Circuits, Systems, and Signal Processing

A novel design method for nearly perfect reconstruction cosine modulated filter banks

--Manuscript Draft--

Manuscript Number:	CSSP-D-25-01690R5
Full Title:	A novel design method for nearly perfect reconstruction cosine modulated filter banks
Article Type:	Original Research
Keywords:	Cosine-modulated filter banks; Transmultiplexer; Software implementation; optimization
Corresponding Author:	Selena Vukotic, Ph. D. Union University, School of Computing Belgrade, SERBIA
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	Union University, School of Computing
Corresponding Author's Secondary Institution:	
First Author:	Selena Vukotic, Ph. D.
First Author Secondary Information:	
Order of Authors:	Selena Vukotic, Ph. D.
	Marko Mladenovic
	Djordje Babic
Order of Authors Secondary Information:	
Funding Information:	
Abstract:	This paper presents a novel two-phase design method for nearly perfect reconstruction cosine-modulated filter banks, aimed at reducing filter order while preserving signal fidelity. Traditional approaches increase the prototype filter order to minimize distortion, which in turn increases complexity due to more adders and multipliers. The proposed method overcomes this trade-off by first optimizing filter banks with a small number of channels, and then extending the design to a larger number of channels using a Farrow structure. The initial coefficients of the Farrow structure are derived from the prototype filter designed for the small number of channels, and then optimized for filter banks with a large number of channels. This structured approach s particularly suitable for offline application, also enabling an efficient scaling while maintaining low amplitude and aliasing distortion. Implemented in MATLAB with minimal user input, the method provides improvements over the existing techniques in terms of filter order and reconstruction quality.
Response to Reviewers:	No further comments are required.

A novel design method for nearly perfect reconstruction cosine modulated filter banks

Selena Vukotić[1*], Marko Mladenović[1] and Djordje Babić[1]

[1*]School of Computing, Union University, Kneza Mihaila 6/VI, Belgrade, 11000, Serbia.

*Corresponding author(s). E-mail(s): svukotic@raf.edu.rs;
Contributing authors: mmladenovic@raf.edu.rs; djbabic@raf.edu.rs;

Abstract

This paper presents a novel two-phase design method for nearly perfect reconstruction cosine-modulated filter banks, aimed at reducing filter order while preserving signal fidelity. Traditional approaches increase the prototype filter order to minimize distortion, which in turn increases complexity due to more adders and multipliers. The proposed method overcomes this trade-off by first optimizing filter banks with a small number of channels, and then extending the design to a larger number of channels using a Farrow structure. The initial coefficients of the Farrow structure are derived from the prototype filter designed for the small number of channels, and then optimized for filter banks with a large number of channels. This structured approach is particularly suitable for offline application, also enabling an efficient scaling while maintaining low amplitude and aliasing distortion. Implemented in MATLAB with minimal user input, the method provides improvements over the existing techniques in terms of filter order and reconstruction quality.

Keywords: Cosine-modulated filter banks, Transmultiplexer, Software implementation, Optimization

[0]svukotic@raf.edu.rs ORCID; mmladenovic@raf.edu.rs ORCID; djbabic@raf.edu.rs ORCID

1 Introduction

Cosine-modulated filter banks (CMFBs) represent an important type of filter banks (FB) because of their ability to easily generate the entire bank by designing a single prototype filter. Once the prototype filter is determined, the entire bank is obtained by applying cosine modulation to it. They find a wide range of applications in numerous fields: subband coding; image, video, or audio compression; transmultiplexer design; electrocardiogram (ECG) signal processing and signal processing, see [8, 17, 19, 25].

The literature identifies two main categories of CMFBs: orthogonal filter banks with approximate reconstruction (also referred to as NPR) and perfect reconstruction (PR) FBs. In both cases, if the prototype filter's impulse response exhibits even symmetry, the overall phase becomes linear, and the group delay remains constant. As a result, phase distortion is absent, a characteristic adopted in this paper. However, there are two other types of distortion: amplitude and aliasing distortions. These distortions determine the performance of the filter banks, and their quality depends on the prototype filter applied in CMFB.

For lossless coding, PR FBs are applied. On the other hand, for lossy coding, NPR FBs are applied, and they should produce smaller amplitude and aliasing errors compared to those caused by coding. Besides, with NPR it is possible to achieve better frequency selectivity of the channel filters than by PR, see [3].

This paper presents a novel design method for the maximally decimated M-band near perfect reconstruction (NPR) orthogonal cosine-modulated filter banks. We have designed algorithms for filter banks with a small number of channels and large number of channels. For the small number of channels, the prescribed filter bank parameters include filter order, number of channels, amplitude distortion, aliasing error, and stopband attenuation. For a large number of channels, the input parameters are filter order, number of channels, passband and stopband deviations.

The novel design method is implemented using MATLAB with advanced parameter tuning and new data structures. The proposed implementation requires minimal user input; aside from the previously mentioned parameters, no other initial conditions are needed. Furthermore, the proposed method enables the design of a FB with lower amplitude or aliasing distortion maintaining or decreasing the filter order N whilst keeping similar distortion values. By reducing the filter order, the number of multipliers and adders in the prototype filter realization are also reduced. The prototype filter in the proposed method is either Type I or Type II. In Type I N is even, in Type II N is odd, and in both cases the filter coefficients have even symmetry. The efficient realization of Type I filter, which exploits the coefficients symmetry, requires $N/2+1$ multipliers and N two-input adders. Similarly, for Type II filter it is required $(N+1)/2$ multipliers and N two-input adders.

1.1 Literature overview

This subsection surveys methods for generating NPR FBs that are currently present in the literature. A significant body of work has been reported in developing filter banks based on modulation, where a prototype filter is designed to meet the requirements of the analysis and synthesis filter section ([5, 15, 21, 22]). A finite impulse response (FIR) filter is often preferred over the infinite impulse response (IIR) filter for its design of low-pass prototype filters that yield certain desirable features such as: linear phase, inherent stability and negligible quantization noise ([20]). On the other hand, FIR filters often have higher computational requirements in comparison with IIR filters. Thus, many alternative design approaches have been developed such as sign power of two, multiplierless design, and interpolated FIR filters (IFIR) ([10, 15, 16, 23]).

