Design of Multiplier-less Cosine Modulated Filter Banks with Sharp Transition using Evolutionary Algorithms

Shaeen K
Department of Electronics and Communication Engg.
National Institute of Technology Calicut,
Kerala, India.

Elizabeth Elias
Department of Electronics and Communication Engg.
National Institute of Technology Calicut,
Kerala, India.

ABSTRACT

This paper proposes a totally multiplier-less approximate reconstruction cosine modulated filter bank. The prototype filter is designed to obtain a computationally efficient, sharp transition width, finite impulse response (FIR) filter using the frequency response masking (FRM) approach combined with the canonic signed digit (CSD) representation of the coefficients. Meta-heuristic algorithms are deployed to optimize the performance of the filter bank. The algorithms are modified to suit the optimization problem, in which the objective function is non linear and the search space consists of integers.

Keywords:

Multiplierless Cosine Modulated Filter Banks, Canonic Signed Digit, Frequency Response Masking, Artificial Bee Colony Algorithm, Harmony Search Algorithm

1. INTRODUCTION

Cosine Modulated Filter banks (CMFB) are widely used in different multirate applications [12]. The main advantages of the Cosine Modulated filter banks are that, they have computationally efficient design and all the coefficients are real. It is sufficient to design only the prototype filter. All the analysis and synthesis filters are derived from this filter by cosine modulation. Since all the analysis and synthesis filters are modulated versions of the prototype filter, the shapes of their amplitude responses are the same as those of the prototype filter. Therefore the optimization of the passband ripple and stopband attenuation can be reduced to that of the prototype filter.

If the prototype filter has very sharp response, then the modulated filters will have reduced amount of overlap. Designing FIR filters with very narrow transition width, results in very high order. Hence, as the transition width of the prototype filter decreases, the implementation cost increases. Cosine Modulated filter banks with sharp transition can be achieved with less complexity by the frequency response masking (FRM) approach[5, 1]. The CMFB-FRM coefficients can be efficiently converted to signed power of two (SPT)[6] representation and this will result in a multiplier-less implementation of the cosine modulated filter banks with narrow transition width.

In this paper, a totally multiplier-less design of near perfect reconstruction cosine modulated filter banks with sharp transition is proposed. Conventional FRM filter is used as the prototype filter. FRM approach results in a tremendous amount of reduction in the number of multipliers because of the large number of sparse coefficients [5]. To further reduce the implementation complexity, the FRM filter coefficients are represented with minimum number of SPT terms using canonic signed digit (CSD) representation[19]. The design of the FRM filter in the discrete space can degrade the filter performance. This calls for the use of non-linear optimization techniques. Meta-heuristic algorithms are used in this paper because classical gradient based techniques cannot be directly applied due to the fact that the search space consists of integers. This results in the proper tuning of the parameters with respect to a specific design problem, which in turn can lead to global solution [18]. The resulting prototype filter is multiplierless and hence the design leads to a totally multiplierless cosine modulated filter bank. Since multipliers are the most power consuming circuits and occupy large space, the multiplierless realization will reduce complexity, power consumption and silicon area. Such a multiplierless filter bank is hitherto not reported in the literature.

Cosine Modulated Filter banks with sharp transition width using frequency response masking approach has been proposed in [1] and [14]. In [1], they have used a conventional FRM filter as the prototype filter. They have worked on cosine modulated transmultiplexers and their aim was to reduce the optimization parameters. In [14], they have used a modified FRM filter with non linear phase response as the prototype filter and their main aim was to reduce the arithmetic complexity.

This paper is organized as follows. Section 2 gives a brief review of the cosine modulated filter banks. An overview of the frequency response masking approach is presented in section 3. Section 4 describes canonic signed digit representation. Section 5 discusses the proposed design of the CSD coefficient CMFB using FRM approach. The first part of this section presents the continuous coefficient filter bank design and the second part deals with the CSD coefficient filter bank and the different objective functions. Section 6 describes the different metaheuristic algorithms for the optimization of the CSD encoded prototype filter. The performance comparison of the different optimization algorithms is done in Section 7 and finally Section 8 concludes the paper.

2. COSINE MODULATED FILTER BANKS

The conventional M channel maximally decimated filter bank is shown in Figure 1[12]

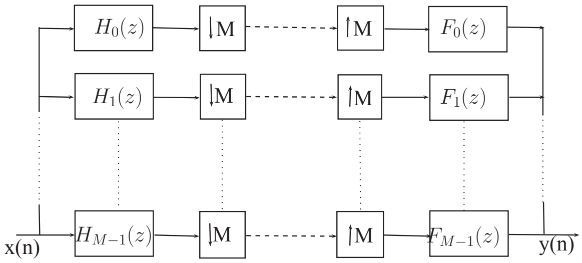

Fig. 1. M Channel maximally decimated filter bank

The reconstructed output, $Y(z)$ is given by the following expression [12]

$$Y(z) = T_0(z)X(z) + \sum_{l=1}^{M-1} T_l(z)X(ze^{-j2\pi/M}) \quad (1)$$

where $T_0(z)$ is the distortion transfer function and $T_l(z)$ is the aliasing transfer function.

