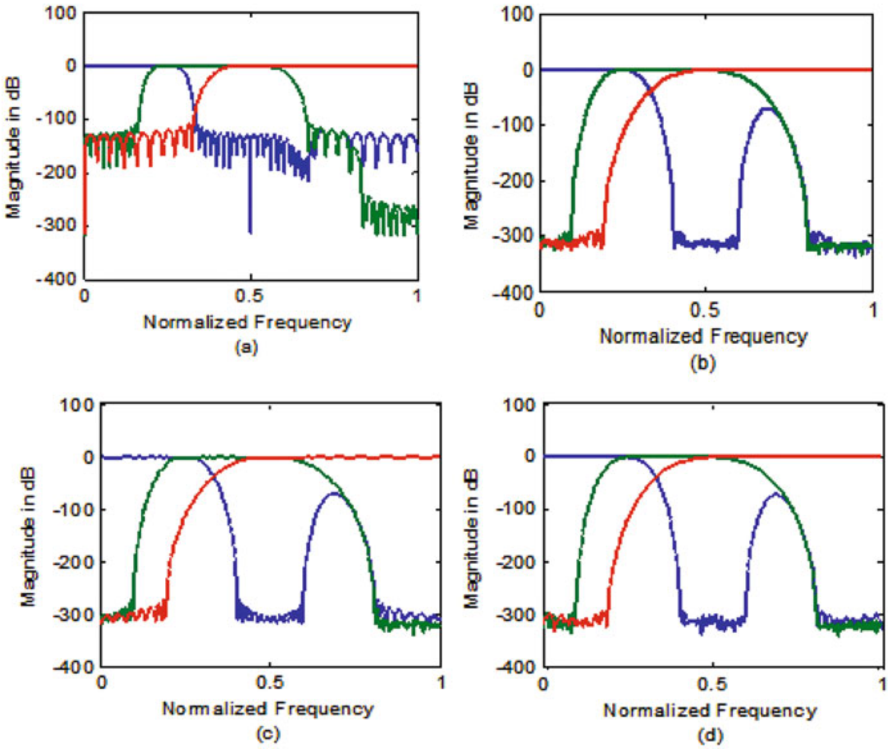


Fig. 4 Magnitude response of synthesis filters of three-channel nonuniform TMUX, (**a**) Kaiser, (**b**), Saramaki, (**c**) Dolph–Chebyshev, and (**d**) Transitional windows

Table 1 Performance of proposed method for three-channel nonuniform TMUX for different filter taps

Band (*M*)	Window	*N*	*ICI*(dB)	*SI*(dB)	*SICI*(dB)	*SISI*(dB)	*ISI*(dB)
Three band (4, 4, 2)	Kaiser	48	−232.00	−64.77	116.12	−51.11	−51.11
		64	−245.45	−65.12	128.71	−51.61	−51.61
		128	−259.70	−65.08	146.51	−47.11	−47.11
		256	−253.99	−65.05	145.07	−42.86	−42.86
		384	−259.14	−65.01	151.67	−41.45	−41.45
	Saramaki	48	−188.99	−63.91	56.32	−68.76	−68.76
		64	−215.76	−64.38	94.04	−56.34	−56.34
		128	−359.23	−65.20	241.63	−53.38	−53.38
		256	−455.15	−65.49	341.98	−46.68	−46.68
		384	−431.14	−65.01	321.61	−43.51	−43.51
	Dolph–Chebyshev	48	−187.97	−63.87	53.52	−70.56	−70.56
		64	−213.58	−64.35	92.17	−56.05	−56.05
		128	−351.03	−65.19	233.59	−52.23	−52.23
		256	−448.21	−65.48	336.02	−46.69	−46.69
		384	−486.42	−65.01	377.94	−43.47	−43.47
	Transitional	48	−189.46	−63.93	57.68	−68.84	−68.84
		64	−216.43	−64.39	96.57	−56.46	−56.46
		128	−362.62	−65.08	244.48	−53.05	−53.05
		256	−432.63	−65.50	320.45	−46.67	−46.67
		384	−436.11	−65.01	326.54	−43.53	-43.53