A computationally efficient way to design NPR orthogonal CMFB is to use the window approach [7, 17, 18, 21, 25, 29]. In this approach, it is possible to minimize the FB distortion by properly optimizing the cut-off frequency of the ideal filter. The main

advantage of this method lies in the fact that only a small number of design parameters are required, while the disadvantage is the lack of control over aliasing and distortion errors. In most cases, these design methods take into account only the distortion caused by two neighboring channels. In order to overcome the aforementioned drawbacks, the symmetry of the prototype filter and the distortions should be taken into account in a controlled manner, with upper bounds set for the distortion error.

The high computational requirements of a direct-form FIR filter can be decreased by a cascaded FIR filter realization in the sense of the frequency response masking (FRM) ([4, 11, 27]) and IFIR ([20, 26]) techniques. These structures enable the design of a narrow transition band filter with lower order filters as building blocks. In this way, the number of parameters to optimize NPR CMFB decreases drastically, so the FB can be designed for a large number of channels M.

Another approach, presented in [14] proposes a design for maximally-decimated cosine-modulated filter banks by formulating the prototype filter design as a convex optimization problem. The authors of this paper impose constraints to ensure near-perfect reconstruction and use convex tools to optimize the prototype filter's frequency response, i.e., its stopband attenuation, reconstruction error. Their approach allows systematic trade-offs and better performance than many earlier heuristic or window-based designs, especially in controlling errors in amplitude and aliasing.

More recently, in [13] a method for designing prototype filters was proposed using a constrained least-squares approach with single-parameter optimization. The method ensures rapid convergence and achieves low distortion levels. Its effectiveness is demonstrated through ECG signal sub-band coding, highlighting its potential in healthcare and communication applications.

An alternative approach, based on a direct-form FIR filter, represents a sparse prototype filter ([32]), whose advantage is a low number of non-zero coefficients in the prototype filter. Consequently, the number of multipliers and adders is reduced.

Furthermore, various objective functions are used in optimization models, with the purpose of simplifying the optimization problem at the cost of introducing some amount of distortion e.g. [2, 6, 13, 14, 19, 21, 24, 28, 32].

1.2 Contributions

The contribution of this paper is the novel design method for the maximally decimated orthogonal NPR CMFB and the corresponding MATLAB implementation. It consists of two dedicated algorithms, the first one for filter banks with small number of channels and the second one for the large number of channels. For the small number of channels, the prescribed filter bank parameters include filter order, number of channels, amplitude distortion, aliasing error, and stopband attenuation. For a large number of channels, the input parameters are filter order, number of channels, passband, and stopband deviations. The Farrow structure is used and adapted for a higher number of channels, which enables filter order reduction. FIR subfilters in the structure have a linear phase, and the Farrow structure could be referred to as the modified Farrow structure, but we refer to it as the Farrow structure for simplicity. Several design examples demonstrate the benefits of the proposed method over the existing ones. Compared to the approach with sparse prototype filters and other related filter design methods, the computational results presented in this paper demonstrate that the filter order could be decreased, as well as the number of adders and multipliers, maintaining the amplitude and reconstruction distortions at similar or lower values. In conclusion, it is possible to design FB with high number of bands M, by using the Farrow structure. The proposed solution can be used as a good initial prototype filter with a lower M and then it could be implemented to generate FB with a large M.

2 CMFB design setup

In this section, the basic theory behind CMBFs is presented, as well as the proposed optimization method and prototype filter design technique.

Figure 1 depicts an M-channel maximally decimated FB with analysis FB (AFB) filters $H_k(z)$ and synthesis FB (SFB) filters $F_k(z)$.

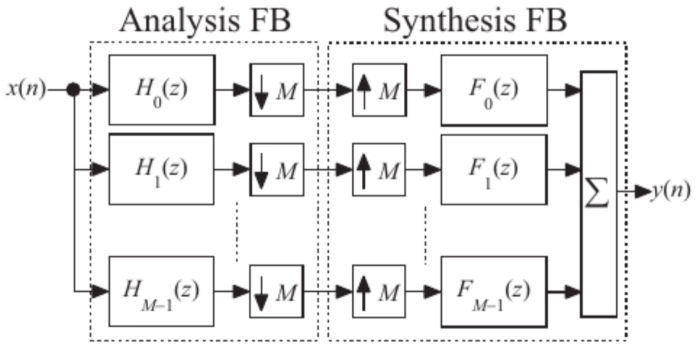

Fig. 1 M-channel maximally decimated FB ([3, 9])

The output signal of the filter bank depicted in the figure can be represented as:

$$Y(z) = T_0(z)X(z) + \sum_{l=1}^{M-1} T_l(z)X(zW_M^l), \tag{1}$$

where $W_M = \exp(-\jmath M 2\pi)$. The distortion function is

$$T_0(z) = \frac{1}{M}\sum_{k=0}^{M-1} F_k(z)H_k(z), \tag{2}$$

and the aliasing transfer functions are

$$T_l(z) = \frac{1}{M}\sum_{k=0}^{M-1} F_k(z)H_k(zW_M^l), \quad l = 1, \ldots, M-1. \tag{3}$$

For a PR FB, $T_0(z) = z^{-N}$ and $T_l(z) = 0$, where N is the prototype filter order and represents the FB delay.

Based on the prototype filter coefficients $h_p(n)$, the AFB and SFB filters are obtained by the following formulae:

$$h_k(n) = 2h_p(n)\cos\left[(2k+1)\frac{\pi}{2M}\left(n-\frac{N}{2}\right)+(-1)^k\frac{\pi}{4}\right], \tag{4}$$

$$f_k(n) = 2h_p(n)\cos\left[(2k+1)\frac{\pi}{2M}\left(n-\frac{N}{2}\right)-(-1)^k\frac{\pi}{4}\right], \tag{5}$$

where $k = 0, \ldots, M-1$, see [3].

It should be emphasized that the formulas derived for the cosine-modulated filter bank are valid for both odd and even prototype filter orders N. In the case of odd N, the filters retain linear phase symmetry, although the center of symmetry lies between two samples rather than at a single sample index.

2.1 Formulation of the optimization problem

For orthogonal NPR FBs, some tolerance is allowed in $T_0(z)$ and $T_l(z)$, and the problem of designing prototype filters can be stated as the least square or minimax continuous optimization problem.