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

$$T_l(z) = \frac{1}{M}\sum_{k=0}^{M-1} F_k(z)H_k(ze^{-j2\pi l/M}) \quad (3)$$

$$l = 1, 2.........M-1$$

In the cosine modulated filter banks, if $p_0(n)$ is the impulse response coefficients of the prototype filter, the coefficients of the analysis $(h_k(n))$ and synthesis $(f_k(n))$ filters are given by the following expressions [12].

$$h_k(n) = 2p_0(n)cos\left(\frac{\pi}{M}(k+0.5)(n-\frac{N}{2}) + (-1)^k\frac{\pi}{4}\right) \quad (4)$$

$$f_k(n) = 2p_0(n)cos\left(\frac{\pi}{M}(k+0.5)(n-\frac{N}{2}) - (-1)^k\frac{\pi}{4}\right) \quad (5)$$

$$k = 0, 1, 2.........M-1$$

$$n = 0, 1, 2.........N-1$$

Amplitude distortion error is given by

$$E_r = \max_{\omega} \mid [\mid MT_0(e^{j\omega}) \mid -1] \mid \quad (6)$$

The worst case aliasing distortion is given by

$$E_a = \max_{\omega}(T_{alias}(\omega)) \quad (7)$$

where

$$T_{alias}(\omega) = \frac{1}{M}\left[\sum_{l=1}^{M-1} \mid T_l(e^{j\omega}) \mid^2\right]^{\frac{1}{2}} \quad (8)$$

Cosine modulation results in a class of filters with only real coefficients. Adjacent channel aliasing cancellation is inherent in the filter bank design itself [12]. If the initial filter chosen is a linear phase filter, then the overall response will have linear phase, thereby eliminating the phase distortion.

In this paper a multiplierless near perfect reconstruction cosine modulated filter bank that allows small amplitude and aliasing errors causing small distortions to the signal is designed. Although there exists small errors, these type of filter banks are widely used in different applications like MPEG audio coder [7].

3. FREQUENCY RESPONSE MASKING APPROACH

Frequency Response Masking Approach [5] is a cost efficient way of designing linear phase FIR filters with sharp transition and low complexity. In the normal FIR filter design technique, as the transition bandwidth decreases the design complexity increases since the transition bandwidth is inversely proportional to the order of the filter. The FRM technique allows to realise arbitrary bandwidth sharp transition FIR filter specifications. FRM technique uses the concept that interpolation by L reduces the transition bandwidth by L times [5]. It is also possible to easily quantize the coefficients of the FRM filter into signed power of two terms.The structure of the FRM filter is shown in Figure 2 [5].

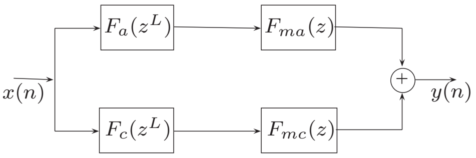

Fig. 2. A realization structure of conventional FRM filter

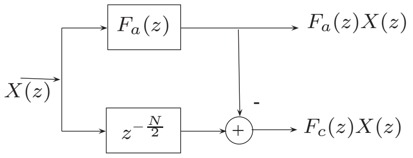

Fig. 3. Realization structure of complementary filter

Here, $F_a(z)$ is the bandedge shaping filter, which is a linear phase FIR filter of even order N and $F_c(z)$ is its complementary filter. $F_c(z)$ can be easily obtained from $F_a(z)$ by subtracting the output of $F_a(z)$ from the (N/2)th delayed version of its input as shown in Figure 3, where N is the filter order. $F_{ma}(z)$ and $F_{mc}(z)$ are the two masking filters, which are used to eliminate the undesired bands in the band edge shaping and the complementary filters respectively. $F_a(z^L)$ and $F_c(z^L)$ are the interpolated filters of $F_a(z)$ and $F_c(z)$ respectively and their transition bandwidths are L times smaller. The frequency responses of the different sub filters of the FRM filter are shown in Figure 4 [5].

Here, $|H(e^{j\omega})|$ is the required filter response with sharp bandwidth. The transition band of the prototype filter is determined by $F_a(z)$ or $F_c(z)$ and results in two different realizations, case A and case B respectively[16]. Figure 4.c shows case A realization in which the transition band is determined by $F_a(z)$. For case B realization, the transition band is determined by the complementary band edge shaping filter $F_c(z)$. In this paper, the conventional FRM filter design proposed by Y.C. Lim in [5] is used. The interpolation factor L is related to the number of channels M by L = (4m + 1)M for case A design and L = (4m - 1)M for case B design [14].