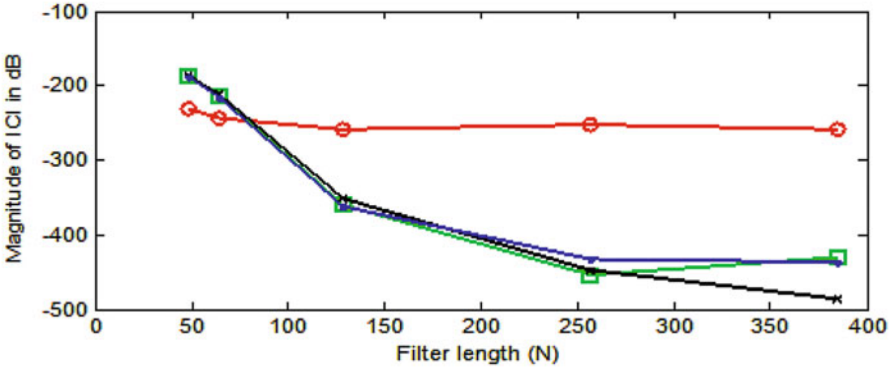


Fig. 5 Variation of *ICI* in three-channel nonuniform TMUX with different filter taps (*Red* Kaiser, *green* Saramaki, *black* DC, *Blue* Transitional)

Design Example II Four-channel with decimation factors (8, 8, 4, and 2) nonuniform TMUX system is designed using different design specifications. Design results obtained with $A_{\mathrm{s}} = 120\ dB,\ A_{\mathrm{p}} = 0.00034$ dB $\omega_{\mathrm{s}} = 0.65\pi$, and $\omega_{\mathrm{p}} = 0.51\pi$ are depicted in Figs. 6 and 7 and Table 2. From the design results, it is

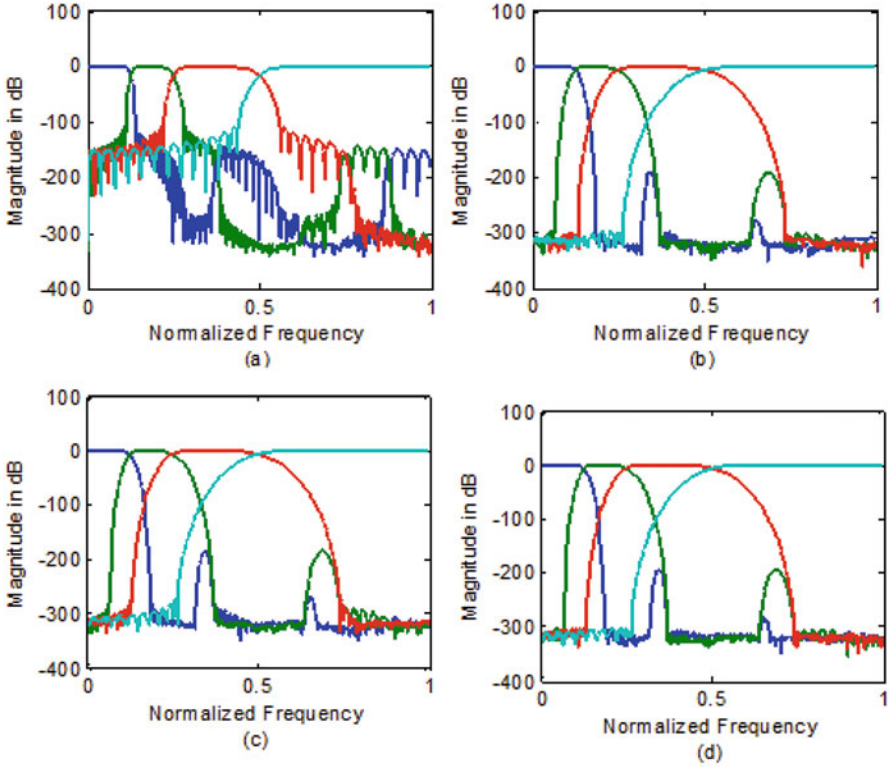


Fig. 6 Magnitude response of synthesis filters of four-channel nonuniform TMUX. (**a**) Kaiser, (**b**) Saramaki, (**c**) Dolph–Chebyshev, and (**d**) Transitional windows

evidenced that Transitional window gives the best performance in terms of low *ICI* and *ISI* and higher values of *SICI*, *SISI*, and *SI*.

From Fig. 7, it is reflected that the performance of *ICI* for four-channel nonuniform TMUX is nearly same as for three-channel nonuniform TMUX. Table 3 shows a comparison of performance parameters with the earlier reported work [30–32, 37, 38], and it was found that the proposed method gives better performance compared to earlier published work [30–32, 37, 38] in terms of *ICI*, *SICI*, *SISI*, and *SI*. For filter lengths 48, 64, and 128, *ICI* is the minimum for Transitional window, which gives the overall best performance. Signal to overall interference and *ISI* is approximately same for the given filter lengths.