The problem formulation is as follows: find the prototype filter coefficients with transfer function $H_p(z)$ and coefficients symmetry property, i.e., $h_p(N-n) = h_p(n)$:

$$H_p(z) = \sum_{n=0}^{N} h_p(n) z^{-n}, \tag{6}$$

that minimizes E_2 (7) (in the case of least square optimization)

$$E_2 = \int_{\omega_s}^{\pi} |H_p(e^{j\omega})|^2 \, d\omega, \tag{7}$$

or E_∞ (8) (in the case of minimax optimization),

$$E_\infty = \max_{\omega\in[\omega_s,\pi]} |H_p(e^{\jmath\omega})| \,. \tag{8}$$

while the given parameters are $\omega_s = \dfrac{(1+\rho)\pi}{2M}$ with $\rho > 0$, M and N, and $\jmath$ denotes the imaginary constant.

In both optimization problems, the following constraints are imposed:

$$||T_0(e^{\jmath\omega})| - 1| \leq \delta_1, \text{ for } \omega \in [0,\pi], \tag{9}$$

$$|T_l(e^{\jmath\omega})| \leq \delta_2, \ \omega \in [0,\pi], \ l = 1,\ldots,M. \tag{10}$$

The constraints are applied over the full frequency range to account for the complete system behavior, including the combined image responses from all subbands. This approach guarantees that both the distortion function $T_0(\omega)$ and the aliasing terms $T_l(\omega)$ are controlled across the entire spectrum, not just in their most critical bands. Besides δ_1 and δ_2, other commonly used measurements of distortion are the amplitude distortion function magnitude $e_{am}(\omega)$ and the total aliasing transfer function magnitude $e_a(\omega)$, whose maximums are denoted as e_{am} and e_a, respectively ([32]):

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

$$e_a = \max_{\omega} \left\{ \left[\sum_{l=1}^{M-1} |T_l(e^{\jmath\omega})|^2 \right]^{\frac{1}{2}} \right\}, \ \omega \in [0,\pi], l = 1,\ldots,M. \tag{12}$$

2.2 The proposed optimization method for a low number of channels

The design presented in this paper begins by solving the optimization problem for a smaller number of channels, followed by adapting Farrow's structure to solve it for the target value of M. The choice of this smaller channel count depends on the user's

time and hardware constraints. In this work, we define a "lower number of channels" as cases where $M \leq 8$.

In the proposed solution, the stopband edge is set to $\omega_s = \frac{\pi}{M}$ and the transition bandwidth is slightly lower than $\frac{\pi}{2M}$, as in [21]. The minimax optimization is applied in a discrete manner, discretizing the frequency axis. In the proposed solution, $4N$ equidistant points of $H_p(e^{\jmath\omega})$ are used across the full frequency range $[0, 2\pi]$, with a proportionally smaller number of points in the specific range of interest for optimization, i.e., $[\omega_s, \pi]$. The functions $T_0(e^{\jmath\omega})$ and $T_l(e^{\jmath\omega})$ are discretized at N equidistant points within the interval $[0, \pi]$. In contrast, [28] employs $20N$ equidistant points of $H_p(e^{\jmath\omega})$ in the range $[\omega_s, \pi]$, and N points of $T_0(e^{\jmath\omega})$ and $T_l(e^{\jmath\omega})$ in the segment $\left[0, \frac{\pi}{M}\right]$. Since the existing literature does not provide an explicit rule for discretisation, the choice of frequency resolution has been determined empirically.

We conducted experiments with different resolutions of the frequency axis, and the results demonstrated that both reducing and increasing the number of discretization points might degrade the solution quality, either by weakening numerical stability, worsening the objective function under given constraints or prolonging the execution time. For example, in the case of $M = 8$ and $N = 88$ in the proposed method (presented in Table 2 and Figure 3), using $2N$ points instead of $4N$ for the discretization of $H_p(e^{\jmath\omega})$ causes the algorithm to fail to converge. If we set $8N$, we obtain $e_{am} = 1.777 \cdot 10^{-4}$, $e_a = 4.985 \cdot 10^{-6}$ and $a_s = 64.4$ dB, which is very close to the solution for the $4N$ case. However, the execution time increases from 30.7 s to 58.7 s, thus reducing computational efficiency. Thus, the adopted discretization strategy was selected as the most robust in terms of optimization efficiency and final reconstruction quality.

To control the stopband attenuation of the prototype filter, additional constraints are imposed; thus, the discretization of $H_p(e^{\jmath\omega})$ is repeated using N points instead of $4N$. The objective function is minimized similarly to [28], but the constraint functions

differ significantly not only from those in [28], but also from all other previously reported methods.

The `fmincon` function in MATLAB is applied, and the objective function is the discretized version of (8). Similarly, by discretizing the expressions in (9) and (10) in the manner described above, the constraint functions have been set – each discrete frequency gives one constraint function for the distortion transfer function (TDF) and each aliasing transfer function (ATF). In addition, each discrete frequency in the range $[\omega_s, \pi]$ produces a constraint function by setting the condition that the modulo of $H_p(e^{j\omega})$ in the stopband should be less than a specific value, i.e. $|H_p(e^{j\omega}| \leq \delta_3 = 10^{\frac{-a_s}{20}}$ for $\omega \in [\omega_s, \pi]$. Since the optimization requires an initial prototype filter, it is created as an equiripple FIR filter with a desired order N and weight coefficients set such that the stopband attenuation is met. The pseudo code for the entire design is presented in Algorithm 1.

We note here that the initial filter does not have to be exclusively equiripple for the algorithm to converge. For example, one could design a filter using a window function. To compare with the results obtained for $M = 8$ and $N = 88$ presented in Table 2, we designed several initial filters with the Kaiser, Blackman and Hamming window functions, specifying the filter order, cutoff frequency, and also β for Kaiser window, so that the desired attenuation in the stopband is achieved, as in the case of the equiripple filter. The choice of parameters, namely cutoff frequency (CF) and β, proved to be highly sensitive.

In the case of Kaiser window function, when $CF = 0.8 \cdot \omega_s$ and $\beta = 5$, or $CF = 0.6 \cdot \omega_s$ and $\beta = 5$, the algorithm converges. While for $CF = 0.5 \cdot \omega_s$ and $\beta = 5$, or $CF = 0.5 \cdot \omega_s$ and $\beta = 6$, the algorithm does not converge.