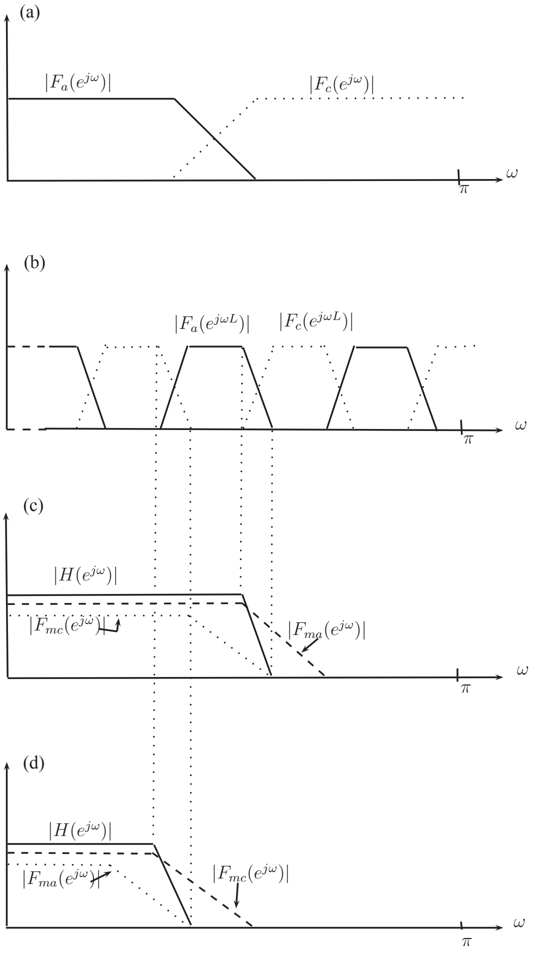

Fig. 4. Frequency Responses of subfilters in FRM filter

4. CANONIC SIGNED DIGIT REPRESENTATION

The multiplications in the digital filters can be implemented using shift and add operations. The number of shifters and adders depends on the number of 1's in the binary multiplier. The number of shifters and adders can be reduced by CSD representations, which contain a minimum number of nonzeros [2].
CSD representation is a radix two number system which uses the digit set $\{0, 1, -1\}$. For a given binary number, the corresponding CSD representation will be unique and they have the property that the adjacent digits are never both non-zero. For an n bit 2's complement number, there can be n non-zero digits, but for a CSD number there are at most n/2 non-zero digits.
Digital filter coefficients can be converted to the nearest CSD representation and this can reduce the number of major operations(adders/subtractors and shifters). They allow the successful implementation of the multipliers, with less complexity. A 16 bit precision CSD representaton of the filter coefficients is performed in this work.

5. DESIGN OF CSD COEFFICIENT CMFB USING FRM APPROACH

5.1 Design of Continuous Coefficient Filter bank

In CMFB, the prototype filter is cosine modulated to obtain the analysis and synthesis filters. Aliasing between the adjacent channels is eliminated in the design and the linear phase prototype filter eliminates the phase distortion, What remains is the amplitude distortion. The total amplitude distortion depends on the degree of overlap between the adjacent filter responses. The passband and stopband edge frequencies are carefully chosen so that the adjacent filter responses intersect approximately at the 3dB level. If this condition is not satisfied, there will be amplitude distortion around the transition frequencies $(k + 1)\pi/M$ where k = 0,1,.......M - 1 [12]. In the proposed design, the prototype filter is a conventional FRM filter and the various sub filters are designed as proposed in [5] with the interpolation factor taken as L = (4m + 1)M for case A design and L = (4m - 1)M for case B design [14].

Design Specifications
Passband Ripple: $0.004dB$
Stopband Attenuation: $60dB$
Passband Edge Frequency: 0.062π
Stopband Edge Frequency: 0.06336π
Number of Channels: 8
The interpolation factor, L is taken as 24 and ω_{3dB} obtained for the above design is 0.0625π which is equal to $\frac{1}{2M}\pi$. The frequency response plot of the prototype filter is shown in Figure 5. The frequency responses of the analysis filters are shown in Figure 6. The amplitude distortion function plot and aliasing distortion function plot are given in Figure 7 and Figure 8 respectively.

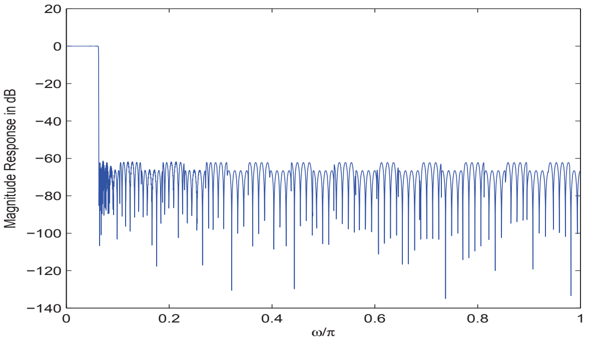

Fig. 5. Frequency response of prototype filter(Continuous Coefficients)

5.2 Design of CSD Coefficient Filter bank

The coefficients of the different subfilters in the FRM prototype filter are converted to fixed point CSD representation with restricted number of signed power of two terms. For CSD conversion, a Look Up Table is generated which consists of four fields. They are index, 16 bit CSD representation, its corresponding decimal value and number of non zero terms in the CSD representation. A typical entry of the look up table is shown in Table 1. Using the look up table method, the CSD conversion can be done fast and the filter coefficients can be rounded for fixed number of non-zeros terms. Using the minimum number of signed power of two terms without

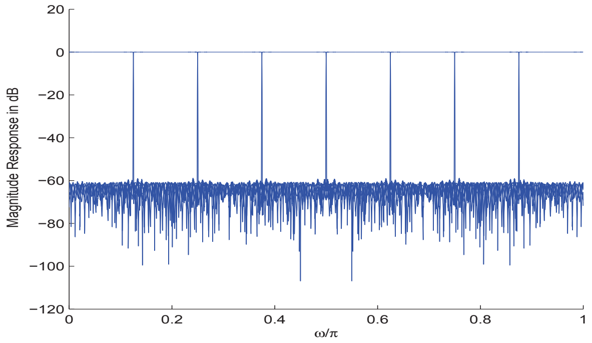

Fig. 6. Frequency response of Analysis Filters (Continuous Coefficients)

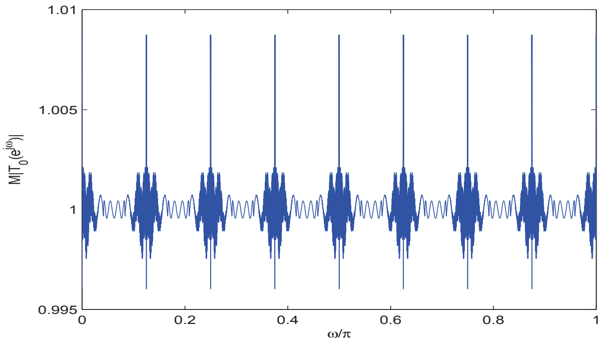

Fig. 7. Amplitude Distortion Function Plot (Continuous Coefficients)