In Table 4, a comparison has been carried out between four-channel uniform [30, 31] and four-channel nonuniform transmultiplexer on the basis of interferences and signal to interference ratios. It was found that nonuniform transmultiplexer systems have significantly lower values of *ICI* and *ISI* and higher values of *SICI*, *SISI*, and *SI*.

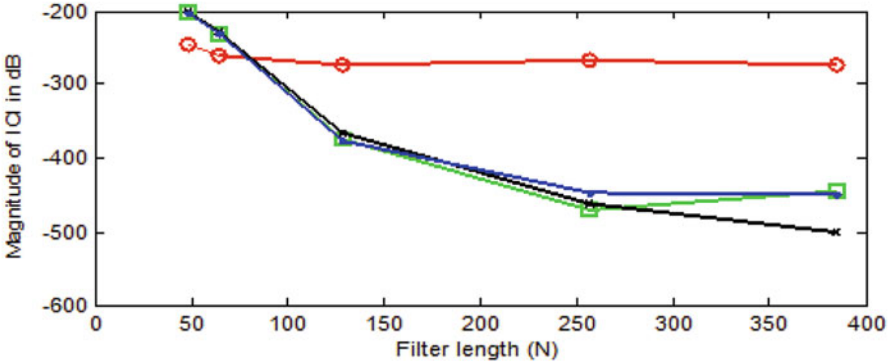


Fig. 7 Variation of *ICI* in four-band nonuniform transmultiplexer with different filter lengths (*Red* Kaiser, *green* Saramaki, *black* DC, *blue* Transitional)

Table 2 Performance of proposed method four-channel nonuniform TMUX with different filter taps

Band (*M*)	Window	*N*	*ICI*(dB)	*ISI*(dB)	*SICI*(dB)	*SISI*(dB)	*SI*(dB)
Four band (8, 8, 4, 2)	Kaiser	48	−246.94	−79.59	116.79	−51.67	−51.67
		64	−260.85	−80.84	128.87	−51.13	−51.13
		128	−274.79	−80.20	146.10	−47.47	−47.47
		256	−268.15	−80.39	145.32	−42.43	−42.43
		384	−274.88	−80.45	151.62	−41.80	−41.80
	Saramaki	48	−203.48	−78.85	56.92	−68.70	−68.70
		64	−230.12	−79.27	94.09	−56.75	−56.75
		128	−374.02	−80.91	241.71	−53.38	−53.38
		256	−470.64	−80.24	342.74	−46.65	−46.65
		384	−445.10	−80.35	321.51	−43.22	−43.22
	Dolph–Chebyshev	48	−202.43	−78.81	53.78	−70.82	−70.82
		64	−228.26	−79.24	92.23	−56.78	−56.78
		128	−366.32	−80.90	233.07	−52.34	−52.34
		256	−463.17	−80.23	336.31	−46.62	−46.62
		384	−501.08	−80.35	377.53	−43.19	−43.19
	Transitional	48	−204.5	−78.86	57.07	−68.66	−68.66
		64	−231.99	−79.28	96.96	−56.74	−56.74
		128	−377.3	−80.08	244.97	−53.40	−53.40
		256	−447.68	−80.24	320.75	−46.67	−46.67
		384	−450.61	−80.35	326.01	−43.23	-43.23

Table 3 Comparison the proposed method over earlier reported

Type of algorithm	*N*	*ICI* (dB)	*ISI* (dB)	*SICI* (dB)	*SISI* (dB)	*SI* (dB)
Algorithm in [37]	48	−103.50	−77.52	–	–	–
Algorithm in [38]	128	−126.23	−57.88	–	–	–
Algorithm in [30]	64	−92.22	−44.03	–	–	–
Algorithm in [31]	48	−49.81	−46.06	–	–	–
Algorithm in [32]	48	−157.65	−72.24	−29.07	−114.48	−114.49
Algorithm in [32]	64	−164.06	−72.24	−37.41	−129.22	−129.23
Proposed (transitional)	48	−189.46	−63.93	57.68	−68.84	−68.84
Proposed (transitional)	64	−216.58	−64.35	96.17	−63.05	−63.05
Proposed (transitional)	128	−362.03	−65.19	244.59	−56.23	-56.23