When applying Blackman or Hamming window function, choosing $CF = 0.55 \cdot \omega_s$ achieves the desired attenuation in the stopband, but in both cases, the algorithm does not converge. The proposed method assumes $\omega_s = 1/M$ and expects the user to

input the desired attenuation in the stopband. The drawback of the window approach is that the parameters used to design the window functions do not allow us to directly control these values.

2.3 The proposed method of optimization for a large number of channels

In [8], it has been shown that a FB with a high value of M could be generated by the Farrow structure. However, like any regular filter design method, the number of constraints increases with M, and the optimization becomes more demanding and time consuming. The benefit of the method proposed in [8] is that it allows to make the initial filter with a small value of M, and then conduct the optimization based on that initial filter producing FB with allowed distortion level δ defined as follows:

$$|H_p(e^{\jmath\omega})| \leq \delta, \text{ for } \omega \in [\omega_s, \pi], \tag{13}$$

$$|P(e^{\jmath\omega}) - 1| \leq \delta, \text{ for } \omega \in [0, \frac{\pi}{M}]. \tag{14}$$

Algorithm 1 Pseudo code for the proposed method

function PROPOSED_METHOD( )
  if lower channel number M **then**
    $\langle M, N, \delta_1\_in, \delta_2\_in, \delta_3\_in \rangle \leftarrow$ READ( ) ▷ Input parameters
    if $N \mod 2 = 0$ **then**
      $solution \leftarrow$ SOLVE1$(M, N, \delta_1_in, \delta_2_in, \delta_3_in)$
    else
      $solution \leftarrow$ SOLVE2$(M, N, \delta_1_in, \delta_2_in, \delta_3_in)$
    end if
  else
    $\langle M, m, \delta\_in, N_s \rangle \leftarrow$ READ( ) ▷ Input parameters, $m < M$
    ▷ $N \leftarrow m \cdot N_s + m - 1$
    $solution \leftarrow$ SOLVE2(m, δ_in, N_s) ▷ $\delta\_in = \delta_1\_in = \delta_2\_in = \delta_3\_in$
    $farrow\_coef \leftarrow$ COMPUTEFARROW$(solution, m, N_s)$
    $solution \leftarrow$ SOLVE3$(M, N_s, \delta_in, farrow_coef)$
  end if
  return *solution*
end function

Equation (14) is referred to as the power complementary property, where P is defined as:

$$P(e^{\jmath\omega}) = |H_p(e^{\jmath\omega})|^2 + |H_p(e^{\jmath(\omega-\pi/M)})|^2 \approx 1, \text{ for } \omega \in [0, \frac{\pi}{M}]. \tag{15}$$

The input entered by the user corresponding to the Algorithm 1, where, in the case of designing a bank with a small number of channels, the user enters M, N, the allowed distortion (δ_1_in, δ_2_in), and the attenuation of the prototype filter (i.e., stopband ripple δ_3_in), as well as the ratio of the passband and stopband ripple. The user can easily obtain this ratio, for example, through `fvtool` or `filterDesigner` in MATLAB by setting the same order, band edge frequencies, and attenuation for the equiripple filter (the initial filter in this stage) as the prototype.

When designing a bank with a large number of channels, the first stage of designing with a small number of channels should be applied first, where the allowed distortion and attenuation of the filter are defined by a single parameter, δ_in; i.e., $\delta_1\_in = \delta_2\_in = \delta_3\_in = \delta\_in$. The other parameters are set as already described for the first phase, taking into account that M and N are connected by the Farrow structure parameter N_s, which is also chosen by the user and is the same for both phases. In the second phase, when a bank with a large number of channels is designed, the prototype filter obtained in the first phase serves as one of the input data (as an initial filter). The other input data consist of the desired value M, the parameter N_s (to which N is connected), and δ_in. More precisely, we follow the following guideline:

- We choose the number of channels in the FB with a small number of channels and the parameter N_s because they determine the number and order of FIR filters that make up the Farrow structure. That is, the number of its coefficients. That number does not change in any phase of the algorithm.

- We design the initial filter by applying the algorithm for a small number of channels, where the order of the filter is selected according to the chosen parameter N_s and the number of channels according to the relationship given in the following text. In doing so, δ_1_in, δ_2_in, and δ_3_in are set to be equal to δ_in.
- After the initial prototype filter is designed for a lower number of channels, the initial Farrow coefficients are calculated from it.
- The desired large number of channels in the FB is selected, which together with N_s determines the order of the prototype filter according to the same relationship between the number of channels and N_s as in bullet point 2.
- Based on the initial Farrow coefficients and the desired amount of distortion δ_in, the optimization algorithm calculates the final Farrow coefficients and from them the coefficients of the FB filter prototype with a large number of channels.

In this study, we demonstrate that the prototype filter, as described in Section 2.2, can serve as the initial filter in the algorithm to design large CMFBs M, which represents one of the key contributions of this work. In addition, we have shown that proper adjustment of the Farrow structure makes it possible to design a CMBF with a lower value of N than in [8] while maintaining the same M and δ errors. The coefficients of the Farrow structure are obtained from the prototype filter designed for a small number of channels, which differs from the methods previously reported, based on the time or frequency domain approach [1].

In [8], there are $L + 1$ subfilters of order N_s forming a Farrow structure. The prototype filter order N and number of channels M are related as $N = N_sM + M - 1$. The novelty of our method lies in the unique approach used to generate the initial Farrow coefficients, which differs from previously reported techniques. These initial coefficients are derived from a prototype filter designed using the procedure outlined in Section 2.2 for a lower value of M. The relation between the prototype filter coefficients and the Farrow coefficients is defined by the system of equations $h_p(nM + m) =$

$\sum_{k=0}^{L} s_k(n) \left(-\frac{m}{M} + \frac{1}{2} - \frac{1}{2M} \right)^k$, where $m = 0, 1, ..., M-1$ and $s_k(n)$ are coefficients in the kth subfilter in the Farrow structure, see (28) in [8]. The choice $L = M - 1$ is directly motivated by the structure of the governing equation, which leads to a square system matrix with respect to the Farrow coefficients. This square form guarantees the existence of a unique solution for the coefficient set. Furthermore, we can notice that the number of prototype filter coefficients regarding the previously mention relation of N and M is $M(N_s+1)$, and the number of Farrow coefficients is $(L+1)(N_s+1)$. By setting $L = M - 1$ we can calculate the prototype filter coefficients from the Farrow coefficients and vice versa. We emphasize the fact that M in this phase corresponds to the lower value from the first phase of the algorithm. Once the Farrow coefficients are obtained from the prototype filter for lower M, these coefficients are optimized in order to obtain the prorotype filter coefficients for higher M. The previously mentioned relation of prototype and Farrow coefficient is also valid for higher M, so the prototype filter coefficients are still uniquely determined by the Farrow coefficient. The next step is to conduct optimization of Farrow coefficients. The advantage of this approach is that the number of optimization variables equals the number of the Farrow coefficients. It is the same for lower and higher M and significantly smaller than the number of prototype filter coefficients for higher M. The number of unknown Farrow coefficients in an optimization procedure depends on L and N_s, and is given by (11) in [8]. By selecting $M = 4$, we obtain the same L as in [8]. In our implementation, we set N_s to 11, while in [8] N_s is fixed to 15. Because of that, N is lower to the mentioned relation of N and M, and there is also fewer unknown Farrow coefficients to optimize.