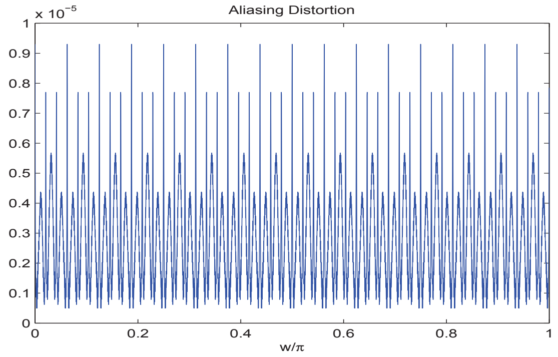

Fig. 8. Aliasing Distortion Function Plot (Continuous Coefficients)

affecting the performance, reduces the complexity further. In the look up table, 14 bits are used to represent the fractional part and 2 bits are allotted to represent the integer part. But the design of the FRM filter in the discrete space can degrade the performance.This calls for efficient non linear optimization techniques. Here, gradient based optimization techniques cannot be applied because the search space consists of integers. Hence meta heuristic algorithms are used in this work to tune the parameters properly with respect to the design problem in order to get a global solution. The frequency response of the prototype filter using CSD coefficients, the frequency responses of the analysis filters and the amplitude distortion function plot are shown in Figures 9, 10 and 11 respectively.

Table 1. A typical CSD Look Up Table Entry

Index	CSD Equivalent	Decimal Equivalent	Number of Non zeros
8814	00100010100-10-101	0.5379	6

5.2.1 Objective function formulation. The optimization goals in the proposed CMFB-FRM using CSD coefficients is to minimise the following objective functions.

$$F_1 = \max_{0<\omega<\frac{\pi}{M}} \left\{ | P_0(e^{j\omega}) |^2 + | P_0(e^{(j\omega-\frac{\pi}{M})}) |^2 - 1 \right\} \quad (9)$$

$$F_2 = \max_{\omega>\frac{\pi}{2M}} | P_0(e^{j\omega}) | \quad (10)$$

$$F_3 = \max_{0<\omega<\frac{\pi}{2M}} | P_0(e^{j\omega}) - 1 | \quad (11)$$

The objective function given by equation(9) is used to reduce the total amplitude distortion of $T_0(e^{j\omega})$. Since $T_0(e^{j\omega})$ has a period of $\frac{\pi}{M}$, the range of ω is taken as $[0, \frac{\pi}{M}]$. The objective function given by equation(10) is the maximum deviation in the stopband of the prototype filter. Good stopband attenuation results in less aliasing between the non adjacent bands. Objective function given by equation(11) is the maximum deviation in the passband of the prototype filter. The combined objective function is given as follows

$$\min \phi = \alpha_1 F_1 + \alpha_2 F_2 + \alpha_3 F_3 \quad (12)$$

where α_1, α_2 and α_3 are the trade off parameters, which defines the relative importance given to each objective function.

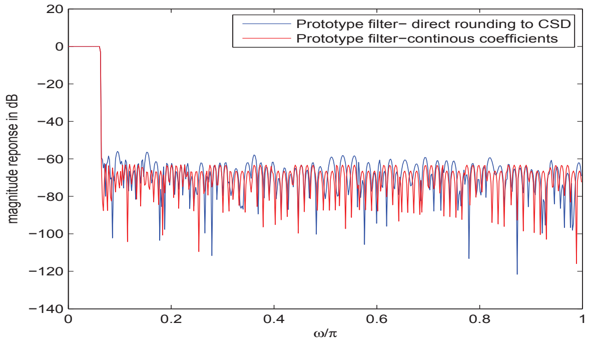

Fig. 9. Frequency response of prototype filter

6. OPTIMIZATION OF CSD ENCODED PROTOTYPE FILTER

6.1 Artificial Bee Colony (ABC) Algorithm

ABC Algorithm is a population based search technique introduced by D. Karaboga and Basturk [3]. Employed Bees, OnLooker Bees and Scout Bees constitute the artificial colony of honey bees. An

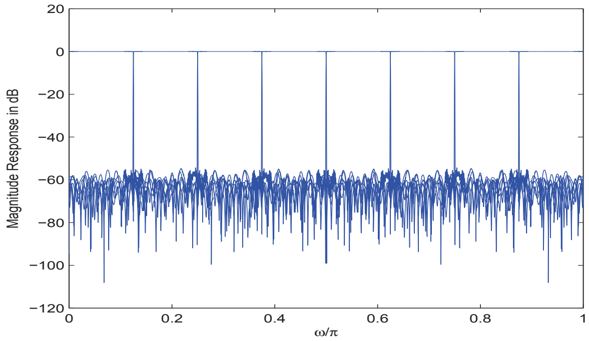

Fig. 10. Frequency response of Analysis Filters (CSD Rounded Coefficients)

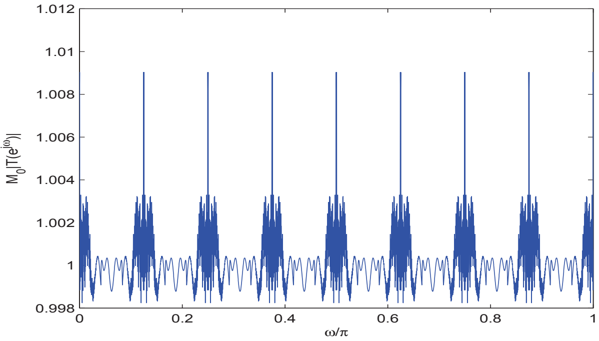

Fig. 11. Amplitude Distortion Function Plot (CSD Rounded Coefficients)

employed bee is the bee who goes to the previously visited food source. Onlooker bee is the bee waiting in the dance area for taking the decision to choose a food source. Random search is done by Scout Bees. Possible solution of the problem is represented as the food source and the corresponding fitness is the amount of the nectar of the food source.