Table 4 Comparison of four-channel uniform [30, 31] and nonuniform TMUX

Proposed method	Window	*N*	*ICI (dB)*	*ISI (dB)*
Martin et al. 2004	Modified Blackman	32	−57.55	−58.42
Martin et al. 2003	Kaiser	64	−108.15	−56.22
	Blackman	98	−110.03	−67.25
	Blackman	128	−119.18	−68.46
Proposed	Transitional	32	−176.26	−77.67
Proposed	Transitional	64	−216.58	−79.29
Proposed	Transitional	98	−300.38	−79.72
Proposed	Transitional	128	−362.0	−79.92

5 Conclusion

In this chapter, an improved prototype filter for nonuniform TMUX has been designed using windowing technique. The simulation results illustrate that ICI is significantly reduced as compared to earlier published results. Signal to overall interference ratio is also satisfactory in case of nonuniform transmultiplexer system as compared to uniform transmultiplexer systems.

References

1. Vaidyanathan, P. P. (1993). *Multirate systems and filter banks*. Englewood Cliffs, NJ: Prentice- Hall.
2. Vishwakarma, A., Kumar, A., & Singh, G. K. (2016). Transmultiplexer filter bank systems: a research overview. *International Journal of Signal and Imaging Systems Engineering, 9*(3), 146–154.
3. Kumar, A., Anurag, S., & Singh, G. K. (2015). An optimized cosine-modulated nonuniform filter bank design for subband coding of ECG signal. *Journal of King Saud University - Engineering Sciences, 27*(2), 158–169.

4. Kumar, A., Singh, G. K., & Anand, R. S. (2011). A closed form design method for two-channel quadrature mirror filter banks. *Signal Image and Video Processing, 5*(1), 121–131.
5. Eghbali, A., Johansson, H., & Löwenborg P. (2011). Reconfigurable nonuniform transmultiplexers using uniform modulated filter banks. *IEEE Transactions on Circuits and Systems-I: Regular Paper, 58*(3), 539–547.
6. Liang, L., Shi, G., & Xie, X. (2011). Non-uniform directional filter banks with arbitrary frequency partitioning. *IEEE Transactions on Image Processing, 20*(1), 283–289.
7. Xie, X. M., Chan, S. C., & Yuk, T. I. (2006). Design of linear-phase recombination nonuniform filter banks. *IEEE Transactions on Signal Processing, 54*(7), 2806–2814.
8. Soni, R., Jain, A., & Saxena, R. (2013). An optimized design of nonuniform filter bank using variable-combinational window function. *AEU - International Journal of Electronics and Communications, 67*(7), 595–601.
9. Xie, X. (2007). Design of nonuniform cosine-modulated filter-banks with the perfect-reconstruction property and arbitrary filter length. *Progress in Natural Science, 17*(3), 340–345.
10. Liu, T., & Chen, T. (2001). Design of multi-channel nonuniform transmultiplexers using general building blocks. *IEEE Transactions on Signal Processing, 49*(1), 91–99.
11. Ho C. Y. F., Ling B. W. K., Liu Y.,Tarn P. K. S., & Teo K. L. (2003). Optimum nonuniform transmultiplexer design. *Proceedings of International Conference on Neural Networks and Signal Processing, ICNNSP'031*, Nanjing, pp. 740–743.
12. Eghbali A., Johansson H., & Löwenborg P. (2008). An arbitrary-bandwidth transmultiplexer and its application to flexible frequency-band reallocation networks, *European Conference on Circuit Theory and Design*, Seville, pp. 248–251.
13. Eghbali, A., Johansson, H., & Lowenborg, P. L. (2008). A multimode transmultiplexer structure. *IEEE Transactions on Circuits and Systems II: Express Briefs, 55*(3), 279–283.
14. Eghbali A., Johansson H., & Löwenborg P. (2008). A Farrow structure based multimode transmultiplexer. *Proceedings IEEE International Symposium on Circuits and Systems*, Seattle, pp. 3114–3117.
15. Manoj, V. J., & Elias, E. (2012). Artificial bee colony algorithm for the design of multiplierless nonuniform filter bank transmultiplexer. *Information Sciences, 192*, 193–203.
16. Manoj, V. J., & Elias, E. (2009). Design of multiplier-less nonuniform filter bank transmultiplexer using genetic algorithm. *Signal Processing, 89*(11), 2274–2285.
17. Manoj V. J., & Elias E. (2009). On the design of multiplier-less nonuniform filter bank transmultiplexer using particle swarm optimization. *World Congress on Nature and Biologically Inspired Computing, NABIC 2009 - Proceedings*, Coimbatore, pp. 55–60.
18. Manoj, V. J., & Elizabeth, E. (2009). Design of nonuniform filter bank transmultiplexer with canonical signed digit filter coefficients. *IET Signal Processing, 3*(3), 211–220.
19. Lin, Y., & Vaidyanathan, P. P. (1998). A Kaiser window approach for the design of the prototype filters of cosine modulated filter bank. *IEEE Signal Processing Letter, 5*(6), 132–134.
20. Cruz-Roldán, F., Lopez, P. A., Bascon, S. M., & Lawson, S. S. (2002). An efficient and simple method for designing prototype filters for cosine modulated pseudo QMF banks. *IEEE Signal Processing Letters, 9*(1), 29–31.
21. Zhao, Y., Luo, S., & Wan, J. (2013). Modified design algorithm for cosine-modulated subband filter banks based on iteration. *Journal of Data Acquisition and Processing, 28*(1), 110–116.
22. Soni, R. K., Jain, A., & Saxena, R. (2013). A design of IFIR prototype filter for cosine modulated filter bank and transmultiplexer. *AEU - International Journal of Electronics and Communications, 67*(2), 130–135.
23. Kumar, A., Singh, G. K., & Kuldeep, B. (2011). An improved and simplified approach for designing cosine modulated filter bank using window technique. *Journal of Mathematical Modelling and Algorithms, 10*(3), 213–226.
24. Kumar, A., & Kuldeep, B. (2012). Design of cosine modulated filter bank using improved exponential window. *Journal of Franklin Institute, 349*(3), 1304–1315.