After obtaining the Farrow coefficients from an initial prototype filter, the CMBF with the desired higher M and corresponding N is obtained by setting the objective function as in (8). The objective function and the constraint functions are obtained from the Farrow coefficients since the prototype filter coefficients are calculated from them. In addition, minimax optimization is applied by discretizing the frequency axis

in (8), where $4M$ equidistant points of $H_p(e^{\jmath\omega})$ are used in the frequency range $[0,\, 2\pi]$ and proportionally less in the frequency range of interest for optimization – $[\omega_s,\, \pi]$. Note that, choosing $2M$ or $8M$ points results in the algorithm failing to converge, so $4M$ is empirically estimated to be an adequate choice. We set the constraint functions as in (13) and (14), where $20M$ equidistant points are taken in the frequency range $[\omega_s,\, \pi]$ in (13) and M in the frequency range $[0,\, \pi/M]$ in (14). Through simulations, we determined this number of frequency points, following the method described in Section 2.2, as using lower or higher numbers leads to result degradation. An additional ablation analysis was performed to evaluate the role of the initialization phase and constraint sets. The results indicate that using the proposed initialization leads to faster convergence and lower filter order compared to trivial or random initialization, while the distortion parameters directly control the trade-off between filter order and performance. These findings confirm the necessity of the two-phase design.

2.4 Time complexity and sensitivity analysis

In this subsection, we provide an in-depth performance analysis of the proposed design. Table 1 indicates that modifications to key input parameters influence the output metrics in the desired fashion. See Section 3 for detailed comparisons. It should be noted that parameters such as δ_{in} are not optimization targets, but upper bounds in inequality constraints (cf. (9)–(10)). Consequently, the optimization process (e.g., `fmincon`) stops once a feasible local minimum of the objective function (8) is found (not necessarily driving the constraints to equality). Therefore, in some cases the achieved performance, e.g., δ or a_s, may be strictly better than the imposed bounds, which explains situations such as the improvement of a_s when δ_1_in or δ_2_in or δ_3_in is relaxed while the optimizer finds a different local minimum.

Key input parameters			Key output metrics		
δ_1_in	δ_2_in	δ_3_in/a_s_in[db]	δ_1	δ_2	as [db]
Tuning δ_3_in					
10^{-4}	10^{-6}	0.001 / 60	1.0002×10^{-4}	1.0047×10^{-6}	63.2
10^{-4}	10^{-6}	0.002 / 54.0	1.0002×10^{-4}	1.0047×10^{-6}	64.2
10^{-4}	10^{-6}	0.005 / 46	1.0000×10^{-4}	1.6584×10^{-6}	64.3
Tuning δ_2_in					
10^{-4}	10^{-6}	0.001 / 60	1.0002×10^{-4}	1.0047×10^{-6}	63.2
10^{-4}	5×10^{-6}	0.001 / 60	1.0000×10^{-4}	5.0006×10^{-6}	67.1
10^{-4}	1.5×10^{-5}	0.001 / 60	1.0000×10^{-4}	1.5000×10^{-5}	68.2
Tuning δ_1_in					
10^{-4}	10^{-6}	0.001 / 60	1.0002×10^{-4}	1.0047×10^{-6}	63.2
4×10^{-4}	10^{-6}	0.001 / 60	3.8583×10^{-4}	1.8470×10^{-6}	66.9
5×10^{-2}	10^{-6}	0.001 / 60	5×10^{-2}	1.7800×10^{-6}	82.5

Table 1 Sensitivity analysis based on $\delta_i, i = 1, \ldots 3$ parameters, $M = 4$ and $N = 88$

The code was executed on a computer with the following specifications: CPU - AMD Ryzen 7 3700X, RAM - 64GB (2x32GB in dual-channel, @3200MT/s), OS - Manjaro Linux, Kernel 6.6 LTS. Parallel Computing Toolbox with 8 workers was used in MATLAB R2022a. While not primarily intended for real-time operation, the proposed method is well-suited to offline scenarios, with potential applications in tasks such as channelization. Note that, depending on the system architecture the results can slightly deviate.

To provide an overview of the time complexity, Figure 2 illustrates the relationship between the execution time and $M \times N$. It can be observed that for both lower and higher M values, time complexity curves exhibit approximately linear behavior. Note that, although the range of $M \times N$ is several order of magnitude higher for larger M, the execution times have a lower gradient, e.g., for $M \times N = 704$ the execution time would be almost 32 seconds, whilst in phase 2, for $M \times N = 760$, it takes 2.3 seconds, see Figure 2. Regarding memory usage, it was observed that it reaches approximately 0.5GB during Phase 1 and up to 2GB in Phase 2. These values indicate a moderate memory footprint, which remains acceptable for modern computing systems and suggests the method is feasible for practical applications involving larger-scale models.