Integer coded ABC algorithm is proposed for the design of non-uniform filter bank transmultiplexer in [8] and a modified ABC algorithm is proposed for the design of FRM lowpass filter in the CSD space in [9]. In this paper a modified integer coded ABC algorithm is proposed for the design of multiplier-less, sharp bandwidth, cosine modulated filter banks. The different phases involved in the optimization are given below [15].

6.1.1 Initialization . The filter coefficients of the different sub filters of the FRM prototype are CSD rounded and concatenated as a vector to form the initial food source. Initial random population is obtained by randomly perturbing this food source. The fitness value of each food source is evaluated and sorted according to its fitness value. N vectors with good fitness values are passed on to the next stage. The different parameters of the algorithm are initialised as given in Table 2

6.1.2 Employed Bee phase. Employed bees choose a food source within the neighbourhood of the food source in their memory. The new solution vector is formed adjacent to the existing vectors. The new food source at the i^{th} position is obtained as follows:
$v_{ij} = x_{ij} + \lfloor \phi\delta_{ij} \rfloor$

Table 2. Parameters of ABC Algorithm

Population Size	Limit Cycles	Dimension of Food Source	Maximum No: of Iterations
100	95	396	100

where ϕ is the random variable within [-1,1] and δ_{ij} is defined as $\delta_{ij} = x_{ij} - x_{kj}$
x_{ij} is the j^{th} parameter of the i^{th} food source. The newly generated food sources are prevented from crossing the boundaries of the look up table [9]. If $v_{ij} < v_{lb}$ then $v_{ij} = v_{lb}$
If $v_{ij} > v_{ub}$ then $v_{ij} = v_{ub}$
where v_{lb} and v_{ub} are the lower and upper bounds of the look up table respectively. Now the fitness value of the new vector is evaluated and if it is better, then the old vector will be replaced by the new one. This is called greedy selection mechanism.

6.1.3 Onlooker Bee phase. Onlooker bees take the information provided by the employed bees regarding the fitness function. Onlooker bee selects the food source based on the fitness function. The probabilty with which the onlooker bee chooses the food source is given by [9]

$$\frac{fit_i}{\sum_{j=1}^{N} fit_i} \tag{13}$$

where fit_i is the fitness function of the i^{th} food source and N is the total number of food sources. As a result, the food source with high fitness value will get more onlookers. Like the employed bees, the onlooker bees also search for better food source in the neighbourhood of the current food source. Similar to the employed bee phase, a greedy selection mechanism is done to select the new food source.

6.1.4 Scout Bee phase. If the nectar quality of a food source is not improved after a certain number of iterations called the limit cycles, it is abandoned. The employed bee associated with the abandoned food source becomes a scout. The scout bee randomly finds a food source as given below.
v = randi([lb,ub],'dim')
where randi denotes the random integer values from the uniform discrete distribution within the interval [lb,ub] with the dimension of the food source specified by 'dim'.

6.1.5 Termination. Termination is achieved after a maximum number of iterations is reached, otherwise steps *6.1.2* to *6.1.4* are repeated. After the termination condition is satisfied, the food source with the best nectar quality is decoded using the look up table and the optimal filter coefficients are obtained. The frequency response of the optimum filter is shown in Figure 12 and the amplitude distortion plot is given in Figure 13

6.2 Harmony Search Algorithm(HSA)

HSA is a population based random search technique introduced by Z.W.Geem [4]. The algorithm is developed inspired by the music improvisation scheme[17]. The decision variables are represented as musicians and solutions are represented as harmonics. Aesthetics is equal to the fitness function and the pitch range denotes the range of values of the optimization variables.

A harmony Memory(HM) is initialized, in which the solution variables resemble different musical notes. Musicians improve the harmonies for getting better aesthetics. Similarly the harmony search algorithm explores the search space for finding the candidate so-

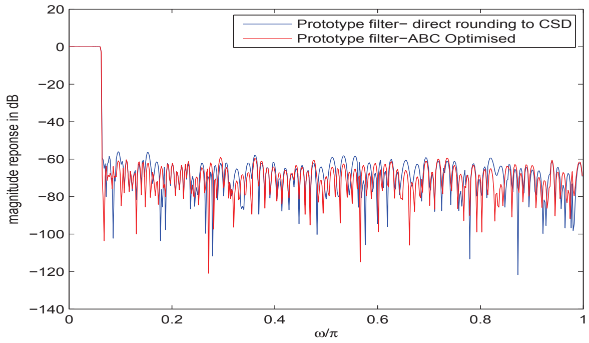

Fig. 12. Frequency response of prototype filter optimized using ABC

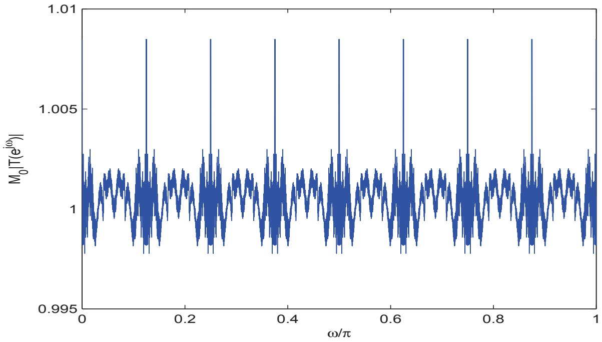

Fig. 13. Amplitude Distortion Plot (ABC)

lutions with good fitness value. In this algorithm a new solution is formed by the following three rules [10].