25. Datar, A., Jain, A., & Sharma, P. C. (2013). Design and performance analysis of adjustable window functions based cosine modulated filter banks. *Digital Signal Processing, 23*(1), 412–417.
26. Kumar, A., Singh, G. K., & Anand, R. S. (2011). A simple design method for the cosine modulated filter banks using weighted least square technique. *Journal of the Franklin Institute, 348*(1), 606–621.
27. Berger, S. W. A., & Antoniou, A. (2007). An efficient closed form design method for cosine modulated filter banks using window function. *Signal Processing, 87*(5), 811–823.
28. Kumar, A., Singh, G. K., & Anand, R. S. (2011). An improved closed form design method for the cosine modulated filter banks using windowing technique. *Applied Soft Computing, 11*(3), 3209–3217.
29. Creusere, C. D., & Mitra, S. K. (1995). A simple method for designing high quality prototype filters for M-band pseudo QMF banks. *IEEE Transactions on Signal Processing, 43*(4), 1005–1007.
30. Martin, P., Cruz-Roldan, F., & Saramaki, T. (2003). A windowing approach for designing critically sampled nearly perfect-reconstruction cosine-modulated transmultiplexers and filter banks. *IEEE ISPA, 2*, 755–760.
31. Martin, P., Cruz-Roldan, F., & Saramaki T. (2004). A new window for the design of cosine-modulated multirate systems. *IEEE ISCAS, 3*, 529–532.
32. Soni, R. K., Jain, A., & Saxena, R. (2010). An improved and simplified design of pseudo-transmultiplexer using Blackman window family. *Digital Signal Processing, 20*(3), 743–749.
33. Vishwakarma, A., Kumar, A., & Singh, G. K. (2015). A prototype filter design for cosine modulated transmultiplexer using weighted constrained least squares technique. *AEU-International Journal of Electronics and Communications, 69*(6), 915–922.
34. Mitra, S. K. (2006). *Digital signal processing: A computer based approach*. New York: McGraw Hill.
35. Saramaki, T. (1991). Adjustable windows for the design of FIR filters–a tutorial. *6th Mediterranean Electro-technical Conference*, LJubljana, pp. 28–33.
36. Kumar, A., Singh, G. K., & Anurag, S. (2013). Design of nearly perfect reconstructed non-uniform filter bank by constrained Equiripple FIR technique. *Applied Soft Computing, 13*(1), 353–360.
37. Cruz-Roldán, F., Bravo-Santos, Á. M., Martín-Martín, P., & Jiménez-Martínez, R. (2003). Design of multi-channel near-perfect-reconstruction transmultiplexers using cosine-modulated filter banks. *Signal Processing, 83*(5), 1079–1091.
38. Singh, N., & Saxena, R. (2012). Novel window and its application in NPR type transmultiplexer design. *Electrical and Electronic Engineering, 2*(6), 342–350.

Index

G.J. Dolecek (ed.), *Advances in Multirate Systems*,
DOI 10.1007/978-3-319-59274-9