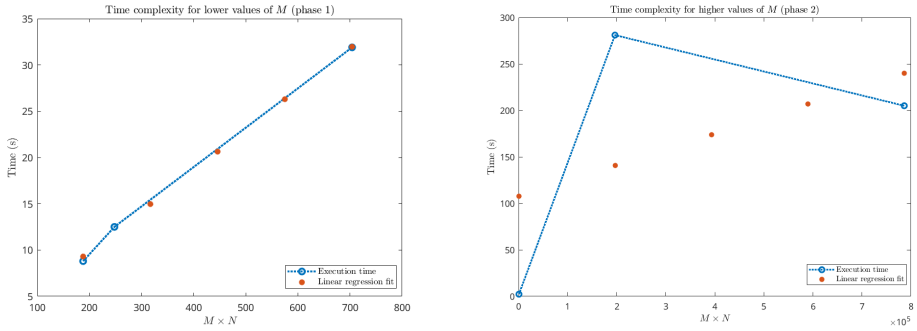

Fig. 2 Execution times for lower and higher M values with linear regression fit

Therefore, the proposed algorithm demonstrates appropriate scalability with respect to both the increasing number of channels and higher filter orders. As these parameters increase, the algorithm maintains consistent computational performance without exhibiting exponential increases in execution time or memory usage. This indicates that the method is well-suited for applications involving high-dimensional data or more complex filtering requirements. For larger values of M ($M \geq 32$), direct optimization is practically infeasible due to excessive execution time and high memory requirements. The proposed two-phase method, by contrast, provides a scalable and efficient solution even for large M, demonstrating its practical advantage.

3 Simulation results and discussion

In this section, we present the achieved results and provide a comparison to the current state of the art. We note that the comparisons may be affected by the platform on which the codes were executed and by the stopping conditions. First, we demonstrate the case when the number of channels M has lower values.

In [32] the sparse prototype filter design is considered. The advantage of [32] over other optimization techniques, including the weighted constrained least square algorithm (WCLS), the quantum-behaved particle swarm optimization algorithm (QBPSO), and the gradient information algorithm (GI), refers to a lower number of

nonzero taps of the filter keeping comparable levels of the distortions. In our work, we do not re-implement the sparse filter design; instead, we compare our approach with it. The main advantage of the sparse design lies in reducing the number of multipliers, whereas in our method this reduction is achieved by lowering the filter order. In the proposed method, the filter order can be equal or less to the number of nonzero taps than in [32] while maintaining a similar level of distortion, as presented in Table 2. From the table, we can observe that for $M = 8$, a filter of order $N = 88$ is obtained, and according to the formulas from the introduction, it has $(88/2)+1 = 45$ multipliers, 88 two-input adders, and 88 delay elements. The sparse filter with which we are comparing is of order $N = 160$ and has 98 non-zero taps, which means that it has $(98/2)+1 = 50$ multipliers, 98 two-input adders, and 160 delay elements. To ensure a fair comparison, in Table 2, we calculated the errors according to the same definitions as in the respective references, based on formulas (11) and (12).

Design method	M	N	Number of nonzero taps	e_{am}	e_a
WCLS	4	80	80	3.1×10^{-3}	1.06×10^{-6}
QBPSO	4	80	80	7.01×10^{-4}	8.49×10^{-8}
Xu et al.	4	140	62	1.7×10^{-3}	5.06×10^{-7}
Proposed method	4	**62**	62	1.10×10^{-3}	6.2156×10^{-7}
WCLS	8	144	144	1.80×10^{-3}	3.90×10^{-7}
QBPSO	8	132	132	2.14×10^{-5}	1.30×10^{-5}
Xu et al.	8	160	98	5.20×10^{-4}	9.85×10^{-6}
Proposed method	8	**88**	88	1.7773×10^{-4}	4.9604×10^{-6}
WCLS	16	224	224	1.30×10^{-3}	9.03×10^{-7}
QBPSO	16	256	256	4.73×10^{-4}	2.34×10^{-6}
Xu et al.	16	254	166	5.21×10^{-4}	4.24×10^{-6}
Proposed method	16	**166**	166	9.3672×10^{-4}	3.7248×10^{-6}

Table 2 Comparison of results with sparse prototype filters and other related filter design methods

The magnitude responses of the proposed prototype filters, in Table 2, are presented in Figure 3. In [32], we observe that in all three scenarios, the stopband attenuation is less than 60 dB and the same is accomplished in the proposed method.

In [18], CMFB is designed by the logarithmic window function, where the cut-off frequency (ω_c) of the prototype filter is optimized by the incremental ant colony optimization with local search (IACOR - LS). The designed uniform CMFB has lower aliasing errors and comparable amplitude distortion, compared to the previously reported [18].

In [13], the prototype filter, obtained by a weighted constrained least square method where the cutoff frequency is the only parameter to be optimized, exhibits minimized maximum reconstruction errors, surpassing those from alternative methods reported previously. Table 3 compares the proposed method and [13, 18] where $M = 8$. In [18], authors proposed α, the window shape parameter, to be 4 or 5. In [13], the error ratio k, referring to the prototype filter characteristics in the passband and stopband, varies from 1 to 20. Accordingly, we choose to represent the results for $k = 1$ and $k = 20$ as the final values and a_s is the stopband attenuation of the prototype filter. The results show that the proposed method provides a significant reduction in the order of the prototype filter while offering a competitive trade-off between amplitude and aliasing distortion.

Design Method	Filter order	δ_1	δ_2	a_s
Keerthana et al., $k = 1$	144	9.77×10^{-4}	1.05×10^{-6}	-85dB
Keerthana et al., $k = 20$	144	9.77×10^{-4}	7.27×10^{-7}	-85dB
Proposed method	144	5.27×10^{-4}	1.73×10^{-6}	-95dB
Kumar and Sunkaria, $\alpha = 5$	160	6.64×10^{-3}	3.00×10^{-7}	-
Kumar and Sunkaria, $\alpha = 4$	160	7.42×10^{-3}	5.00×10^{-7}	-
Keerthana et al., $k = 1$	160	9.77×10^{-4}	9.83×10^{-7}	-80dB
Keerthana et al., $k = 20$	160	9.78×10^{-4}	6.46×10^{-7}	-80dB
Proposed method	160	4.48×10^{-4}	2.57×10^{-6}	-80dB

Table 3 Comparison of results with the windowing methods and the weighted constrained least-square methods for prototype filter design