(1) Memory Consideration: Selects any one value from the harmony memory.

(2) Pitch adjustment: Selects an adjacent value from harmony memory.

(3) Random Selection: Selects a random value from the possible range.

The various phases involved in HS algorithm are explained below [10].

6.2.1 Initialization. The harmony search algorithm is controlled using the parameters namely, Harmony Memory Size(HMS), Harmony Memory Considering Rate(HMCR) and Pitch Adjusting Rate(PAR)and they are initialized as given in Table 3. By perturbing the initial solution or initial harmony vector, various solutions are obtained. The initial number of harmony memory locations are taken to be an integer multiple of the number of memory locations(HMS). In this problem, a harmony vector in the harmony memory corresponds to the filter coefficients of all the sub filters of the FRM filter in the CSD encoded form. The fitness function of each vector is evaluated and the best solutions are passed on to the subsequent stages of optimization.

6.2.2 Harmony Improvisation. A new harmony vector is generated from the harmony memory as follows

Table 3. Parameters of HSA Algorithm

Harmony Memory Size(HMS)	Harmony Considering Rate(HMCR)	Pitch Adjusting Rate(PAR)	Maximum No: of Iterations
50	0.3	0.01	100

Memory consideration. Select the value of the i^{th} element in the harmony vector in the harmony memory with a probability HMCR.

Pitch adjustment. Pitch adjustment is done with probability given in PAR as given below
$x_i^{new} = x_i + \lfloor rand(1,-1)FW(i) \rfloor$ $FW(i)$ is an arbitrary distance band width for the i^{th} design variable and $rand(1,-1)$ is a uniformly distributed random number between -1 and 1.

Random Selection. Generate random elements for the harmony vector with a probability [1 - HMCR].

6.2.3 Memory Updates. The fitness function of the new harmony vector is evaluated and if it is found better, then the worst harmony vector is replaced with the new vector.

6.2.4 Termination. Termination is reached when the specified number of iterations are reached, otherwise steps *6.2.2* and *6.2.3* are repeated. The frequency response of the prototype filter is shown in Figure 14 and the amplitude distortion plot is shown in Figure 15.

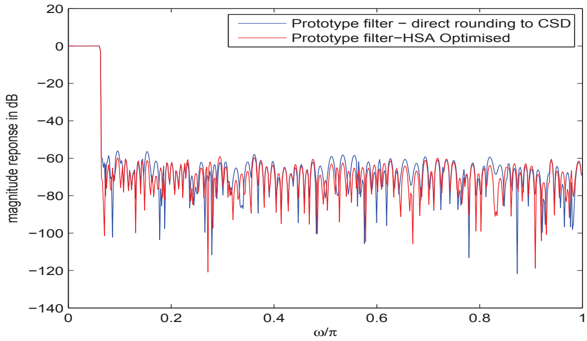

Fig. 14. Frequency response of prototype filter optimized using HSA

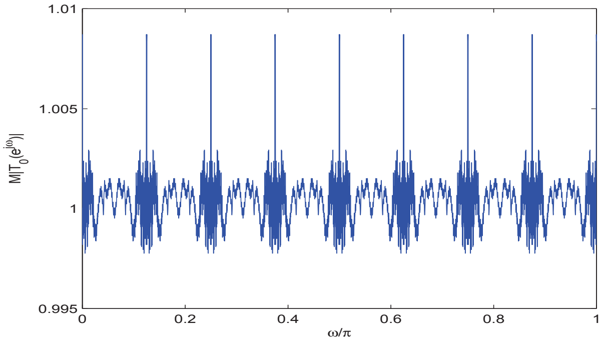

Fig. 15. Amplitude Distortion Plot (HSA)

6.3 Gravitational Search Algorithm (GSA)

GSA is a population based heuristic algorithm proposed by Rashedi in 2009 [13]. GSA is based on Newtonian law of gravity and motion [13]. A modified GSA algorithm for the design of 2D sharp wideband filter is proposed in [11]. GSA can be considered as an artificial world of masses, where every mass represents a solution to the problem. Each mass has four specifications: position, inertial mass, active gravitational mass and passive gravitational mass. The position of mass is equivalent to the solution and the corresponding gravitational and inertial masses are determined by the fitness function. Masses attract each other by force of gravity and the masses will be attracted by the heaviest mass which gives an optimum solution. The positions of the masses are updated in each iteration. The different phases involved in the GSA optimization are explained below [11].

6.3.1 Initialization. A mass or agent is formed by concatenating the CSD encoded filter coefficients of the sub-filters of the FRM prototype filter. Let N be the total number of agents or masses. Intial population is obtained by randomly perturbing the CSD encoded filter coefficients. The different parameters of the algorithm: number of masses, gravitational constant G_0, user specified constant α and maximum number of iterations (MNI) are initialised as given in Table 4.