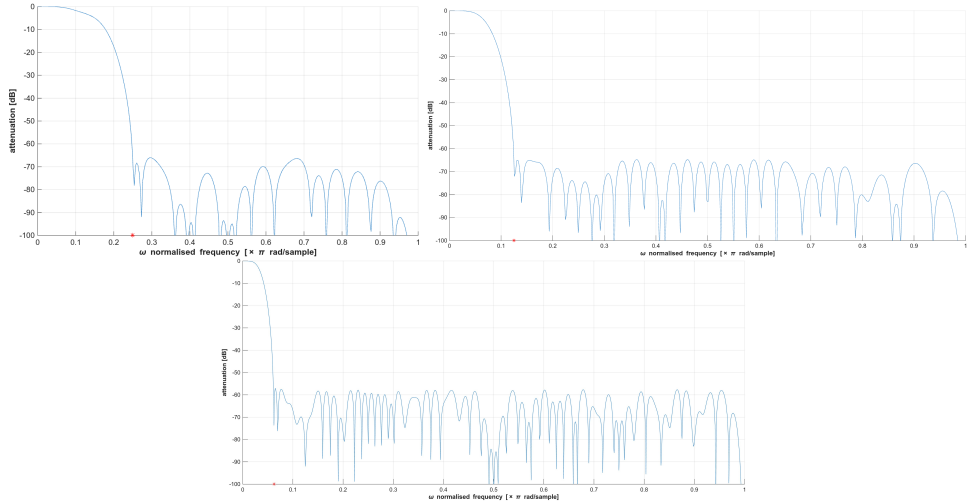

Fig. 3 The magnitude responses of the proposed prototype filters for $M = 4$, $N = 62$ (top left); $M = 8$, $N = 88$ (top right) and $M = 16$ and $N = 166$. The red dot denotes the stopband frequency.

Design method	Filter order	M	N_s	δ
Eghbali and Johansson	127	8	15	1.92×10^{-3}
Eghbali and Johansson	65535	4096	15	1.924×10^{-3}
Proposed method	95	8	11	1.5×10^{-3}
Proposed method	3071	256	11	1.92×10^{-3}
Proposed method	6143	512	11	1.924×10^{-3}

Table 4 Comparison of results between FBs design methods based on the Farrow structure

The magnitude response of the proposed prototype filter from Table 3 and with $N = 144$ is presented in Figure 4. The remainder of this section is focused on higher values of M.

In [8], the CMBF is created with $M = 8$, $L = 3$, and $N_s = 15$, leading to $\delta = 1.92 \cdot 10^{-3}$. Based on that initial prototype filter, with a low M, another CMBF with a high M can be created, such as the one with $M = 4069$. At the same time, the relation between N and M remains unchanged ([8]):

$$N = N_s \cdot M + M - 1.$$

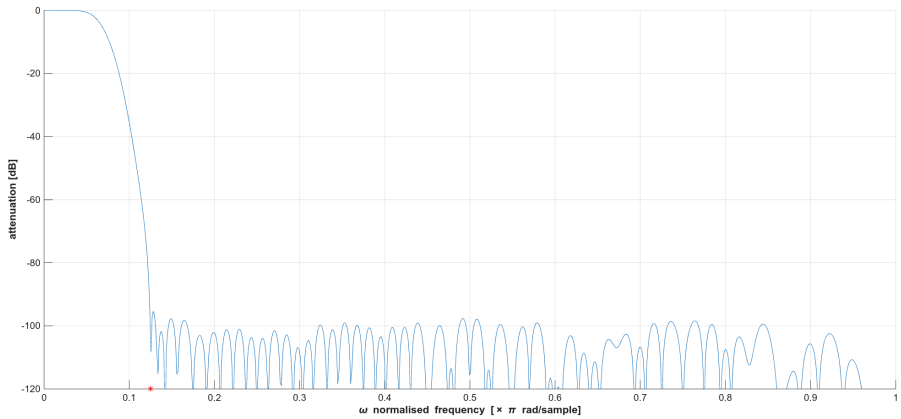

Fig. 4 The magnitude response of the prototype filter for $M = 8$ and $N = 144$. The red dot denotes the stopband frequency.

With the appropriate choice of M in the initial filter described in Section 2.3, CMBF with the parameter $N_s = 11$ is created, leading to a reduction of filter order for the same M by approximately 25%, while the filter order can be approximated as $N \approx (N_s + 1)M$. The reduction of filter order is indeed achieved by decreasing the parameter N_s from 15 to 11, while maintaining the same or slightly lower δ distortion compared to [8]. A comparison of the proposed method and [8] is presented in Table 4. One can examine that for $N = 127$, there are 64 multipliers and 127 two-input adders required, and for $N = 95$, 48 multipliers and 95 two-input adders are required.

4 Conclusion and future work

We have presented a novel design method and software implementation of the NPR CMBF prototype filter design. The proposed method consists of two dedicated algorithms: one for filter banks with a small number of channels, and another for those with a large number of channels. One of the contributions is the introduction of a new two-phase strategy for designing CMFB with a large number of channels. The CMFB design for handling a high number of channels incorporates the Farrow structure, which is configured to achieve a prototype filter order that is approximately 25%

lower than reported in [8]. By reducing the prototype filter order, the number of multipliers and adders in the CMFB realization becomes proportionally lower. We have shown by means of examples for both cases, i.e., for low and high number of channels, that the proposed design method results in a lower prototype filter order while maintaining similar distortion error compared to the state of the art solutions. Conversely, for a given prototype filter order, the proposed design method is capable of achieving a lower distortion error compared to state of the art solutions. It is not best suited for real-time result delivery and can require parameter tuning to balance the desired output. However, it provides tangible improvement on existing solutions and a stable hands-on solution for offline usage.

The proposed method can find application in channelization in 5G networks [12] and cognitive radio. In [12], the CMBF is combined with code-division multiple access (CDMA), where the data symbols are transmitted on different sub-bands and are also spread in each sub-band by a CDMA code. In future work, the authors will examine cyclic features of the signal obtained by combining CMBF and CDMA as a promising technique in future wireless communication networks, and the possibilities of its detection and classification [30, 31] as part of spectrum sensing. Moreover, as part of future work, the benefits of the proposed design will be demonstrated on digital signal processing in communication systems mentioned above.

Acknowledgments

The authors express sincere gratitude to the anonymous reviewers for their valuable comments and suggestions, which helped improve the quality and clarity of this paper.

Declarations

Data & code Availability

Data and code are available at research.raf.edu.rs/public_repos/svukotic/.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] D. Babic. Windowing design method for polynomial-based interpolation filters. *Circuits, Systems, and Signal Processing*, 32(2):759–780, 2013.

[2] S. W. Bergen and A. Antoniou. An efficient closed-form design method for cosine-modulated filter banks using window functions. *Signal Processing*, 87(5):811–823, 2007.