Table 4. Parameters of GSA Algorithm

Number of Agents	Gravitational constant	Constant α	Maximum No: of Iterations
100	100	20	100

6.3.2 Fitness Evaluation. The fitness of all agents in each iteration is evaluated and the best and worst fitnesses are found at each iteration as follows.

$$worst(t) = \max_{j\epsilon 1,2...N} fit_j(t) \tag{14}$$

$$best(t) = \min_{j\epsilon 1,2...N} fit_j(t) \tag{15}$$

where $fit_j(t)$ represents the fitness value of the agent i at time t.

6.3.3 Compute the different parameters. The gravitational and inertial masses of each agent are calculated using the following equations

$$M_{ai} = M_{pi} = M_{ii} = M_i \tag{16}$$

$$i = 1, 2.....N$$

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \tag{17}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{i=1} N m_i(t)} \tag{18}$$

where M_{ai},M_{pi} and M_{ii} represents the active gravitational mass, passive gravitational mass and inertial mass respectively of the i^{th} agent.
Gravitational constant at each iteration t is computed by the equation (19)

$$G(t) = G_0 e^{-\alpha t/T} \tag{19}$$

where T is the total number of iterations.

6.3.4 Calculate Acceleration of agents. $F_{ij}^d(t)$ is the force acting on the mass 'i' from mass 'j' at time t in the d^{th} dimension

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t)M_{ai}(t)}{R_{ij}(t) + \varepsilon}(x_i^d(t) - x_j^d(t)) \tag{20}$$

$R_{ij}(t)$ is the Euclidean distance between two agents i and j, ε is a small constant
The total force acting on an agent 'i' in a dimension of d is given as

$$F_i^d(t) = \sum_{j=1, j\neq i} N rand_j F_{ij}^d(t) \tag{21}$$

$rand_j$ is a random number in the interval [0,1]. The total force is expressed as a randomly weighted sum of the d^{th} components of the forces exerted from other agents.
The acceleration of the i^{th} agent at time t in the d^{th} dimension is given by

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \tag{22}$$

where $M_{ii}(t)$ is the inertial mass

6.3.5 Update the velocity and position of agents. The velocity of the agent in the next iteration is represented as a fraction of its current velocity added to its acceleration. The new position and velocity are calculated as

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \tag{23}$$

$$x_i^d(t+1) = \lfloor x_i^d(t) + v_i^d(t+1) \rfloor \tag{24}$$

The new positions are prevented from crossing the boundaries of the look up table . If $v_{ij} < v_{lb}$, then $v_{ij} = v_{lb}$
If $v_{ij} > v_{ub}$, then $v_{ij} = v_{ub}$
where v_{lb} and v_{ub} are the lower and upper bounds of the look up table respectively.

6.3.6 Termination. The program will be terminated when the maximum number of iterations is reached, otherwise steps *6.3.2* to *6.3.4* will be repeated. The magnitude response of the prototype filter is shown in Figure 16 and the amplitude distortion is shown in Figure 17

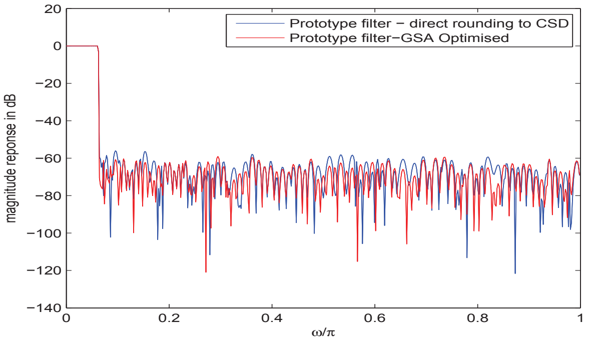

Fig. 16. Frequency response of prototype filter optimized using GSA

Table 5. Performance Comparison

	Continuous Coefficients	CSD Rounded Coefficients	ABC	HSA	GSA
Error in amplitude distortion function	0.0082	0.009	0.0084	0.0087	0.009
Worst Aliasing Distortion	5.09×10^{-5}	1.08×10^{-5}	1.18×10^{-5}	1.09×10^{-5}	1.17×10^{-5}
Max. Stopband Attenuation(dB)	62.9	56.004	59.04	59.039	59.04
Minimum Passband Ripple(dB)	0.008	0.012	0.0076	0.0083	0.009
Run Time(sec.)			197.22	440.1	195.07
Average of final objective function value			1.0044	1.0046	1.0052
No: of multipliers	283				
No: of coefficient adders			282	282	282
Adders due to SPT terms			464	465	460
Total no: of adders			746	747	742

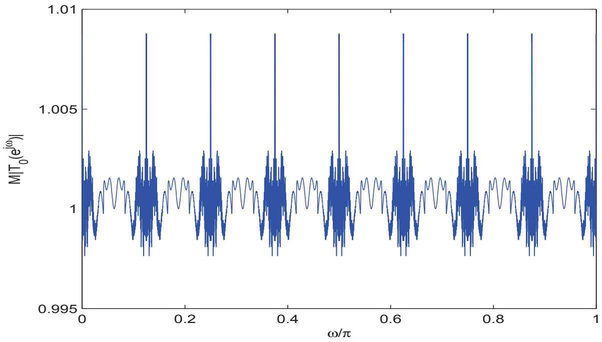

Fig. 17. Amplitude Distortion Plot (GSA)

7. SIMULATION RESULTS AND DISCUSSIONS

All the simulations are performed on a Dual Core AMD Opteron processor operating at 2.17GHz, using MATLAB 7.12.0.