[3] R. Bregović. *Optimal design of perfect-reconstruction and nearly perfect-reconstruction multirate filter banks.* PhD thesis, Tampereen Teknillinen Korkeakoulu, Tampere University of Technology, PO Box 527, FIN-331 01 Tampere, Finland, 2003.

[4] R. Bregović, Y. C. Lim, and T. Saramäki. Frequency response masking based design of two-channel FIR filterbanks with rational sampling factors. In *2005 International TICSP Workshop on Spectral Methods and Multirate Signal Processing*, pages 11–18, 2005.

[5] C. D. Creusere and S. K. Mitra. A simple method for designing high-quality prototype filters for M-band pseudo QMF banks. *IEEE Transactions on signal*

processing, 43(4):1005–1007, 1995.

[6] F. Cruz-Roldán, Á. M. Bravo-Santos, P. Martì-Martìn, and R. Jiménez-Martìnez. Design of multi-channel near-perfect-reconstruction transmultiplexers using cosine-modulated filter banks. *Signal Processing*, 83(5):1079–1091, 2003.

[7] A. Datar, A. Jain, and P. C. Sharma. Design of kaiser window based optimized prototype filter for cosine modulated filter banks. *Signal Processing*, 90(5):1742–1749, 2010.

[8] A. Eghbali and H. Johansson. On efficient design of high-order filters with applications to filter banks and transmultiplexers with large number of channels. *IEEE Transactions on Signal Processing*, 62(5):1198–1209, 2014.

[9] A. Eghbali, H. Johansson, and P. Löwenborg. Reconfigurable nonuniform transmultiplexers using uniform modulated filter banks. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 58(3):539–547, 2010.

[10] Z. G. Feng and K. L. Teo. A discrete filled function method for the design of FIR filters with signed-powers-of-two coefficients. *IEEE Transactions on signal processing*, 56(1):134–139, 2007.

[11] M. B. Furtado, P. S. Diniz, and S. L. Netto. Optimized prototype filter based on the FRM approach for cosine-modulated filter banks. *Circuits, Systems and Signal Processing*, 22(2):193–210, 2003.

[12] L. E. Ghorab, E. F. Badran, A. I. Zaki, and W. K. Badawi. Multicarrier technique for 5G massive MIMO system based on CDMA and CMFB. *Optical and Quantum Electronics*, 55(1):25, 2023.

[13] B. Keerthana, N. Raju, R. Anbazhagan, T.-h. Kim, F. Mohammad, et al. Designing optimal prototype filters for maximally decimated cosine modulated filter banks with rapid convergence. *Heliyon*, 10(11), 2024.

[14] H. H. Kha, H. D. Tuan, and T. Q. Nguyen. Efficient design of cosine-modulated filter banks via convex optimization. *IEEE Transactions on Signal Processing*, 57(3):966–976, 2008.

[15] R. D. Koilpillai and P. Vaidyanathan. Cosine-modulated FIR filter banks satisfying perfect reconstruction. *IEEE Transactions on signal processing*, 40(4): 770–783, 1992.

[16] K. A. Kotteri, A. E. Bell, and J. E. Carletta. Multiplierless filter bank design: structures that improve both hardware and image compression performance. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(6):776–780, 2006.

[17] A. Kumar and B. Kuldeep. Design of M-channel cosine modulated filter bank using modified exponential window. *Journal of the Franklin Institute*, 349(3): 1304–1315, 2012.

[18] A. Kumar and R. K. Sunkaria. Design of uniform cosine modulated filter bank using IACOR-LS and its application in baseline wander removal from ECG signal. *AEU-International Journal of Electronics and Communications*, 150:154198, 2022.

[19] A. Kumar, G. K. Singh, and R. S. Anand. 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, 2011.

[20] R. Kumar Soni, A. Jain, and R. Saxena. A design of IFIR prototype filter for cosine modulated filterbank and transmultiplexer. *AEU-international Journal of Electronics and Communications*, 67(2):130–135, 2013.

[21] Y.-P. Lin and P. Vaidyanathan. A kaiser window approach for the design of prototype filters of cosine modulated filterbanks. *IEEE signal processing letters*, 5(6):132–134, 1998.

[22] H. S. Malvar. Modulated QMF filter banks with perfect reconstruction. *Electronics letters*, 26(13):906–907, 1990.

[23] Y. Neuvo, D. Cheng-Yu, and S. Mitra. Interpolated finite impulse response filters. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(3):563–570, 1984.

[24] T. Q. Nguyen. Digital filter bank design quadratic-constrained formulation. *IEEE Transactions on Signal Processing*, 43(9):2103–2108, 1995.

[25] S. C. Prema and K. Dasgupta. An iterative design with variable step prototype filter for cosine modulated filter bank. *Radioengineering*, 25(1):156–160, 2016.

[26] N. R. Rayavarapu and N. R. Prakash. A computationally efficient design for prototype filters of an M-channel cosine modulated filter bank. *International Journal of Electronics and Communication Engineering*, 2(7):1473–1475, 2008.

[27] L. Rosenbaum, P. Löwenborg, and H. Johansson. An approach for synthesis of modulated-channel FIR filter banks utilizing the frequency-response masking technique. *EURASIP Journal on Advances in Signal Processing*, 2007:1–13, 2006.

[28] T. Saramaki and R. Bregović. An efficient approach for designing nearly perfect-reconstruction cosine-modulated and modified DFT filter banks. In *2001*

IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings, volume 6, pages 3617–3620. IEEE, 2001.

[29] R. K. Soni, A. Jain, and R. Saxena. An optimized transmultiplexer using combinational window functions. *Signal, Image and Video Processing*, 5(3):389–397, 2011.

[30] S. Vukotić and D. Vućić. Detection and clasiffication of OFDM/QAM and OFDM/OQAM signals based on cyclostationary features. In *2015 23rd telecommunications forum Telfor (TELFOR)*, pages 232–235. IEEE, 2015.

[31] S. Vukotić and D. Vučić. Cyclic feature-based joint detection and clasiffication of OFDM/QAM, OFDM/OQAM and SC-FDMA signals. In *2016 24th Telecommunications Forum (TELFOR)*, pages 1–4. IEEE, 2016.

[32] W. Xu, Y. Li, J. Miao, J. Zhao, and X. Gao. A novel design of sparse prototype filter for nearly perfect reconstruction cosine-modulated filter banks. *Algorithms*, 11(5):77, 2018.