For the specifications given in section 5.1, a cosine modulated filter bank with sharp transition bandwidth is designed. The coefficients of the different sub filters F_a, F_{ma} and F_{mc} of the prototype filter are converted to finite precision canonic signed digit representation using a 16 bit look up table. For the joint optimization of different subfilters, the coefficients of F_a, F_{ma} and F_{mc} are concatenated together and is taken as the design vector for the optimization problem. Since all the subfilters have linear phase, only half of the symmetrical coefficients of each subfilter need to be extracted. This reduces the computation time and the dimensionality. The CSD coefficient prototype filter is further optimized using different modified meta heuristic algorithms. The performances of the CSD coefficient filter bank which is optimized using different algorithms are shown in Table 5.

From the Table, it can be seen that ABC algorithm is more suited for this filter bank. All the three algorithms are run for 100 iteration. For 100 iteration, HSA algorithm takes double the time compared to ABC algorithm. ABC and GSA take almost equal time for 100 iterations, but ABC algorithm gives better performance of the filter bank in terms of amplitude and aliasing distortion, minimum passband ripple and maximum stop band attenuation. The objective function is calculated as given in equation (12) and is found to be optimum for ABC algorithm.

8. CONCLUSION

In this paper, an approximate reconstruction totally multiplierless cosine modulated filter bank with sharp transition bandwidth using FRM approach is proposed. The coefficients are synthesized using canonic signed digit representation which are optimized using different meta heuristic algorithms. A continuous coefficient eight channel cosine modulated sharp transition filter bank is initially designed. The filter bank is optimized in the canonic signed digit space. The performance of the CSD coefficient filter bank for different modified optimization algorithms are compared. ABC algorithm is observed to be the most suited one to obtain the multiplierless cosine modulated filter bank.

9. REFERENCES

[1] M.B. Furtado Jr, P.S.R. Diniz, S.L. Netto, and T. Saramaki. On the Design of high-complexity Cosine-Modulated Transmultiplexers based on the Frequency-Response Masking approach. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 52(11):2413–2426, 2005.

[2] Richard I Hartley. Subexpression Sharing in Filters using Canonic Signed Digit Multipliers. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 43(10):677–688, 1996.

[3] Dervis Karaboga and Bahriye Basturk. On the Performance of Artificial Bee Colony (ABC) Algorithm. *Applied Soft Computing*, 8(1):687–697, 2008.

[4] K.S. Lee and Z.W. Geem. A new Structural Optimization Method based on the Harmony Search Algorithm. *Computers & Structures*, 82(9):781–798, 2004.

[5] Y. C. Lim. Frequency-Response Masking Approach for the synthesis of Sharp Linear Phase Digital Filters. *Circuits and Systems, IEEE Transactions on*, 33(4):357–364, 1986.

[6] Yong Ching Lim, Rui Yang, Dongning Li, and Jianjian Song. Signed Power-of-two term Allocation Scheme for the Design

of Digital Filters. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 46(5):577–584, 1999.

[7] Yuan-Pei Lin and PP Vaidyanathan. A Kaiser Window Approach for the Design of Prototype Filters of Cosine Modulated Filter Banks. *Signal Processing Letters, IEEE*, 5(6):132–134, 1998.

[8] V. J Manoj and E. Elias. Artificial Bee Colony Algorithm for the Design of Multiplier-less Non Uniform Filter Bank Transmultiplexer. *Information Sciences*, 192:193–203, 2012.

[9] M. Manuel and E. Elias. Design of Frequency-Response Masking FIR filter in the Canonic Signed Digit Space using modified Artificial Bee Colony Algorithm. *Engineering Applications of Artificial Intelligence*, 2012.

[10] M. Manuel and E. Elias. Design of Sharp 2D Multiplier-less Circularly Symmetric FIR Filter using Harmony Search Algorithm and Frequency Transformation. *Journal of Signal and Information Processing*, 3(3):344–351, 2012.

[11] M. Manuel, Remya Krishnan, and E. Elias. Design of Multiplierless 2-D Sharp Wideband Filters using FRM and GSA. *Global Journal of Research Engineering*, 12(5-F), 2012.

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

[13] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi. GSA: A Gravitational Search Algorithm. *Information Sciences*, 179(13):2232–2248, 2009.

[14] L. Rosenbaum, P. Lowenborg, and M. Johansson. An Approach for Synthesis of Modulated M-Channel FIR Filter Banks utilizing the Frequency-Response Masking Technique. *EURASIP Journal on Advances in Signal Processing*, 2007, 2006.

[15] Bindiya T.S. and Elizabeth Elias. Article: Design of multiplier-less reconfigurable non-uniform channel filters using meta-heuristic algorithms. *International Journal of Computer Applications*, 59(11):1–11, December 2012. Published by Foundation of Computer Science, New York, USA.

[16] Ronghuan Yang, Bede Liu, and Yong Ching Lim. A new Structure of Sharp Transition FIR Filters using Frequency-Response Masking. *Circuits and Systems, IEEE Transactions on*, 35(8):955–966, 1988.

[17] Xin-She Yang. *Music-inspired harmony search algorithm*. Springer, 2009.

[18] Xin-She Yang. *Nature-Inspired Metaheuristic Algorithms*. Luniver Press, 2011.

[19] Ya Jun Yu and Yong Ching Lim. Genetic Algorithm Approach for the Optimization of Multiplierless Sub-filters generated by the Frequency-Response Masking Technique. In *Electronics, Circuits and Systems, 2002. 9th International Conference on*, volume 3, pages 1163–1166. IEEE, 2002.